CONTROLLER DESIGN FOR A TRAJECTORY TRACKING MISSILE BY USING INVERSE DYNAMICS METHOD

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND
ELECTRONICS ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By

Hüseyin Türkoğlu

January 1996

CONTROLLER DESIGN FOR A TRAJECTORY TRACKING MISSILE BY USING INVERSE DYNAMICS METHOD

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Hüseyin Türkoğlu

January 1996

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

Assoc. Prof. Dr. Ö. Morgül(Principal Advisor)

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

Prof. Dr. Erol Sezer

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

B. Özgüler

Prof. Dr. A. B. Özgüler

Approved for the Institute of Engineering and Sciences:

M. Baray

Prof. Dr. Mehmet Baray
Director of Institute of Engineering and Sciences

ABSTRACT

CONTROLLER DESIGN FOR A TRAJECTORY TRACKING MISSILE BY USING INVERSE DYNAMICS METHOD

Hüseyin Türkoğlu
M.S. in Electrical and Electronics Engineering
Advisor: Assoc. Prof. Dr. Ö. Morgül
January 1996

In this thesis, a controller for a missile desired to track a given trajectory is designed by using the nonlinear inverse dynamics(NID) method. The nonlinear dynamic equations describing the motion of the missile are linearized from input to the output by using the NID method without any need to the linearizing approximations as in the conventional linearizing method. Hence, designed controller for the linearized system stabilizes the nonlinear system over entire range of the operating points. Two approaches used in the controller design are presented in this thesis.

Keywords : Autopilot, guidance, missile, nonlinear systems, inverse dynamics, gain scheduling, two-time scale systems

ÖZET

TERS DİNAMİK YÖNTEMİ KULLANILARAK YÖRÜNGE TAKİP EDEN BİR FÜZE İÇİN DENETLEYİCİ TASARIMI

Hüseyin Türkoğlu
Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans
Danışman: Assoc. Prof. Dr. Ö. Morgül
Ocak 1996

Bu tezde, doğrusal olmayan ters dinamik yöntemi kullanılarak yörünge takip eden bir füze için kontrol sistemi tasarlanmıştır. Bu yöntem kullanılarak, klasik anlamdaki doğrusallaştırma yönteminde yapılan herhangi bir varsayıma gerek kalmadan, sistemin doğrusal olmayan girdi çıktı ilişkisi doğrusallaştırılmıştır. Dolayısıyla, tasarlanan doğrusal denetleyici, sistemi daha geniş bir alanda kontrol edebilmiştir. Denetleyici tasarımında kullanılan iki tür yaklaşım bu tezde anlatılmıştır.

Anahtar Kelimeler : Otopilot, güdüm, füze, doğrusal olmayan sistemler. ters dinamik, kazanç tablolaması, iki zamanlı sistemler.

ACKNOWLEDGMENTS

I would like to thank to Assoc. Prof. Dr. Ö. Morgül for his supervision, guidance, suggestions and encouragement through the development of this thesis.

I would also like to thank to Prof. Dr. Erol Sezer and Prof. Dr. A. B. Özgüler for reading and commenting on the thesis.

It is a pleasure to express my thanks to all my friends for their valuable discussions and helps.

TABLE OF CONTENTS

1 Introduction **1**

2 The Equations of Motion **4**

2.1 Coordinate Systems 4

2.2 Basic Dynamic Equations 6

2.2.1 Translational Dynamics 6

2.2.2 Rotational Dynamics 8

2.3 The Attitude of Missile with respect to the Earth 10

2.4 Transformation Between Coordinate Frames 11

2.5 Kinematic Equations 12

2.6 Modeling of the Aerodynamic Forces 13

2.7 Separation of The Equations of Motion 15

3 Input/Output (I/O) Linearization Using Inverse Dynamics Method **17**

3.1 I/O Linearization Problem 21

3.2 I/O Linearization of SISO Non-linear Systems 22

3.3 Multi Input Multi Output (MIMO)Case 26

4 The Use of Nonlinear Inverse Dynamics(NID) Method in Controller Design for Trajectory Tracking Missile **29**

4.1 The First Method 29

4.1.1 The Guidance Design by Using Inverse Dynamics Method 30

4.2 Acceleration Autopilot Design 36

4.3 Second Method 41

5 SIMULATIONS **44**

5.1 The Simulation Program 44

5.2 Simulation Results 46

6 CONCLUSION **50**

LIST OF FIGURES

2.1 Body Frame 6

2.2 Euler Angles 10

2.3 Angle of attack and sideslip angle 14

3.1 I/O linearization scheme 22

3.2 State feedback with inverse dynamics 28

4.1 Guidance and Autopilot 30

4.2 Autopilot Configuration 39

5.1 Desired trajectory 46

5.2 Position Error 47

5.3 Angle of attack and Canard deflection 47

5.4 Desired trajectory for 6-DOF simulation 48

5.5 Position error for 6-DOF simulation 49

5.6 Position errors in Y and Z axis 49

5.7 Position Error for Second Method 49

5.8 Angle of Attack and Canard Deflection 49

Chapter 1

Introduction

All systems in the nature are nonlinear. Hence, analyzing and controlling these systems require deep understanding of the nonlinearity. This may not be obvious when we consider the linearized version of the nonlinear system. As long as the nonlinear system is in the vicinity of a chosen operating point, the designed linear controller for the linearized version of the nonlinear system at this operating point does its job. However, when the operating point of the system changes due to the external disturbance or some other reasons, the system may become unstable. For this reason, we have to develop a method to prevent this possibility. A straight-forward way of overcoming this problem may be "**gain scheduling method**", [3], [4]. In this method, linear-time invariant controllers are designed for each linearized representation of the system at the selected operating points, so that the stability and certain performance objectives are achieved; these controllers are then linked together in order to obtain a single controller for the entire range of the system operation.

In between operating points, there is no guarantee that the system remains stable. Hence, extensive computer simulations are needed to validate the robustness of the controller designed by the gain scheduling technique. There are some heuristic rules-of-thumb in choosing the scheduling variable for the gain scheduling. The scheduling variable should vary slowly and capture the plant's nonlinearity, [3].

Dynamic inversion method is another way of designing controllers for the nonlinear systems [1]. The nonlinear system may be transformed into input/output linear form by using this method without any approximations.

As a result, the designed nonlinear controller stabilizes the nonlinear system over the entire range of operating points except for certain singular points. Nevertheless, this method requires the exact knowledge of the mathematical model of the nonlinear system. However, the exact modeling is not possible in practice. So, some robust control methods should be combined with this method, [6].

There are also some other nonlinear controller design techniques available. In [5], a nonlinear feedback law is derived for a nonlinear system by combining the linear controllers designed for a family of operating points. In [8], a partial dynamic inversion is proposed to overcome some difficulties in exact inversion theory. The model following approach is also a way of designing nonlinear controller, [13].

In this thesis, the controller for a trajectory tracking missile is designed by using dynamic inversion method with some modifications. The missile is controlled by using the four canards located on the surface of the missile. The deflection of canards has the effect of changing the aerodynamic forces and moments applied to the missile. Hence, the position of the missile can be controlled by deflecting the appropriate canard with a required deflection. The aim is to design a controller computing the required canard deflections so that the missile tracks a given trajectory.

Missile dynamics is composed of 12-state nonlinear dynamical equations. The derivation of these dynamical equations are presented in chapter 2. The modeling of the aerodynamic forces and moments are given in this chapter. For more details, see [9], [10], [11]..

In chapter 3, the input/output(i/o) linearization (dynamic inversion) method is presented. The second section considers only the nonlinear single input single output(SISO) system. I/O linearization for multi input multi(MIMO) output case is explained in the third section. The more detailed analysis of these techniques can be found in [12].

In chapter 4, the controller design strategies for the trajectory tracking missile are presented. Two approaches used in controller design are given. The dynamics of the missile is separated into two parts as slow and fast dynamics in the first approach. This method has been known as "**two-time scale approach**" and has been used in flight controller design for an aircraft, [2]. The two-time scale approach to the system resulted in order reduction, which simplifies the controller design. In our case, the separated parts are controlled independently under certain constraints. For example,the difference between speeds of the dynamics of these separated parts should be large enough. This rule puts some restriction on the dynamics of the controllers for these parts. The controller designed for the slow part is called as **guidance unit**, and the controller designed for the fast part is called **autopilot unit**. The guidance unit computes the required acceleration for the missile to track the given trajectory. This computed acceleration is given to the autopilot as a reference command. The autopilot unit computes the required canard deflection to track the commanded acceleration.

In the guidance design, the dynamic inversion method is used. But critical model parameters have not appeared in the controller coefficients. Hence, the disadvantage of dynamic inversion method is avoided by using this approach. In autopilot design, the gain scheduling method is used.

In the second approach, the required canard deflection is directly found by using the dynamic inversion method. Hence, the gain scheduling technique is not used in this approach. Nevertheless, the critical model parameters appeared in this controller algorithm. Since dynamic inversion method is based on the cancellation of the nonlinearity, any modeling error may cause unmodeled nonlinear dynamics. As a result of this, the controlled system may be unstable due to this unmodeled nonlinearity..

In chapter 5, the simulation results are presented and interpreted. In last chapter, the conclusions are given.

Chapter 2

The Equations of Motion

This chapter is devoted to the derivation of the mathematical equations describing the motion of a missile. The missile can be considered as a rigid body having a certain geometric structure for a stable flight. Derivation of the basic dynamic equations of a rigid body in six-degree-of-freedom (6-DOF) environment is based on the Newtonian Mechanics and can be found in standard text books on robotics or flight mechanics, see [9], [10], [11]. In these derivations, certain coordinate frames are defined. Hence, before getting into derivations of the motion equations, explanation of the coordinate frames used in derivations are presented.

2.1 Coordinate Systems

Any vector p in an n dimensional space R^n can be represented in different ways depending on the chosen orthonormal basis for this space. Let $X = \{x^i\}_{i=1}^n$ be an orthonormal basis for R^n. The *coordinates* p with respect to X are denoted $[p]^X$ and are defined implicitly by the equation:

$$p = \sum_{i=1}^{n} [p]_i^X x^i$$

X is also called as an orthonormal *coordinate frame.* It can easily be proven that kth coordinate of p with respect to X is given by

$$[p]_k^X = p \cdot x^k$$

where $\cdot$ denotes the inner product operation. Let $E = \{e^1, e^2, e^3\}$ and $B = \{b^1, b^2, b^3\}$ be two coordinate frames for R^3. The representations of a vector $p \in R^3$ with respect to these coordinate frames are

$$[p]^E = \begin{bmatrix} p \cdot e^1 & p \cdot e^2 & p \cdot e^3 \end{bmatrix}^T \tag{2.1}$$

$$[p]^B = \begin{bmatrix} p \cdot b^1 & p \cdot b^2 & p \cdot b^3 \end{bmatrix}^T \tag{2.2}$$

where superscript T denotes the transpose operation.

When a vector representation in one frame is given, the representation of the same vector with respect to other coordinate frame can be found by using a coordinate transformation. Coordinate transformation corresponds to a matrix multiplication

$$[p]^E = A[p]^B \tag{2.3}$$

where the elements of the matrix A are $A_{ij} = e^i \cdot b^j$ for $i, j = 1, 2, 3$. This matrix is orthogonal which is useful for calculations. Orthogonality of a matrix is expressed in mathematical form as

$$A^{-1} = A^T$$

In flight mechanics, certain coordinate frames are used depending on the controller design techniques. In the remaining part of this thesis, two coordinate frame will be used. These are called as *inertial frame* and *body frame*. The inertial reference frame is fixed in space. So it has no motion. But in practice, the frame fixed to the Earth is used as inertial frame, for the missiles flying over short ranges. For controller design, the difference between inertial and Earth frames are not so much. Hence, the Earth frame and inertial frame will be used interchangeable after this point. The direction of orthogonal axes X, Y, and Z of the Earth frame are as follows; X is directed to the north, Y is directed to the east and Z is directed to the center of the Earth.

The body frame(or body-fixed axis system) is attached to the missile such that its center is at the center of the gravity of the missile and rotates with it. In this coordinate frame, the axis OX_b points to nose, OY_b points to right wing, and the axis OZ_b points down as shown in Figure (2.1). This coordinate frame will be denoted by $B = \{i, j, k\}$ with i, j, k in the direction of OX_b, OY_b, OZ_b respectively

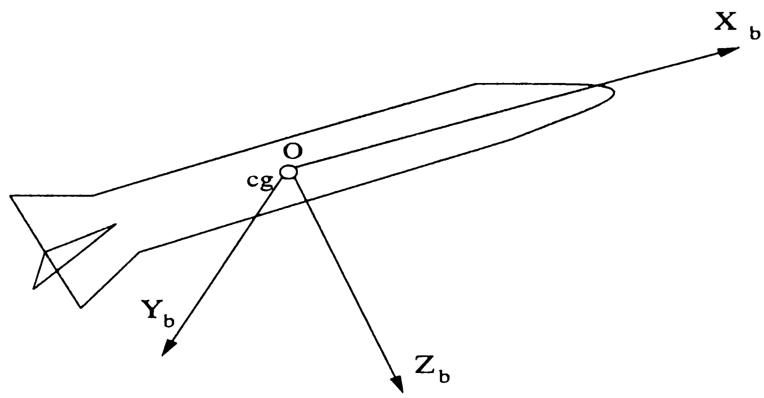

Figure 2.1: Body Frame

2.2 Basic Dynamic Equations

The mathematical equations describing the motion of a rigid body can be obtained by using the Newton's second law of motion. It states that the change of momentum of a body is equal to the force applied on it. Mathematically, translational dynamic is described by

$$\vec{F} = \frac{d}{dt}(m\vec{V}_T)]_I \tag{2.4}$$

and rotational dynamic is described by

$$\vec{M} = \frac{d\vec{H}}{dt}]_I \tag{2.5}$$

where F is the applied force, m is mass , V_T is total velocity of the object, $\vec{M}$ is torque, and $\vec{H}$ is angular momentum. The symbol $]_I$ indicates the time rate of change with respect to the inertial frame.

2.2.1 Translational Dynamics

The equation (2.4), when the mass is assumed to be constant, can be rewritten as,

$$\vec{F} = m\frac{d}{dt}(\vec{V}_T)]_I \tag{2.6}$$

The time rate of change of velocity with respect to inertial frame can be expanded as

$$\frac{d}{dt}(\vec{V}_T)]_I = \frac{d}{dt}(V_T)\vec{1}_{V_T} + \vec{\omega} \times \vec{V}_T \tag{2.7}$$

where $\vec{1}_{V_T}$ is the unit vector along the total velocity vector, $\vec{\omega}$ is the total angular velocity. When the equation (2.7) is put in the equation (2.6), we have

$$\vec{F} = m\left[\frac{d}{dt}(V_T)\vec{1}_{V_T} + \vec{\omega} \times \vec{V}_T\right] \tag{2.8}$$

Let the representation of the total translational and angular velocities of the missile in the body frame be

$$[V_T]^B = \{U, V, W\} \tag{2.9}$$

and

$$[\omega_T]^B = \{p, q, r\} \tag{2.10}$$

then the cross product of the vectors $\vec{\omega}_T$ and $\vec{V}_T$ becomes

$$\vec{\omega} \times \vec{V}_T \equiv \det\begin{bmatrix} i & j & k \\ p & q & r \\ U & V & W \end{bmatrix} \tag{2.11}$$

Here the determinant notation is used in a formal sense to indicate how to evaluate the cross product. In this case, the unit vectors {i,j,k} are treated as if they were scalars for the purpose of computing the determinant. When the equation (2.11) is expanded, we have

$$\vec{\omega} \times \vec{V}_T = i(qW - rV) + j(rU - pW) + k(pV - qU) \tag{2.12}$$

The first term in right hand side of the equation (2.8) can be written as

$$\frac{d}{dt}(V_T)\vec{1}_{V_T} = i\dot{U} + j\dot{V} + k\dot{W} \tag{2.13}$$

When the representation of the total force in the body frame is taken as

$$[F_T]^B = \{F_1, F_2, F_3\} \tag{2.14}$$

the equation (2.8) can be rewritten by using equations (2.12), (2.13) as

$$iF_1 + jF_2 + kF_3 = i(\dot{U} + Wq - Vr) + j(\dot{V} + Ur - Wp) + k(\dot{W} + Vp - Uq) \tag{2.15}$$

after equating the vector components of two sides and arranging certain terms, we have the following three translational dynamic equations.

$$\dot{U} = \frac{F_1}{m} - Wq + Vr \tag{2.16}$$

$$\dot{V} = \frac{F_2}{m} - Ur + Wp \tag{2.17}$$

$$\dot{W} = \frac{F_3}{m} - Vp + Uq \tag{2.18}$$

These equations can be written in a compact form as follows,

$$\begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} = -w. \begin{bmatrix} U \\ V \\ W \end{bmatrix} + \begin{bmatrix} \frac{F_1}{m} \\ \frac{F_2}{m} \\ \frac{F_3}{m} \end{bmatrix} \tag{2.19}$$

where w is a skew symmetric matrix of the following form

$$w = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \tag{2.20}$$

2.2.2 Rotational Dynamics

The same procedure used to derive the translational motion equations can be applied to the derivation of the angular motion equations. For a rigid body, angular momentum is defined as:

$$\vec{H} = I\vec{\omega} \tag{2.21}$$

where I is inertia matrix of the form

$$I = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{xy} & I_y & -I_{yz} \\ I_{xz} & -I_{yz} & I_z \end{bmatrix} \tag{2.22}$$

where I_x, I_y, I_z denote moment of inertia terms , and I_{xy}, I_{xz}, I_{yz} are product of inertia. For the symmetric missile under consideration, product of inertia terms are zero and $I_y = I_z$. So the equation (2.22) has simplified form

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{2.23}$$

Let the representation of the total angular momentum and torque on missile in the body frame be in the following forms,

$$[\vec{H}]^B = \{H_1, H_2, H_3\} \tag{2.24}$$

$$[\vec{M}]^B = \{M_x, M_y, M_z\} \tag{2.25}$$

By using the equations (2.24), and (2.23), the equation (2.21) can be rewritten as

$$\begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} = \begin{bmatrix} pI_x \\ qI_y \\ rI_z \end{bmatrix}$$

The rate of change of H with respect to the inertial frame is given by

$$\frac{d}{dt}(\vec{H})]_I = \frac{d}{dt}(H)\vec{1}_{H_T} + \vec{\omega} \times H \tag{2.26}$$

when the equation (2.21) is inserted in the equation (2.26), we obtain

$$\frac{d}{dt}(\vec{H})]_I = I\left[\frac{d}{dt}(\omega)\vec{1}_{H_T} + \vec{\omega} \times \vec{\omega}\right] + \vec{\omega} \times H \tag{2.27}$$

Hence, by using the equations (2.5) and (2.27), we have

$$\vec{M} = I\left[\frac{d}{dt}(\omega)\vec{1}_{H_T}\right] + \vec{\omega} \times H \tag{2.28}$$

Note that $\vec{\omega} \times \vec{\omega} = 0$ is used in the last equation. Hence, when the right hand side of the equation (2.28) is expanded and when the components of each sides are equated, we have the following three angular motion equations:

$$\dot{p} = \frac{M_x}{I_x} \tag{2.29}$$

$$\dot{q} = \frac{M_y}{I_y} - \frac{pr(I_x - I_z)}{I_y} \tag{2.30}$$

$$\dot{r} = \frac{M_z}{I_z} - \frac{pq(I_x - I_y)}{I_z} \tag{2.31}$$

2.3 The Attitude of Missile with respect to the Earth

The orientation of the missile with respect to the earth axes can be specified by using a set of angles called "Euler angles". These angles are depicted in Figure (2.2).

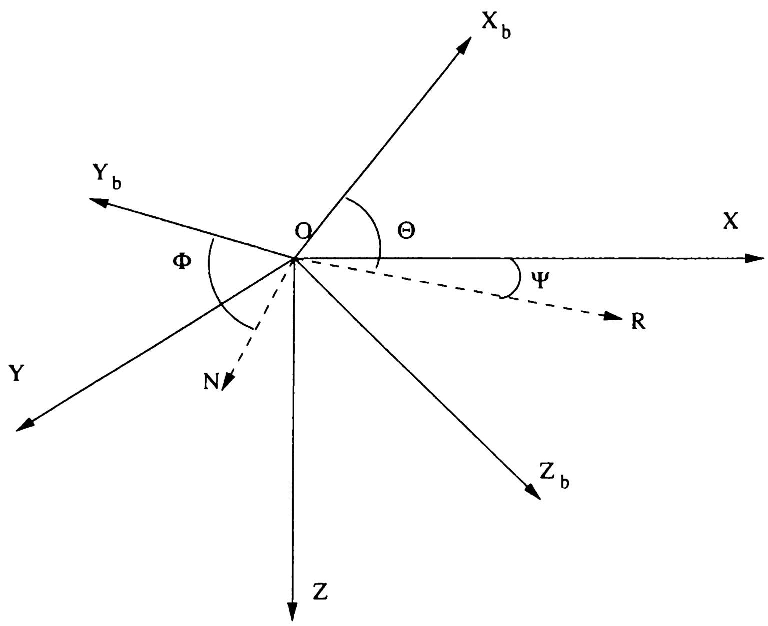

Figure 2.2: Euler Angles

The definitions of these angles are,

Φ is called as "roll" angle which is between ON and OY_b axis measured in the OY_bZ_b plane

Θ is called as "pitch" angle which is between OX_b and the projection of OX_b axis on the horizontal plane including OX and OY axes.

Ψ is called as "yaw" angle which is between OX and the projection of the OX_b axis on the horizontal plane.

2.4 Transformation Between Coordinate Frames

It is well known that, any coordinate system can be obtained from any other one by a sequence of three rotations. Each rotation corresponds to a transformation matrix. The total transformation matrix is obtained by multiplication of these three matrices. The Euler angles are used in these sequence of rotations. Let the inertial reference frame be denoted by $E = \{e_1, e_2, e_3\}$.When the inertial frame is rotated around the e_3 vector by an angle Ψ, corresponding transformation matrix will be,

$$[T]_{E_1}^{E} = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) & 0 \\ -\sin(\Psi) & \cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.32}$$

where coordinate frame $E_1 = \{e_1^{'}, e_2^{'}, e_3\}$ is rotated version of the inertial frame $E = \{e_1, e_2, e_3\}$. When the frame E_1 is rotated around $e_2^{'}$ vector by an angle Θ, resulting transformation matrix is of the form,

$$[T]_{E_2}^{E_1} = \begin{bmatrix} \cos(\Theta) & 0 & -\sin(\Theta) \\ 0 & 1 & 0 \\ \sin(\Theta) & 0 & \cos(\Theta) \end{bmatrix} \tag{2.33}$$

where $E_2 = \{e_1^{''}, e_2^{'}, e_3^{'}\}$ is new frame. When this new frame is rotated about the vector $e_1^{''}$ by an angle Φ, corresponding transformation matrix is

$$[T]_{E_3}^{E_2} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) \\ 0 & -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \tag{2.34}$$

where resulted coordinate frame $E_3 = \{e_1^{''}, e_2^{''}, e_3^{''}\}$ is body frame $B = \{i, j, k\}$.

The overall transformation matrix is found by multiplication of these fundamental rotation matrices as $[T]_B^E = [T]_{E_3}^{E_2}.[T]_{E_2}^{E_1}[T]_{E_1}^{E}$. When this multiplication is done, transformation matrix from inertial frame to body frame is found to be

$$[T]_B^E = \begin{bmatrix} c\Theta c\Psi & c\Theta s\Psi & -s\Theta \\ s\Phi s\Theta c\Psi - c\Phi s\Psi & s\Phi s\Theta s\Psi + c\Phi c\Psi & s\Phi c\Theta \\ c\Phi s\Theta c\Psi + s\Phi s\Psi & c\Phi s\Theta s\Psi - s\Phi c\Psi & c\Phi c\Theta \end{bmatrix} \tag{2.35}$$

The transformation matrix from the body frame to the Earth frame is just the transpose of the matrix given above

$$[T]_E^B = \begin{bmatrix} c\Theta c\Psi & s\Phi s\Theta c\Psi - c\Phi s\Psi & c\Phi s\Theta c\Psi + s\Phi s\Psi \\ c\Theta s\Psi & s\Phi s\Theta s\Psi + c\Phi c\Psi & c\Phi s\Theta s\Psi - s\Phi c\Psi \\ -s\Theta & s\Phi c\Theta & c\Phi c\Theta \end{bmatrix} \tag{2.36}$$

where c denotes cosine and s denotes sine of the angle. Afterwards, the notation for the transformation matrix given by (2.36) will be used without superscript or subscript, $T = [T]_E^B$.

2.5 Kinematic Equations

The position and the attitude of the missile with respect to the inertial frame are found by solving the kinematic equations given in this section. Translational kinematic equations are as follows

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = T. \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{2.37}$$

where X, Y, and Z are the position components of the missile in the inertial frame along corresponding axes.

Since T is transformation matrix, it is orthonormal. Hence we can write the following

$$T^{-1} = T^T \tag{2.38}$$

or

$$TT^T = I_3 \tag{2.39}$$

where I_3 is 3×3 identity matrix. When time derivatives of the both sides of the equation (2.39) are taken, we have

$$(\dot{T}T^T) + (\dot{T}T^T)^T = 0 \tag{2.40}$$

which requires that $\dot{T}T^T$ is a skew-symmetric matrix.

$$\dot{T}T^T = w \tag{2.41}$$

where w is given by the equation (2.20).

When we use the equations (2.41) and (2.20), we can obtain the following kinematic equations,

$$\dot{\Theta} = q\cos\Phi - r\sin\Phi \tag{2.42}$$

$$\dot{\Phi} = p + q\sin(\Phi)\tan(\Theta) + r\cos(\Phi)\tan(\Theta) \tag{2.43}$$

$$\dot{\Psi} = q\frac{\sin(\Phi)}{\cos(\Theta)} + r\frac{\cos(\Phi)}{\cos(\Theta)} \tag{2.44}$$

These equations can also be obtained by geometric projections of angular rate vectors to corresponding axes.

The angular velocities p, q, and r are measured by three rate gyros located on yhe missile and used to find Euler angles by using the above equations.

2.6 Modeling of the Aerodynamic Forces

The external forces and torques on the missile may come from different sources such as thrust, wind, and aerodynamic forces. The total external force can be written as

$$F_T = i(F_x + T_x + mg_x) + j(F_y + T_y + mg_y) + k(F_z + T_z + mg_z) \tag{2.45}$$

where m is mass of the missile, F_x, F_y, F_z are aerodynamic force components. T_x, T_y, T_z is thrust force components and g_x, g_y, g_z are gravitational acceleration components along the body axes. The aerodynamic forces and moments applied on the missile are function of some flight parameters such as angle of attack(α), side slip angle(β), mach number(M), air density(ρ), temperature(T), angular rates (p, q, r) etc.The explanations of these flight parameters are given below.

Angle of attack and side slip angles specify the orientation of the missile velocity (V_T) with respect to the body axes as shown in the Figure (2.3)

The expressions of angle of attack and side slip angle are as follows

$$\alpha = \arctan(\frac{W}{U}) \tag{2.46}$$

and

$$\beta = \arctan(\frac{V}{V_T}) \tag{2.47}$$

When W and V are small compared with U, which is acceptable , angle of attack can be approximated as

$$\alpha \cong \frac{W}{U} \tag{2.48}$$

and side slip angle is approximated as

$$\beta \cong \frac{V}{V_T} \tag{2.49}$$

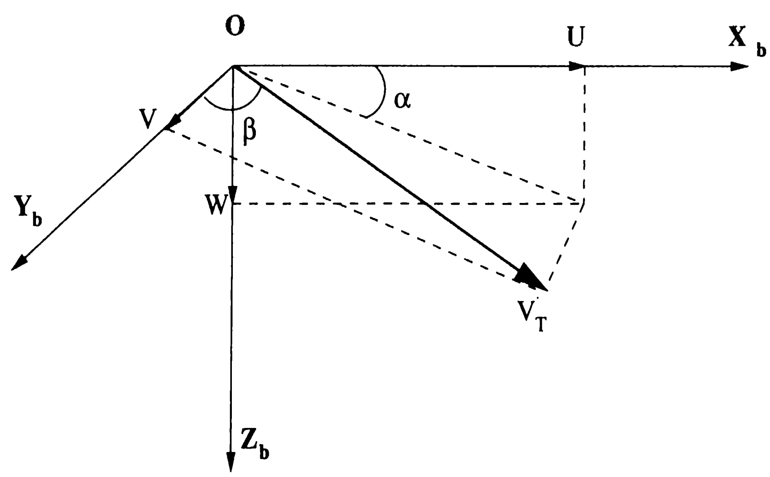

Figure 2.3: Angle of attack and sideslip angle

Mach number is defined as the ratio of the missile's velocity to the velocity of the sound

$$M = \frac{V_t}{C} \tag{2.50}$$

where speed of the sound is

$$C = \begin{Bmatrix} \sqrt{kRT_o[1 - 0.00002256h]} & \text{for } h \leq 10000 \text{ m} \\ \sqrt{0.7744.kR.T_o} & \text{for } h > 10000 \text{ m} \end{Bmatrix} \tag{2.51}$$

The form of aerodynamic forces and moments are

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = Q_d A \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \tag{2.52}$$

and

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = Q_d A d \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} \tag{2.53}$$

where d is missile diameter, A is the missile cross sectional area, Q_d is dynamic pressure. The dynamic pressure is defined as

$$Q_d = \frac{1}{2}\rho V_T^2 \tag{2.54}$$

where ρ is the air density depending on altitude(h) in the following way,

$$\rho = \left\{ \begin{array}{l} \rho_o[1 - 0.00002256h]^{4.256} \text{ for } h \leq 10000 \text{ m} \\ 0.412e^{-0.000151(h-10000)} \text{ for } h > 10000 \text{ m} \end{array} \right\} \tag{2.55}$$

The terms $C_x, C_y, C_z, C_m, C_n, C_l$ in the equations (2.52) and (2.53) are the dimensionless aerodynamic force and moment coefficients. These coefficients are dependent on the flight parameters $M, \alpha, \beta, \delta, q, r$ etc.

2.7 Separation of The Equations of Motion

There are twelve non-linear differential equations completely describing the motion of the missile. However, under certain assumptions, some equations may be separated from others.

When the missile is assumed to move in only the pitch plane(such that $\Psi = 0, r = 0, V = 0$), and without any rotating motion (so that $\Phi = 0, P = 0$), six dynamic equations are reduced to three dynamic equations with the following form,

$$\dot{U} = \frac{F_x}{m} - Wq + g_x \tag{2.56}$$

$$\dot{W} = \frac{F_y}{m} + Uq + g_z \tag{2.57}$$

$$\dot{q} = \frac{M_y}{I_y} \tag{2.58}$$

Kinematic equations for pitch plane motion are,

$$\begin{bmatrix} \dot{X} \\ \dot{Z} \end{bmatrix} = T_m \begin{bmatrix} U \\ W \end{bmatrix} \tag{2.59}$$

where T_m is of the form,

$$T_m = \begin{bmatrix} c\Theta & s\Theta \\ -s\Theta & c\Theta \end{bmatrix} \tag{2.60}$$

and

$$\dot{\Theta} = q \tag{2.61}$$

When the missile is assumed to move in only the yaw plane(such that $\Theta = 0, q = 0, W = 0$), and without any rotating motion (so that $\Phi = 0, P = 0$), the equations of motion will be

$$\dot{U} = \frac{F_x}{m} + Vr + g_x \tag{2.62}$$

$$\dot{V} = \frac{F_y}{m} - Ur + g_y \tag{2.63}$$

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

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = T_{my} \begin{bmatrix} U \\ V \end{bmatrix} \tag{2.65}$$

$$T_{my} = \begin{bmatrix} c\Psi & -s\Psi \\ s\Psi & c\Psi \end{bmatrix} \tag{2.66}$$

$$\dot{\Psi} = r \tag{2.67}$$

Chapter 3

Input/Output (I/O) Linearization Using Inverse Dynamics Method

A nonlinear system may be transformed into a i/o linear form by means of a nonlinear state feedback. When such a nonlinear state feedback is found, the resulting system will be i/o linear and the linear controller design methods can be used without any need to linearization approximations as in the conventional linearization method. This approach has found a lot of applications in certain areas with some special modifications.

The basic idea in i/o linearization is to differentiate the output function with respect to time as many times as required until the input term appears in the equations. Then, provided that the coefficient matrix multiplying the input term is nonsingular, this equation may be used to obtain linear and decoupled equations between the output and a new reference input vectors. To illustrate this idea, for simplicity, let us consider a single input single output (SISO) linear system:

$$\dot{x} = Ax + bu, \ y = cx \tag{3.1}$$

where $x \in R^n$. The time derivative of the output is

$$\dot{y} = c\dot{x} = cAx + cbu \tag{3.2}$$

Let v be a reference input. If the coefficient (cb) multiplying the input term

(u) is nonzero, we choose the input as following

$$u = \frac{v - cAx}{cb} \tag{3.3}$$

otherwise ($cb = 0$), we continue differentiation of the output until we have the input term. Let r be the order of differentiation required to have the input term. Then we have

$$y^{(r)} = cA^r x + cA^{r-1} bu \tag{3.4}$$

where $cA^{r-1}b \neq 0$. When we choose the input as

$$u = \frac{v - cA^r x}{cA^{r-1} b} \tag{3.5}$$

we have the following i/o relation, which is linear.

$$y^{(r)} = v \tag{3.6}$$

The finite number of differentiation of the output required to have an input term is called **relative degree** and it is less than the dimension of the system. This fact can be easily proven for the linear SISO system (3.1). Let us assume that the relative degree r is greater than or equal to the system dimension n. Then we have

$$cA^i b = 0 \quad for \;\; i = 0, 1, ..., n-1, n, ..., r-1 \tag{3.7}$$

We can deduce from the equation (3.7) by using the *Cayley-Hamilton* theorem that

$$cA^i b = 0 \quad for \;\; i = 0, 1, ..., \infty \tag{3.8}$$

This contradicts the definition of the relative degree. Hence, the relative degree is always less than the dimension of the system.

It can be shown that, when the relative degree is r, the vectors $\{c,\ cA,\ ...,\ cA^{r-1}\}$ are linearly independent. Moreover, one can find the vectors $\{h_{r+1},\ h_{r+2}, ...,\ h_n\}$ such that the set $\{c,\ cA,\ ...,\ cA^{r-1},\ h_{r+1},\ h_{r+2}, ...,\ h_n\}$ is linearly independent and $< h_i, b >= 0 \;\; i = r+1, .., n$. With the new coordinates

$$\begin{aligned} z_i &= cA^{i-1} x \qquad i = 1, 2, ..., r \\ z_j &= h_j x \qquad\quad j = r+1, ..., n \end{aligned}$$

and the control law (3.5), state equations in the new variables become

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
& \\
\dot{z}_r &= v \\
\dot{z}_u &= A_{ur} z_o + A_{rr} z_u \\
y &= z_1
\end{aligned} \tag{3.9}$$

where $z_o = [z_1,\ z_2,\ ..., z_r]^t$, $z_u = [z_{r+1},\ ...., z_n]$. Clearly, z_u denotes the unobservable states. Moreover, for stability, obviously A_{rr} must be a stable matrix. The dynamics given by

$$\dot{z}_u = A_{rr} z_u \tag{3.10}$$

is called "**zero dynamics**".

Fact. Let the transfer function of the linear system have the following form

$$G(s) = c(sI - A)^{-1}b = \frac{n(s)}{d(s)} \tag{3.11}$$

The zeros of the system, i.e. z such that $n(z) = 0$, are eigenvalues of the closed loop system matrix (3.10).

Proof. For a system with relative degree one, closed loop system matrix can be written by inserting the input term (3.3) into the equation (3.1). The resulting closed loop system matrix is found to be

$$A_c = (I - \frac{bc}{cb})A \tag{3.12}$$

If z is the zero of the system (3.1), it satisfies the following equations

$$Ax_o + bu_o = zx_o \tag{3.13}$$

$$cx_o = 0$$

When we multiply the equation (3.13) by the output matrix c from left, we have

$$cAx_o + cbu_o = 0 \tag{3.14}$$

the input term u_o is found from the equation (3.14) as $u_o = -\frac{cA}{cb}x_o$. When we put this input term into the equation (3.13)

$$A_c x_o = zx_o \tag{3.15}$$

where A_c is the closed loop state matrix given by (3.12). We can see from the equation (3.15) that the zeros of the system are the eigenvalues of the closed loop system matrix A_c. Suppose that the relative degree r is greater than one. In this case, when we put the computed input (3.5) into the system equation (3.1), we have

$$\dot{x} = (I - \frac{bcA^{r-1}}{cA^{r-1}b})Ax + b\frac{v}{cA^{r-1}b}$$

So closed loop matrix has the expression

$$A_c = (I - \frac{bcA^{r-1}}{cA^{r-1}b})A \tag{3.16}$$

When we multiply the equation (3.13) by the term cA^{r-1} from the left, we have

$$cA^r x_o + cA^{r-1}bu_o = cA^{r-1}zx_o \tag{3.17}$$

Note that right-hand side of the equation (3.17) is zero. In order to see this fact, we use the expression of Ax_o given by (3.13)

$$cA^{r-1}zx_o = zcA^{r-2}(Ax_o) = zcA^{r-2}(zx_o - bu_o) = z^2cA^{r-2}x_o \tag{3.18}$$

Note that, due to the definition of the relative degree, $cA^{r-2}bu_o = 0$. By this way, we can proceed as follows

$$cA^{r-1}zx_o = z^2cA^{r-2}x_o = \ldots = z^r cx_o = 0$$

Hence, we can rewrite the equation (3.17) as

$$cA^r x_o + cA^{r-1}bu_o = 0 \tag{3.19}$$

When we extract the input term u_o and put it into (3.13), we have

$$A_c x_o = zx_o \tag{3.20}$$

where A_c is given by (3.16). Hence, z is an eigenvalue of the closed loop system matrix. So, the proof is completed.□

The alternative way of seeing this fact is the following. Suppose that the system has a relative degree of r. In Laplace domain. we can write the relation between the reference input and the output of the system by using the equation (3.6) as

$$Y(s) = \frac{V(s)}{s^r} = G(s)U(s) = \frac{p(s)}{q(s)}U(s)$$

So, we have the following dynamics(transfer function) in the input generation process

$$\frac{U(s)}{V(s)} = \frac{q(s)}{p(s)s^r} \tag{3.21}$$

As a result of this fact, when the system given by (3.1) has zeros on the right-half complex plane (has unstable zeros), the zero dynamics of the closed loop system will be unstable. Because, unstable zeros of the open system turn out to be the eigenvalues of the zero dynamics. Consequently, inverse dynamics method can not be applied to the control of the non-minimum phase systems. This fact can be applied to the nonlinear systems also. In that case, definition of the poles and zeros of the system are modified in an appropriate way. More detailed analysis of this fact can be found in [12]

3.1 I/O Linearization Problem

Let the non-linear system be described by the following state and output equations

$$\dot{x} = f(x) + \sum_{i}^{m} g_i(x)u_i \text{ ,i=1..m} \tag{3.22}$$

$$y = h(x) \tag{3.23}$$

where $x \in R^n$ is the state vector; $y \in R^m$ is output vector; $f : R^n \to R^n$, $g_i : R^n \to R^n$, and $h : R^n \to R^m$ are vector fields.

The i/o linearization problem can be stated as follows:

I/O Linearization Problem: *Find a state control law of the form*

$$v = q(x) + S(x)u, \tag{3.24}$$

where $q(x) \in R^m$, and $S(x) \in R^{m\times m}$ smooth functions of x, *such that the new input-output relationship is in the following form*

$$\frac{d^{r_i}y_i}{dt^{r_i}} = v_i, \; i = 1..m, \tag{3.25}$$

which means that i^{th} channel is decoupled from other channels and i/o relation at each channel is linear.

This feedback scheme can be seen in Figure (3.1)

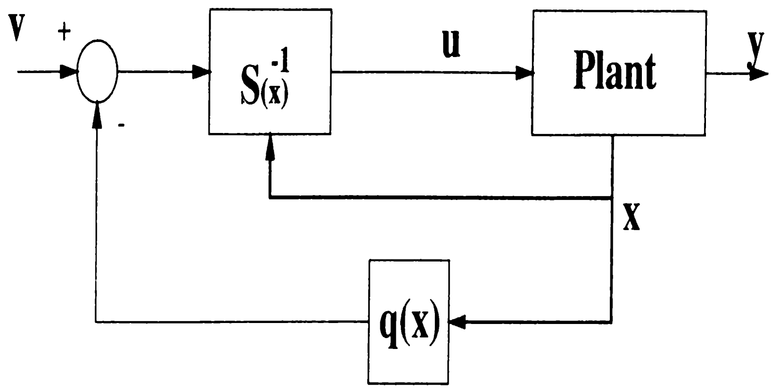

Figure 3.1: I/O linearization scheme

3.2 I/O Linearization of SISO Non-linear Systems

Let us consider the single-input, single-output(SISO) nonlinear system described by the following state equations,

$$\dot{x} = f(x) + g(x)u, \; y = h(x) \tag{3.26}$$

The following definition will be used in the subsequent derivations .

Lie Derivative: Let $h : R^n \rightarrow R$ and $f : R^n \rightarrow R^n$ be smooth functions. The map

$$x \longmapsto \frac{\partial h}{\partial x} \cdot f(x)$$

is called as the Lie Derivative of the function h with respect to the vector field f, and denoted by $L_f h$. The kth order Lie derivative is defined recursively as

$$L_f^k h = L_f(L_f^{k-1} h)$$

The time derivative of the output in the equation (3.26) is

$$\dot{y} = (L_f h)_{(x)} + u(L_g h)_{(x)} \tag{3.27}$$

Let us assume that there exists $\epsilon > 0$ such that

$$(L_g h)_{(x)} \neq 0 \quad for \quad x \in B_\epsilon(x_o) = \{x : \|x - x_o\| < \epsilon, x \in R^n\} \tag{3.28}$$

We define the functions $q(x)$ and $S(x)$ in the equation (3.24) as,

$$q(x) = (L_f h)_{(x)}, \; S(x) = (L_g h)_{(x)} \tag{3.29}$$

so that when we choose the input to the system as,

$$u = \frac{v - q(x)}{S(x)} \tag{3.30}$$

we obtain the following linear relation between the reference input (v) and the output of the system (3.26) in the ball defined in (3.28)

$$\dot{y} = v$$

Suppose that we can not find a ball $B_\epsilon(x_o)$ for any ϵ such that $(L_g h)_{(x)} \neq 0$ for all $x \in B_\epsilon$. Hence, the required input can not be found from the equation (3.30). Let us suppose that r^{th} derivative of the output is in the following form,

$$y^{(r)} = (L_f^r h)_{(x)} + u(L_g L_f^{r-1} h)_{(x)} \tag{3.31}$$

such that

$$(L_g L_f^k h)_{(x)} = 0, \quad k = 0, 1, .., r - 2 \tag{3.32}$$

$$(L_g L_f^{r-1} h)_{(x)} \neq 0 \tag{3.33}$$

for $x \in B_\epsilon$. Then the reference input can be defined as

$$v = (L_f^r h)_{(x)} + u(L_g L_f^{r-1} h)_{(x)} \tag{3.34}$$

When we extract the input term from the equation (3.34), we have

$$u = \frac{v - (L_f^r h)_{(x)}}{(L_g L_f^{r-1} h)_{(x)}} \tag{3.35}$$

When we apply this input to the nonlinear system, we have the following i/o relation

$$y^{(r)} = v \tag{3.36}$$

It can be shown that, the relative degree (r) is less than the dimension of the system, see[12]. Let us define the following new state variables

$$z_i = L_f^{i-1} h(x) \qquad i = 1, 2, ..., r$$
$$z_j = \varphi_j(x) \qquad j = r + 1, ..., n$$

such that

$$L_g\varphi_j(x) = 0 \quad j = r+1, ..., n$$

where $\varphi_j(x)$ are smooth functions of x and the set

$$\{dh(x),\ dL_f^1 h(x), dL_f^2 h(x),\ ..., dL_f^{r-1}h(x), d\varphi_{r+1}, d\varphi_{r+2}, ..., d\varphi_n\} \tag{3.37}$$

at $x = x_o$ is independent. The state equations in the new variables with the control law (3.35) become

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ & \\ \dot{z}_r &= v \\ \dot{z}_u &= \Phi(z_o, z_u) = \Phi_1(x) \\ y &= z_1 \end{aligned} \tag{3.38}$$

where $\Phi_1(x) = [L_f\varphi_{r+1},\ L_f\varphi_{r+2}, ..., L_f\varphi_n]^T$, $z_o = [z_1,\ z_2,\ ..., z_r]^T$, $z_u = [z_{r+1},\ ...., z_n]^T$. Clearly, z_u denotes the unobservable states. The dynamics given by

$$\dot{z}_u = \Phi(0, z_u) \tag{3.39}$$

is called "**zero dynamics**" of the SISO nonlinear system (3.26), and should be stable for application of the NID law in stabilization of the nonlinear system. The following example illustrates application of the NID law to i/o linearize a nonlinear system.

Example. Let us take the SISO non-linear system described by equations (3.26) with,

$$f(x) = \begin{bmatrix} x_2 \\ x_1 + x_1x_2 \\ 2x_1 + x_2 \end{bmatrix},\ g(x) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix},\ h(x) = x_3 \tag{3.40}$$

we have the following Lie derivatives

$$(L_fh)_{(x)} = 2x_1 + x_2,\ (L_gh)_{(x)} = 0$$

$$(L_f^2h)_{(x)} = 2x_2 + x_1 + x_1x_2\ ,\ (L_gL_fh)_{(x)} = 1$$

So the relative degree (r) of this system is 2. When we choose the following new state variables

$$z_1 = h(x) = x_3$$

$$z_2 = (L_f h)(x) = -2x_1 + x_2$$

$$z_3 = \varphi_3(x) = x_1 + x_2$$

and the input as

$$u = v + 2x_1 - x_1 - x_1.x_2$$

we have the following new state equations

$$\dot{z}_1 = z_2$$

$$\dot{z}_2 = v$$

$$\dot{z}_3 = -2z_1 + 2z_2 + 2z_3$$

$$y = z_1$$

Note that, the zero dynamics of this closed loop system described by the following equation is unstable.

$$\dot{z}_3 = 2z_3 \tag{3.41}$$

When we linearize the system about the equilibrium point $(x_1, x_2, x_3) = (0, 0, 0)$, we have

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -2 & 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x$$

where $x = [x_1\ x_2\ x_3]^T$. The transfer function of the linearized system is

$$G(s) = \frac{s-2}{s(s^2-1)}$$

Note that the zero of the open loop system is unstable, which becomes the eigenvalue of the zero dynamics (3.41). Hence, we can not use the NID law to control this system.

Note that, when the linearized version of the system about an equilibrium point has an unstable zero, NID law can not be applied to the nonlinear system to stabilize it about that equilibrium point. However, this condition gives only the necessary but not sufficient condition for the application of the NID law. In other words, showing that linearized system does not have any unstable zero is not sufficient to use NID law.

3.3 Multi Input Multi Output (MIMO)Case

The same method (used in SISO case) with minor modifications can be used in MIMO case. First the definition of the relative degree is slightly different. It is not a scalar but a vector of the form

$$r = \begin{bmatrix} r_1 & r_2 & .... & r_m \end{bmatrix}^t \tag{3.42}$$

such that

$$(L_{g_j} L_f^k h_i)_{(x)} = 0, \ k = 0, 1, ..., (r_i - 2) \tag{3.43}$$

and the matrix $S(x)$ of the form

$$S = \begin{bmatrix} (L_{g_1} L_f^{r_1-1} h_1)_{(x)} & (L_{g_2} L_f^{r_1-1} h_1)_{(x)} & (L_{g_m} L_f^{r_1-1} h_1)_{(x)} \\ (L_{g_1} L_f^{r_2-1} h_2)_{(x)} & (L_{g_2} L_f^{r_2-1} h_2)_{(x)} & (L_{g_m} L_f^{r_2-1} h_2)_{(x)} \\ \vdots & \vdots & \vdots \\ (L_{g_1} L_f^{r_m-1} h_m)_{(x)} & (L_{g_2} L_f^{r_m-1} h_m)_{(x)} & (L_{g_m} L_f^{r_m-1} h_m)_{(x)} \end{bmatrix} \tag{3.44}$$

which is non singular in the ball B_ϵ about an operating point x_o. By defining q as a vector with elements,

$$q_i = L_f^{r_i} h_i, \tag{3.45}$$

the resulting i/o relation of the system (3.22) with the new reference input defined in (3.24) becomes in the form (3.25). The proof of the followin theorem can be found in [12].

Theorem.1. Suppose the system (3.22), (3.23) has relative degree vector r as in (3.42) with $rd = \sum_{i=1}^{m}(r_i)$. Then there exists a local diffeomorphism T around x_o, such that, in terms of the new state vectors $z = T(x)$, and with the reference input v , the system dynamics is described by

$$\frac{d}{dt}\left(\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \\ z_u \end{bmatrix}\right) = \begin{bmatrix} A_1 & 0 & & 0 \\ 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & & A_m \\ 0 & 0 & & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} (f_u(z) + \sum_{j=1}^{m} g_{uj}(z) v_j) \tag{3.46}$$

$$y_i = z_{i,1}, \ i = 1, 2, .., m$$

where $z_i = \begin{bmatrix} z_{i,1} & z_{i,2} & \cdots & z_{i,r_i} \end{bmatrix}^T \in R^{r_i}$, $b_i = \begin{bmatrix} 0 & 0 & & 1 \end{bmatrix}^T \in R^{r_i}$ for $i = 1, 2, .., m$, $z_u \in R^{n-rd}$, $v = \begin{bmatrix} v_1 & v_2 & \cdots & v_m \end{bmatrix}^T \in R^m$ is the reference input vector, and A_i is a $r_i \times r_i$ matrix with the following form(companion form).

$$A_i = \begin{bmatrix} 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & & 0 \end{bmatrix} \tag{3.47}$$

The transformed system equations (3.46) can be partitioned as,

$$\dot{z}_i = A_i z_i + b_i v_i, \ y_i = c_i z_i, \ i = 1, 2, ..., m \tag{3.48}$$

and

$$\dot{z}_u = f_u(z) + \sum_{j=1}^{m} g_{uj}(z) v_j \tag{3.49}$$

Note that states z_u are not observable and does not have any effect on other states.

Definition.(Zero Dynamics for MIMO nonlinear system)

Let the transformed state vector z be partitioned as $z = [z_o \ z_u]^T$, then the following dynamic equation is called the zero dynamics of the system (3.46),

$$\dot{z}_u = f_u(0, z_u) \tag{3.50}$$

Clearly, z_o denotes the observable states, and z_u denotes the unobservable states of the system. The observable states are evolved by the following dynamic equation

$$\dot{z}_o = A_o z_o + B_o v \tag{3.51}$$

where

$$A_o = \begin{bmatrix} A_1 & 0 & & 0 \\ 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & & A_m \end{bmatrix} \text{ and } B_o = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

If the zero dynamics of the system is stable, the observable part of the system can be stabilized by using certain linear controller design techniques. For example, we can use the following state feedback

$$v = -Kz_o$$

to stabilize the system. When we apply this outer-loop control, the closed loop system matrix will be $A_s = A_o - B_oK$. Since the system (3.51) is controllable, we can place the poles of the closed loop system at any place in the complex plane. This stabilization scheme is depicted in Figure (3.2)

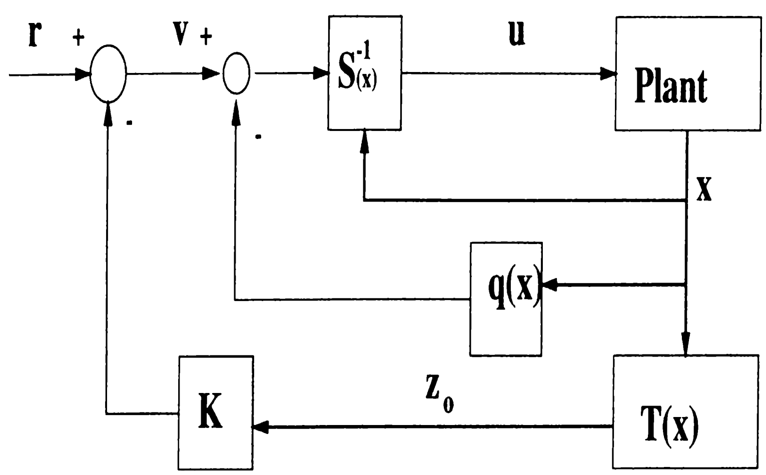

Figure 3.2: State feedback with inverse dynamics

Second part called as autopilot unit operates on fast dynamics and computes the required canard deflection to achieve the commanded acceleration computed in the guidance unit. This part is designed by using conventional controller design technique. The overall controller structure is shown in the Figure (4.1)

4.1.1 The Guidance Design by Using Inverse Dynamics Method

In this subsection, the guidance algorithm is derived by using the nonlinear inverse dynamics method . The problem is to design a guidance unit computing the required acceleration so that when the missile tracks this computed acceleration, it also tracks a given trajectory .

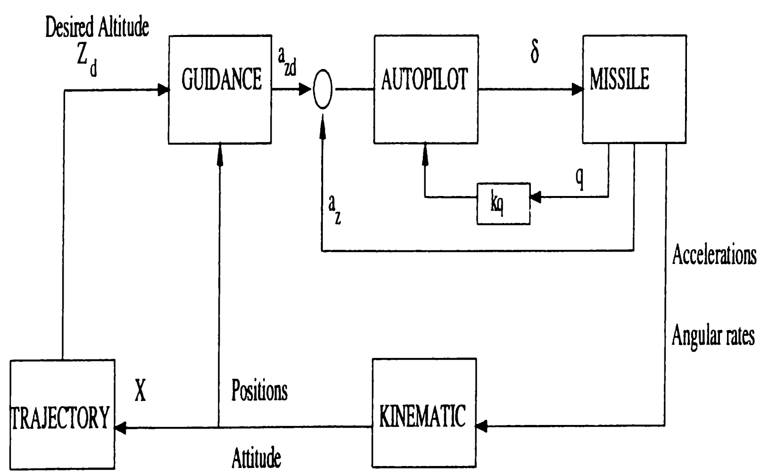

Figure 4.1: Guidance and Autopilot

The derivation of the guidance law for the pitch plane is presented first. After that, the full formulation for guidance law is presented, in which case, all the equations of motion of the missile are taken into account without any assumption.

The Guidance Law for Pitch Plane.

The motion of the missile in the pitch plane is described by the equations (2.56), (2.57), (2.58), (2.59), and (2.61). In the sequel, pitch and yaw motions are assumed to be decoupled. If the motion in yaw plane is small and there is no rolling motion, this assumption may be valid.

Let the desired trajectory be given in the following form,

$$Z_d = f(X) \tag{4.1}$$

where f is continuous, double differentiable function of X , and Z_d is the desired altitude of the missile at X. The position error is defined as,

$$e = Z - Z_d \tag{4.2}$$

The first derivative of the error with respect to time is,

$$\dot{e} = -\frac{\partial Z_d}{\partial X}\dot{X} + \dot{Z}$$

or

$$\dot{e} = \begin{bmatrix} -\frac{\partial f}{\partial X} & 1 \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Z} \end{bmatrix} \tag{4.3}$$

when the equation (2.59) are inserted in (4.3), we obtain

$$\dot{e} = GT_m \begin{bmatrix} U \\ W \end{bmatrix}$$

where G is defined as

$$G = \begin{bmatrix} -\frac{\partial f}{\partial X} & 1 \end{bmatrix}$$

and T_m is the transformation matrix given by (2.60). The second time derivative of error is,

$$\ddot{e} = \dot{G}T_m \begin{bmatrix} U \\ W \end{bmatrix} + G\dot{T}_m \begin{bmatrix} U \\ W \end{bmatrix} + GT_m \begin{bmatrix} \dot{U} \\ \dot{W} \end{bmatrix} \tag{4.4}$$

Since T_m is transformation matrix, its derivative with respect to time can be written as,

$$\dot{T}_m = T_m w \tag{4.5}$$

where w is skew-symmetric matrix with the following form,

$$w = \begin{bmatrix} 0 & q \\ -q & 0 \end{bmatrix}$$

Also noting that,

$$\begin{bmatrix} \dot{U} \\ \dot{W} \end{bmatrix} = -w \begin{bmatrix} U \\ W \end{bmatrix} + \begin{bmatrix} F_x/m \\ F_z/m \end{bmatrix}$$

so the equation (4.4) can be rewritten as,

$$\ddot{e} = GT_m \begin{bmatrix} F_x/m + g_x \\ F_z/m + g_y \end{bmatrix} - \frac{\partial^2 f}{\partial X^2}(\dot{X})^2 \tag{4.6}$$

Let r be reference input. When we equate second time derivative of the error (4.2) to this reference input,

$$\ddot{e} = r \tag{4.7}$$

then the aerodynamic acceleration (a_z) in Z direction (F_z/m) is found to be

$$F_z/m = \frac{[(r-g) + \frac{\partial^2 f}{\partial X^2}(\dot{X})^2] + (\frac{\partial f}{\partial X}\cos(\Theta) + \sin(\Theta))a_x}{(\cos(\Theta) - \frac{\partial f}{\partial X}\sin(\Theta))} \tag{4.8}$$

If the reference input is chosen as,

$$r = -k_d\dot{e} - k_p e \tag{4.9}$$

then the following second order linear differential equation describing the error dynamic can be found by using (4.7) and (4.9),

$$\ddot{e} + k_d\dot{e} + k_p e = 0 \tag{4.10}$$

where the parameters k_d and k_p are real positive numbers and found with respect to the desired error dynamics. By inserting the equation (4.9) into the equation (4.8), desired acceleration is found to be

$$a_{zd} = \frac{[(-k_d\dot{e} - k_p e - g) + \frac{\partial^2 f}{\partial X^2}(\dot{X})^2] + (\frac{\partial f}{\partial X}\cos(\Theta) + \sin(\Theta))a_x}{(\cos(\Theta) - \frac{\partial f}{\partial X}\sin(\Theta))} \tag{4.11}$$

This acceleration is given to the autopilot unit as acceleration command.

In the equation (4.11), singularity problem arises when denominator equals to zero. If the denominator is equated to zero, we have

$$\frac{\partial f}{\partial X} = \cot(\Theta) \tag{4.12}$$

This situation occurs when the missile is perpendicular to the trajectory. As long as the given trajectory is smooth and controller is robust enough, possibility of this situation is very small.

The Guidance Law for 6-DOF Motion of The Missile

In this subsection, we try to develop a guidance algorithm for tracking the trajectory in three dimensional space. Hence, there will be two outputs to be tracked and two control inputs. It should be noted that there will be no assumptions in the derivation of the guidance law, which is not the case in the previous subsection. The output vector to be controlled is defined as

$$\mathbf{y} = \begin{bmatrix} Y \\ Z \end{bmatrix} \tag{4.13}$$

where Y, Z are the position components of the missile in the inertial frame. When time-derivative of output is taken, we have

$$\dot{\mathbf{y}} = \begin{bmatrix} \dot{Y} \\ \dot{Z} \end{bmatrix} = GT \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{4.14}$$

where T is the transformation matrix given in (2.36) and G is defined as

$$G = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

when double time derivative of the output is taken, we have

$$\ddot{\mathbf{y}} = G\dot{T} \begin{bmatrix} U \\ V \\ W \end{bmatrix} + GT \begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} \tag{4.16}$$

When the equations (2.41), and (2.19) are inserted in place of $\dot{T}$ and $\begin{bmatrix} \dot{U} & \dot{V} & \dot{W} \end{bmatrix}^T$ respectively in the equation (4.16), we obtain

$$\ddot{\mathbf{y}} = GT \left\{ \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} \right\} \tag{4.17}$$

When we use the following equality in the equation (4.17)

$$T \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{4.18}$$

we have

$$\ddot{\mathbf{y}} = GT \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} 0 \\ g \end{bmatrix} \tag{4.19}$$

Let us denote ijth element of the transformation matrix as T_{ij} Then, from the equation (4.19), we obtain

$$\ddot{\mathbf{y}} = \begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix} \begin{bmatrix} a_y \\ a_z \end{bmatrix} + \begin{bmatrix} T_{21} a_x \\ T_{31} a_x + g \end{bmatrix} \tag{4.20}$$

Let $\mathbf{v} = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^T$ be a reference input vector. When we equate the double time derivative of the output y to the reference input vector v , we obtain

$$\ddot{\mathbf{y}} = \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{4.21}$$

We can extract the required inputs $(a_y,\ a_z)$ from the equation (4.20) to linearize the system from the reference inputs (v_1, v_2) to the outputs $(Y,\ Z)$. The required control inputs are

$$\begin{bmatrix} a_y \\ a_z \end{bmatrix} = \begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} - \begin{bmatrix} T_{21} a_x \\ T_{31} a_x + g \end{bmatrix} \right\} \tag{4.22}$$

under the condition that the following matrix is non singular.

$$T_{e1} = \begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix} \tag{4.23}$$

If we take the determinant of the matrix T_{e1}, we have,

$$\det(T_{e1}) = c\Theta s\Psi \tag{4.24}$$

Hence, singularity of the matrix T_{e1} occurs when one of the following conditions is satisfied;

$$\Psi = 0^\circ \quad or \quad 180^\circ$$
$$\Theta = 90^\circ \quad or \quad 270^\circ$$

Let the desired trajectory be given in the following form,

$$\mathbf{y_d} = \begin{bmatrix} Y_d(X) \\ Z_d(X) \end{bmatrix} \tag{4.25}$$

We define the position error vector as

$$\mathbf{e} = \begin{bmatrix} Y - Y_d \\ Z - Z_d \end{bmatrix} \tag{4.26}$$

Let the reference input vector $\mathbf{v} = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^T$ be defined by

$$\mathbf{v} = \mathbf{\ddot{y}_d} + \mathbf{K_1}\mathbf{\dot{e}} + \mathbf{K_2}\mathbf{e} \tag{4.27}$$

where $\mathbf{K_1}$, and $\mathbf{K_2}$ are 2×2 positive definite matrices. In particular, they can be chosen as diagonal matrices with positive elements. If we insert the equation (4.27) into the equation (4.21), the following equation determining the error dynamics is obtained

$$\mathbf{\ddot{e}} + \mathbf{K_1}\mathbf{\dot{e}} + \mathbf{K_2}\mathbf{e} = 0 \tag{4.28}$$

Since the trajectory is a function of the range (X), the time derivative of the desired output can be written as

$$\mathbf{\dot{y}_d} = \frac{\partial \mathbf{y_d}}{\partial X}.\dot{X} \tag{4.29}$$

and double time derivative of the desired output will be

$$\mathbf{\ddot{y}_d} = \frac{\partial^2 \mathbf{y_d}}{\partial X^2}.(\dot{X})^2 + \frac{\partial \mathbf{y_d}}{\partial X}.\ddot{X} \tag{4.30}$$

Note that $\ddot{X}$ depends on inputs a_y, a_z in the following way

$$\ddot{X} = \begin{bmatrix} T_{12} & T_{13} \end{bmatrix} \cdot \begin{bmatrix} a_y \\ a_z \end{bmatrix} + T_{11}.a_x \tag{4.31}$$

where T_{ij} are the transformation matrix elements. When the equation (4.30) with the equation (4.31) is inserted in the equation (4.27), we have

$$\mathbf{v} = \frac{\partial^2 \mathbf{y_d}}{\partial X^2}.(\dot{X})^2 + \frac{\partial \mathbf{y_d}}{\partial X}\left\{\begin{bmatrix} T_{12} & T_{13} \end{bmatrix} \cdot \begin{bmatrix} a_y \\ a_z \end{bmatrix} + T_{11}.a_x\right\} + \mathbf{K_1}\dot{\mathbf{e}} + \mathbf{K_2}\mathbf{e} \quad (4.32)$$

By using the equations (4.32), (4.21), and (4.20), we can write

$$\begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix}\begin{bmatrix} a_y \\ a_z \end{bmatrix} = \frac{\partial \mathbf{y_d}}{\partial X}\begin{bmatrix} T_{12} & T_{13} \end{bmatrix} \cdot \begin{bmatrix} a_y \\ a_z \end{bmatrix} + P_1 \quad (4.33)$$

where

$$P_1 = \frac{\partial^2 \mathbf{y_d}}{\partial X^2}.(\dot{X})^2 + \mathbf{K_1}\dot{\mathbf{e}} + \mathbf{K_2}\mathbf{e} - \begin{bmatrix} T_{21}a_x \\ T_{31}a_x + g \end{bmatrix} + \frac{\partial \mathbf{y_d}}{\partial X}T_{11}.a_x$$

If we take the input terms to the left hand side of the equation (4.33), we have

$$\left\{\begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix} - \frac{\partial \mathbf{y_d}}{\partial X}\begin{bmatrix} T_{12} & T_{13} \end{bmatrix}\right\}\begin{bmatrix} a_y \\ a_z \end{bmatrix} = P_1 \quad (4.34)$$

so the required accelerations to have an error dynamics described by (4.28) are found from (4.34) as

$$\begin{bmatrix} a_y \\ a_z \end{bmatrix} = T_{e2}^{-1}P_1 \quad (4.35)$$

where the matrix T_{e2} has the following form

$$T_{e2} = \begin{bmatrix} T_{22} & T_{23} \\ T_{32} & T_{33} \end{bmatrix} - \frac{\partial \mathbf{y_d}}{\partial X}\begin{bmatrix} T_{12} & T_{13} \end{bmatrix} \quad (4.36)$$

It should be noted that the matrix T_{e2} should be non singular to apply the equation (4.35) for finding the required accelerations. The singularity condition of the matrix T_{e2} may be derived, but does not give a clear physical interpretation.

4.2 Acceleration Autopilot Design

The conventional method is used to design the autopilot. The conventional methods use the linear controller design methods for the linearized version of the nonlinear system about certain points in state space. In general, this point (x_e) is chosen such that the following equality holds,

$$\dot{x}|_{x=x_e} = 0 \tag{4.37}$$

This point is called as the equilibrium point or trim point. In the vicinity of the trim point, the linearized version of nonlinear system reflects the properties of nonlinear system.. Hence, as long as the system is close to the trim point, the linear controller designed to stabilize the linearized system also stabilize the nonlinear system.

In our problem, system dynamics is parameter varying also, so trim points is changing with time due to the changing parameters. Hence, designed controller gains should be changed accordingly.Namely, for a set of changing parameter values, the controller gains are found indexed with these changing parameters. This method of adapting the controller gains due to the parameter variation is known as "gain scheduling technique" in literature. The Mach number, which is the main variable causing the system parameter variations, will be used to update these controller gains.

Linearization of Fast Dynamics by Using Conventional Method The fast dynamics consist of the states such as pitch angular rate (q) and the velocity component in Z_b direction. First, since the aerodynamic coefficients are given in terms of the flight parameters; angle of attack, pitch rate, canard deflection and Mach number, a kind of state transformation under certain approximations is made to obtain the angle of attack as a state of the system.

When the angle of attack is assumed to be small and the motion of the missile is restricted to the pitch plane, the time derivative of the angle of attack can be written as,

$$\dot{\alpha} = \frac{\dot{W}}{U} - \frac{W\dot{U}}{U^2} \tag{4.38}$$

In general, second term in above equation is very small compared with the first one, so the second term can be ignored and 4.38 can be approximated as,

$$\dot{\alpha} = \frac{\dot{W}}{U} \tag{4.39}$$

If the equation (2.57) is inserted in the equation (4.39), we have

$$\dot{\alpha} = \frac{F_z}{Um} + q + \frac{g_z}{U} \tag{4.40}$$

The acceleration term ($\frac{\dot{a}_z}{U}$) is very small compared with other terms in the above equation. Hence, it will be ignored in the following derivations.

The other state equation to be cosiderd as a part of fast dynamics is

$$\dot{q} = M/I_y \tag{4.41}$$

When the aerodynamic force F_z and torque M expressions are inserted in equations (4.40) and (4.41), we have,

$$\dot{\alpha} = \frac{Q_d A C_z(M, \alpha, q, \delta)}{Um} + q \tag{4.42}$$

$$\dot{q} = \frac{Q_d A d C_m(M, \alpha, q, \delta)}{I_y} \tag{4.43}$$

the equations (4.42) and (4.43) can be linearized after finding the equilibrium point by equating the time rates of change to zero. This calculations show that $(\alpha_e, q_e, \delta_e)$=(0, 0, 0) an be taken as equilibrium point for small angle of attack and small canard deflection. The resulting linearized equations are

$$\dot{\alpha} = \frac{Z_\alpha}{U}\alpha + (1 + \frac{Z_q}{U})q + \frac{Z_\delta}{U}\delta \tag{4.44}$$

$$\dot{q} = J_\alpha \alpha + J_q q + J_\delta \delta \tag{4.45}$$

where

$$Z_\alpha = \frac{Q_d A}{m}\frac{\partial C_z(M, q, \alpha, \delta)}{\partial \alpha} \tag{4.46}$$

$$Z_q = \frac{Q_d A}{m}\frac{\partial C_z(M, q, \alpha, \delta)}{\partial q} \tag{4.47}$$

$$Z_\delta = \frac{Q_d A}{m}\frac{\partial C_z(M, q, \alpha, \delta)}{\partial \delta} \tag{4.48}$$

$$J_\alpha = \frac{Q_d A d}{I_y}\frac{\partial C_m(M, \alpha, q, \delta)}{\partial \alpha} \tag{4.49}$$

$$J_q = \frac{Q_d A d}{I_y}\frac{\partial C_m(M, \alpha, q, \delta)}{\partial q} \tag{4.50}$$

$$J_\delta = \frac{Q_d A d}{I_y}\frac{\partial C_m(M, \alpha, q, \delta)}{\partial \delta} \tag{4.51}$$

All these linearized coefficients are found as a function of Mach number at the trim point $\alpha = 0$, $\delta = 0$, $q = 0$ by using special software programs such as missile-dot-com, or by using wind-tunnel tests.

The output of the fast dynamics is taken as the acceleration in Z_b axis,

$$a_{zb} = \frac{F_z}{m} = Z_\alpha \alpha + Z_q q + Z_\delta \delta \tag{4.52}$$

The Design of Autopilot The aim is to design a linear controller (**autopilot**) so that the missile tracks the commanded acceleration in an acceptable way. The linearized fast dynamics of the missile has the following state space form,

$$\dot{x} = Ax + b\delta \tag{4.53}$$

$$a_{zb} = cx + d\delta \tag{4.54}$$

where $x = \begin{bmatrix} \alpha & q \end{bmatrix}^T$, $A = \begin{bmatrix} \frac{Z_\alpha}{U} & 1 + Z_q/U \\ J_\alpha & J_q \end{bmatrix}$, $b = \begin{bmatrix} \frac{Z_\delta}{U} & J_\delta \end{bmatrix}^T$, $c = \begin{bmatrix} Z_\alpha & J_q \end{bmatrix}$, and $d = Z_\delta$. The controller configuration used in the autopilot design is shown in the Figure (4.2).

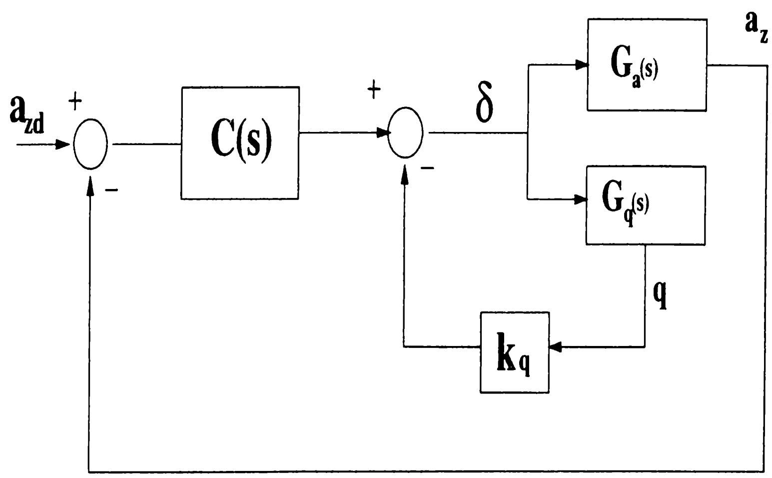

Figure 4.2: Autopilot Configuration

In this figure, the transfer functions G_a, and G_q are of the following form

$$G_a(s) = \frac{a_{zb}(s)}{\delta(s)} = c(sI - A)^{-1}b + d \tag{4.55}$$

and

$$G_a(s) = \frac{q(s)}{\delta(s)} = \begin{bmatrix} 0 & 1 \end{bmatrix} (sI - A)^{-1}b + d \tag{4.56}$$

where s is the complex variable, $a_{zb}(s)$, $q(s)$ and $\delta(s)$ are Laplace transformation of related variables. When the equations (4.55), and (4.56) are expanded, transfer functions are found to be

$$G_a(s) = \frac{\lambda_1 s^2 + \lambda_2 s + \lambda_3}{s^2 + \kappa_1 s + \kappa_2} \tag{4.57}$$

and

$$G_q(s) = \frac{n_1 s + n_2}{s^2 + \kappa_1 s + \kappa_2} \tag{4.58}$$

where λ_i, κ_i, and n_i are

$$\lambda_1 = Z_\delta$$

$$\lambda_2 = \frac{M_\delta Z_\alpha - M_q Z_\delta}{U}$$

$$\lambda_3 = Z_\alpha M_\delta - M_\alpha Z_\delta$$

$$\kappa_1 = -\frac{(Z_\alpha + M_q U)}{U}$$

$$\kappa_2 = \frac{(Z_\alpha J_q - J_\alpha U - J_\alpha Z_q)}{U}$$

$$n_1 = M_\delta$$

$$n_2 = \frac{M_\alpha Z_\delta - Z_\alpha M_\delta}{U}$$

The closed loop transfer function of the controlled system shown in the figure (4.2) is found to be

$$G(s) = \frac{a_{zb}(s)}{R(s)} = \frac{C(s)G_a(s)}{1 + C(s)G_a(s) + G_q(s)k_q}$$

where $C(s)$ is PI(proportional-integral) controller with the following form,

$$C(s) = k_p + \frac{k_i}{s}$$

and k_q, k_i, and k_p are controller gains.

The controller gains are found by using the pole placement method. This method is straightforward, which requires only the solution of three algebraic equations with three unknowns. The location of the poles are chosen to satisfy the dynamic response requirements of the autopilot. These requirements are determined with respect to the limit of the control capability of the missile's canards and other factors related with mission requirements.

Since the parameters of the system depends on the Mach number which is time-varying, controller gains are found for different values of Mach number over a specified range and tabulated. For a calculated value of the Mach number, corresponding controller gains are found by linear interpolation method. The fact that Mach number does not change rapidly make stability of the system over operating range possible.

4.3 Second Method

In this method, required canard deflection to make the position error between desired trajectory and actual trajectory zero with an acceptable dynamic is calculated by using the NID law directly. Hence, separating the controller structure into two part is not considered in this approach. Derivation of this controller law follows the same steps used in the derivation of the guidance law. In the equation (4.8), we insert the expression of F_z given by the equation (4.52)

$$\frac{F_z}{m} = Z_\alpha \alpha + Z_q q + Z_\delta \delta = \frac{[(r-g) + \frac{\partial^2 f}{\partial X^2}(\dot{X})^2] + (\frac{\partial f}{\partial X}\cos(\Theta) + \sin(\Theta))a_x}{(\cos(\Theta) - \frac{\partial f}{\partial X}\sin(\Theta))} \tag{4.59}$$

where the definitions of the coefficients Z_α, Z_q, and Z_δ are given by (4.46), (4.47), and (4.48) respectively.

The required canard deflection to linearize the nonlinear system from reference input to the output is found by using the equation (4.59). Its expression is

$$\delta = \frac{[(r-g) + \frac{\partial^2 f}{\partial X^2}(\dot{x})^2] + (\frac{\partial f}{\partial X}\cos(\Theta) + \sin(\Theta))F_x}{(\cos(\Theta) - \frac{\partial f}{\partial X}\sin(\Theta))Q_d A C_{z\delta}} - \frac{C_{z\alpha}}{C_{z\delta}}\alpha - \frac{C_{zq}}{C_{z\delta}}q \tag{4.60}$$

When the reference input is chosen as,

$$r = -k_d \dot{e} - k_p e \tag{4.61}$$

then the required canard deflection for the missile to track the given trajectory with the error dynamic determined by the coefficients k_d ,and k_p takes the following form,

$$\delta = \frac{[(-k_d\dot{e} - k_p e - g) + \frac{\partial^2 f}{\partial X^2}(\dot{X})^2] + (\frac{\partial f}{\partial X}\cos(\Theta) + \sin(\Theta))F_x}{(\cos(\Theta) - \frac{\partial f}{\partial X}\sin(\Theta))Q_d A C_{z\delta}} - \frac{C_{z\alpha}}{C_{z\delta}}\alpha - \frac{C_{zq}}{C_{z\delta}}q \tag{4.62}$$

Let us cosider the following dynamic equations which are used in the derivation of the equation 4.62.

$$\dot{Z} = -s\Theta U + c\Theta W \tag{4.63}$$

$$\dot{W} = qU + Z_\alpha \frac{W}{U} + Z_\delta \delta \tag{4.64}$$

$$\dot{\Theta} = q \tag{4.65}$$

$$\dot{q} = J_\alpha \frac{W}{U} + J_q q + J_\delta \delta \tag{4.66}$$

It is assumed that U is slowly varying (its derivative is taken as zero) and the angle of attack is small in (4.64), and (4.66). The nonlinear state equations can be written in a compact form as,

$$\dot{x} = f(x) + bu, \quad y = cx \tag{4.67}$$

where $x = \begin{bmatrix} Z & W & \Theta & q \end{bmatrix}$, $b = \begin{bmatrix} 0 & Z_\delta & 0 & J_\delta \end{bmatrix}^T$, $c = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$,and f is the nonlinear vector valued function of the following form

$$f(x) = \begin{bmatrix} -s\Theta U + c\Theta W \\ qU + Z_\alpha \frac{W}{U} + Z_q q \\ q \\ J_\alpha \frac{W}{U} + J_q q \end{bmatrix} \tag{4.68}$$

the trim point of the system can be found by solving the following algebraic equation

$$f(x) = 0$$

when we solve above equation, trim point is found to be

$$x_e = \begin{bmatrix} Z_e \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.69}$$

When we linearize the system (4.67) about this trim point, the following linear system is obtained

$$\dot{x}' = Ax' + bu \tag{4.70}$$

where $x' = x - x_e$ and $A = \frac{\partial f}{\partial x}\rfloor_{x=x_e}$ is

$$A = \begin{bmatrix} 0 & c\Theta & -(Uc\Theta + Ws\Theta) & 0 \\ 0 & \frac{Z_\alpha}{U} & 0 & Z_q + U \\ 0 & 0 & 0 & 1 \\ 0 & \frac{J_\alpha}{U} & 0 & J_q \end{bmatrix}_{x=x_e} \tag{4.71}$$

After inserting the trim values of the states into the equation (4.71), we have the system matrix

$$A = \begin{bmatrix} 0 & 1 & -U & 0 \\ 0 & \frac{Z_\alpha}{U} & 0 & Z_q + U \\ 0 & 0 & 0 & 1 \\ 0 & \frac{J_\alpha}{U} & 0 & J_q \end{bmatrix} \tag{4.72}$$

The transfer function of the linear system (4.70) is

$$G(s) = c(sI - A)^{-1}b \tag{4.73}$$

When we expand the equation (4.73), the numerator of the resulting expression is found to be

$$p(s) = s^2 - \frac{(Z_\delta J_q + J_\delta Zq)}{Z_\delta} J_q s + \frac{(J_\delta Z_\alpha - Z_\delta J_\alpha)}{Z_\delta} \tag{4.74}$$

In order to apply the dynamic inversion to control the system (4.70), the zeros of the linearized system should be on the left-half complex plane. This requires that the coefficients of the equation (4.74) should be positive,

$$\begin{array}{c} \frac{(Z_\delta J_q + J_\delta Zq)}{Z_\delta} J_q < 0 \\ \frac{(J_\delta Z_\alpha - Z_\delta J_\alpha)}{Z_\delta} > 0 \end{array} \tag{4.75}$$

This analysis gives us only partial conditions required for controllability of the i/o linearized system by using inverse dynamics method. Because, we have only considered the linearized system.

Chapter 5

SIMULATIONS

In this chapter, the simulation programs will be explained, and the simulation results are presented and interpreted.

5.1 The Simulation Program

The simulation programs are written in Matrix-X environment. The nonlinear and parameter varying equations decribing the motion of the missile are used in these programs. In MatrixX environment, the blocks are the basic composing units for building the complex systems. These blocks can be used to implement various types of dynamics.

The dynamic system equations may be solved by using different integration algorithms. The best integration algorithm is dependent on the system structure. In our model, there exist slow and fast dynamics. Hence, Variable-Step Kutta-Merson method is used as the integration algorithm.This method combines the accuracy of the Fixed-step Kutta-Merson with a variable-step implementation for improved speed.

The initial conditions for the states of the system are chosen according to certain considerations. When the missile is assumed to be not moving before firing, the initial velocities are taken as zero. The initial attitude of the missile is found by using an initial alignment process. This process uses the information coming from the sensors to find the initial Euler angles. The initial position of the missile with respect to the inertial frame is found by using certain techniques used in the cartography. In our simulations, the initial position is chosen as the origin of the inertial frame.

The inertia terms, the mass, and the center of gravity(cg.) location of the missile are time-varying for the initial few second of the flight.These few seconds is called as the boost phase of the flight. In this phase, the thrust created by the burning of the propellant gives the energy of the missile. The mass, inertia, and cg location of the missile changes from the initial values(m_0, I_0, cg_0) to the final values (m_1, I_f, cg_f) during the consumption of the propellant. This variance is modeled by the following mathematical equation,

$$m(t) = m_0 - \frac{m_0 - m_1}{I_{tot}} \int_0^t I_t(t)dt \tag{5.1}$$

$$I(t) = I_0 - \frac{I_0 - I_1}{I_{tot}} \int_0^t I_t(t)dt \tag{5.2}$$

$$cg(t) = cg_0 - \frac{cg_0 - cg_1}{I_{tot}} \int_0^t I_t(t)dt \tag{5.3}$$

where t_f is the end time of the boost phase, $I_t(t)$ is the impulse created by thrust at time t, I_{tot} is the total impulse.

The controller is opened at the end of the boost phase. This means that the aerodynamic control is used to guide the missile. The four canards located on the missile surface are used to change the aerodynamic forces and moments applied on the missile. The maximum allowable canard deflection is taken as 10^o. This limitation is used to keep the missile in the vicinity of the trim point which is used in the autopilot design.

For six-degree-of-freedom (6-DOF) simulations, the rotation motion of the missile is not controlled. The commanded canard deflections computed by controller is for non-rotated missile, which is equal to the $\Phi = 0^o$. Hence, a kind of input transformation is used when the missile rotates. This transformation corresponds to a matrix multiplication which resolves the commanded canard deflections into the canards This matrix has the following form,

$$T_c = \begin{bmatrix} \cos(\Phi) & \sin(\Phi) \\ -\sin(\Phi) & \cos(\Phi) \end{bmatrix} \tag{5.4}$$

The controller gains are found for the different mach numbers and they are interpolated for a computed mach number. A program is written in matrixX to calculate the controller gains for the specified set of mach numbers and pole locations. A table containing these controller gains as a function of mach number is prepared and entered into simulation model.

5.2 Simulation Results

The simulations for the first and second methods are done with the same initial conditions for comparison. The initial Euler angles are chosen as

$$\begin{bmatrix} \Phi(0) \\ \Theta(0) \\ \Psi(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5\,rad \\ 0 \end{bmatrix} \tag{5.5}$$

In all cases of simulations, controller is opened at 4th second.

In the first method, there are two cases. In the first case, the desired trajectory is given in a single plane(pitch plane) only. It is in the following form,

$$Z_d = k_{1z}\sin(k_{1x}X) + k_{2z}\sin(k_{2x}X) + k_{3z}\sin(k_{3x}X) \tag{5.6}$$

where $k_{1z} = -3000m$, $k_{2z} = 600$, $k_{3z} = 180$, $k_{1x} = \frac{\pi}{35000m}$, $k_{2x} = (2.4).k_{1x}$, $k_{3x} = (2.7).k_{1x}$. This trajectory is shown in the Figure (5.1)

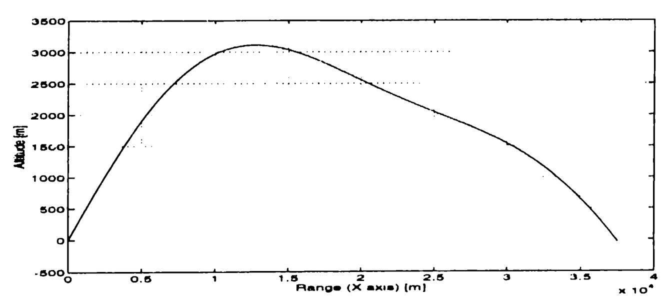

Figure 5.1: Desired trajectory

The guidance algorithm for the pitch plane is used in this case. The guidance control parameters are chosen to obtain the following position error dynamics,

$$\ddot{e} + (1.2)\cdot\dot{e} + e = 0$$

The autopilot poles are chosen as

$$\begin{aligned} p_1 &= -20 \\ p_2 &= -6 + 8j \\ p_3 &= -6 - 8j \end{aligned} \tag{5.7}$$

These choices are done under the following considerations:

1. The dynamics of the guidance should be sufficiently small compared with the dynamics of the autopilot

2. The autopilot dynamics should be kept small enough to prevent the canard deflections from saturations. The position error for this case is shown in Figure (5.2). As can be seen from this graph, position error goes to zero with an acceptable dynamics.

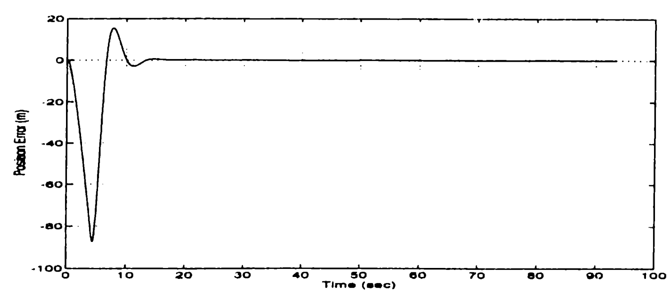

Figure 5.2: Position Error

The angle of attack and canard deflections are shown in the Figure (5.3). The angle of attack is small enough which is assumed in the design of the autopilot, and the canard deflection is not saturated.

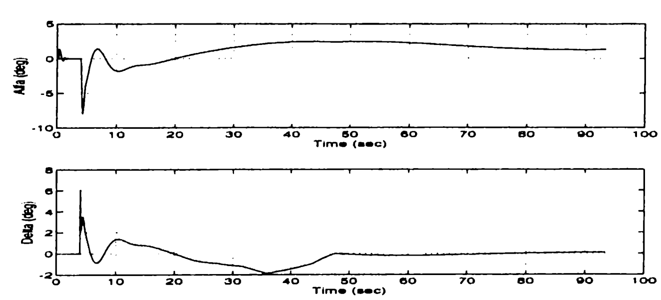

Figure 5.3: Angle of attack and Canard deflection

In the second case, the trajectory is given in two dimensional plane as a function of X. In this case, there is a desired position vector of the following form,

$$y_d = \begin{bmatrix} Y_d \\ Z_d \end{bmatrix} = \begin{bmatrix} (0.1) \cdot [k_{1z}\sin(k_{1x}X) + k_{2z}\sin(k_{2x}X) + k_{3z}\sin(k_{3x}X)] \\ k_{1z}\sin(k_{1x}X) + k_{2z}\sin(k_{2x}X) + k_{3z}\sin(k_{3x}X) \end{bmatrix} \tag{5.8}$$

This trajectory in three dimension is shown in the Figure (5.4).

The guidance parameters for both planes are chosen same with the previous case. The position error in three dimensional plot is shown in the Figure(5.5).

The position errors in Y and Z axis is shown in the Figure (5.6)

The simulation results for second method are only for the pitch plane. The error dynamics is chosen as

$$\ddot{e} + (1.2) \cdot \dot{e} + e = 0$$

The position error goes to zero with an acceptable way as can be seen in the Figure (5.7). However, there exists high oscillations in canard deflection and (as a result) in the angle of attack shown in the Figure (5.8). But this does not cause a problem in the simulation environment. Nevertheless, this may cause some problems in the real environment.

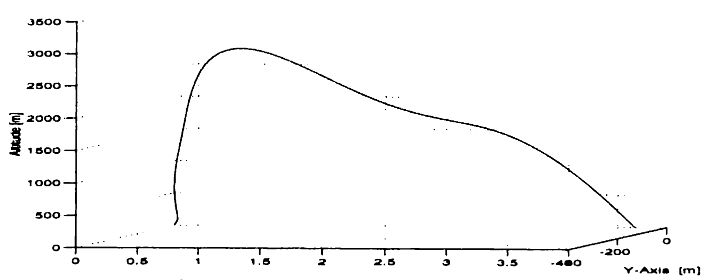

Figure 5.4: Desired trajectory for 6-DOF simulation

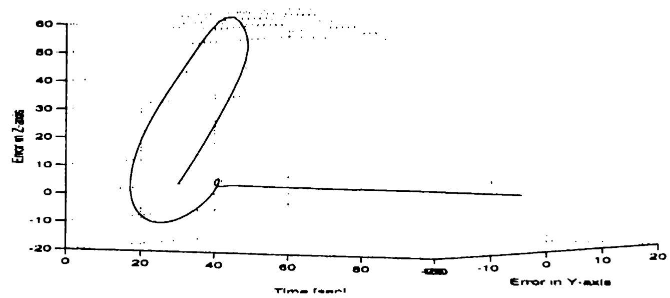

Figure 5.5: Position error for 6-DOF simulation

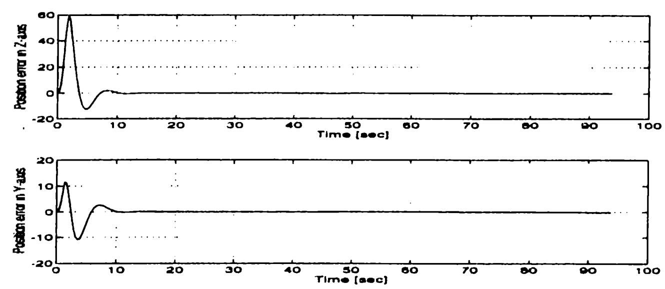

Figure 5.6: Position errors in Y and Z axis

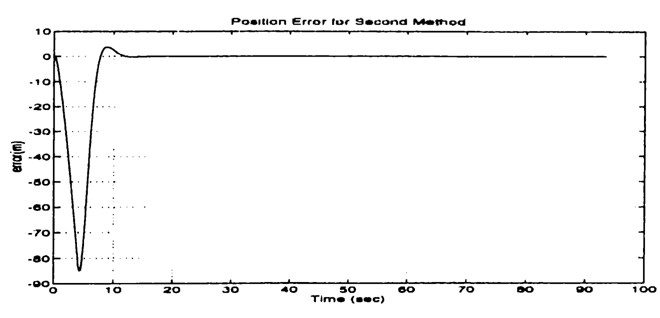

Figure 5.7: Position Error for Second Method

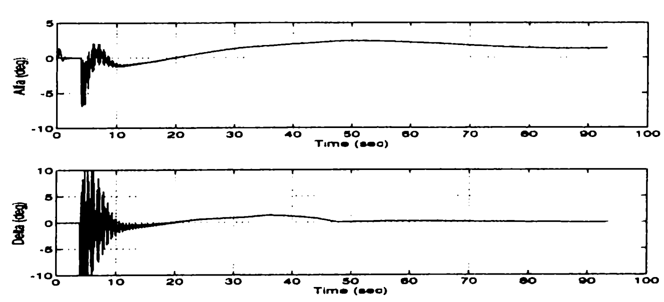

Figure 5.8: Angle of Attack and Canard Deflection

Chapter 6

CONCLUSION

In this thesis, we designed two controllers for a trajectory tracking missile by using dynamic inversion method. The approach used in the design of the first controller is based on the two-time scale idea. When we analyzed the dynamical equations of the missile, we observed that two-time scale approach is suitable for order reduction of these equations. The position and attitude of the missile change slowly compared with linear and angular velocities of the missile. Hence, each part is controlled independently. This scheme can be thought as inner and outer loop. The inner loop controls the fast dynamics, and the outer loop controls the slow dynamics. The designs are done such that the outer loop dynamics is sufficiently slow compared with the inner loop dynamics. Extensive simulations are done to validate this assumption.

The direct application of the dynamic inversion method to the controller design for the missile results in the second controller. The simulation results reveals that this second method works fine for the perfect model. But the high oscillations in control input may create problems in real time environment.

The disadvantage of using NID method directly in the control design is that the system parameters appear in the expression of the control input. Since the i/o linearization by the inverse dynamics is based on the cancellation of the nonlinearity, any unmodeled dynamics may result in uncancelled nonlinearity, which, in turn, may cause the system to be unstable. The high oscillations in control input is also a resut of the direct apllication of the NID method in the controller design. In the first method, this disadvantage of using NID is avoided by seperating the system dynamics into two part.

Simulation results show also that the zero-dynamics of the system is stable. The necessary condition for the stability of the zero-dynamics of the system is given in the chapter 4.

REFERENCES

[1] Lane, S., and Stengel, R. F., "Flight Control design Using nonlinear Inverse Dynamics", Automatica, Vol. 24, No. 4, pp. 471-483, 1988

[2] S. Antony Snell, Dale F. Enns, William L. Gerrard, " Nonlinear Inversion flight control for a Supermaneuverable Aircraft", Journal of Guidance, Control, and Dynamics Vol. 15, No. 4, pp 976-984, July-August 1992

[3] Shamma, J. S., Athans, M., "Analysis of Gain Scheduled Control for Nonlinear Plants", IEEE Trans. Automat. Contr., vol. 35, No. 8, pp. 898-907, Aug. 1990

[4] S. M. Shahruz, S. Behtash, "Design of Controllers for linear Parameter Varying Systems by the Gain Scheduling Technique" J. Math. Anal. Appl., vol. 168, no. 1, pp. 195-217, 1992.

[5] Wang, J., Rugh, W. J., "Feedback Linearization Families for Nonlinear Systems", IEEE Transactions on Aut. Cont., Vol. AC-32, No. 10, pp 935-940, Oct. 1987

[6] Richard J. Adams and Siva S. Banda, "Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis",IEEE Transactions on Control Systems Tech., Vol.1, No. 2, pp. 80-91, Jun. 1993

[7] Richard J. Adams, Jmaes M. Buffington, and Siva S. Banda, "Design of Nonlinear Laws for High-Angle-of-Attack Flight", Journal of Guidance, Control, and Dynamics Vol. 15, No. 4, pp 976-984, July-August 1992

[8] Wilson J. Rugh, "An Extended Linearization Approach to Nonlinear Sytem Inversion",IEEE Transactions on Automatic Control, Vol. AC-31, No. 8, pp.725-733, Aug. 1986

[9] Etkin B. (1972). Dynamics of Atmospheric Flight. John Wiley, New York.

[10] Blakelock, J. H. (1991), "Automatic Control of Aircraft and Missiles" 2nd Ed. John Wiley,

[11] Donald Mclean. (1990), "Automatic Flight Control Systems". Prentice Hall

[12] Vidyasagar, M, "Nonlinear System Analysis". 2nd Ed. Prentice Hall

[13] Remzi Barlas, M, (1992)"Model-Following Control Applications to Nonlinear Mechanical Systems",M.S.E. Thesis Virginia Polytechnic Institute and State University.