CONTROL ACTUATION SYSTEMS AND SEEKER UNITS OF AN
AIR-TO-SURFACE GUIDED MUNITION

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

BY

ELZEM AKKAL

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2003

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. Mübeccel Demirekler
Head of the 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 Leblebicioğlu
Supervisor

Examining Committee Members

Prof. Dr. Kemal Özgören   _____

Prof. Dr Kemal Leblebicioğlu   _____

Prof. Dr. Uğur Halıcı   _____

Assoc. Prof. Dr. Ozan Tekinalp   _____

M. Sc. Özgür Ateşoğlu   _____

ABSTRACT

CONTROL ACTUATION SYSTEMS AND SEEKER UNITS OF AN

AIR-TO-SURFACE GUIDED MUNITION

Akkal, Elzem

M.S., Department of Electrical and Electronics Engineering

Supervisor : Prof. Dr. Kemal Leblebicioğlu

December 2003, 120 pages

This thesis proposes a modification to an air to surface guided munition (ASGM) from bang-bang control scheme to continuous control scheme with a little cost. In this respect, time domain system identification analysis is applied to the control actuation system (CAS) of ASGM in order to obtain its mathematical model and controller is designed using pulse width modulation technique. With this modification, canards would be deflected as much as it is commanded to. Seeker signals are also post-processed to obtain the angle between the velocity vector and target line of sight vector. The seeker is modeled using an artificial neural network. Non-linear flight simulation model is built using MATLAB Simulink and obtained seeker and CAS models are integrated to the whole flight simulation model having

6 degrees of freedom. As a flight control unit, fuzzy logic controller is designed, which is a suitable choice if an inertial measurement sensor will not be mounted on the munition. Finally, simulation studies are carried out in order to compare the performance of the "ASGM" and "improved ASGM" and the superiority of the new design is demonstrated.

Keywords: System Identification, Laser Seeker, Control Actuation System, Artificial Neural Network, Pulse Width Modulation, Fuzzy Logic Control.

ÖZ

HAVADAN KARAYA GÜDÜMLÜ BİR MÜHİMMAT İÇİN KONTROL TAHRİK SİSTEMİ VE ARAYICI BİRİMLERİ

Akkal, Elzem

Yüksek Lisans, Elektrik-Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Kemal Leblebicioğlu

Aralık 2003, 120 sayfa

Bu tez, havadan karaya güdümlü bir mühimmatın (HKGM) az bir maliyetle bang-bang kontrolden sürekli kontrol uygulamasına geçirilmesi değişikliğini önermektedir. Bu bağlamda, havadan karaya güdümlü mühimmatın kontrol tahrik sisteminin matematiksel modelinin oluşturulması için sistem tanımlama çalışmaları uygulanmıştır ve darbe genişliği modulasyonu tekniği kullanılarak kontrol algoritmaları tasarlanmıştır. Bu değişiklikle kanatçıklar istenilen açılara getirilebileceklerdir. Hız vektörüyle görüş hattı vektörünün oluşturduları açıyı elde edebilmek için arayıcı sinyalleri de işlenmiştir. Arayıcı yapay sinir ağları kullanılarak modellenmiştir. Doğrusal olmayan uçuş modeli MATLAB Simulink

kullanılarak oluşturulmuş, ve kontrol tahrik sistemi modeliyle arayıcı modeli de 6 serbestlik dereceli uçuş simülasyon modeline entegre edilmiştir. Uçuş denetleyicisi olarak da, mühimmata ataletsel ölçüm sensörleri takılmayacaksa da kullanılabilecek bulanık mantık denetleyicisi tasarlanmıştır. Son olarak da, "HKGM" ile "geliştirilen HKGM"nin performanslarını karşılaştırabilmek için, simülasyon çalışmaları yapılmış ve yeni tasarımın üstünlüğü gösterilmiştir.

Anahtar Kelimeler: Sistem Tanımlama, Lazer Arayıcı, Kontrol Tahrik Sistemi, Yapay Sinir Ağları, Darbe Genişlik Modulasyonu, Bulanık Mantık Denetleyicisi.

Dedicated to the most important person

I could ever have in my life,

Bülent Semerci...

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

NOMENCLATURE

γ_p (or γ_{xz})   Pitch Plane Error Angle

γ_Y (or γ_{xy})   Yaw Plane Error Angle

h   Altitude

m   Mass of the munition

M   Mach number

Q_d Dynamic pressure

R   Universal gas constant

S   Aerodynamic reference area

V_T Total velocity

C   Speed of sound

T   Atmospheric temperature

ρ Atmospheric density

γ Specific heat ratio

α Angle of attack

β Side slip angle

Φ Roll angle of the munition

Θ Pitch angle of the munition

Ψ	Yaw angle of the munition
$T_B{}^E$	Body fixed frame to Earth fixed frame transformation matrix
X_B, Y_B, Z_B	Body fixed reference frame axes set
X_E, Y_E, Z_E	Earth fixed reference frame axes set
g_x, g_y, g_z	Gravitational acceleration components in body-fixed frame
ASGM	Air-to-surface guided munition
DOF	Degree of freedom
CAS	Control Actuation System
LOS	Line of Sight
PWM	Pulse Width Modulation
TOF	Time of Flight

CHAPTER 1

INTRODUCTION

1.1 General Information About the ASGM

The ASGM studied in this thesis is taken as a generic laser guided air to surface munition, which has a guidance system that detects and guides the weapon to a target illuminated by an external laser source. It is released from an aircraft and it has no thrust. The stabilizing wings, which provide lift to the system, are located at the aft of the munition. The only control mechanism is canards, which are attached to the front of the body. Its seeker is mounted through gimbals to the missile body. Consequently it aligns itself along the velocity vector. This structure enables the weapon to correct its flight path relative to the target.

When detector senses the position of the target with respect to the flight path of ASGM, guidance electronics commands the control actuation system, in order to correct the trajectory. When the CAS is commanded, it deflects the canards to their full position, which is about 10 degrees. This control technique is called as "bang-bang control".

Delivery accuracy and the range of the munition can be increased, if the control technique is changed from "bang-bang" to continuous. In this aspect, seeker and control actuation system are needed to be modified.

Detector in this system is a 4 quadrant laser detector whose signals are processed to determine only if the LOS (Line of Sight) vector is pointing below/above and/or left/right of the velocity vector. According to command generated by seeker electronics, canards are moved to full deflection angles either + or − 10 degrees. Whereas, the same detector can be used to estimate the angle formed between LOS and velocity vectors. There are some studies related with estimating the position of laser spot on the detector, and tracking the laser spot by moving the mirror, which faces the detector at some angle [8], [24].

Moreover, with a little modification, CAS can be made to move to any desired deflection angle. Suggested modification is controlling CAS using on/off solenoid valves via PWM in order to hold it at the desired angle. Varseveld and Bone have done a very similar task using PWM. The actuator in their system is also a pneumatic actuator and valves are solenoid valves. They aimed to control the position of the actuator using a novel PWM scheme [17]. Linnet and Smith also worked on positioning the pneumatic actuator using on/off valves, but their concern was to hold the actuator at the desired position with a very high accuracy like 1 mm rather than making the actuator to follow commanded position [18].

After the improvements made on CAS and seeker, a flight controller is left to be designed. Since the most important constraint is minimum cost, munition should be able to be controlled without adding any other component, such as accelerometer or gyroscope, to the system, if possible. Therefore a fuzzy logic controller, whose inputs and outputs are error angles and desired canard deflections respectively, could be an adequate solution to the flight controller question. Especially in recent years, there is a considerable amount of research about controlling the missiles and aircrafts via fuzzy logic. Fuzzy logic implementation of proportional navigation guidance in missiles is one of the examples among these studies [23],[25]. However application similar to the one mentioned in this thesis can not be found, since flight controller problem here is a very particular problem.

1.2 About Thesis

Since this thesis aims to improve the ASGM, the proof of any improvement can only be demonstrated with flight simulations. Therefore, nonlinear flight simulation model is built first. The derivation of these nonlinear flight equations are explained in Chapter 2.

In the 3rd Chapter, mathematical model of the control actuation system is estimated using system identification toolbox of MATLAB. An input is applied to the system and the corresponding output is measured by an encoder and potentiometer. Using these measurements, mathematical model that would create this input-output data is estimated.

In the next chapter, pulse width modulation algorithm is explained and a controller is designed for the plant estimated in the previous chapter.

In Chapter 5, the structure of the seeker is explained and a model for estimating error angles in pitch and yaw plane is suggested. Suggested estimator is obtained by an artificial neural network.

In the beginning of Chapter 6 brief description of fuzzy logic controller is made and fuzzy logic controller designed as a flight controller is described. Afterwards integration of fuzzy logic controller, modified CAS and seeker models are explained. Next, the improved ASGM is tested and its performance is compared with the ASGM before modification. In order to test the overall model performance, it is run for several scenarios, which are described in Chapter 7. Finally Chapter 8 gives the concluding remarks of this thesis.

CHAPTER 2

MISSILE EQUATIONS OF MOTION

2.1 Introduction

The trajectory, velocity and orientation of a missile are estimated by solving 12 non-linear differential equations. These equations can be analyzed in two groups, dynamic equations and kinematics equations. Dynamic equations are derived by applying Newton's laws of motion, which relate the summation of the external forces and moments to the linear and angular accelerations of the body. Kinematic equations are the consequences of transformation matrix applications, which build the bridge between the reference axis systems through the use of Euler angles [1] [2] [3].

2.2 Development of Equations of Motion

2.2.1 Reference Axis Systems and Axis Transformations

The form of the equations of motion is governed by the choice of the axis system. In this thesis two reference frames are used in the derivation as given in Figure 2-1; Earth fixed axis system as inertial frame and body fixed axis system. The earth

fixed system will be regarded as an inertial reference frame, in which Newton's

laws of motion are valid. This means that the rotation of the Earth is neglected.
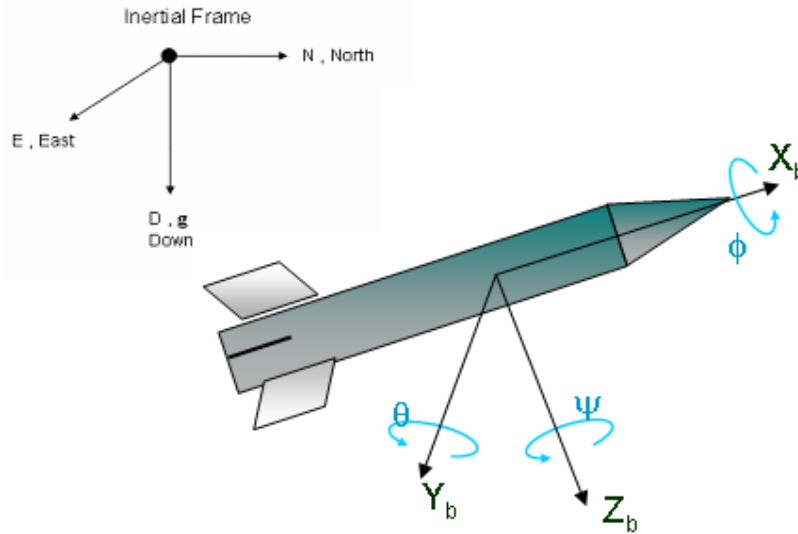
Figure 2-1 Inertial (earth fixed) and body fixed axis systems

Earth fixed frame is chosen such that X_E points north, Y_E points east and Z_E points

down in the direction of gravity vector. The center of the earth fixed frame can be

anywhere on Earth, whereas center of the body fixed frame is missile's center of

gravity and X_B points the nose of the missile, Y_B points the right control surface of

the missile and Z_B points down.

After introducing two reference axis systems, earth fixed and body frames, a

relationship needs to be established between the vectorial quantities expressed in

specified 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. The sequence of rotation is customarily taken as follows:

✓ Rotate earth axes through some azimuthal angle Ψ about the axes Z_E, to obtain intermediate axes X_1, Y_1 and Z_1.

✓ Next rotate these axis X_1, Y_1 and Z_1 through some angle of elevation Θ, about the axes Y_1, to obtain a second intermediate set of axes X_2, Y_2 and Z_2.

✓ Finally, rotate the new axes X_2, Y_2 and Z_2 through an angle of bank Φ about the axis, X_2, to reach the body axes X_B, Y_B and Z_B.

The most common rotation order is azimuth Ψ, pitch Θ and roll Φ, which are called Euler angles. The angular orientation of the body is given by the Euler angles. Following is the corresponding rotation matrices:

$$\mathbf{T}_\Psi = \begin{bmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad\qquad (2.1)$$

$$\mathbf{T}_\Theta = \begin{bmatrix} \cos\Theta & 0 & -\sin\Theta \\ 0 & 1 & 0 \\ \sin\Theta & 0 & \cos\Theta \end{bmatrix} \qquad\qquad (2.2)$$

$$\mathbf{T}_\Phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} \qquad\qquad (2.3)$$

The final transformation matrix is called the direction cosine matrix and is defined as:

$$\mathbf{T}_E^B = \begin{bmatrix}\mathbf{T}_\Phi\end{bmatrix}\begin{bmatrix}\mathbf{T}_\Theta\end{bmatrix}\begin{bmatrix}\mathbf{T}_\Psi\end{bmatrix} \qquad\qquad (2.4)$$

$\mathbf{T_E^B}$ representation stands for the transformation from earth axis to body axis system. Open form of the final transformation matrix is the following:

$$\mathbf{T_E^B} = \begin{bmatrix} c\Theta c\Psi & c\Theta s\Psi & -s\Theta \\ s\Phi s\Theta c\Psi - c\Phi s\Psi & s\Phi s\Theta s\Psi + c\Phi c\Psi & s\Phi c\Theta \\ c\Phi s\Theta c\Psi + s\Phi s\Psi & c\Phi s\Theta s\Psi - s\Phi c\Psi & c\Phi c\Theta \end{bmatrix} \qquad (2.5)$$

where c stands for cosine and s stands for sine.

Transformation matrices are orthogonal matrices and following property holds for this type of matrices.

$$\mathbf{T_B^E} = \left(\mathbf{T_E^B}\right)^{-1} = \left(\mathbf{T_E^B}\right)^{T} \qquad (2.6)$$

2.2.2 Derivation of Dynamic Equations

As it is mentioned in the introduction part of this chapter that derivations of equations of motion is based on Newton's second law. This law can be expressed by two vector equations,

$$\mathbf{F} = \frac{d}{dt}\left(m\mathbf{V_T}\right)\bigg|_{I} \qquad (2.7)$$

$$\mathbf{M} = \frac{d}{dt}\left(\mathbf{H}\right)\bigg|_{I} \qquad (2.8)$$

where **F** represents the sum of aerodynamic, gravitational and propulsive forces, **M** represents the sum of torques resulted from lift and drag, **H** is the angular momentum and the derivatives are to be taken with respect to an inertial reference frame. Before proceeding with the derivation, we need to make some assumptions.

- Missile is a rigid body

- Mass of the missile remains constant

Considering these assumptions, Equation (2.7) and (2.8) can be re-expressed as,

$$\mathbf{F} = m\frac{d}{dt}(\mathbf{V_T})\bigg|_I \qquad\qquad (2.9)$$

$$\mathbf{M} = \frac{d}{dt}(\mathbf{H})\bigg|_I \qquad\qquad (2.10)$$

Derivation of the equations for translational and rotational motions, which are the results of force and moment equations respectively, will be dealt separately in the following two sub-sections.

2.2.2.1 Translational Motion

Force equation given in (2.7) is written with respect to inertial frame (earth fixed frame), so differentiation with respect to inertial frame while expressing the vectorial quantities in body frame yields the following expression.

$$\frac{d}{dt}(\mathbf{V_T})\bigg|_I \cong \frac{d}{dt}(\mathbf{V_T})\bigg|_E = \left(\frac{d}{dt}(\mathbf{V_T})\bigg|_B + \boldsymbol{\omega} \times \mathbf{V_T}\right) \qquad (2.11)$$

where ω is the angular velocity of the missile with respect to the earth fixed axis system. $\mathbf{V_T}$ and $\boldsymbol{\omega}$ can be expressed in body frame as follows,

$$\mathbf{V_T} = \mathbf{i}u + \mathbf{j}v + \mathbf{k}w \qquad\qquad (2.12)$$

$$\boldsymbol{\omega} = \mathbf{i}p + \mathbf{j}q + \mathbf{k}r \qquad\qquad (2.13)$$

$$\frac{d}{dt}\mathbf{V_T}\bigg|_B = \mathbf{i}\dot{u} + \mathbf{j}\dot{v} + \mathbf{k}\dot{w} \qquad\qquad (2.14)$$

where $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are the unit vectors along X_B, Y_B and Z_B. Cross product term in Equation 2.11 is given by,

9

$$\boldsymbol{\omega} \times \mathbf{V_T} = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ p & q & r \\ u & v & w \end{bmatrix} \tag{2.15}$$

Expanding,

$$\boldsymbol{\omega} \times \mathbf{V_T} = \mathbf{i}(qw - rv) + \mathbf{j}(ru - pw) + \mathbf{k}(pv - qu) \tag{2.16}$$

$\mathbf{F}$ can be written in terms of its components as follows,

$$\mathbf{F} = \mathbf{i}F_x + \mathbf{j}F_y + \mathbf{k}F_z \tag{2.17}$$

Equating the X, Y, Z components of equations (2.14), (2.16) and (2.17), equations of translational motion are obtained [15].

$$F_x = m(\dot{u} + qw - vr) \tag{2.18}$$

$$F_y = m(\dot{v} + ru - pw) \tag{2.19}$$

$$F_z = m(\dot{w} + pv - qu) \tag{2.20}$$

2.2.2.2 Rotational Motion

In order to obtain the equations of angular motion, first of all, expression for angular momentum must be given.

$$\mathbf{H} = \mathbf{I}\boldsymbol{\omega} \tag{2.21}$$

where I is the inertia matrix in the body-fixed frame defined as follows,

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \tag{2.22}$$

where I_{ii} denotes a moment of inertia and I_{ij} a product of inertia $j \neq i$. In general missiles are symmetrical about the plane XZ and YZ, and consequently all product

of inertia values are equal to zero and the moment of inertias about the y and z axis are equal to each other. Therefore angular momentum is expressed as;

$$\mathbf{H} = \mathbf{i}I_{xx}p + \mathbf{j}I_{yy}q + \mathbf{k}I_{zz}r \qquad (2.23)$$

where

$$\boldsymbol{\omega} = \mathbf{i}p + \mathbf{j}q + \mathbf{k}r \qquad (2.24)$$

Now, differentiation of angular momentum with respect to inertial frame yields the similar expression to Equation (2.11).

$$\mathbf{M} = \left(\frac{d}{dt}(\mathbf{H})\bigg|_{B} + \boldsymbol{\omega} \times \mathbf{H} \right) \qquad (2.25)$$

where

$$\frac{d}{dt}\mathbf{H}\bigg|_{B} = \mathbf{i}p\&+ \mathbf{j}q\&+ \mathbf{k}r\& \qquad (2.26)$$

and

$$\boldsymbol{\omega} \times \mathbf{H} = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ p & q & r \\ I_{xx}p & I_{yy}q & I_{zz}r \end{bmatrix} \qquad (2.27)$$

Expanding the cross product;

$$\boldsymbol{\omega} \times \mathbf{H} = \mathbf{i}(qrI_{zz} - qrI_{yy}) + \mathbf{j}(prI_{xx} - prI_{zz}) + \mathbf{k}(pqI_{yy} - pqI_{xx}) \qquad (2.28)$$

The first term becomes zero in this expression, because $I_{zz}=I_{yy}$ due to the symmetry of the ASGM under study.

If we write $\mathbf{M}$ as follows, and equate the components of Equation (2.26) and Equation (2.28), we obtain the rotational equations of motion

$$\mathbf{M} = \mathbf{i}L + \mathbf{j}M + \mathbf{k}N \qquad (2.29)$$

Rotational equations of motion are;

$$L = I_{xx}\dot{p} \tag{2.30}$$

$$M = I_{yy}\dot{q} + (I_{xx} - I_{zz})pr \tag{2.31}$$

$$N = I_{zz}\dot{r} + (I_{yy} - I_{xx})pq \tag{2.32}$$

2.2.3 Derivation of Kinematic Equations

Position and attitude with respect to earth axis system is found using kinematic equations.

2.2.3.1 Translational Kinematics

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

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{T_B^E} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.33}$$

where $\mathbf{T_B^E}$ is the transformation matrix from body axes system to earth axis system, which is the transpose of $\mathbf{T_E^B}$ (Equation (2.5)), x, y and z are the location of the body center of gravity, u, v and w are the velocity components along the body axes. The open form of this equation is as follows:

$$\begin{aligned} \dot{x} &= u\cos\Theta\cos\Psi + v(\sin\Phi\sin\Theta\cos\Psi - \cos\Phi\sin\Psi) \\ &\quad + w(\cos\Phi\sin\Theta\cos\Psi + \sin\Phi\sin\Psi) \end{aligned} \tag{2.34}$$

$$\begin{aligned} \dot{y} &= u\cos\Theta\sin\Psi + v(\sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi) \\ &\quad + w(\cos\Phi\sin\Theta\sin\Psi - \sin\Phi\cos\Psi) \end{aligned} \tag{2.35}$$

$$\dot{z} = -u\sin\Theta + v\sin\Phi\cos\Theta + w\cos\Phi\cos\Theta \tag{2.36}$$

2.2.3.2 Rotational Kinematics

It was stated that for transformation matrices, $\mathbf{T_B^E} = \left(\mathbf{T_E^B}\right)^{-1} = \left(\mathbf{T_E^B}\right)^{T}$, so from this, following expression can be written.

$$\mathbf{T_E^B}\left(\mathbf{T_E^B}\right)^{T} = \mathbf{I} \tag{2.37}$$

Taking the derivative of this expression,

$$\dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T} + \mathbf{T_E^B}\left(\dot{\mathbf{T}}_E^B\right)^{T} = 0 \tag{2.38}$$

$$\dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T} = -\left(\dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T}\right)^{T} \tag{2.39}$$

The above equation indicates that $\dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T}$ is a skew symmetric matrix and so there exists a skew symmetric matrix $\mathbf{\Omega}$ generated from a column $\boldsymbol{\omega}$ such that $\mathbf{\Omega} = \dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T}$. $\boldsymbol{\omega}$ is composed of the components of the angular velocity of the body frame with respect to earth fixed frame expressed at the body frame. Following is the expression for $\boldsymbol{\omega}$.

$$\boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.40}$$

and the skew symmetric form of this vector is as follows,

$$\mathbf{\Omega} = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \tag{2.41}$$

Solving $\mathbf{\Omega} = \dot{\mathbf{T}}_E^B\left(\mathbf{T_E^B}\right)^{T}$ for the Euler angle rates, the rotational kinematic equations are found as follows [15],

$$\dot{\Psi} = p + (q\sin\Phi + r\cos\Phi)/\cos\Theta \tag{2.42}$$

$$\dot{\Theta} = q \cos \Phi - r \sin \Phi \qquad\qquad (\,2.43\,)$$

$$\dot{\Phi} = p + (q \sin \Phi + r \cos \Phi) \tan \Theta \qquad\qquad (\,2.44\,)$$

This equation set has singularity when θ is equal to 90 (or 270) degrees. An alternative method, which is called quaternions, is frequently used not to encounter this singularity problem. For the munition considered here, 90 degrees pitch angle is only attained when it hits the target, which is the end of the simulation. Therefore quaternion equations are not required to be used for this munition.

2.3 Forces and Moments Acting on a Missile

Now we are in a position that if we know the forces and moments acting on the munition, we can estimate the location of its center of gravity on earth, its velocity and its heading. Therefore we need to define these forces and moments first, and explain the ways to determine them.

There are mainly three types of forces acting on a rigid body missile; gravitational force, aerodynamic force and thrust. The ASGM under study is a free falling projectile without thrust, so we are left with gravitational and aerodynamic force. Moments are the consequences of only aerodynamic forces, because gravity acts at the center of gravity of the bomb, which is the center for body axes system.

2.3.1 Gravitational Force

The gravitational force acting on a missile is most obviously expressed in terms of earth fixed frame. With respect to these axes the gravitational force is directed

along the Z_E axis and its magnitude is assumed to be constant. Derived equations of motion uses the body axes system, therefore gravitational force should be resolved into body axes system by using transformation matrix from earth to body ($\mathbf{T}_E^B$).

$$
\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = T_E^B \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}
\tag{2.45}
$$

Result of this multiplication is as follows;

$$
\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = \begin{bmatrix} -g\sin\Theta \\ g\cos\Theta\sin\Phi \\ g\cos\Theta\cos\Phi \end{bmatrix}.
\tag{2.46}
$$

Gravitational forces represented in body frame are added to the force equations obtained in Translational Motion section and the final form of these equations is as follows,

$$
F_{Ax} - mg\sin\Theta = m(\dot{u} + qw - vr)
\tag{2.47}
$$

$$
F_{Ay} + g\cos\Theta\sin\Phi = m(\dot{v} + ru - pw)
\tag{2.48}
$$

$$
F_{Az} + g\cos\Theta\cos\Phi = m(\dot{w} + pv - qu)
\tag{2.49}
$$

where F_{Ax}, F_{Ay} and F_{Az} represent the aerodynamic forces.

2.3.2 Aerodynamic Forces and Moments

The aerodynamic forces and moments are represented by F_{Ax}, F_{Ay}, F_{Az} and L, M, N respectively. The most general expression for aerodynamic forces and moments are as follows:

$$\begin{bmatrix} F_{Ax} \\ F_{Ay} \\ F_{Az} \end{bmatrix} = Q_d A \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \tag{2.50}$$

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = Q_d A d \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix} \tag{2.51}$$

where Q_d is the dynamic pressure, A is the maximum cross-sectional area of the bomb, d is the bomb diameter and C_i's are the dimensionless aerodynamic force and moment coefficients. The expression for dynamic pressure is as follows,

$$Q_d = \frac{1}{2}\rho V_T^2 \tag{2.52}$$

where $V_T = \sqrt{U^2 + V^2 + W^2}$

In dynamic pressure expression ρ is the air density and it changes with altitude h, which is down component of position vector with minus sign. Its change with altitude is given below.

$$\rho = \begin{cases} \rho_0(1 - 0.00002256h)^{4.256} & h \leq 10000m \\ 0.412 \times 10^{(-0.000151(h-10000))} & h > 10000m \end{cases} \tag{2.53}$$

where ρ_0 is the air density at sea level (1.223 kg/m³). This formula is obtained by curve fitting to the density variation in the ICAO standard atmosphere [15]. Dimensionless aerodynamic force and moment coefficients are designated as,

C_x : Axial force coefficient

C_y : Side force coefficient

C_z : Normal force coefficient

C_l : Rolling moment coefficient

C_m : Pitching moment coefficient

C_n : Yawing moment coefficient.

These aerodynamic force and moment coefficients are the functions of the Mach number and flight parameters such as the angle of attack, side slip angle, the control surface deflections ($\delta_e, \delta_r, \delta_a$), the body angular rates p, q, r. Subscripts e, r and a stand for elevator, rudder and aileron deflections, respectively. So C_i can be written as,

$$C_i = C_i(M, \alpha, \beta, \delta_e, \delta_r, \delta_a, p, q, r) \qquad (2.54)$$

Mach number is a dimensionless number defined as ratio of the total speed of the body to the speed of sound estimated at the altitude of the body. The expression for Mach number is given below.

$$M = \frac{V_T}{C} \qquad (2.55)$$

where C is the speed of sound and can be calculated as follows,

$$C = \sqrt{\gamma R T} \qquad (2.56)$$

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

$$T = \begin{cases} T_0(1 - .000002256) & h \leq 10000m \\ T_0 \times 0.7744 & h > 10000m \end{cases} \qquad (2.57)$$

where T_0 is the temperature at sea level and is equal to 293 K. This formula is obtained by curve fitting to the temperature variation in the ICAO standard atmosphere [15].

Angle of attack and side slip angle are defined by the following expressions. These angles are important because they orient the lift and drag forces with respect to body frame.

$$\alpha = a\,\tan\!\left(\frac{w}{u}\right) \tag{2.58}$$

$$\beta = a\,\sin\!\left(\frac{v}{V_T}\right) \tag{2.59}$$

Angle of attack and side slip angle expressions can be further simplified due to the fact that w and v being much smaller when compared with u. Moreover u can be taken equal to V_T, since v and w are much smaller than u. Simplified expressions for angle of attack and side slip angle are given below.

$$\alpha = \frac{w}{u} \tag{2.60}$$

$$\beta = \frac{v}{u} \tag{2.61}$$

As it is mentioned in the previous paragraphs, aerodynamic coefficients are nonlinear functions of Mach number and flight parameters. If this relationship is linearized using Taylor series expansion, following expression is obtained.

$$\begin{aligned}
C_i &= C_{i0}(M) + C_{i\alpha}(M)\alpha + C_{i\beta}(M)\beta + C_{i\delta e}(M)\delta_e + \\
&\quad C_{i\delta r}(M)\delta_r + C_{i\delta a}(M)\delta_a + C_{ip}(M)p\frac{d}{2V_T} + \\
&\quad C_{iq}(M)q\frac{d}{2V_T} + C_{ir}(M)r\frac{d}{2V_T} + H.O.T
\end{aligned} \tag{2.62}$$

where H.O.T stands for higher order terms. $C_{i\theta}$ and $C_{i\dot\theta}$ are the short hand notations for the derivative of aerodynamic coefficient with respect to θ and $\dot\theta$, as given

18

below. θ and $\dot{\theta}$ stand for the angle quantities such as α, β etc., and angle rate quantities, such as p, q and r, respectively.

$$C_{i\theta}(M) = \frac{\partial C_i}{\partial \theta} \qquad (2.63)$$

$$C_{i\dot{\theta}}(M) = \frac{\partial C_i}{\partial \left(\dot{\theta} \frac{d}{2V_T} \right)} \qquad (2.64)$$

These aerodynamic coefficients can be estimated using wind tunnels, CFD (Computational Fluid Dynamics) methods and softwares like Missile Datcom. In this thesis Missile Datcom is used in the calculation of aerodynamic coefficients. Missile Datcom is run for a number of Mach number values and the coefficients are inserted in a look-up table as a function of Mach number, which calculates the aerodynamic coefficients by interpolating the coefficients corresponding to nearby Mach number values.

Expressions (2.63) and (2.64) are general expressions for aerodynamic force and moment coefficients. Particular expressions for each aerodynamic force and moment coefficients are given below.

$$C_x = C_{x0}(M) \qquad (2.65)$$

$$C_y = C_{y\beta}(M)\beta + C_{y\delta r}(M)\delta_r + C_{yr}(M)r\frac{d}{2V_T} \qquad (2.66)$$

$$C_z = C_{z\alpha}(M)\alpha + C_{z\delta e}(M) \cdot \delta_e + C_{zq}(M)q\frac{d}{2V_T} \qquad (2.67)$$

$$C_l = C_{l\delta a}(M)\delta_a + C_{lp}(M)p\frac{d}{2V_T} \qquad (2.68)$$

$$C_m = C_{m\alpha}(M)\alpha + C_{m\delta e}(M)\delta_e + C_{mq}(M)q\frac{d}{2V_T} \qquad (2.69)$$

$$C_n = C_{n\beta}(M)\beta + C_{n\delta r}(M)\delta_r + C_{nr}(M)r\frac{d}{2V_T} \qquad (2.70)$$

Number of aerodynamic coefficients used in the simulations and controller design can be reduced considering Maple-Synge rotational symmetry analysis on linear transverse aerodynamic forces and moments for a missile which has a symmetry with respect to its y and z axis. This analysis shows that, [15]

$$C_{z\alpha} = C_{y\beta}$$

$$C_{z\delta e} = C_{y\delta r}$$

$$C_{zq} = -C_{yr}$$

$$C_{m\alpha} = -C_{n\beta}$$

$$C_{m\delta e} = -C_{n\delta r}$$

$$C_{mq} = C_{nr}$$

CHAPTER 3

SYSTEM IDENTIFICATION FOR THE CONTROL ACTUATION SYSTEM

3.1 Introduction

System identification deals with the problem of building mathematical models of dynamical systems based on observed data from the systems. In other words, the problem of system identification is the determination of the system model from records of input u(t) and output y(t). There are different types of system identification analysis which can be used depending on the application. Frequency response identification, impulse response identification by deconvolution and identification from correlation functions are the examples of the ones that do not result in transfer function. There are also other approaches that find the coefficients of the assumed parametric transfer functions, such as least squares, maximum likelihood (Prediction Error Method). In this thesis we used the MATLAB System Identification Toolbox in order to solve system identification problem.

Input of the CAS is the voltages applied to the solenoid valves. Solenoid valves are either closed or open according to the voltages applied which are either 24 V or

ground. Output of the control actuation system is the resulted shaft angle. Input and output data pairs are collected by the PC with data acquisition cards. Cards are chosen to be compatible with MATLAB Real Time Windows Target toolbox. Shaft angles are measured by two transducers, absolute encoder and arc segment potentiometer.

This chapter starts with the description of the Control Actuation System of this munition. Because MATLAB System Identification Toolbox is used in system identification studies, a brief introduction to this toolbox is given next. At the end, various CAS models, estimated using this toolbox, are given together with their fitness values to the validation data set.

3.2 Control Actuation System Setup

3.2.1 Description of Control Actuation System

Control actuation system in a missile realizes the autopilot commands in order to steer the missile to the target. There are mainly three types of control actuation system: Electro-hydraulic, electromechanical and electro-pneumatic. All of them find use in missiles according to their mission. For example hydraulic actuators are mostly used in long-range ballistic missiles such as Patriot, Pershing I and II, where high power is required to maneuver the missile and large space is available for huge hydraulic CAS. Missiles such as stinger having small control surfaces use electromechanical actuators usually. Pneumatic actuators are used in missiles which require high power, explosion-proof character and easy maintenance.

Copperhead and Tomahawk are the examples of such systems having pneumatic actuators. The difference between the hydraulic and pneumatic actuators is in the type of the fluids that they make use of to transmit power. Pneumatic actuators use gas, such as air or nitrogen gas, whereas hydraulic actuators use liquids, mostly oil. The missile under consideration is a type of ASGM (Air to Surface Guided Munition) and it uses pneumatic CAS. The fluid it uses is nitrogen. Due to the bang-trial-bang nature of the missile, pneumatic CAS was also designed in bang-trial-bang nature. In other words, canards drived by CAS are in either 10 degrees deflection, or -10 degrees deflection. The trial case corresponds to the situation in which no guidance commands are generated. In that case canards are aligned with the streamline.

CAS under consideration is composed of following components;

- *Nitrogen bottle*: Pressure vessel where the pressurized (55 MPa) nitrogen gas is kept.

- *Regulator*: It regulates the pressure of the nitrogen coming from the pressure vessel to 6.5 to 9 Mpa.

- *Dampers*: There are two dampers in the system. They are used in damping aerodynamic forces on the canards. They are both viscous and spring dampers.

- *Solenoid valves*: There are two five-way solenoid valves in the system, which are controlling the motion in pitch and yaw planes. Each solenoid valve can be thought of union of two three-way solenoid valves, which can be energized independent of each other. One is responsible for the positive deflection of the canard, the other one is responsible for the negative deflection. After this point

23

we will call them as *pos-solenoid valve* and *neg-solenoid valve*. The main function of solenoid valve is to let the gas go to one of the cylinders and empty the other cylinder.
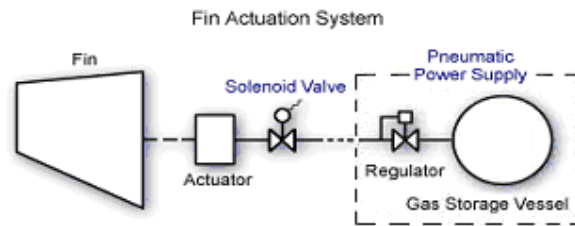
Figure 3-1 Operation of CAS

Once the nitrogen bottle opening is perforated, nitrogen gas goes to the regulator and its pressure drops to 7.5 MPa approximately and stops at the solenoid valves waiting for the guidance commands to realize. For example when "up" command is created by the guidance unit, solenoid valve opens the gate to the one of the cylinders to push the piston up, and at the same time other cylinder is emptied by the same valve to let the shaft rotate in the direction that results in "up" motion. Diagram of the Control Actuation System is given in Figure 3-2.
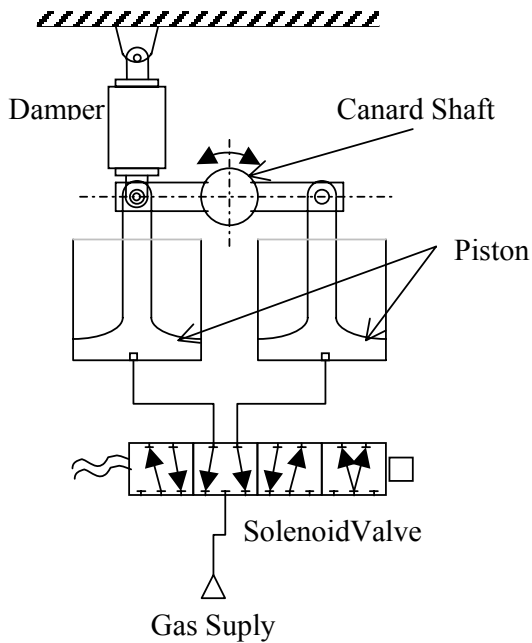
Figure 3-2 Schematic diagram of the actuator

3.2.2 Description of Test Setup

Our test setup consists of a PC, transducers, power supplies and control actuation system unit, whose principle of operation is explained in the previous paragraph. CAS unit is mounted on a table firmly, but canards could not be integrated to the setup, instead masses are attached to the canard shaft, whose mass moment of inertia is equal to that of canards.

PC is going to be used for data collection and for driving and controlling the solenoid valves. For this purpose, data acquisition cards are integrated to the PC. One is a 12 channel relay actuator card, the other one has 8 differential or 16

single-ended analog input channels, 2 analog output channels and 24 channel digital I/O.

There are two different angle measurement devices in the test setup; one is an arc segment potentiometer, the other one is a Heidenhein absolute encoder with model number ROC 417. Arc segment potentiometer is especially used in fin or wing actuators, because the maximum angle that fin can rotate is limited to some angle, therefore a potentiometer that can measure a full rotation is not required for these types of applications, indeed arc segment potentiometer is preferred in order to save space. Encoder is both incremental and absolute encoder. Its incremental measurement has the resolution of 8192 lines. Absolute output of the encoder is a digital signal in EnDAT format with 17 bit resolution. An electronic card is designed to be able to read the digital output of the encoder, which is required to convert from EnDAT data format to parallel digital data. Specifications of these measurement devices are given in APPENDIX B.

A block schema showing the connection of these measurement devices and data acquisition cards is given in Figure 3-3. Energizing voltage for the solenoid valves are switched by the relays and relays are controlled by the PC. Analog potentiometer signal is recorded by the 16 bit A/D board and digital encoder signals are recorded by the 24 Channel Digital I/O board. There seems no problem in recording the resulted canard shaft rotation by potentiometer and encoder, but input signal level, which is 24 V, is too high to be collected by 16 bit A/D board. Therefore another input signal with 1 Volt level is switched by the relay card. This

signal is assumed to be input signal, in this way we could include the delay due to
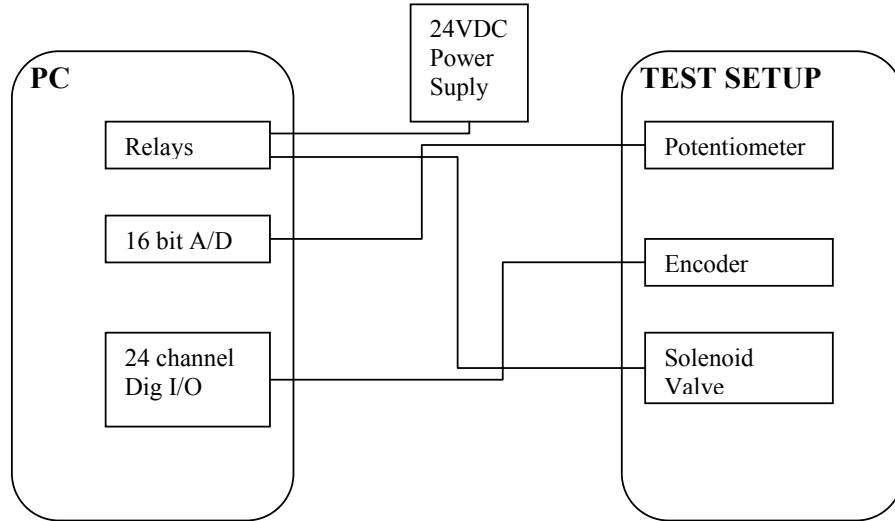
the relay switching action in our input signal.

Figure 3-3 Hardware schematic

3.3 MATLAB System Identification Toolbox

The system identification process starts with recording input and output data and

importing them to the System Identification GUI. Next step is choosing a model

structure from the table given below.

Table 3-1 Available linear model structures in MATLAB

ARX	$A(q^{-1})y = B(q^{-1})u + e$
ARMAX	$A(q^{-1})y = B(q^{-1})u + C(q^{-1})e$
OE (Output Error)	$y = \left(B(q^{-1})/F(q^{-1})\right)u + e$
BJ (Box Jenkins)	$y = \left(B(q^{-1})/F(q^{-1})\right)u + \left(C(q^{-1})/D(q^{-1})\right)e$
State Space	$x_{k+1} = Ax_k + Bu_k + Ke_k$ $y_k = Cx_k + Du_k + e_k$

Here q^{-1} stands for the delay operator ($q^{-1} \equiv z^{-1}$). After choosing a model, order of the polynomials and dead-time must be defined. For example when ARX model is chosen, order of $A(q^{-1})$ and $B(q^{-1})$ polynomials and dead-time (if there is) must be entered in the window. If state space model structure is chosen, only the order of the model is required to be defined. There are several solution algorithms of system identification problem; Least Squares Estimation, Instrumental Variable and Prediction Error Method. Each model is assigned with one or two of these solution methods in MATLAB System Identification Toolbox. (Table 3-2)

Table 3-2 Solution methods of MATLAB System Identification Toolbox

	Least Squares	Instrumental Variable	Prediction Error	Subspace Method (n4sid)
ARX	⛶	⛶		
ARMAX			⛶	
OE			⛶	
BJ			⛶	
State Space			⛶	⛶

3.4 System Identification for CAS

In system identification, it is important to have an input signal which can excite all the modes of the system. Such a signal could be white noise, sums of sinusoids, pseudorandom binary signal and etc. In our experiment, input signal has to be a two level signal because valves are either closed or open, therefore pseudorandom binary signal is the only choice to be applied. Pseudo random binary signal may be obtained by the expression given below.

$$u(t) = \frac{1}{2}(u_1 + u_2) + \frac{1}{2}(u_1 - u_2)\text{sign}(w(t))$$ (3.1)

where u_1 and u_2 are the input signal levels, and $w(t)$ is computer generated white noise.

System identification studies are carried out using two different schemes. In the first one, pseudo random binary signal is generated in the PC, whose level is between 0 and 1 (means relay open and relay close), and this signal applied to the neg-solenoid valve and pos-solenoid valve separately. For example in the neg-solenoid valve using case, canard shaft moves between 0 degrees and approximately -10 degrees and in the pos-solenoid valve case, it moves between 0 degrees and approximately 10 degrees.

In the second one, the level of the binary signal changes from -1 to 1. Here -1 means energizing the neg-solenoid valve, while the pos-solenoid valve is in normal position and +1 means energizing the pos-solenoid valve, while the neg-solenoid valve is brought to normal position. With this scheme, canard shaft deflection changes from approximately -10 degrees to +10 degrees. These two schemes will make two different pulse width modulation schemes possible, which are explained in the fourth chapter.

3.4.1 System Identification Using First Scheme

Input-output data is collected with sampling frequency 1 kHz for both negative and positive shaft rotations for 2 minutes. The first 1 minute duration data is used for identification, the last 1 minute data is used for validation. First 45 second of this data set is given in figure for positive shaft rotation.
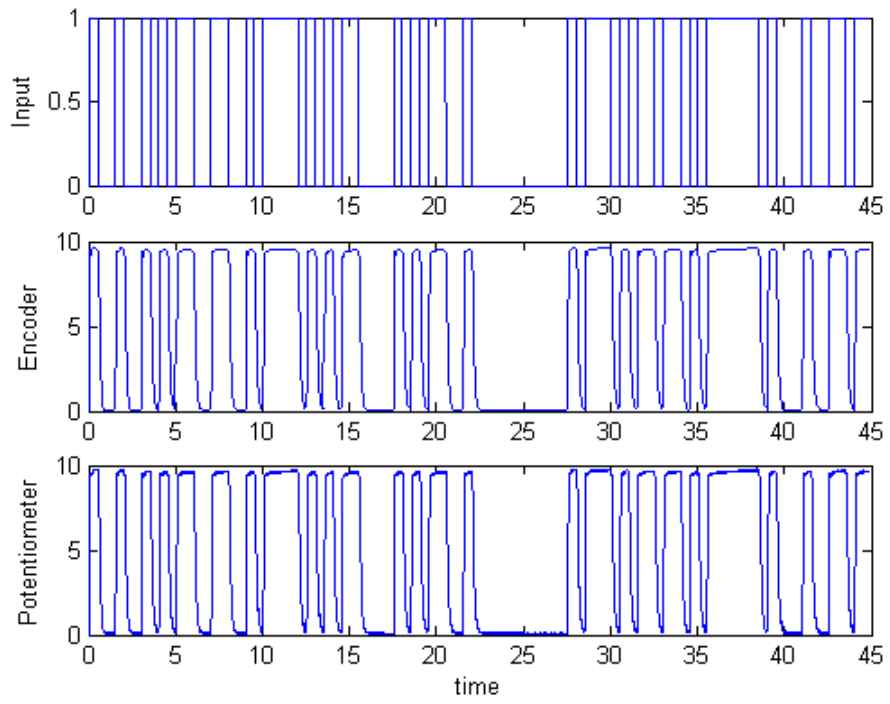
Figure 3-4 Some portion of input-output data set for positive deflection
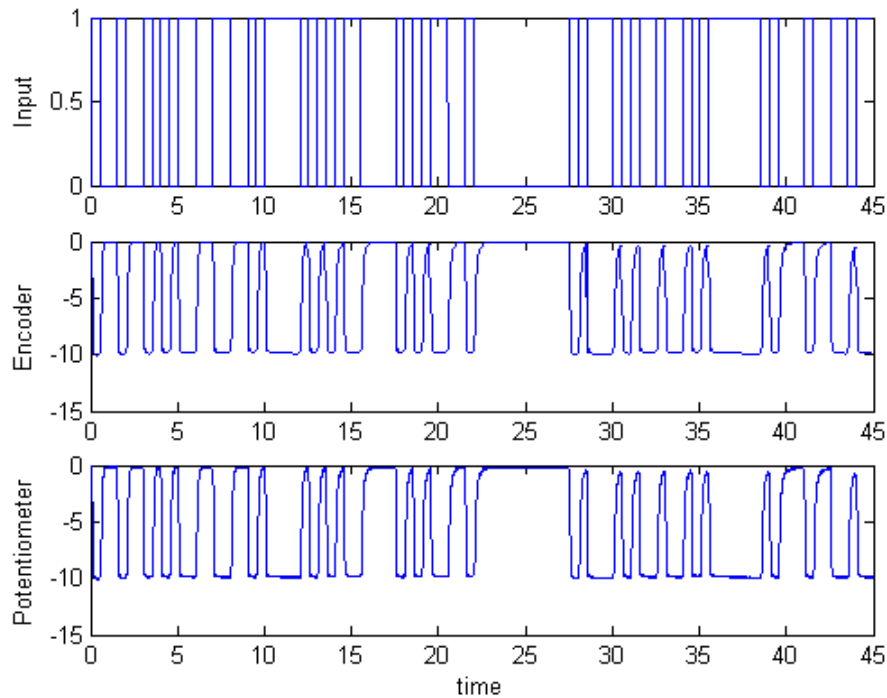
Figure 3-5 Some portion of input-output data set for negative deflection

When we look at the data, it is seen that negative shaft rotation settles at a point which is a little greater than the positive shaft rotation. This shows that estimating two separate models for positive and negative shaft deflections could be more realistic than estimating one model for whole range of deflections. When the data sets are examined more closely, it is seen that shaft settles to 9.5 (or -9.8) degrees with an abrupt change at around 9.2 (or -9.5) degrees, after this point, shaft motion becomes smoother, whereas the behavior from full deflection to 0 degrees doesn't have such a change, which indicates the nonlinearity in the system. Moreover when the plot is zoomed further, it is seen that there are wiggles at the place where this abrupt change occurs. All these behaviors can be seen in Figure 3-6. As a summary

if we interpret this motion, it seems that at the beginning the effect of damper on the shaft motion is quite less, but after some deflection angle, it damps the motion very quickly. Although it seems very difficult to fit a linear model to these data sets showing this much of non-linearity, this is the only choice unless a parametric nonlinear dynamic model is built.
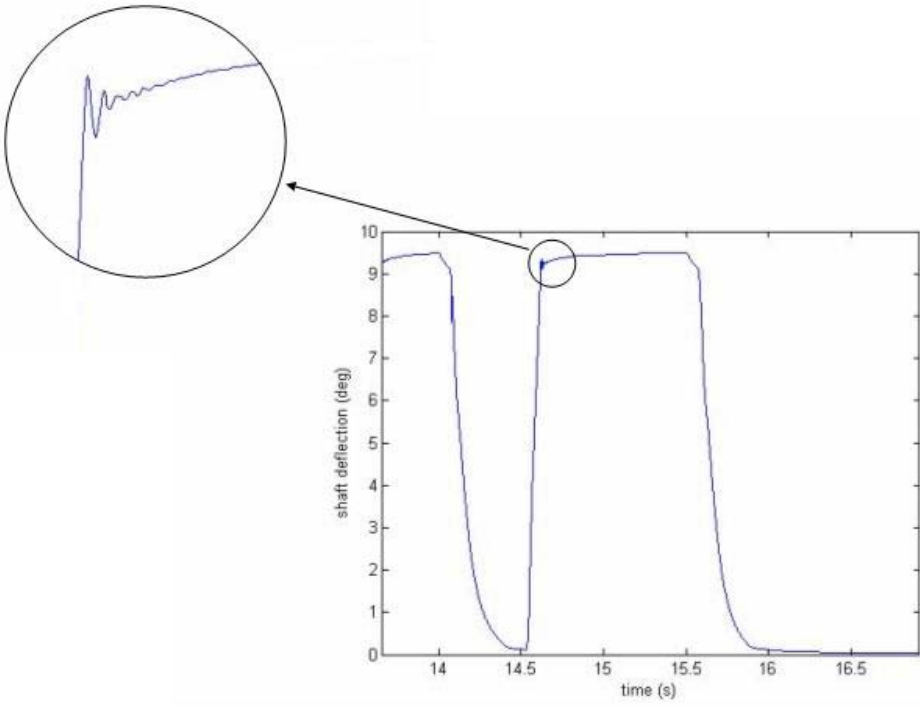
Figure 3-6 Zoomed view of the output data

The order of the model structure can be at most 4, because moving mass (plunger) in the solenoid valve and rotating shaft can be modeled as second order systems. However we have no idea about the number of zeros and delays. In order to decide

on which model is any good, the fitness of the estimated model simulations to the validation data must be considered. In the light of this information, many models are estimated but, bests among them are given in Table 3-4 and Table 3-5, and models are estimated by using Box-Jenkins model structure because it can describe a system more freely. Box-Jenkins model structure is given below.

$$y(t) = \frac{B(q)}{F(q)} u(t - n_k) + \frac{C(q)}{D(q)} e(t)$$ (3.2)

Table 3-3 Box-Jenkins model structures used in the estimations

Model Name	Model Structures
Mod1	$y(t) = \dfrac{b_1 + b_2 q^{-1} + b_3 q^{-2} + b_4 q^{-3}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3}} u(t) + \dfrac{1}{1 + d_1 q^{-1}} e(t)$
Mod2	$y(t) = \dfrac{b_1 + b_2 q^{-1} + b_3 q^{-2}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3}} u(t) + \dfrac{1}{1 + d_1 q^{-1}} e(t)$
Mod3	$y(t) = \dfrac{b_1 + b_2 q^{-1}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3} + f_4 q^{-4}} u(t) + \dfrac{1}{1 + d_1 q^{-1}} e(t)$

Table 3-4 Estimated model transfer functions from positive shaft rotation data

set

Mod1	Pot. Data	$\dfrac{0.0019509(s+4292)(s^2+97.06s+3.596\cdot10^4)}{(s+11.53)(s^2+39.59s+2762)}$
	Encoder Data	$\dfrac{0.004183(s+2309)(s^2-122s+5.619\cdot10^4)}{(s+13.38)(s^2+45.24s+4263)}$
Mod2	Pot. Data	$\dfrac{0.0076361(s+1045)(s^2-12.78s+4.21\cdot10^4)}{(s+11.71)(s^2+40.08s+3023)}$
	Encoder Data	$\dfrac{0.0076361(s+1106)(s^2-139.3s+4.987\cdot10^4)}{(s+12.71)(s^2+45.64s+4176)}$
Mod3	Pot. Data	$\dfrac{0.00051537(s+727.7)(s+5.913)(s^2+809.5s+8.491\cdot10^5)}{(s+11.62)(s+5.913)(s^2+40.54s+2877)}$
	Encoder Data	$\dfrac{0.00052444(s+875.4)(s+25.18)(s^2+1006s+1.18\cdot10^6)}{(s+29.28)(s+12.96)(s^2+47.31s+3778)}$

Table 3-5 Estimated model transfer functions from negative shaft rotation

data set

Mod1	Pot. Data	$$\frac{-0.0022485(s+1.053\cdot10^4)(s^2+9.074s+1.199\cdot10^4)}{(s+12.51)(s^2+38.12s+2377)}$$
	Encoder Data	$$\frac{-0.010159(s+0.5898)(s^2+230.9s+4.837\cdot10^5)}{(s+0.5898)(s^2+41.47s+529.2)}$$
Mod2	Pot. Data	$$\frac{0.0098938(s+1069)(s-482)(s+2.249)}{(s+2.248)(s^2+42.86s+554.6)}$$
	Encoder Data	$$\frac{-0.01282(s+1065)(s^2-48.41s+3.247\cdot10^4)}{(s+14.08)(s^2+51.36s+3212)}$$
Mod3	Pot. Data	$$\frac{-0.0021936(s+209.2)(s+74.46)(s^2+921.6s+4.917\cdot10^5)}{(s+74.42)(s+12.1)(s^2+39.1s+1956)}$$
	Encoder Data	$$\frac{-0.0031143(s+154.6)(s+25.4)(s^2+942.2s+4.858\cdot10^5)}{(s+25.39)(s+13.88)(s^2+37.02s+1751)}$$

Models estimated by MATLAB System Identification Toolbox are all discrete time but we preferred to interpret the results in continuous time. Therefore, before tabulating the results, estimated models are converted to continuous time with *d2c* MATLAB function. In other words, continuous time models are only used to be presented in these tables, and in controller design process for example, discrete time models are used.

As can be seen from the Table 3-4 and Table 3-5, most of the estimated models have a real negative pole around 12 and a couple of complex poles with $\omega_n \approx 65$ and $\zeta \approx 0.3$. Moreover when we look at the validation data fitness, which is given in Figure 3-7 and Figure 3-8, for positive rotations, there seems to be no difference between all the estimated models, they all fit the validation data with fitness around 77.5 %. However for negative rotation validation data set, the best fitted model is "Mod2enc". Percentage of fitness shown on the right hand side of the figure is evaluated by the following formula.

$$\text{fit} = 100 \frac{1 - \left\| y_{act} - y_{est} \right\|}{\left\| y_{est} - \text{mean}(y) \right\|} \tag{3.3}$$
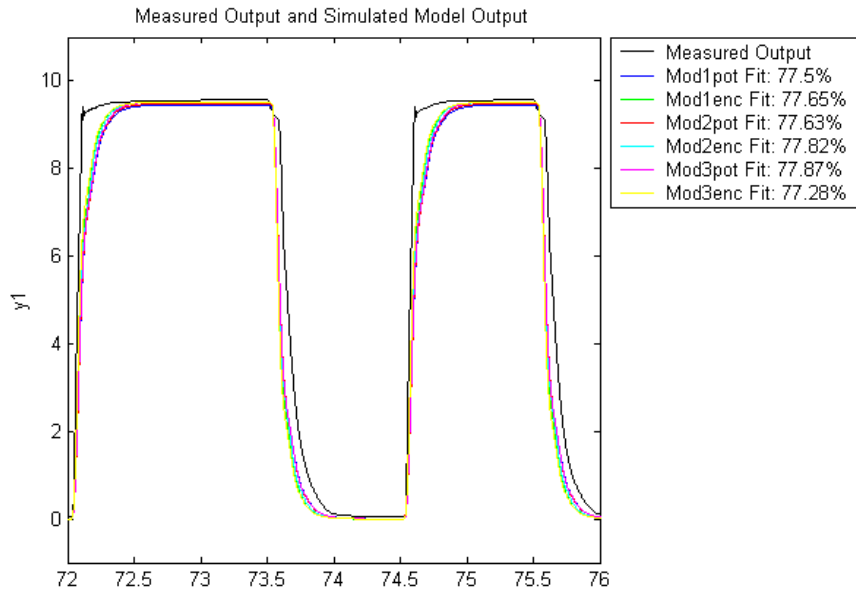
Figure 3-7 Positive rotation validation data fitness plot
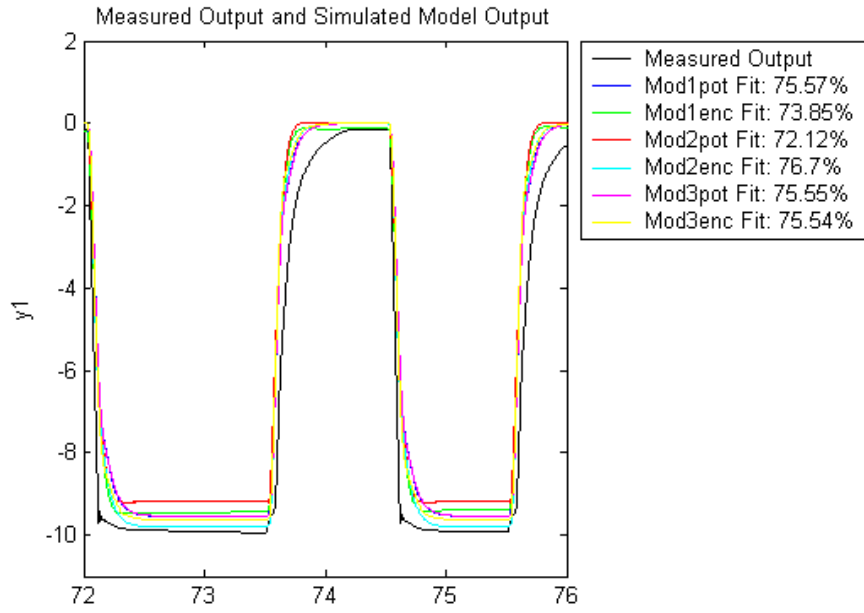
Figure 3-8 Negative rotation validation data fitness plot

As a result for positive and negative rotation models, "Mod2enc" models are chosen, whose continuous and discrete time transfer functions are given below.

For positive rotations:

$$G_{pos}(s) = \frac{0.0076361(s+1106)(s^2 - 139.3s + 4.987 \cdot 10^4)}{(s+12.71)(s^2 + 45.64s + 4176)}$$ (3.4)

$$G_{pos}(z) = \frac{0.0091481z(z^2 - 2.101z + 1.155)}{(z-0.9874)(z^2 - 1.951z + 0.9554)}$$ (3.5)

For negative rotations:

$$G_{neg}(s) = \frac{-0.01282(s+1065)(s^2 - 48.41s + 3.247 \cdot 10^4)}{(s+14.08)(s^2 + 51.36s + 3212)}$$ (3.6)

$$G_{neg}(z) = \frac{-0.01282z(z^2 - 2.018z + 1.052)}{(z-0.986)(z^2 - 1.947z + 0.9499)}$$ (3.7)

3.4.2 System Identification Using Second Scheme

Input-output data is collected with sampling frequency 1 kHz for 2 minutes. The first 1 minute duration data is used for identification, the last 1 minute data is used for validation. First 45 second of this data set is given in figure below.

Figure 3-9 Some portion of input-output data set
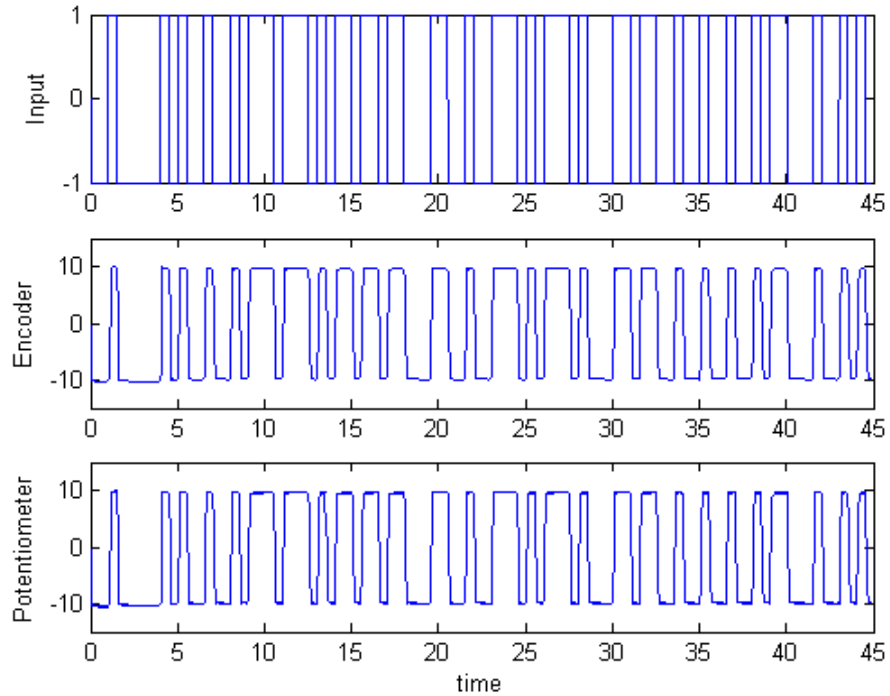
For this data set, there is no difference in the settling characteristic in positive full deflection angle and the settling characteristic in negative full deflection angle, which was not the case in the first scheme. Therefore it is expected to better fit a model using this data set. Again Box-Jenkins model structures given in Table 3-6 are used in the model structures.

Table 3-6 Box-Jenkins model structures used in the estimations

Model Name	Model Structures
Mod1	$y(t) = \dfrac{b_1 + b_2 q^{-1} + b_3 q^{-2} + b_4 q^{-3}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3}} u(t) + \dfrac{1}{1 + d_1 q^{-1}} e(t)$
Mod2	$y(t) = \dfrac{b_1 + b_2 q^{-1} + b_3 q^{-2}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3}} u(t) + \dfrac{1}{1 + d_1 q^{-1}} e(t)$
Mod3	$y(t) = \dfrac{b_1 + b_2 q^{-1} + b_3 q^{-2}}{1 + f_1 q^{-1} + f_2 q^{-2} + f_3 q^{-3} + f_4 q^{-4}} u(t)$

Table 3-7 Estimated model transfer functions

Mod1	Pot. Data	$\dfrac{0.00081026(s - 34980)(s - 499.6)(s + 45.55)}{(s + 203.7)(s^2 + 26.02s + 334.7)}$
	Encoder Data	$\dfrac{0.0043903(s + 4334)(s + 191.1)(s + 9.264)}{(s + 7.724)(s^2 + 28.11s + 451.5)}$
Mod2	Pot. Data	$\dfrac{0.013658(s + 745.8)(s + 417.7)(s + 0.01013)}{(s + 0.01013)(s^2 + 32.21s + 445.4)}$
	Encoder Data	$\dfrac{0.0045734(s + 195.4)(s^2 + 1239s + 995700)}{(s + 196.3)(s^2 + 36.07s + 487.9)}$
Mod3	Pot. Data	$\dfrac{0.0081722(s^2 + 11.87s + 1849)(s^2 + 1550s + 1038000)}{(s^2 + 57.85s + 926.9)(s^2 + 28.47s + 1721)}$
	Encoder Data	$\dfrac{0.0084893(s^2 + 11.5s + 1932)(s^2 + 1550s + 1038000)}{(s^2 + 57.57s + 919.9)(s^2 + 29.38s + 1886)}$

40

The estimated models are given in Table 3-7. From the table it can be said that, estimated models from potentiometer data are not identical with the estimated models from encoder except for "Mod3". This model has a fourth order dynamics and it also fits the validation data best, which is given below in Figure 3-10. Therefore this model is chosen and its continuous-time and discrete-time transfer functions are given in Equation (3.8) and (3.9).
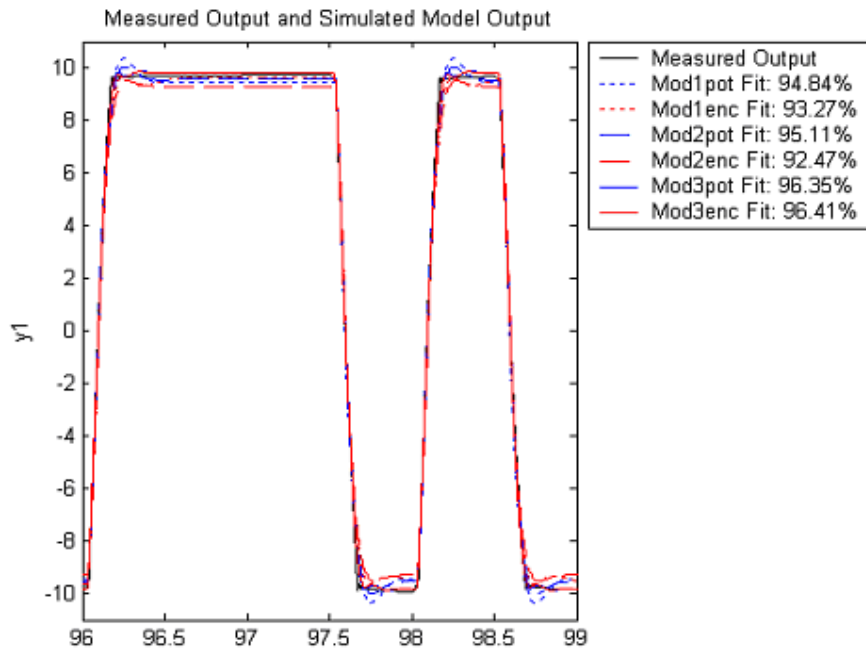

Figure 3-10 Validation data fitness plot

$$G(s) = \frac{0.0084893(s^2 + 11.5s + 1932)(s^2 + 1550s + 1.038 \cdot 10^6)}{(s^2 + 57.57s + 919.9)(s^2 + 29.38s + 1886)} \qquad (3.8)$$

$$G(z) = \frac{0.0084893z^2(z^2 - 1.987z + 0.9886)}{(z^2 - 1.943z + 0.9441)(z^2 - 1.969z + 0.971)} \qquad (3.9)$$

CHAPTER 4

CONTROLLER DESIGN WITH PWM FOR CONTROL ACTUATION SYSTEM

4.1 Introduction

After we estimate control actuation system models, a controller is required to be designed. But although the mathematical model of CAS can take values different from 1, it is known that physically it doesn't take such values. Therefore controller output should be composed of two values, which corresponds to energized solenoid valve and normal solenoid valve situations. The most famous solution for this type of situations is "Pulse Width Modulation", which means adjusting the width of the "energize the solenoid valve" signal according to the magnitude of the controller signal.

This chapter starts with the definition of PWM and continues with description of controller and simulation results of the closed loop control actuation system.

4.2 Pulse Width Modulation

"Pulse Width Modulation" has extensively been used in electronic and electrical systems including attitude control systems, adaptive control systems, signal processing, power control systems, modelling of neuron behaviour, etc. The most well-known modern application is the attitude control of satellites and space vehicles. One advantage of PWM control is the simplicity of its realization: the control variable assumes only two or three values, say +1, -1 and 0, and hence, the control action is realized through the operation of a switch [16]. A PWM signal is on for part of its period, and off for the rest. Examples are given below in Figure 4-1.
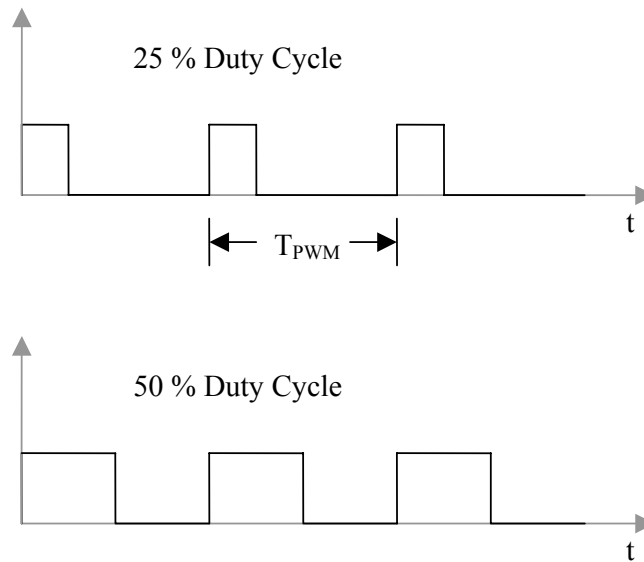
Figure 4-1 PWM signal examples

The duty cycle refers to the percentage of the period for which the signal is on. The duty cycle can be anywhere from 0, the signal is always off, to 1, where the signal is constantly on. T_{PWM} is called modulating period.

When using on-off solenoid valves to control the deflection of the shaft, the controller output must be resolved into the individual pulsing of neg-solenoid valve and pos-solenoid valve. Many researchers have used a linear pulse width modulation algorithm to control solenoid valves for pneumatic and hydraulic actuators. Van Varseveld and Bone (1997) have developed a PWM algorithm which results in linear model description for their applications. But in their test setup there is no other component except for the opposing valve that can also resist the motion, only when the other valve is energized the opposing force is created. While in our case, there is always opposing force created by the damper, therefore with the help of their study, we have developed our two different PWM schemes. The first one makes use of damper to go forth and back around the commanded deflection. For example, if we command 7 degrees to CAS, it uses the gas pressure to go to this angle if its initial position is less than 7, but if it is greater than 7, then it releases the gas and do not fill the other cylinder, and it still goes back to 7, because there is spring that pulls the shaft to neutral position, which is 0. Therefore we can hold the shaft at 7 degrees position by using only one valve, say pos-solenoid valve in this example, and spring in damper. In the other case, we can forget about the spring, and we can fill the other cylinder to go back to 7, when the initial position of the shaft is greater than 7.

44

In designing controller, these two schemes should use two different mathematical models, because in the first one pulling force is created only by spring, while in the second one it is created by the opposing valve, which is expected to result in faster response. Hence mathematical model estimated using the first scheme in the previous chapter is suitable to be used with first PWM scheme, and the other model estimated from the second scheme is suitable to be used with the second PWM scheme. In both cases, there will be linear relationship between the duty cycle and the controller output, which are shown in Figure 4-2 and Figure 4-3.
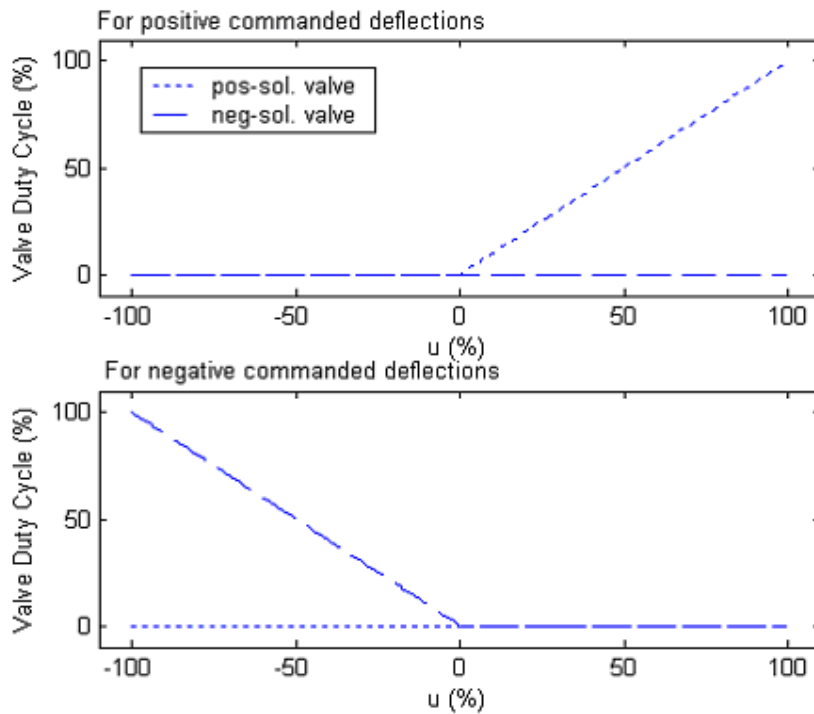

Figure 4-2 First PWM scheme
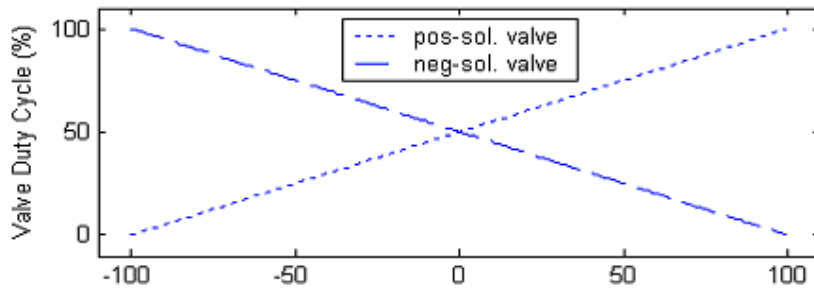
Figure 4-3 Second PWM scheme

4.3 Controller with PWM

The block diagram of the closed loop CAS is given in Figure 4-4. Since PWM has its own modulating frequency, controller can be designed at most at the modulating frequency. Moreover, the higher the modulating frequency, the more successful the closed loop dynamics. However there is always a limit due to the physical constraints, therefore 100 Hz discrete time controller is decided to be designed for each PWM scheme, which are explained in the following two sub-sections.

Figure 4-4 Closed loop CAS block diagram

4.3.1 Controller for First PWM Scheme

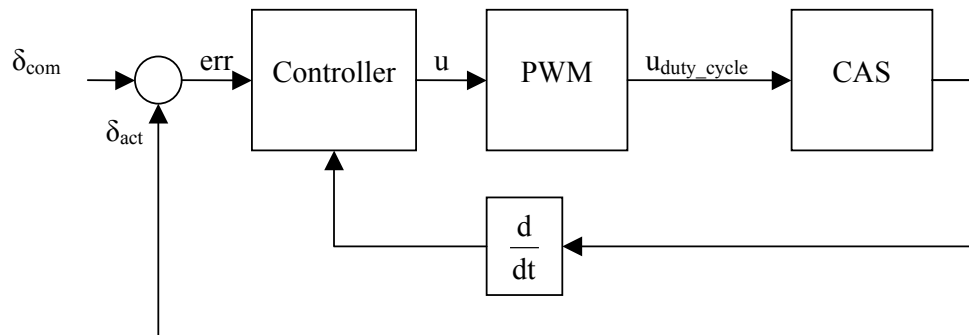Controller structure is chosen to be PI with rate feedback, since taking derivative from a digital position sensor can be said to be noiseless. As it is mentioned before in this scheme, mathematical model of CAS is the one that is given in Section 3.4.1. For convenience, its discrete-time mathematical model is repeated below.

For positive rotations

$$G_{pos}(z) = \frac{0.0091481z(z^2 - 2.101z + 1.155)}{(z - 0.9874)(z^2 - 1.951z + 0.9554)}$$ (4.1)

For negative rotations

$$G_{neg}(z) = \frac{-0.01282z(z^2 - 2.018z + 1.052)}{(z - 0.986)(z^2 - 1.947z + 0.9499)}$$ (4.2)

As can be seen from the transfer functions, the magnitude of the poles, the zeros and the gains are very close to each other. The sign of them are also the same in two transfer functions, except for gains. Therefore reducing these transfer functions into single transfer function for controller design purposes will not create a problem, as long as we show that controller, which is designed for the single transfer function, works well with these two transfer functions.

Reducing the model into single transfer function is made by averaging the poles, zeros and absolute value of the gains. Single transfer function is given below

$$G_{mean}(z) = \frac{0.010984z(z^2 - 2.06z + 1.102)}{(z - 0.9867)(z^2 - 1.949z + 0.9526)}$$ (4.3)

Controller for this single transfer function is designed by trail and error process. Figure 4-5 shows the closed loop model built in MATLAB Simulink and Figure 4-6 shows inside of the designed controller.
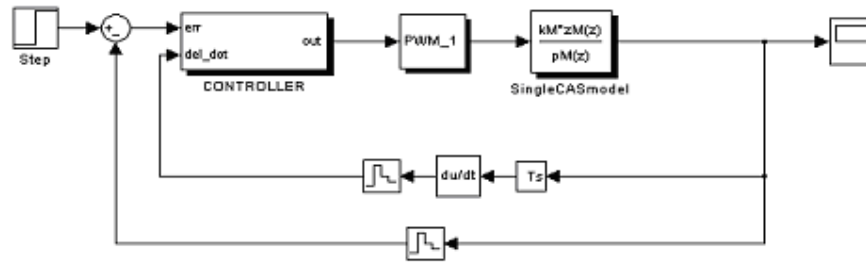
Figure 4-5 Closed loop CAS model built in MATLAB
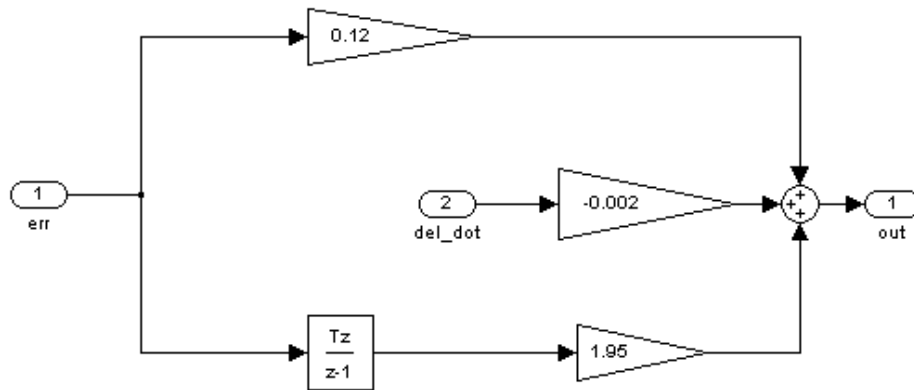

Figure 4-6 Inside of MATLAB controller block for first PWM scheme

Below figures show the response of this closed loop CAS model to step inputs with different magnitudes and sine inputs.

Figure 4-7 Model response to step input with magnitude 8

Figure 4-8 Model response to step input with magnitude 5

Figure 4-9 Model response to step input with magnitude 2
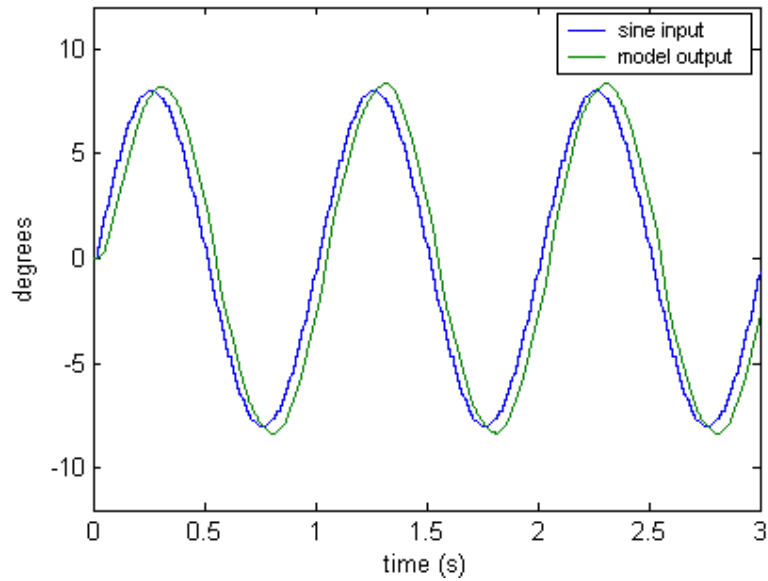
Figure 4-10 Response of the model to the sine input with amplitude 8 and

frequency 1 Hz

In the beginning of this section it is said that a single model transfer function is going to be used and a controller is designed for the model containing this transfer function and it is added that designed controller will be confirmed to be working with the actual estimated transfer functions. Figure 4-11 and Figure 4-12 show the step responses of closed loop CAS models containing both, $G_{mean}(z)$, $G_{pos}(z)$ and $G_{neg}(z)$.
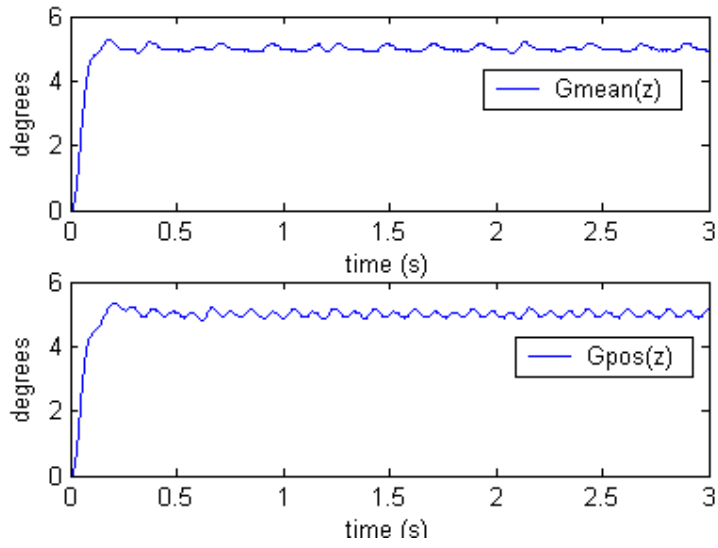
Figure 4-11 Step responses of closed loop models containing G_{pos}(z) and

G_{mean}(z)
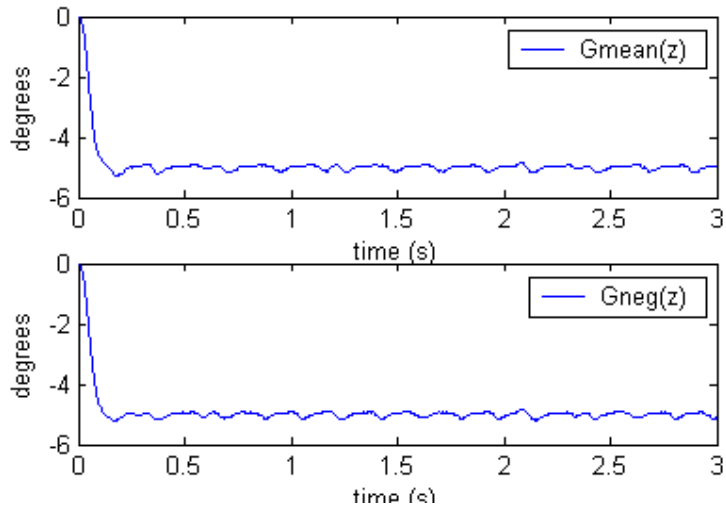
Figure 4-12 Step responses of closed loop models containing G_{neg}(z) and

G_{mean}(z)

As can be seen from these plots, we can say that designed controller works well with the actual estimated models too. Hence in the non-linear flight simulations, $G_{mean}(z)$ is going to be used.

4.3.2 Controller for Second PWM Scheme

The same controller structure, that is PI with rate feedback, is used in the second PWM scheme too. In this PWM scheme mathematical model of the CAS is the one that is given in Section 3.4.2., because this PWM scheme uses both pos-solenoid and neg-solenoid valves together. This mathematical model is repeated below.

$$G(z) = \frac{0.0084893z^2(z^2 - 1.987z + 0.9886)}{(z^2 - 1.943z + 0.9441)(z^2 - 1.969z + 0.971)} \qquad (4.4)$$

Controller is designed by trial and error process, which is built in MATLAB and shown below
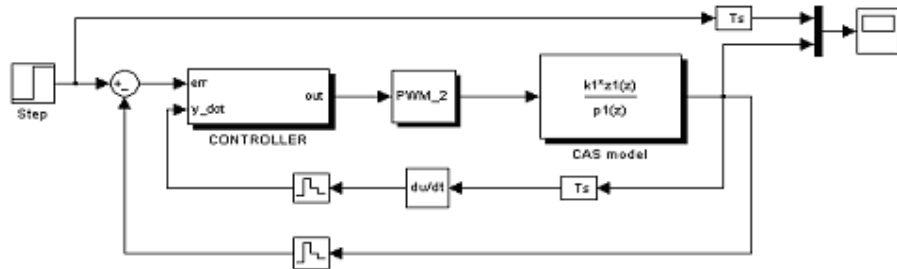
Figure 4-13 Closed loop CAS model built in MATLAB
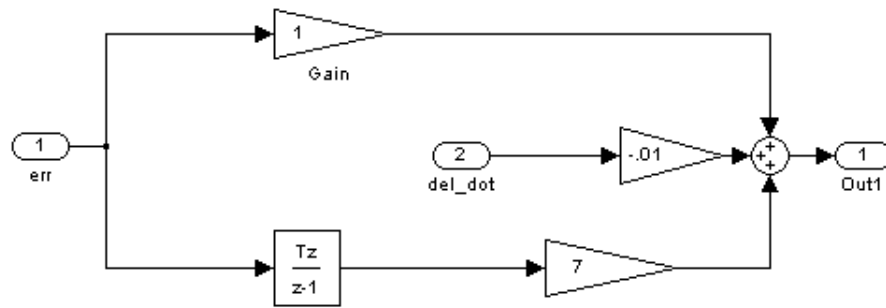
Figure 4-14 Inside of MATLAB controller block for second PWM scheme

Below figures show the response of this closed loop CAS model to step inputs with different magnitudes and sine inputs.
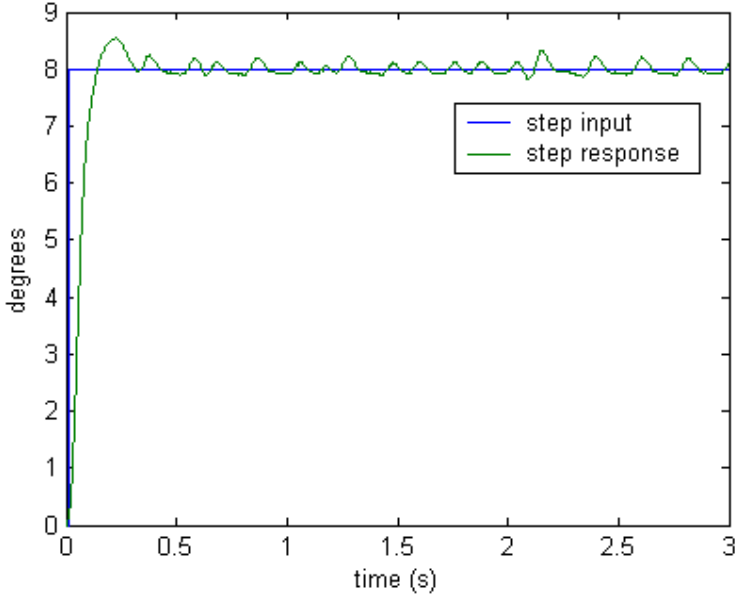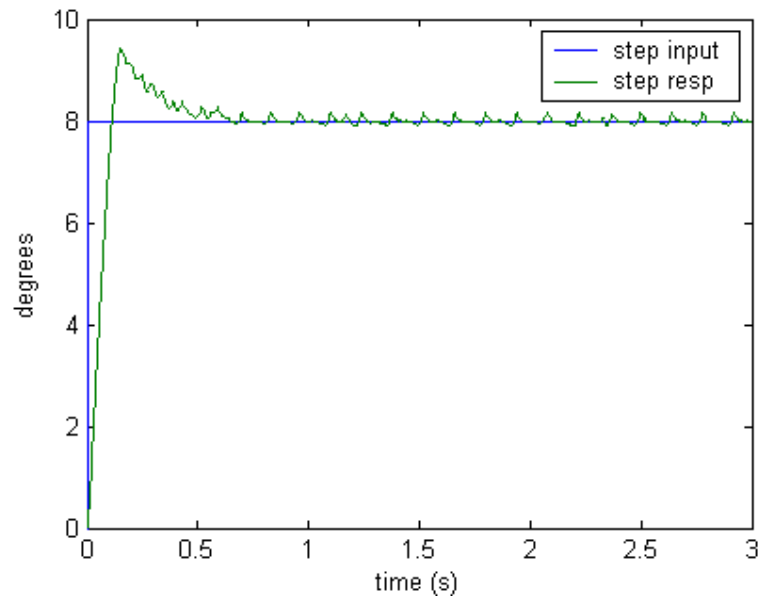

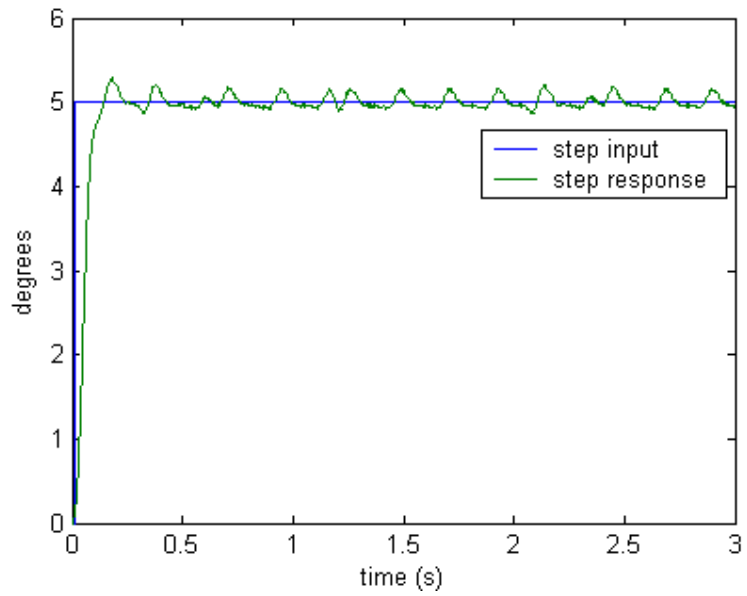**Figure 4-15 Model response to step input with magnitude 8**
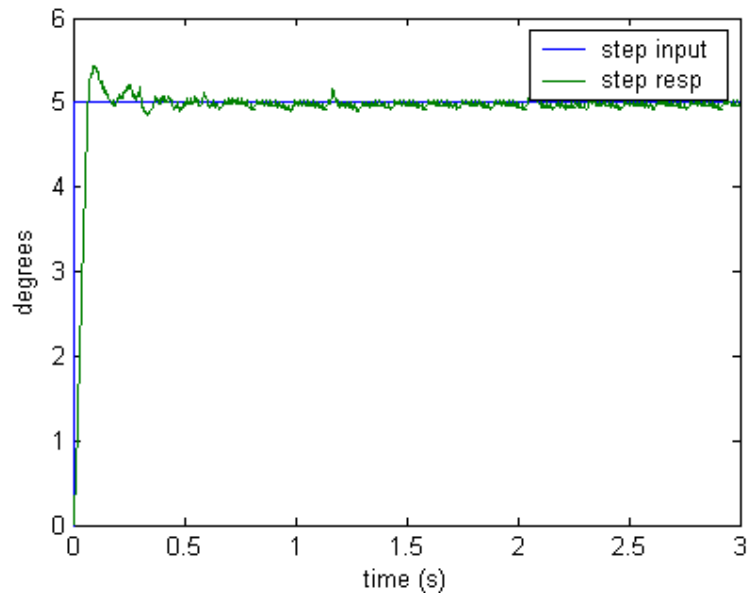
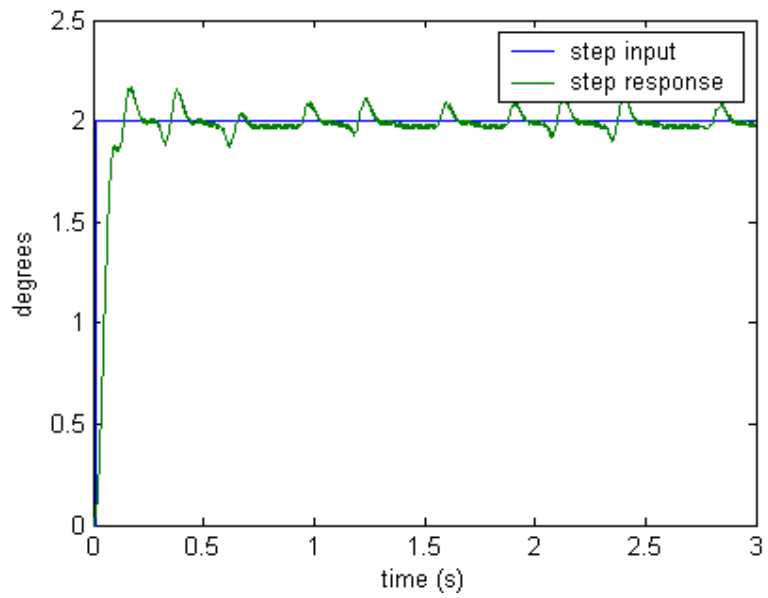**Figure 4-16 Model response to step input with magnitude 5**
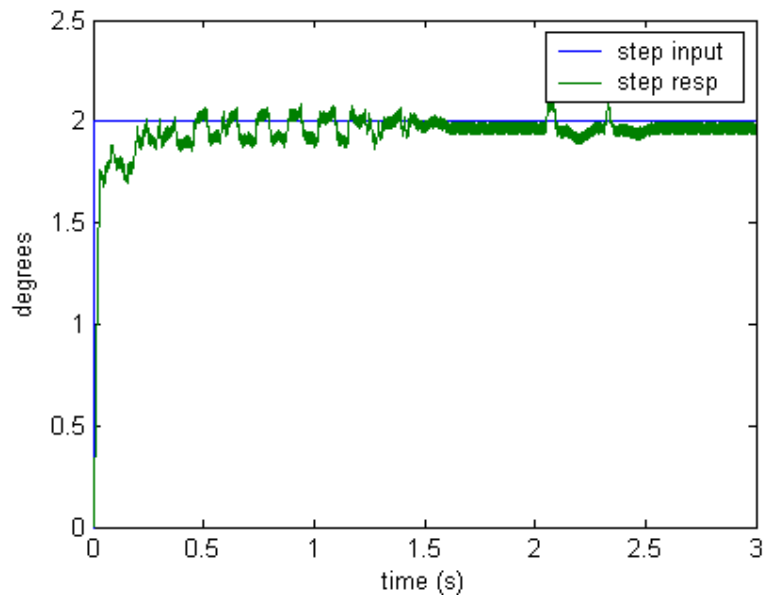
**Figure 4-17 Model response to step input with magnitude 2**
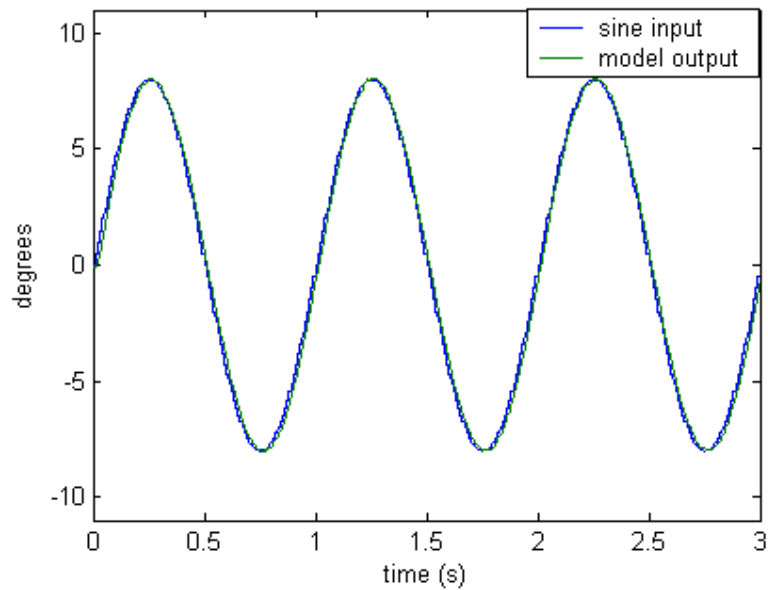
Figure 4-18 Response of the model to the sine input with amplitude 8 and

frequency 1 Hz

Although step responses of the closed loop model seem to have large overshoot when compared with the closed loop model of the previous PWM scheme, its response to 1 Hz frequency sine input is much better. Because estimated CAS model in the second PWM scheme has rather fast dynamics when compared with the model used in the first PWM scheme.

CHAPTER 5

SEEKER MODEL

5.1 Introduction

Seeker is the part of a missile that is used for detection and tracking of the target, and ASGM (Air to Surface Guided Munition) use laser seekers to get the relative position information about the target with respect to the munition. In semi active guidance system, target is illuminated by a laser source and the seeker, which then guides the weapon to the target, detects reflected laser beam. To increase the field of view, the gimbaled type seeker is used, which looks in the direction of the velocity vector. Figure showing this concept is given below.
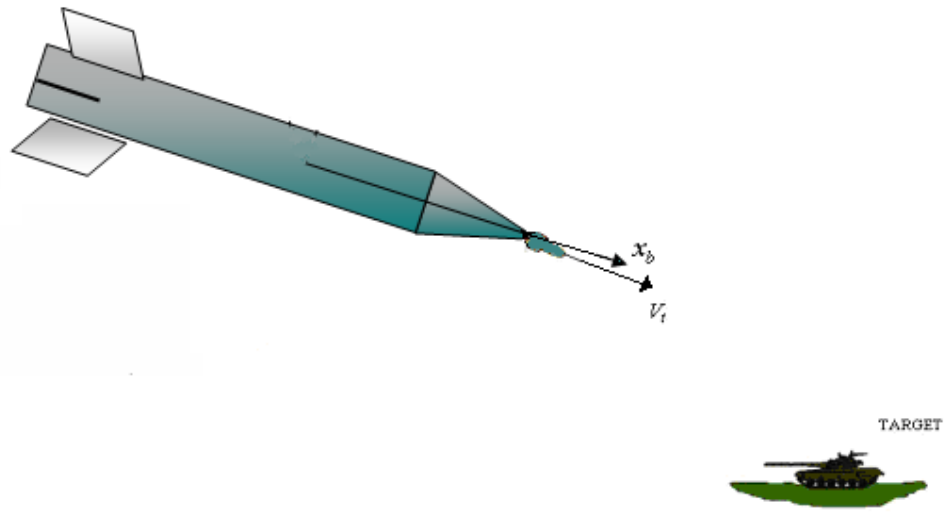
Figure 5-1 ASGM seeker concept

Laser seeker is a combination of optical, electronic and mechanical components and the detector is the heart of the seeker system because it converts scene radiation into measurable electrical signal. Four quadrant detectors have four distinct photosensitive elements separated by a minuscule gap. A light spot illuminating just one element produces photocurrent only in that element. When the light traverses across the surface of the detector, photocurrent produced from one quadrant, where the light is directed, starts to decrease as that of the opposite quadrant starts to increase. The main disadvantage of this type of detectors is that detector only provides position information when all 4 quadrants are illuminated by the light; otherwise laser spot is formed on only one quadrant, which means that location of the spot on this quadrant is not exactly known. To avoid this, spot

diameter should be at least equal to the detector radius, one of the ways of achieving this spot size is defocusing.

We are going to generate a model for seeker, which processes the detector output and extract the angle information from it. Processing the detector outputs, which are 4 voltage values corresponding to each quadrant, to obtain the laser spot location is the first step in the modeling studies. Second step is estimating the error angle, whose definition is made in the next section, using spot location information and some additional information about the configuration of detector and lens. These can be better understood from Figure 5-2. Error angle estimation can be obtained from geometry and optics, whereas spot location estimation, from detector voltages, requires to be formed a more difficult relationship. In this chapter, first we are going to explain how the error angle is estimated, once we know the laser spot center location. Next we are going to give the usual and most frequently used method of spot center estimation. After this, spot center estimation using artificial neural network is introduced and explained. Finally results of these two methods will be examined.
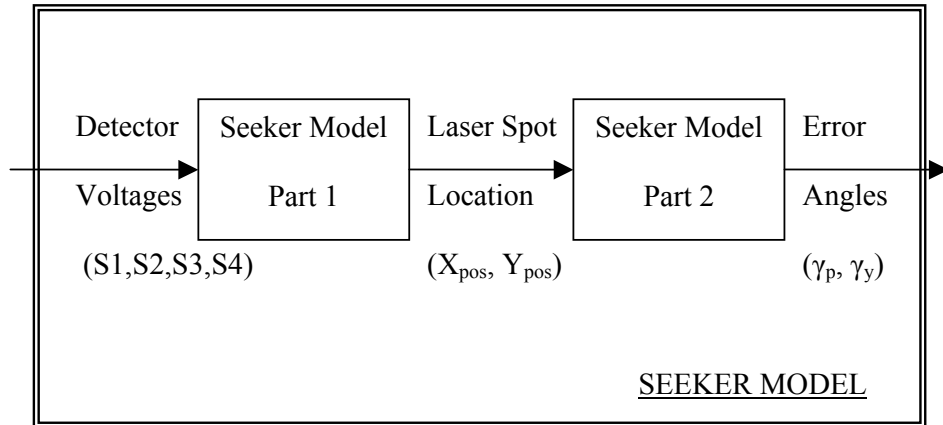
Figure 5-2 Seeker model block schema

5.2 Relation of the Spot Location with the Error Angle

Error angles are formed between LOS (Line of Sight) vector and velocity vector (because seeker looks in the direction of velocity) in two planes namely pitch and yaw planes. Let us call these angles γ_p and γ_Y, for pitch and yaw planes, respectively. Estimation of these angles is based on the fact that, ray of light passing the lens from its center doesn't bend. Moreover it forms the center of the laser spot. Hence if we know the distance of detector to the lens and if we can estimate the spot center location, we can estimate the error angles by the following expressions. This is illustrated in Figure 5-3 for pitch plane only.

$$\gamma_p = a\tan\left(\frac{\Delta y}{d}\right) \qquad (5.1)$$

60

$$\gamma_y = a \tan\left(\frac{\Delta x}{d}\right) \qquad\qquad\qquad (\,5.2\,)$$
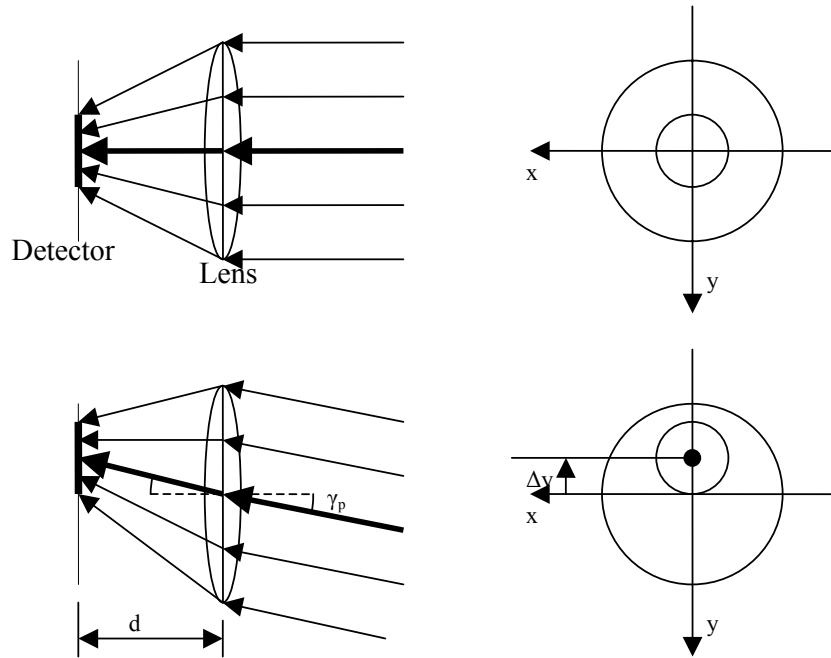
Figure 5-3 Lens and detector configuration and pitch plane error angle

5.3 Estimation of Spot Location

First of all we assume that laser spot is circular with radius r being equal to half of the radius of the detector. Next we assume that 4 area values can be used instead of 4 voltage values, which are calculated using geometry once the spot center location is given. Usage of areas instead of voltages doesn't alter the results of estimation methods because both methods use the normalized voltages (or areas).

The location of the center of the spot is determined with the following formula 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 [7]. Figure 5-4 shows the area values if the spot center location is
X_{pos} and Y_{pos}.

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

(5.3)

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

(5.4)
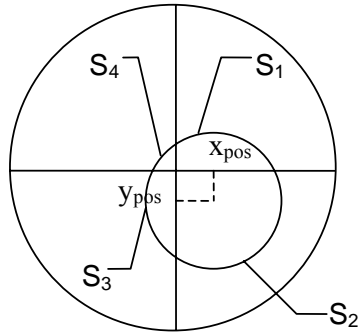
Figure 5-4 Seeker spot position

The problem with this formula is that as the spot center goes far away from the
detector center, the error in spot center estimation increases. A primary aim is to
find a better relationship that works well for all possible spot center points. Also it
would be good to observe the performance of the ASGM under different seeker
models. Therefore an artificial neural network model is suggested, whose structure
is explained in the following sections.

5.3.1 Artificial Neural Networks

Biologists have studied biological neural networks for many years. As information about the functions of the brain was accumulated, a new technology emerged and the quest for an artificial neural network started [5]. Neural Networks are composed of simple elements operating in parallel. As in nature, the network function is determined largely by the connections between elements. We can train a neural network to perform a particular function by adjusting the values of connections between elements [6].

In this thesis we have used radial basis neural network structure with two layers because they are introduced for the solution of the real multivariable interpolation problems like we have here. Our aim is to find a function taking the detector values as input arguments and outputting the spot center location. Below is the schematic of the two layer radial basis network structure.
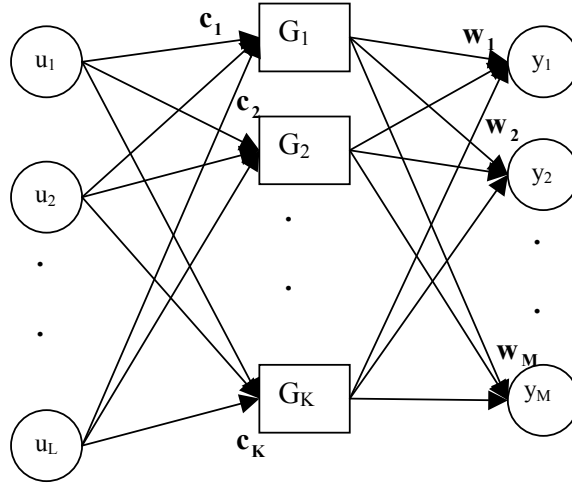
Figure 5-5 General neural network structure

The mathematical expression for the output is as follows;

$$\mathbf{y} = \mathbf{f}(\mathbf{u}) = \mathbf{W}\mathbf{G} \qquad \mathbf{f} : R^L \rightarrow R^M \tag{5.5}$$

$$G_i(\|\mathbf{x} - \mathbf{c_i}\|) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp(-\frac{\|\mathbf{x} - \mathbf{c_i}\|}{2\sigma_i^2}) \tag{5.6}$$

where

$$\mathbf{u} = \begin{bmatrix} u_1 \\ M \\ u_L \end{bmatrix}, \ \mathbf{G} = \begin{bmatrix} G_1(\|\mathbf{x} - \mathbf{c_1}\|) \\ M \\ G_K(\|\mathbf{x} - \mathbf{c_K}\|) \end{bmatrix}, \ \mathbf{y} = \begin{bmatrix} y_1 \\ M \\ y_M \end{bmatrix}, \ \mathbf{W} = \begin{bmatrix} \mathbf{w_1} \\ M \\ \mathbf{w_M} \end{bmatrix}$$

$$\mathbf{w_i} \in R^K \text{ and } \mathbf{c_i} \in R^L$$

The training of radial basis neural network is done using parabolic evolutionary algorithm. The disadvantage of the classical evolutionary algorithms is slow

converges due to its probabilistic characteristics. The Parabolic Evolutionary Algorithm (PEA) (Halıcı U. 1998) is a new approach which speeds up the evolutionary algorithm through a new parabolic approximation. The parabolic approximation is based on the idea that the shape of the fitness functions, when near a maxima, can be modeled as a parabola. If the argument of the cost function was one dimensional, parabola expression is defined by the equation;

$$f(x) = c_2 x^2 + c_1 x + c_0 \qquad (5.7)$$

If three points $(x^a, f(x^a))$, $(x^b, f(x^b))$ and $(x^c, f(x^c))$ are defined, then c_0, c_1 and c_2 can be determined as follows

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 & x^a & (x^a)^2 \\ 1 & x^b & (x^b)^2 \\ 1 & x^c & (x^c)^2 \end{bmatrix} \begin{bmatrix} f(x^a) \\ f(x^b) \\ f(x^c) \end{bmatrix} \qquad (5.8)$$

It has a maximum/minimum at the point satisfying

$$f'(x) = 2c_2 x + c_1 \quad \Rightarrow \quad \mathrm{x}_{max} = -\frac{c_1}{2c_2} \qquad (5.9)$$

But when $x \in R^N$, a parabolid in terms of $\mathbf{x}$ is defined by the equation

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{C_2} \mathbf{x} + \mathbf{c_1}^T \mathbf{x} + c_0 \qquad (5.10)$$

where $\mathbf{C_2}$ is a symmetric matrix of dimension $N \times N$, $\mathbf{c_1} \in R^N$ is a vector and c_0 is a scalar. To determine these coefficient matrices, value of f($\mathbf{x}$) is needed for N(N+1)/2+N+1 different $\mathbf{x}$ vectors. Once $\mathbf{C_2}$, $\mathbf{c_1}$ and c_0 are determined, the point $\mathbf{x}_{max} \in R^N$ maximizing f($\mathbf{x}$) can be found by solving below equation [4].

$$\nabla f(\mathbf{x}_{max}) = 2\mathbf{C_2} \mathbf{x}_{max} + \mathbf{c_1} \qquad (5.11)$$

Parabolic Evolutionary Algorithm is given in APPENDIX A.

5.3.2 Determination of Training Data Set

Input and output of the ANN (Artificial Neural Network) model is 4 area values and, X and Y components of spot center location. 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 success 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 doesn't have to be uniform. Hence, a training data set is created with more data than required for training most probably. In other words, created training data set is possibly overdetermined.
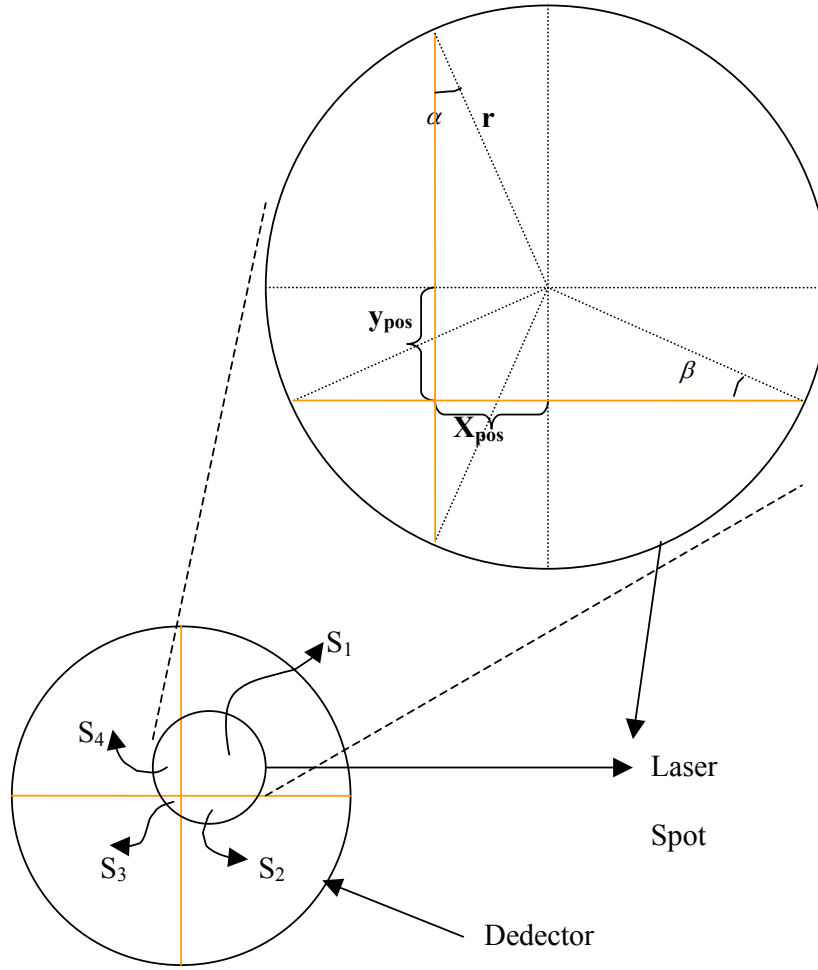
Figure 5-6 Detector spot area schematic

$$S_1 = \pi \frac{r^2}{4} + \frac{r^2}{2} a\sin(x_{pos}/r) + \sqrt{r^2 - x_{pos}^2}\, \frac{x_{pos}}{2} + \frac{r^2}{2} a\sin(y_{pos}/r)$$
$$+ \sqrt{r^2 - y_{pos}^2}\, \frac{y_{pos}}{2} + x_{pos}\, y_{pos} \tag{5.12}$$

$$S_2 = \pi \frac{r^2}{4} + \frac{r^2}{2} a\sin(x_{pos}/r) + \sqrt{r^2 - x_{pos}^2}\, \frac{x_{pos}}{2} - \frac{r^2}{2} a\sin(y_{pos}/r)$$
$$- \sqrt{r^2 - y_{pos}^2}\, \frac{y_{pos}}{2} - x_{pos}\, y_{pos} \tag{5.13}$$

$$S_3 = \pi \frac{r^2}{4} - \frac{r^2}{2} a \sin\left(x_{pos}/r\right) - \sqrt{r^2 - x_{pos}^2} \frac{x_{pos}}{2} - \frac{r^2}{2} a \sin\left(y_{pos}/r\right)$$
$$- \sqrt{r^2 - y_{pos}^2} \frac{y_{pos}}{2} + x_{pos} y_{pos}$$

(5.14)

$$S_4 = \pi \frac{r^2}{4} - \frac{r^2}{2} a \sin\left(x_{pos}/r\right) - \sqrt{r^2 - x_{pos}^2} \frac{x_{pos}}{2} + \frac{r^2}{2} a \sin\left(y_{pos}/r\right)$$
$$+ \sqrt{r^2 - y_{pos}^2} \frac{y_{pos}}{2} - x_{pos} y_{pos}$$

(5.15)

where "r" is the spot radius. Equations given above are obtained using geometry. In the light of above explanations, uniformly spread output points (spot location values) are picked and corresponding input values (4 area values) are estimated. Since the possible spot center locations make a circle with the radius half of the detector radius, 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 221 input-output pair is created in order to train the radial basis neural network.

5.4 Results of the Radial Basis Neural Network Model of Seeker

Neural Network model of the seeker is formed by using two layers and 4 neurons. The schema of the model is given below.
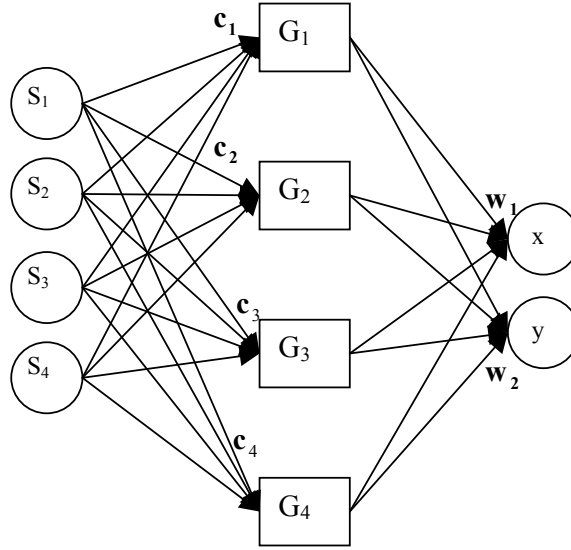
Figure 5-7 Seeker neural network model

In genetic algorithm a fitness function must be defined in order to create the next generation among the best fitted genes. In this study fitness is defined as given in Equation (5.17). Training of the network is done by training data sets explained in the previous chapter. 5250 iterations are performed to minimize the cost, and the algorithm reach the cost value of 0.05 after 2000 iterations and the cost can not be reduced further after this point.

$$\cos t = \sqrt{\frac{\sum_{i=1}^{n}(x_{est}(i) - x_{real}(i))^2 + (y_{est}(i) - y_{real}(i))^2}{n}}$$ (5.16)

$$fitness = 1 - \cos t$$ (5.17)

The plot of the fitness versus iteration number is given in Figure 5-8.

69
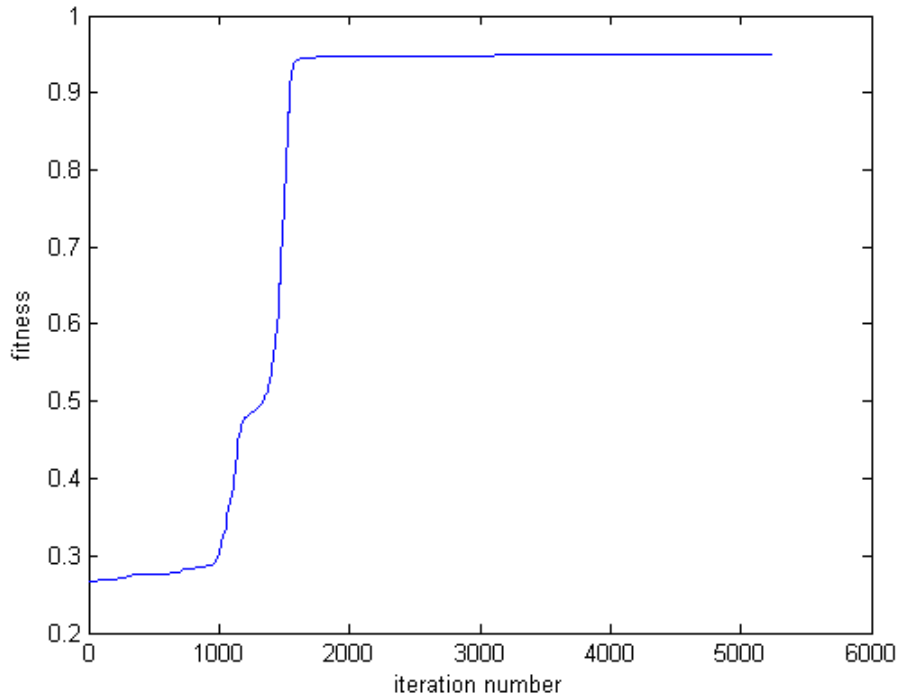
Figure 5-8 Fitness value change during training

In order to see the performance of the ANN model, it is run with input points other than training inputs. Therefore another data set is created using expressions (5.12) – (5.15), which contains training data as well as other input-output pairs. Then the actual output, spot center location X_{pos} and Y_{pos}, is compared with the output obtained by ANN and the other relation given in Equations (5.3) and (5.4). In comparison, output space, which is a circle with radius R, is divided into 5 subspaces by dividing R into 5. The square sum of the differences between the actual outputs and estimated outputs, obtained by either ANN or other relation given in Equations (5.3) and (5.4), are computed for each subspaces. Finally in Table 5-1 the square root of the results are tabulated.

70

Table 5-1 Comparison of actual, ANN and other relation (Equation (5.3) and (5.4)) outputs

	$\overline{R}$ (normalized radius)				
	[0,0.2]	**(0.2,0.4]**	**(0.4,0.6]**	**(0.6,0.8]**	**(0.8,1]**
$\sqrt{\sum (x_{ANN} - x_{ACT})^2}$	0.1215	0.2522	0.3044	0.2469	0.4000
$\sqrt{\sum (x_{OTH} - x_{ACT})^2}$	0.2447	0.5578	0.7968	0.8971	0.814
$\sqrt{\sum (y_{ANN} - y_{ACT})^2}$	0.1618	0.2626	0.3012	0.2325	0.3927
$\sqrt{\sum (y_{OTH} - y_{ACT})^2}$	0.2540	0.5781	0.8227	0.9183	0.8159

As can be seen from the table, spot center location estimate by ANN is much more close to the actual ones.

5.5 Summary

In this chapter, first seeker and detector are described and error angles in pitch and yaw planes are defined. These angles can be determined by the detector outputs in two steps. In the first part, laser spot center location is estimated from the detector voltages and in the second part, knowing the spot center location, error angles are estimated. The problem of error angles estimation from center spot location is solved by considering optic knowledge stating that ray of light passing the lens from its center doesn't bend.

The estimation of spot center location from detector voltages are separately examined. There already exists a relationship given in Equations (5.3) and (5.4), but its success, as the spot center goes far away from the detector center, decreases. Therefore an ANN estimation model is suggested instead of them. After the ANN has learnt the relationship, its success is compared with the other equations taking the actual values as basis. It is seen that the amount of error in ANN estimates are smaller than that of the other equations. However more meaningful results about the success of the seeker model will be obtained, when both ANN and other relationships are placed in 6 degree of freedom simulations.

CHAPTER 6

OVERALL ASGM MATHEMATICAL MODEL

6.1 Introduction

After we worked on seeker and CAS units of the munition, a controller for the munition is left to be designed. Since no sensors like accelerometers or gyros are included in the munition, some common autopilots, such as acceleration autopilot, can not be designed. However we can design a controller built on error angles estimated by seeker unit and in this situation fuzzy control could be a good solution. In this chapter first a brief introduction to fuzzy control will be given and designed fuzzy control will be described. Afterwards overall ASGM model will be explained with the input and output signal definition of each block (CAS block for example) built in MATLAB Simulink.

6.2 Fuzzy Logic Control for Missile

6.2.1 What is Fuzzy Logic Control

The concept of Fuzzy Logic (FL) was conceived by Lotfi Zadeh, a professor at the University of California at Berkley, and presented not as a control methodology,

but as a way of processing data by allowing partial set membership rather than crisp set membership or non-membership. Professor Zadeh reasoned that people do not require precise, numerical information input, and yet they are capable of highly adaptive control. If feedback controllers could be programmed to accept noisy, imprecise input, they would be much more effective and perhaps easier to implement.

In this context, FL is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PC or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. FL's approach to control problems mimics how a person would make decisions, only much faster [18].

The fuzzy controller is composed of four main components called fuzzification, inference, knowledge base and defuzzification shown as in Figure 6-1.
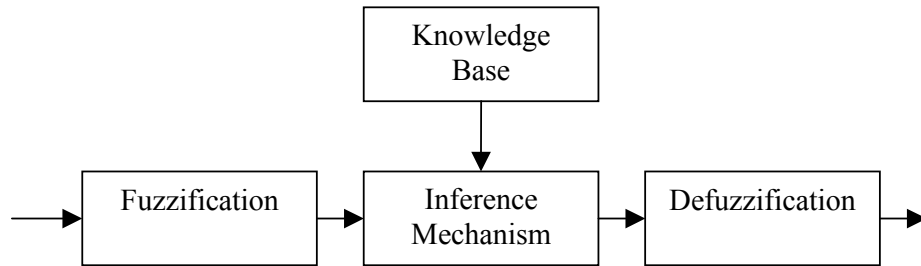
```
                    ┌─────────────┐
                    │  Knowledge  │
                    │    Base     │
                    └──────┬──────┘
                           │
                           ▼
┌────────────────┐   ┌──────────────┐   ┌──────────────────┐
│                │   │  Inference   │   │                  │
──▶│ Fuzzification  │──▶│  Mechanism   │──▶│ Defuzzification  │──▶
└────────────────┘   └──────────────┘   └──────────────────┘
```

Figure 6-1 Basic configuration of a fuzzy logic controller

In fuzzification phase input data is converted into suitable linguistic values that may be viewed as labels of fuzzy sets. In order to understand the fuzzification, the definitions of fuzzy set and membership functions must be made. A classical set might be expressed as

$$A = \{x \mid x > 6\} \tag{6.1}$$

A fuzzy set is an extension of a classical set. If X is the universe of discourse and its elements are denoted by x, then a fuzzy set A in X is defined as a set of ordered pairs.

$$A = \{x, \mu_A(x) \mid x \in X\} \tag{6.2}$$

$\mu_A(x)$ is called the membership function of x in A. The membership function maps each element of X to a membership value between 0 and 1. A membership value of zero indicates that the element is not a member of the set A. A membership value of one indicates that the element is fully a member of the set. A value between zero and one indicates that a fuzzy relation exists between the element and the fuzzy set.

75

The only condition a membership function must really satisfy is that it must vary between 0 and 1. The function itself can be an arbitrary curve whose shape we can define as a function that suits us from the point of view of simplicity, convenience, speed, and efficiency.

Knowledge base involves the control policy for the human expertise, so it is composed of a set of rules created by the human expertise. The rules are simple if-then statements and the simplest if-then rule assumes the form given below.

If x is A then y is B

where x and y are the input and output variables respectively and, A and B are the input and output fuzzy sets respectively.

An inference mechanism evaluates which control rules are relevant at the current time and then decides what the input to the plant should be. This requires evaluation of each fired rule, which contain logical operations (AND, OR, NOT) on fuzzy sets. In more general terms, we need to define fuzzy intersection (AND), fuzzy union (OR), and fuzzy complement. The generally used evaluation methods for AND, OR and NOT operations are defined below.

$$\mu_A(u) \text{ and } \mu_B(u) = \min\{\mu_A(u), \mu_B(u)\}$$

$$\mu_A(u) \text{ or } \mu_B(u) = \max\{\mu_A(u), \mu_B(u)\}$$

$$\text{not } \mu_A(u) = 1 - \mu_A(u)$$

The defuzzification interface converts the conclusions reached by the inference mechanism into the inputs to the plant. The typical defuzzification technique is

calculating the center of gravity. It is actually calculating the moment of area of the fired output membership functions [20][21][22][23].

6.2.2 Description of Designed Fuzzy Logic Control

The input of the fuzzy logic controller are error angle in one plane and its derivative, which are abbreviated as "err" and "err-dot", output is the canard deflection angle in the corresponding plane, which is also abbreviated as "def-ang". Designed fuzzy logic controller will be used for both pitch and yaw planes. Input membership functions are chosen to be Gaussian, because more rules can be fired using this membership function. Output membership functions are chosen to be triangular, because output membership functions do not need to be overlapped as much as input membership functions do. The limits of the input variables and membership functions are tuned by trial and error process. The finalized input membership functions for each input are given in Figure 6-2 and Figure 6-3.
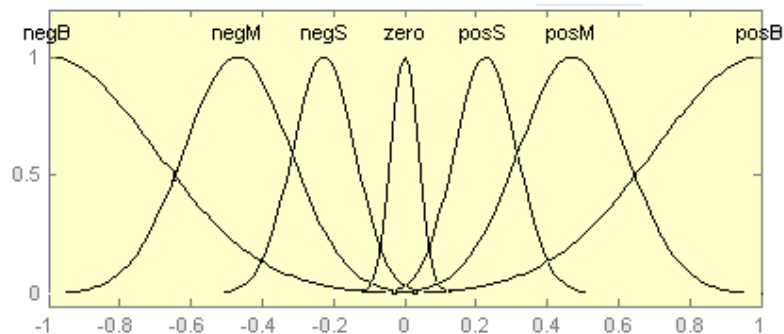

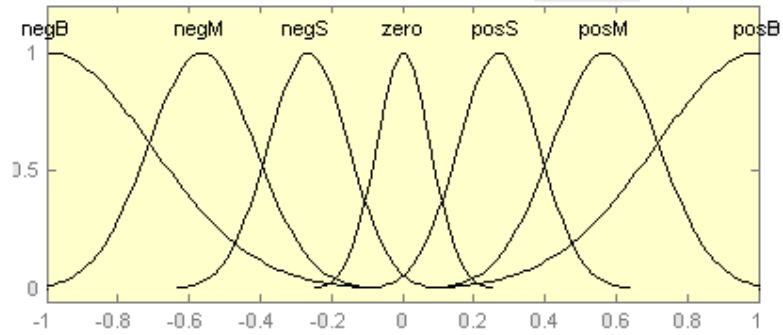**Figure 6-2 Membership functions for input variable "error angle"**

Figure 6-3 Membership functions for input variable "error angle derivative"

The long forms of linguistic abbreviations used in these figures are given below.

negB: Negative Big

negM: Negative Medium

negS: Negative Small

zero: Zero

posS: Positive Small

posM: Positive Medium

posB: Positive Big

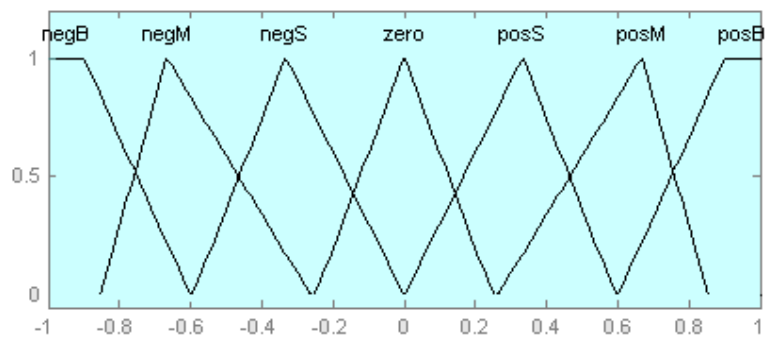The membership functions of the output variables are given in Figure 6-4.

Figure 6-4 Membership functions for output variable "canard deflection"

The rule base of the fuzzy logic control for this ASGM is given in Table 6-1.

Table 6-1 Rule-base for ASGM

"def-ang"		"err"						
		negB	*negM*	*negS*	*Zero*	*posS*	*posM*	*posB*
"err-dot"	*negB*	*negB*	*negB*	*negB*	*negM*	*negM*	*negS*	*zero*
	negM	*negB*	*negB*	*negM*	*negM*	*negS*	*zero*	*posS*
	negS	*negB*	*negM*	*negM*	*negS*	*zero*	*posS*	*posM*
	zero	*negM*	*negM*	*negS*	*Zero*	*posS*	*posM*	*posM*
	posS	*negM*	*negS*	*zero*	*posS*	*posM*	*posM*	*posB*
	posM	*negS*	*zero*	*posS*	*posM*	*posM*	*posB*	*posB*
	posB	*Zero*	*posS*	*posM*	*posM*	*posB*	*posB*	*posB*

This table is read as given in the following example: "If err is negB and if err-dot is negB, then def-ang is negB". The logical operation "and" is evaluated as taking the minimum and defuzzification method is calculating the center of gravity. The mapping from inputs "err" and "err-dot" to output "def-ang" is shown as a surface plot in Figure 6-5.
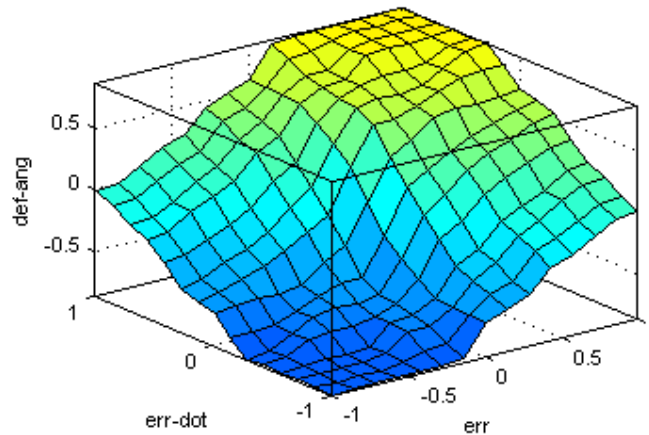
Figure 6-5 Input-Output mapping of the fuzzy logic controller

6.3 Integration of CAS and Seeker Models to the Non-linear Flight Model

Nonlinear flight simulations will be performed for bang-bang control and for continuous control schemes separately, whose mathematical models are built in MATLAB Simulink environment. Bang-bang controlled ASGM model contains seeker model but do not include error angle estimations. The CAS plant used in these simulations is the open loop CAS model given in Equation 4.3. ASGM model with continuous control contains seeker model together with error angle estimations using Artificial Neural Network approach, closed loop CAS models obtained in Sections 4.3.1 and 4.3.2 and fuzzy logic controller. Simulation model of bang-bang controlled ASGM is very straightforward, seeker determines the direction of the target as right or left and up or down, therefore in the simulations output of the seeker is 1 or -1. Output of the seeker directly enters the open-loop

CAS model, input "1" brings the canards to positive full deflection angle, input "-1" brings them to negative full deflection angle.

Mathematical model of modified ASGM is mainly composed of four modules, nonlinear flight model, seeker model, closed loop CAS model and fuzzy logic controller.

In order to integrate the seeker model, first we estimate the error angles, which are required to generate detector voltages. After the detector voltages, in other words detector areas, are obtained, seeker model is inserted to the system. The input of the seeker model is 4 quadrant detector voltages, output is the error angles. As a summary error angles are estimated twice and second one is used in the simulations.

These error angles enter the block called "Fuzzy Logic Controller". The output of this block is the commanded canard deflection angles, which are the inputs of the closed loop CAS models.

Commanded deflection angles enter the CAS block and its realization is sent to the 6-DOF nonlinear flight simulation block, which closes the loop. The block schema of the integration structure is given below in Figure 6-6.
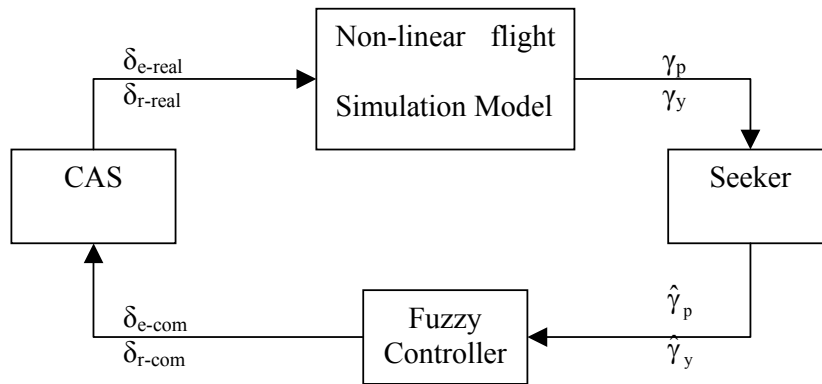
Figure 6-6 Block schema of the integrated munition system

CHAPTER 7

SIMULATION STUDIES

7.1 Introduction

Simulation studies are carried out in order to see the success of the modifications made on ASGM. Comparison is made between the hit accuracy of bang-bang control scheme and continuous control scheme with modified CAS, seeker units and fuzzy logic controller.

7.2 Simulation Scenarios

7.2.1 Scenario 1

In this scenario, ASGM is fired onto a stationary target, which is aligned with the ASGM in x-z plane (pitch plane). The initial state values of ASGM and location of target is summarized in Table 7-1.

Table 7-1 Initial state values of simulation and target location table for

Scenario 1

ASGM					
Initial Position			**Initial Attitude**		
x	y	z	Φ	θ	Ψ
0 m	0 m	-5000 m	0 deg	-10 deg	0 deg
Initial velocity, α and β			**Initial Angular Rates**		
V_t	α	β	p	q	r
295 m/s	0 deg	0 deg	0	0	0
Target Location					
x	y	z			
8500 m	0 m	0			

 Below figures are drawn for three different configurations; first one is bang-bang, second one is continuous control with fuzzy logic using CAS model with second PWM scheme and the last configuration is again continuous control but with first PWM scheme.
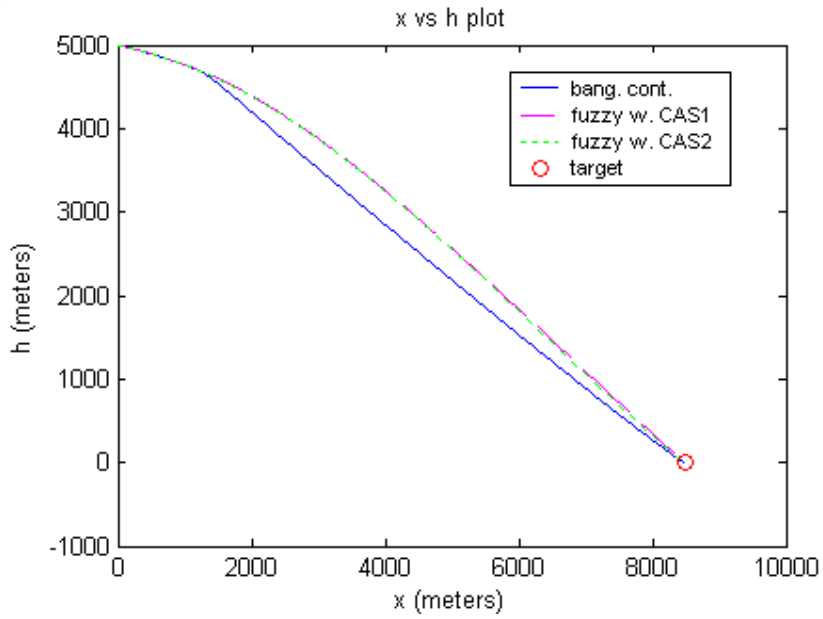
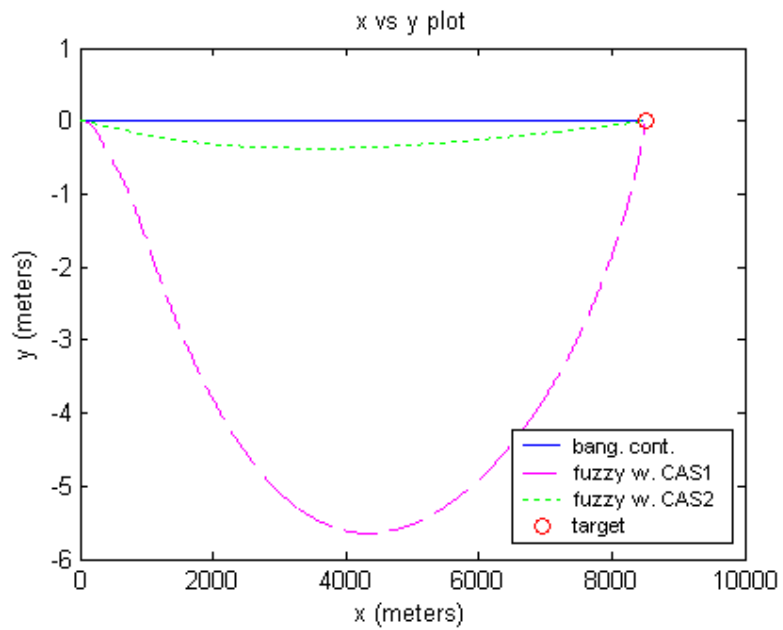**Figure 7-1 x-h plane trajectory of ASGM for Scenario 1**

Figure 7-2 x-y plane trajectory of ASGM for Scenario 1

Figure 7-1 and Figure 7-2 show that all the configurations of ASGM hit the target with a hit accuracy less than 10 meters, whose details are given in Table 7-2. Interesting trajectory in x-y plane occurs due to continuous control. Because when there is an error angle less than 1 degree, then there is no control action in bang-bang control simulations. In other words this is zero region for bang-bang control scheme. However in continuous control, any error angle no matter how small it is causes CAS to be commanded to some angle. As a result this trajectory is observed in x-y plane.

This can also be seen from Φ, θ and Ψ angle histories given in Figure 7-3. While some little amount of Φ and Ψ angles are realized in continuous control scheme, in bang-bang control scheme no Φ and Ψ angles are realized. In addition to this, with continuous control scheme, large oscillations in pitch angle occurred (θ) with bang-bang control scheme is overcome.
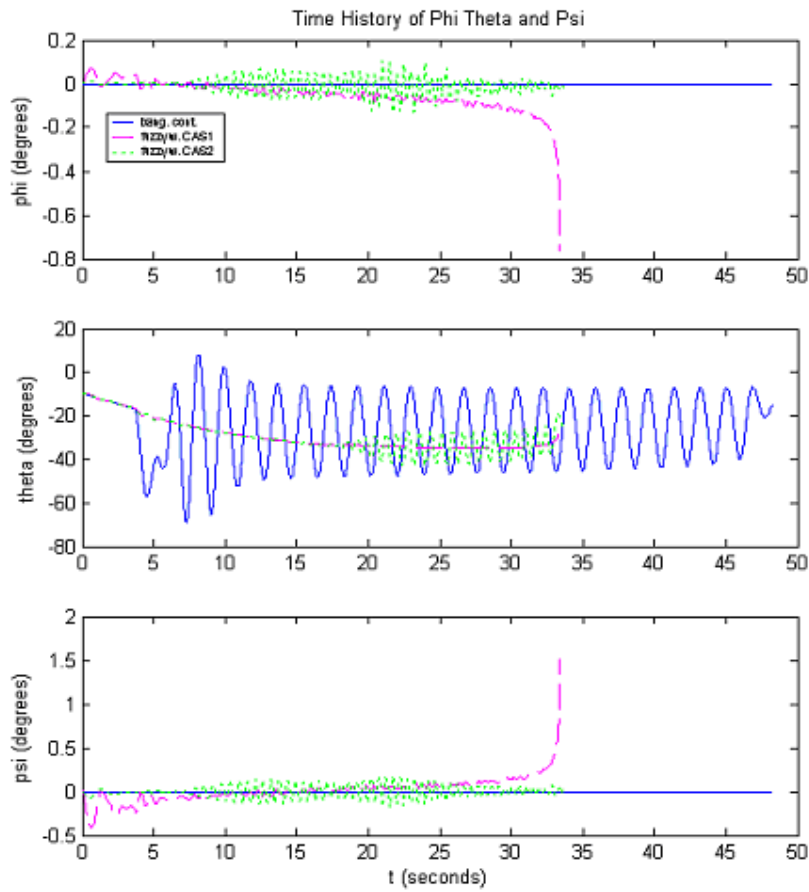
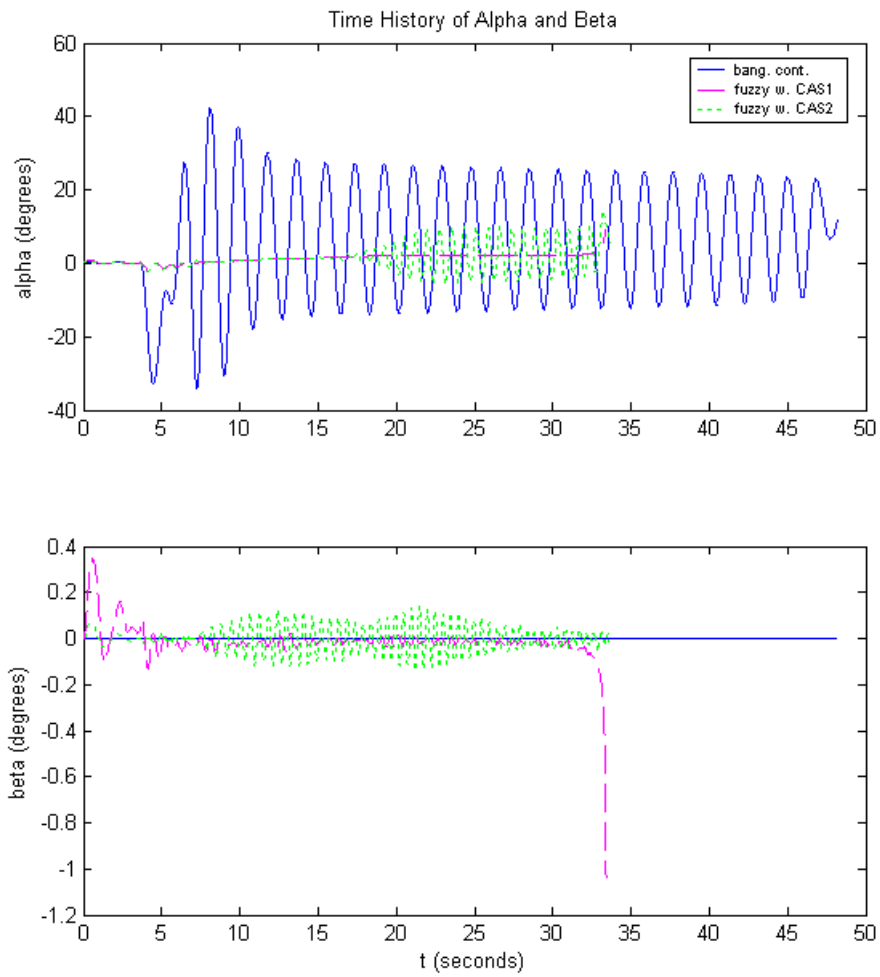**Figure 7-3 Time history of Φ, θ and Ψ angles for Scenario 1**

Figure 7-4 Time history of angle of attack (α) and side slip angle (β) for

Scenario 1

Figure 7-4 shows that oscillations in especially angle of attack due to the bang-bang control are also overcome with continuous control scheme.
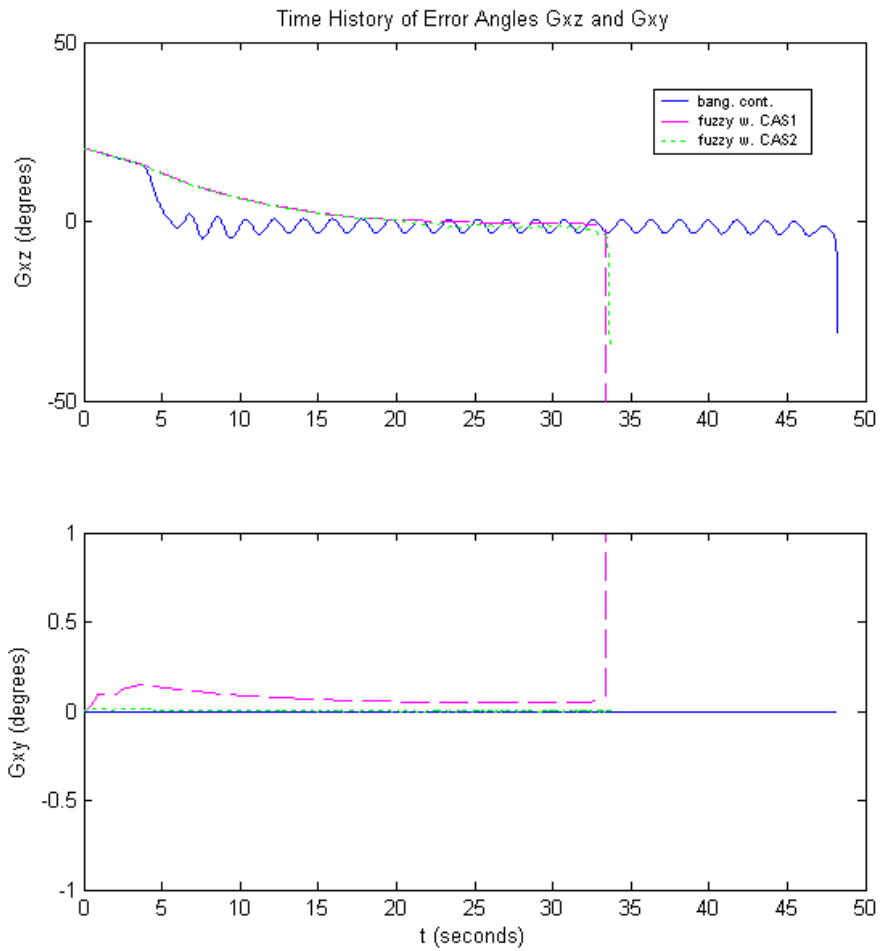
Figure 7-5 Time history of error angles γ_P (or γ_{xz}) and γ_Y (or γ_{xy}) for scenario1

Figure 7-5 shows how error angles are brought to zero, by continuous and bang-bang control schemes. As expected error angle time history in bang-bang control scheme oscillates around zero degrees. On the other hand in continuous control scheme error angle is reduced to zero without oscillations and when it reaches to zero fuzzy controller holds it at zero degrees error angle.
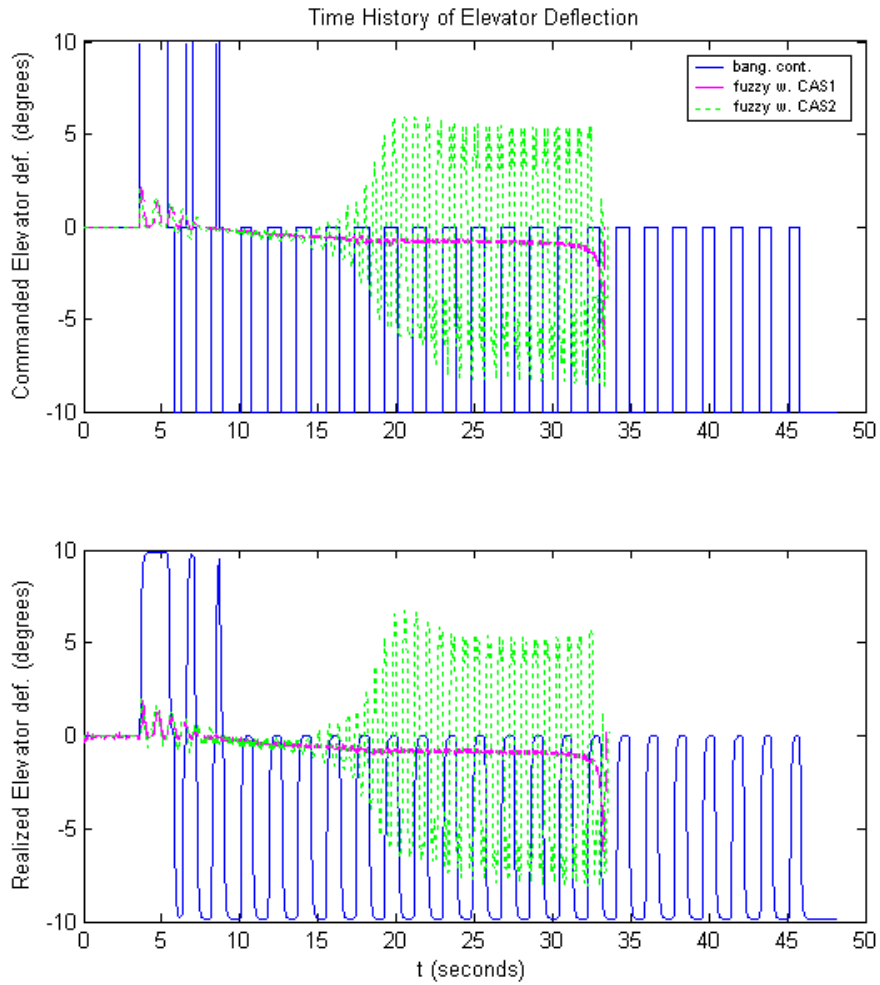
Figure 7-6 Time history of horizontal canard deflection angle for Scenario1

In Figure 7-6, horizontal canard deflection angle time history plot is shown. The dotted line in this plot shows a very oscillatory trend, this is because of CAS used in this simulation, whose mathematical model and step response plots were given in Section 4.3.1. This also shows the importance of CAS in this ASGM system. Despite of using the same fuzzy controller in these two continuous control

configuration, CAS models having different time domain characteristics, such as settling time, response time etc, causes considerable changes in the behavior of whole ASGM system.
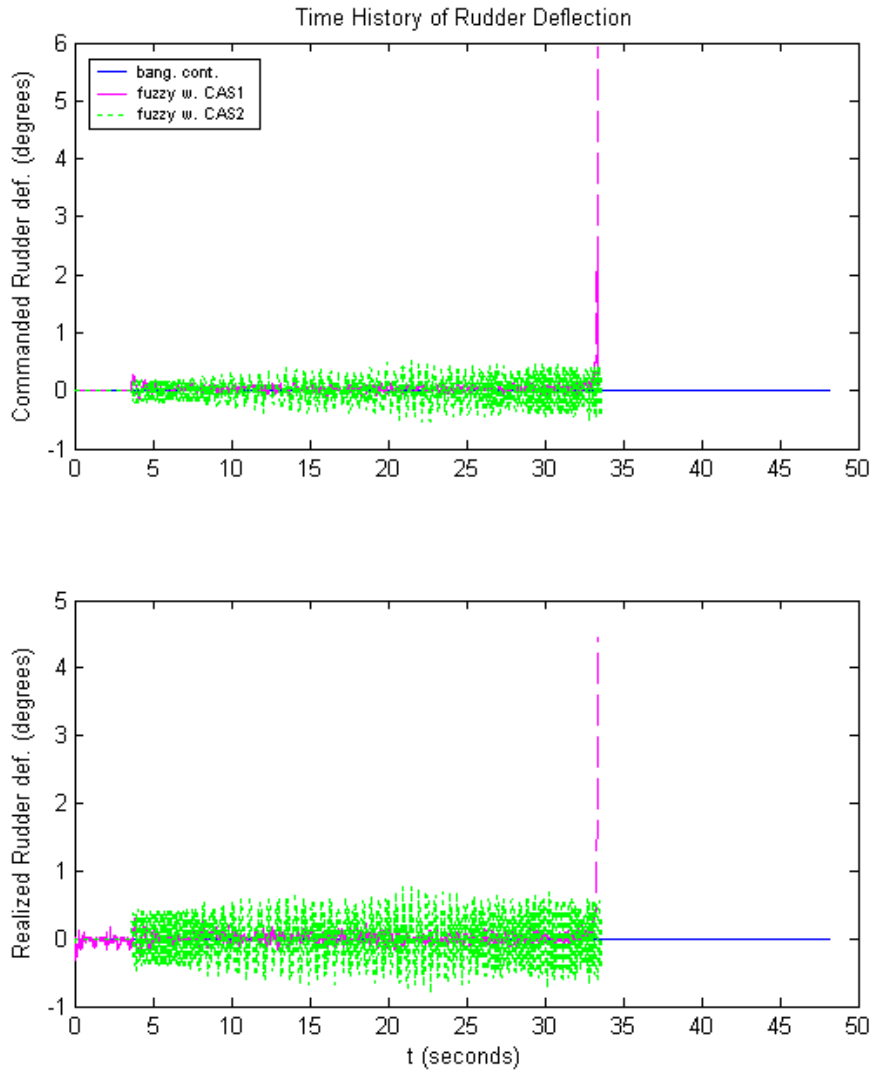

Figure 7-7 Time history of vertical canard deflection angle for Scenario1

One more thing to notice is that shorter time of flight in continuous control scheme simulations, which can also be seen in Table 7-2.

Table 7-2 Performance of each configuration for Scenario 1

	Miss Distance (meter)	Time of Flight (second)
Bang-Bang Control	6.7	48.2
Continuous Cont with Second PWM scheme CAS	1.0	33.4
Continuous Cont with First PWM scheme CAS	9.5	33.6

Table 7-2 also shows that hit accuracy of ASGM is increased when it uses modified CAS with second PWM scheme and ANN seeker model. Moreover since the deviation of modified ASGM from 0 error-angle is much more less than actual ASGM, time of flight is also decreased by considerable amount. All these results are valid for the ASGM directed right onto the target, but what happens if it is not directed right onto the target. This is examined in second and third scenarios.

7.2.2 Scenario 2

In this scenario, the location of the target is 8500 meters in x direction, 900 meters in y direction and its height is 0 meter, whereas ASGM is fired from a height of 5000 meters. Initial condition of ASGM and location of stationary target is summarized in Table 7-3.

Table 7-3 Initial state values of simulation and target location table for

Scenario 2

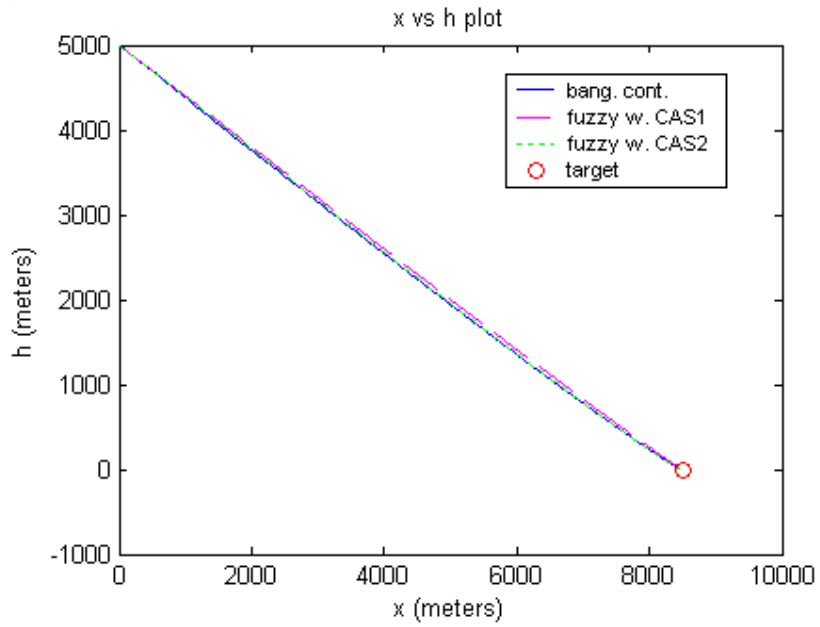| ASGM					
Initial Position			**Initial Attitude**		
x	y	z	Φ	θ	Ψ
0 m	0 m	-5000 m	0 deg	-30 deg	5 deg
Initial Velocity, α and β			**Initial Angular Rates**		
V_t	α	β	p	q	r
295 m/s	0 deg	0 deg	0	0	0
Target Location					
x	y	z			
8500 m	900 m	0			

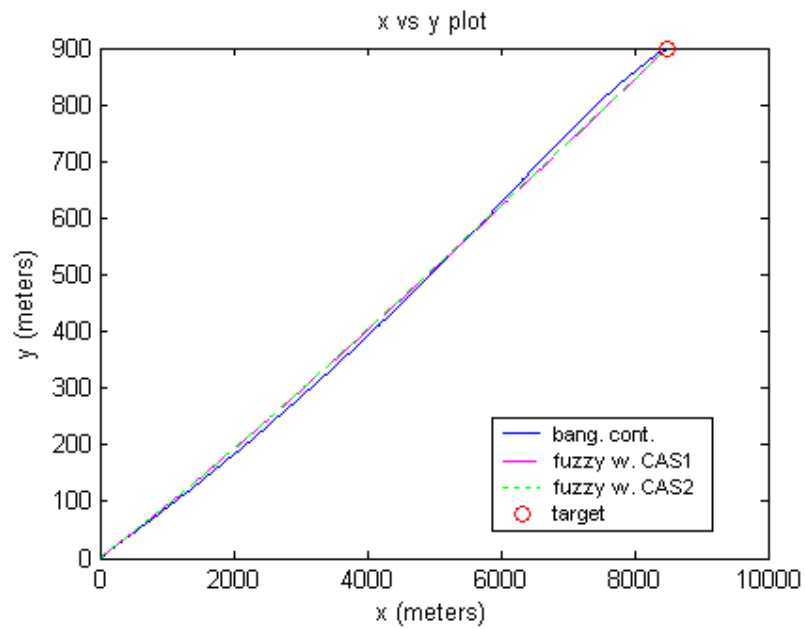**Figure 7-8 x-h plane trajectory of ASGM for Scenario 2**

Figure 7-9 x-y plane trajectory of ASGM for Scenario 2

Figure 7-8 and Figure 7-9 show the trajectory of ASGM (bang-bang control) and modified ASGM (continuous control). From these plots, it seems all of them hit the target, but actual hit accuracy is given in Table 7-4. θ history (Figure 7-10) for bang-bang control is very similar to the one observed in scenario 1. Again these oscillations are damped by the use of fuzzy logic controller.
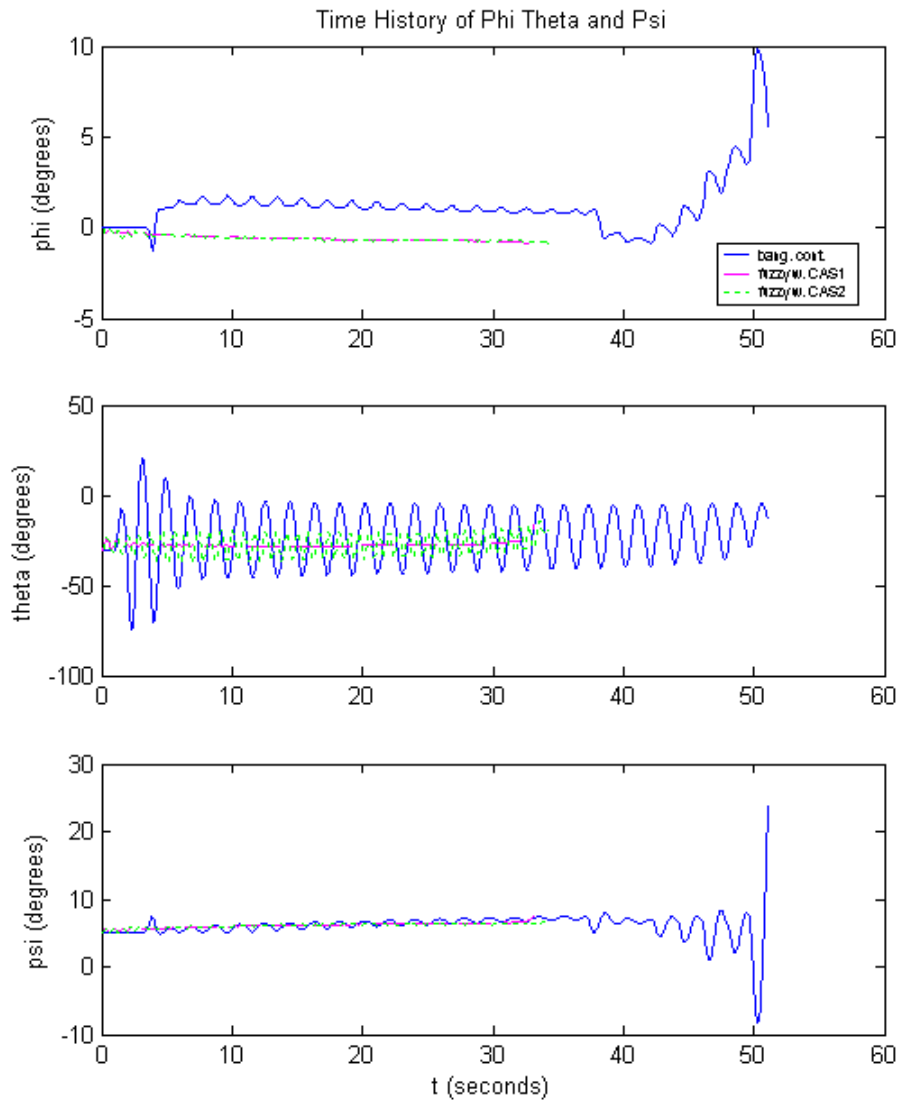
Figure 7-10 Time history of Φ, θ and Ψ angles for Scenario 2

Figure 7-11 Time history of angle of attack (α) and side slip angle (β) for

Scenario 2

As can be seen in Figure 7-11, in bang-bang control simulations, angle of attack oscillates too much, and these oscillations are removed by the use of fuzzy logic control.

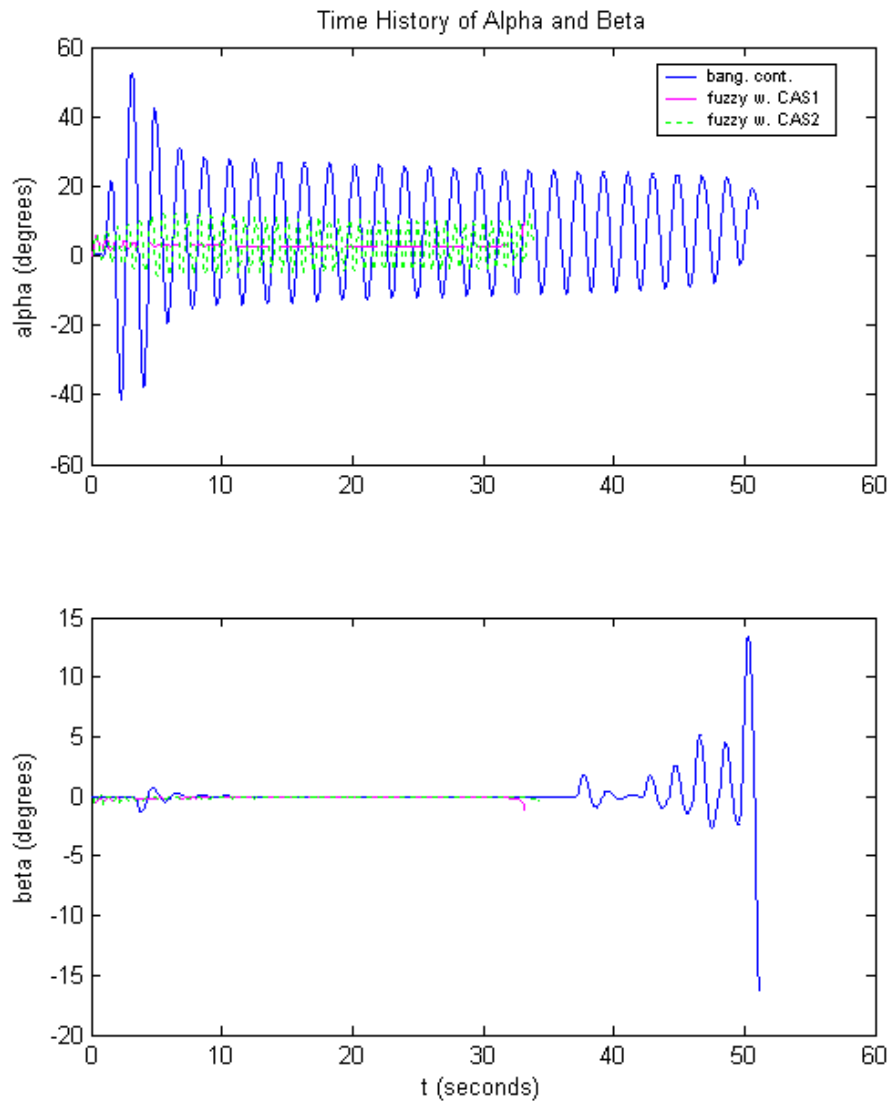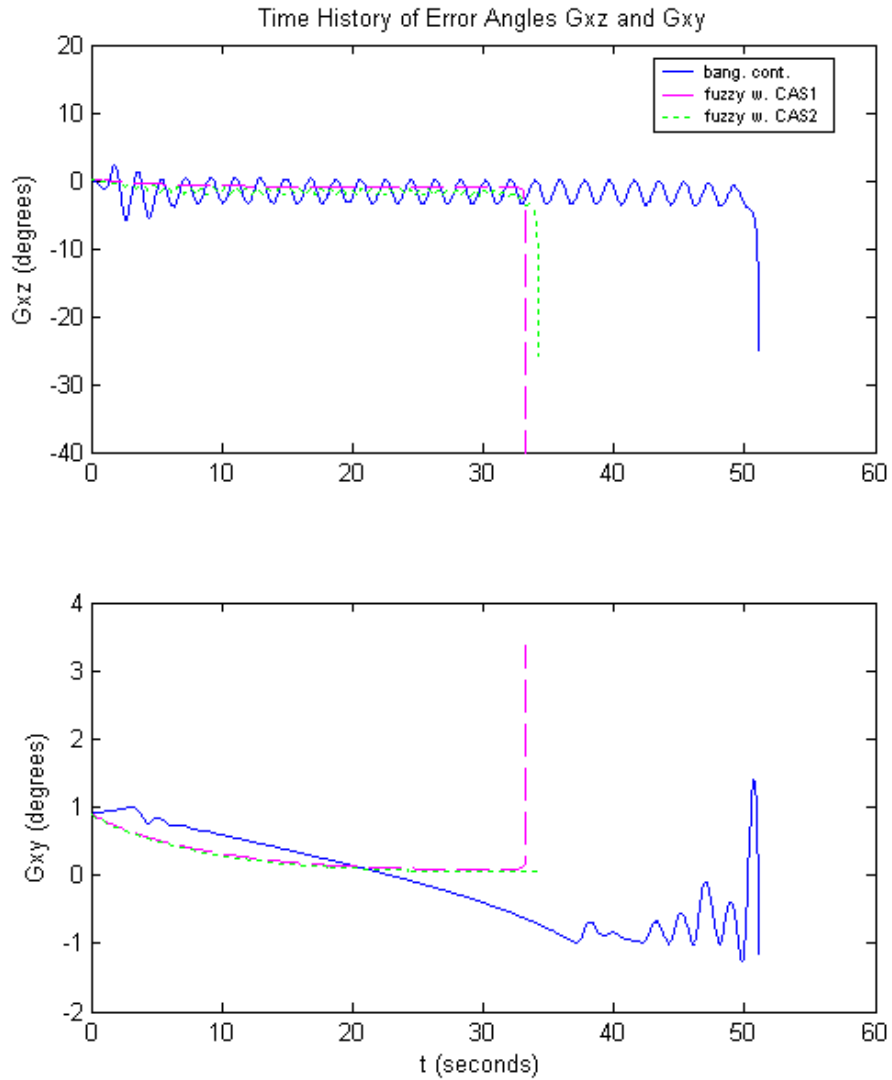**Figure 7-12 Time history of error angles γ_P (or γ_{xz}) and γ_Y (or γ_{xy}) for scenario2**

Figure 7-12 shows the success of the continuous control when compared with the bang-bang control, because bang-bang control causes munition to oscillate around zero error angle, whereas in continuous control case once it is brought to zero error angle, it is held at this position without oscillations.
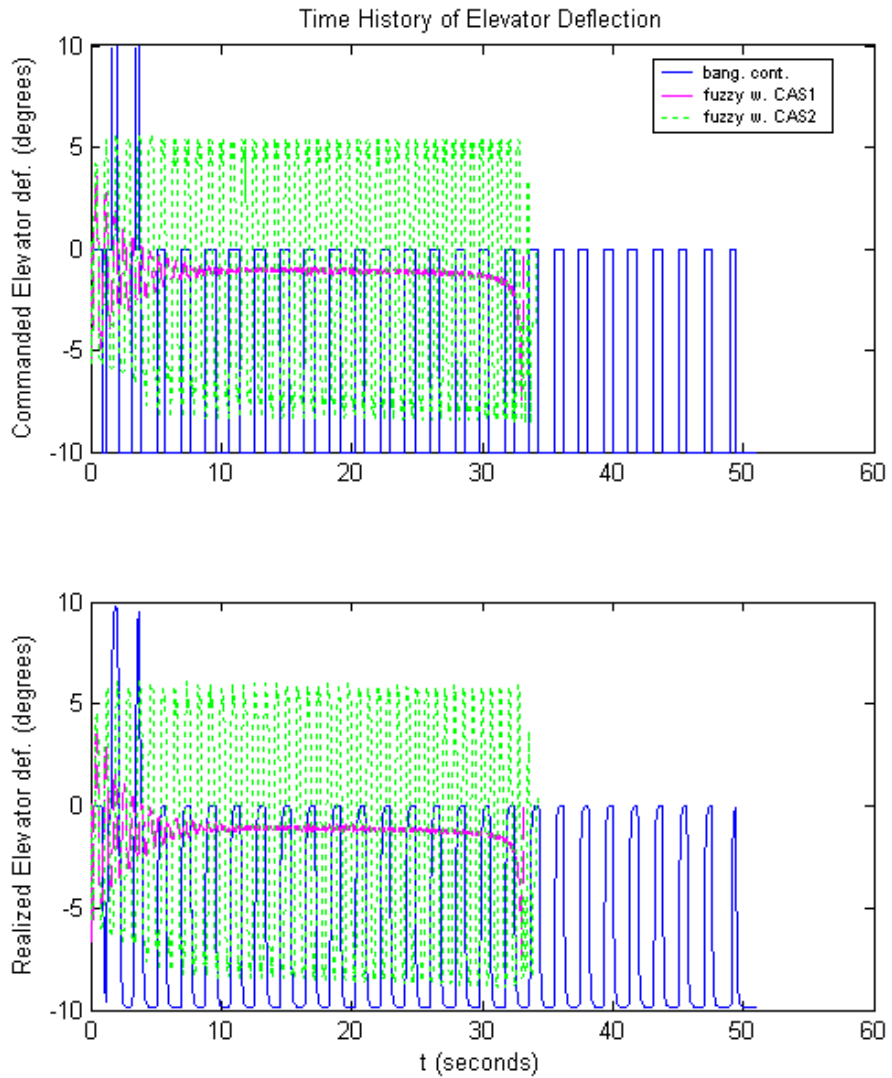
Figure 7-13 Time history of horizontal canard deflection angle for Scenario2

In Figure 7-13 resultant horizontal canard deflection of bang-bang control simulations are obvious, but there is one more horizontal canard angle history that oscillates much, too. This angle history is the result of continuous control with first PWM scheme CAS. This is because CAS itself has a very oscillatory response, which can be seen in Section 4.3.1.
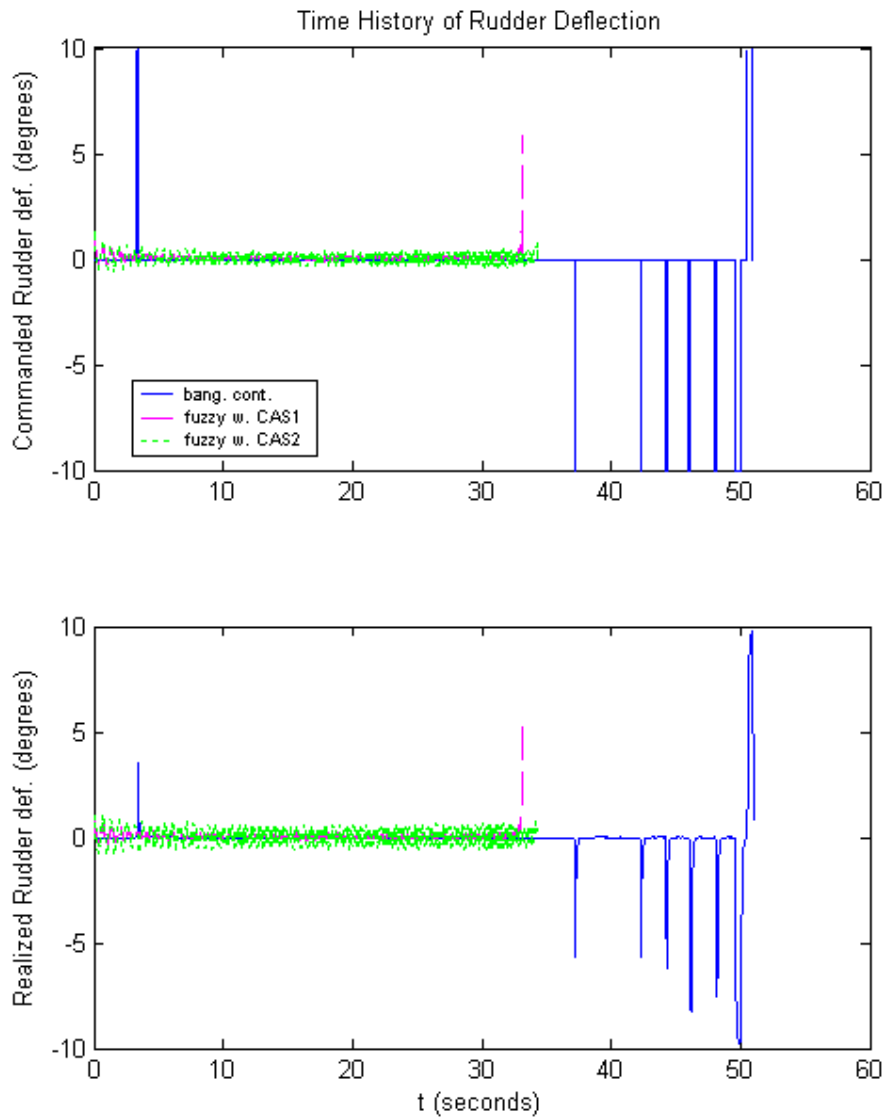
Figure 7-14 Time history of vertical canard deflection angle for Scenario2

Since error angle in x-y plane is very small, small vertical canard deflections would be enough to steer the munition. This fact is realized in bang-bang control

simulation as most of the time zero vertical canard deflection and several times full
deflection commands for a little time.

Table 7-4 Performance of each configuration for Scenario 2

	Miss Distance (meter)	Time of Flight (second)
Bang-Bang Control	15.8	51.1
Continuous Cont with Second PWM scheme CAS	1.9	33.2
Continuous Cont with First PWM scheme CAS	12.8	34.3

Table 7-4 summarizes the performance of bang-bang control and continuous
control scheme. According to this table, as expected, ASGM with fuzzy logic
controller using second PWM scheme hits the target with accuracy less than 2
meters. The unsatisfactory performance of continuous control using first PWM
scheme is due to insufficient closed loop performance of CAS. In the third scenario
effect of initial large error angle will be examined.

7.2.3 Scenario 3

In scenario 3, the location of target is not altered, but ASGM has different orientation than the second scenario. Initial state values and target location are given in Table 7-5.

Table 7-5 Initial state values of simulation and target location table for

Scenario 3

ASGM					
Initial Position			**Initial Attitude**		
x	y	z	Φ	θ	Ψ
0 m	0 m	-5000 m	0 deg	-10 deg	0 deg
Initial velocity, α and β			**Initial Angular Rates**		
V_t	α	β	p	q	r
295 m/s	0 deg	0 deg	0	0	0
Target Location					
x	y	z			
8500 m	900 m	0			

In this scenario, bang-bang controlled ASGM loses the acquisition with the target after some time, which can be seen in Figure 7-15 and Figure 7-16.
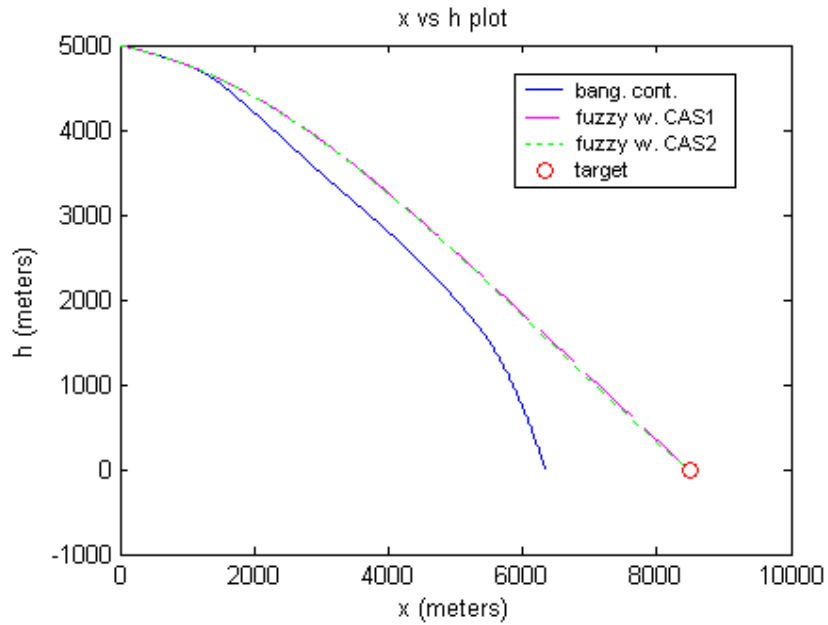
Figure 7-15 x-h plane trajectory of ASGM for Scenario 3
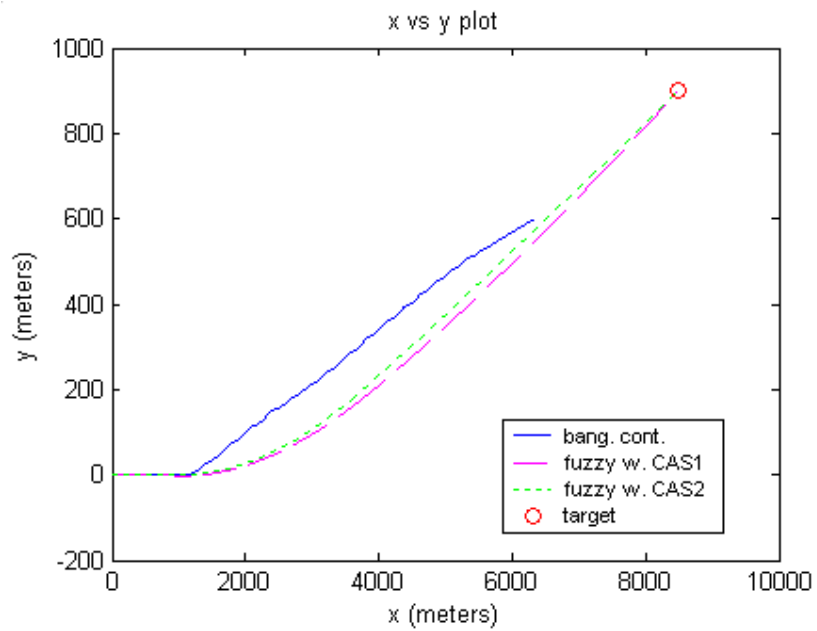
Figure 7-16 x-y plane trajectory of ASGM for Scenario 3
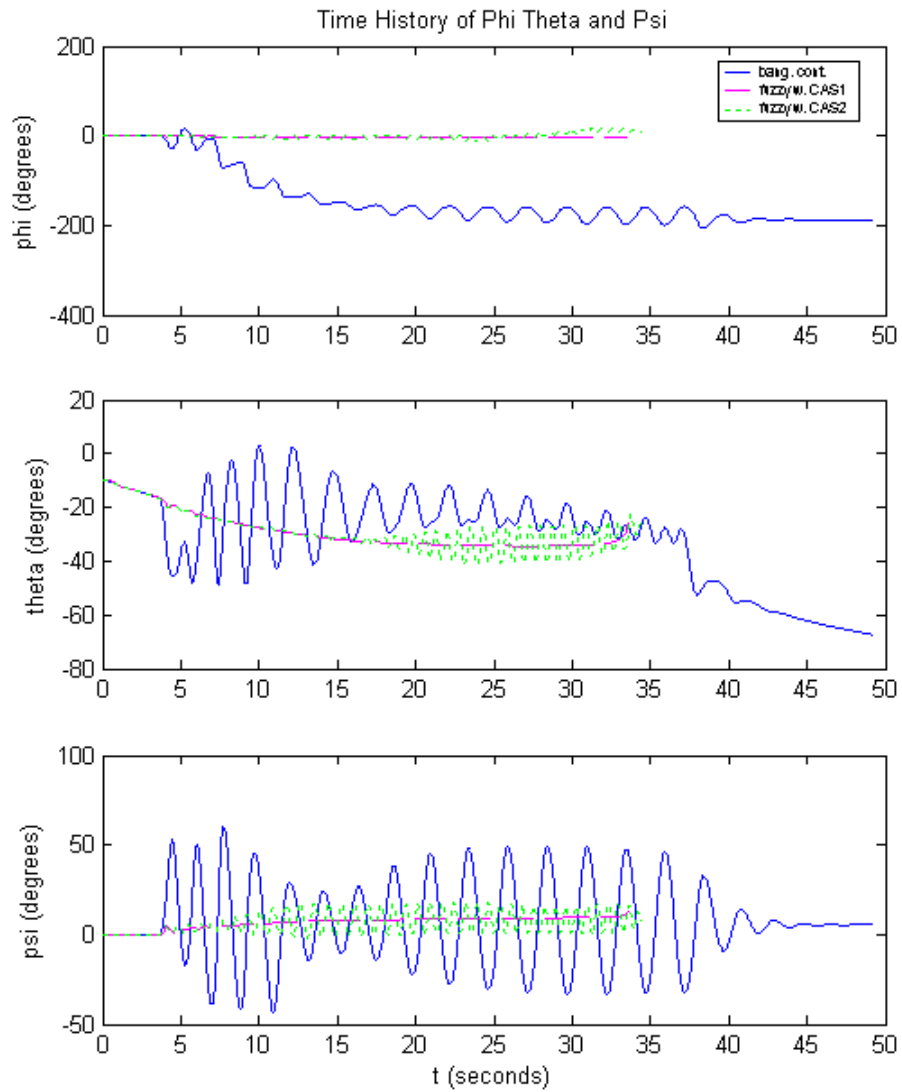
Figure 7-17 Time history of Φ, θ and Ψ angles for scenario 3

Figure 7-17 shows that due to full horizontal canard and vertical canard deflections in bang-bang control scheme, munition attains large roll angles, but this doesn't create a problem in controlling point of view. Because borders, which separate the

quadrants, of the detector are aligned with the canards, so estimated error angles are actually on the same plane with the horizontal canard and vertical canard planes. If an error angle is estimated on one plane, control surface on the same plane is deflected.
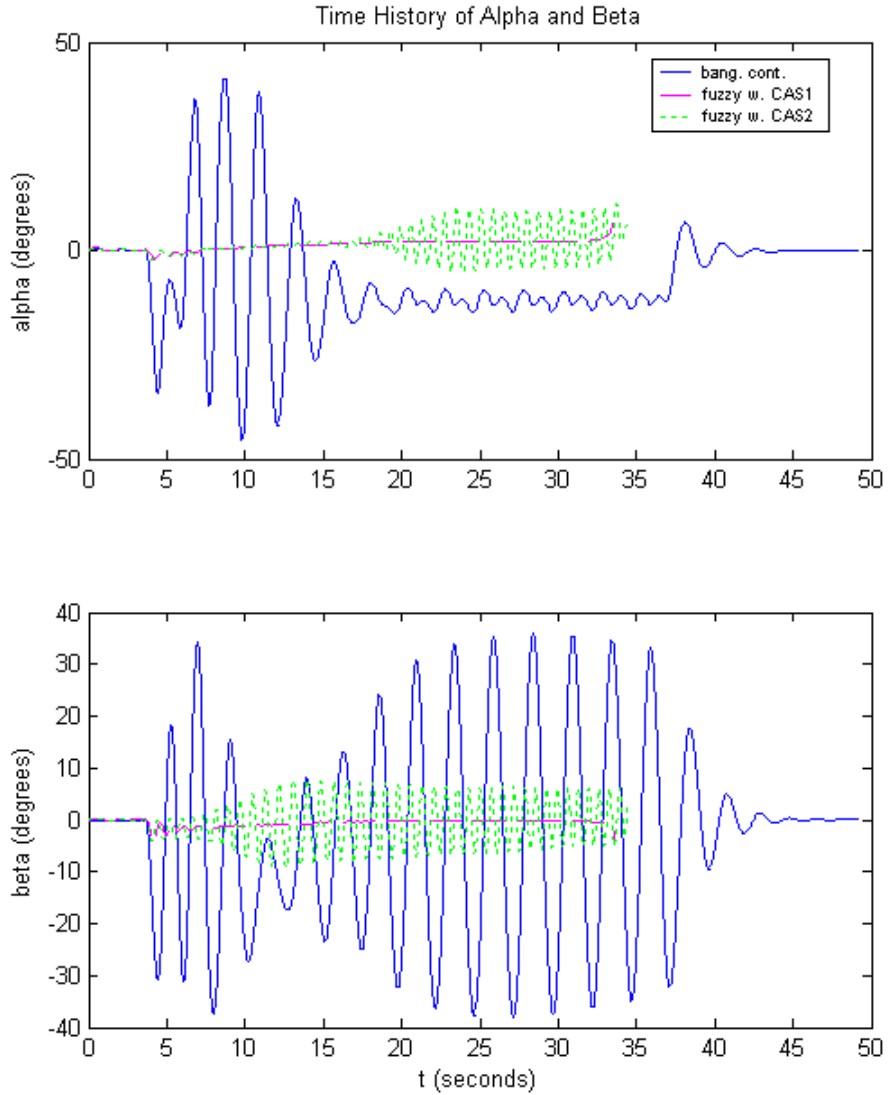
Figure 7-18 Time history of angle of attack (α) and side slip angle (β) for

scenario 3

In Figure 7-18, oscillations in angle of attack and side slip angle for bang-bang control scheme are present as expected and again continuous control with second PWM scheme CAS ASGM damped these oscillations.
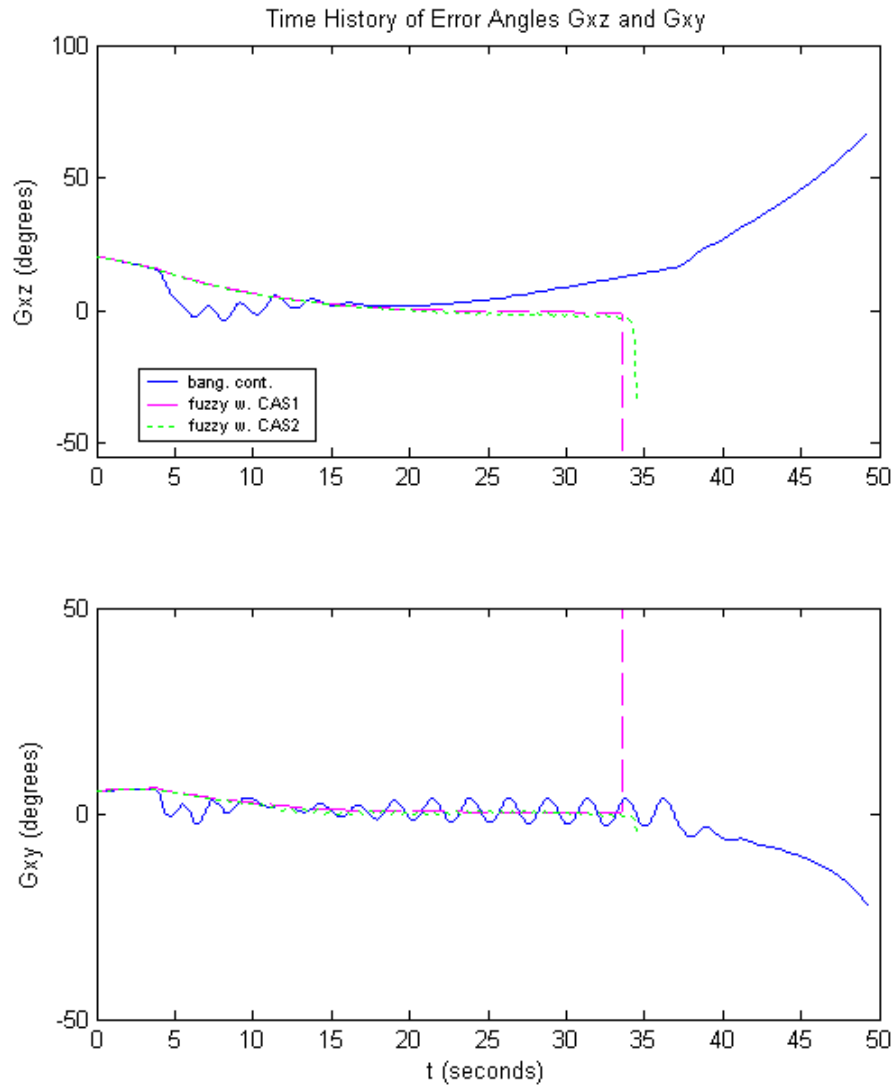
Figure 7-19 Time history of error angles γ_P (or γ_{xz}) and γ_Y (or γ_{xy}) for scenario3

Figure 7-19 shows the error angle history in both planes and demonstrates very similar trend to the previous two scenarios. From these plots, the point where acquisition starts and ends can also be found out, at around 36th second acquisition is lost and flight is turned to ballistic flight in bang-bang controlled ASGM. Accordingly vertical and horizontal canard defections are zero after this time, which can be seen in Figure 7-19.
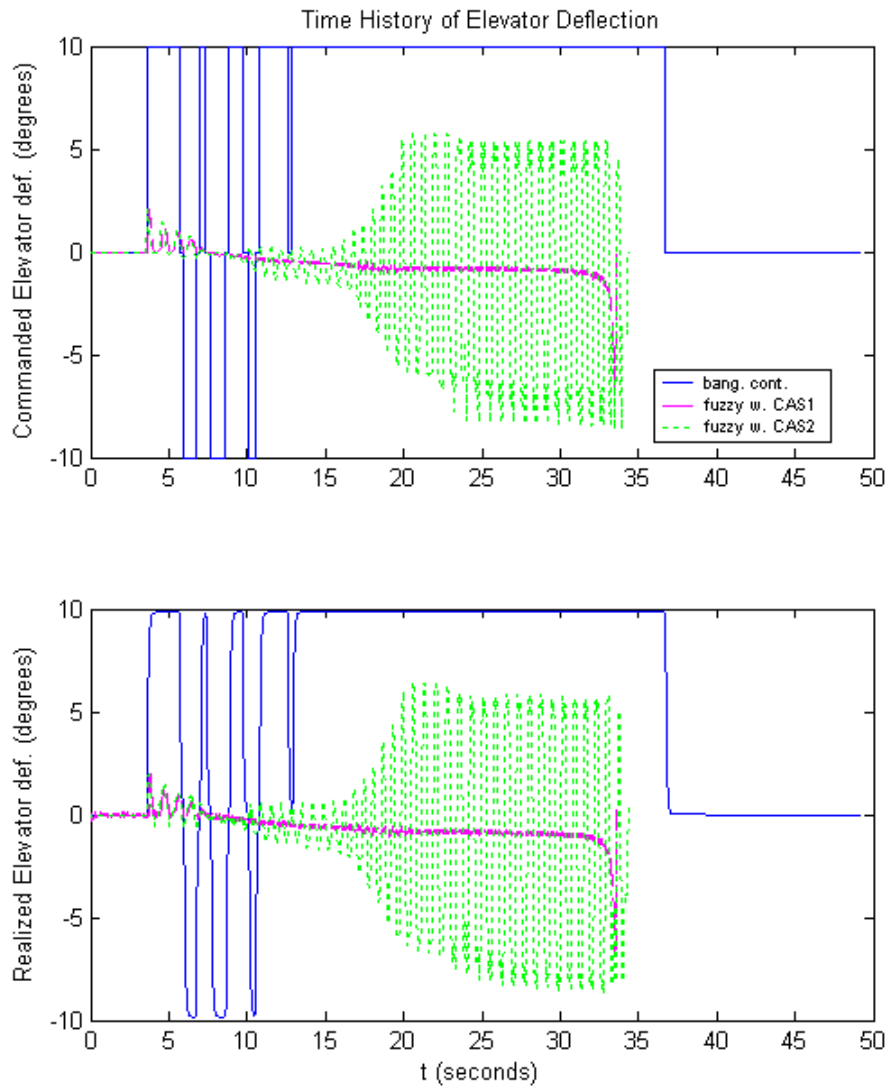
Figure 7-20 Time history of horizontal canard deflection angle for scenario 3
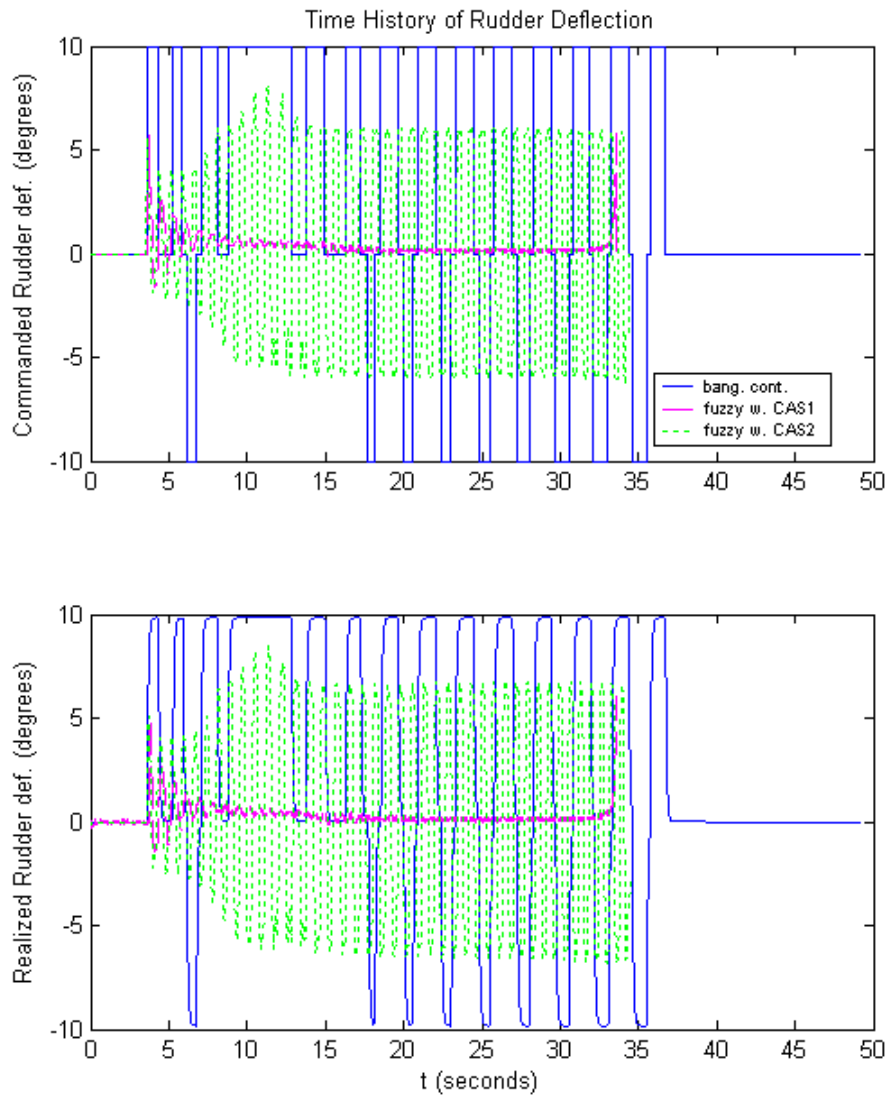
Figure 7-21 Time history of vertical canard deflection angle for scenario 3

Table 7-6 Performance of each configuration for scenario 3

	Miss Distance (meter)	Time of Flight (second)
Bang-Bang Control	- (acquisition lost)	49.3
Continuous Cont with Second PWM scheme CAS	1.1	33.6
Continuous Cont with First PWM scheme CAS	13.5	34.5

Table 7-6 indicates that when control algorithm is continuous and the performance of CAS is satisfactory and compatible with the munition, the release envelope of the munition can be enlarged.

CHAPTER 8

CONCLUSION

This thesis aims to improve the mentioned ASGM by modifying some of its components to reduce miss distance. For this purpose, first seeker unit of ASGM was modified then control actuation system was changed to the one that is continuous. In other words, canards rotate to the desired deflection, not full deflection.

Modification on seeker unit is made by changing the algorithm. Before modification detector was used to decide on whether the target is up/below or right/left of the munition, but with the modification the error angles can be estimated now. Although there is a popular equation that is used to locate the laser spot, for accuracy in estimation of the laser spot we suggested ANN seeker model which estimated the laser spot location more accurately.

The major improvement is made on the control actuation system, because after the modification it could operate continuously. Two mathematical models were

estimated using system identification, and they were used with two different PWM schemes.

In order to see the effect of these improvements on the overall system, simulations are performed for three scenarios. The first and second scenarios showed us that the miss distance decreased to around 2 meters, whereas in bang-bang control it is around 16 meters. Moreover time of flight value is decreased when compared with the bang-bang controlled ASGM. However it is also seen that the oscillatory characteristic of the first PWM scheme CAS increased the miss distance to a value that could also be accomplished by bang-bang control. This showed us that the performance of the CAS is as important as the performance of flight controller.

In the third scenario, it is shown that, modified ASGM has a larger delivery envelope when compared with the bang-bang controlled ASGM. Because acquisition with the target is lost after some time in this scenario, whereas improved ASGM again hits the target with an accuracy of 1 meter.

Future studies could be building a nonlinear parametric model of CAS and estimating the unknown parameters of this model, because it is difficult to fit a linear mathematical model; as indicated in Chapter 3. Moreover, in order to damp the oscillations, model predictive controller can be designed instead of PI with rate feedback controller for the control actuation system.

REFERENCES

[1]   Blakelock, J. H., "Automatic Control of Aircraft and Missiles", John Wiley & Sons Inc., 1991.

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

[3]   Roskam, J., "Airplane Flight Dynamics and Automatic Flight Controls" DARcorporation, 1995.

[4]   Halici U., Leblebicioğlu K., Jain L.C. *"Training Radial Basis Function Neural Networks Through Parabolic Evolutionary Algorithm"* International Journal of Knowledge-Based Intelligent Engineering Systems, Vol.5, No.3, July 2001.

[5]   Kartalopoulos, S. V., "Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications", IEEE Press, 1996.

[6]   MATLAB Neural Network Toolbox User Guide.

[7]   UDT Instruments "Non-Contact Position Sensing Using Optical Detectors" (www.udt.com).

[8]   Efe, M., "Laserle Aydınlatılmış Hedef Konumu Belirleme ve Hedefe Kilitlenme", Yüksek Lisans Tezi, Elektronik Mühendisliği Bölümü, Ankara Üniversitesi, Ankara 1994.

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

[10]   Sinha, N. K. and Kustza, B., "Modeling and Identification of Dynamic Systems", Van Nostrand Reinhold Co., 1983.

[11]   MATLAB System Identification Toolbox User's Guide.

[12]   Özkan, A., "Development of a Control Actuation System for an Inertially Guided Missile", M.S. Thesis, Mechanical Engineering Department, METU, Ankara, 1996.

[13]   Perkgöz, C., "A Guidance and Control Method for a Tail Controlled Bomb", M.S. Thesis, Electrical and Electronics Engineering Department, METU, Ankara, 2002.

[14]   Ljung, L., "System Identification: Theory for the User", P T R Prentice Hall, Inc., 1987.

[15]   Ateşoğlu, Ö., "Different Autopilot Designs Their Performance Comparison For Guided Missiles" M.S. Thesis, Aeronautical Engineering Department, METU, Ankara 1996.

[16]   Hou, L. and Michel A. N. *"Stability Analysis of Pulse-Width-Modulated Feedback Systems"*, Automatica, Vol.37, pp.1335-1349, 2001.

[17]   van Varseveld, R. B. and Bone, G. M. *"Accurate Position Control of a Pneumatic Actuator Using On/Off Solenoid Valves"*, IEEE/ASME Transactions on Mechatronics, Vol.2, No.3, pp. 195-204, September 1997.

[18]   Linnet, J.A. and Smith, M. C. *"An Accurate Low Friction Pneumatic Position Control System"* Proceeding of the Institution of Mechanical Engineers Part B, Vol.203, No.33, pp.159-165, April 1989.

[19]   MATLAB Fuzzy Logic Toolbox User's Guide.

[20]   Lin, C. L. and Su, H. W., *"Intelligent Control Theory in Guidance and Control System Design: an Overview"*, Proc. Natl. Sci. Counc. ROC(A), Vol.24 No.1, pp.15-30, 2000.

[21]   O'Hagan, M., "A Fuzzy Decision Maker", (www.fuzzysys.com)

[22]   Passino, K. M., Yurkovich, S., "Fuzzy Control" Addison Wesley Longman Inc., 1998.

[23]   Vural, A. Ö., "Fuzzy Logic Guidance System Design for Guided Missiles" M.S. Thesis, Mechanical Engineering Department, METU, Ankara, 2003.

[24]   Jamshidi, M., Vadiee, N. and Ross, T, "Fuzzy Logic and Control" PTR Prentice Hall, Inc., 1993.

[25]   Rajasekhar V. and Sreenatha A. G. "Fuzzy Logic Implementation of Proportional Navigation Guidance" Acta Astronautica Vol.46, No.1, pp.17-24, 2000.

[26]   Ateşoglu, Ö. "Fuzzy Guidance and Control Inference Design for a Non-linear Air to Surface Missile Model" Nordic Matlab Conference, pp.148-154, 2001.

APPENDIX A

PARABOLIC EVOLUTIONARY ALGORITHM

Initialize Population $G_n=G_0$ with size$(G_n)=P$

repeat

- **Mutation:** Mutatate G_n except the best one considering mutation rate M_r
- **Breeding:** Generate temporary population G_t through **parabolic crossover** and **classical crossover** such that size$(G_t)=P+2*\lceil P*C_r \rceil$
  {

  - **Parabolic crossover:** $L*3*N$ new chromosomes are generated for G_t as follows:
    repeat for loop=1..L

    {

    - select three vectors $\mathbf{x}^a$, $\mathbf{x}^b$, $\mathbf{x}^c$ from G_n in random (however for a predetermined ratio, $\mathbf{x}^a$ is not selected in random but set to $\mathbf{x}_{best}$ in G_n)
    - repeat for i=1..N
    {

    - determine
      $(x_i)^a_{max}$ using three pairs

        $(x^a_i,f^a(x^a_i))$, $(x^b_i,f^a(x^b_i))$, $(x^c_i,f^a(x^c_i))$;

      $(x_i)^b_{max}$ using three pairs

        $(x^a_i,f^b(x^a_i))$, $(x^b_i,f^b(x^b_i))$, $(x^c_i,f^b(x^c_i))$;

      $(x_i)^c_{max}$ using three pairs

        $(x^a_i,f^c(x^a_i))$, $(x^b_i,f^c(x^b_i))$, $(x^c_i,f^c(x^c_i))$;

    where x^a_i, x^b_i and x^c_i are the ith compenent of the vectors $\mathbf{x}^a$, $\mathbf{x}^b$ and $\mathbf{x}^c$ respectively,

    - add
      $\mathbf{x}(\mathbf{x}^a,x_i)_{max}=\mathbf{x}(\mathbf{x}^a,(x_i)^a_{max})$,

      $\mathbf{x}(\mathbf{x}^b,x_i)_{max}=\mathbf{x}(\mathbf{x}^b,(x_i)^b_{max})$,

$\mathbf{x}(\mathbf{x}^c, x_i)_{max} = \mathbf{x}(\mathbf{x}^c, (x_i)^c_{max})$ to G_t

　}

}

- **Classical Crossover:** Generate $2*\lceil P*C_r \rceil - L*3*N$ chromosomes through classical two point crossover and add them to temporary population G_t in addition to G_n,
}
- **Selection:** Select P-1 chromosomes from G_t by roulette-wheel rule and add the best chromosomes of G_t to generate G_{n+1}

until Stopping Condition $S(G_{n+1})$ is satisfied

APPENDIX B

ANGLE MEASUREMENT DEVICES

Two angle sensors are used in the test setup: Encoder (Heidenhain ROC 417) and arc segment potentiometer. ROC 417 is an absolute, single turn rotary encoder with its own bearing, whose picture is given in Figure B-1. It has both incremental and absolute angle measurement. It has 8192 line counts that are used for its incremental measurements. Absolute signals are digital with 17 bits in EnDat (**En**coder **Da**ta) data format. EnDat interface makes it possible to transmit position values as well as parameters (e.g. datum shift) quickly and reliably with only 4 signal lines. However extracting the position data from the encoder requires designing an electronic card, which sends the clock and mod signals and receives position data. This card is also designed, manufactured and tested.

Figure B-1 Heidenhain ROC 417 encoder

Summary of its specifications are also given in Table B-1.

Table B-1 ROC 417 encoder specifications

ROC 417	
Incremental Signals	1 V_{PP} (Sine shape signal)
Line Counts	8192
Absolute position values	EnDat
Positions/rev.	131072 (17 bits)
Power supply	5 V
Operating temperature	Max. 80 ˚C

Arc segment potentiometer is the other angle sensor, whose drawing is given in Figure B-2.
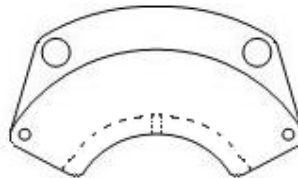

Figure B-2 Drawing of arc segment potentiometer

Its arc shape makes it special for applications which don't have full rotation, like in the fin actuation systems. In this way less space is occupied by the angle sensors. Its specifications are given in Table B-2.

Table B-2 Arc segment potentiometer specifications

Electrical angle	± 35˚ min
Resistance	2500 ± 250 ohms
Resolution	Infinite
Output Smoothness	0.1 % max
Starting torque	2.0 oz.in max
Backlash	0.05˚ max