AUTOPILOT AND GUIDANCE ALGORITHMS FOR INFRARED GUIDED MISSILES

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

BY

KAYHAN ÇAĞLAR KILIÇ

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

DECEMBER 2006

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Kemal LEBLEBİCİOĞLU
Co-Supervisor

Prof. Dr. Zafer ÜNVER
Supervisor

Examining Committee Members

Prof. Dr. Uğur HALICI (METU, EE) _____

Prof. Dr. Zafer ÜNVER (METU, EE) _____

Prof. Dr. Kemal LEBLEBİCİOĞLU (METU, EE) _____

Assist.Prof.Dr. Afşar SARANLI (METU, EE) _____

M. Sc. Ali Rıza GÜVENLİK (ASELSAN) _____

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:
Kayhan Çağlar KILIÇ

Signature:

ABSTRACT

AUTOPILOT AND GUIDANCE ALGORITHMS
FOR IR GUIDED MISSILES

KILIÇ, Kayhan Çağlar
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Zafer Ünver
Co-Supervisor: Prof. Dr. Kemal Leblebicioğlu

December 2006, 129 pages

Guided missiles are among the most effective threats against air platforms. Aircraft and helicopter losses in the last decades were mostly due to guided missiles, 70% of which were infrared guided missiles. Today, there are as many as 500,000 shoulder-fired missiles in military arsenals around the world, whose guidance algorithms enable them to track the desired trajectories very precisely.

In this thesis, main focus is on defining infrared missile guidance and control algorithms in order to study on various target-missile scenarios for the effectiveness of these algorithms. First, a mathematical model of a generic missile is given. Then the flight control system of the missile is created by using LQR and PID controllers. Different kinds of PNG algorithms applied to an infrared missile are presented. The seeker part of the infrared missile is also discussed. The effectiveness of guidance algorithms are studied based on different target – missile scenarios and the responses of them to IR countermeasures are also observed.

This study shows that different guidance algorithms can be used for different scenarios. If suitable algorithms are combined and suitable constants are applied, the guided missile can track the target very precisely. In addition, the seeker part has to be improved with tracking algorithms in order to recognize IR-countermeasures and not to follow them.

Keywords: Missile, Infrared Guidance, Infrared Seeker, PNG, Flare

ÖZ

KIZILÖTESİ GÜDÜMLÜ FÜZELER İÇİN
OTOPİLOT VE GÜDÜM ALGORİTMALARI

KILIÇ, Kayhan Çağlar
Yüksek Lisans., Elektrik-Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Zafer Ünver
Ortak Tez Yöneticisi: Prof. Dr. Kemal Leblebicioğlu

Aralık 2006, 129 sayfa

Güdümlü füzeler, hava platformlarına karşı kullanılan en etkin tehditlerden biridir. Son yıllardaki uçak ve helikopter kayıplarının çoğuna, yüzde 70'i kızıl ötesi güdümlü olan füzeler sebep olmuştur. Bugün dünya genelindeki askeri cephanelerde, 500,000 kadar omuzdan atılabilir füze bulunmaktadır Bu füzelerin güdüm algoritmaları, belirtilen rotaların çok düşük hata oranları ile izlenebilmesini sağlamaktadır.

Bu tezde temel olarak, çeşitli hedef-füze senaryoları içinde güdüm algoritmalarının etkinliklerinin üzerinde çalışmak için kızılötesi füze güdüm ve kontrol algoritmaları tanımlamaya odaklanılmıştır. İlk olarak genel bir füze için matematiksel model verilmiştir. Ardından uçuş kontrol sistemi, LQR ve PID kontrol mekanizmaları kullanılarak yaratılmıştır. Kızılötesi füze için uygulanan farklı PNG algoritmaları sunulmuştur. Kızılötesi füzenin arayıcı kısmı da ayrıca tartışılmıştır. Güdüm algoritmalarının etkinlikleri, farklı hedef-füze tabanlı senaryolar bazında incelenmiştir ve bunların kızılötesi karşıtedbirlere karşı tepkileri de ayrıca gözlenmiştir.

Bu çalışma, farklı güdüm algoritmalarının değişik senaryolarda kullanılabileceğini göstermiştir. Uygun algoritmaların birleştirilmesi ve uygun katsayıların kullanılmasıyla güdümlü füze hedefini hassas şekilde takip edebilmektedir. Buna ek olarak, kızılötesi karşı tedbirlerin füze tarafından tanınıp takip edilmemesi için füze arayıcı kısmının takip algoritmaları ile geliştirilmesi gerekmektedir.

Anahtar Kelimeler: Füze, Kızılötesi Güdüm, Kızılötesi Arayıcı, PNG, Ateştopu

ACKNOWLEDGMENT

I would like to express my appreciation to my supervisors Prof. Dr. Zafer ÜNVER and Prof. Dr. Kemal LEBLEBİCİOĞLU for their guidance, advice, criticism, encouragement and insight throughout the research.

Additionally, I thank Mrs. Yelda ALTUĞ for her guidance, valuable support, endless patience and tolerance throughout this thesis work.

I also thank to Özden ERDEM and my family for their encouragements and support through the study.

I would like to thank to ASELSAN INC. for facilities provided for the completion of this thesis.

TABLE OF CONTENTS

PLAGIARISM iii
ABSTRACT iv
ÖZ vi
ACKNOWLEDGMENT viii
TABLE OF CONTENTS ix
LIST OF TABLES xi
LIST OF FIGURES xii
LIST OF SYMBOLS xv
1. INTRODUCTION 1
2. MATHEMATICAL MODEL AND AUTOPILOT DESIGN 5
2.1. Introduction 5
2.2. Mathematical Model of a Missile 5
2.3. PID Design 6
2.3.1. Autopilot Design Using Target Position Information as Input 9
2.3.2. Autopilot Design Using Target – Missile Angle Information as Input 12
2.4. LQR Design 15
2.4.1. Tracking Problem with LQR 17
2.4.2. Autopilot Design Using Target Position Information as Input 18
2.4.3. Autopilot Design Using Target – Missile Angle Information as Input 24
3. GUIDANCE DESIGN 29
3.1. Introduction 29
3.2. Seeker Model 29
3.2.1. Error-Angle Estimation 31
3.2.2. Spot Location Estimation 32
3.2.3. Target Area Estimation 35
3.3. Guidance Models 35
3.3.1. Classical PNG 36
3.3.2. Modified Classical PNG 41
3.3.3. Gain Scheduled PNG 41
3.3.4. Augmented Proportional Navigation 42
3.3.5. Modified Augmented Proportional Navigation 43
3.3.6. Ideal PNG 43
3.3.7. Summary of the PNG Laws 45
3.4. Scenarios 46
3.4.1. Static Target 46
3.4.2. Slow Moving Target 48
3.4.3. Fast Moving Target 58

3.4.4.Slow S Movement in Both Planes 66
3.4.5.Fast S Movement in Both Planes 68
3.4.6.Circular Movement 69
3.4.7.Target is Dispensing IR Countermeasures 70
3.5. Conclusions 81

4. CONCLUSIONS 82

REFERENCES 84

APPENDIX 86

A.1.MATHEMATICAL MODELING 86

A.1.1. Coordinate Systems and Transformation Matrices 86
A.1.2. Equations of a Missile 90
A.1.3. Forces and Moments Acting on a Missile 96
A.1.4. Summary of Missile Equations 100
A.1.5. Linearization of the Equations 102

A.2.MISSILE DATCOM USER MANUAL 106

A.2.1. Input File (for005.dat) 106
A.2.2. Output File (for006.dat) 124
A.2.3. Automated Processing Recommendations for the Output File 126

LIST OF TABLES

Table 1 Effects of Kp, Kd, and Ki on a closed-loop system 8
Table 2 Gain Scheduled PNG 41
Table 3 Summary of the PNG Laws 45
Table 4 Performance Scores for the Slow Moving Target 57
Table 5 Performance Scores for the Fast Moving Target 66
Table 6 Performance Score for the Slow S Movement 67
Table 7 Performance Score for Fast S Movement 69
Table 8 Performance Score for the Circular Movement 70
Table 9 Control Card Inputs 107
Table 10 Reference Quantity Inputs 111
Table 11 Additional Flight Condition Inputs 113
Table 12 Axisymmetric Body Inputs for Option 1 114
Table 13 Axisymmetric Body Inputs for Option 2 117
Table 14 General Properties of Infrared Guided Surface to Air Missiles 123
Table 15 Output Files Contents 124
Table 16 Standard output for the file for005.dat 127

LIST OF FIGURES

Figure 1 Guidance and Control System..... 2
Figure 2 Structure of a PID System..... 7
Figure 3 Schematics for a Closed Loop Control System 7
Figure 4 Reference (Blue) and Tracked (Red) Trajectories in 3D Plot..... 9
Figure 5 Longitudinal Autopilot Architecture with PIDs..... 10
Figure 6 Target's Altitude Position (Blue) and Response of the Missile (Red)..... 10
Figure 7 Lateral Autopilot Architecture with PIDs..... 11
Figure 8 Target's Lateral Position (Blue) and Response of the Missile (Red) 11
Figure 9 Longitudinal Autopilot Architecture with PIDs where the Pitch Angle is the Input..... 12
Figure 10 Actual (blue) and Tracked (red) Pitch Angle for 13
Figure 11 Lateral Autopilot Architecture with PIDs where 13
Figure 12 Actual (blue) and Tracked (red) Pitch Angles for Lateral Autopilot Design..... 14
Figure 13 Actual (blue) and Tracked (red) Pitch Angles for Lateral Autopilot Design..... 14
Figure 14 Plant and Feedback Controller for Linear Regulator Problems..... 16
Figure 15 Altitude Tracking (Reference is Blue and Tracked is Red)..... 19
Figure 16 Altitude Tracking (Zoomed) (Reference is Blue and Tracked is Red).... 20
Figure 17 Longitudinal (x-z) Movement in Simulink……..21
Figure 18 Lateral (x-y) Movement in Simulink……...…22
Figure 19 Lateral Movement Tracking (Reference is Blue and Tracked is Red) 23
Figure 20 Lateral Movement Tracking (Zoomed)..... 23
Figure 21 Lateral (x-y) Movement in Simulink……..25
Figure 22 Yaw Error Angle Tracking (Reference is Blue and Tracked is Red) 26
Figure 23 Yaw Error Angle Tracking (Reference is Blue and Tracked is Red) 27
Figure 24 Lateral (x – y) Movement in Simulink……28
Figure 25 Structure of the System……..…… ……30
Figure 26 Pitch Error Angle 31
Figure 27 Direct Spot Location Estimation..... 32
Figure 28 Spot Location Estimation..... 34
Figure 29 Four Quadrant Detector. Target is: 35
Figure 30 Classical PNG in 2D Space..... 37
Figure 31 Pure PNG 39
Figure 32 True PNG 40
Figure 33 Classical PNG for Static Target in 3D 46
Figure 34 Classical PNG for Static Target in x – z 46
Figure 35 Classical PNG for Static Target in x – y 47
Figure 36 Modified Classical PNG for Static Target in x – y 47
Figure 37 Modified Classical PNG for Static Target in x – z 48
Figure 38 Modified Classical PNG for Static Target in 3D 48
Figure 39 Classical PNG for Slow Moving Target in 3D 49
Figure 40 Classical PNG for Slow Moving Target in x – z 49

Figure 41 Classical PNG for Slow Moving Target in x – y 50
Figure 42 Modified Classical PNG for Slow Moving Target in x – z 50
Figure 43 Modified Classical PNG for Slow Moving Target in 3D 51
Figure 44 Modified Classical PNG for Slow Moving Target in x – y 51
Figure 45 Gain Scheduled PNG for Slow Moving Target in x – y 52
Figure 46 Gain Scheduled PNG for Slow Moving Target in x – z 52
Figure 47 Gain Scheduled PNG for Slow Moving Target in 3D 53
Figure 48 Augmented PNG for Slow Moving Target in x – z 53
Figure 49 Augmented PNG for Slow Moving Target in x – y 54
Figure 50 Augmented PNG for Slow Moving Target in 3D 54
Figure 51 Modified Augmented PNG for Slow Moving Target in x – y 55
Figure 52 Modified Augmented PNG for Slow Moving Target in x – z 55
Figure 53 Modified Augmented PNG for Slow Moving Target in 3D 56
Figure 54 Performance Analysis 57
Figure 55 Classical PNG for Fast Moving Target in 3D 58
Figure 56 Classical PNG for Fast Moving Target in x – y 59
Figure 57 Classical PNG for Fast Moving Target in x – z 59
Figure 58 Modified Classical PNG for Fast Moving Target in x – z 60
Figure 59 Modified Classical PNG for Fast Moving Target in x – y 60
Figure 60 Modified Classical PNG for Fast Moving Target in 3D 61
Figure 61 Gain Scheduled PNG for Fast Moving Target in x – y 61
Figure 62 Gain Scheduled PNG for Fast Moving Target in x – z 62
Figure 63 Gain Scheduled PNG for Fast Moving Target in 3D 62
Figure 64 Augmented PNG for Fast Moving Target in x – y 63
Figure 65 Augmented PNG for Fast Moving Target in x – z 63
Figure 66 Augmented PNG for Fast Moving Target in 3D 64
Figure 67 Modified Augmented PNG for Fast Moving Target in x – y 64
Figure 68 Modified Augmented PNG for Fast Moving Target in x – z 65
Figure 69 Modified Augmented PNG for Fast Moving Target in 3D 65
Figure 70 Slow S Movement (Target is Blue and the Missile is Red) 67
Figure 71 Fast S Movement (target is blue and the missile is red) 68
Figure 72 Circular Movement of the Target in x – y 69
Figure 73 Active IR Countermeasure Example – ALQ-144 [20] 71
Figure 74 Active Countermeasure Example 2 – DIRCM [20] 71
Figure 75 Flares Ejected by a C-130 Hercules Military Transport [21] 72
Figure 76 Spectral Specification of a Flare [3] 72
Figure 77 Main Structure of a Flare [3] 73
Figure 78 Flare Trajectory Analysis 73
Figure 79 Target is Dispensing a Flare From Left – Modified Classical PNG (3D) 74
Figure 80 No Flare – Modified Classical PNG (3D) 74
Figure 81 Target is Dispensing a Flare From Left – Modified Classical PNG (x – y) 75
Figure 82 No flare – Modified Classical PNG (x – y) 75
Figure 83 Target is Dispensing a Flare From Left – Modified Classical PNG (x – z) 76
Figure 84 No flare – Modified Classical PNG (x – z) 76
Figure 85 Dispensing a Flare From Left – Modified Augmented PNG (x – z) 77
Figure 86 Dispensing a Flare From Left – Modified Augmented PNG (x – y) 77

Figure 87 Dispensing a Flare From Left – Modified Augmented PNG (3D) 78
Figure 88 Dispensing a Flare From Right – Modified Augmented PNG (x – y)..... 78
Figure 89 Dispensing a Flare From Right – Modified Augmented PNG (x – z)..... 79
Figure 90 Dispensing a Flare From Right – Modified Augmented PNG (3D)..... 79
Figure 91 Dispensing Two Flares From Right – Modified Augmented PNG (x – y)..... 80
Figure 92 Dispensing Two Flares From Right – Modified Augmented PNG (x – z)..... 80
Figure 93 Dispensing Two Flares From Right – Modified Augmented PNG (3D)..... 81
Figure 94 Six Degrees of Freedom [15]..... 87
Figure 95 Missile Body and Earth Axes..... 88
Figure 96 (a) Rotation in x, y, z Order (b) Rotation in z, y, x Order..... 89
Figure 97 Angle of Attack Description [16]..... 97
Figure 98 Angle of Sideslip Description [16]..... 98
Figure 99 Missile Model..... 110
Figure 100 A Typical Input for the Reference Quantity Input..... 112
Figure 101 Asymmetric Body Structure..... 116
Figure 102 A Typical Input for the Axisymmetric Body Input..... 117
Figure 103 Finset Structure I..... 119
Figure 104 Finset Structure II..... 119
Figure 105 Finset Structure III..... 120
Figure 106 A Typical Finset Input..... 121
Figure 107 Tail Finset Structure..... 121
Figure 108 Generic IR Guided Missile..... 124

LIST OF SYMBOLS

$\dot{R}$	Absolute center of gravity
$\bar{q}$	Dynamic pressure
$\dot{\lambda}$	LOS angle rate
ψ	Yaw angle
Θ	Inertia tensor
θ	Pitch angle
ϕ	Roll angle
α	Angle of attack
β	Angle of sideslip
γ	The specific heat ratio of the air
ρ_o	Air density at sea level
a_{M1}, a_{M2}	Missile acceleration components
C	Speed of the sound
C_l	Rolling moment coefficient
C_m	Pitching moment coefficient
C_n	Yawing moment coefficient
Cx	Axial force coefficient
C_y	Side force coefficient
C_z	Normal force coefficient
d_{11}, d_{12}, etc.	Direction cosines
F	Force
F_{ax}, F_{ay}, F_{az}	Forces due to aerodynamic effects on the missile flight
F_X, F_Y, F_Z	Components of the net force acting on the missile in the body fixed reference frame
g_x, g_y, g_z	The body frame components of the gravitational acceleration
H	Angular momentum
H	Altitude
HE	Heading angle of the missile

i, j, k	Unit vectors
I_{xx}, I_{yy}, I_{zz}	Moment of inertia
I_{xy}, I_{xz}, I_{yz}	Product of inertia
J	Performance index
K	Feedback gains in LQR
K_3	LOS angle constant
K_D	Derivative gain in the PID
K_I	Integral gain in the PID
K_p	Proportional gain in the PID
L	Linear momentum
l	Missile length
L, M, N	Moments due to aerodynamic effects on the missile flight
L_T, M_T, N_T	Components of the thrust moment
M	External moment
m	Mass of the missile
M	Mach number
M_L, M_M, M_N	Components of the net moment acting on the missile in the body fixed reference frame
N’	Navigation constant
n_c	Missile acceleration value created by the guidance system
n_T	Target acceleration
p, q, r	Components of angular velocity in the body fixed reference frame with respect to the earth fixed reference frame
Q, R	Weighting matrices
Q_d	Dynamic pressure
R	Air gas constant
R_{TM}	Target to missile distance
s	Reference area
S_1, S_2, S_3, S_4	Area values for the four quadrant detector
T	ambient temperature
T_{321}	Transformation matrix
T_x, T_y, T_z	Components of the thrust force
u	Velocity along x axis
v	Velocity along y axis
V_c	Closing velocity between target and missile

V_T	Target velocity
$V_{T1,}$ V_{T2}	Components of the velocity vector in the Earth coordinate system
w	Velocity along z axis
W	Velocity of the central of gravity relative to the body
x,y,z	Axes of body fixed reference frame
X,Y,Z	Axes of earth fixed reference frame
X_b, Y_b, Z_b	Axes of body fixed reference frame
X_{pos}, Y_{pos}	Target's estimated spot location on the seeker
γ_p	Pitch angle error
γ_r	Yaw angle error
δ_a	Effective aileron deflection
δ_e	Effective elevator deflection
δ_r	Effective rudder deflection
ρ	Air density

CHAPTER 1

INTRODUCTION

Guided missiles are among the most effective threats for air platforms. Aircraft and helicopter losses in the last decades were due to guided missiles, 70% of which were infrared guided. Today, there are as many as 500,000 shoulder-fired missiles in military arsenals around the world [1]. The popularity of the infrared missiles comes from their guidance algorithms and seeker parts, which provide tracking the desired trajectories very precisely. Effectiveness of the guidance algorithms with a constant seeker part can be analyzed by studying different target / missile scenarios.

Missile guidance system consists of a seeker part, a main guidance system and a control system (Figure 1).

The seeker part is the heart of the guidance system because the estimation of the target's position is carried out by the seeker. The seeker consists of detectors for detecting the radiant intensity of the target.

The main guidance system provides the necessary flight path dynamics in order to achieve the mission objective of the missile by using the seeker's outputs. There are many guidance techniques which are already used in literature such as command guidance, homing guidance, present guidance, and proportional navigation [2].

The control system, in summary, has the following objectives: (i) controlling the missile dynamic stability for all phases of the flight, (ii) controlling the aerodynamic loads on the missile to attain guidance objectives, (iii) decoupling the pitch and yaw control channels. The guidance commands generated throughout the flight are processed by the flight control system to stabilize the missile with respect to these commands. This stabilization is done in pitching, yawing, and rolling motions of the flight. The flight control system generates its own commands for control mechanisms, in order to properly execute the guidance commands [3].

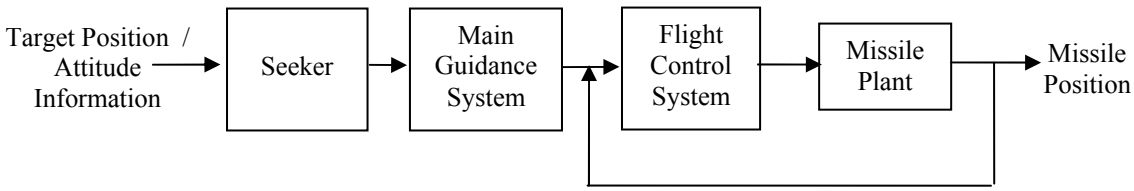


Figure 1 Guidance and Control System

Flight control system is implemented by autopilots. There are several ways of designing autopilots based on mathematical models developed for infrared guided missiles. Mathematical model of a missile is based on the moment and force equations acting on it and kinematics properties of the scenario [4]. There are lots of different mathematical models for a missile according to the control surfaces to be used. Jae Weon Choi has used rudder deflection, aileron deflection, bank angle, yaw rate, roll rate, sideslip angle, and washout filter as states and aileron and rudder commands as inputs [5]. Eliezer Y. Shapiro has used eight state variables: sideslip angle, bank angle, roll rate, yaw rate, lateral directional flight path, rudder deflection, aileron deflection, and canard deflection (Figure 94, Figure 97, and Figure 98). The three control variables are rudder command, aileron command, and canard command [6]. Mathematical model of a generic missile is composed of 12 nonlinear equations.

In this thesis, the proportional navigation guidance (PNG) algorithms are presented in detail and some simulations are performed in MATLAB environment in order to evaluate the advantages and disadvantages of these algorithms.

In Chapter 2, the mathematical model of a generic rigid missile is given. Missile DATCOM software is used for obtaining the aerodynamic coefficients. Lateral position, lateral velocity, yaw angle, roll rate, yaw rate, and roll angle are used as states for the lateral motion and longitudinal position and velocity, pitch angle and pitch rate are used as states for the longitudinal motion throughout the thesis. Then autopilot designs are applied to missile model in order to constitute the flight control system using PID (Proportional Integral Derivative) and LQR (Linear Quadratic Regulator) techniques.

In Chapter 3, an infrared seeker model is presented. Then different PNG algorithms are used for tracking a complex trajectory created for an air vehicle. All control and guidance algorithms are applied to the lateral and the longitudinal planes separately since they are decoupled. Missile's trajectory is observed for a static target, a slow moving target, a fast moving target, and a target with IR-countermeasure.

In Chapter 4, conclusions are given including the advantages and disadvantages of guidance algorithms based on the simulation results. Contributions of this study and suggestions for future works are also summarized.

In Appendix A.1, the mathematical model for a generic rigid missile is derived starting from the moment and force equations for a generic air vehicle. Linearized model, aerodynamic forces, moments, and coefficients acting on a missile are also obtained and presented.

Much effort has been put on for obtaining an improved version of the user manual for Missile DATCOM Software based on a previous version of the same manual. This revised and developed user manual is presented in Appendix A.2.

CHAPTER 2

MATHEMATICAL MODEL AND AUTOPILOT DESIGN

2.1. Introduction

Missiles can be determined mathematically by 12 nonlinear first order differential equations. Autopilots use this model to make the movement of the missile possible according to the target's trajectories and form the flight control system. Inputs for an autopilot system are determined by the seeker and guidance systems. Position coordinates of the target could be the inputs as well as the target – missile line of sight error angles. Outputs of the autopilot system are deflection angles which are the inputs for the missile's plant.

Proportional Integral Derivative (PID) and Linear Quadratic Regulator (LQR) techniques are used throughout the thesis. Both position and angle input types are presented here, but only the position input type is used for the guidance system.

2.2. Mathematical Model of a Missile

The mathematical model of a missile with six degrees of freedom can be described by 12 nonlinear first order differential equations. These equations can be analyzed in two groups: kinematics equations, which are the consequences of transformation matrix applications that constitute a relationship between the reference axis systems through the use of Euler angles; and dynamic equations, which are derived by applying Newton's laws of motion that relate the summation of the external forces and moments to the linear and angular accelerations of the body [7] [8]. Six degrees of freedom (6DOF) are longitudinal, lateral, vertical movements and attitude of the motion which are presented in Figure 94.

The 12 nonlinear first order differential equations are linearized under the assumptions described in Appendix A.1.5. The linear state equations can be grouped as:

a) Pitch Plane (Longitudinal) State Equation:

$$\begin{bmatrix} \dot{z} \\ \dot{w} \\ \dot{\theta} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & 1 & -u & 0 \\ 0 & \frac{Q_d A C_{z_\alpha}}{um} & 0 & \frac{Q_d A C_{z_q} d}{2um} + u \\ 0 & 0 & 0 & 1 \\ 0 & \frac{Q_d A d C_{m_\alpha}}{u I_y} & 0 & \frac{Q_d A d^2 C_{m_q}}{2u I_y} \end{bmatrix} \begin{bmatrix} z \\ w \\ \theta \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{Q_d A C_{z_{\delta_e}}}{m} \\ 0 \\ \frac{Q_d A d C_{m_{\delta_e}}}{I_y} \end{bmatrix} \delta_e \qquad (1)$$

b) Roll-Yaw Plane (Lateral) State Equation:

$$\begin{bmatrix} \dot{y} \\ \dot{v} \\ \dot{\psi} \\ \dot{r} \\ \dot{p} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & u & 0 & 0 & 0 \\ 0 & \frac{Q_d A C_{y_\beta}}{um} & 0 & \frac{Q_d A C_{y_r} d}{2um} - u & \frac{Q_d A C_{y_p} d}{2um} & g\cos(\theta_0) \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{Q_d A d C_{n_\beta}}{u I_z} & 0 & \frac{Q_d A d^2 C_{n_r}}{2u I_z} & \frac{Q_d A d^2 C_{n_p}}{2u I_z} & 0 \\ 0 & \frac{Q_d A d C_{l_\beta}}{u I_x} & 0 & \frac{Q_d A d^2 C_{l_r}}{2u I_x} & \frac{Q_d A d^2 C_{l_p}}{2u I_x} & 0 \\ 0 & 0 & 0 & \tan(\theta_0) & 1 & 0 \end{bmatrix} \begin{bmatrix} y \\ v \\ \psi \\ r \\ p \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{Q_d A C_{y_{\delta_r}}}{m} & \frac{Q_d A C_{y_{\delta_a}}}{m} \\ 0 & 0 \\ \frac{Q_d A d C_{n_{\delta_r}}}{I_z} & \frac{Q_d A d C_{n_{\delta_a}}}{I_z} \\ \frac{Q_d A d C_{l_{\delta_r}}}{I_x} & \frac{Q_d A d C_{l_{\delta_a}}}{I_x} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_r \\ \delta_a \end{bmatrix} \qquad (2)$$

Aerodynamic coefficients in the above equations can be found using Missile DATCOM software. Details of the mathematical model are given in Appendix A.1.

2.3. PID Design

The basic structure of a PID is shown in Figure 2, and the transfer function of the PID controller is given below:

$$K_p + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_p s + K_I}{s} \qquad (3)$$

where K_p is "proportional gain", K_I is "integral gain", and K_D is "derivative gain".

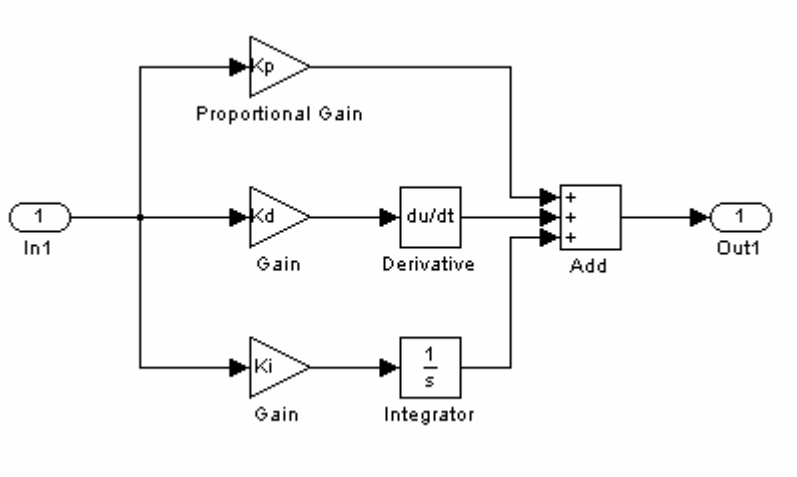


Figure 2 Structure of a PID System

How the PID controller works in a closed-loop system using the schematic shown in Figure 3 can be presented as follows: The plant stands for the system to be controlled. The difference between the desired input value (R) and the actual output (Y) is the tracking error and denoted by (e). This error signal (e) is sent to the PID controller, and the controller computes both the derivative and the integral of this error signal. The output signal (u) of the controller is now equal to the proportional gain (Kp) times the magnitude of the error plus the integral gain (Ki) times the integral of the error plus the derivative gain (Kd) times the derivative of the error [9].

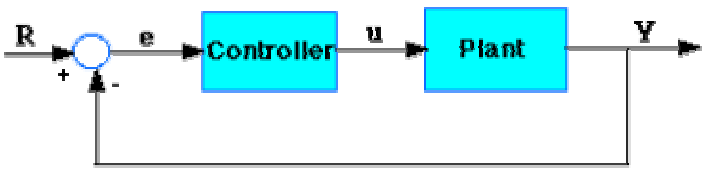


Figure 3 Schematics for a Closed Loop Control System

$$u = K_p e + K_I \int e dt + K_D \frac{de}{dt} \quad (4)$$

This signal (u) will be sent to the plant, and the new output (Y) will be obtained. The new output (Y) is sent back to the sensor again to find the new error signal (e). The controller takes this new error signal and computes its derivative and its integral again. This process goes on continuously [9].

A proportional controller (Kp) will have the effect of reducing the rise time and will reduce, but never eliminate, the steady-state error. An integral control (Ki) will have the effect of eliminating the steady-state error, but it may make the transient response worse. A derivative control (Kd) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response [9] [10]. The effects of Kp, Kd, and Ki on a closed-loop system are summarized in Table 1.

Table 1 Effects of Kp, Kd, and Ki on a closed-loop system

Cl Response	Rise Time	Overshoot	Settling Time	S-S Error
Kp	Decrease	Increase	Small Change	Decrease
Ki	Decrease	Increase	Increase	Eliminate
Kd	Small Change	Decrease	Decrease	Small Change

Table 1 should only be used as a reference when determining the values for Ki, Kp and Kd, because changing one of these variables can change the effect of the others.

MATLAB Simulink's Response Optimization Toolbox is a useful tool for determining a variable in a system if the signal attribute in a line can be estimated. The error line in the PID controllers has a known attribute, so Response Optimization Toolbox is used for determining the gain values for PID controllers.

2.3.1. Autopilot Design Using Target Position Information as Input

Inner and outer loops are used by classical PID control design. Inner loops stabilize the fast dynamics such as rotational rates where outer loop controls the main input, position of the target. Figure 4 shows the reference and tracked trajectories in x-y-z axes.

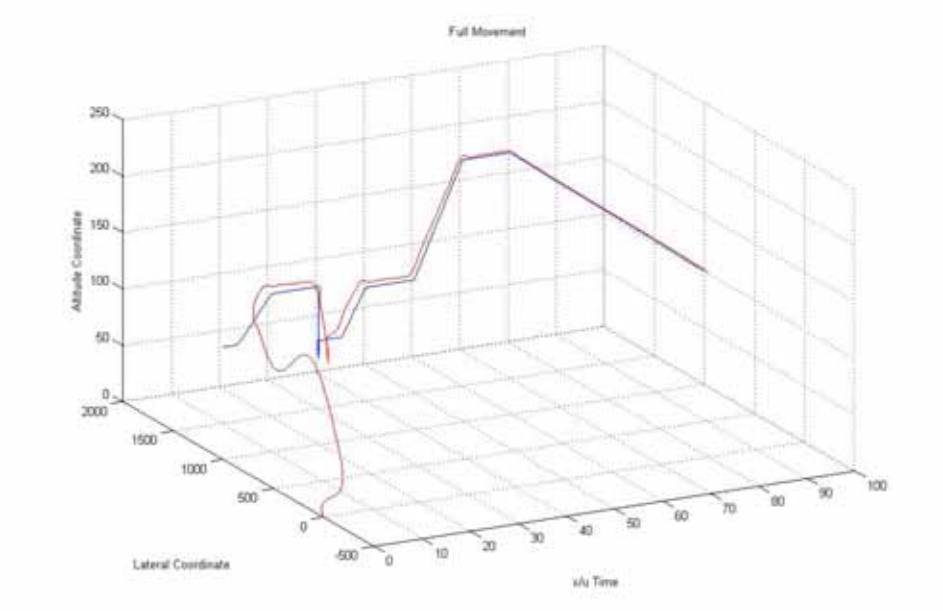


Figure 4 Reference (Blue) and Tracked (Red) Trajectories in 3D Plot

2.3.1.1. Longitudinal Movement

Longitudinal movement is the movement in the vertical plane. The inner loop stabilizes the pitch rate q (rotational rate) where the outer loop controls the altitude of the target as in Figure 5. the target's altitude movement and the tracked path for the missile are shown in Figure 6 by blue and red lines, respectively.

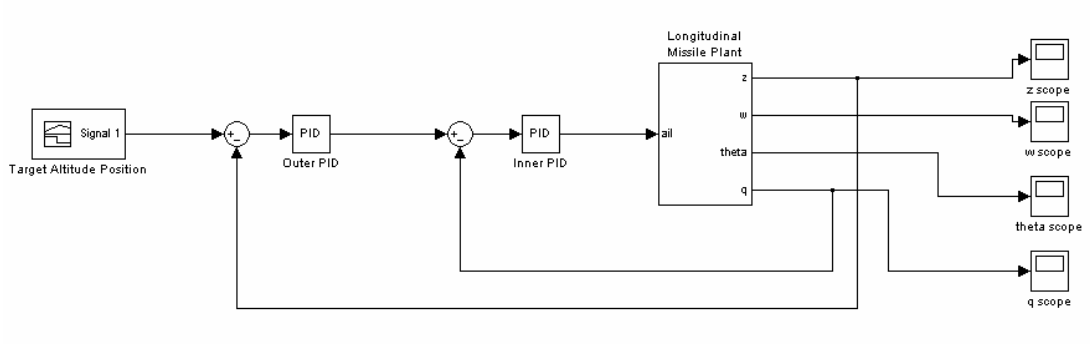


Figure 5 Longitudinal Autopilot Architecture with PIDs

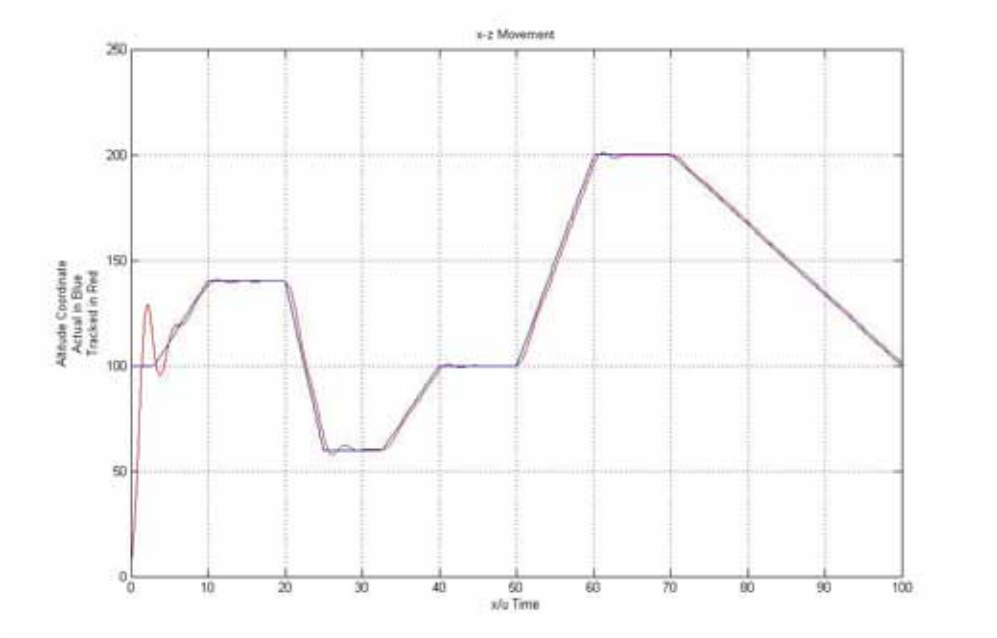


Figure 6 Target's Altitude Position (Blue) and Response of the Missile (Red)

2.3.1.2. Lateral Movement

Lateral movement describes the movement in the horizontal plane. The inner loop stabilizes the yaw rate r and roll rate p (rotational rates) where the outer loop controls

the lateral position of the target as in Figure 7. The target's lateral movement and the tracked path for the missile are shown in Figure 8 by blue and red lines, respectively.

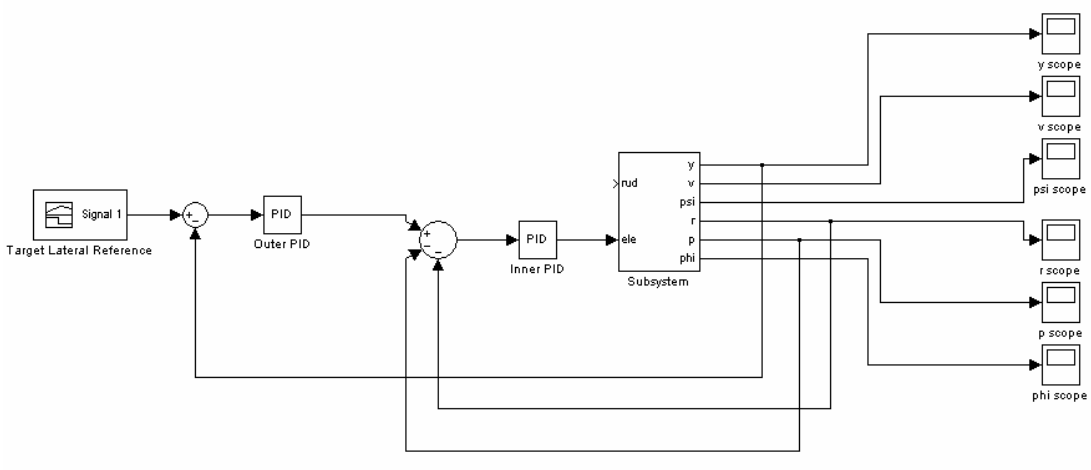


Figure 7 Lateral Autopilot Architecture with PIDs

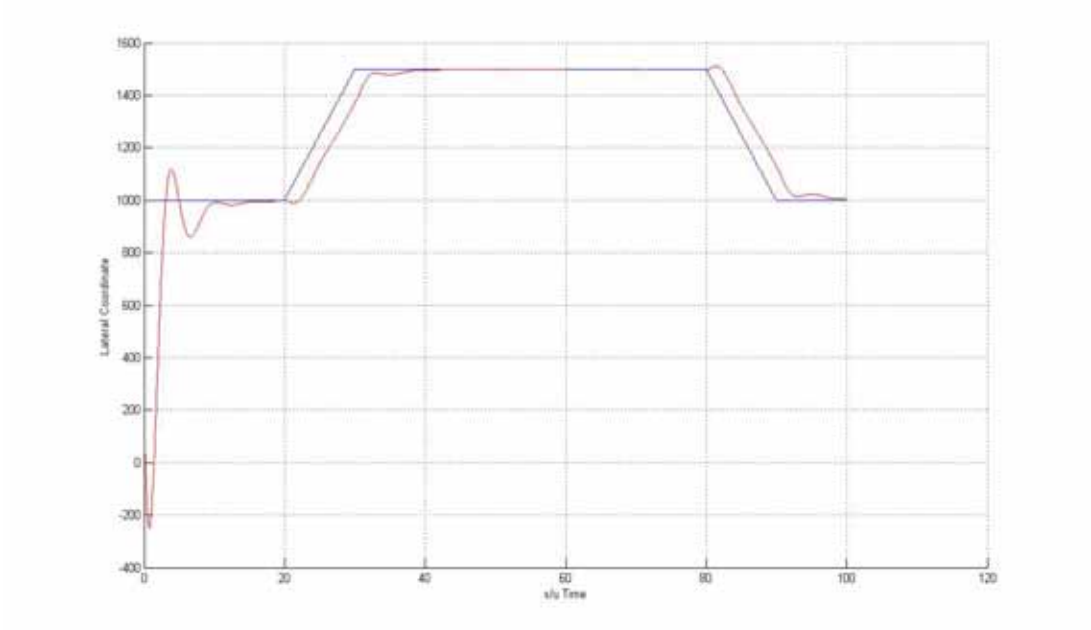


Figure 8 Target's Lateral Position (Blue) and Response of the Missile (Red)

2.3.2. Autopilot Design Using Target – Missile Angle Information as Input

Inner and outer loops are used by classical PID control design. The inner loops stabilize the fast dynamics such as rotational rates where the outer loops control the main input such as position of the target or the angle between the missile and the target.

2.3.2.1. Longitudinal Movement

As in Figure 9, the inner loop stabilizes the pitch rate q (rotational rate) where the outer loop controls the pitch angle (the angle between the missile's velocity vector and missile – target line). The actual and tracked pitch angles for the system are shown in Figure 10 by blue and red lines, respectively.

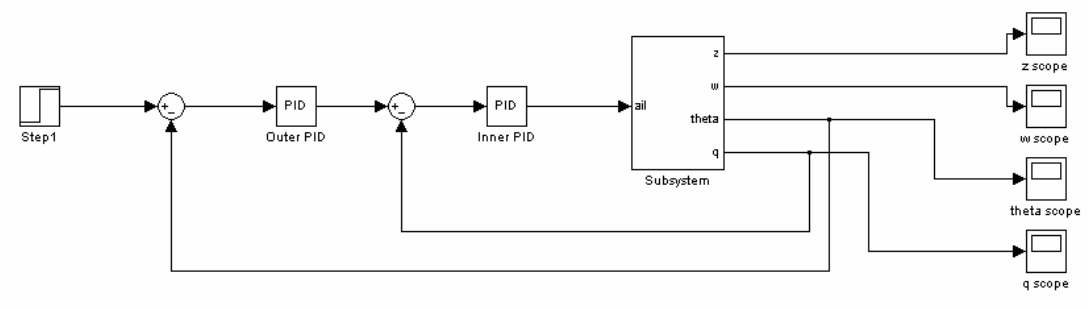


Figure 9 Longitudinal Autopilot Architecture with PIDs where the Pitch Angle is the Input

2.3.2.2. Lateral Movement

The inner loop stabilizes the yaw rate r and the roll rate p (rotational rates) where the outer loop controls the yaw angle (the angle between the missile's velocity vector and missile – target line's projection into x-y plane) as in Figure 11. The actual and tracked yaw angles for the system are shown in Figure 12 and Figure 13 by blue and red lines, respectively.

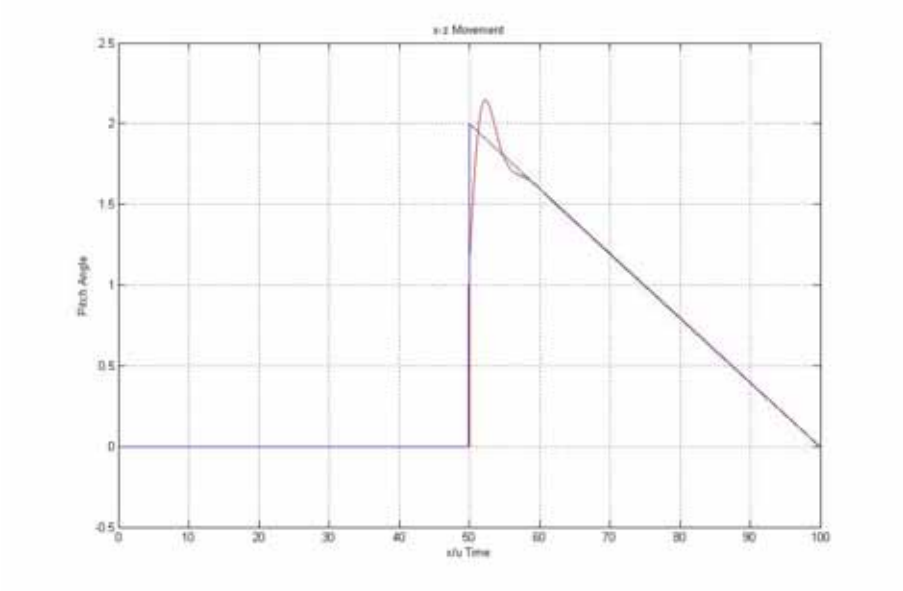


Figure 10 Actual (blue) and Tracked (red) Pitch Angle for Longitudinal Autopilot Design

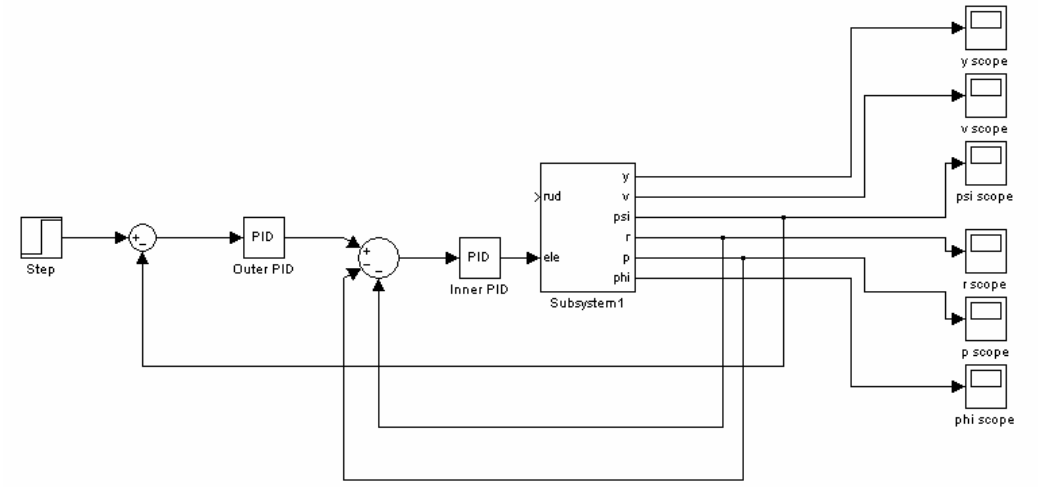


Figure 11 Lateral Autopilot Architecture with PIDs where the Yaw Angle is the Input

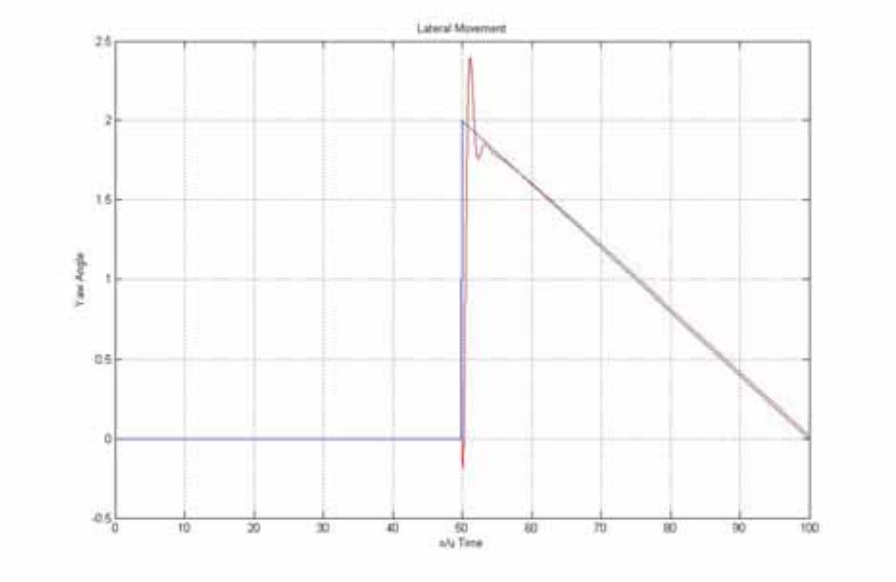


Figure 12 Actual (blue) and Tracked (red) Pitch Angles for Lateral Autopilot Design

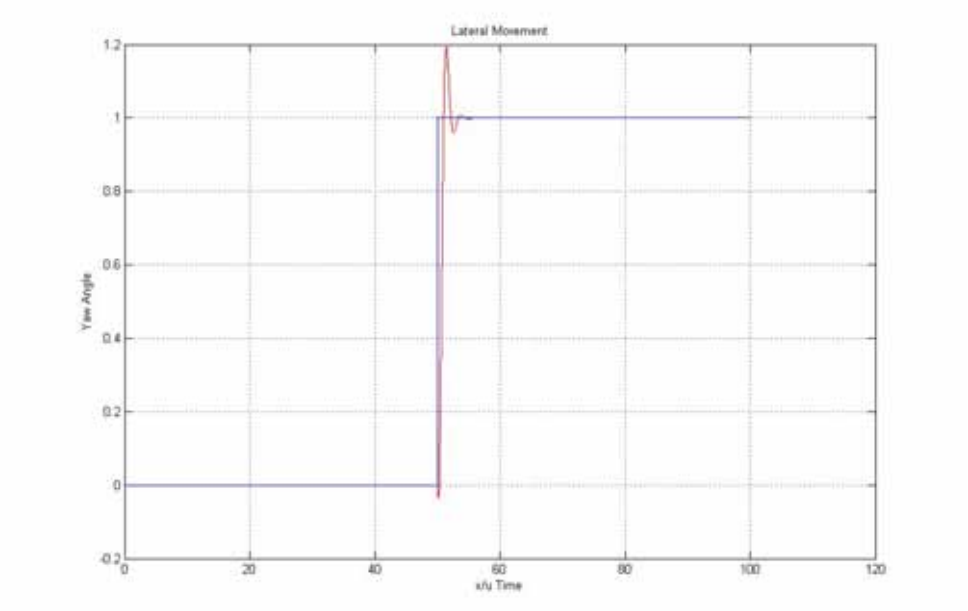


Figure 13 Actual (blue) and Tracked (red) Pitch Angles for Lateral Autopilot Design Under Step Input

2.4. LQR Design

The plant which is going to be considered in LQR design is,

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{5}$$

The plant is time-invariant for our case, so it is as in Figure 14 and Eq.5,

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{6}$$

Main purpose of the LQR design is to find the matrix of the feedback gains [11] as in Eq. 7,

$$u = -Kx \tag{7}$$

so as to minimize the performance index in Eq. 8,

$$J = \frac{1}{2}\int_0^\infty (xQx^T + uRu^T)dt \tag{8}$$

where J is the performance index, Q and R are the symmetric semi-definite matrices called weighting matrices. The Q and R matrices will be determined according to the requirements on time-domain criteria (settling time, rise time) and control effort.

The solution for the regulator problem is achieved by transforming the dynamical optimization into static one which is easier to solve. If supposed that there is a symmetric, positive-semi-definite matrix P such as

$$\frac{d}{dt}(x^T Px) = -x^T(Q + K^T RK)x \tag{9}$$

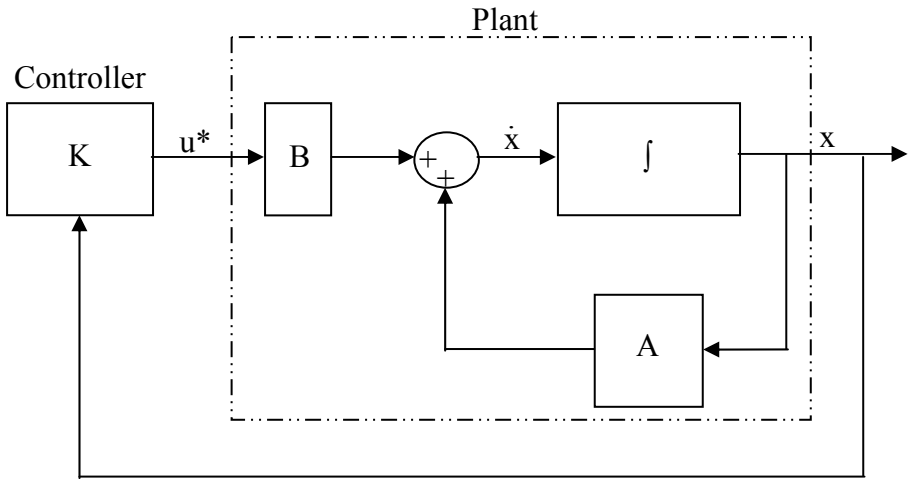


Figure 14 Plant and Feedback Controller for Linear Regulator Problems

then J, the performance index can be written as,

$$J = \frac{1}{2}x^T(0)Px(0) - \frac{1}{2}\lim_{t\to\infty} x^T(t)Px(t) \tag{10}$$

Under the asymptotically stable closed-loop assumption, J becomes,

$$J = \frac{1}{2}x^T(0)Px(0) \tag{11}$$

For the closed loop system,

$$\dot{x} = (A - BK)x = A_c x \tag{12}$$

so, Eq. 9 becomes,

$$-x^T(Q + K^T RK) = \dot{x}^T Px + x^T P\dot{x} = x^T(A_c^T P + PA_c)x \tag{13}$$

This equivalence must hold for every initial condition. Hence,

$$g = A_c^T P + P\dot{A}_c + K^T RK + Q = 0 \tag{14}$$

so the optimization problem can be stated as finding out the feedback gains K that minimize J in Eq. 8 subject to Eq. 7. The Lagrange multiplier approach is used to solve this optimization problem. The Hamiltonian is

$$H = \frac{1}{2} x^T(0) P x(0) + gS \tag{15}$$

where S is a symmetric matrix of Lagrange multipliers. The necessary conditions are:

$$0 = \frac{\partial H}{\partial S} = g = A_c^T P + P\dot{A}_c + K^T RK + Q \tag{16}$$

$$0 = \frac{\partial H}{\partial P} = A_c S + S\dot{A}_c^T + x(0) \tag{17}$$

$$0 = \frac{1}{2}\frac{\partial H}{\partial K} = RKS - BPS \tag{18}$$

If R is positive definite then the feedback gains can be obtained as,

$$K = R^{-1} B^T P \tag{19}$$

so it is not necessary to solve the equation for S to obtain the feedback gains. Using $K = R^{-1} B^T P$ and the combination of the equations 16 -19,

$$0 = A^T P + PA + \dot{Q} - PBR^{-1} B^T P \tag{20}$$

which is called the Algebraic Riccati Equation (ARE) [11].

2.4.1. Tracking Problem with LQR

It is easy to regulate all states through zero by the help of the LQR design. On the other hand, in the tracking problem the states should go to the desired values from

their initial values. It is possible to track the desired values by LQR by assigning errors as new states. For this purpose, an error state is defined as

$$e = x_{com} - x \tag{21}$$

Let

$$\xi = \begin{bmatrix} \dot{x} \\ e \end{bmatrix} \tag{22}$$

then

$$\dot{\xi} = \begin{bmatrix} \ddot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ e \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} v \tag{23}$$

where

$$\boldsymbol{v = \dot{u}} \tag{24}$$

Using the results of the Section 2.4, the control law v will be

$$v = \begin{bmatrix} -K_1 & K_2 \end{bmatrix} \begin{bmatrix} \dot{x} \\ e \end{bmatrix} \tag{25}$$

$$u = \int v = -K_1 x - \int K_2 e \tag{26}$$

By this control law, LQR will regulate the error state through zero which is needed for tracking. The overshoot and rise time can be controlled by the weighting matrix Q.

2.4.2. Autopilot Design Using Target Position Information as Input

2.4.2.1. Longitudinal Movement

States for the longitudinal movement are z (altitude position), w (x-z velocity), θ(pitch angle) and q (pitch rate). Tracked state is z.

An error state has to be determined in order to regulate the error through zero with LQR design. Since

$$\boldsymbol{e = z_{com} - z} \tag{27}$$

The A and B matrices have to be renewed as discussed in Section 2.4.1.

The feedback gains are found by MATLAB's lqr command, and the results are used in the simulink model as in Figure 17. The actual and the tracked trajectories for the 3 degrees angle of attack and 1.0 Mach number are shown in Figure 15 and Figure 16.

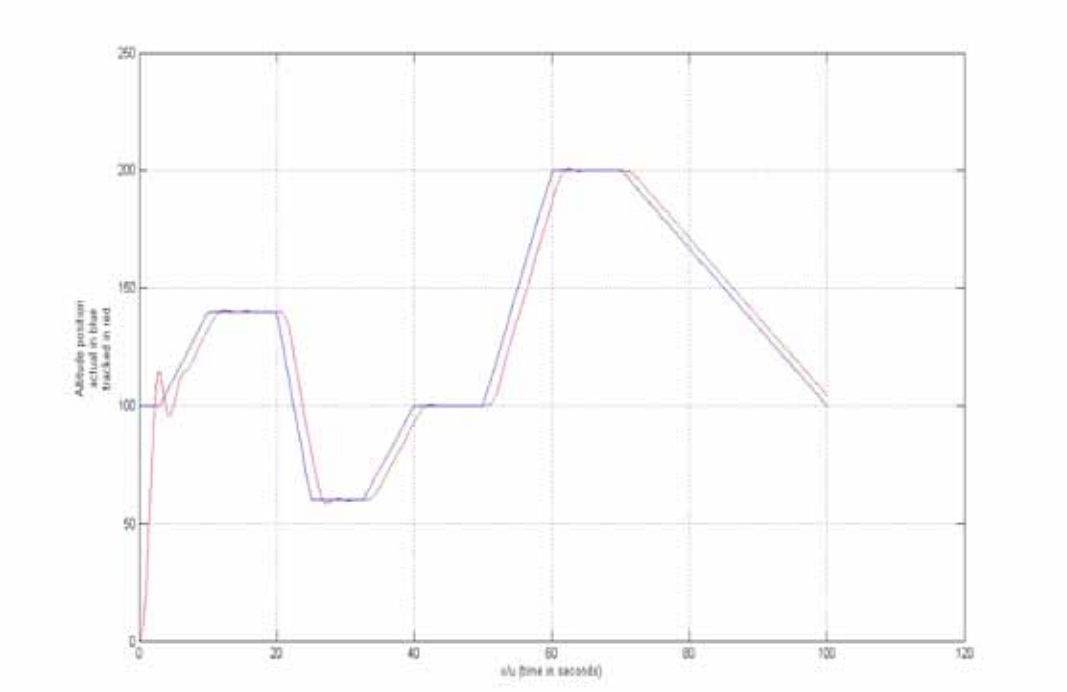


Figure 15 Altitude Tracking (Reference is Blue and Tracked is Red)

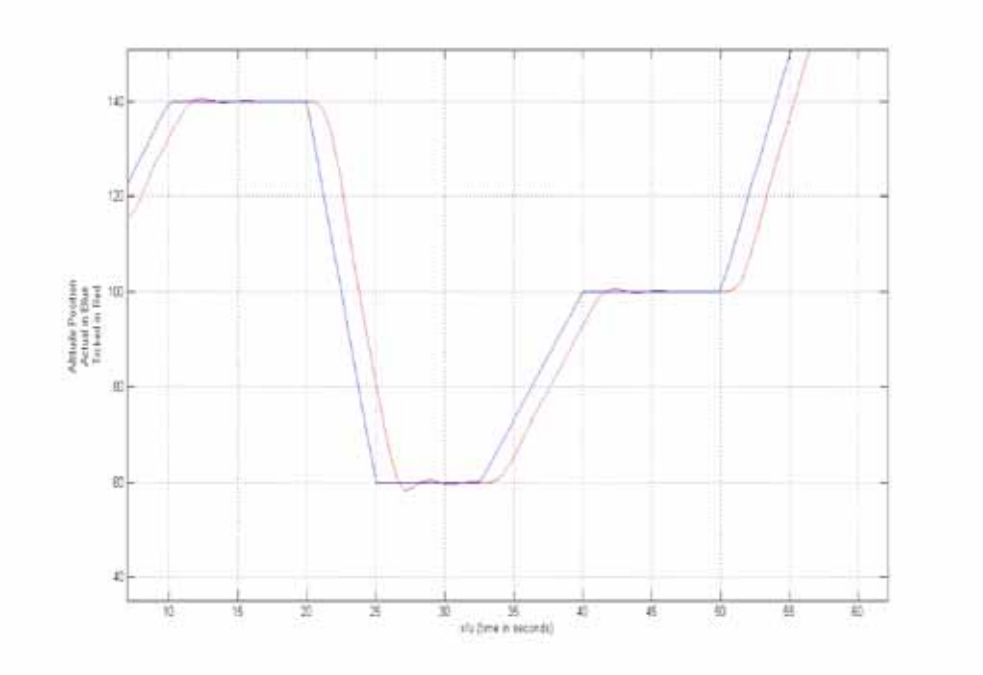


Figure 16 Altitude Tracking (Zoomed) (Reference is Blue and Tracked is Red)

2.4.2.2. Lateral Movement

States for the lateral movement are y (side position), v (side velocity), ψ (yaw angle), p (roll rate), r (yaw rate) and φ (roll angle). The tracked state will be determined by the guidance algorithm; it is taken as y below.

An error state has to be determined in order to regulate the error through zero with LQR design. Since

$$e = y_{com} - y \tag{28}$$

The A and B matrices have to be renewed as discussed in Section 2.4.1.

The feedback gains are found by MATLAB's lqr command, and the results are used in the simulink model as in Figure 18.

Actual and tracked trajectories are shown in Figure 19 and Figure 20 by blue and red lines, respectively.

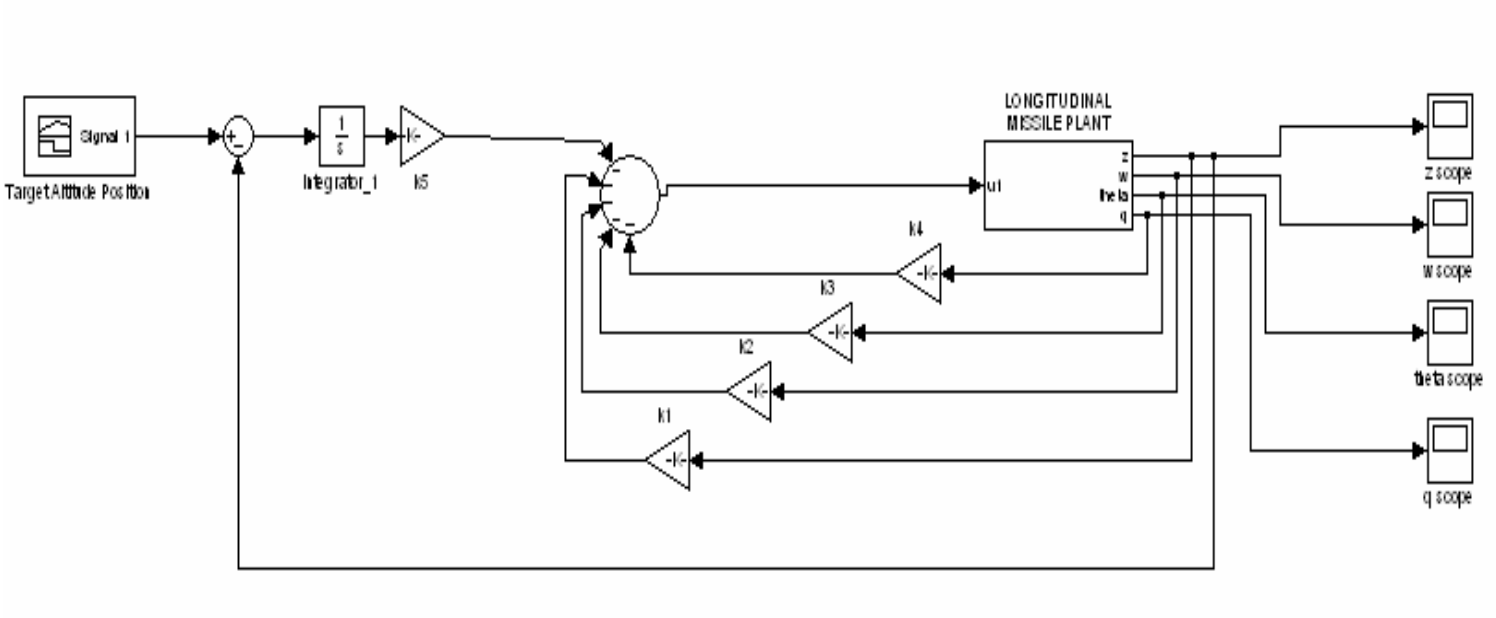


Figure 17 Longitudinal (x-z) Movement in Simulink

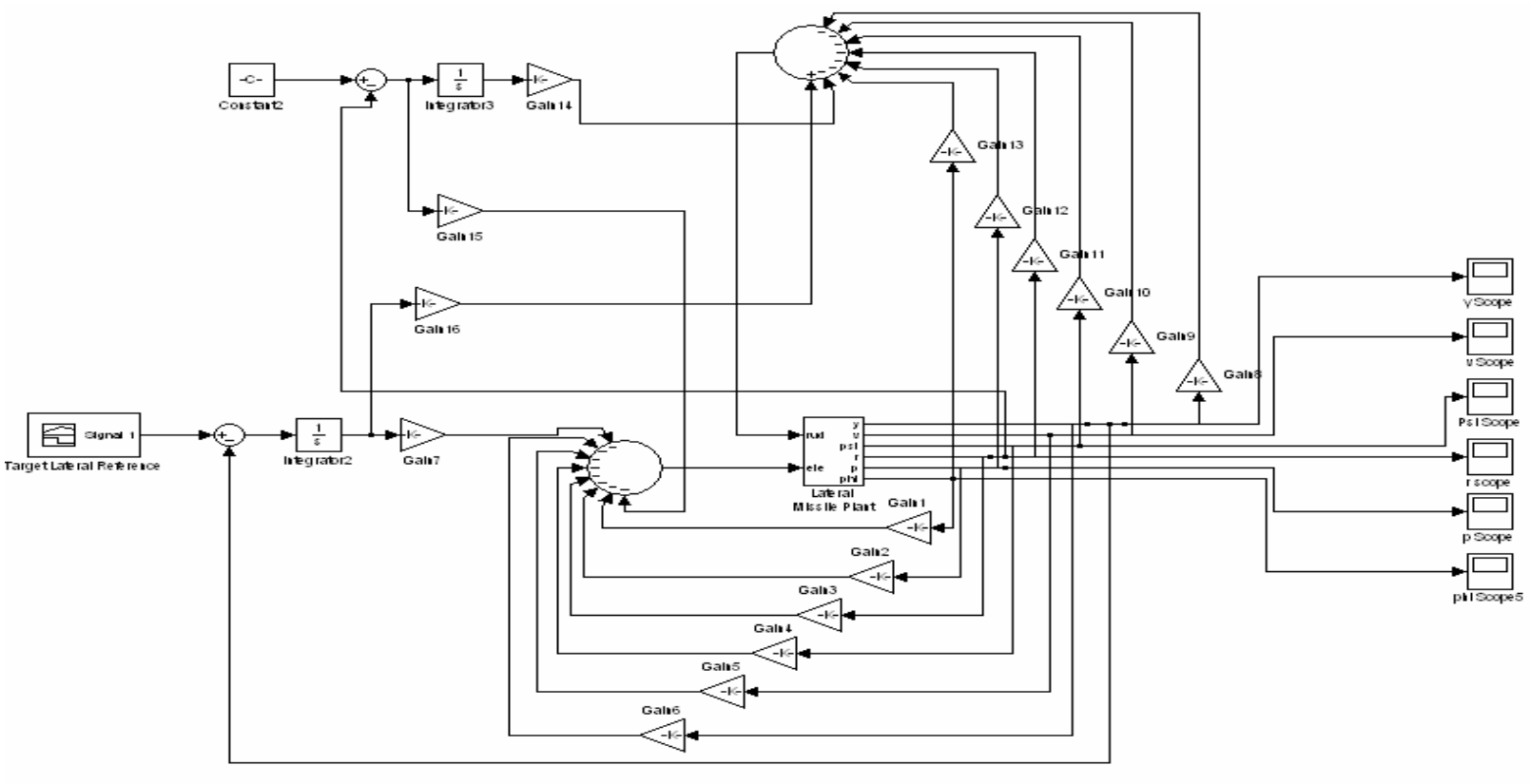


Figure 18 Lateral (x-y) Movement in Simulink

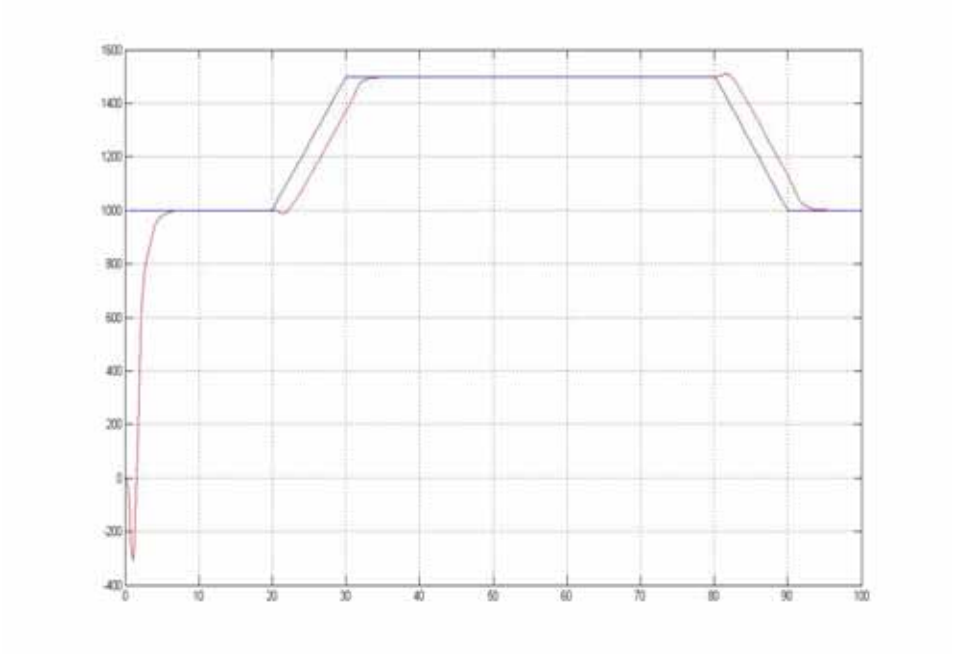


Figure 19 Lateral Movement Tracking (Reference is Blue and Tracked is Red)

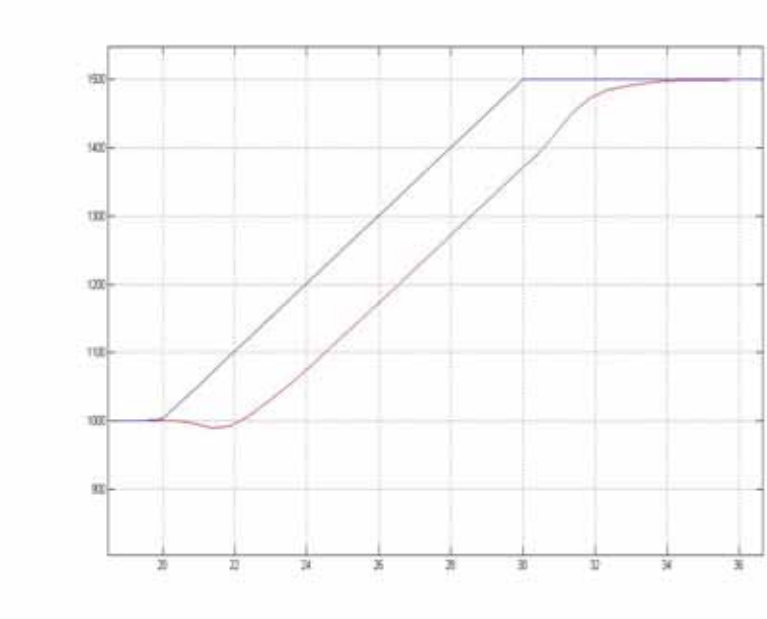


Figure 20 Lateral Movement Tracking (Zoomed)

(Reference is Blue and Tracked is Red)

2.4.3. Autopilot Design Using Target – Missile Angle Information as Input

2.4.3.1. Lateral Movement

States for the lateral movement are y (side position), v(side velocity), ψ (yaw angle), p (roll rate), r(yaw rate) and φ (roll angle). The tracked state will be determined by the guidance algorithm, but is going to be ψ (yaw angle) for now.

An error state has to be determined in order to regulate the error through zero with LQR design. Since

$$e = \Psi_{com} - \Psi \tag{29}$$

the A and B matrices have to be renewed as discussed in Section 2.4.1.

The feedback gains are found by MATLAB's lqr command, and the results are used in the simulink model as in Figure 21.

The actual and the tracked trajectories are shown in Figure 22 by blue and red lines, respectively.

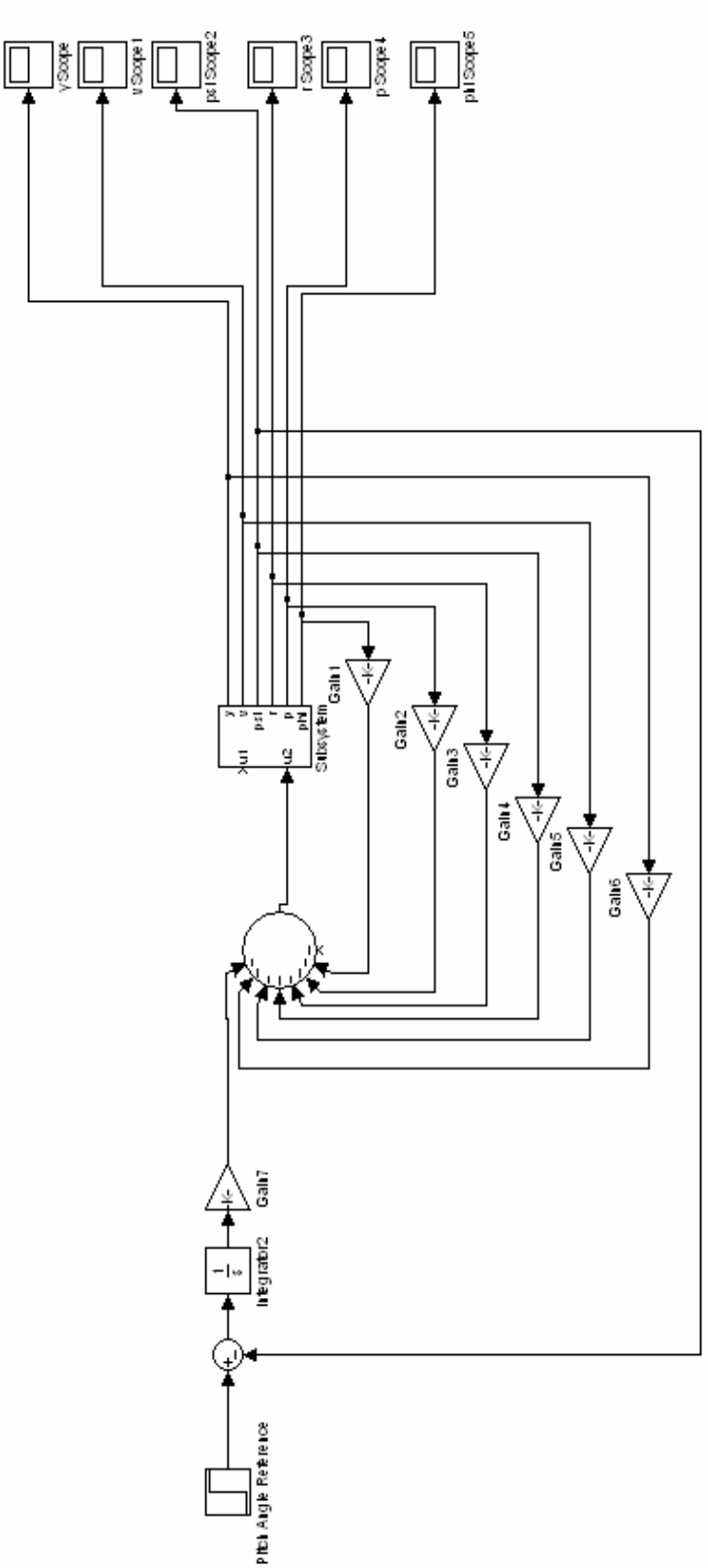


Figure 21 Lateral (x-y) Movement in Simulink

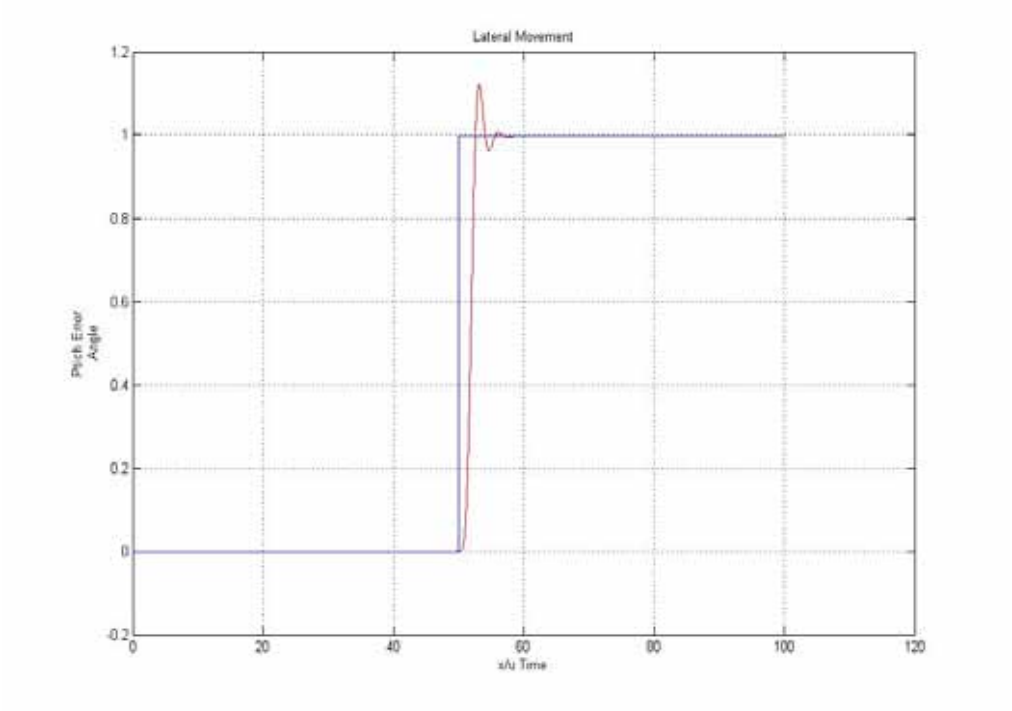


Figure 22 Yaw Error Angle Tracking (Reference is Blue and Tracked is Red)

2.4.3.2. Longitudinal Movement

States for the longitudinal movement are z (altitude position), w (x-z velocity), θ(pitch angle) and q (pitch rate). The tracked state is z.

An error state has to be determined in order to regulate the error through zero with LQR design. Since

$$e = \theta_{com} - \theta \tag{30}$$

the A and B matrices have to be renewed as discussed in Section 2.4.1.

The feedback gains are found by MATLAB's lqr command, and the results are used in the simulink model as in Figure 24.

The actual and the tracked trajectories for the 3 degrees angle of attack and 1.0 Mach number are shown in Figure 23 by blue and red lines, respectively.

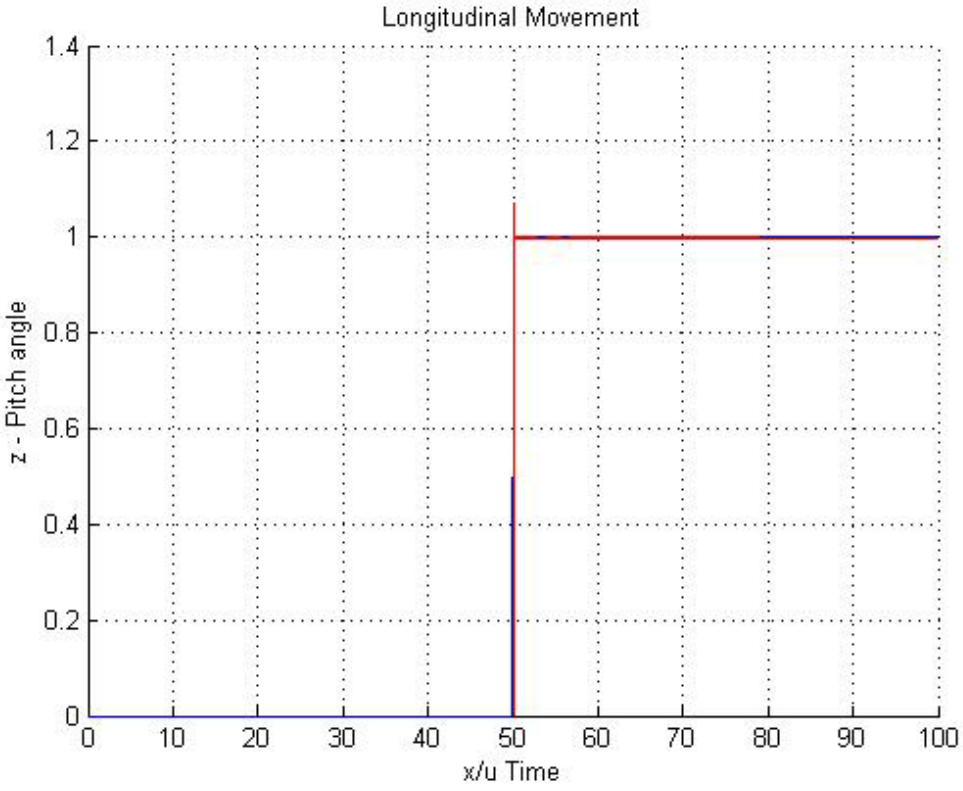


Figure 23 Yaw Error Angle Tracking (Reference is Blue and Tracked is Red)

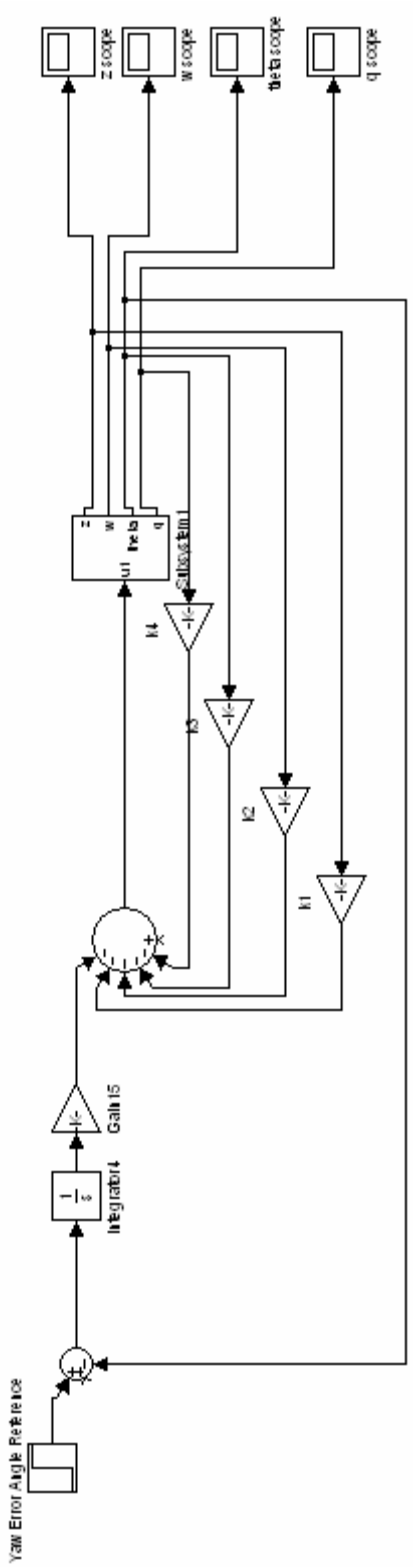


Figure 24 Lateral (x – y) Movement in Simulink

CHAPTER 3

GUIDANCE DESIGN

3.1. Introduction

Seeker and guidance systems are parts of the missile which are used for detecting and tracking the target. Inputs for the autopilot systems are generated by the seeker and the guidance system of the missile (Figure 25).

This chapter represents the seeker model for an infrared guided missile to be used, and then the guidance algorithms to be applied to the system are studied.

3.2. Seeker Model

The seeker is a combination of electronic, optical, and mechanical components. The heart of the seeker is the detector because it converts scene radiation into measurable electrical signal. The seeker used in this study has a four-quadrant detector which has four distinct photosensitive elements. This means that the electrical signal produced by the detector is composed of four different detector outputs. If the electrical signal produced by the detector is composed of only one detector output, then the target's location can be estimated too.

The seeker model which is used throughout the thesis is an improved model of the one which was studied by Elzem Akkal in December, 2003 [8].

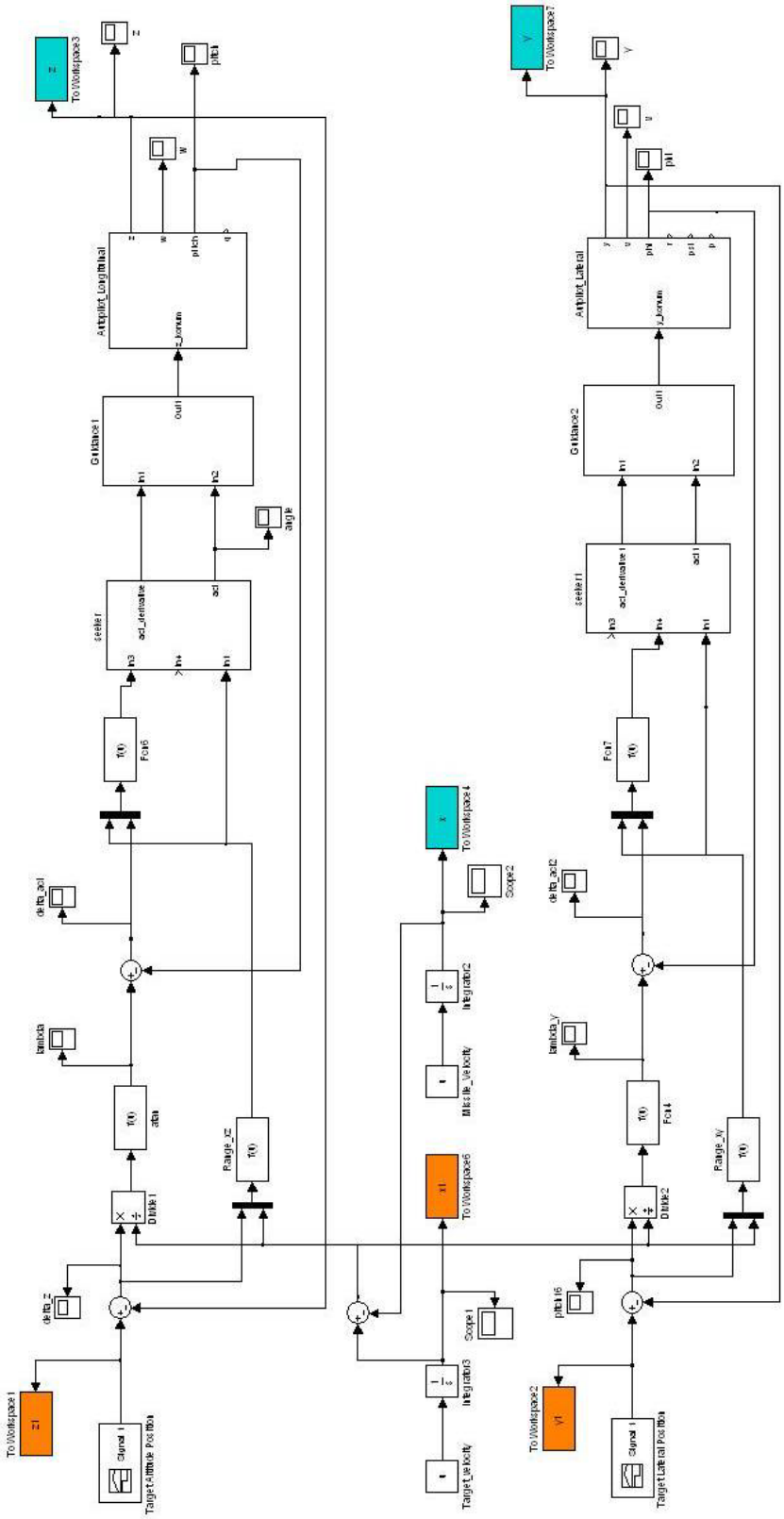


Figure 25 Structure of the System

3.2.1. Error-Angle Estimation

Error-angles are pitch and yaw angles. Pitch error angle is the angle between the velocity vector of the missile and the Line of Sight vector between the missile and the target where as yaw error angle is the angle between the velocity vector of the missile and the Line of Sight vector's projection into ground. It is obvious that ray of light passing the lens from its center does not bend; so, the values of these angles could be estimated. Hence if the distance of detector to the lens is known, and if the spot center location could be estimated, the error angles can be estimated by the following expressions (Figure 26):

$$\text{Pitch Angle} = \theta = \gamma_p = a\tan\left(\frac{\Delta y}{d}\right) \tag{31}$$

$$\text{Yaw Angle} = \Psi = \gamma_y = a\tan\left(\frac{\Delta z}{d}\right) \tag{32}$$

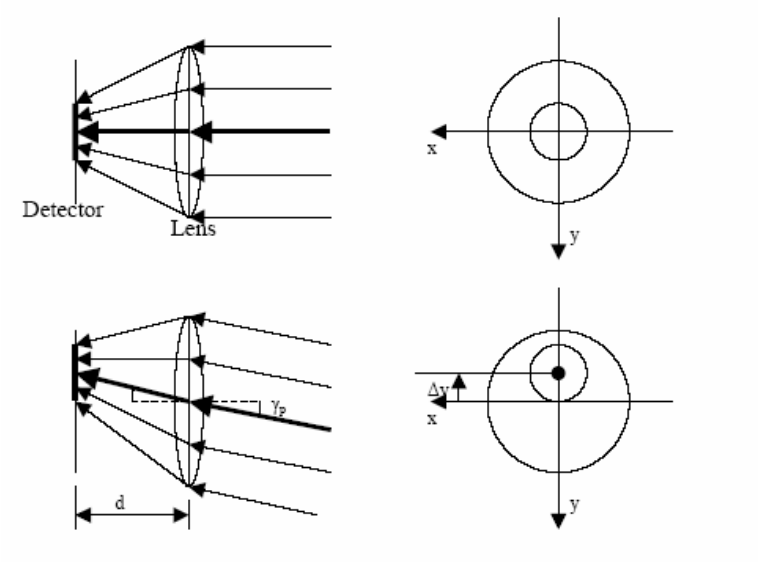


Figure 26 Pitch Error Angle

3.2.2. Spot Location Estimation

3.2.2.1. Direct Spot Location Estimation

If the infrared spot and the detector's field of view are as in Figure 27, the spot center location coordinates could be found by Eq. 33 and Eq. 34. These equations are used in most of the applications, which is an approximation to the actual relationship between the amount of shift in the voltages and the amount of shift in the spot center location. Four area values can be used instead of four voltage values, which are calculated using geometry once the spot center location is given.

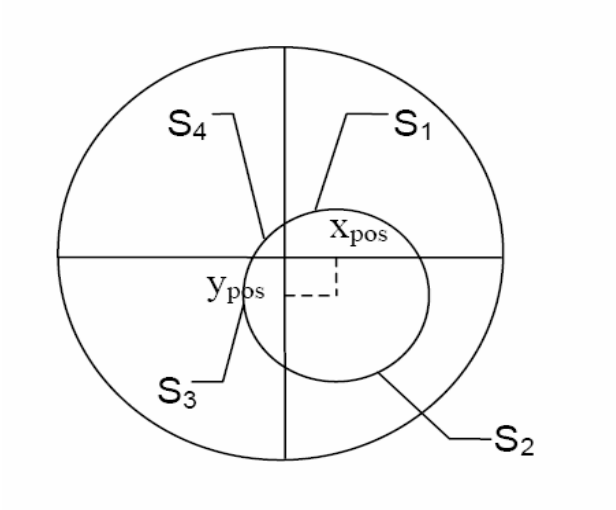


Figure 27 Direct Spot Location Estimation

$$X_{pos} = \frac{(S_1 + S_2) - (S_3 + S_4)}{S_1 + S_2 + (S_3 + S_4} \tag{33}$$

$$Y_{pos} = \frac{(S_1 + S_4) - (S_3 + S_2)}{S_1 + S_2 + S_3 + S_4} \tag{34}$$

3.2.2.2. Spot Location Estimation with Radial Basis Neural Network

The four area values are inputs and X and Y components of the spot center location are the outputs of the Artificial Neural Network model. Hence training data set must

be composed of some input values and corresponding output values. If the input points cover the input space uniformly, the possibility of learning the relationship between inputs and outputs will be almost the same. However in our case, input-output pairs are created using the expressions given below and the input of these expressions are the outputs of the neural network, namely X_{pos} and Y_{pos}. Therefore, we can only select output points, X_{pos} and Y_{pos}, covering output space uniformly, the spread of input points does not have to be uniform. Hence, a training data set is created with more data than required for training. In other words, created training data set is possibly over determined.

The target is considered as a circle. Since the possible spot center locations make a circle too with the radius changing according to range between target and the missile, training data set must be formed by selecting many X_{pos} and Y_{pos} values in this circle and estimating the areas corresponding to these locations. In this way 1319 input output pair is created in order to train the radial basis neural network.

If "r" is the radius of the spot, then the spot location formulas can be found as follows using the geometrical properties of the structure in Figure 28.

$$S_1 = \pi\frac{r^2}{4} + \frac{r^2}{2}a\sin(\frac{x_{pos}}{r}) + \sqrt{r^2 - x^2{}_{pos}}\,\frac{x_{pos}}{2} + \frac{r^2}{2}a\sin(\frac{y_{pos}}{r}) \qquad (35)$$
$$+ \sqrt{r^2 - y^2{}_{pos}}\,\frac{y_{pos}}{2} + x_{pos}y_{pos}$$

$$S_2 = \pi\frac{r^2}{4} + \frac{r^2}{2}a\sin(\frac{x_{pos}}{r}) + \sqrt{r^2 - x^2{}_{pos}}\,\frac{x_{pos}}{2} - \frac{r^2}{2}a\sin(\frac{y_{pos}}{r}) \tag{36}$$

$$-\sqrt{r^2 - y^2{}_{pos}}\,\frac{y_{pos}}{2} - x_{pos}y_{pos}$$

$$S_3 = \pi\frac{r^2}{4} - \frac{r^2}{2}a\sin(\frac{x_{pos}}{r}) - \sqrt{r^2 - x^2{}_{pos}}\,\frac{x_{pos}}{2} - \frac{r^2}{2}a\sin(\frac{y_{pos}}{r}) \tag{37}$$

$$-\sqrt{r^2 - y^2{}_{pos}}\,\frac{y_{pos}}{2} + x_{pos}y_{pos}$$

$$S_4 = \pi\frac{r^2}{4} - \frac{r^2}{2}a\sin(\frac{x_{pos}}{r}) - \sqrt{r^2 - x^2{}_{pos}}\,\frac{x_{pos}}{2} + \frac{r^2}{2}a\sin(\frac{y_{pos}}{r}) \tag{38}$$

$$+\sqrt{r^2 - y^2{}_{pos}}\,\frac{y_{pos}}{2} - x_{pos}y_{pos}$$

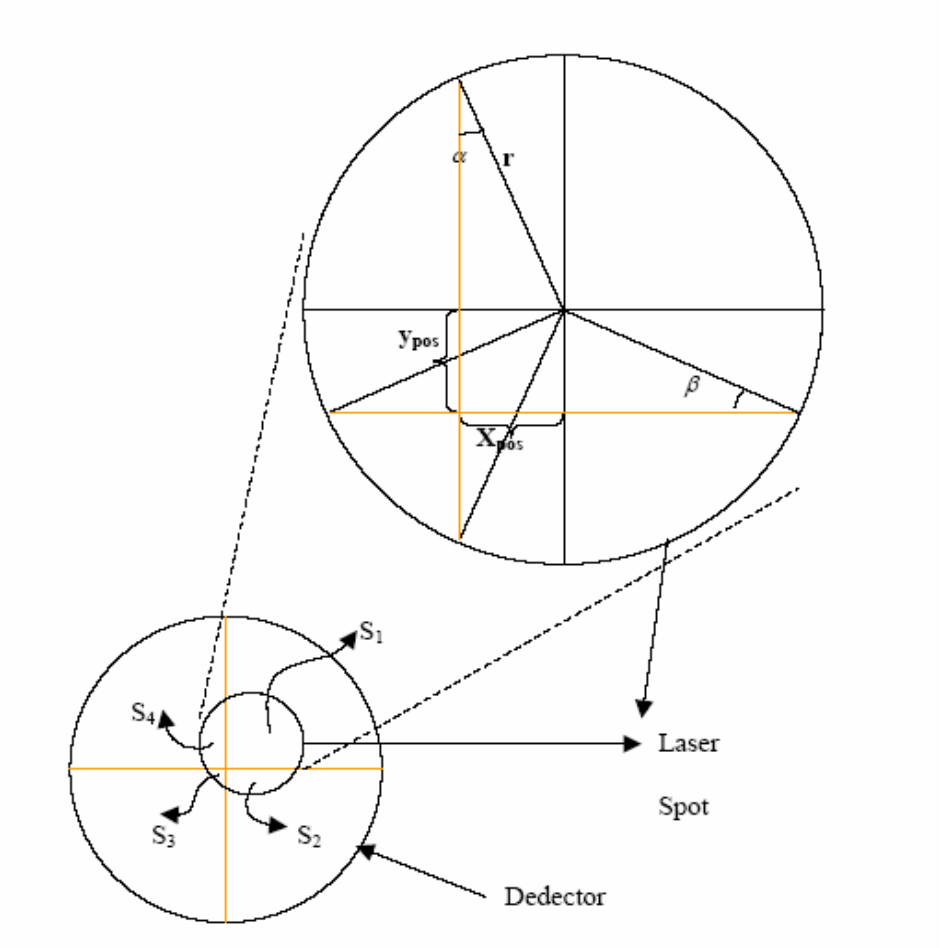


Figure 28 Spot Location Estimation

3.2.3. Target Area Estimation

In infrared guided missiles, the range from missile to its target is unknown. So, only the area of the target in its current position is an input for the seeker system.

The target is assumed as a circle with 2 meters radius and the target area's projections, spot area, onto the seeker detectors for different ranges (from 20 meters to 6000 meters) are estimated. A faraway target has a very small area in the seeker where a close target has a very large area. So, the missile does not need to know the distance (Figure 29).

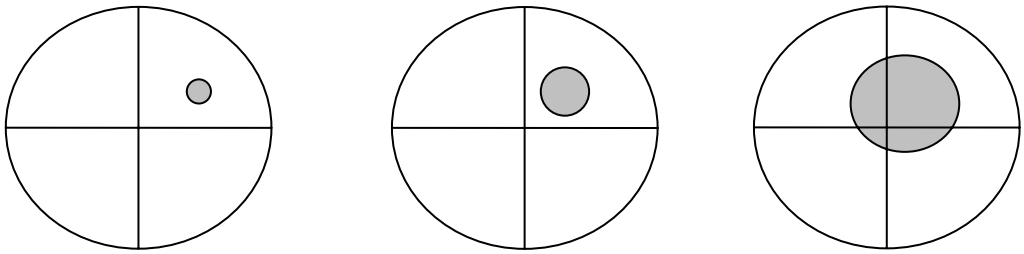


Figure 29 Four Quadrant Detector. Target is:
(a) Faraway (b) Middle (c) Close

After the spot area estimation, the spot center and the error-angle estimation can be made.

3.3. Guidance Models

Guidance system's inputs are error angles in both pitch and yaw planes which are created by the seeker part. Guidance system makes some estimation and creates the necessary acceleration (or position) values for the missile in order to give the directions for the target's next position. Since the pitch and yaw planes are decoupled, guidance systems are designed for pitch and yaw planes separately [17].

The most popular guidance method is Proportional Navigation Guidance (PNG) which has several subsets such as Classical PNG, Modified PNG, etc.

3.3.1. Classical PNG

The PNG law seeks to null the line of sight (LOS) rate by making the missile turn rate be directly proportional to the LOS rate.

The proportional navigation guidance law issues acceleration commands perpendicular to the instantaneous missile target line of sight, which is proportional to the line of sight rate and closing velocity [2]. Mathematically, the guidance law can be expressed as;

$$n_c = N' V_c \dot{\lambda} \tag{39}$$

In IR guided missiles, the line of sight rate is measured whereas the closing velocity is taken as a constant [17].

The closing velocity V_C is the negative rate of change of the distance from the missile to the target. The closing velocity will be zero when R_{TM} is minimum,

$$V_c = -\dot{R}_{TM} \tag{40}$$

Since the target acceleration n_T is perpendicular to the target velocity vector, the angular velocity of the target ca be expressed as,

$$\dot{\beta} = \frac{n_T}{V_T} \tag{41}$$

where V_T is the magnitude of the target velocity.

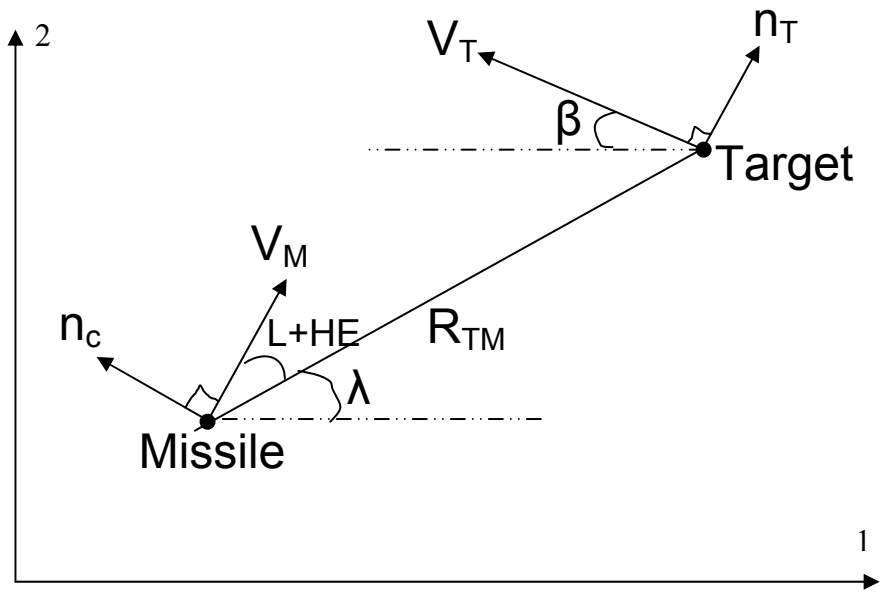


Figure 30 Classical PNG in 2D Space

The components of the target velocity vector in the Earth or the inertial coordinate system can be found by,

$$V_{T1} = -V_T \cos\beta \tag{42}$$

$$V_{T2} = -V_T \sin\beta \tag{43}$$

The target position components in the Earth fixed coordinate system can be found by directly integrating the target velocity components,

$$\dot{R}_{T1} = V_{T1} \tag{44}$$

$$\dot{R}_{T2} = V_{T2} \tag{45}$$

Similarly, the missile velocity and position differential equations are given by

$$\dot{V}_{M1} = a_{M1} \tag{46}$$

$$\dot{V}_{M2} = a_{M2} \tag{47}$$

$$\dot{R}_{M1} = V_{M1} \tag{48}$$

$$\dot{R}_{M2} = V_{M2} \tag{49}$$

where a_{M1} a_{M2} are the missile acceleration components in the Earth coordinate system.

The relative missile-target separations can be defined as,

$$R_{TM1} = R_{T1} - R_{M1} \tag{50}$$

$$R_{TM2} = R_{T2} - R_{M2} \tag{51}$$

From Figure 30, line of sight angle can be found by

$$\lambda = \tan^{-1} \frac{R_{TM2}}{R_{TM1}} \tag{52}$$

If the relative velocity components in Earth coordinates are defined by

$$V_{TM1} = V_{T1} - V_{M1} \tag{53}$$

$$V_{TM2} = V_{T2} - V_{M2} \tag{54}$$

then the line of sight rate can be calculated by the direct differentiation of the expression,

$$\dot{\lambda} = \frac{R_{TM1}V_{TM2} - R_{TM2}V_{TM1}}{R^2_{TM}} \tag{55}$$

The relative separation between the target and missile R_{TM} can be expressed in terms of its inertial components,

$$R_{TM} = \sqrt{R^2{}_{TM1} + R^2{}_{TM2}} \tag{56}$$

Since the closing velocity is defined as the negative rate of change of the missile target separation, it can be obtained by differentiating the preceding equation,

$$V_C = -\dot{R}_{TM} = \frac{-(R_{TM1}V_{TM1} + R_{TM2}V_{TM2})}{R_{TM}} \tag{57}$$

The magnitude of the missile guidance command n_C can be found from the definition of the proportional navigation,

$$n_c = N'V_c\dot{\lambda} \tag{58}$$

The classical PNG can be separated into two groups: Pure PNG and True PNG.

Pure PNG (PPNG)

The commanded acceleration vector is perpendicular to the velocity vector of the missile [13] (Figure 31).

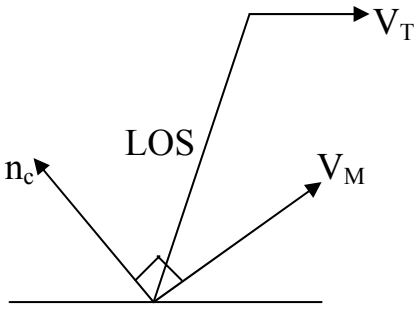


Figure 31 Pure PNG

Since the acceleration command is perpendicular to the velocity vector of the missile, the missile acceleration components in Earth coordinates can be found by,

$$a_{M1} = -n_C \sin(L + HE + \lambda) \tag{59}$$

$$a_{M2} = n_C \cos(L + HE + \lambda) \tag{60}$$

True PNG (TPNG)

The commanded acceleration vector is perpendicular to the LOS [13] (Figure 32).

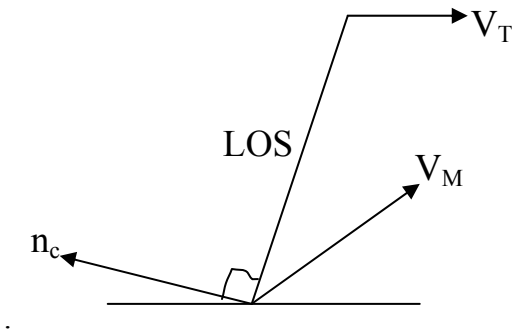


Figure 32 True PNG

Since the acceleration command is perpendicular to the instantaneous line of sight, the missile acceleration components in Earth coordinates can be found by trigonometry:

$$a_{M1} = -n_C \sin\lambda \tag{61}$$

$$a_{M2} = n_C \cos\lambda \tag{62}$$

3.3.2. Modified Classical PNG

Classical PNG rule is a subject of the LOS angle rate. When the distance from the target to the missile is very long, or the target is in static mode and the distance from the missile to the target is not changing the derivative of the LOS angle is meaningless, so the LOS angle itself can be added to the PNG rule as below:

$$n_c = N'V_c \dot{\lambda} + K_3\lambda \tag{63}$$

where K_3 is the LOS angle constant.

3.3.3. Gain Scheduled PNG

Since the derivative of the LOS angle is important for small distances where the LOS angle is important in large distances, a gain scheduling can be made by using three or more range values. The LOS angle and the navigation constants can be changed according to the range as shown in Table 2.

Table 2 Gain Scheduled PNG

	Near	**Middle**	**Far**	**For the unknown Range**
Navigation Constant	N'(1)	N'(2)	N'(3)	Interpolation
LOS Angle Constant	K(1)	K(2)	K(3)	Interpolation

3.3.4. Augmented Proportional Navigation

The line of sight angle can also be expressed as

$$\lambda = \frac{y}{R_{TM}} = \frac{y}{V_C(t_f - t)} \tag{64}$$

where y is the relative target-missile separation, and R_{TM} is the range from the missile to the target. The line of sight rate is,

$$\dot{\lambda} = \frac{y + \dot{y} t_{go}}{V_C t^2_{go}} \tag{65}$$

where t_{go} is the time to go till intercept and can be defined as

$$t_{go} = t_F - t \tag{66}$$

The proportional navigation guidance law can be defined mathematically as [2]

$$n_c = N' V_c \dot{\lambda} = \frac{N'(y + \dot{y} t_{go})}{t^2_{go}} \tag{67}$$

If the target maneuvers, the zero effort miss must be augmented by an additional term. The new equation becomes [2],

$$ZEM_{TGTMVR} = y + \dot{y} t_{go} + 0.5 n_T t^2_{go} \tag{68}$$

where n_T is the maneuver target acceleration level.

So, a perfectly plausible guidance law is [2],

$$n_{c.APN} = \frac{N' ZEM_{TGTMVR}}{t_{go}^2} = N' V_c \dot{\lambda} + \frac{N' n_T}{2} \tag{69}$$

3.3.5. Modified Augmented Proportional Navigation

Modified augmented navigation is a combination of augmented and modified navigation rules. It uses the LOS angle for the large distances, the LOS angle rate for small distances and the acceleration of the target for the moving targets.

$$n_{c.APN} = \frac{N' ZEM_{TGTMVR}}{t_{go}^2} = N' V_c \dot{\lambda} + \frac{N' n_T}{2} + K\lambda \tag{70}$$

3.3.6. Ideal PNG

Ideal PNG requires range information, so it is not suitable for IR applications, but for RF applications.

In all forms of PN-based interception schemes, command acceleration, ac, which is calculated based on the current states of the target and the interceptor, can be expressed in the following general form [18]:

$$a_c = \lambda L x \dot{\theta}_{los} \tag{71}$$

where θ is the angular velocity of the LOS angle, λ the navigation gain, and L is a vector that varies for different navigation laws. The L directions used by the two well-known guidance laws are defined below[18]:

Ideal proportional navigation guidance (IPNG):

$$L = \dot{r} e_r + r \dot{\theta} e_\theta \tag{72}$$

Pure proportional navigation guidance (PPNG):

$$L = -V_1 \text{ (interceptor velocity)} \tag{73}$$

Although, the IPNG may be of less practical use in missile guidance compared to the PPNG, its mathematical tractability and robustness to the initial conditions of the interceptor make it especially suitable for robotic interception [18].

The control input in an IPNG interception scheme, in an acceleration command form, is given as

$$a_{IPNG} = \lambda \dot{r} \, x \, \dot{\theta}_{los} \tag{74}$$

where

r : position difference vector between the target and the missile

λ : navigation gain

$\dot{\theta}_{LOS}$: angular velocity of the LOS angle

$\dot{\theta}_{LOS}$ can also be expressed as a function of r and $\dot{r}$ as follows,

$$\dot{\theta}_{LOS} = \left\{ \frac{r \quad x \quad \dot{r}}{|r|^2} \right\} \tag{75}$$

Substituting (75) into (74), one obtains [18]

$$a_{IPNG} = \frac{\lambda}{|r|^2} \left\{ \dot{r} \, x (r x \dot{r}) \right\} \tag{76}$$

Since $\dot{r} \, x (r x \dot{r}) = r(\dot{r} . \dot{r}) - \dot{r}(r . \dot{r})$,

$$a_{IPNG} = K_d(r, \dot{r}, \lambda)\dot{r} + K_p(r, \dot{r}, \lambda)r \tag{77}$$

where Kp and Kd are calculated as [18]

$$K_p(r, \dot{r}, \lambda) = \lambda\left(\frac{|\dot{r}|}{|r|}\right)^2 \tag{78}$$

$$K_d(r, \dot{r}, \lambda) = -\lambda\left(\frac{(r.\dot{r})}{|r|^2}\right) \tag{79}$$

3.3.7. Summary of the PNG Laws

PNG laws can be summarized as in Table 3.

Table 3 Summary of the PNG Laws

	IR Applications	**Long Distances**	**Small Distances**	**Range Information Necessity**	**Response to Target Movement**
Classical PNG	X		X		x
Modified PNG	X	X	X		x
Gain Scheduled PNG	X	X	X		x
Augmented PNG	X		X		X
Modified Aug. PNG	X	X	X		X
Ideal PNG		X	X	X	X

3.4. Scenarios

3.4.1. Static Target

Suppose that the target is at x=y=z=100 position. Missile's position under classical PNG and modified classical PNG algorithms are shown in figures below.

When the target is in the static mode, the augmented PNG is the same as the classical PNG; and the modified augmented PNG is the same as the modified classical PNG.

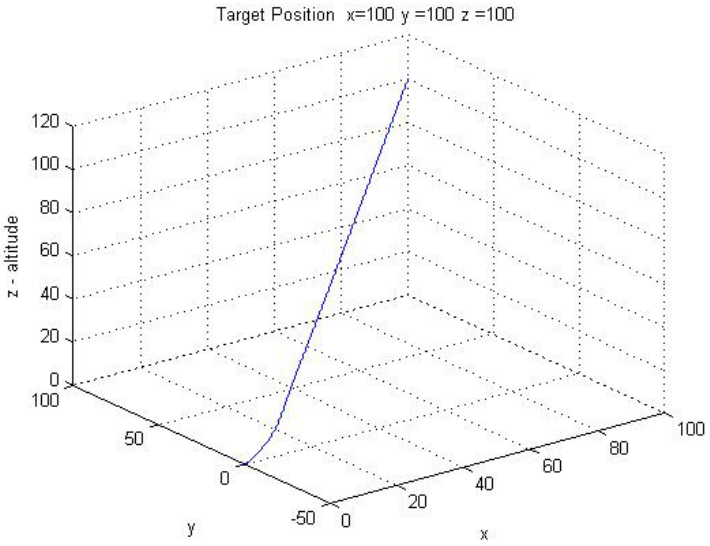


Figure 33 Classical PNG for Static Target in 3D

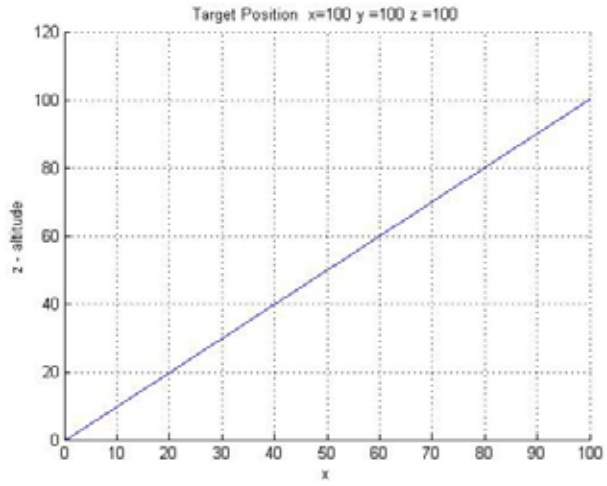


Figure 34 Classical PNG for Static Target in x – z

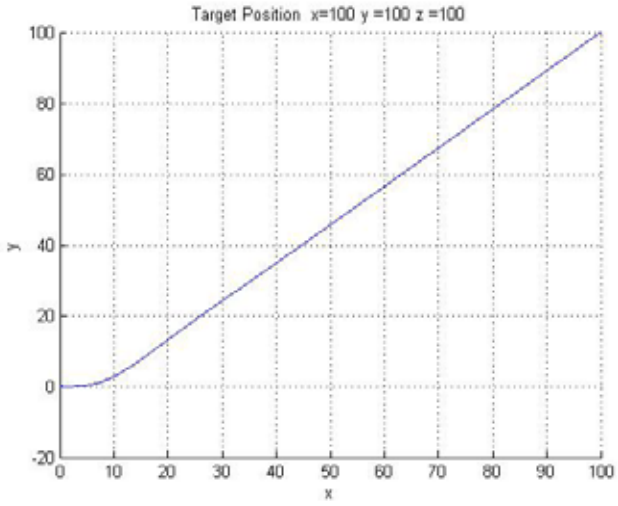


Figure 35 Classical PNG for Static Target in x – y

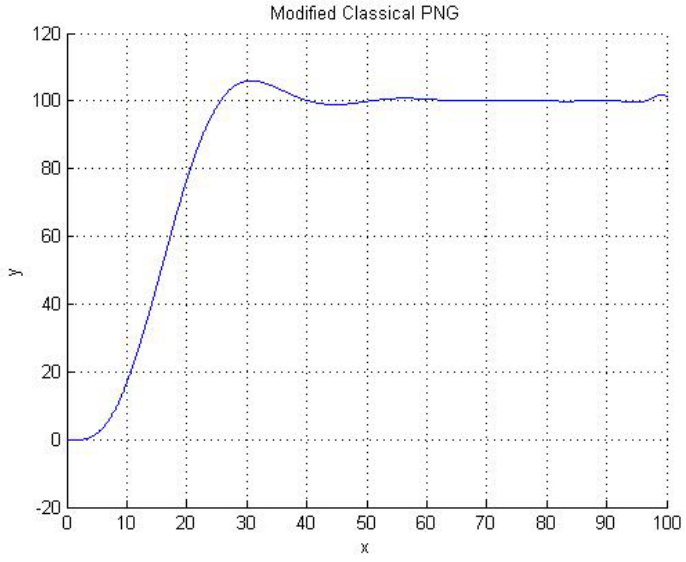


Figure 36 Modified Classical PNG for Static Target in x – y

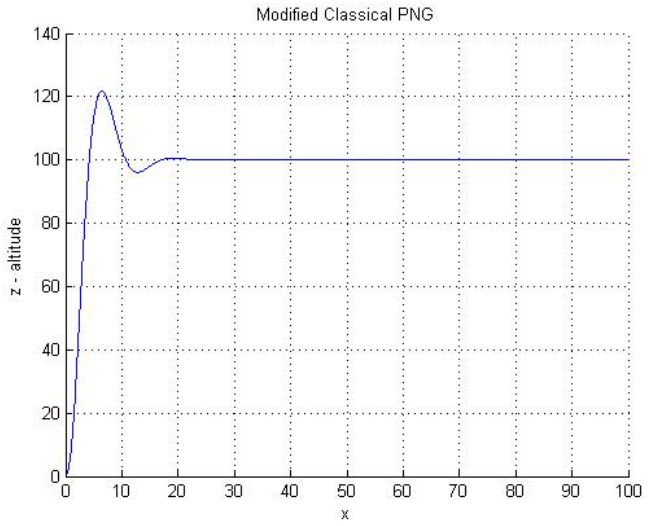


Figure 37 Modified Classical PNG for Static Target in x – z

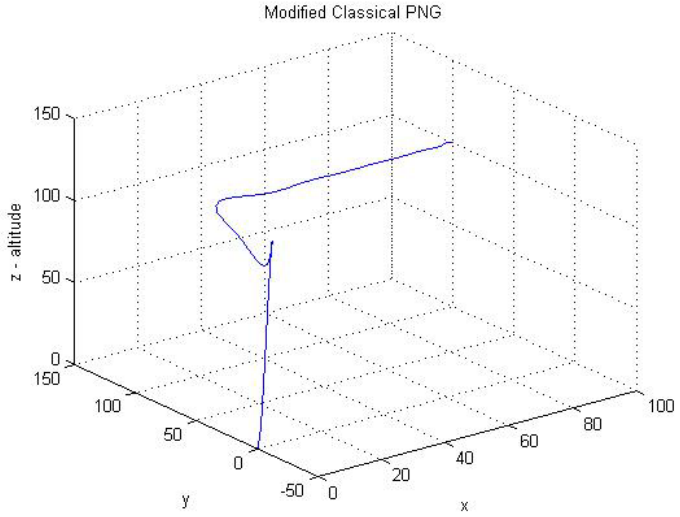


Figure 38 Modified Classical PNG for Static Target in 3D

3.4.2. Slow Moving Target

Target is at y=z=100 for the first 50 seconds, then move to y=200 and z=200 in 20 seconds. Target is at y=z=200 for the last 30 seconds of the movement.

Missile's position under different kinds of PNG algorithms are shown in figures below.

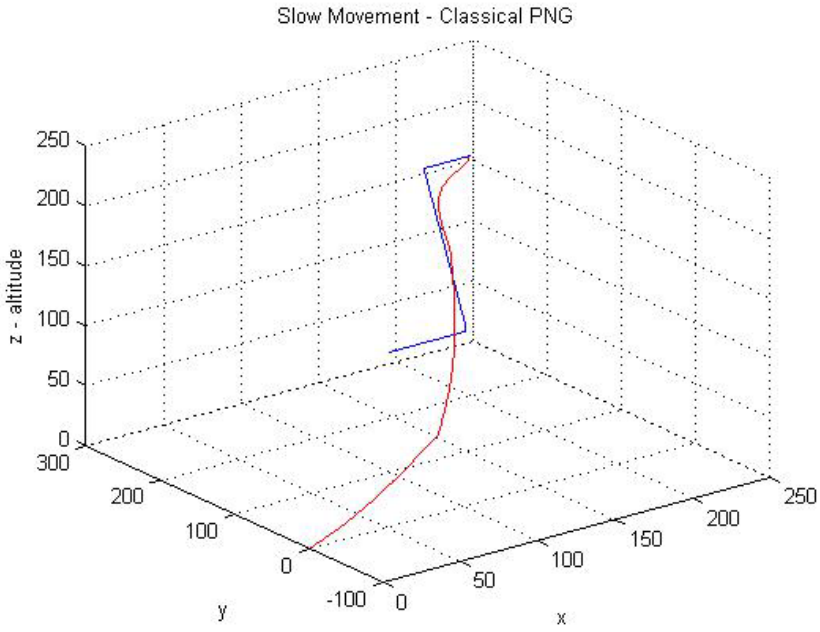


Figure 39 Classical PNG for Slow Moving Target in 3D
(Target is Blue and the Missile is Red)

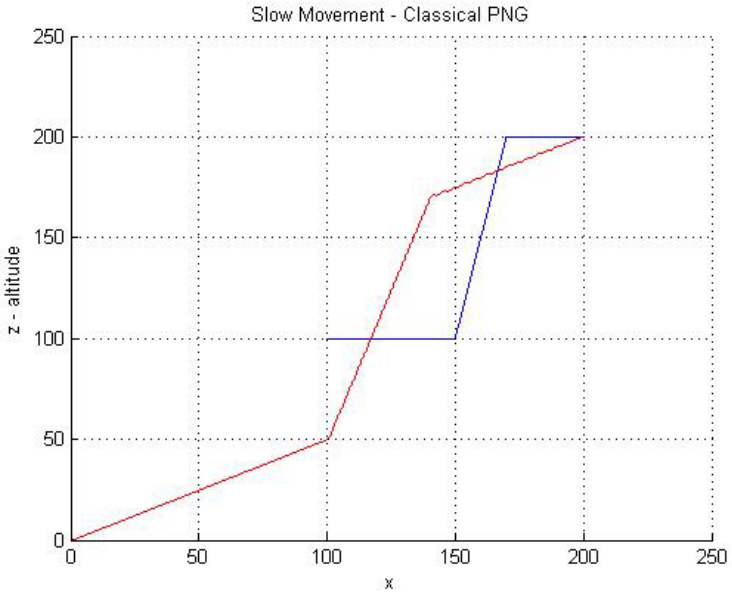


Figure 40 Classical PNG for Slow Moving Target in x – z
(Target is Blue and the Missile is Red)

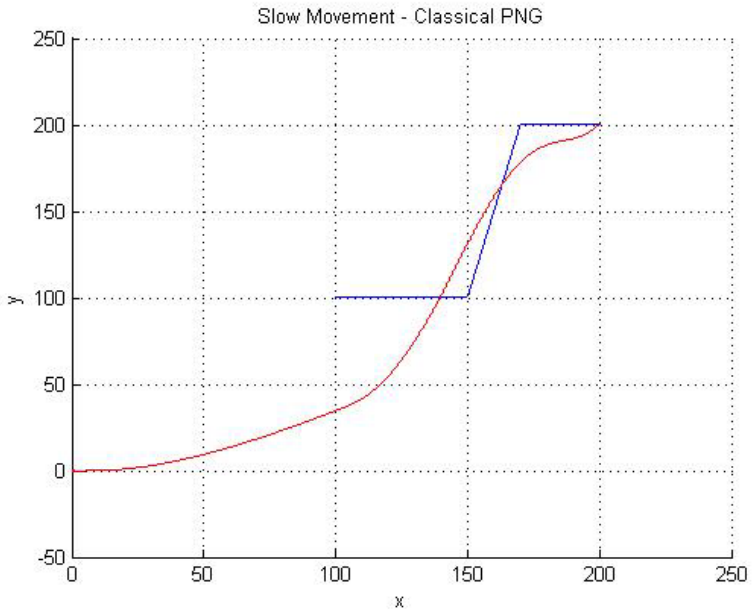


Figure 41 Classical PNG for Slow Moving Target in x – y
(Target is Blue and the Missile is Red)

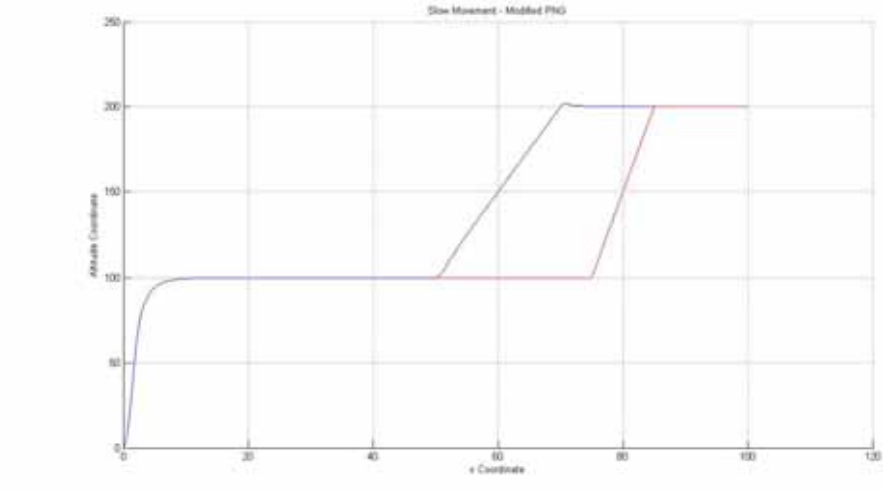


Figure 42 Modified Classical PNG for Slow Moving Target in x – z
(Target is Blue and the Missile is Red)

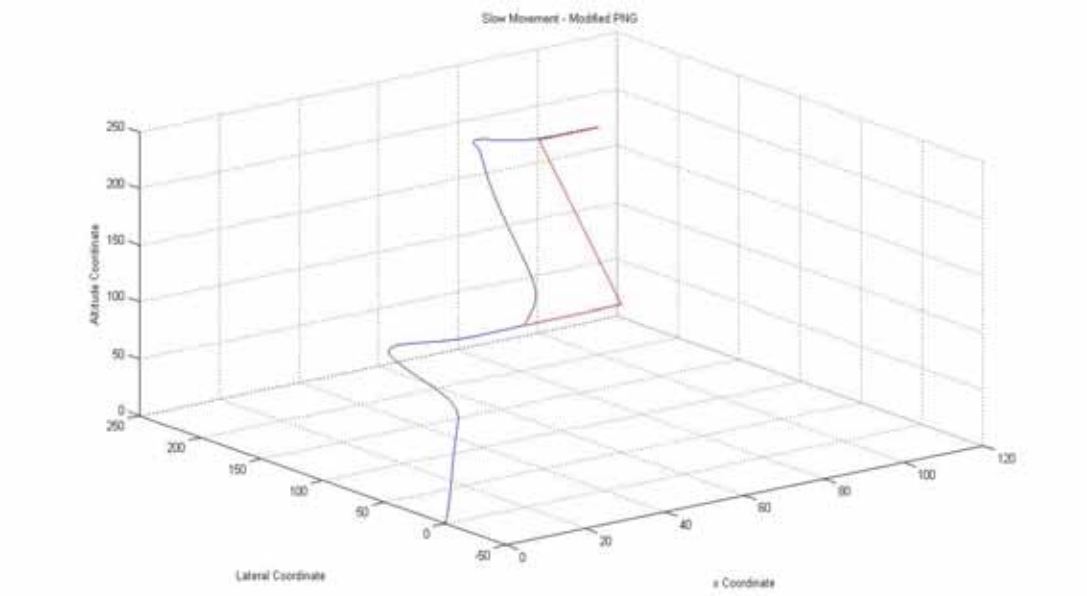


Figure 43 Modified Classical PNG for Slow Moving Target in 3D
(Target is Blue and the Missile is Red)

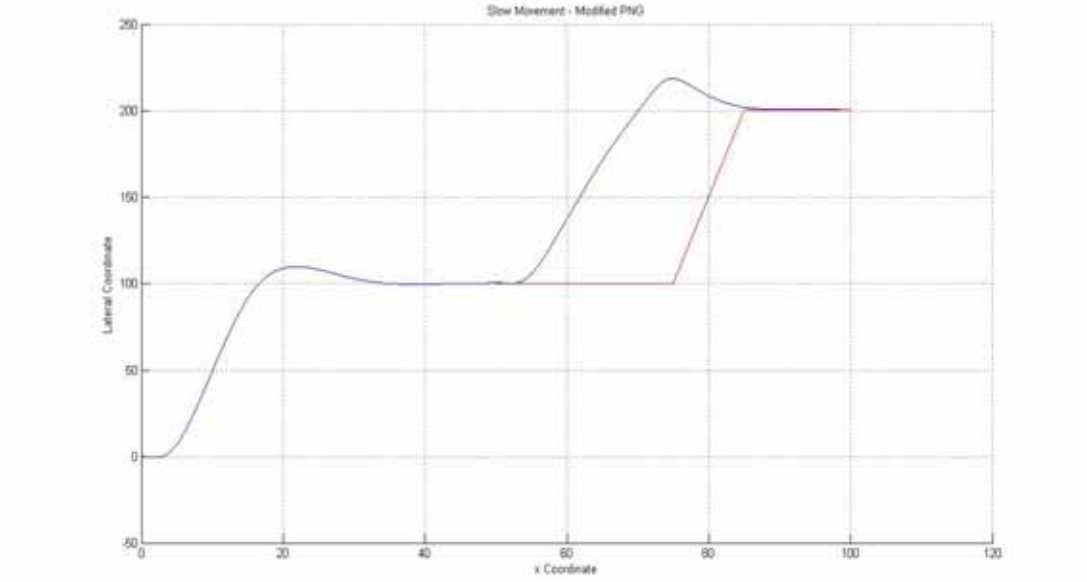


Figure 44 Modified Classical PNG for Slow Moving Target in x – y
(Target is Blue and the Missile is Red)

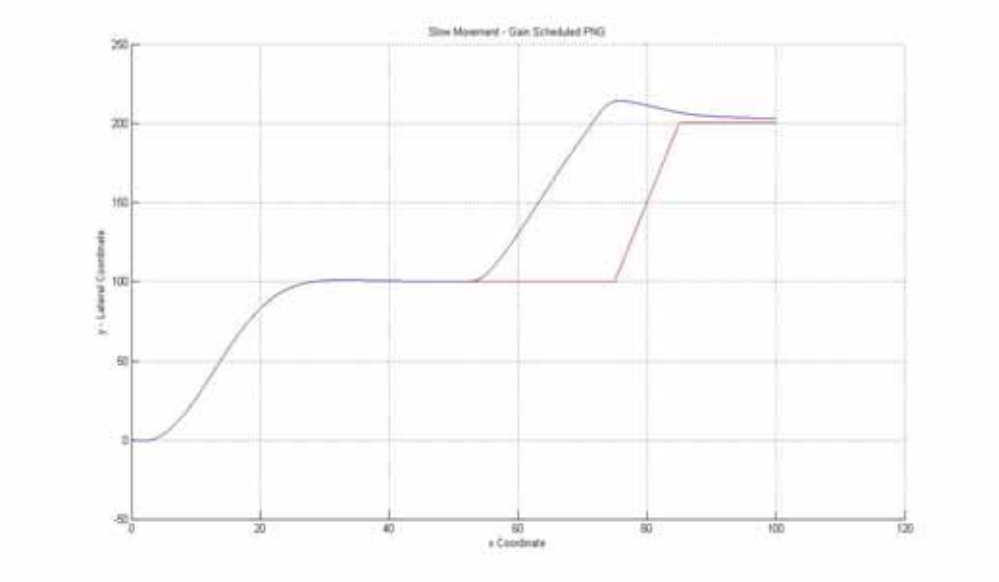


Figure 45 Gain Scheduled PNG for Slow Moving Target in x – y
(Target is Blue and the Missile is Red)

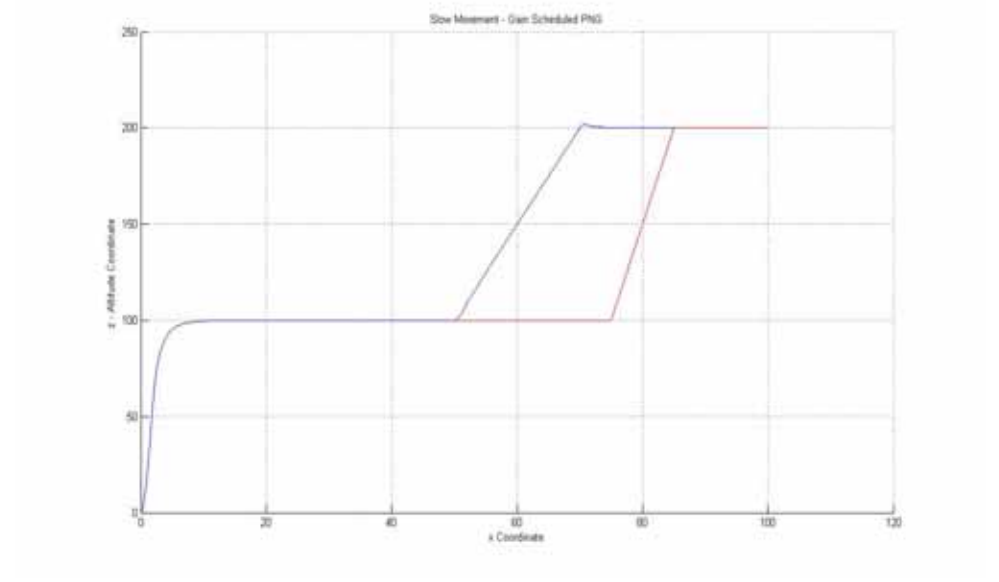


Figure 46 Gain Scheduled PNG for Slow Moving Target in x – z
(Target is Blue and the Missile is Red)

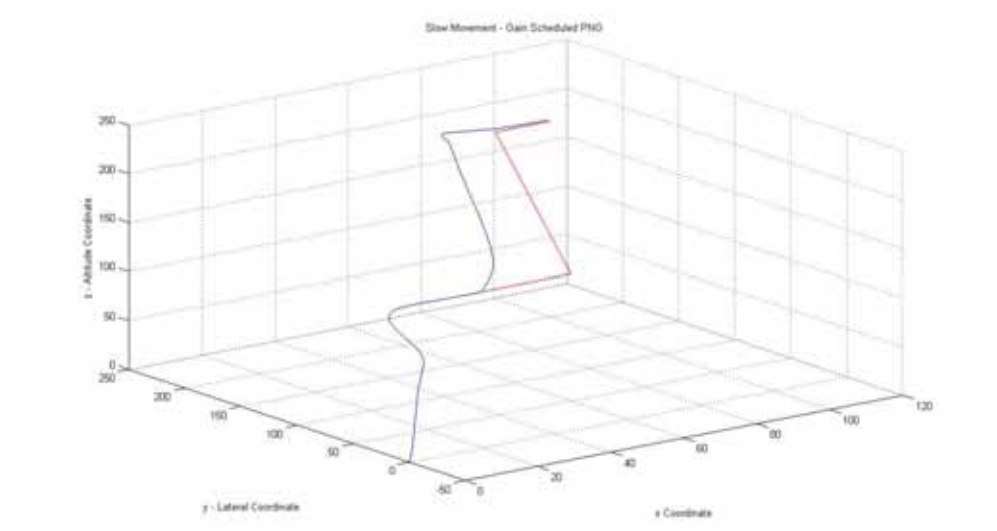


Figure 47 Gain Scheduled PNG for Slow Moving Target in 3D
(Target is Blue and the Missile is Red)

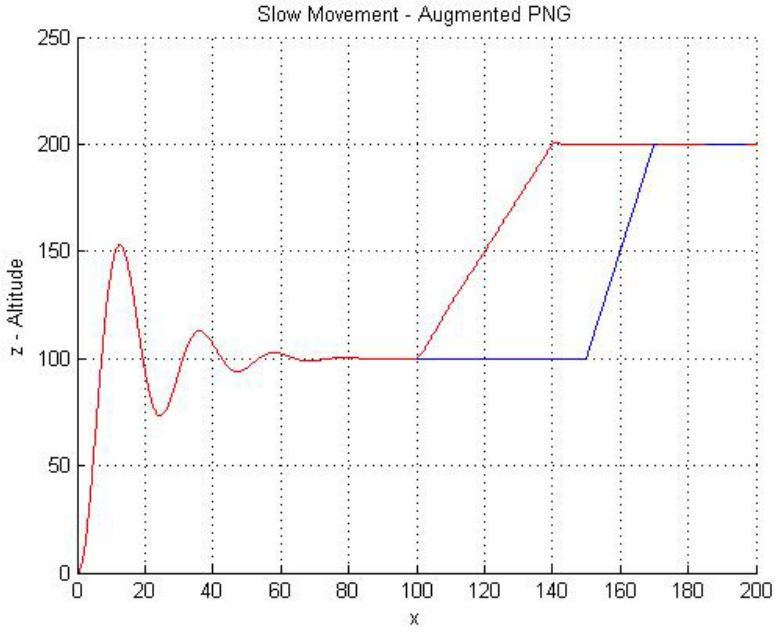


Figure 48 Augmented PNG for Slow Moving Target in x – z
(Target is Blue and the Missile is Red)

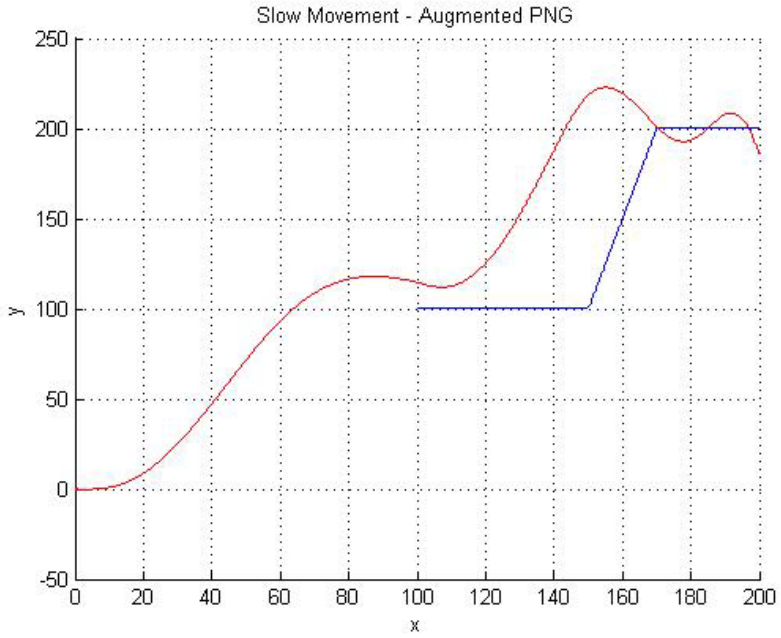


Figure 49 Augmented PNG for Slow Moving Target in x – y
(Target is Blue and the Missile is Red)

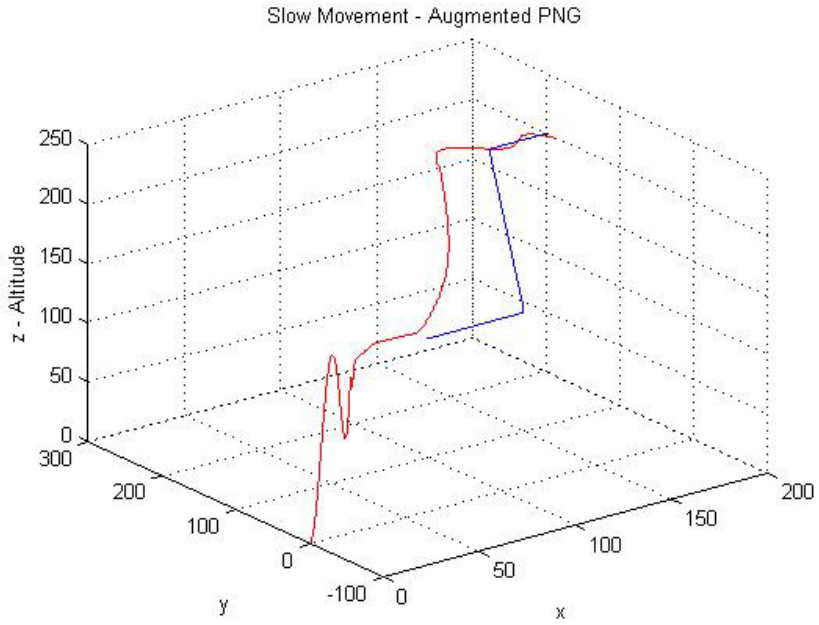


Figure 50 Augmented PNG for Slow Moving Target in 3D
(Target is Blue and the Missile is Red)

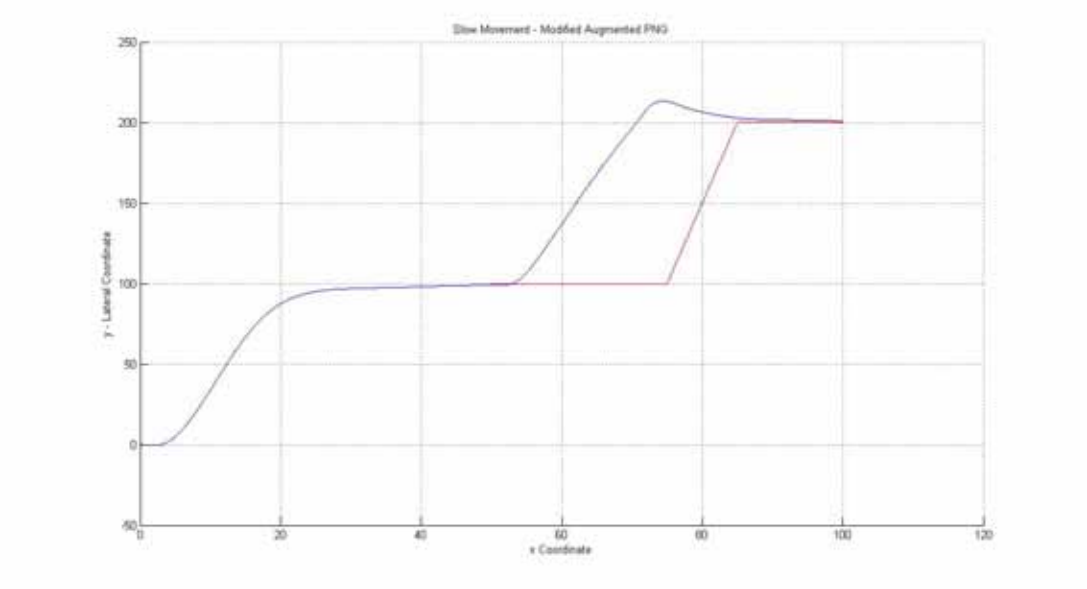


Figure 51 Modified Augmented PNG for Slow Moving Target in x – y (target is blue and the missile is red)

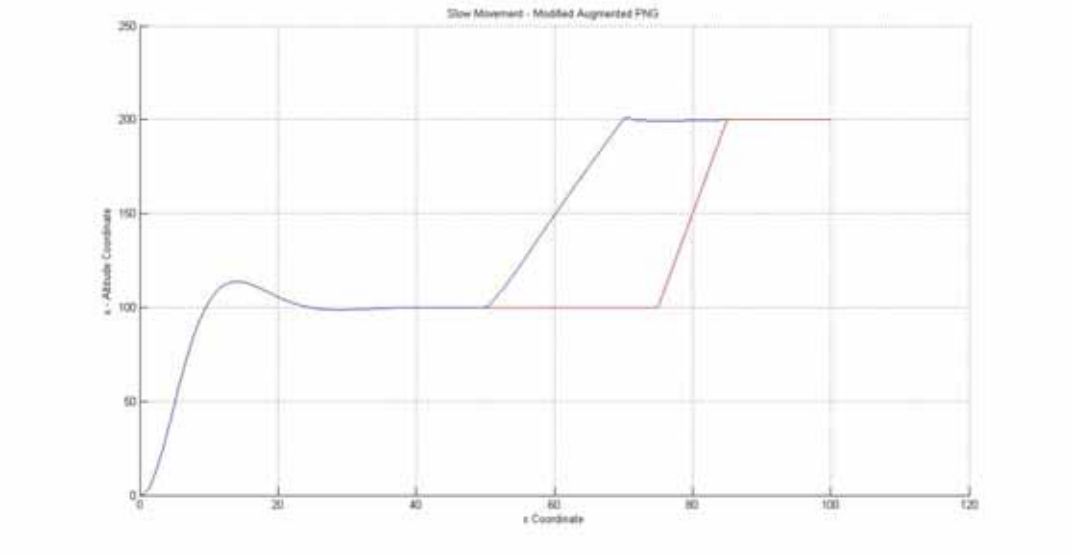


Figure 52 Modified Augmented PNG for Slow Moving Target in x – z (target is blue and the missile is red)

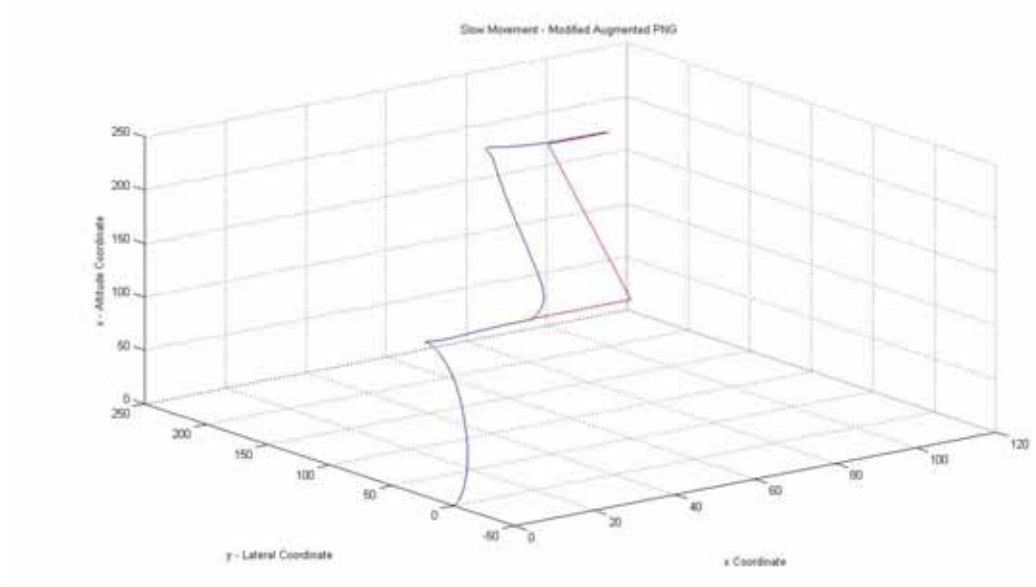


Figure 53 Modified Augmented PNG for Slow Moving Target in 3D
(target is blue and the missile is red)

The least square error denotes the position difference between the target movement and the missile's movement. Minimum least square error can be 0 (missile is exactly moving like the target) where the maximum can be infinity. For the performance analysis of the guidance algorithms, minimum least square error is choosen as zero and its performance score is 100, the maximum lease square error is choosen as the missile is always at the y=z=0 position Figure 54.

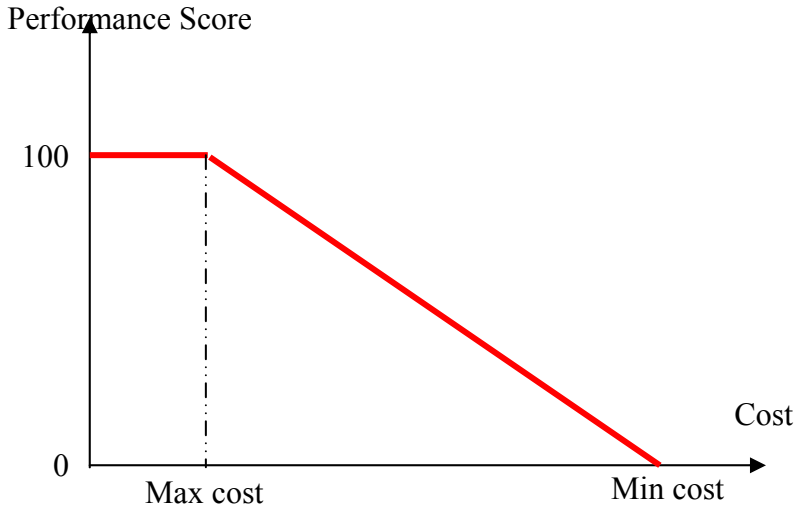


Figure 54 Performance Analysis

Performance analysis for the slow moving target is given in Table 4.

Table 4 Performance Scores for the Slow Moving Target

	PERFORMANCE SCORE (min=0, max=100)	
PNG Algorithm	**Yaw Plane**	**Pitch Plane**
Classical PNG	74.55	68.5
Modified Classical PNG	97.8	92.75
Augmented PNG	78.63	70.11
Modified Augmented PNG	97.82	92.89
Gain Scheduled PNG	97.98	93.84

3.4.3. Fast Moving Target

Target is at z=100 for the first 50 seconds, then move to z=200 in 2 seconds. z=200 for 8 seconds, then move to z=0 in 10 seconds. y plane movement is the same.

Missile's position under different kinds of PNG algorithms are shown in figures below.

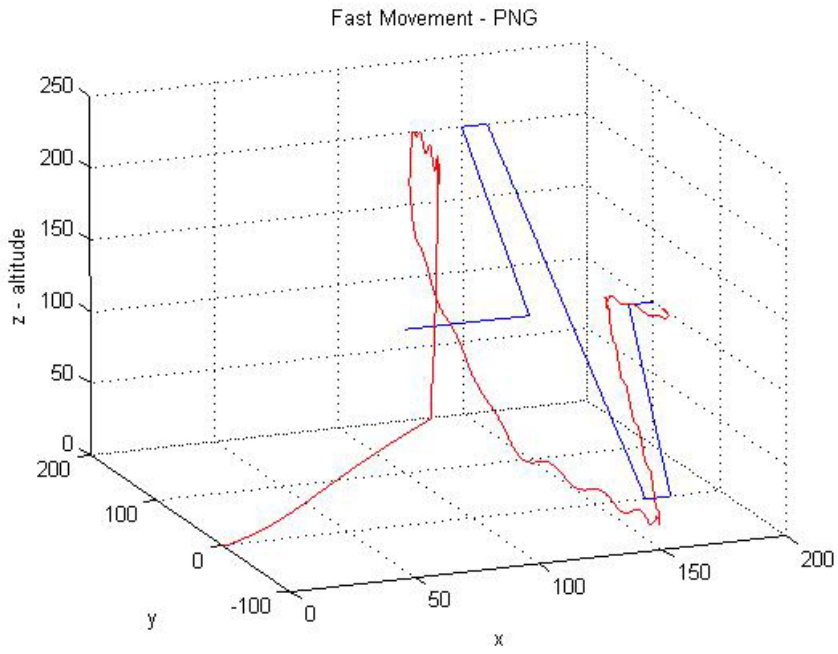


Figure 55 Classical PNG for Fast Moving Target in 3D
(target is blue and the missile is red)

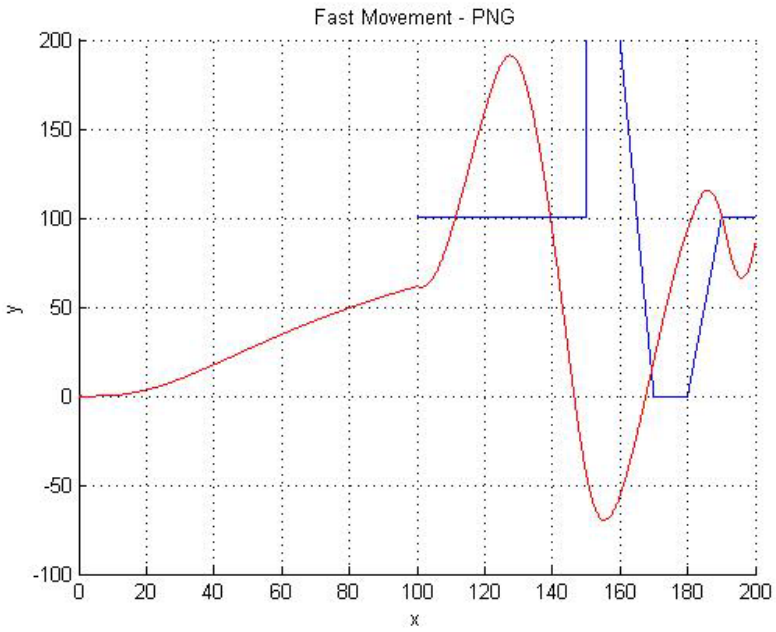


Figure 56 Classical PNG for Fast Moving Target in x – y (target is blue and the missile is red)

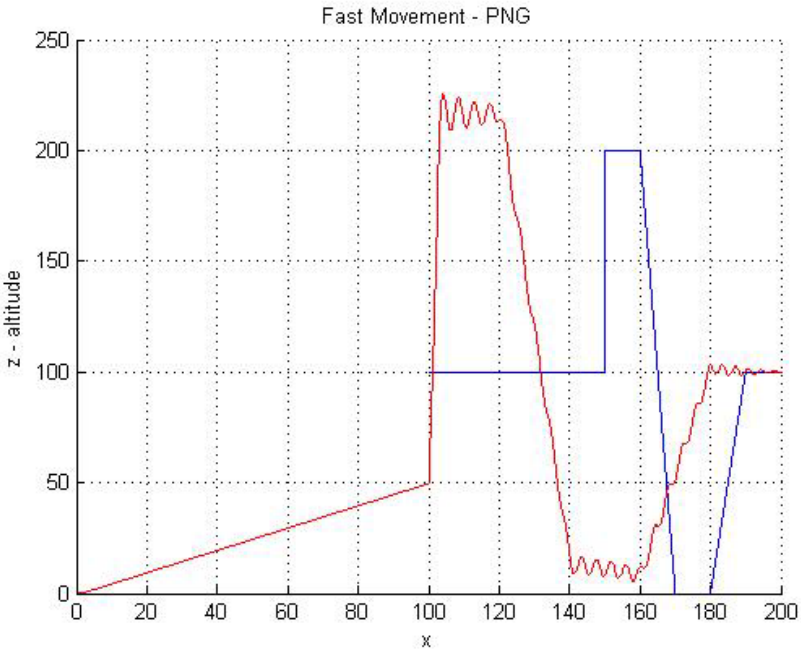


Figure 57 Classical PNG for Fast Moving Target in x – z (target is blue and the missile is red)

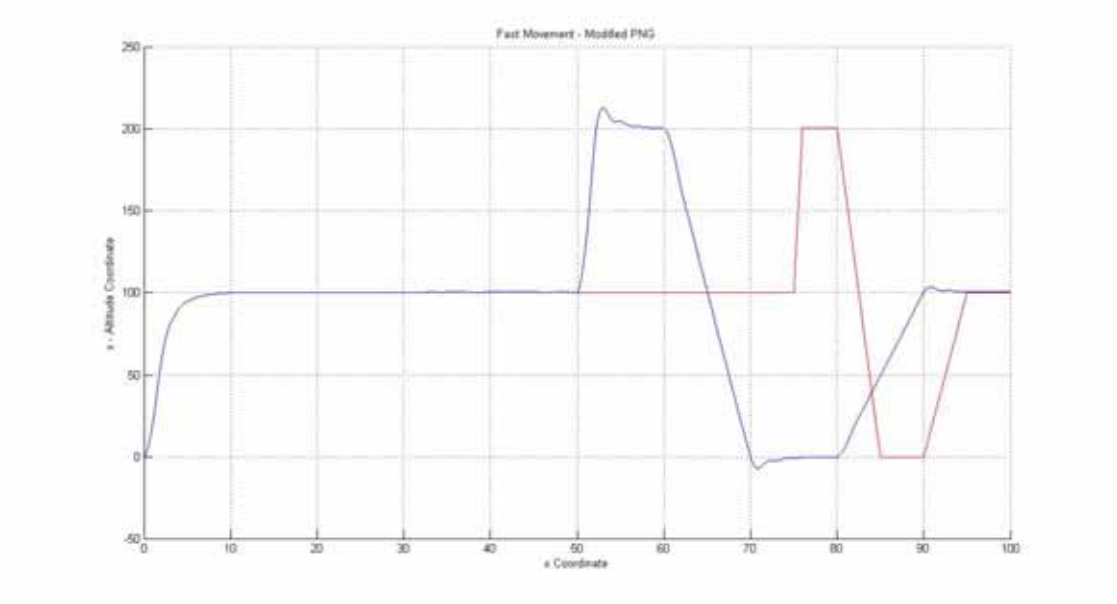


Figure 58 Modified Classical PNG for Fast Moving Target in x – z
(target is blue and the missile is red)

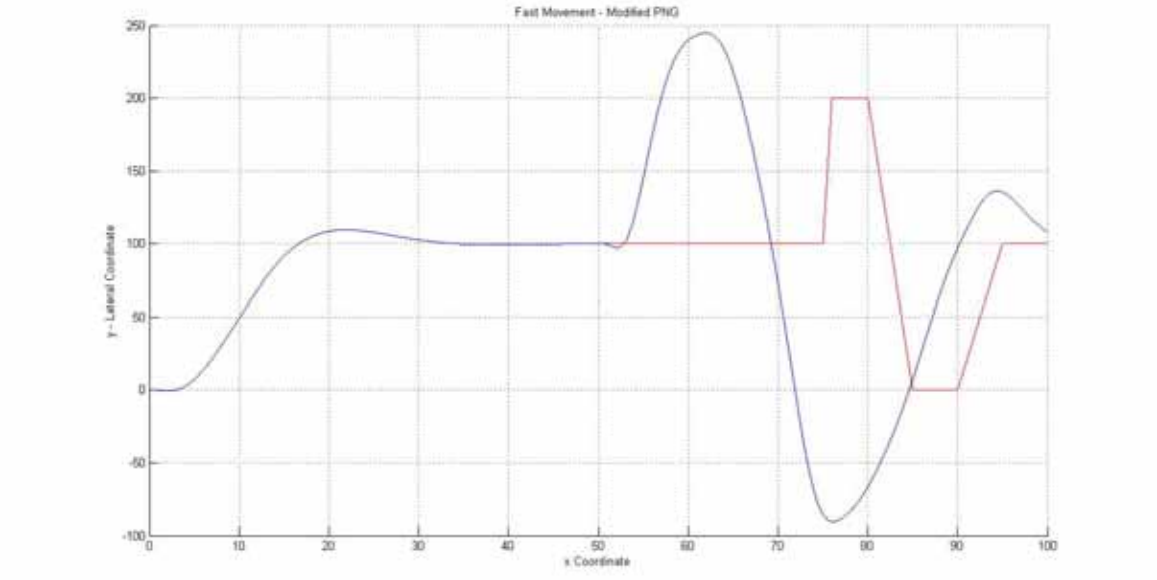


Figure 59 Modified Classical PNG for Fast Moving Target in x – y
(target is blue and the missile is red)

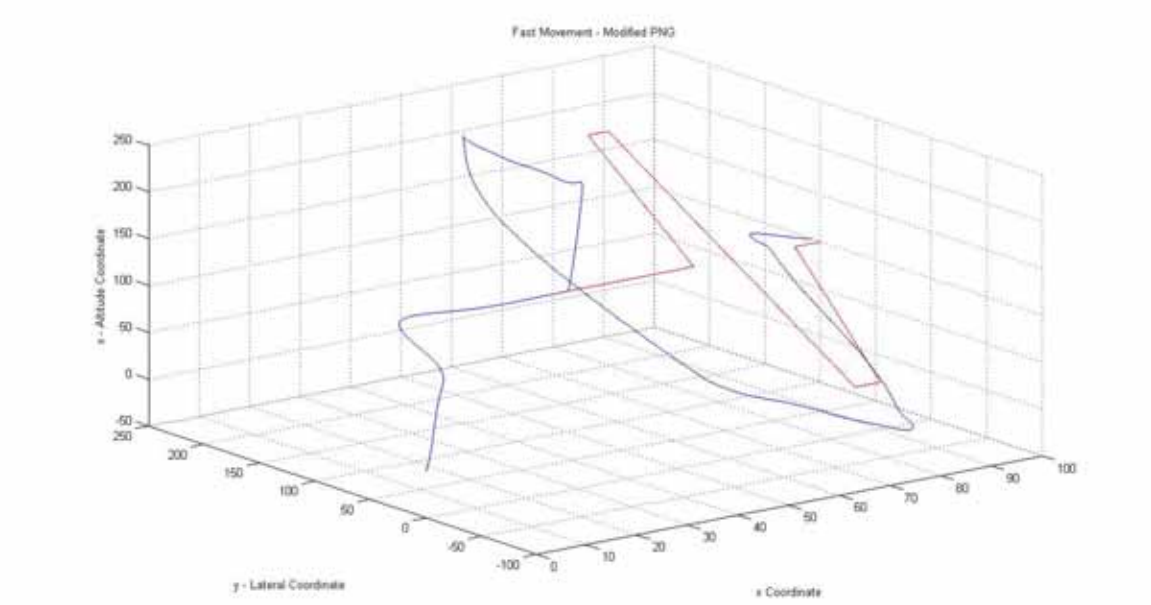


Figure 60 Modified Classical PNG for Fast Moving Target in 3D
(target is blue and the missile is red)

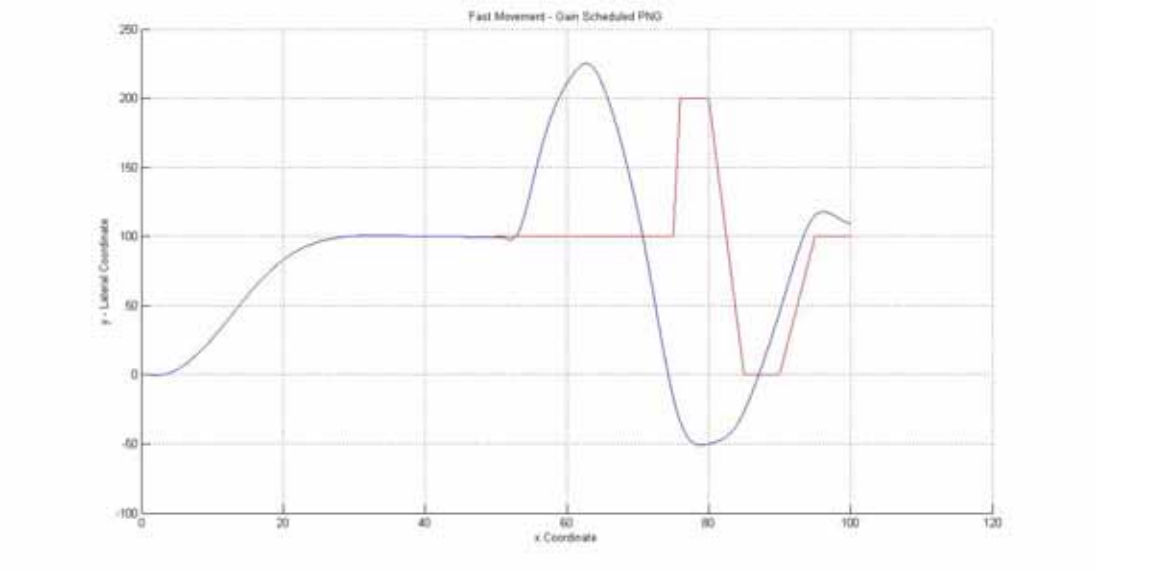


Figure 61 Gain Scheduled PNG for Fast Moving Target in x – y
(target is blue and the missile is red)

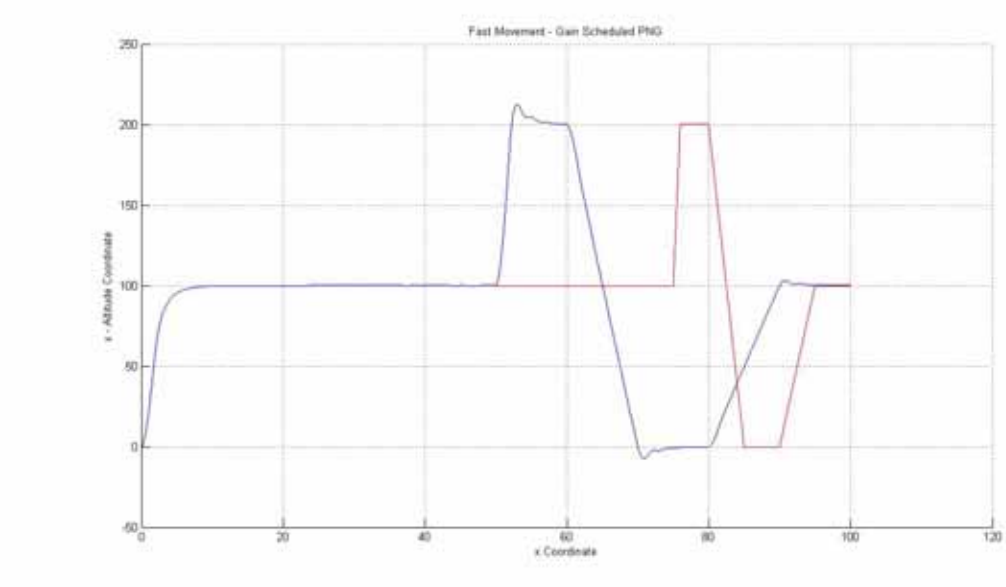


Figure 62 Gain Scheduled PNG for Fast Moving Target in x – z
(target is blue and the missile is red)

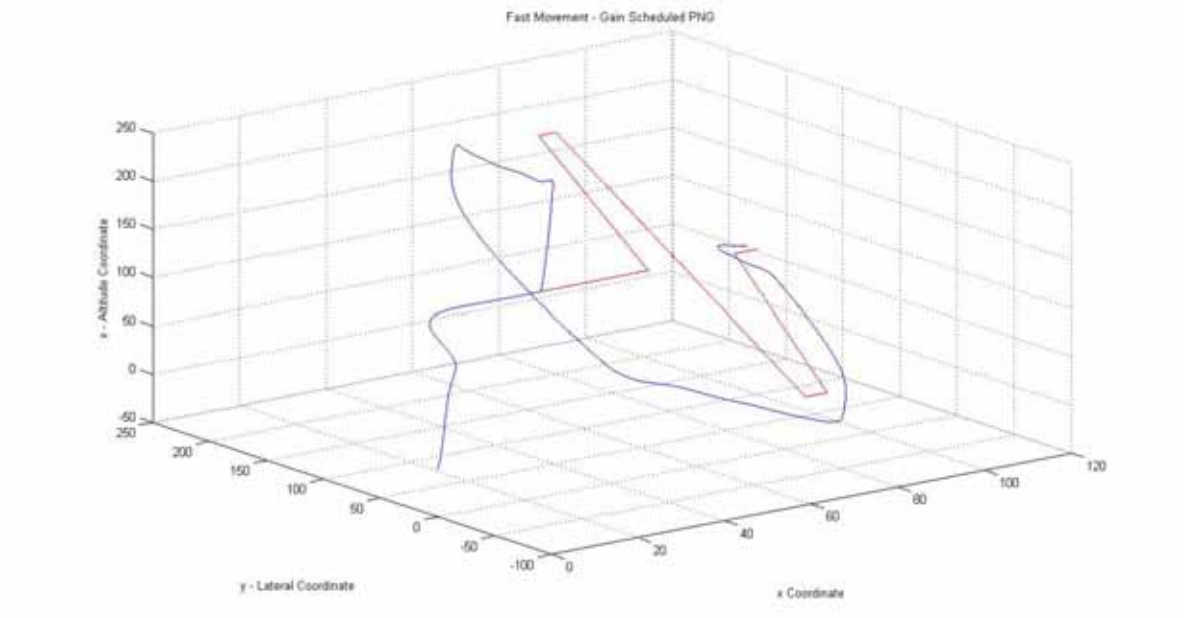


Figure 63 Gain Scheduled PNG for Fast Moving Target in 3D
(target is blue and the missile is red)

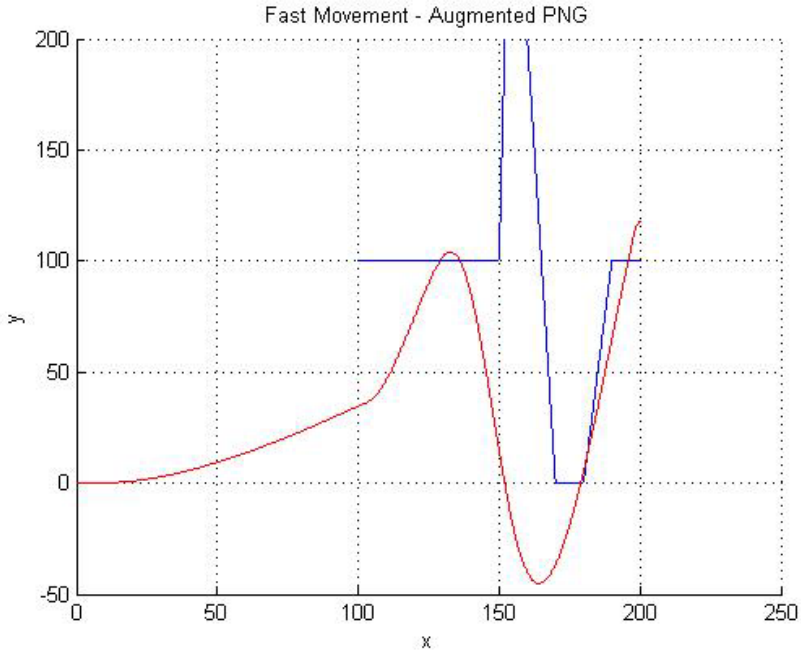


Figure 64 Augmented PNG for Fast Moving Target in x – y (target is blue and the missile is red)

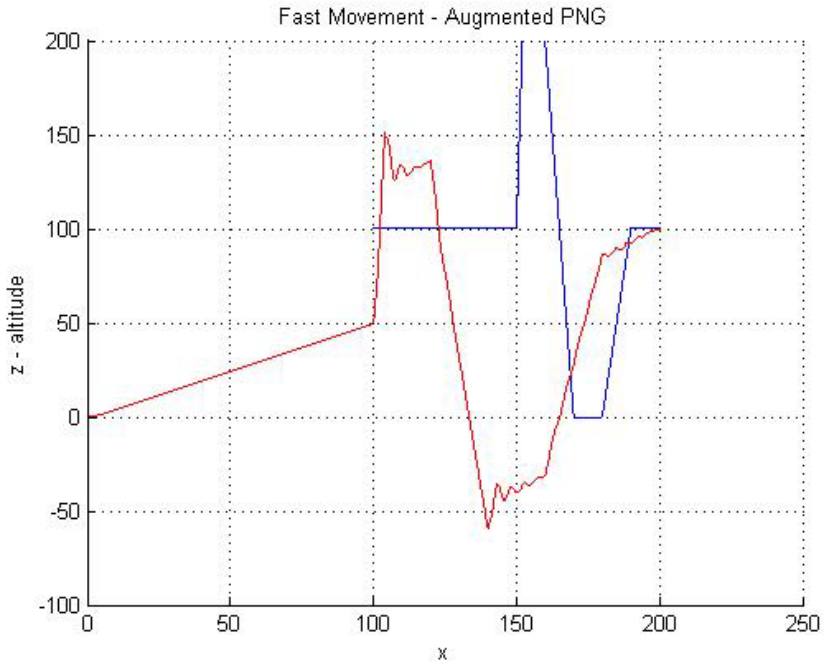


Figure 65 Augmented PNG for Fast Moving Target in x – z (target is blue and the missile is red)

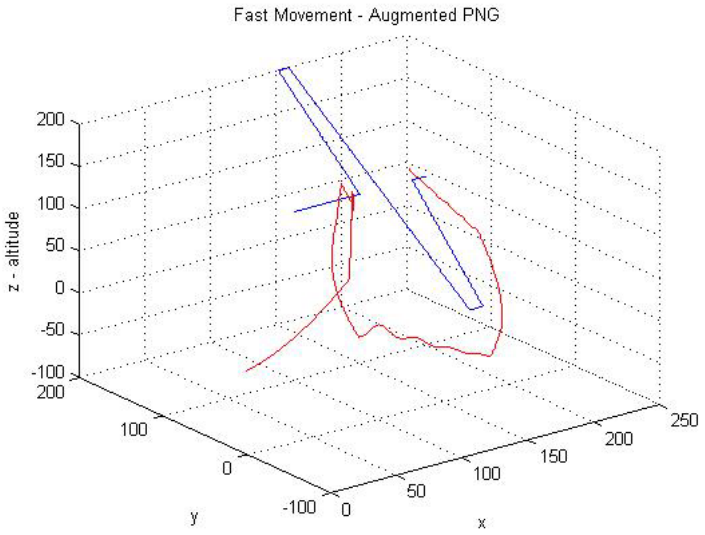


Figure 66 Augmented PNG for Fast Moving Target in 3D
(target is blue and the missile is red)

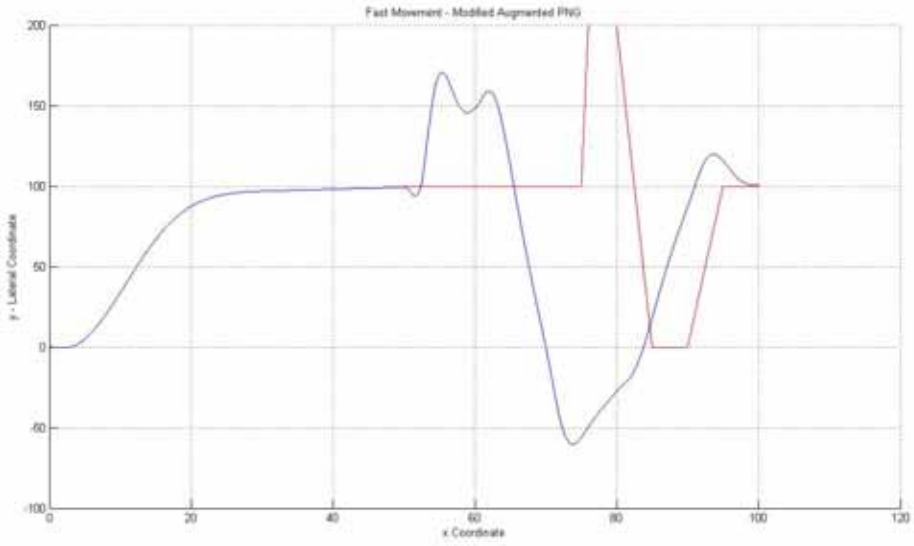


Figure 67 Modified Augmented PNG for Fast Moving Target in x – y
(target is blue and the missile is red)

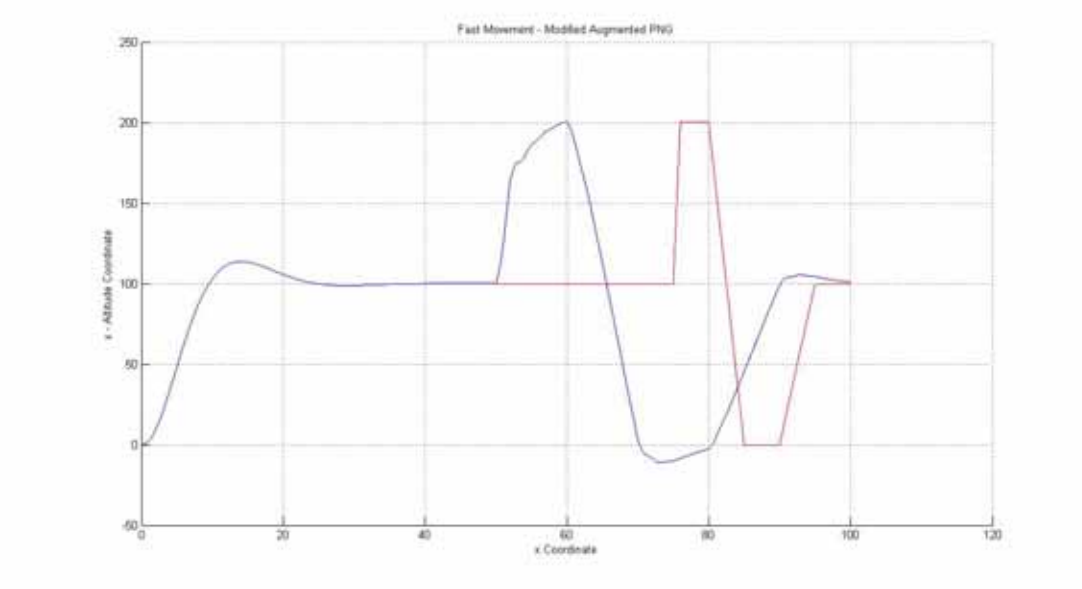


Figure 68 Modified Augmented PNG for Fast Moving Target in x – z (target is blue and the missile is red)

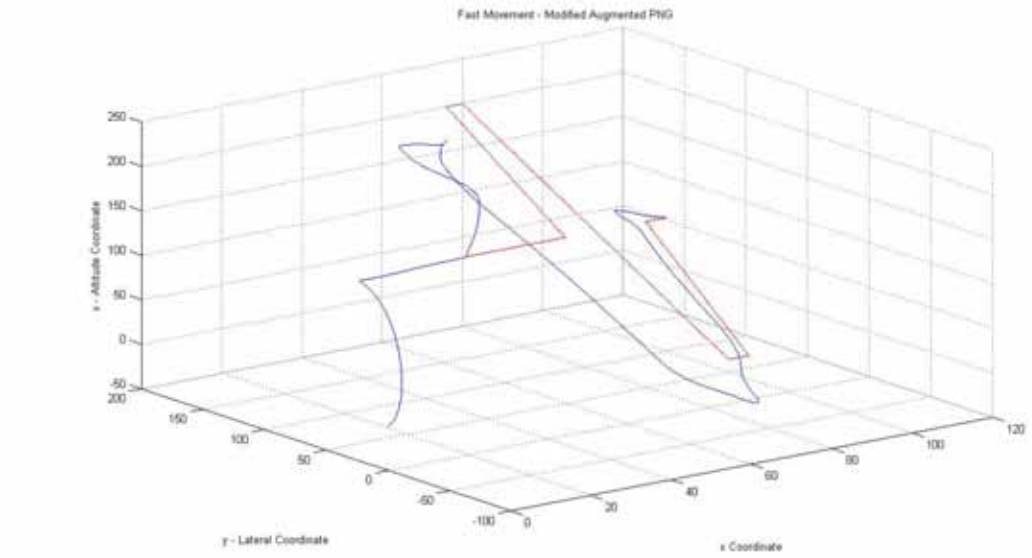


Figure 69 Modified Augmented PNG for Fast Moving Target in 3D (target is blue and the missile is red)

Performance score is given as discussed in Section 3.4.2.6. Performance analysis for the fast moving target is given in Table 5.

Table 5 Performance Scores for the Fast Moving Target

	PERFORMANCE SCORE (min=0, max=100)	
PNG Algorithm	**Yaw Plane**	**Pitch Plane**
Classical PNG	59.28	32.97
Modified Classical PNG	96.98	70.5
Augmented PNG	49.03	26.12
Modified Augmented PNG	97.14	73.80
Gain Scheduled PNG	97.96	75.26

3.4.4. Slow S Movement in Both Planes

Target movement is a sinusoidal movement in both planes with 0.05 Hz frequency and 10 meters amplitude.

Performance score is given as discussed in Section 3.4.2.6. Performance analysis for the fast moving target is given in Table 5.

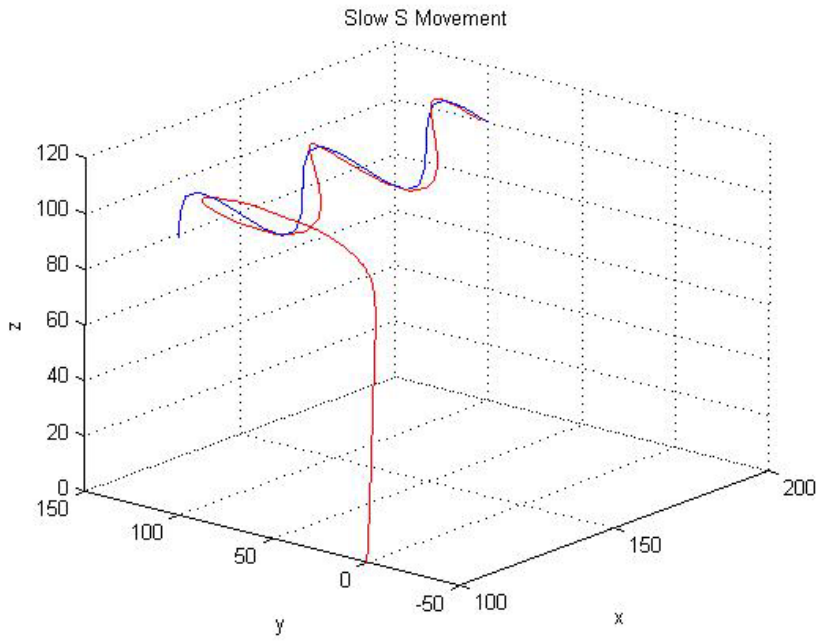


Figure 70 Slow S Movement (Target is Blue and the Missile is Red)

Table 6 Performance Score for the Slow S Movement

	PERFORMANCE SCORE (min=0, max=100)	
PNG Algorithm	**Yaw Plane**	**Pitch Plane**
Classical PNG	43.53	39.28
Modified Classical PNG	96.1001	86.4257
Augmented PNG	76.58	63.44
Modified Augmented PNG	96.8	86.87
Gain Scheduled PNG	97.24	87.23

3.4.5. Fast S Movement in Both Planes

Target movement is a sinusoidal movement in both planes with 0.05 Hz frequency and 1000 meters amplitude.

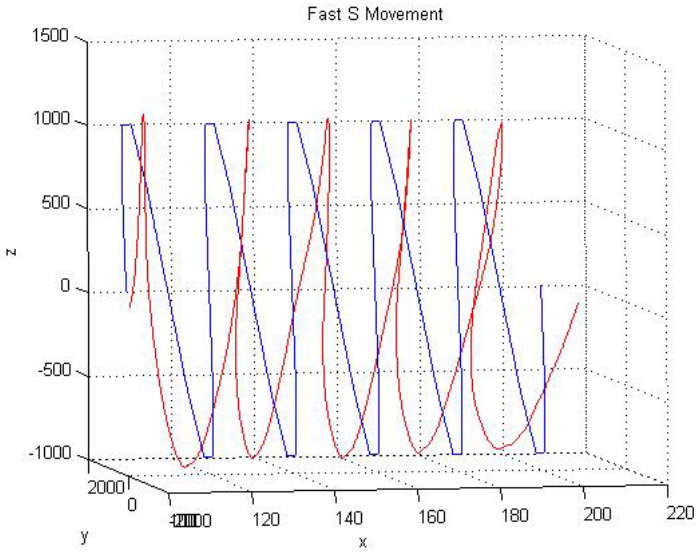


Figure 71 Fast S Movement (target is blue and the missile is red)

Performance score is given as discussed in Section 3.4.2.6. Performance analysis for the fast moving target is given in Table 7.

Table 7 Performance Score for Fast S Movement

	PERFORMANCE SCORE (min=0, max=100)	
PNG Algorithm	**Yaw Plane**	**Pitch Plane**
Classical PNG	32.25	22.41
Modified Classical PNG	84.16	68.21
Augmented PNG	56.42	35.64
Modified Augmented PNG	84.94	69.47
Gain Scheduled PNG	86.2	71.28

3.4.6. Circular Movement

Target is moving on a circular path in 100 meters height.

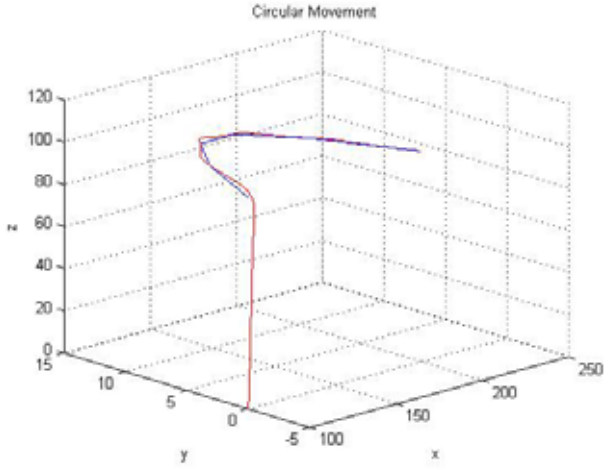


Figure 72 Circular Movement of the Target in x – y
(target is blue and the missile is red)

Performance score is given as discussed in Section 3.4.2.6. Performance analysis for the fast moving target is given inTable 8.

Table 8 Performance Score for the Circular Movement

	PERFORMANCE SCORE (min=0, max=100)	
PNG Algorithm	**Yaw Plane**	**Pitch Plane**
Classical PNG	98.7	98.5
Modified Classical PNG	99.54	99.305
Augmented PNG	98.9	98.7
Modified Augmented PNG	99.75	99.412
Gain Scheduled PNG	99.86	99.687

3.4.7. Target is Dispensing IR Countermeasures

The term “countermeasure” in formal literature is used to define an action taken to offset another action [12]. Countermeasure in IR band means preventing the weapon or platform against infrared guided missiles.

There are several ways for IR countermeasuring which are already in use throughout the world. Active countermeasures like ALQ-144 (Figure 73) and DIRCM (Figure 74) are used for confusing an infrared guided missile which is tracking a platform by adding coded infrared signal to the platform’s own infrared signal. Passive

countermeasures, known as flares, are burning chemicals which are used for hiding the protected platform from infrared guided missile's seeker. Flares are "very hot fireworks" dispensed from air vehicles (Figure 75).General spectral specification of a flare is shown in Figure 76 [13]. Main structure of a flare is shown in Figure 77 .

Figure 73 Active IR Countermeasure Example – ALQ-144 [20]

Figure 74 Active Countermeasure Example 2 – DIRCM [20]

Figure 75 Flares Ejected by a C-130 Hercules Military Transport [21]

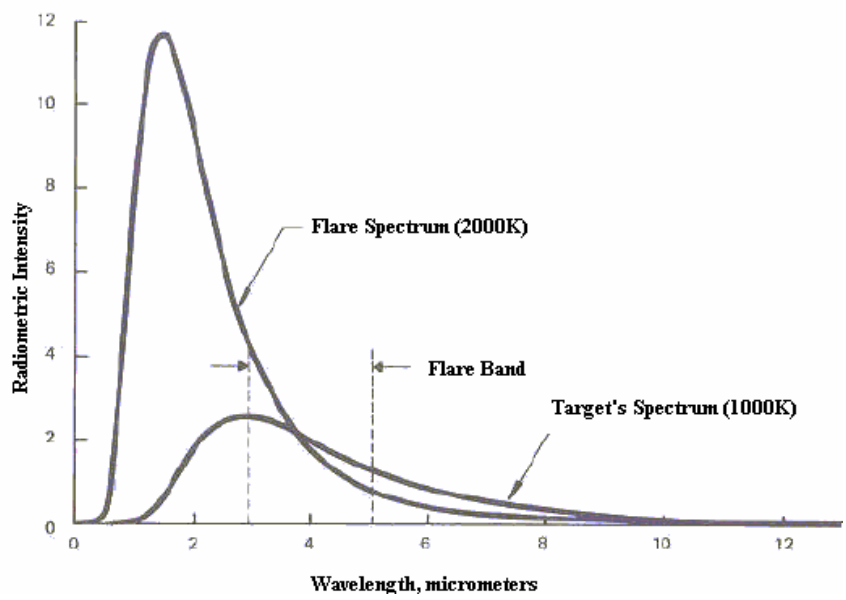


Figure 76 Spectral Specification of a Flare [3]

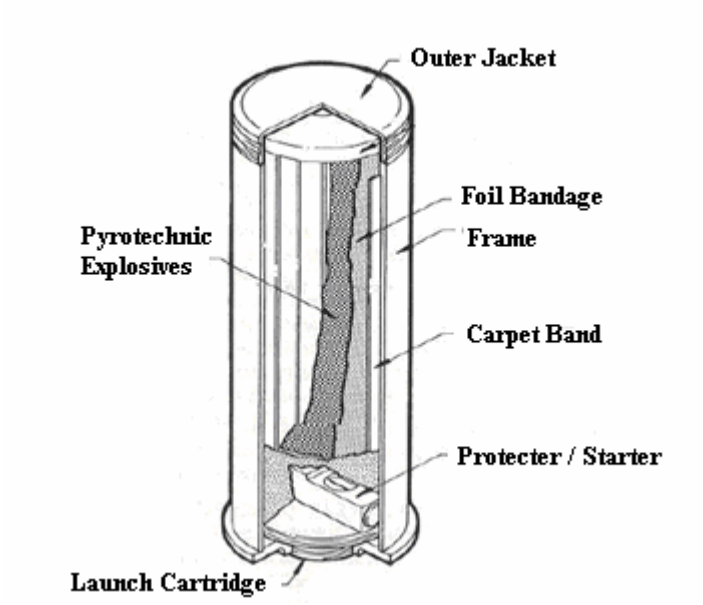


Figure 77 Main Structure of a Flare [3]

Trajectory of a flare can be estimated by using Figure 78. Angle of the dispensing system determines the values of V_x, V_y, and V_z values. If the drag coefficient is taken as zero, this is the worst case for the missile, because the flare moves faster and goes faraway from the target.

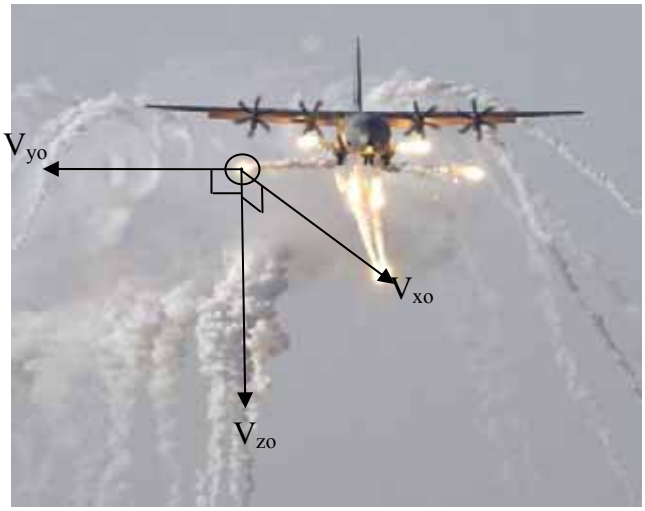


Figure 78 Flare Trajectory Analysis

V_{z0} is zero, V_{y0} is 50 meter/second and V_{x0} is 320 meter/second in the simulations.

3.4.7.1. Modified Classical PNG

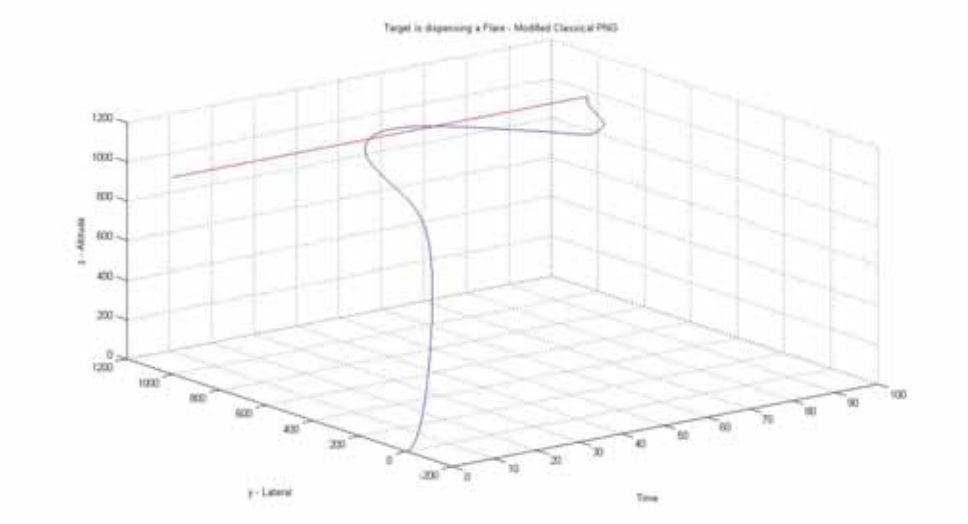


Figure 79 Target is Dispensing a Flare From Left – Modified Classical PNG (3D)
(target is blue and the missile is red)

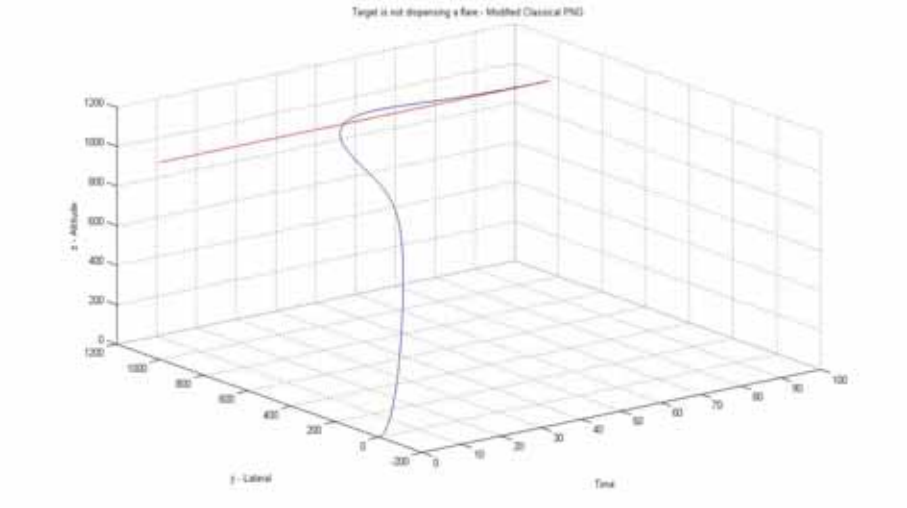


Figure 80 No Flare – Modified Classical PNG (3D)
(target is blue and the missile is red)

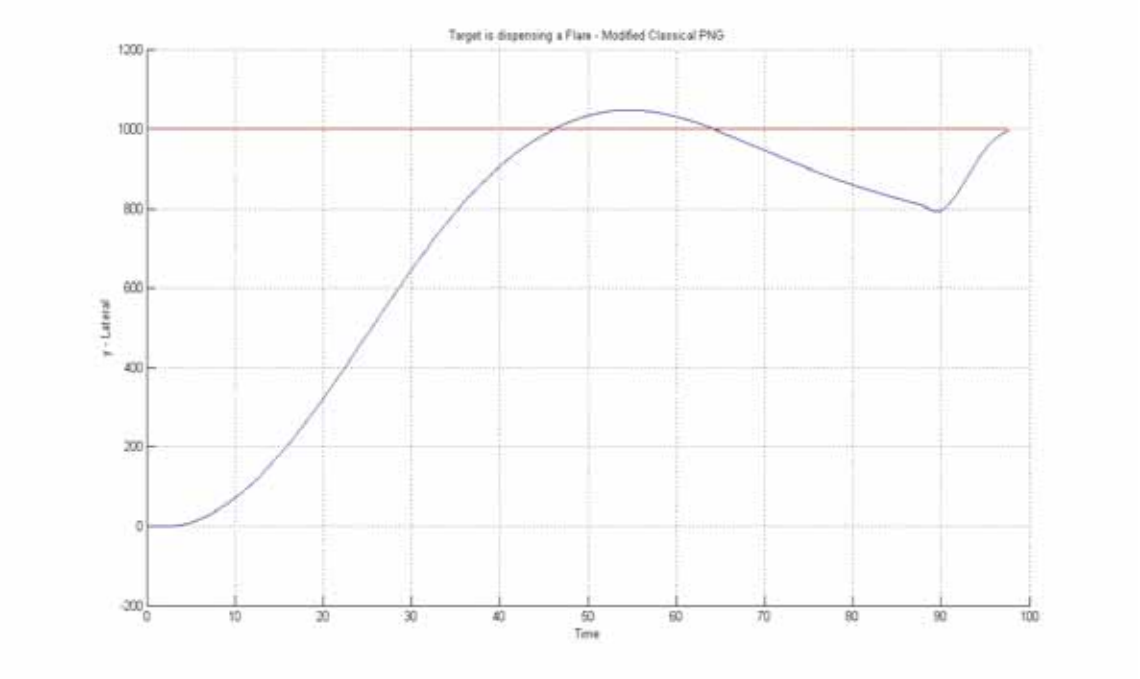


Figure 81 Target is Dispensing a Flare From Left – Modified Classical PNG (x – y)
(target is blue and the missile is red)

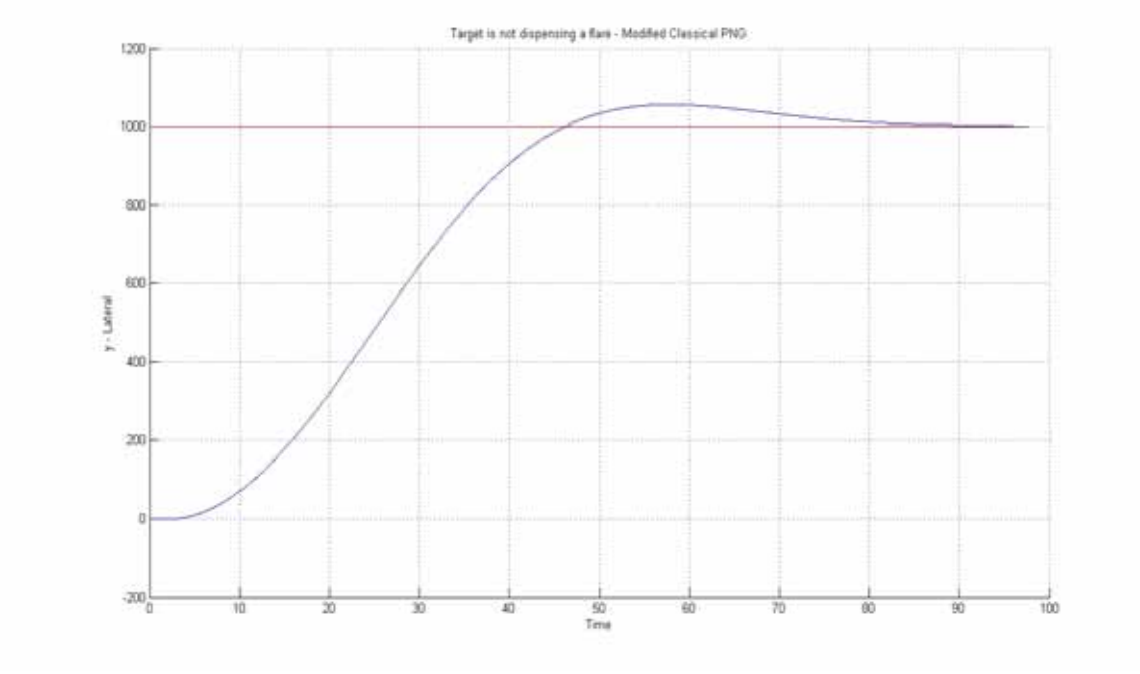


Figure 82 No flare – Modified Classical PNG (x – y)
(target is blue and the missile is red)

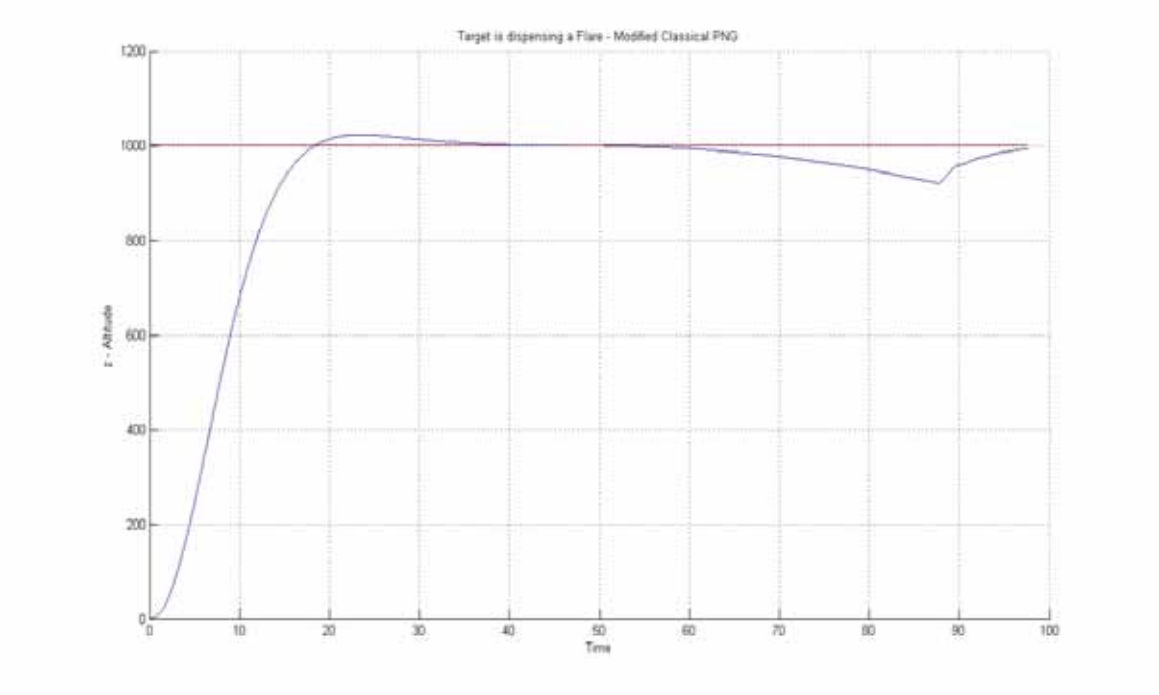


Figure 83 Target is Dispensing a Flare From Left – Modified Classical PNG (x – z)
(target is blue and the missile is red)

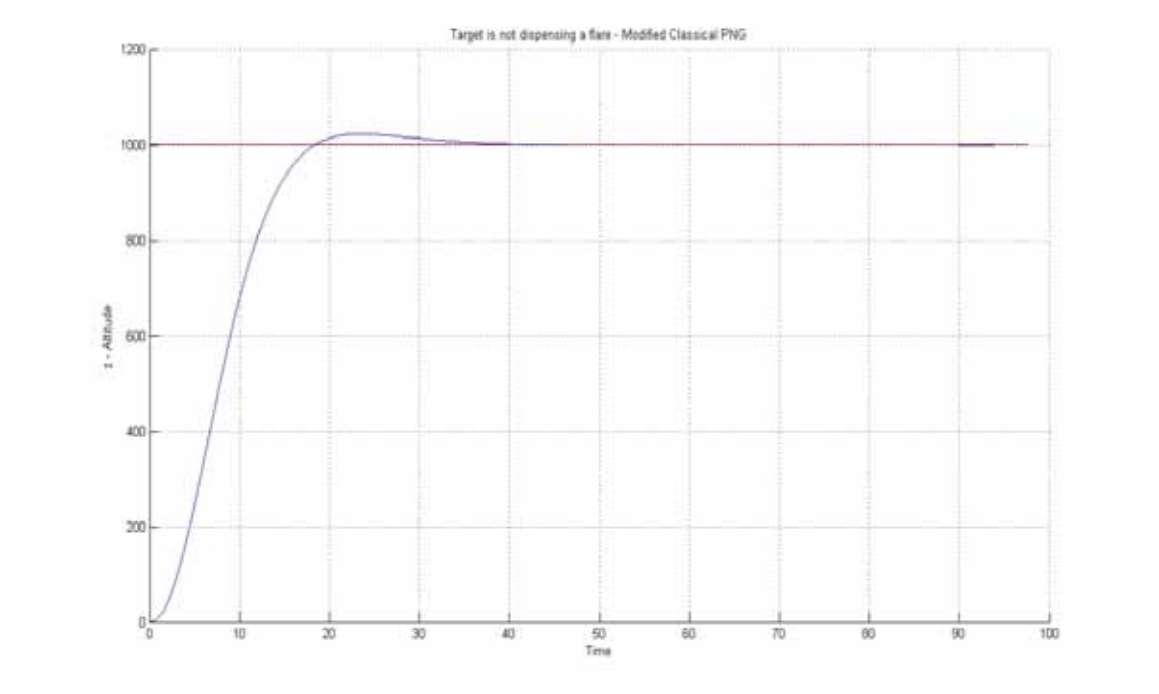


Figure 84 No flare – Modified Classical PNG (x – z)
(target is blue and the missile is red)

3.4.7.2. Modified Augmented PNG

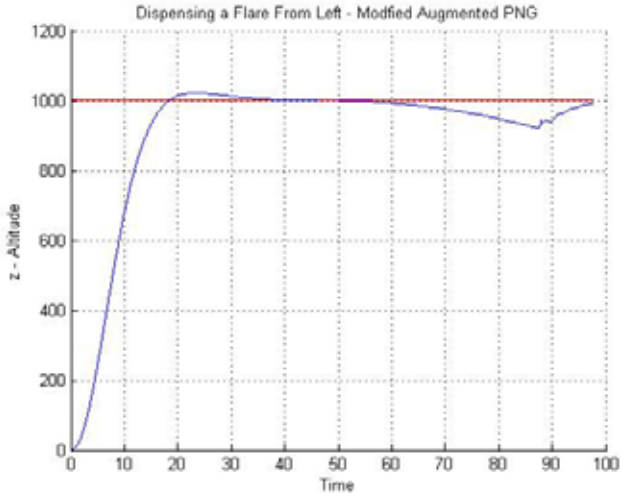


Figure 85 Dispensing a Flare From Left – Modified Augmented PNG (x – z) (target is blue and the missile is red)

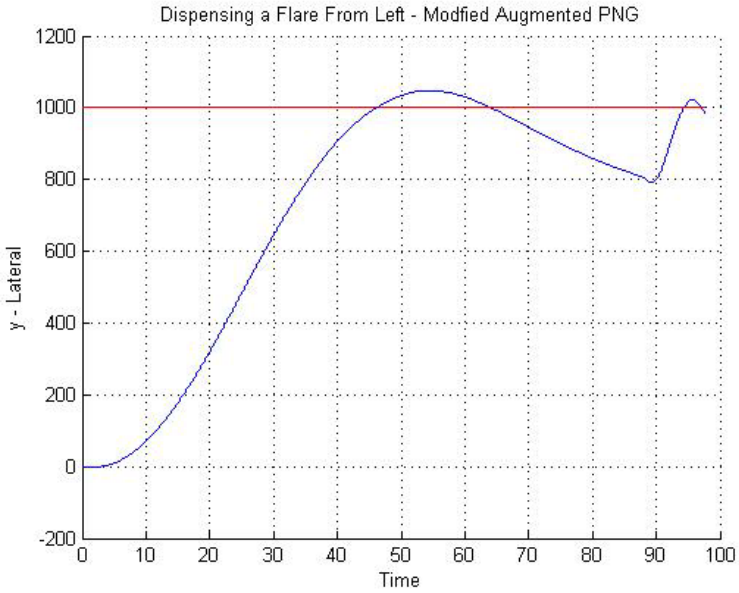


Figure 86 Dispensing a Flare From Left – Modified Augmented PNG (x – y) (target is blue and the missile is red)

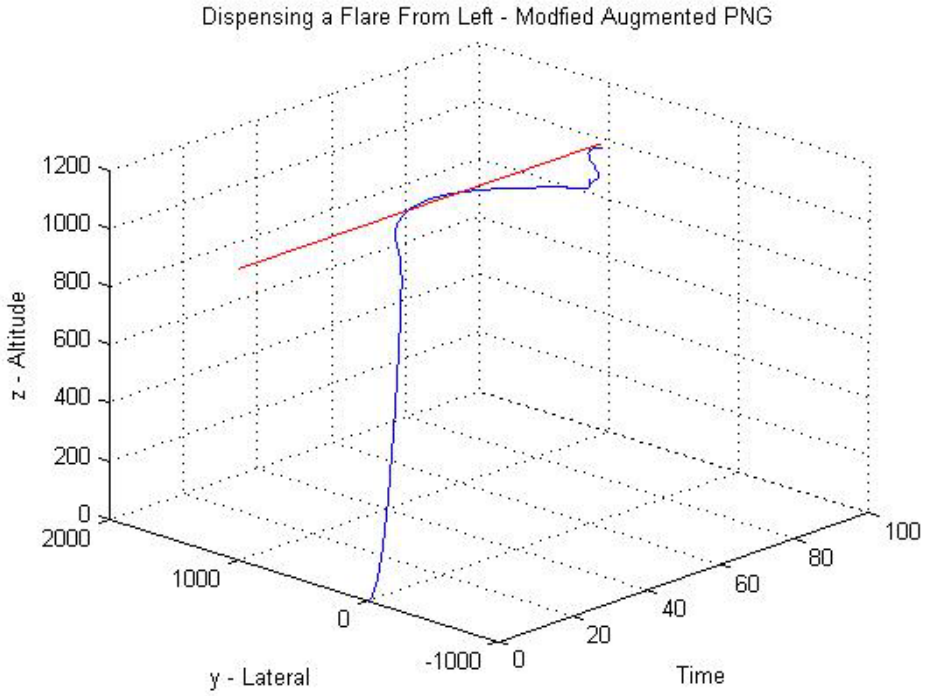


Figure 87 Dispensing a Flare From Left – Modified Augmented PNG (3D)
(target is blue and the missile is red)

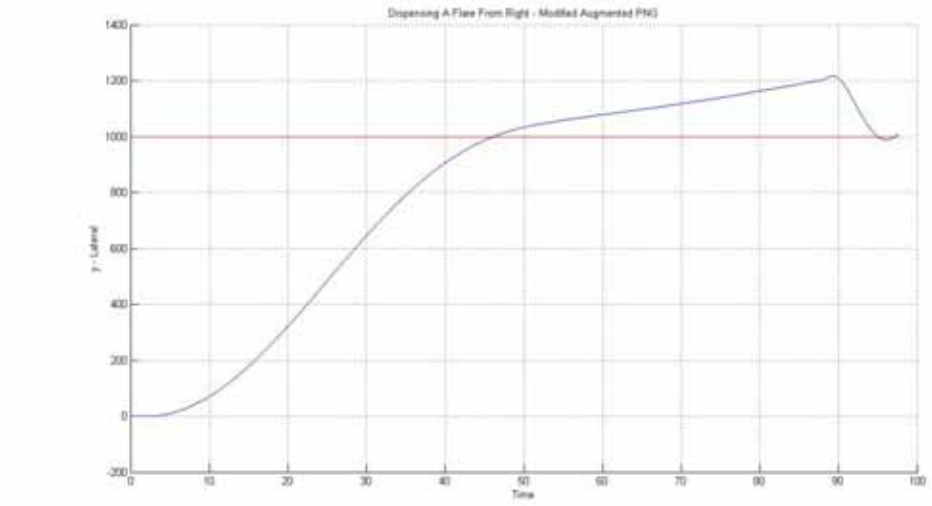


Figure 88 Dispensing a Flare From Right – Modified Augmented PNG (x – y)
(target is blue and the missile is red)

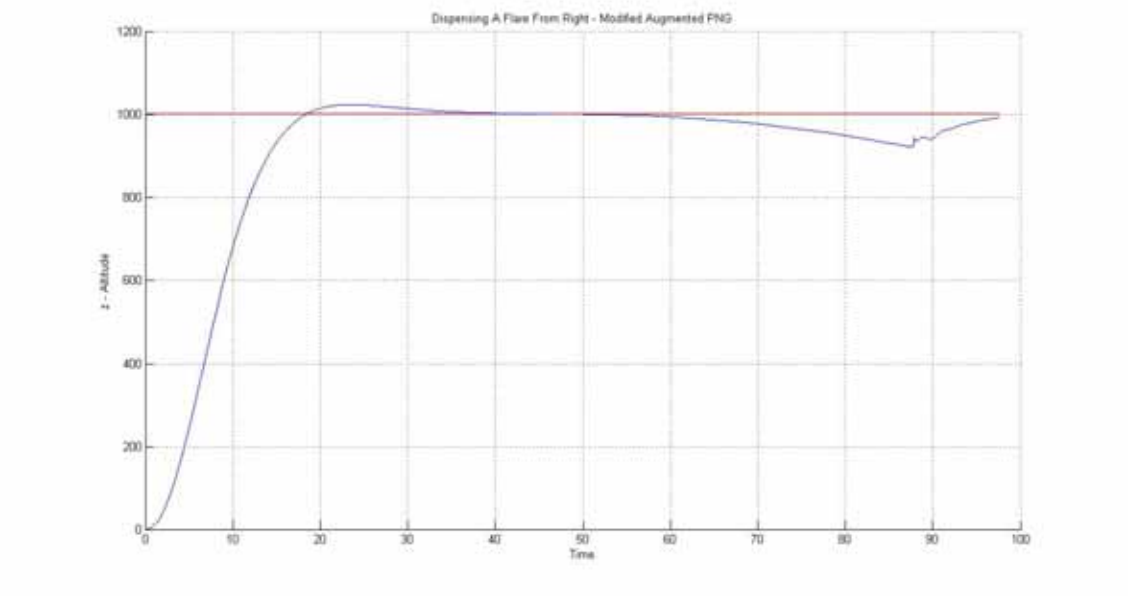


Figure 89 Dispensing a Flare From Right – Modified Augmented PNG (x – z)
(target is blue and the missile is red)

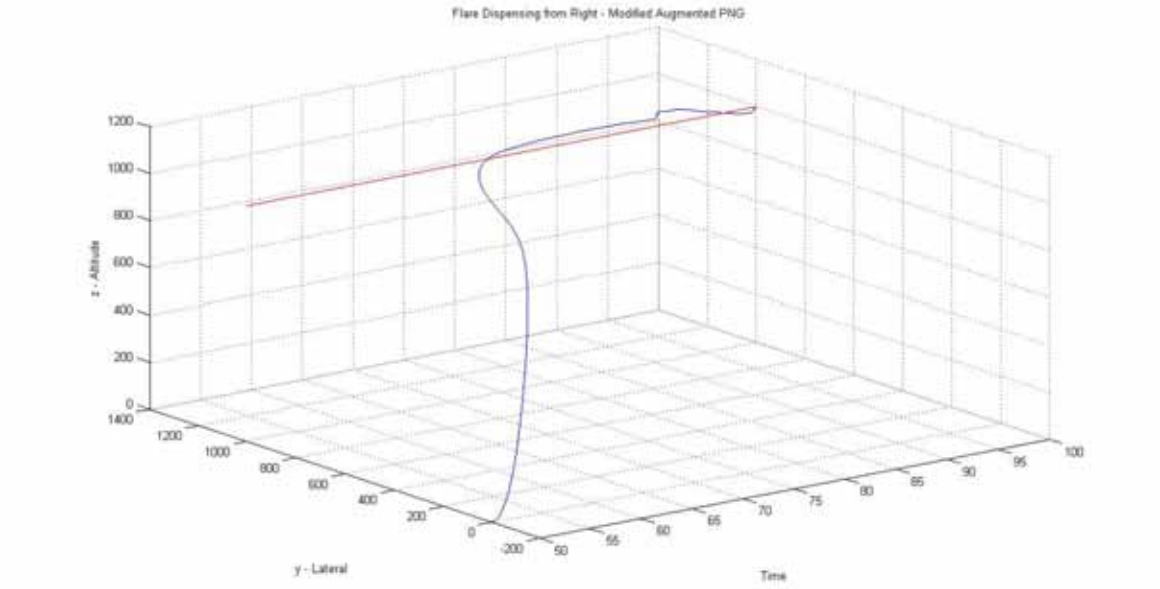


Figure 90 Dispensing a Flare From Right – Modified Augmented PNG (3D)
(target is blue and the missile is red)

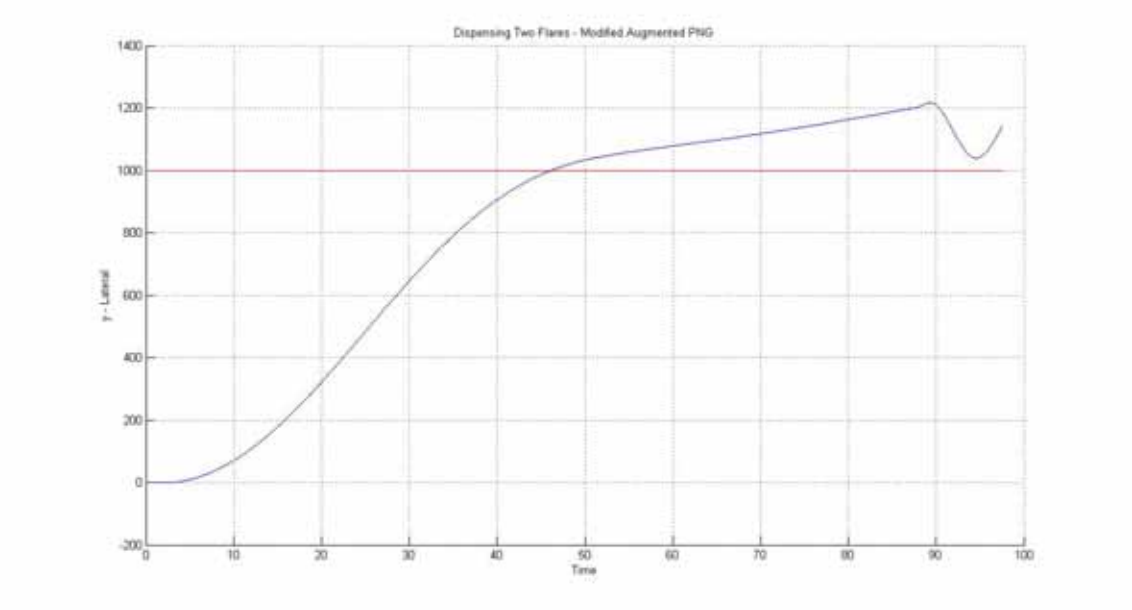


Figure 91 Dispensing Two Flares From Right – Modified Augmented PNG (x – y) (target is blue and the missile is red)

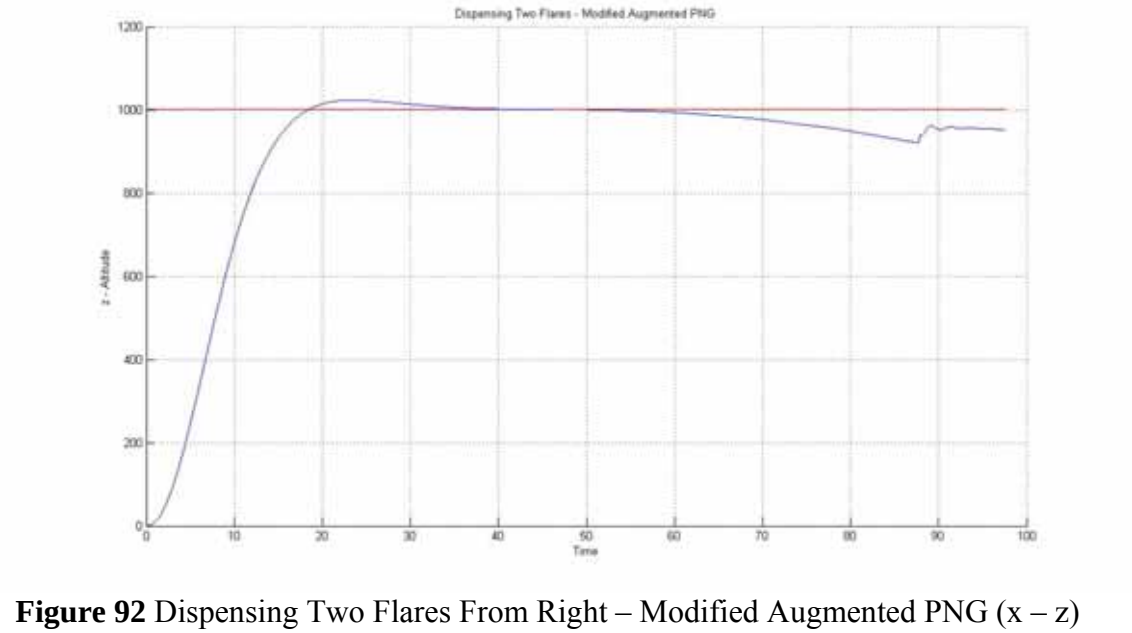


Figure 92 Dispensing Two Flares From Right – Modified Augmented PNG (x – z) (target is blue and the missile is red)

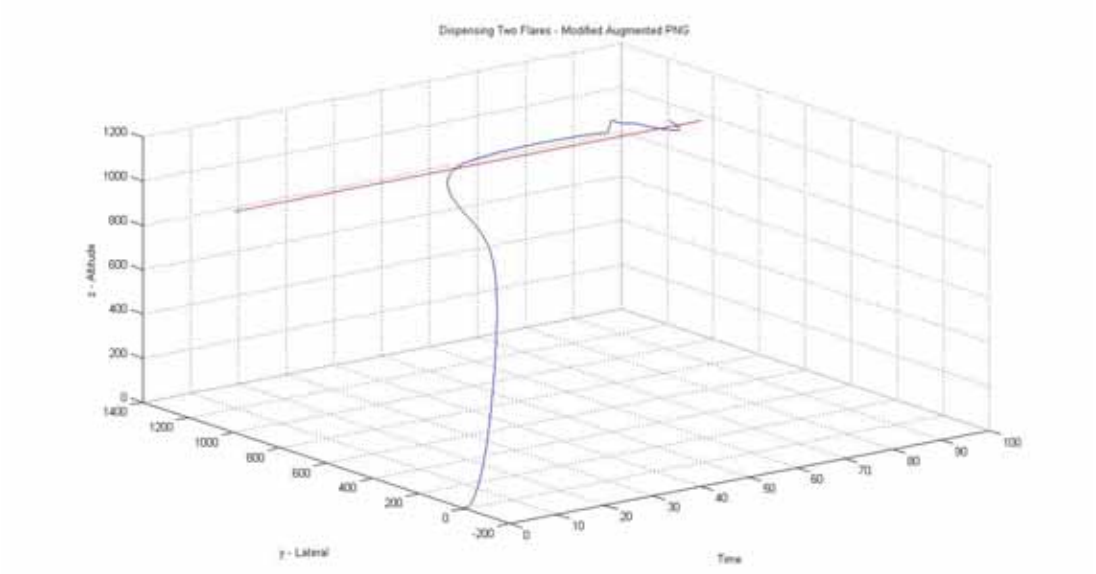


Figure 93 Dispensing Two Flares From Right – Modified Augmented PNG (3D) (target is blue and the missile is red)

3.5. Conclusions

Some of the PNG laws which are suitable for IR guided missiles have been explained in this chapter. Effectiveness of these PNG algorithms has been studied on Matlab Simulink simulations including target, missile, and IR countermeasure flares.

Performance scores of the different guidance algorithms based on different scenarios show that the gain scheduled PNG is the best guidance algorithm discussed in this thesis. The missile under gain scheduled PNG algorithm follows its target with 99% accuracy for some of the scenarios. In addition to that, modified PNG and modified augmented PNG are also effective guidance algorithms with more than 90% tracking accuracies.

The simulations have been showed that when the target dispenses flares, the missile will start to track the flares. So, tracking a target under flare dispensing scenarios is not a subject of the guidance algorithms. It is related with the seeker part of the system.

CHAPTER 4

CONCLUSIONS

In this study, seeker part and advanced proportional navigation guidance algorithms for an infrared guided missile are developed.

The work done in the thesis can be separated into two parts as autopilot and guidance designs. The missile is stabilized and get ready to be commanded by the help of the autopilots, and the target is tracked by the missile by the help of the guidance subsystem.

The mathematical model of a generic missile is given in the first place in order to understand the dynamics. The complexity of the dynamics increases the importance of the mathematical model since unreliable models may cause serious problems in field applications. Missile DATCOM software is used in order to find the aerodynamic coefficients for the forces and moments acting on a missile. Some programs have been created for automatic processing of the Missile DATCOM software outputs.

PID and LQR techniques are used to design the autopilots. Inputs for the autopilots are considered as real position of the target and the missile – target error angles separately. Then these methods are applied to the linearized model in lateral and longitudinal planes.

The seeker is modeled with a four-quadrant detector capability. Spot area, spot center, and error angle estimations are done by the seeker part.

Different kinds of proportional navigation guidance algorithms are used as the guidance method. Advantages and disadvantages of the guidance algorithms have been studied based on some scenarios including the slow and fast movements and flare dispensing targets.

This study shows that different guidance algorithms can be used for different scenarios. If suitable algorithms are combined and suitable constants are applied, the guided missile can track the target very precisely. In addition, the seeker part has to be improved with tracking algorithms in order to recognize IR-countermeasures and not to follow them.

As a future wok, seeker and guidance algorithms could be improved with tracking algorithms. Target's mathematical model could be implemented as well. Next, some studies could be done for obtaining an optimum flare dispensing technique (such as dispensing one flare, dispensing two flares, dispense flare and run away, etc.).

Contributions of the present study can be summarized as:

- Development of a four-quadrant detector, range independent seeker.
- Improvement of present PNG algorithms for a particular guided missile.
- Development of a base for optimum flare dispensing studies.
- Creating a "user manual" for the software "Missile DATCOM".

REFERENCES

[1] Laurenzo, R., "Antimissile Systems for Airliners", Aerospace America, pp.33-37, March 2005.

[2] Zarchan , P., "Tactical and Strategic Missile Guidance", Aerospace Institute of Aeronautics and Astronautics, 2002.

[3] Ateşoğlu, Ö., "Different Autopilot Designs and Their Performance Comparison for Guided Missiles", M.S. Thesis, Middle East Technical University, Ankara, Dec 1996.

[4] Karasu, Ç., "Small-Size Unmanned Model Helicopter Guidance and Control", M.S. Thesis, Middle East Technical University, Ankara, Nov 2004.

[5] Choi, W. J., "A Simultaneous Assignment Methodology of Right / Left Eigenstructures", IEEE Transactions On Aerospace And Electronic Systems Vol. 34, NO. 2, Apr 1998.

[6] Shapiro, E., Y., "Eigenstructure Assignment for Design of Multimode Flight Control Systems", IEEE Control Systems Magazine, pp.9-15, May 1985.

[7] McLean, D., "Automatic Flight Control Systems", Prentice Hall International (UK) Ltd., 1990.

[8] Akkal, E., "Control Actuation Systems And Seeker Units Of An Air-To-Surface Guided Munitions", M.S. Thesis, Middle East Technical University, Ankara, Dec 2003.

[9] www.aerospaceweb.org/question/electronics/q0191.shtml, as in 29 August 2006

[10 Ogata, K., "Modern Control Engineering", Prentice Hall Inc., 1997.

[11] Kirk, D., E., "Optimal Control Theory, An Introduction", Prentice Hall Inc., 1990.

[12] Prof. Miller, A. G., "www.wordnet.princeton.edu/perl/webwn", Cognitive Science Laboratory at Princeton University, as in 29 August 2006.

[13] Avşar, S., "Infrared Seekers, Infrared Countermeasures And Simulations", M.S. Thesis, Hacettepe University, Ankara, 2003.

[14] Kolk, W. R., "Modern Flight Dynamics", Prentice Hall Inc., 1961.

[15] www.simlabs.arc.nasa.gov/vms/six_degrees.html as in 25 April 2005

[16] "klabsorg/DEI/References/contents_detailed/aircraft_nomenclature/index.thm" as in 19 September 2004

[17] Duflos, E., Penel, P., Vanheeghe, P., "3D Guidance Law Modeling", IEEE Transactions On Aerospace And Electronic Systems Vol. 35, No. 1, January 1999.

[18] Mehrandezh, M., Sela, M., Fenton, R., Benhabib, B., "Robotic Interception of Moving Objects Using Ideal Proportional Navigation Guidance Technique", IEEE Transactions On Systems, Man, and Cybernetics, Part A Systems and Humans, Vol. 30, No. 3, May 2000.

[19] Siouris, G. M., "Missile Guidance and Control Systems", Springer., 2005

APPENDIX

A.1. MATHEMATICAL MODELING

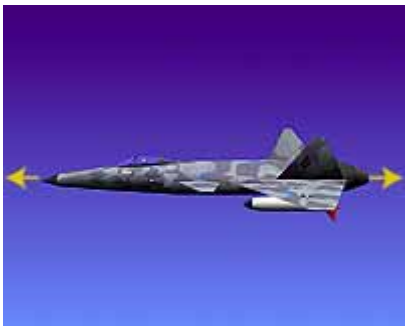
The mathematical model of a missile with six degrees of freedom can be described by twelve nonlinear first order differential equations. These equations can be analyzed in two groups: kinematics equations, which are the consequences of transformation matrix applications, that constitute a relationship between the reference axis systems through the use of Euler angles, and dynamic equations, which are derived by applying Newton's laws of motion, that relate the summation of the external forces and moments to the linear and angular accelerations of the body [7] [8]. Six degrees of freedom (6DOF) represents for longitudinal, lateral, vertical movements and attitude of the motion. They are presented in Figure 94.

A.1.1. Coordinate Systems and Transformation Matrices

Two reference frames can be used to describe the motion of a missile: earth fixed coordinate frame and missile body coordinate frame. These reference frames are shown in Figure 95.

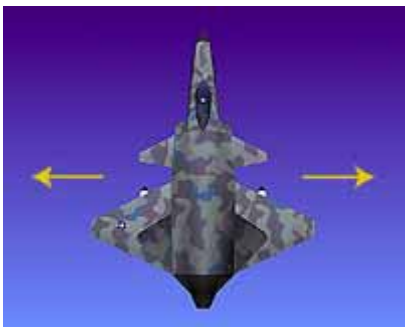
A.1.1.1. Earth Fixed Coordinate System

The Earth fixed coordinate system is assumed to be inertial because the range of the missile is short compared to the radius of the earth, and the motion of the missile is much faster compared to the earth motion [1]. Its axes are X, Y, and Z. The X-axis points towards north, the Z-axis points downwards to Earth's center, and the Y-axis is the complementing orthogonal axis found by the right hand rule.

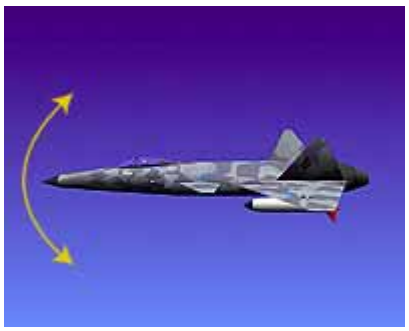


Longitudinal (forward and backward thrust)

Vertical (aircraft moves upward and downward)



Lateral (aircraft moves from side to side)



Pitch (nose pitches up or down)

Roll (wings roll up or down)

Yaw (nose moves from side to side)

Figure 94 Six Degrees of Freedom [15]

A.1.1.2. Missile Body Coordinate System

The origin is located at the missile centre of gravity, and the axes are fixed relative to the missile body. Its axes are x, y, and z; or X_b, Y_b, and Z_b. The x-axis is along the center line of the missile body and positive forward. The y-axis is perpendicular to the x-axis and is along the lateral side of the missile; it is positive to the right. The z-axis is mutually perpendicular to the x and y axes and contained in the plane of the centered yaw motivators. If y is horizontal, then z is positive downwards.

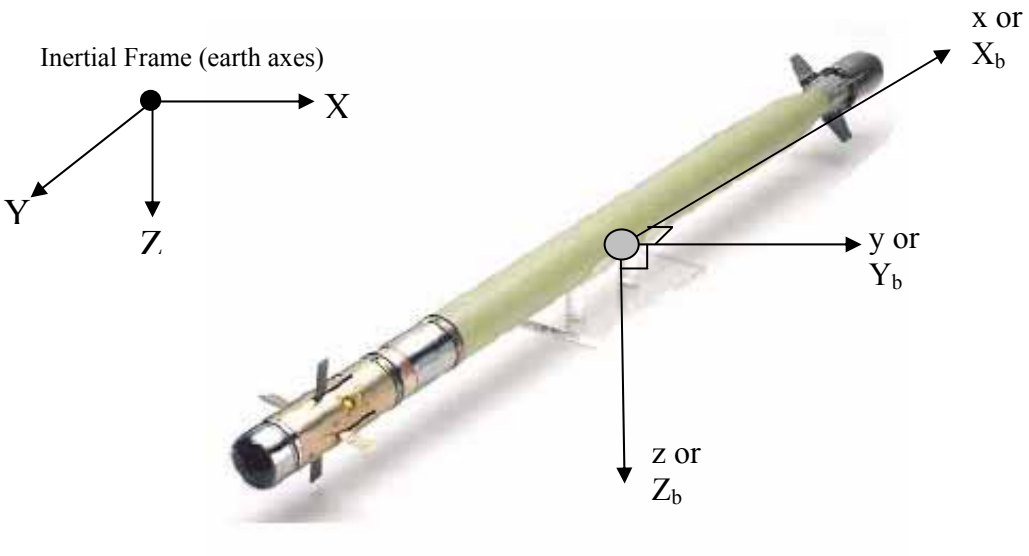


Figure 95 Missile Body and Earth Axes

A.1.1.3. Rotation Matrices

There is a relationship between the coordinate axes that makes a connection between vectorial quantities specified in different coordinate systems. This is achieved by axis transformations. Any set of axes can be obtained from any other set by a sequence of three rotations. For each rotation a transformation matrix is applied to the vectorial quantities. The final transformation matrix is simply the product of the three matrices, multiplied in the order of the rotations. [8]

It is important to note that the sequences of rotations affect the final position [14]. This condition can be seen clearly in Figure 96.

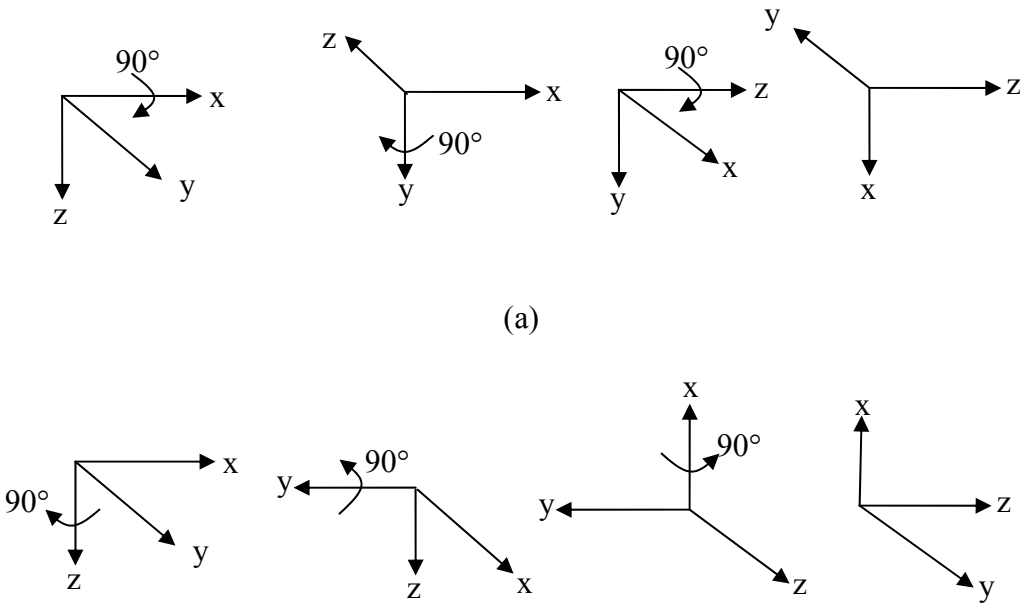


Figure 96 (a) Rotation in x, y, z Order (b) Rotation in z, y, x Order

Rotation of the systems own axes is arbitrary, but the common law of the aeronautics states that first rotate the system by ψ (yaw angle) about Z axes, then by θ (pitch angle) about new Y axes, and by ϕ (roll angle) about the latest X axes which are called **Euler Angles** [14].

Transformation matrices are used for converting one type of reference system to another. They are usually the direction cosines matrices defined in terms of the Euler rotation angles. The angular orientation of the body is given by the Euler angles. Following is the corresponding direction cosine matrix, which is the combination of the rotation matrices:

$$T_{321}(\Psi,\theta,\Phi) = R_1 R_2 R_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (59)$$

T_{321} representation stands for the transformation from the Earth axis to the body axis system.

A.1.2. Equations of a Missile

A.1.2.1. Dynamic Equations

A.1.2.1.1. Force Equations (Translational Dynamic Equations)

The time derivative of a body's linear momentum is equal to the external forces applied to it [14] as shown by the following equation:

$$\sum F = \dot{L} \quad (60)$$

where **F** is force and **L** is linear momentum.

Some assumptions have to be made before starting the derivation:

- Mass of the missile remains constant.
- Missile is a rigid body.

Linear momentum is product of mass and velocity. If an infinitesimal element of mass is chosen in a body, and a point p fixed is fixed in space specified at random which creates a vector **S** connecting the elemental mass and the fixed point p; the linear momentum **L** becomes,

$$dL = \dot{S}\, dm_i \quad (61)$$

By spreading this supposition to the whole body, the linear momentum can be formulated as follows,

$$L = m(\dot{R} - W) \tag{62}$$

where m is mass, **W** is the velocity of the center of gravity relative to the body, and $\dot{R}$ is the absolute center of gravity.

If W is small enough to be neglected, the force equation can be reduced to,

$$F = \dot{L} = m\ddot{R} \tag{63}$$

The linear velocity has scalar components u, v, and w along the x, y, and z axes; and the angular velocity has scalar components p, q, and r, about x, y, and z. Denoting the unit vectors by i,j, and k linear velocity can be stated as:

$$\dot{R} = ui + vj + wk \tag{64}$$

Then, the acceleration of the center of gravity is,

$$\ddot{R} = \overset{x}{\dot{R}} + \omega * \dot{R} \tag{65}$$

where $\overset{x}{\dot{R}} = \dot{u}\,i + \dot{v}\,j + \dot{w}k$ is the scalar derivative of velocity. Then,

$$\omega * \dot{R} = \begin{vmatrix} i & j & k \\ p & q & r \\ u & v & w \end{vmatrix} = i(qw - rv) + j(ru - pw) + k(pv - qu) \tag{66}$$

Denoting the force components by F_X, F_Y, and F_Z we have,

$$F_X = m(\dot{u} + qw - rv)$$

$$F_Y = m(\dot{v} + ru - pw) \tag{67}$$

$$F_Z = m(\dot{w} + pv - qu)$$

A.1.2.1.2. Moment Equations (Rotational Dynamic Equations)

The time derivative of a body's angular momentum is equal to the external moment applied to it [14].

Angular momentum can be stated as follows:

$$H = R * (\dot{R} - W)m + \oint r_i * w * r_i dm_i) \tag{68}$$

The first term of the angular momentum is the vector product of the linear momentum and the position vector of center of gravity. The second term can be written as,

$$\begin{gathered} r_i = xi + yj + zk \\ \omega = pi + qj + rk \\ \omega * r_i = \begin{vmatrix} i & j & k \\ p & q & r \\ x & y & z \end{vmatrix} = i(qz - ry) + j(rx - pz) + k(py - qx) \\ r_i * \omega * r_i = \begin{vmatrix} i & j & k \\ x & y & z \\ qz - ry & rx - pz & py - qx \end{vmatrix} = \begin{matrix} i\{p(y^2 + z^2) - qxy - rzx\} \\ + j\{p(z^2 + x^2) - ryz - pxy\} \\ + k\{p(x^2 + y^2) - pzx - qyz\} \end{matrix} \end{gathered} \tag{69}$$

$$\oint r_i * \omega * r_i dm_i = \begin{matrix} i\{p\oint(y^2 + z^2)dm_i - q\oint xydm_i - r\oint zxdm_i\} \\ j\{q\oint(z^2 + x^2)dm_i - r\oint yzdm_i - p\oint xydm_i\} \\ k\{r\oint(x^2 + y^2)dm_i - p\oint zxdm_i - q\oint yzdm_i\} \end{matrix} \tag{70}$$

The last equation can be written in compact form as follows:

$$\oint r_i * \omega * r_i dm_i = \begin{matrix} i\{pI_{xx} - qI_{xy} - rI_{xz}\} \\ j\{qI_{yy} - rI_{yz} - pI_{yx}\} \\ k\{rI_{zz} - pI_{zx} - qI_{zy}\} \end{matrix} = \Theta\omega \tag{71}$$

And finally, the angular momentum of the body is,

$$H = R * (\dot{R} - W)m + \Theta\omega \tag{72}$$

If the position vector R = 0 is chosen for all circumstances, the angular momentum becomes the matrix product of the inertia tensor and the angular velocity vector.

As we mentioned before, the time derivative of a body's angular momentum is equal to external moment applied to it.

$$M = \dot{H} = \frac{d}{dt}(\Theta\omega) \tag{73}$$

The external moment has L, M, and N components about the x, y, and z axes; and the angular momentum **H** has scalar components H_x, H_y, H_z where

$$\begin{aligned} H_x &= pI_{xx} - qI_{xy} - rI_{xz} \\ H_y &= qI_{yy} - rI_{yz} - pI_{yx} \\ H_z &= rI_{zz} - pI_{zx} - qI_{zy} \end{aligned} \tag{74}$$

Differentiating the angular momentum:

$$\dot{\boldsymbol{H}} = \overset{x}{\boldsymbol{H}} + \boldsymbol{\omega} * \boldsymbol{H} \tag{75}$$

$$= \dot{H}_x\, i + \dot{H}_y\, j + \dot{H}_z\, k + \begin{vmatrix} i & j & k \\ p & q & r \\ H_x & H_y & H_z \end{vmatrix} \tag{76}$$

$$\begin{aligned} & i\{\dot{p}\, I_{xx} + (rp - \dot{q})I_{xy} - (\dot{r} + pq)I_{zx} + (r^2 - q^2)I_{yz} + qr(I_{zz} - I_{yy})\} \\ = \; & j\{\dot{q}\, I_{yy} + (pq - \dot{r})I_{yz} - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + rp(I_{xx} - I_{zz})\} \\ & k\{\dot{r}\, I_{zz} + (qr - \dot{p})I_{zx} - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + pq(I_{yy} - I_{xx})\} \end{aligned} \tag{77}$$

The components of the external moment can be found as follows:

$$\begin{aligned} M_L &= \dot{p}\, I_{xx} + (rp - \dot{q})I_{xy} - (\dot{r} + pq)I_{zx} + (r^2 - q^2)I_{yz} + qr(I_{zz} - I_{yy}) \\ M_M &= \dot{q}\, I_{yy} + (pq - \dot{r})I_{yz} - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + rp(I_{xx} - I_{zz}) \\ M_N &= \dot{r}\, I_{zz} + (qr - \dot{p})I_{zx} - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + pq(I_{yy} - I_{xx}) \end{aligned} \tag{78}$$

In general, I_{xy} and I_{yz} either vanish identically or are small enough to be neglected because there is a mass symmetry about the x-z plane [14]. The moment equations can be reduced to,

$$\begin{aligned} M_L &= \dot{p}\, I_{xx} - (\dot{r} + pq)I_{zx} + qr(I_{zz} - I_{yy}) \\ M_M &= \dot{q}\, I_{yy} + (p^2 - r^2)I_{zx} + rp(I_{xx} - I_{zz}) \\ M_N &= \dot{r}\, I_{zz} + (qr - \dot{p})I_{zx} + pq(I_{yy} - I_{xx}) \end{aligned} \tag{79}$$

Also, due to the rotational symmetry of the missile, I_{yy} and I_{zz} terms are taken to be equal to each other. Then,

$$\begin{aligned} M_L &= \dot{p}\, I_{xx} \\ M_M &= \dot{q}\, I_{yy} + rp(I_{xx} - I_{zz}) \\ M_N &= \dot{r}\, I_{zz} + pq(I_{yy} - I_{xx}) \end{aligned} \tag{80}$$

A.1.2.2. Kinematics Equations

A.1.2.2.1. Translational Kinematics Equations

Velocity expressed in body axis system can be transformed to earth axis system by the following expression:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = T^{T}_{321} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{81}$$

where T_{321} is the direction cosine matrix shown in Eq. 59; X,Y,Z are the position of the body with respect to the Earth fixed reference frame; and u,v, and w are the velocities with respect to body frame.

$$\begin{aligned} \dot{x} &= uc\theta c\psi + v(s\phi s\theta c\psi - c\phi s\psi) + w(s\phi s\theta c\psi + s\phi s\psi) \\ \dot{y} &= uc\theta s\psi + v(s\phi s\theta s\psi + c\phi c\psi) + w(c\phi s\theta s\psi - s\phi c\psi) \\ \dot{z} &= -us\theta + vs\phi c\theta + wc\phi c\theta \end{aligned} \tag{82}$$

A.1.2.2.2. Rotational Kinematics Equations

The Euler rates with respect to the body rotational rates are given by the following expressions [14]:

$$\begin{aligned} \dot{\Phi} &= p + \tan\theta\{q\sin\Phi + r\cos\Phi\} \\ \dot{\theta} &= q\cos\Phi - r\sin\Phi \\ \dot{\Psi} &= \frac{r\cos\Phi + q\sin\Phi}{\cos\theta} \end{aligned} \tag{83}$$

A.1.3. Forces and Moments Acting on a Missile

The forces and moments acting on a missile are given as F_X, F_Y, F_Z, and M_L, M_M, and M_N. Actually, these forces and moments are total of some distinct forces and moments. Aerodynamic, thrust, and the gravitational effects have to be taken into consideration while talking about these forces and moments. In fact, these forces and moments can be rewritten as:

$$\begin{aligned} F_X &= F_{ax} + mg_x + T_x \\ F_Y &= F_{ay} + mg_y + T_y \\ F_Z &= F_{az} + mg_z + T_z \end{aligned} \tag{84}$$

$$\begin{aligned} M_L &= L + L_T \\ M_M &= M + M_T \\ M_N &= N + N_T \end{aligned} \tag{85}$$

In the above equations, F_{ax}, F_{ay}, F_{az}, L, M, and N are the forces and moments which only occur due to aerodynamic effects on the missile flight. g_x, g_y, and g_z are the body frame components of the gravitational acceleration. T_x, T_y, T_z, L_T, M_T, N_T are the thrust force and thrust moment components.

A.1.3.1. Introduction to Motion Parameters

Aerodynamic forces and moments can be expressed by the help of some motion parameters. These are explained in detail below:

Angle of Attack: The angle between the x-axis and the projection of the missile velocity vector on the reference plane (Figure 97). It is positive when the missile velocity component along the y-axis is positive. It is shown by α.

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{86}$$

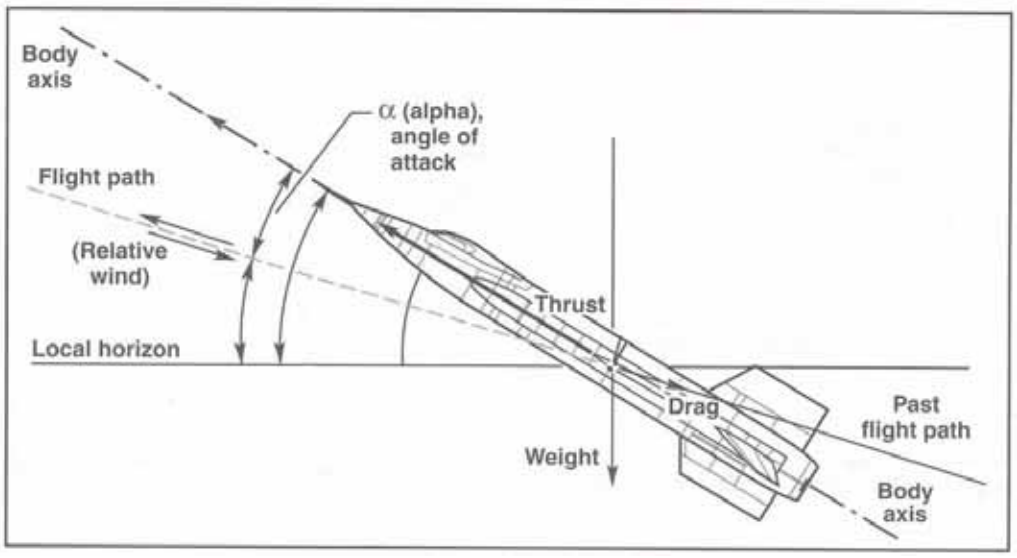


Figure 97 Angle of Attack Description [16]

Angle of Sideslip: The angle which the missile velocity vector makes with the reference plane of the missile (Figure 98). It is shown by β.

$$\beta = \tan^{-1}\left(\frac{v}{u}\right) \tag{87}$$

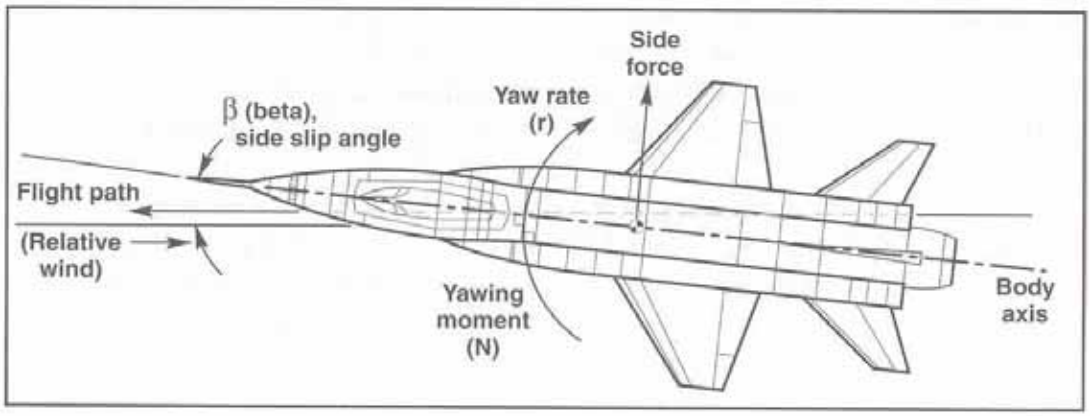


Figure 98 Angle of Sideslip Description [16]

Mach Number: The ratio of the airspeed to the speed of sound. If C is the speed of the sound, then Mach number can be expressed as:

$$M = \frac{V_T}{C} \tag{88}$$

where $V_T = \sqrt{u^2 + v^2 + w^2}$. C can be stated as $C = \sqrt{\gamma RT}$ where R is the universal air gas constant, T is the ambient temperature which changes with altitude, and γ is the specific heat ratio of the air.

Air Density: Mass of air per unit volume. This parameter can be expressed as:

$$\bar{q} = \frac{1}{2}\rho V_T^{\,2} \tag{89}$$

The term ρ is the air density and calculated as follows:

$$\rho = \begin{cases} \rho_0(1 - 0.00002256h)^{4.256}, & h \le 10000m \\ 0.412x10^{(-0.000151(h-10000))}, & h > 10000m \end{cases} \quad (90)$$

where ρ_o is the air density at sea level (1.223 kg/m^3), h is the altitude and V is the missile velocity.

A.1.3.2. Force and Moment Coefficients

Aerodynamic forces and moments can be represented with their coefficients. If reference area is s, reference length is l, dynamic pressure is $\bar{q}$, and C_X, C_Y, and C_Z are coefficients for forces respectively; then the forces can be stated as follows:

$$F_{ax} = \bar{q}\, sC_X \quad (91)$$

$$F_{ay} = \bar{q}\, sC_Y \quad (92)$$

$$F_{az} = \bar{q}\, sC_Z \quad (93)$$

If C_L, C_M, and C_N are moment coefficients, moments can be stated as:

$$L = \bar{q}\, slC_L \quad (94)$$

$$M = \bar{q}\, slC_M \quad (95)$$

$$N = \bar{q}\, slC_N \quad (96)$$

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.

Force and moment coefficients can be found from lookup tables that are created from Mach number, angle of attack, angle of sideslip, pitch fin deflection, roll fin deflection, and yaw fin deflection. Force and moment coefficients' table are going to be created by the help of the software Missile DATCOM. Functional forms of the coefficients, which are out of the scope of this discussion, can be found in other books.

The derivation of non-dimensional form of aerodynamic forces and moment coefficients can be found as follows [14]:

$$C_X = C_d \tag{97}$$

$$C_y = C_{y_\beta}\beta + C_{y_r} r\frac{d}{2V} + C_{y\delta_r}\delta_r + C_{y_p} p\frac{d}{2V} + C_{y\delta_a}\delta_a \tag{98}$$

$$C_z = C_{z_\alpha}\alpha + C_{y_{qr}} q\frac{d}{2V} + C_{y\delta_e}\delta_e \tag{99}$$

$$C_l = C_{l_p} p\frac{d}{2V} + C_{l_\beta}\beta + C_{l\delta_a}\delta_a + C_{l_r} r\frac{d}{2V} + C_{l\delta_r}\delta_r \tag{100}$$

$$C_m = C_{m_\alpha}\alpha + C_{m_{qr}} q\frac{d}{2V} + C_{m\delta_e}\delta_e \tag{101}$$

$$C_n = C_{n_\beta}\beta + C_{n_r} r\frac{d}{2V} + C_{n\delta_r}\delta_r + C_{n_p} p\frac{d}{2V} + C_{n\delta_a}\delta_a \tag{102}$$

A.1.4. Summary of Missile Equations

Translational Dynamic Equations:

$$\sum X/m = (\dot{u} + qw - rv) - d_{31}g$$

$$\sum Y/m = (\dot{v} + ru - pw) - d_{32}g \tag{103}$$

$$\sum Z/m = (\dot{w} + pv - qu) - d_{33}g$$

$$\begin{aligned} \dot{u} &= -qw + rv + \sum X/m - g\sin\theta \\ \Rightarrow\Rightarrow\ \dot{v} &= pw - ru + \sum Y/m + g\sin\phi\cos\theta \\ \dot{w} &= qu - pv + \sum Z/m + g\cos\phi\cos\theta \end{aligned} \tag{104}$$

$$\begin{aligned} \dot{u} &= -qw + rv + \frac{\bar{q}s}{m}C_X + \frac{T_X}{m} - g\sin\theta \\ \dot{v} &= -ru + pw + \frac{\bar{q}s}{m}C_Y + \frac{T_Y}{m} + g\sin\phi\cos\theta \\ \dot{w} &= -pv + qu + \frac{\bar{q}s}{m}C_Z + \frac{T_Z}{m} + g\cos\phi\cos\theta \end{aligned} \tag{105}$$

Rotational Dynamic Equations:

$$\begin{aligned} L &= \dot{p}I_{xx} + (rp - \dot{q})I_{xy} - (\dot{r} + pq)I_{zx} + (r^2 - q^2)I_{yz} + qr(I_{zz} - I_{yy}) \\ M &= \dot{q}I_{yy} + (pq - \dot{r})I_{yz} - (\dot{p} + qr)I_{xy} + (p^2 - r^2)I_{zx} + rp(I_{xx} - I_{zz}) \\ N &= \dot{r}I_{zz} + (qr - \dot{p})I_{zx} - (\dot{q} + rp)I_{yz} + (q^2 - p^2)I_{xy} + pq(I_{yy} - I_{xx}) \end{aligned} \tag{106}$$

$$\begin{aligned} \dot{p}I_{xx} &= L - (rp - \dot{q})I_{xy} + (\dot{r} + pq)I_{zx} - (r^2 - q^2)I_{yz} - qr(I_{zz} - I_{yy}) \\ \Rightarrow\Rightarrow\ \dot{q}I_{yy} &= M - (pq - \dot{r})I_{yz} + (\dot{p} + qr)I_{xy} - (p^2 - r^2)I_{zx} - rp(I_{xx} - I_{zz}) \\ \dot{r}I_{zz} &= N - (qr - \dot{p})I_{zx} + (\dot{q} + rp)I_{yz} - (q^2 - p^2)I_{xy} - pq(I_{yy} - I_{xx}) \end{aligned} \tag{107}$$

Because of the symmetry about x and z axes, I_{xy} and I_{yz} are very small as to be insignificant. Then,

$$\begin{aligned} \dot{p} &= (\dot{r} + pq)\frac{I_{zx}}{I_{xx}} - qr\left(\frac{I_{zz} - I_{yy}}{I_{xx}}\right) + \sum L/I_{xx} \\ \Rightarrow\Rightarrow\ \dot{q} &= -(p^2 - r^2)\frac{I_{zx}}{I_{yy}} - rp\left(\frac{I_{xx} - I_{zz}}{I_{yy}}\right) + \sum M/I_{yy} \\ \dot{r} &= -(qr - \dot{p})\frac{I_{zx}}{I_{zz}} - pq\left(\frac{I_{yy} - I_{xx}}{I_{zz}}\right) + \sum N/I_{zz} \end{aligned} \tag{108}$$

Briefly, six nonlinear differential equations for missiles can be written as:

$$\dot{p} = (\dot{r} + pq)\frac{I_{zx}}{I_{xx}} - qr(\frac{I_{zz} - I_{yy}}{I_{xx}}) + \frac{C_L \bar{q} sl}{I_{xx}}$$

$$\dot{q} = -(p^2 - r^2)\frac{I_{zx}}{I_{yy}} - rp(\frac{I_{xx} - I_{zz}}{I_{yy}}) + \frac{C_M \bar{q} sl}{I_{yy}} \qquad (109)$$

$$\dot{r} = -(qr - \dot{p})\frac{I_{zx}}{I_{zz}} - pq(\frac{I_{yy} - I_{xx}}{I_{zz}}) + \frac{C_N \bar{q} sl}{I_{zz}}$$

Translational Kinematical Equations of Motion:

$$\dot{x} = uc\theta c\psi + v(s\phi s\theta c\psi - c\phi s\psi) + w(s\phi s\theta c\psi + s\phi s\psi)$$

$$\dot{y} = uc\theta s\psi + v(s\phi s\theta s\psi + c\phi c\psi) + w(c\phi s\theta s\psi - s\phi c\psi)$$

$$\dot{z} = -us\theta + vs\phi c\theta + wc\phi c\theta \qquad (110)$$

Rotational Kinematical Equations of Motion:

$$\dot{\Phi} = p + \tan\theta\{q\sin\Phi + r\cos\Phi\}$$

$$\dot{\theta} = q\cos\Phi - r\sin\Phi \qquad (111)$$

$$\dot{\Psi} = \frac{r\cos\Phi + q\sin\Phi}{\cos\theta}$$

A.1.5. Linearization of the Equations

The practical controller design in aerospace studies involves linearization of the 12 nonlinear equations of motion. Linearization is based on certain assumptions which are stated below:

For each point of linearization, a constant ambient temperature (T), ambient density (ρ) and velocity (Mach number) are assumed.

Angle of attack (α), angle of sideslip (β), fin deflections (δ_a, δ_e, δ_r) are assumed to be small ($\alpha < 15^o$, $\beta < 15^o$, $\delta_{a,e,r} < 15^o$).

Rolling motion is constant and very small ($\varphi \rightarrow 0^o$, $p \rightarrow 0^o$/sec).

Components of the gravitational accelerations are assumed to be external disturbances (neglected).

Since the angle of attack and sideslip are small, they are approximated as:

$$\alpha \approx \frac{w}{u}$$
$$\beta \approx \frac{v}{V} \tag{112}$$

Taking into account the equations above and remembering $V = \sqrt{u^2 + v^2 + w^2}$, we can say that $V \cong u$ and so $\dot{u} \cong 0$. Now, we can reduce the 12 nonlinear state equations (Eq. 105, Eq. 109, Eq. 110, Eq. 111) taking into account the assumptions made above.

A.1.5.1.1. Linearized Translational Dynamic Equations of Motion

In Eq. 2.47, $\dot{u} \cong 0$, Φ ≈ 0, p ≈ 0 and also the gravity terms are neglected and so:

$$\dot{v} = \frac{F_y}{m} - r \cdot u$$
$$\dot{w} = \frac{F_z}{m} + q \cdot u \tag{113}$$

A.1.5.1.2. Linearized Rotational Dynamic Equations of Motion

The missile is mechanically symmetric, so the lateral moments of inertias are taken to be equal i.e. Iy = Iz. Using this fact together with p ≈ 0, we have the following:

$$\begin{aligned} \dot{p} &= \frac{L}{I_x} \\ \dot{q} &= \frac{M}{I_y} \\ \dot{r} &= \frac{N}{I_z} \end{aligned} \tag{114}$$

A.1.5.1.3. Linearized Translational Kinematics Equations of Motion

Using the assumptions $\Phi \approx 0$, small θ and small Ψ, (i.e: $s\theta \approx \theta$ and $c\theta \approx 1$), we have:

$$\begin{aligned} \dot{y} &= u \cdot \psi + v \\ \dot{z} &= -u \cdot \theta + w \end{aligned} \tag{115}$$

where u = constant (0.86 Mach). Also, we assumed that $s\theta s\Psi$ term is a higher order term (HOT) and so it is neglected.

A.1.5.1.4. Linearized Rotational Kinematics Equations of Motion

Using the assumptions $\Phi \approx 0$, small θ and small Ψ and a negligible $r\tan\theta$, we have

$$\begin{aligned} \dot{\phi} &= p \\ \dot{\theta} &= q \\ \dot{\psi} &= r \end{aligned} \tag{116}$$

To continue linearization, the expansion of aerodynamic force and moment coefficients are substituted into Eq. 2.33 – Eq. 2.38 and then the Eq. 2.39 – Eq. 2.44 are utilized, so we have the following aerodynamic force and moment equations:

$$
\begin{aligned}
F_y &= (\frac{Q_d AC_{y_\beta}}{u})\cdot v + (\frac{Q_d AC_{y_r} d}{2u})\cdot r + (Q_d AC_{y_{\delta_r}})\cdot \delta_r + Q_d AC_{y_{\delta_a}}\delta_a + \frac{Q_d AC_{y_p} d}{2u} p \\
F_z &= (\frac{Q_d AC_{z_\alpha}}{u})\cdot w + (\frac{Q_d AC_{z_q} d}{2u})\cdot r + (Q_d AC_{z_{\delta_e}})\cdot \delta_e \\
L &= (\frac{Q_d Ad^2 C_{l_p}}{2u})\cdot p + (Q_d AdC_{l_{\delta_a}})\cdot \delta_a + Q_d AdC_{l_{\delta_r}}\delta_r + \frac{Q_d Ad^2 C_{l_r}}{2u} r + \frac{Q_d AC_{l_\beta}}{u}\beta \\
M &= (\frac{Q_d AdC_{m_\alpha}}{u})\cdot w + (\frac{Q_d Ad^2 C_{m_q}}{2u})\cdot q + (Q_d AdC_{m_{\delta_e}})\cdot \delta_e \\
N &= (\frac{Q_d AdC_{n_\beta}}{u})\cdot v + (\frac{Q_d Ad^2 C_{n_r}}{2u})\cdot r + (Q_d AdC_{n_{\delta_r}})\cdot \delta_r + Q_d AdC_{n_{\delta_a}}\delta_a + \frac{Q_d Ad^2 C_{n_p}}{2u} p
\end{aligned}
\tag{117}
$$

Substituting the force and moment expansions in Eq. 2.59 into Eq. 2.55 – Eq. 2.58 together with the assumption of decoupling (very small roll motion), the linear state equations can be grouped as:

c) Pitch Plane (Longitudinal) State Equations

$$
\begin{bmatrix} \dot{z} \\ \dot{w} \\ \dot{\theta} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & 1 & -u & 0 \\ 0 & \frac{Q_d AC_{z_\alpha}}{um} & 0 & \frac{Q_d AC_{z_q} d}{2um} + u \\ 0 & 0 & 0 & 1 \\ 0 & \frac{Q_d AdC_{m_\alpha}}{uI_y} & 0 & \frac{Q_d Ad^2 C_{m_q}}{2uI_y} \end{bmatrix} \begin{bmatrix} z \\ w \\ \theta \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{Q_d AC_{z_{\delta_e}}}{m} \\ 0 \\ \frac{Q_d AdC_{m_{\delta_e}}}{I_y} \end{bmatrix} \delta_e \tag{118}
$$

d) Roll-Yaw Plane (Lateral) State Equations

$$
\begin{bmatrix} \dot{y} \\ \dot{v} \\ \dot{\psi} \\ \dot{r} \\ \dot{p} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & u & 0 & 0 & 0 \\ 0 & \frac{Q_d AC_{y_\beta}}{um} & 0 & \frac{Q_d AC_{y_r} d}{2um} - u & \frac{Q_d AC_{y_p} d}{2um} & g\cos(\theta_0) \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{Q_d AdC_{n_\beta}}{uI_z} & 0 & \frac{Q_d Ad^2 C_{n_r}}{2uI_z} & \frac{Q_d Ad^2 C_{n_p}}{2uI_z} & 0 \\ 0 & \frac{Q_d AdC_{l_\beta}}{uI_x} & 0 & \frac{Q_d Ad^2 C_{l_r}}{2uI_x} & \frac{Q_d Ad^2 C_{l_p}}{2uI_x} & 0 \\ 0 & 0 & 0 & \tan(\theta_0) & 1 & 0 \end{bmatrix} \begin{bmatrix} y \\ v \\ \psi \\ r \\ p \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{Q_d AC_{y_{\delta_r}}}{m} & \frac{Q_d AC_{y_{\delta_a}}}{m} \\ 0 & 0 \\ \frac{Q_d AdC_{n_{\delta_r}}}{I_z} & \frac{Q_d AdC_{n_{\delta_a}}}{I_z} \\ \frac{Q_d AdC_{l_{\delta_r}}}{I_x} & \frac{Q_d AdC_{l_{\delta_a}}}{I_x} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_r \\ \delta_a \end{bmatrix} \tag{119}
$$

since $\dot{x} = cons\tan t$ and $\dot{u} = 0$.

A.2. MISSILE DATCOM USER MANUAL

Missile DATCOM is useful software that helps to create force and moment coefficients' lookup tables. It uses missile's physical properties and Mach number, angle of attack, angle of sideslip for filling the table. 20 Mach numbers and 20 angle of attack values can be used in a run with the software.

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.

A.2.1. Input File (for005.dat)

A.2.1.1. Control Card Inputs

Control cards are one line commands which select program options. Although they are not required inputs, they permit user control over program execution and the types of output desired. Control cards enable the following:

Printing internal data array results for diagnostic purposes (DUMP)

Outputting intermediate calculations (PART, BUILD, PRESSURES, PRINT AERO, PRINT EXTRAP, PRINT GEOM, PLOT, NAMELIST, WRITE, FORMAT)

Selecting the system of units to be used (DIM, DERIV)

Defining multiple cases, permitting the reuse of previously input namelist data or deleting namelists of a prior case (SAVE, DELETE, NEXT CASE)

Adding case titles or comments to the input file and output pages (*, CASEID)

Limits the calculations to longitudinal aerodynamics (NO LAT)

Missile DATCOM input file starts with the control card inputs which determine the output formats. Many different control cards are available. There is no limit to the number of control cards that can be present in a case. If two or more control cards

contradict each other, the last control card input will take precedence. All control cards must be input as shown, including any blanks. Control cards can start in any column but they cannot be continued to a second card. Misspelled cards are ignored. Control cards can be located anywhere within a case.

Some of the control card inputs can be summarized as in Table 9:

Table 9 Control Card Inputs

Name	**Explanations**
CASEID MISSILE	*A user supplied title to be printed on each output page specified. (up yo 73 characters)*
DIM M	*This option is used for determining the units for the user inputs and outputs. "DIM M" means meters. "DIM IN (inches)", "DIM FT (feet)" and "DIM CM(centimeters)" are also possible.*
SOSE	*Second-order shock expansion method will be used. This option has to used for the Mach numbers greater than 2.0.*
DERIV DEG	*All output derivatives are set to either degree or radian (DERIV RAD) measure.*
DAMP	*Dynamic derivatives are computed and the results output for the configuration.*
PART	*Permits printing of partial aerodynamic output.*
BUILD	*This control card instructs the program to print the results of a configuration build-up.*
DUMP CASE	*Internal data blocks, used in the computation of the case, are written on tape unit 6 ("for006.dat"). This control card automatically selects partial output (PART). This control card is effective only for the case in which it appears.*
HYPER	*This control card causes the program to select the Newtonian flow method for bodies at any Mach number above 1.4. HYPER should normally be selected at Mach numbers greater than 6.*
NAMELIST	*This control card instructs the program to print all namelist data. This is useful when multiple inputs of the same variable or namelist are used. This control card is effective only for the case in which it appears.*

Table 9 (continued)

NEXT CASE	*This card indicates termination of the case input data and instructs the program to begin case execution. It is required for multiple case "runs". This card must be the last card input for the case.*
NOGO	*This control card permits the program to cycle through all of the input cases without computing configuration aerodynamics. It can be present anywhere in the input stream and only needs to appear once.* *This option is useful for performing error checking to insure all cases have been correctly set up.*
NO LAT	*This control card inhibits the calculation of the lateral-directional derivatives due to sideslip angle, and the roll rate and yaw rate derivatives if the control card DAMP is selected. Large savings in computation time can be realized by using this option. This option is automatically selected when using TRIM.*
PLOT	*A data file for use with a post-processing plotting program is provided when this control card is used. A formatted file is written to tape unit 3 ("for003.dat").*
PRESSURES	*This control card instructs the program to print the body and fin alone pressure coefficient distributions at supersonic speeds. Only pressure data to 15 degrees angle of attack for bodies and at zero angle of attack for fins are printed. The body pressure output at positive angle of attack is written to tape unit 10 ("for010.dat"). The fin pressure output is written to tape unit 11 ("for011.dat"). The body pressure output and local Mach number at zero angle of attack are written to tape unit 12 ("for012.dat"). This control card is effective only for the case in which it appears.*
SPIN	*When the SPIN control card is input, spin and magnus derivatives are computed for body alone. If the configuration being run is a body + fin sets, the spin derivatives are still computed for body alone. A PART or BUILD card must be input for body alone derivatives to be printed out. This control card is effective only for the case in which it appears.*
TRIM	*This control card causes the program to perform a trim calculation. Component buildup data cannot be dumped if TRIM is selected. The use of this control card is the same as if the namelist TRIM was included except that the defaults for namelist TRIM are used. This control card is effective only for the case in which it appears.*

A.2.1.2. Namelist Inputs

After the control card inputs, "namelist inputs" takes place. "Namelist input parts" start with "$" and the name of the namelist, and end with "$". All properties of a missile are determined by namelists.

Namelist inputs are column independent, and can begin in any column including the first. If a namelist is continued to a second card, the continued card must leave column 1 blank. Also, the card before the continued card must end with a comma. The last usable column is number 79 if column 1 is used and column 80 if column 1 is blank.

The same namelist can be input multiple times for the same input case. The total number of namelists read, including repeat occurrences of the same namelist name, must not exceed 300.

The three namelist inputs

```
$REFQ SREF=1.,$
$REFQ LREF=2.,$
$REFQ ROUGH=0.001,$
```

are equivalent to

```
$REFQ SREF=1.,LREF=2.,ROUGH=0.001,$
```

The last occurrence of a namelist variable in a case is the value used for the calculations.

The three namelist inputs

```
$REFQ SREF=1.,$
$FLTCON NMACH=2.,MACH=1.0,2.0,$
$REFQ SREF=2.,$
```

are equivalent to

```
$REFQ SREF=2.,$
$FLTCON NMACH=2., MACH=1.0, 2.0,$
```

Any variable can be initialized by using the constant UNUSED; for example, LREF=UNUSED sets the reference length to its initialized value.

Certain variables are input as arrays instead of single values, such as ALPHA. If the array list is too long to fit on one card, it must be continued on the following card

with the variable name repeated and the array index of the first continued value. For example:

```
$FLTCON
NALPHA=20., ALPHA=0.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,
ALPHA(12)=22.,24.,28.,32.,36.,40.,44.,48.,52.,
NMACH=5., MACH=0.2,0.8,1.5,2.0,3.0, ALT=0.,10000.,
ALT(3)=20000.,30000.,40000.,$
```

The namelists can be input in any order.

Only those namelists required to execute the case need be entered.

All Missile Datcom namelist inputs are either real numbers or logical constants. Integer constants will produce a nonfatal error message from the error checking routine and should be avoided. All namelist and variables names must be input in capital letters. This also applies to numerical values input in “E” format, i.e. REN=6.0E06 is acceptable, while REN=6.0e06 is not.

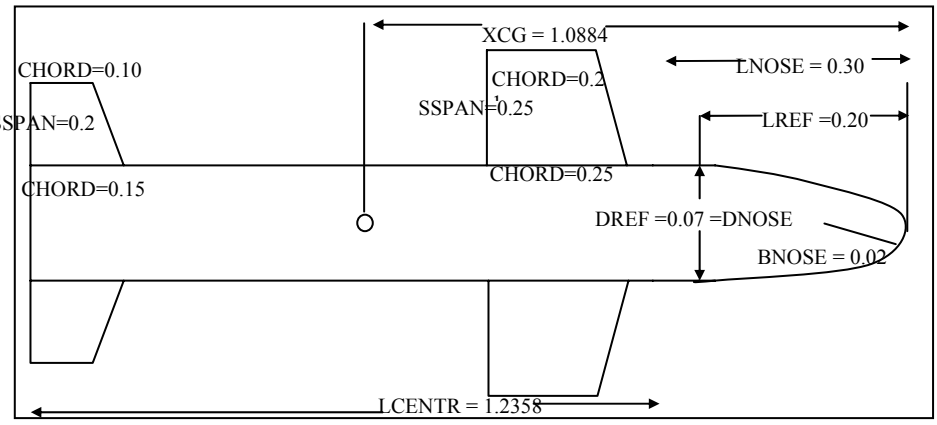


Figure 99 Missile Model

General Properties of Infrared Guided Surface to Air Missiles is shown in Table 14 and a generic IR guided missile is shown in Figure 108.

A.2.1.2.1. Reference Quantities

Inputs for this namelist are optional and are defined in Table 10. A vehicle scale factor (SCALE) permits the user to input a geometry that is scaled to the size desired. This scale factor is used as a multiplier to the user defined geometry inputs; it is not applied to the user input reference quantities (SREF, LREF, LATREF). If no reference quantities are input, they are computed based upon the scaled geometry. XCG is input relative to the origin of the global coordinate system (X=0, Figure 1) and is scaled using SCALE. Instead of specifying the surface roughness height ROUGH, the surface Roughness Height Rating (RHR) can be specified. The RHR represents the arithmetic average roughness height variation in millionths of an inch.

Table 10 Reference Quantity Inputs

Name	**Explanation**
SREF	Reference area
LREF	Longitudinal reference length
LATREF	Lateral reference length
XCG	Longitudinal position of C.G. (+aft)
ZCG	Vertical position of C.G. (+up)
BLAYER	Boundary layer type: TURB for fully turbulent NATURAL for natural transition
ROUGH	Surface roughness height
RHR	Roughness Height Rating
SCALE	Vehicle scale factor

A typical input for the reference quantity input is shown in Figure 100:

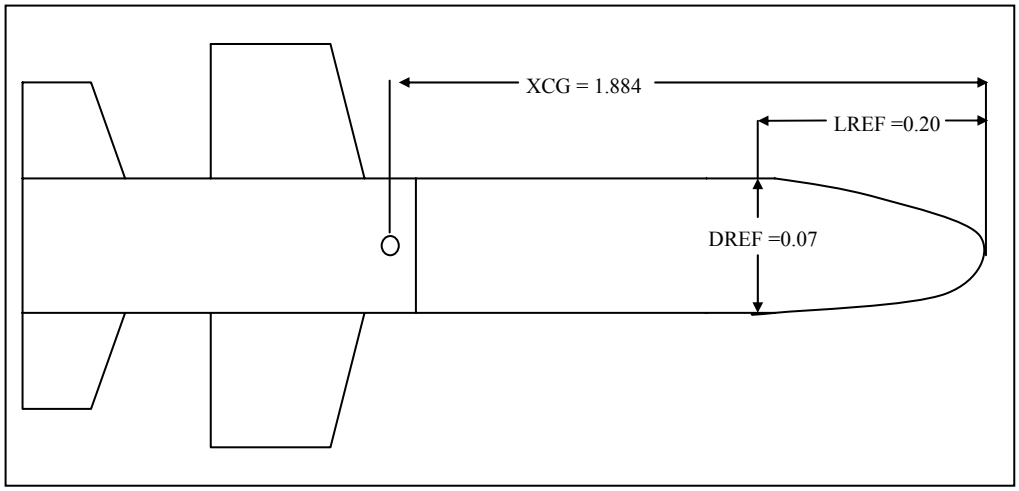


Figure 100 A Typical Input for the Reference Quantity Input

A Reference Input code example is given below:

```
$REFQ
    SREF=0.159043,   reference area
    LREF=0.20,       Longitudinal reference length
    XCG=1.88400,     Longitudinal position of center of gravity
$
```

A.2.1.2.2. Flight Conditions

This namelist defines the flight conditions to be run for the case. The program is limited to 20 angles of attack and 20 Mach number/altitude combinations per case at a fixed sideslip angle, aerodynamic roll angle, and panel deflection angle. Therefore, a "case" is defined as a fixed geometry with variable Mach number/altitude and angles of attack.

A typical input for the flight conditions input is as follows:

$FLTCON

NALPHA=15.0,	*number of angle of attacks*
ALPHA=-21.0,-18.0,-15.0,-12.0,	*angle of attacks*
ALPHA(5)= -9.0,-6.0,-3.0,0.0,3.0,6.0,9.0,	*angle of attacks. (5) means this raw starts with he fifth angle of attack*
ALPHA(12)= 12.0,15.0,18.0,21.0,	*angle of attacks. (12) means this raw starts with he twelfth angle of attack*
NMACH=13.0,	*number of mach numbers*
MACH=0.40,0.60,0.80,0.90,0.95,1.00,1.10,	*Mach numbers*
MACH(8)= 1.20,1.40,1.60,1.80,2.00,2.20,	*Mach numbers. (8) means this raw starts with he eighth Mach number*
BETA=5,	*Sideslip angle*
ALT=4000.0,	*altitude*

$

In addition, the inputs in the Table 11 can also be used.

Table 11 Additional Flight Condition Inputs

Name	**Explanation**
REN	Reynolds numbers per unit length
VINF	Freestream velocities
TINF	Freestream static temperatures
PINF	Freestream static pressures

Any of the following combinations satisfy the minimum requirements for calculating atmospheric conditions (Mach and Reynolds number):

- MACH and REN
- MACH and ALT
- MACH and VINF and TINF
- VINF and ALT
- VINF and TINF and PINF

A.2.1.2.3. Axisymmetric Body Inputs

This namelist defines the axisymmetric body properties of the missile. This namelist can be used according to options.

OPTION 1: The geometry is divided into nose, center body, and aft body sections. The shape, overall length, and base diameter for each section are specified. Note that not all three body sections need to exist on a configuration; for example, a nose-cylinder configuration does not require definition of an aft body.

Inputs for OPTION 1 can be summarized as in Table 12:

Table 12 Axisymmetric Body Inputs for Option 1

Name	**Explanation**
XO or X0	Longitudinal coordinate of nose tip
TNOSE	Type of nose shape: CONICAL or CONE (cone) OGIVE (tangent ogive) POWER (power law) HAACK (L-V constrained) KARMAN (L-D constrained)
POWER	Exponent, n, for power law shape: (r/R)=(x/L)n

Table 12 continued

LNOSE	Nose length
DNOSE	Nose diameter at base
BNOSE	Nose bluntness radius or radius of truncation
TRUNC	Truncation flag: .TRUE. if nose is truncated .FALSE. if nose is not truncated
LCENTR	Centerbody length
DCENTR	Centerbody diameter at base
TAFT	Type of afterbody shape: CONICAL or CONE (cone) OGIVE (tangent ogive)
LAFT	Afterbody length
DAFT	Afterbody diameter at base (must be > 0. And not equal to DCENTR)
DEXIT	Nozzle diameter for base drag calculation DEXIT not defined gives zero base drag DEXIT = 0. Gives "full" base drag DEXIT= exit gives base drag of annulus around exit only
BASE	Flag for base plume interaction: .TRUE. for plume calculations .FALSE. for no plume calculations
BETAN	Nozzle exit angle
JMACH	Jet Mach number at nozzle exit
PRAT	Jet/freestream static pressure ratio
TRAT	Jet/freestream stagnation temperature ratio

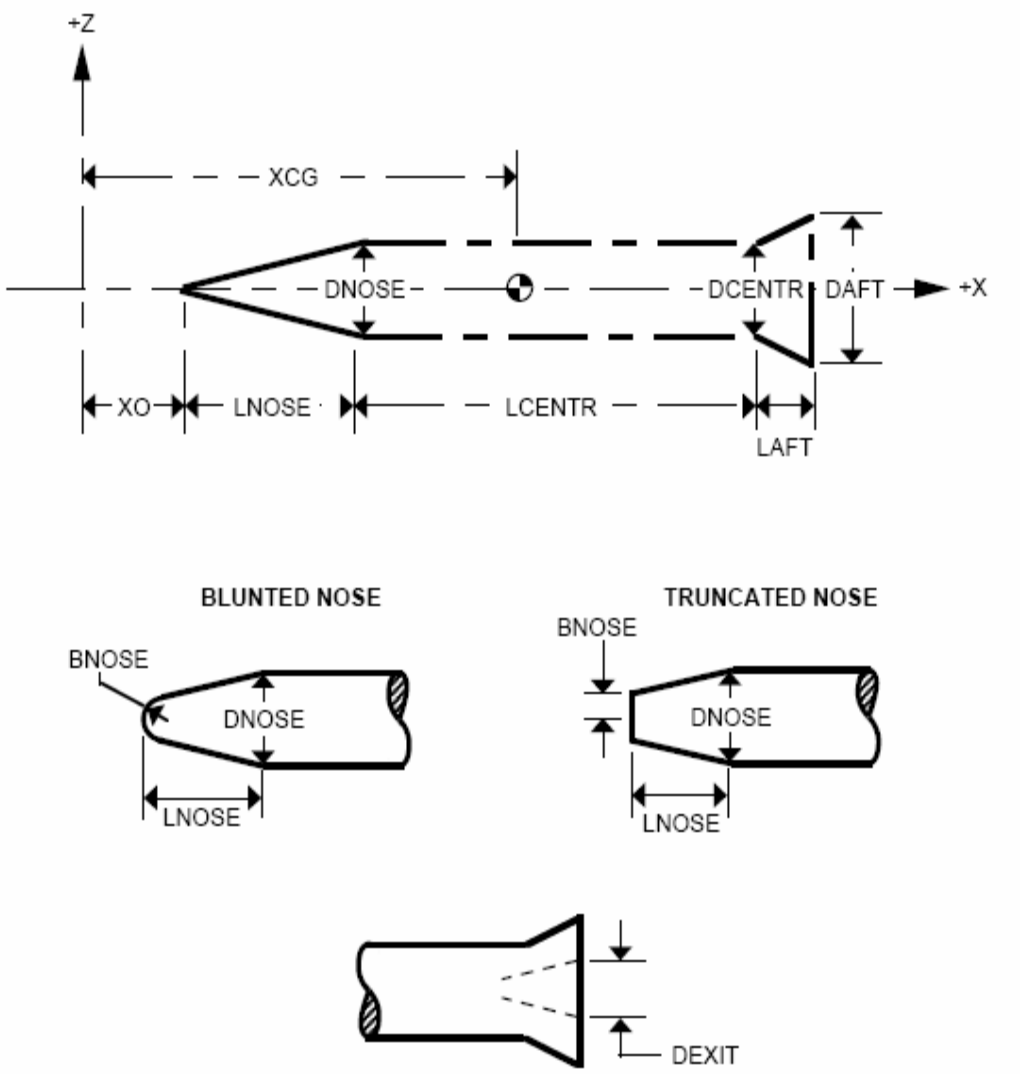


Figure 101 Asymmetric Body Structure

OPTION 2: The longitudinal stations and corresponding body radii are defined, from nose to tail. Inputs for OPTION 2 can be summarized as in Table 13.

Table 13 Axisymmetric Body Inputs for Option 2

Name	**Explanation**
XO or X0	Longitudinal coordinate of nose tip
NX	Number of input stations (2 < NX < 50)
X	Longitudinal coordinates X(NX) must be the end of the body
R	Radius at each X station L
DISCON	Indices of X stations where the surface slope Is discontinuous. Example: X(1)=0.,4.,8.,12.,16.,20., DISCON=3., Defines a discontinuity at X=8. (third value)
BNOSE	Nose bluntness radius or radius of truncation
TRUNC	Truncation flag: .TRUE. if nose is truncated .FALSE. if nose is not truncated
DEXIT	Nozzle diameter for base drag calculation DEXIT not defined gives zero base drag DEXIT = 0. Gives “full” base drag DEXIT= exit gives base drag of annulus Around exit only

A typical input for the axisymmetric body input is shown in Figure 102:

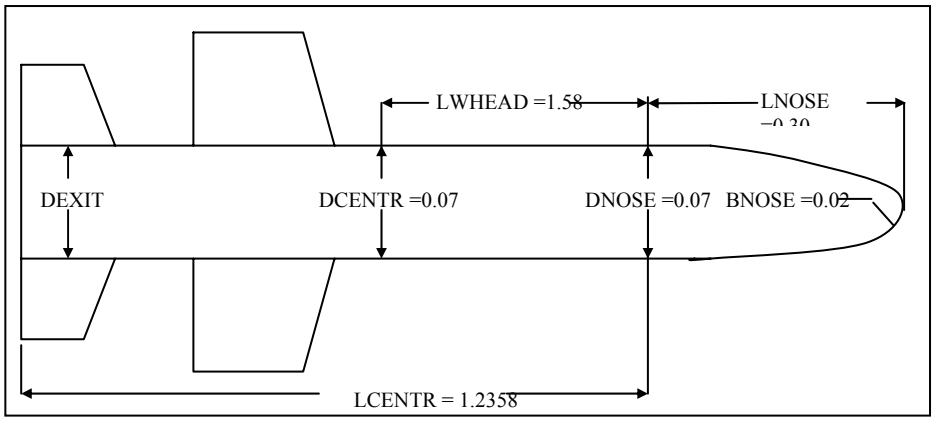


Figure 102 A Typical Input for the Axisymmetric Body Input

$AXIBOD

A typical input for the axibod conditions input is as follows:

TNOSE=OGIVE,	*type of nose shape*
LNOSE=0.30,	*nose length*
DNOSE=0.0700,	*nose diameter at base*
BNOSE=0.02,	*radius of truncation*
LCENTR=1.23580000,	*centerbody length*
DCENTR=0.0700,	*centerbody diameter at base*
DEXIT=0.070,	*nozzle diameter for base drag calculation*

FINSET-1 (WING) INPUTS

A typical input for the finest-1 input is as follows:

$FINSET1

SSPAN=0.0,0.25,	*semi span locations*
CHORD=0.250000,0.219423,	*panel chord at each semi-span location*
XLE=0.58580000000,	*distance from missile nose to chord leading edge*
SWEEP=15.000000,STA=0.0,	*sweepback angle at each span location. STA means chord station used in measuring sweep. STA=0.0 is leading edge, STA=1 is trailing edge*
NPANEL=4.0,	*Numbers of panels in fin set*
PHIF=0.,90.,180.,270.,	*roll angle of each fin measured clockwise from top vertical center looking forward*
ZUPPER=0.01425,0.01425,	*thickness to chord ratio of upper surface. Input separate value for each span station.*
LMAXU=0.0344,0.0344,	*fraction of chord from section leading edge to maximum thickness of upper surface. Input separate*

value for each span station.

LFLATU=0.9312,0.9312,$ *fraction of chord of constant tickness*

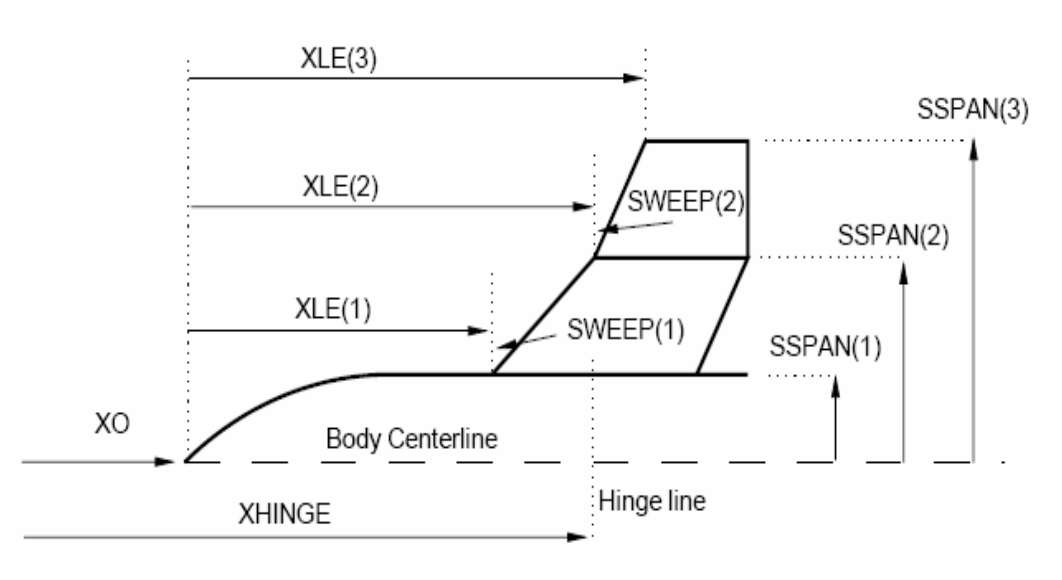


Figure 103 Finset Structure I

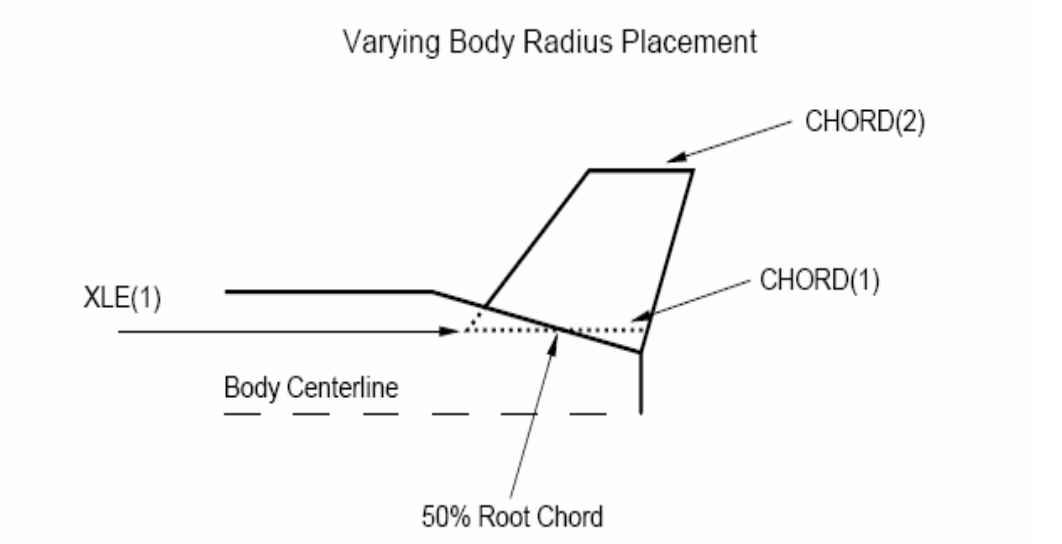


Figure 104 Finset Structure II

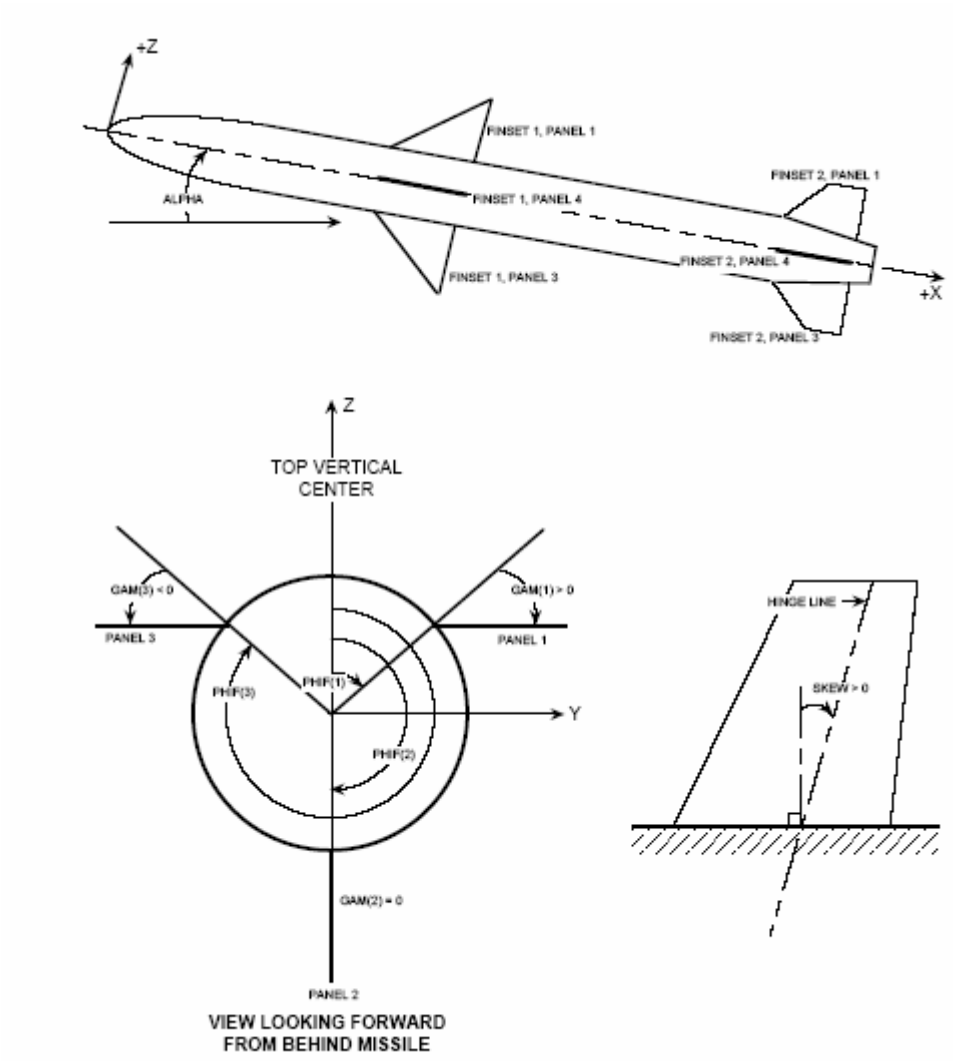


Figure 105 Finset Structure III

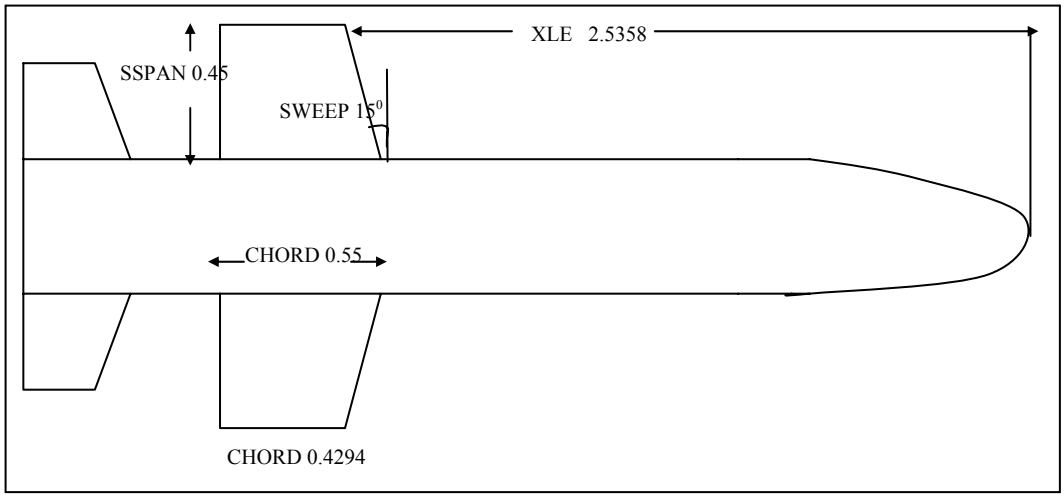


Figure 106 A Typical Finset Input

FINSET-2 (TAIL) INPUTS*

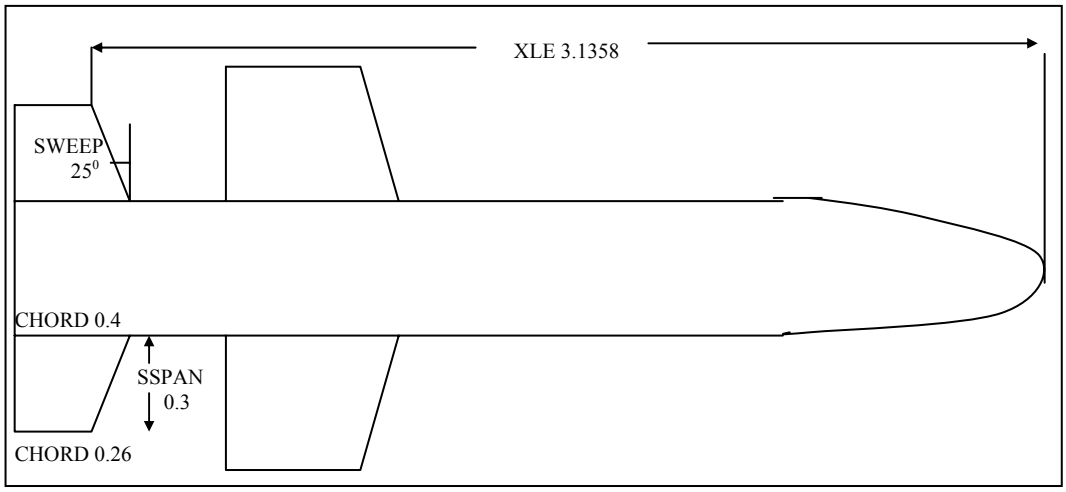


Figure 107 Tail Finset Structure

$FINSET2

SSPAN=0.0,0.200000,
CHORD=0.150000,0.100108,
XLE=1.4358000000,
SWEEP=25.000000,STA=0.0,
NPANEL=4.0,
PHIF=0.,90.,180.,270.,
ZUPPER=0.01425,0.01425,
LMAXU=0.0344,0.0344,
LFLATU=0.9312,0.9312,$

$DEFLCT

This namelist permits the user to fix the incidence angle for each panel in each fin set. The variables are given in Table 14. Note that the panel numbering scheme is assumed to be that shown in Figure 108. The array element of each deflection array corresponds to the panel number. The scheme for specifying deflection angles is unique, yet concise. The scheme used is based upon the body axis rolling moment:

"In Missile Datcom a positive panel deflection is one which will produce a negative (counterclockwise when viewed from the rear) roll moment increment at zero angle of attack and sideslip."

DELTA1=4.000,4.000,4.000,4.000,	*deflection angles for each panel in fin set 1*
DELTA2=0.000,0.000,0.000,0.000,	*deflection angles for each panel in fin set 2*
XHINGE	*Position of the panel hinge line for Each fin set, measured from the coordinate system origin. XHINGE is NOT measured from the Body nose unless XO=0.*
SKEW	*Hinge line sweepback for each fin set*

Table 14 General Properties of Infrared Guided Surface to Air Missiles

	Alternative Name	Missile Length/ /Diam./Span (cm)	Mass (kg)	Range Min/Max (km)	Speed (Mach)	Altitude (m)	Guidance
----------	"SA-9 Gaskin"	180 / 11 / 30	30	0.5/6	1.5	/5000	IR
Ain Sakr	Egyptian Strela	134 / 7 / 10	10	/4.4	-	50/2400	IR
Blowpipe		140 / 8 / 28	11	0.5/3	1.5	/2000	IR/CLOS
Crotale		293 / 16 / 54	85	0.5/10	2.3	4/5000	IR/BR
FIM-43A	Redeye	122 / 7 / 14	8	/3.4	-	/1500	IR
FIM-92A	Stinger	152 / 7 / 9	10	0.3/5.5	2.2	30/4800	IR
FIM-92B/C	***Stinger***	***152 / 7 / 9***	***10***	***0.2/5.5***	***2.2***	***30/4800***	***IR/UV***
HN-5A		-	13	0.8/4	-	80/2500	IR
Igla-1 9K310	"SA-16 Gimlet"	-	13	0.5/6	2.5 -	/5000	IR
MIM-72A	Chaparral	291 / 13 / 54	84	0.5/4	2.5	15/3000	IR
Mistral		-	17	0.3/6	1.5	/3000	IR
QW-1		-	17	0.5/5	-	30/4000	IR
RIM-116		/ 13 / 43	73	/7	1	-	IR+RF
Roland 3		240 / 16 / 50	67	0.5/8	1.7	20/4000	IR/SACLOS
SAVH-IRS		-	-	-	-	-	dual-colour IR
Strela	"SA-7/SA-N-5 Grail"	135 / - / -	9	0.5/3.7	1.8	45/3000	IR
Strela-2M	"SA-7B Grail"	-	10	0.5/5.6	1.8 -	45/4300	IR
TAN-SAM	"Type 81"	270 / 16 / 60	100	0.5/7	2.4	/3000	IR
Tien Kung 1		-	-	/70	4+	100/24k	SAR/IR
	"SA-13 Gopher"	-	-	0.5/5	2	-	IR
----------	"SA-14/SA-N-8 Gremlin"	-	10	0.5/6	1.8	/5000	IR

Figure 108 Generic IR Guided Missile

A.2.2. Output File (for006.dat)

Missile DATCOM provides aerodynamic coefficients for the missile characterized by the input file according to angle of attacks and the 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.

Output file of Missile DATCOM includes the items in Table 15as a summary:

Table 15 Output Files Contents

Name	Explanation
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

Table 15 continued

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, measures\d from the moment reference center, divided by reference length. Positive values indicate 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”.

A.2.3. Automated Processing Recommendations for the Output File

Automated processing of the outputs is a very complex issue due to a large number of coefficients and numbers. The following procedure should be followed in order to obtain aerodynamic coefficient lookup tables:

- Open the for006.dat file in Excel format.
- Be careful for the "." and "," changes. Because Excel reads "." as decimal place separator and txt files reads "," as decimal place separator.
- Save the excel file to the same folder with a different name. (for example deneme.xls)
- Use the
  [NUMERIC,TXT,RAW]=xlsread('.....\deneme.xls');
  command in Matlab to assign all cells of the deneme.xls into the Matlab side.
- Use the command below to found the "axial force coefficient" for example:

```
clc;
[satir,kolon]=size(RAW);
topla=zeros(satir,1);
for i=1:satir
   k=RAW{i,5};
   if isstr(RAW{i,5})
      k=RAW{i,5};
      if strcmp(k,'CA')
         topla(i)=i;
      end
   end
   clear k;
end
sifirolmayan_ca_lar=nonzeros(topla);

clear a;
j=1;
m=1;
clear i;
n=length(sifirolmayan_ca_lar);
for i=1:n
   a(j,m)=RAW{(sifirolmayan_ca_lar(i)+2),5};
   a(j+1,m)=RAW{(sifirolmayan_ca_lar(i)+3),5};
   a(j+2,m)=RAW{(sifirolmayan_ca_lar(i)+4),5};
   a(j+3,m)=RAW{(sifirolmayan_ca_lar(i)+5),5};
   a(j+4,m)=RAW{(sifirolmayan_ca_lar(i)+6),5};
   a(j+5,m)=RAW{(sifirolmayan_ca_lar(i)+7),5};
   a(j+6,m)=RAW{(sifirolmayan_ca_lar(i)+8),5};
   a(j+7,m)=RAW{(sifirolmayan_ca_lar(i)+9),5};
   a(j+8,m)=RAW{(sifirolmayan_ca_lar(i)+10),5};
```

```
    a(j+9,m)=RAW{(sifirolmayan_ca_lar(i)+11),5};
    a(j+10,m)=RAW{(sifirolmayan_ca_lar(i)+12),5};
    a(j+11,m)=RAW{(sifirolmayan_ca_lar(i)+13),5};
    a(j+12,m)=RAW{(sifirolmayan_ca_lar(i)+14),5};
    a(j+13,m)=RAW{(sifirolmayan_ca_lar(i)+15),5};
    a(j+14,m)=RAW{(sifirolmayan_ca_lar(i)+16),5};
    m=m+1;
end
clear s;
for s=1:9
    Cx(:,:,s)=a(:,s*13-12:s*13);
end
```

Some numbers in this code have to be rearranged for the appropriate input file.

In the case that a coefficient that is not directly written to output file like aileron derivative coefficient, a small code must be written to subscribe two output files.

Standard output for the file for005.dat above is shown inTable 16:

Table 16 Standard output for the file for005.dat

********* FLIGHT CONDITIONS AND REFERENCE QUANTITIES *********

MACH NO = 0.70	REYNOLDS NO = 1.010E+07 /M
ALTITUDE= 5000.0 M	DYNAMIC PRESSURE= 18541.54 N/M**2
SIDESLIP = 0.00 DEG	ROLL = 0.00 DEG
REF AREA = 0.159 M**2	MOMENT CENTER = 1.088 M
REF LENGTH =0.20 M	LAT REF LENGTH = 0.20 M

	----- LONGITUDINAL -----			-- LATERAL DIRECTIONAL --		
ALPHA	CN	CM	CA	CY	CLN	CLL
-9.00	-0.748	-1.389	0.023	0.000	0.000	0.000
-8.00	-0.666	-1.229	0.023	0.000	0.000	0.000
-7.00	-0.582	-1.061	0.023	0.000	0.000	0.000
-6.00	-0.496	-0.896	0.023	0.000	0.000	0.000
-5.00	-0.412	-0.731	0.023	0.000	0.000	0.000
-4.00	-0.329	-0.570	0.023	0.000	0.000	0.000
-3.00	-0.245	-0.416	0.023	0.000	0.000	0.000
-2.00	-0.162	-0.271	0.023	0.000	0.000	0.000
-1.00	-0.081	-0.133	0.023	0.000	0.000	0.000
0.00	0.000	0.000	0.023	0.000	0.000	0.000
1.00	0.081	0.133	0.023	0.000	0.000	0.000
2.00	0.162	0.271	0.023	0.000	0.000	0.000
3.00	0.245	0.416	0.023	0.000	0.000	0.000

Table 16 continued

4.00	0.329	0.570	0.023	0.000	0.000	0.000
5.00	0.412	0.731	0.023	0.000	0.000	0.000
6.00	0.496	0.896	0.023	0.000	0.000	0.000
7.00	0.582	1.061	0.023	0.000	0.000	0.000
8.00	0.666	1.229	0.023	0.000	0.000	0.000
9.00	0.748	1.389	0.023	0.000	0.000	0.000

ALPHA	CL	CD	CL/CD	X-C.P.
-9.00	-0.735	0.139	-5.270	1.858
-8.00	-0.656	0.115	-5.694	1.845
-7.00	-0.575	0.094	-6.143	1.823
-6.00	-0.491	0.075	-6.583	1.806
-5.00	-0.409	0.059	-6.961	1.775
-4.00	-0.327	0.046	-7.139	1.730
-3.00	-0.244	0.036	-6.830	1.698
-2.00	-0.161	0.029	-5.659	1.668
-1.00	-0.080	0.024	-3.310	1.647
0.00	0.000	0.023	0.000	1.647
1.00	0.080	0.024	3.310	1.647
2.00	0.161	0.029	5.659	1.668
3.00	0.244	0.036	6.830	1.698
4.00	0.327	0.046	7.139	1.730
5.00	0.409	0.059	6.961	1.775
6.00	0.491	0.075	6.583	1.806
7.00	0.575	0.094	6.143	1.823
8.00	0.656	0.115	5.694	1.845
9.00	0.735	0.139	5.270	1.858

X-C.P. MEAS. FROM MOMENT CENTER IN REF. LENGTHS, NEG. AFT OF MOMENT CENTER

********* FLIGHT CONDITIONS AND REFERENCE QUANTITIES *********

MACH NO = 0.70 REYNOLDS NO = 1.010E+07 /M
ALTITUDE = 5000.0 M DYNAMIC PRESSURE = 18541.54 N/M**2
SIDESLIP = 0.00 DEG ROLL = 0.00 DEG
REF AREA = 0.159 M**2 MOMENT CENTER = 1.088 M
REF LENGTH = 0.20 M LAT REF LENGTH = 0.20 M

Table 16 continued

---------- DERIVATIVES (PER DEGREE) ----------

ALPHA	CNA	CMA	CYB	CLNB	CLLB
-9.00	0.0805	0.1572	-0.0356	-0.0400	0.0105
-8.00	0.0828	0.1639	-0.0350	-0.0459	0.0086
-7.00	0.0850	0.1663	-0.0356	-0.0468	0.0075
-6.00	0.0850	0.1649	-0.0356	-0.0460	0.0065
-5.00	0.0834	0.1632	-0.0353	-0.0445	0.0054
-4.00	0.0835	0.1575	-0.0349	-0.0428	0.0042
-3.00	0.0834	0.1494	-0.0343	-0.0407	0.0031
-2.00	0.0822	0.1417	-0.0336	-0.0382	0.0021
-1.00	0.0812	0.1354	-0.0330	-0.0362	0.0010
0.00	0.0808	0.1331	-0.0327	-0.0355	0.0000
1.00	0.0812	0.1354	-0.0330	-0.0362	-0.0010
2.00	0.0822	0.1417	-0.0336	-0.0382	-0.0021
3.00	0.0834	0.1494	-0.0343	-0.0407	-0.0031
4.00	0.0835	0.1575	-0.0349	-0.0428	-0.0042
5.00	0.0834	0.1632	-0.0353	-0.0445	-0.0054
6.00	0.0850	0.1649	-0.0356	-0.0460	-0.0065
7.00	0.0850	0.1663	-0.0356	-0.0468	-0.0075
8.00	0.0828	0.1639	-0.0350	-0.0459	-0.0086
9.00	0.0805	0.1572	-0.0356	-0.0400	-0.0105

PANEL DEFLECTION ANGLES (DEGREES)

SET	FIN 1	FIN 2	FIN 3	FIN 4	FIN 5	FIN 6	FIN 7	FIN 8
1	0.00	0.00	0.00	0.00				
2	0.00	0.00	0.00	0.00				