DESIGN AND PERFORMANCE EVALUATION OF NEURAL NETWORK CONTROLLER APPLICATION FOR AUTOPILOT OF AN UNMANNED AERIAL VEHICLE (UAV)

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

MASTER OF ENGINEERING

IN

CONTROL AND INSTRUMENTATION

SUBMITTED BY

P. ARACHELVI

[ROLL NO. 8673]

UNDER THE ESTEEMED GUIDANCE

OF

DR. PARMOD KUMAR
(PROFESSOR, HEAD - EED)

DEPARTMENT OF ELECTRICAL ENGINEERING
DELHI COLLEGE OF ENGINEERING
UNIVERSITY OF DELHI
BAWANA ROAD, DELHI -110042, INDIA
JUNE 2006

CERTIFICATE

This is to certify that the work being presented in this dissertation entitled **"DESIGN AND PERFORMANCE EVALUATION OF NEURAL NETWORK CONTROLLER APPLICATION FOR AUTOPILOT OF AN UNMANNED AERIAL VEHICLE (UAV)"**, in partial fulfillment of the requirement for the award of the degree of Master of Engineering in Control and instrumentation Engineering submitted by P.ARACHELVI (18/C&I/04), University Roll No. 3683 to the Department of Electrical Engineering, Delhi College of Engineering, Delhi, is the record of the student's own work carried out under my supervision and guidance.

Date:

(Dr. Parmod Kumar)
Professor & HOD
Deptt. of Electrical Engg
Delhi College of Engineering
Delhi - 110042

ACKNOWLEDGMENT

This report, as we see today is an outcome of persistent effort and a great deal of dedication and it has drawn intellectual and moral support from various people within the institution. I am extremely indebted to my honorable guide Dr. PARMOD KUMAR, Prof. and Head of the Department, Department of Electrical Engineering for his invaluable support and guidance throughout the development of this project. It is a matter of great pride for me to have worked under him, which in itself was a source of inspiration for me to complete the project with great enthusiasm, energy and determination. I also give extra special thanks to him for dedicating his valuable time whenever I needed to discuss project related work.

I would like to thank to Dr. Smirithi Srivastava, Assistant Professor, NSIT, Delhi for her valuable guidance and help. I take this opportunity to thank all members of Electrical Engineering Department for their valuable help in this project I am also thankful to my family members and all my friends who continuously helped and motivated me during the course of this dissertation.

(P. ARACHELVI)
M.E. (Control &Instrumentation)
Roll No. 18/ C &I/ 04
University Roll No. 8673
DCE, Delhi-110042

ABSTRACT

This work is aimed at designing Neural Autopilot controllers for Pitch and Roll channels using linear aircraft model, created with MATLAB / SIMULINK tool-box and investigation of the Model Response during various phases of flight like climb, cruise and landing using MATLAB. The behavior of aircraft can be described using a set of non-linear differential equations, assuming six degrees of freedom (3 linear motions and 3 angular motions) about x, y & z axes. The solutions to these set of equations are very complex and for investigation purpose they can be approximated to linear differential equations assuming 3 degrees of freedom for the subject aircraft model. The input to the system for pitch attitude are U, α,θ, θ_dot and h. The output to pitch control channel is δe , the elevator deflection. The input to the system for roll attitude are β,ϕ,ρ,Ψ and r. The output to roll control channel is δr , the aileron deflection. The simulation data in the above study are converted to discrete space data and used for training neural network controllers based on different architecture and method of training. These neural network controllers will adapt the behavior of conventional controllers in pitch/roll/yaw axes. An analysis of the performance is carried out using trained neural networks and comparative performance study is also done.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

F, M	Forces and moments acting on the aircraft
U, V, W	Linear velocities along x, y ,z axes
P, Q, R	Angular velocities along x, y ,z axes
L, M, N	Moments about x, y, z axes
J	Cross product of inertia
Ixx	Moment of inertia about x-axis
Iyy	Moment of inertia about y-axis
Izz	Moment of inertia about z-axis
ζ_p	Phugoid mode damping
ζ_s	Short period damping
ζ_D	Dutch roll damping
ω np	Phugoid mode natural frequency
ω ns	Short period natural frequency
ω nd	Dutch roll natural frequency
τs	Spiral mode time constant
τr	Roll subsidence time constant
h	Altitude/Height
p	Roll rate
q	Pitch rate
r	Yaw rate
φ	Roll angle
θ	Pitch angle
ψ	Yaw angle
α	Angle of attack
β	Sideslip angle
γ	Flight / Glide path angle
ho	Flare entry height
λ	Angular deviation from the centerline of the runway
δe	Elevator deflection
δa	Aileron deflection
δr	Rudder deflection

$C_{L\alpha}$	Lift curve slope
$C_{m\alpha}$	Pitching moment due to angle of attack
C_{mq}	Pitching moment due to pitch rate
C_{mv}	Pitching moment due to free stream velocity
$C_{m\alpha}$_dot	Pitching moment due to rate of change of angle of attack
C_{dv}	Drag co-efficient due to free stream velocity
$C_{d\alpha}$	Drag co-efficient due to angle of attack
$C_{n\beta}$	Yawing moment due to slide slip
C_{np}	Yawing moment due to rolling velocity
C_{nr}	Yawing moment due to yawing velocity
$C_{n\delta a}$	Yawing moment due to aileron deflection
$C_{n\delta r}$	Yawing moment due to rudder deflection
$C_{l\beta}$	Rolling moments due to sideship
C_{lp}	Rolling moment due to rolling velocity
C_{lr}	Rolling moment due to yawing velocity
$C_{l\delta a}$	Rolling moment due to aileron deflection
$C_{i\delta r}$	Rolling moment due rudder deflection
$C_{m\delta e}$	Elevator effectiveness
C_{xu}	Force in x-direction due to forward velocity
C_{zu}	Force in z-direction due to forward velocity

CHAPTER 1
INTRODUCTION

1.1 PREFACE

The design of an aircraft involves solving a set of non linear equations of motion in six degrees of freedom (three linear motions and three angular motions). The task of solving control problem of airspace vehicles is more complex, since these are required to operate over a wide rage of altitude, air-speed and maneuvers. In addition environment disturbance factors affect aircraft performance to a large extent at lower altitudes. Also in practical terms, stability and manoeuvrability are complementary to each other in aircraft design. So, the aircraft behaviour is unstable in certain operating conditions. Due to non– linearties involved in equations of motion, design of control system components for aircraft is a more challenging task.

Due to advancement in technology, aircraft controls have become more and more sophisticated. Mechanical controls have been replaced with more powerful hydraulic actuators. Today, specialized flight control computers and more sophisticated electro-hydraulic actuators and sensors have been successfully employed in aircraft control in 'fly-by-wire' architecture.

The evolution of UAVs (Unmanned Aerial Vehicle) is also linked to the advancements in technology in the following areas.

- Production of light composite materials used in airframe construction,
- Electrical Flight control system components of fly-by-wire design.
- Advanced flight control computers using modern hardware and software.
- Availability of simulation techniques using software tools like MATLAB and AEROSPACE SIMULATION PROGRAMS conducted by leading aerospace research organizations and defense research establishments.

- Complex and non-linear control problems have been solved using powerful software tools and advanced techniques like fuzzy logic and neural network.
- Today, controllers in many engineering control applications are being designed using neural network architecture and learning algorithms which simplify the process of controller design and take care of most of the complex nonlinear problems. Once programmed, these controllers adapt to changes in system response over time and in few cases are designed to be failure tolerant . The same techniques are being applied in the design of controllers employed in airspace vehicles .

1.2. DESCRIPTION

Though extensive application of neural network technique is found in areas like pattern recognition and intelligent data-processing like data compression, encryption, decoding etc, its application to control systems too is steadily increasing. In this project, using MATLAB / SIMULINK control system / neural network toolboxes and programming, neural network controller design for pitch , roll and landing autopilot control of a model UAV is explored and compared to those of conventional controller design. The salient features of an UAV are:-

- It is fully dependent on onboard controllers and instruments for monitoring and for operating.
- It is intend to be monitored and controlled by remote operator through radio links.
- It is intended to be fully controllable in its flight envelope by on board autopilot controllers.
- It is a light vehicle designed with optimum maneuverability and optimum stability.

These points make the design of autopilot controllers a special case. It is desired to design and study the performance of Neural Network controllers in auto-pilot controllers of an UAV. The work is outlined below.

1.3 OUTLINE OF THE THESIS

In this work, the following methodology is adopted, using MATLAB / Simulink tools:

- First, using state space models, the open-loop response of aircraft in pitch and roll axes are investigated.
- Then PID controllers are introduced in the control loop of pitch and roll axes and the closed loop responses are investigated.
- The PID controller parameters are tuned using Ziegler – Nichols methods for optimum- performance.
- Three layer feed – forward and three layer cascaded feed-forward neural networks are chosen and trained using back – propagation algorithm in MATLAB / Simulink neural network toolbox.
- Requisite training data is generated using model-reference open-loop simulation in pitch and roll and used for training individual neural network controllers for pitch and roll channels.
- Once trained, these controllers are introduced in pitch and roll autopilot closed-loop channels and their performance is evaluated using simulation.
- Next performance of neural network controller in landing phases on glide slope and flare control are studied.
- Finally, a comparative performance analysis for Conventional controllers and
- Neural Network controllers is provided.

1.4. DISSECTION OF DISSERTATION

Chapter-2 presents literature overview about UAVs and neural Net Work theory and applications.

Chapter-3 presents basic aircraft dynamics in 6 degrees of freedom (6-DOF) and de-coupled form in 3-DOF. Derivation of longitudinal (pitch-axis) and lateral (roll-axis) equations of motion and their characteristics , aspects of important stability and control derivatives are explained. Derivation of state-space models in pitch and roll are also presented in this chapter.

Chapter-4 and 5 ,Automatic Flight control system (AFCS)

And autopilots are explained. UAV flight control system architecture is also explained.

Chapter-6 presents conventional auto-pilot architecture using PID controller in pitch and roll. Aircraft Landing system architecture is also explained

Chapter-7 presents MATLAB/ SIMULINK study of conventional autopilot

performance in pitch, roll and landing phases.

Chapter-8 gives a brief introduction to Artificial Neural Networks and their scope in developing autopilot controllers for aircraft.

Chapter-9 presents simulation study results of comparative performance analysis between conventional and neural network autopilot controllers.

CHAPTER 2
LITERATURE REVIEW

2.1. HISTORICAL DEVELOPMENT OF NEURAL NETWORKS.

- ✍ **1943 – McCulloch and Pitts: start of the modern era of neural networks**
  This form of logical calculus of neural networks. A network consists of sufficient number of neurons (using a simple model) and properly set synaptic connections can compute any computable function. A simple logic function is performed by a neuron based upon the weights set in the McCulloch-Pitts neuron. The most important feature of this type of neuron is the concept of threshold. When the net input to a particular neuron is greater than specified threshold by the user, then the neuron fires. Logic circuits are found to use this type of neurons extensively. An explicit statement of a psychological learning rule for ***synaptic modification*** was presented .
- ✍ **1949 – Hebb's book " The Organization of behavior "** suggesting the continuous changing connectivity of brain neuron cells are the basis of learning.
- ✍ **1958 – Rosenblatt introduces Perceptron (Block [1962], Minsky and Papert [1988])** Introduced the concept of adjusting connection weights to reproduce target outputs. They showed that Large number of input samples and corresponding targets achieved convergence, suggesting 'learning' property.
- ✍ **1960 – Widrow and Hoff introduce ADALINE.** Abbreviated from Adaptive Linear Neurons , ADALINE uses LMS (least mean square) or delta rule. Net weights are adjusted to minimise difference between the Q (net-input minus net-output) and T(desired output), using mean square error reduction rule for convergence.

- ✍ **1982 – John Hopfield's networks .** Based on dynamically stable and associative memory networks with 'ísing spin glass' type of model this work showed the 'memory' function of neural net. This useful work led to modeling in systems.
- ✍ **1972 – Kohonen's Self Organizing Maps (SOM)** Making use of topographical mapping on data structures, this work showed the method of reproducing brain functions like recall and reproduce. It found applications in pattern recognition .
- ✍ **1985 – Parker, 1986 – Lecum:** Important work introduced 'back propagation algorithm' and led to its application in many multi-layer network design .
- ✍ **1988 – Grossberg.** This work which introduced application of counter propagation net, uses learning rule similar to Kohonen's self organising maps. Training affects all cell layers without any competition between them.
- ✍ **1987, 1990 – Carpenter and Grossberg.** Introduced Adaptive Resonance Theory (ART) for both digital and continuous systems. The ART net can accept input pattern in any order or sequence.
- ✍ **1988 – Broomhead and Lowe.** Introduced RBF or Radial Basis Functions similar to back propagation algorithm applied on multi-layer neural net.
- ✍ **1990 – Vapnik** developed the support vector machine.
- ✍ **1999-Maass .W.** and **Bishop:** Introduced Spiking Neuron network. With sigmoid or linear transfer function for hidden layer neurons, the network can approximate any function or behave like 'universal approximator'.
- ✍ **2003- Hasler .P. and Duggar.J.-** Explored hardware development of neural networks using electronic hardware like transistor / MOSFET/Analog VLSI circuits.

2.2. JOURNALS AND ORGANIZATIONS

Journals and organizations involved in the advancement of neural network applications, research and development are listed below:

- IEEE Neural Network Homepage
- International Neural Network Society and Networks journal
- IEEE Transactions on neural networks
- IEEE Transactions on Fuzzy systems
- IEEE transaction on evolutionary computation
- Neural Computation

2.3 SOFTWARE RESOURCES AVAILABLE ON THE NET

There is a continuous contribution to neural network theory in the software sector, in line with the research and development work. A vast amount of application software and tools are available on – line. Below is the list of some leading groups in this area:

- ***PDP ++:*** *Parallel Distributed Processing* software in C++
- ***NETLAB Toolbox:*** An extensive MATLAB toolbox for statistical pattern and Neural Networks
- ***SNNS, JavaNNS Simulators:*** *Stuttgart Neural Network Simulator* in C and Java
- ***SOFM Toolbox:*** A MATLAB toolbox for SOFM Simulation Helsinki University Research Laboratories.
- ***GENESIS :*** *General Neural Simulation System Caltech.*
- ***Hodgkin – Huxley Toolbox :*** A MATLAB toolbox for Hodgkin – Huxley model simulation.
- ***SVM toolbox :*** *Support Vector Machine* MATLAB toolbox.

2.4. EVOLUTION OF UAV

The Aircraft history dates back to the beginning of 20th century. Initial designs were lacking aspects of stability, maneuverability and automatic control. Concept of Autopilot was first introduced in 1927 and after the world wars there was a rapid growth in aviation in the areas of materials , designs , turbojet engines, control system components and aviation electronics. Autopilots evolved with more sophistications. Sensors and instruments also saw a radical change in their design and performance. The first experimental automatic landing system was introduced in 1943. In 1949 foundation for the application of adaptive control theory in aircraft systems was laid. This led to application of conventional autopilots employing adaptive control theory. Serious nonlinear and stability problems were successfully tackled using 'gain scheduling' and 'robust controller' designs. In 1950s AFCS architecture were developed using Model-Reference Adaptive control systems. In1960s , Stochastic problems arising due to atmospheric disturbances were investigated and tackled. With the advancements in semiconductor technology, more electronic hardware were introduced in aircraft applications. In 1980s autopilots were introduced in the navigation loops . Using either inertial navigation or radio navigation , the aircraft could be commanded by onboard computers over long inter continental flights.

Development of advanced LSI and VLSI semiconductors for analog and digital applications, readily saw an impact in aviation electronics. More robust and sophisticated control and navigation equipment with far low weight and power consumption are steadily evolving, even today. With advent of GPS and other terrain navigational aids the aircraft can fly virtually pilot-less. With all this developments in the background there was an increasing demand for pilot-less vehicles for espionage, reconnaissance, scientific aerial expeditions and commercial applications in civil survey, agriculture and forestry .With the availability of advanced light weight composite materials, efficient propulsion systems and low power consuming electronic hardware, today, UAVs are viable to produce commercially. The main requirements for UAV application are fully autonomous onboard computers for control, navigation and telemetry. This makes demand for artificial intelligence applications like fuzzy logic and Neural Networks in the development of these components. A large amount of technical and web resources are available today in this area. Some of the important technical references are listed below:

Reference :1. Bringing command and control of UAVs Down to Earth. A paper by Benjamin Bell of CHI Systems,Inc and John Clark , Lockheed-Martin Aeronautics Co.--IEEE 2002.

3. IEEE paper on robust control applications in UAV control (2004) by Hiroki Nakanishi and Koichi Inoue. Discussion on ways to tackle Stochastic uncertainty to wind speed, Robust control system design by use of neural networks and its application to UAV flight control components.

4.Survey of Unmanned Aerial Vehicle , Richard M. Howard and Isaac Kaminer. Proceedings of American control conference, Seattle, Washington, June 1995.

5. Development of flight test system for UAV by Isaac Kaminer, Eric Hallberg, Antonio Pascoal, IEEE Transaction on control systemFeb-1999.

6.Artificial Intelligence and Expert systems for Avionics. By Lee H. Harrison and Pamela J Saunders of Galaxy Scientific Corporation; NJ and Peter J Saraceni, Jr of FAA Technical Centre , NJ.—IEEE control systems journal July 1993.

7. A paper on self-organizing Fuzzy-neural network based Autopilot system for Automated vehicle. Pergamon article on neural network Volume – 14, 2001, Page 10992-1122.

8. AI technique using Fuzzy based applications techniques in UAV: Warren R. Dufrence Jr. Nova South-Eastern University. IEEE A&E Systems Magazine , August-2004.

9. Design of reconfigurable Automated Landing System for VTOL UAVs by Michael Bole, Research Associate and Jaroslov Svoboda Director of Aerospace Programs. Concordia Univ. Montreal; Canada .

10. Intelligent Control Theory in Guidance and control system design: an overview. By Chun-Liang Lin and Huai –Wen Su Proc of National Scientific Council , ROC(A) ; Vol-24 No-1, 2000,pp 15 to 30.
11. Robust Neuro- H∞ Controller design for aircraft auto landing by Yan Li, N. Sundara Rajan, P.Saratchandran and Zhi Feng Wang ; Nanyang Tech. Univ. Singapore. IEEE Trans. on A&E Systems Vol-40 Jan-2004. page 158 to 167.

CHAPTER-3
AIRCRAFT DYNAMICS

3.1. THE BASICS OF AIRCRAFT DYNAMICS

An aircraft has six degrees of freedom (6DOF). They are Linear motions about x, y, z axes and Angular motions about x, y, z axes.

Assuming x, y ,z co-ordinates ,these motions can be represented using laws of motion as follows:

Let the centre of co-ordinates be located at the centre of gravity of the aircraft. Then the linear motions in x, y, z co-ordinates can be written as

ΣΔFx = m (U_dot + WQ -- VR)

ΣΔFy = m (V_dot + UR -- WP)

ΣΔ Fz = m (W_dot + VP – VQ)

where

Fx = force on x-axis

Fy = force on y-axis

Fz = force on z-axis

U,V,W - Linear velocities along x, y, z axes.

P,Q,R - Angular velocities along x, y, z axes.

m - mass of aircraft.

Similarly angular motions about x, y and z axes can be written as

ΣΔL = P_dot Ix _ R_dot Jxz + QR (Iz – Iy) – PQ Jxz

ΣΔM = Q_dot Iy + PR(Ix – Iz) + (P^2- R^2) Jxz

ΣΔn = R_dot Iz – p_dot Jxz + PQ (Iy – Ix) + QRJxz

where,

L, M, N - moments about x, y, z axes

Ix, Iy, Iz - moment of inertia about x, y, z axes

Jxz - cross product of inertia

The above six equations are nonlinear in type and their solutions are complex in nature. Using digital computer the above six equations can be modeled in difference equation form. A complete solution for the given range of aircraft maneuvers like velocities, attitudes etc. involves very large computations. These six equations can be de-coupled into two sets of three equations each and approximated to linear equations to obtain numerical solutions for small-disturbances . Since the Changes in airspace parameters like altitude, temperature and wind velocities do not change quickly as compared to aircraft dynamics they can be assumed constants. Practically this assumption holds true and the model can be reduced to linear form with a good degree of accuracy. This is done so because a majority number of autopilots operate assuming linear models and their responses are restricted to a limited range.

The two set of de-coupled equations are called

1.Longitudinal equations of motion

2.Lateral equation of motion

Our investigation is limited to these de-coupled linear equations.

3.2.LONGITUDINAL EQUATIONS OF MOTIONS

In this longitudinal mode, the aircraft is considered to be in straight, level and steady (un-accelerated) flight and then to be disturbed by deflection of the elevator. This deflection applies a pitching moment about the OY axis causing a rotation about the axis, which eventually causes a change in Fx and Fz, but does not cause a rolling or yawing moment or any change in Fy. Thus we can assume that the angular velocities about x and z axes (P and R) and linear velocity about y axis (V) are negligible and can be omitted.

Hence, $\sum$ Fy, $\sum \Delta$L, $\sum \Delta$N equations may be eliminated.

Therefore, the longitudinal equations of motions can be written as

$$\sum \Delta F_x = m\,(U\_dot + WQ)$$

$$\sum \Delta F_z = m\,(W\_dot - UQ)$$

$$\sum \Delta M = \textbf{Q\_dot Iy}$$

Neglecting small perturbations and other external influences. These equations an be considered as linear and solved for parameters. The final equations will be in terms of stability derivatives and form these equations the transfer-functions of the aircraft in longitudinal mode are derived. Solutions of three longitudinal equations of motion will a quadratic equation and can be written as:

$$(s^2 + 2\zeta_p\omega_{np}\, s + \omega^2_{np})\ (s^2 + 2\zeta_s\omega_{ns}\, s + \omega^2_{ns})$$

where

ζ_p, ω_{np} phugoid mode of damping and natural frequency

ζ_s, ω_{ns} - natural short period damping and natural frequency.

3.3.OPEN-LOOP LONGITUDINAL DYNAMICS

Inspections of the quadratic equation

$$(s^2 + 2\zeta_p\omega_{np}\, s + \omega^2_{np})\ (s^2 + 2\zeta_s\omega_{ns}\, s + \omega^2_{ns})$$

reveals that the roots of this characteristic equation decides the behavior of fixed wing aircraft. For a practical aircraft model, these roots are characterized in two oscillation.

They are:

1. Short period with relatively heavy damping and high frequency oscillations.
2. long period (phugoid) with very light damping. and low frequency oscillations.

On inspection of the above equation we can conclude that the characteristic of these oscillation turn depend on the aircraft design parameters like mass, moment of inertia and geometry of flight. The short period oscillation cause variations in angle of attack α and pitch angle θ and very little change in forward speed u. (u_dot $\approx$ 0) The long period (phugoid) oscillation occurs at a constant angle of attack. It is characterized by long period and low damping constant.

3.4. LATERAL EQUATIONS OF MOTIONS

Aircraft behavior in lateral mode is different from behavior of longitudinal mode.

$$\sum\Delta F_y = m\,(V\_dot + VR - WP)$$

$$\sum\Delta L = P\_dot\ Ix - R\_dot\ Jxz + QR\,(Iz - Iy) - PQ\ Jxz$$

$$\sum\Delta N = R\_dot\ Ix - P\_dot\ Jxz + PQ\,(Iy - Ix) + QR\ Jxz$$

For steady longitudinal conditions and small perturbations, these equations can be simplified to:

$$\sum\Delta L = P\_dot\ Ix - R\_dot\ Jxz$$

$$\sum\Delta F_y = m\,(V\_dot + VR)$$

$$\sum\Delta N = R\_dot\ Ix - P\_dot\ Jxz$$

Examination of these equations, we note the following:

Rolling moment or yawing moment induces angular velocities in all the three axes. It is assumed that pitching moment Q= 0 (de-coupled) for solving the lateral equations of motions.

3.5.OPEN-LOOP LATERAL DYNAMICS

The solution of these equations can be generated as a characteristic equation given below:

$$(S + 1/\tau s)\ (S + 1/\tau r)\ (S^2 + 2\zeta_D\,\omega_{nD}\,S + \omega_n^2{}_D) = 0$$

The solution of this characteristic equation are the parameter of lateral transfer functions, using all approximations. In the above characteristic equation,

$(S + 1/\tau s)$ characterizes spiral mode component

$(S + 1/\tau r)$ characterizes roll subsidence mode component

$(S^2 + 2\zeta_D\,\omega_{nD}\,S + \omega_n^2{}_D)$ characterizes Dutch roll mode

The Dutch roll mode component basically vary with the flight conditions and may lead to instability if the damping co-efficient ζ_D is small, that is, lightly damped case. Other two components are of higher frequency and higher damping. The roll subsidence is the rolling response of the aircraft to an aileron input. The spiral divergences is of very high time constant leading to slowly varying divergence and can be easily controlled.

3.6. STABILITY AND CONTROL DERIVATIVES

To obtain the transfer functions of the aircraft, it is necessary to identify influential components and define their derivatives, relating the changes in aerodynamic forces and moments acting on aircraft. These changes are caused by aircraft motion about three axes, momentum about three axes as well as control surface deflections. The stability derivatives are the dimensionless coefficients which determine the stability of the aircraft by virtue of the design.

These are:

1. $CL\alpha$ - Lift curve slope
2. $Cm\alpha$ - pitching moment due to angle of attack.
3. Cmq – pitching moment due to angle pitch rate.
4. Cmv – pitching moment due to free stream velocity.
5. $Cm\alpha$_dot – pitching moment due to rate of change of angle of attack.
6. Cdv – Drag co-efficient due to free stream velocity.
7. $Cn\beta$ - Yawing moment due to slide slip.
8. Cnp – Yawing moment due to rolling velocity.
9. Cnr – Yawing moment due to yawing velocity.
10. $Cn\delta a$ – Yawing moment due to aileron deflection.
11. $Cn\delta r$ – Yawing moment due to rudder deflection.
12. $Cl\beta$ - Rolling moments due to sideslip
13. Clp – Rolling moment due to rolling velocity.

14. Clr – Rolling moment due to yawing velocity.
15. Clδa – Rolling moment due to aileron deflection.
16. Ciδr – Rolling moment due rudder deflection.

17. Cmδe- Elevator effectiveness.
18. Cxu- Forces in X-Direction due to forward velocity.
19. Czu- Forces in Z-Direction due to forward velocity.

3.7. INFLUENCE OF CONTROL AND STABILITY DERIVATIVES ON AIRCRAFT

The effect of these coefficients on aircraft performance is given in the following tables:

Table-3.1. Effect of stability derivatives in longitudinal Mode

Sl.No	Stability Derivative	Affected parameter	Nature of effect
1.	Cmq	ζs -Short period Damping coefficient	ζ s increases with increase in Cmq
2.	Cmα	ωns - Short period Natural frequency.	ωns increases with increase in Cmα
3.	Cxu	ζp- damping of the phugoid oscillations	ζ p increases with increase in Cxu
4.	Czu	ωnp-Natural frequency of phugoid oscillations	ωnp increases with increase in Czu

Table-3.2. Effect of stability derivatives in lateral Mode

Sl.No	Stability Derivative	Affected parameter	Nature of effect
1.	Cnr	ζD- Damping Coefficient of Dutch roll	ζ s increases with increase in Cmq
2.	Cnβ	ωnD- Natural frequency	ωns increases with increase in

		of Dutch roll	$C_{m\alpha}$
3.	C_{lp}	$(1/\tau r)$ Roll subsidence	$(1/\tau r)$increases with increase in C_{lp}
4.	$C_{l\beta}$	$(s+1/\tau s)$ Spiral divergence	$(s+1/\tau s)$ increases with increase in C_{zu}

3.8. DERIVATION OF TRANSFER FUNCTION AND THE STATE- SPACE MODEL

The small perturbations linear model consists of 12 state variables. The position variables x, y and z do not affect stability, since. a change in these variables occur very slowly compared to other states, they can be assumed constant. The remaining 9 variables are autonomous and affect aircraft behavior. These are:

u – velocity in x axis

α - angle of attack

β - sideslip angle

p – Roll rate

q – pitch rate

r – yaw rate

ψ - yaw angle

θ - pitch angle

ϕ - Roll angle

x, y, z state variables can be neglected (they have no effect on aircraft dynamics). The General State-space Equation can be written as follows:

$$X_{dot} = Ax + Bu$$

$$Y = Cx + Du$$

Since decoupled equations are considered, the above state space model is created for separately for Longitudinal (or pitch)axis and Lateral (or roll) axis.

3.8.1.Longitudinal State-Space Model

Longitudinal State-space model is derived for pitch responses using the following transfer functions :

1. $u/\delta e$: Speed to elevator deflection
2. $\alpha/\delta e$: Angle of attack to elevator deflection
3. $\theta/\delta e$: Pitch to elevator deflection
4. $q/\delta e$: Pitch rate to elevator deflection
5. $h/\delta e$: height to elevator deflection

A is a state matrix containing elements of transfer functions

[$u/\delta e$; $\alpha/\delta e$;$\theta/\delta e$; $q/\delta e$; $h/\delta e$] .

B is a Vector of initial conditions of elevator input [δe] .

C is an Identity matrix[I] .

D is taken as null vector assuming all initial conditions at zero.

3.8.2.Lateral State-space model

The lateral State-space model can be constructed using the following transfer functions

$\beta/\delta a$: Side-slip to aileron deflection

$\phi/\delta a$: Roll angle to aileron deflection

$p/\delta a$: Roll rate to aileron deflection

$\psi/\delta a$: yaw to aileron deflection

$r/\delta a$: yaw rate to aileron deflection

$\beta/\delta r$: Side-slip to rudder deflection

$\phi/\delta r$: Roll angle rudder deflection

$p/\delta r$: Roll rate to rudder deflection

$\psi/\delta r$: yaw to rudder deflection

$r/\delta r$: yaw rate to rudder deflection

For lateral (or roll) axis ,

A is a state matrix containing elements of transfer functions

$$[\psi/\delta a: \psi/\delta r{:}\phi/\delta a: \phi/\delta r: \beta/\delta a : \beta/\delta r]$$

B is an input vector with initial conditions in pitch and roll[δa: δr]

C is an Identity matrix[I]

D is taken as null matrix [$\delta a=0$ $\delta r=0$]

This state-space model in longitudinal, lateral and vertical axes are used in MATLAB / SIMULINK tool box around a given control model, to study the response of UAV in pitch , roll and yaw. Library functions in MATLAB/Simulink are used to create the above transfer functions.

The state space matrices are given in Appendix.

CHAPTER 4

UNMANNED AERIAL VEHICLE (UAV)

4.1 INTRODUCTION

An Unmanned Aerial Vehicle is essentially designed with high degree of stability and required controllability for take-off, climb, cruise, descend, and landing maneuvers, using radio controllers. The on-board autonomous controllers are designed for carrying out mission through radio link and correct position and velocity errors. Most Unmanned Aerial Vehicle rely on Global Positioning System (G.P.S.) for position tracking and solid-state attitude and rate gyros for stabilizing pitch and role altitudes.

4.2 SALIENT FEATURES OF UAV

With advancement in solid state semiconductor technology, the size, weight and power consumption of air data and attitude sensors have come down to a great extent . These sensors apart from above advantages also include elaborate built-in test programs and fault tolerant design features making it suitable for on-board computers in an Unmanned Aerial Vehicle.

An Unmanned Aerial Vehicle mission program is edited on-line from ground station and the vehicle is maneuvered through radio-link. The vehicle also transmits its position, velocity, air-speed and altitude data, making it easier for a ground controller. Apart from the above other instrumentation data like weather data ,aerial photographs, volcano samples and movement of migratory birds have been successfully collected using Unmanned aerial vehicle. It is also noticeable that a large amount of Unmanned Aerial Vehicles are deployed for the purpose of military reconnaissance. Now the most demanding job of a non-board autopilot controller of an Unmanned Aerial Vehicle is to land the vehicle using ground navigational aids like glide path and localizer employed in all civil and military aerodromes. The landing task of an Unmanned Aerial Vehicle is more difficult than take off climb and cruise, since the external disturbances are more influential during landing phase of the airplane.

The salient features of an UAV are:-

1.It is fully dependent on onboard controllers and instruments for monitoring and for operating.

2. It is intend to be monitored and controlled by remote operator through radio links.

3. It is intended to be fully controllable in its flight envelope by on board autopilot controllers.

4. It is a light vehicle designed with optimum maneuverability and optimum stability.

4.3 DESCRIPTION OF UAV

This is an all composite, pusher- propeller aircraft in class-I category powered by an indigenous engine. It has a wing with two stakes and a high-aspect ratio canard with very small chord. the canard has a fill- span flap, which acts as an elevator. Hence, $Cm\delta e$ and $Cl\delta e$ is positive. The high aspect ratio canard is designed to carry almost 30% of the total aircraft lift resulting in a low trim angle-of-attack and consequently in a very low trim drag. The aircraft has an exceptionally high L/D ratio and since the engine has a low Specific Fuel Consumption , the aircraft has very good endurance and range. There is no conventional vertical tail or horizontal tail(empennage), in the UAV. Roll control is provided by ailerons, which are located at outboard stations on the wing. Two fins (with rudder) are located at the wing tips ,which provide directional stability and control. These wing-tip fin provide beneficial end plate effects on the wings increasing its effective aspect ratio ,thereby improving the aerodynamic efficiency of the wings. The fuselage has a very large volume available for installing the payloads and for additional fuel storage. The tri-cycle landing gear system consists of two main wheels and a retractable nose wheel , which can be retracted fully into the fuselage to reduce drag during flight. The UAV is capable of being flown and maneuvered under the control of a Ground Control Station / or with the help of an Onboard Automatic Flight Control, Navigation and Guidance System. This UAV is used for surveillance and Reconnaissance purpose. Picture of an UAV is shown in Fig 4.1.

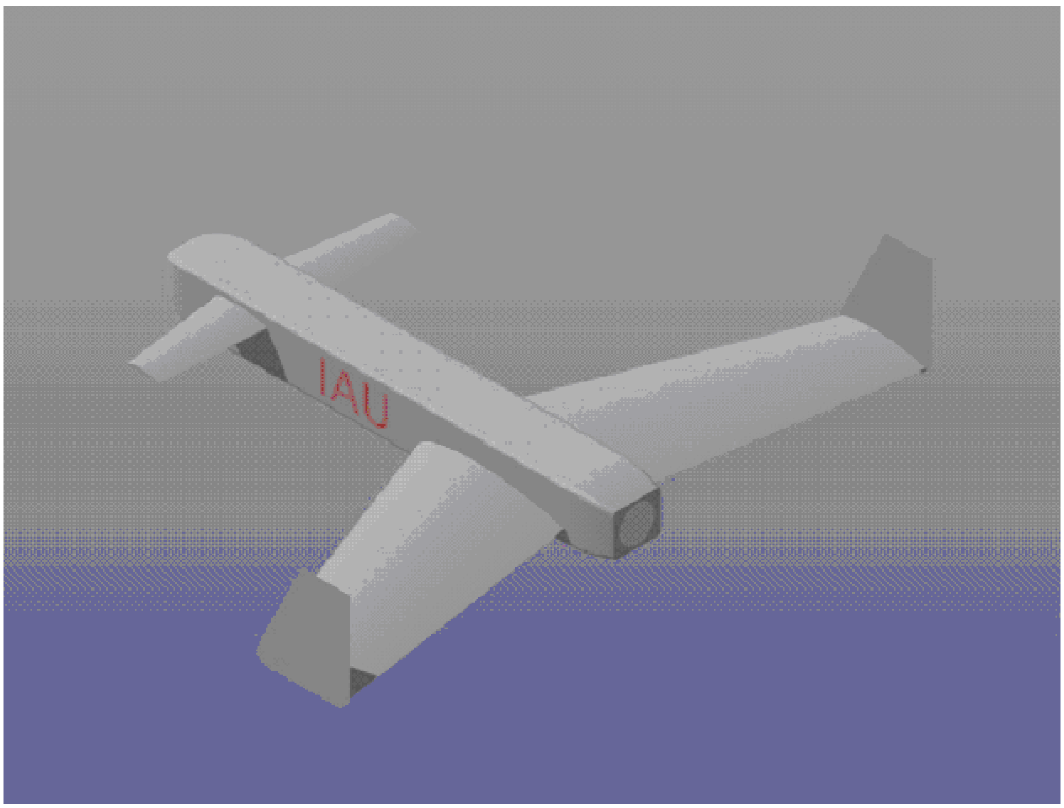


Fig 4.1. Picture of an UAV

CHAPTER 5

AUTOMATIC FLIGHT CONTROL SYSTEM(AFCS)

5.1 INTRODUCTION

So far we have created a state – space model describing the aircraft behavior. This is a linear model which can be analyzed for open-loop step response using MATLAB control system tool box in SIMULINK program.

In our study, we can include a separate transfer function for aileron and elevator actuators. Generally these are second-order systems designed for a particular aircraft. The deflection of aileron and elevator excites the state space model and the response of the aircraft in terms of state variables can be investigated.

In a practical aircraft there is a large variation in altitude and airspeed, which imposes limits to magnitude and rates with which the control surfaces can be deflected for a given magnitude of airspeed and altitude. In other words there is a need to introduce 'Control Laws' which will satisfy this requirement. AFCS is integrated into flight control system and functionally it introduces correction signals at actuator commands through sensor feed-back.

5.2 FUNCTIONS OF AUTOMATIC FLIGHT CONTROL SYSTEM

In a practical aircraft the control surfaces are deflected directly by pilots, who are trained to operate the aircraft in the stable region. However, during trajectory of flight, it is desired to include an automatic control loop for a limited (hold) capability. For example an altitude hold function will correct small disturbances in pitch and stabilize the aircraft in height. For large uncontrollable disturbances, the autopilot control loop is designed to disengage and alert the pilot for a manual take over.

The general motion of the aircraft can be characterized by its velocity vector. If we denote this as x_dot .

then $\dot{x} = f(x, u, \zeta, t)$

where,

x – the position of aircraft in space

u – the control parameter

ζ - the disturbance

t – time

Since aircraft has 6 DOF, the solution to the above equation is complex , depending upon the state function of aircraft. Since in actual flight dynamics the state equations are non-linear in nature, in order to operate the aircraft in a stable envelope, the maneuvers are limited for magnitudes and rates. The aircraft controls are carefully designed using proto-type wind-tunnel models, to incorporate rate-limits and gains, in practical models . Several aircraft models are available for experimental investigations. In our study we shall use one such model for investigation. Most of all unmanned aerial vehicles are designed for stable maneuvers and at the cost of lower maneuverability.

To some degree such an aircraft is stable in state-space and immune to low-amplitude and high frequency disturbances. Investigations of air-space has revealed effect of environmental disturbances encountered by fixed wing aircraft. For a particular aircraft model the influence of disturbances vary with aircraft position vector, velocity vector and attitude vector. This means that an aircraft flying at high altitude at higher subsonic speed in a level flight is more stable than an aircraft flying at low altitude at high pitch and roll rate at low velocities.

The unmanned aerial vehicle requires a flexible autopilot which is adaptive in nature to maneuver the aircraft in the stable region in the entire state-space.

Normally automatic flight control system (A.F.C.S.) employs feedback control to achieve the following :

1. Speed of response in closed-loop better than open loop response.
2. Accuracy in command follow-up is better. The system is capable of suppressing, to some degree, unwanted effects, which have arisen as a result of disturbances affecting the aircraft's flight.

5.3 AUTOMATIC FLIGHT CONTROL SYSTEM (AFCS) ARCHITECTURE

The AFCS is intended to perform the following functions:

1. Collect air data and inertial sensor data in order to apply ' control laws' which limit the magnitude (gain program) and rates of actuator deflections (damping) so that the aircraft is always operated in a stable envelope over the entire flight profile. Air data are the sensor data pertaining to variations in density as a function of altitude, airspeed, mach number (ratio of actual airspeed to that of speed of sound at that altitude), temperature, and angle of attack encountered by wing . Inertial data are those sensor data from attitude gyros , rate gyros and accelerometers pertaining to position, displacement vector and inertial forces acting on aircraft structure.
2. To correct aircraft response in pitch , roll and yaw axes against disturbances caused by inherent non-linearties in aircraft characteristics , sensor and actuator noise and disturbances in airspace. For e.g. the dutch roll introduces oscillations in yaw and roll axes . Yaw damping function of AFCS uses a dedicated rate gyro

sensor and corrects dutch roll by exciting yaw actuator around a closed control loop.

3. in short AFCS acts as an interface between pilots inputs and actuator commands and introduces required damping and gain so that the aircraft maneuvers are stable under all flight conditions. By design AFCS can be interfaced with Autopilot controllers and process autopilot commands as a function of control laws and pass on the commands to actuators.

The general architecture of AFCS is shown in the following figure:

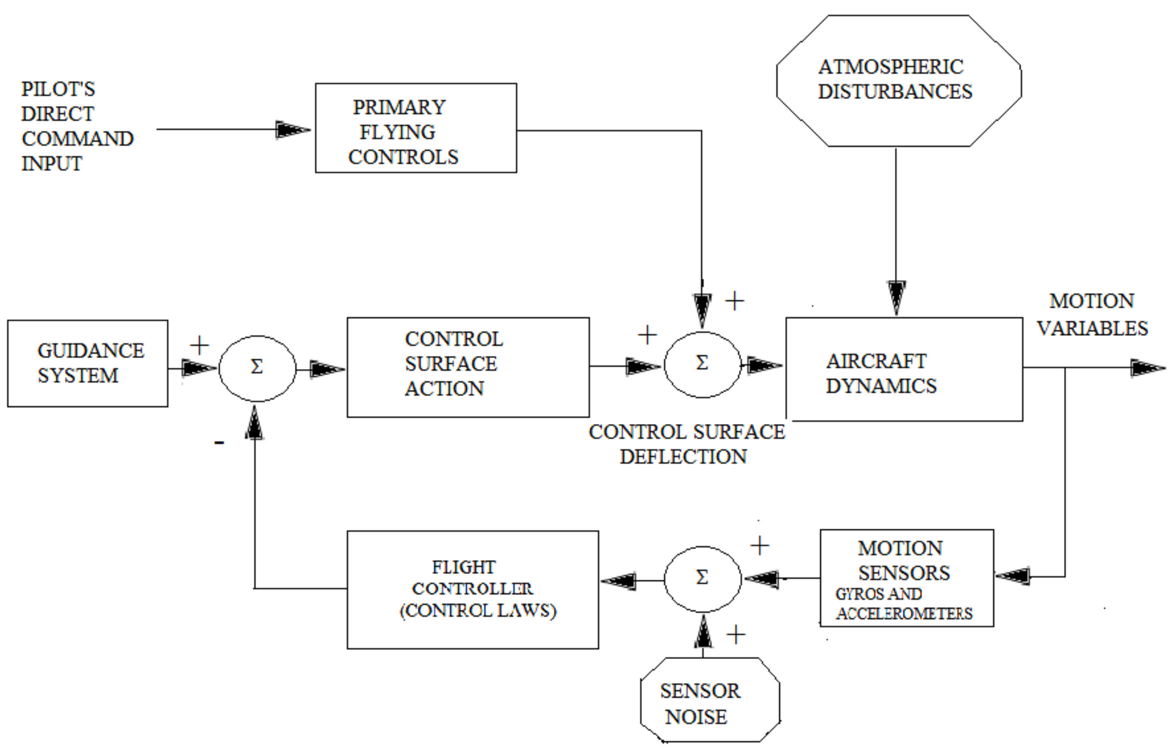


Figure 5.1 General structure of AFCS

CHAPTER-6
AUTOPILOTS

6.1. INTRODUCTION

The basic function of an autopilot is to control the aircraft in flight on pitch and roll in order to steadily maintain a pre-selected pitch or roll attitude or steadily maintain a pre-selected altitude, or heading without the intervention of pilots. They act as command and reference controllers in parallel to pilot inputs. In all designs the pilot input is assigned highest priority over autopilot inputs. i.e. pilot can always over-ride auto pilot commands. In addition to this priority, autopilots engage and operate only over a limited range of operating conditions. They are designed to disengage and alert the pilots in case the amplitude or rate of change of error signals from air data and inertial sensors exceed the design limits.

6.2. AUTOPILOT OF UAVs

In case of UAVs autopilot is the only command unit on board the aircraft. Hence, it is a vital unit and should be designed to perform more reliably and is required to posses learning capability from flight experience. An UAV is required to perform maneuvers continuously and should make necessary corrections in control laws to counter wear and tear in actuators and drifts in sensor and actuator characteristics. It should posses a desired level of failure tolerance against failures in sensor and control components. Modern commercial aircraft employ multiple sensors and actuators and dual AFCS / Autopilot systems to satisfy these requirements.

6.3. DESIGN REQUIREMENTS OF AUTOPILOTS

Rules and guidelines in the design of Autopilot are given below:-

1. Rules

- Always keep the crew informed.
- Do not accept inputs from both the crew and autopilot.
- Issue audio and visual warnings for uncommanded actions.
- The Autopilot should not be given full surface control authority or envelope authority.
- Engagement and disengagement of the autopilot should not result in transient motions.
- The crew should have a single button for lateral and longitudinal controls.
- Limit open – ended modes.
- All crew inputs should be unambiguous.

2. Guidelines.

- The autopilot should be designed for normal flight.
- Use auto – throttle and pitch control to control longitudinal axis, aileron and rudders to control lateral / directional axes.
- Minimize the number of modes.
- Automatic configuration checks should be built in.
- Leaving nothing to chance.
- Global Positioning System (GPS) and terrain databases can replace traditional navigation aids.
- Point and click programming should be the goal.
- Windows like helping menus.

6.4 CLASSIC AUTOPILOT DESIGN USING PID CONTROLLER

The choice of controller depends on open-loop response of aircraft dynamics. For example a small delta-wing fighter plane employs a very complex controller which is adopting to a wide-range of air-speeds, altitudes, attitudes and maneuvers. Such a controller is associated with a fast responding servo-systems stabilized around several feed-backs loops programmed for gains and time constants to minimize position error, peak overshoot, settling time , gain margin , steady state error, velocity error and time

error, not compromising response and stability. Whereas a slow flying large commercial aircraft employs a less complex controller employing fewer adaptive control loops, all tailored for a particular aircraft model.

In general the servo-system consists of an error detector, controller (PI, P or PID) and necessary sensors like pitch & roll gyros, which generate rates and positions. Any change in the control surface moment is also sensed by separate position sensors for a feed-back in in a local-loop to minimize position and velocity errors and limit deflection rates. For example a large commercial aeroplane employs a sensor feed-back to limit deflections in pitch to less-then twenty-five degrees in pitch up altitude at low-speed configurations and this limit is restricted to ten degrees at maximum operating velocity.

For our purposes we select a second order system with damping ratio ζ_n=0.7 and natural frequency (ω_n)= 40 rad/sec. It is designed using MATLAB / SIMULINK toolbox.

$$\mathbf{Ga} = \frac{\mathbf{1024}}{\mathbf{s^2 + 44.8s + 1024}}.$$

If we choose a PID controller in the design architecture then it is desired of the PID controller to produce a command to the input of the actuator according to the relationship.

$$\mathbf{U(s) = K_p((1 + 1/(\tau_i.S) + \tau_d.S)}$$

6.4.1.Zeigler – Nichols Method

By using Zeigler – Nichols Method, we can calculate K_p, Ti and T_d . If the dynamic model of the plant is known, using the tuning rule, we can determine the critical gain **Ker** for the closed-up system which initiates oscillations. Then the corresponding time period **Ter** of the oscillation is determined. Knowing the two values, the PID controller can be tuned using the following results:

$$\mathbf{K_p = 0\,.\,6\ Ker}$$
$$\boldsymbol{\tau_i = 0\,.\,5\ Ter}$$
$$\boldsymbol{\tau_d = 0\,.\,125\ Ter}$$

By using root-locus or Routh criteria method, the critical gain **Ker** can be obtained. But to get accurate result, fine tuning of controller constants that is $\mathbf{K_p}$**,** $\boldsymbol{\tau_i}$ and $\boldsymbol{\tau_d}$ are required.

6.4.2.Guidelines For Designing A PID Controller

When designing a PID controller for a given system, the following procedure is generally adopted:

1. Obtain Open-loop response and determine the parameters to be improved.
2. Add a proportional control (P-only) to improve rise-time (t_r).
3. Add a derivative control to improve the overshoot.
4. Add an integral control to eliminate the steady-state error.
5. Adjust each of K_p, K_i and K_d until you obtain a desired overall response.

The following table summarizes the effect of K_p, K_i and K_d on controller characteristics.

CL RESPONSE → **PARAMETER ↓**	**RISE TIME** **(t_r)**	**OVERSHOOT** **(ξ)**	**SETTLING TIME** **(t_s)**	**S-S ERROR** **($\in$)**
Kp	DECREASES	INCREASES	SMALL CHANGE	DECREASES
Ki	DECREASES	INCREASES	INCREASES	ELIMINATED
Kd	SMALL CHANGE	DECREASES	DECREASES	SMALL CHANGE

Table 6.1 Controller Characteristics

CHAPTER 7

MATLAB SIMULATION STUDY OF CONVENTIONAL AUTOPILOTS

7.1.INTRODUCTION

Using small perturbations state-space models ,which were described in Chapter-3, the closed-loop response of Auto-pilot PID controllers in Pitch and Roll are studied using MATLAB/ Simulink Control systems tool-box. Unit step input is used as pilot command and the response of the two-autopilot modes (pitch & roll) are investigated in this chapter. The aircraft dynamics are derived for flight condition of altitude 3000m and speed 65 m/s. The aircraft state-space model is given in the appendix. In longitudinal modes the positive step input means negative elevator deflection δe. The response of the two landing phase on glide path and flare path are also investigated. During this glide phase , the aircraft velocity is taken as 50 m/s and during flare mode the velocity of the aircraft is taken as 35 m/s and the flare entry height (h_0) is 15.25 m.

7.2. STUDY OF AUTOPILOT PITCH RESPONSE

Pitch attitude autopilot control system is normally used for establishing a reference pitch attitude in vertical profile and for maintaining a reference altitude. This autopilot is normally used when the aircraft is in wings - level flight. The pitch angle θ is the controlled variable. It requires an increase in the damping of short period oscillations for stability and for this, an inner feedback loop utilizing a rate gyro for pitch is added. The design requirements are

- Overshoot : Less than 10%
- Rise Time : Less than 2 seconds
- Settling time : Less than 10 seconds
- Steady – State error : Less than 2%

The Functional Block diagram , Simulation diagram and model response plots are shown in Figure 7.1. Figure 7.2. and Figure 7.3. respectively.

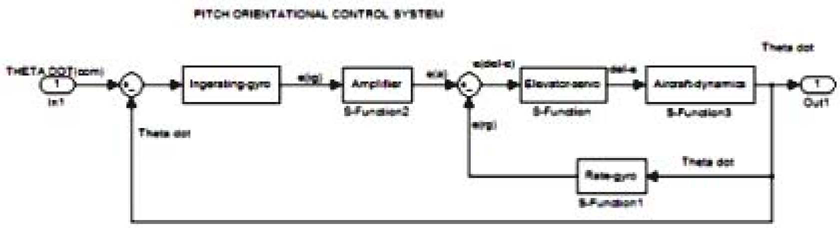


Fig-7.1. Functional block diagram of pitch channel

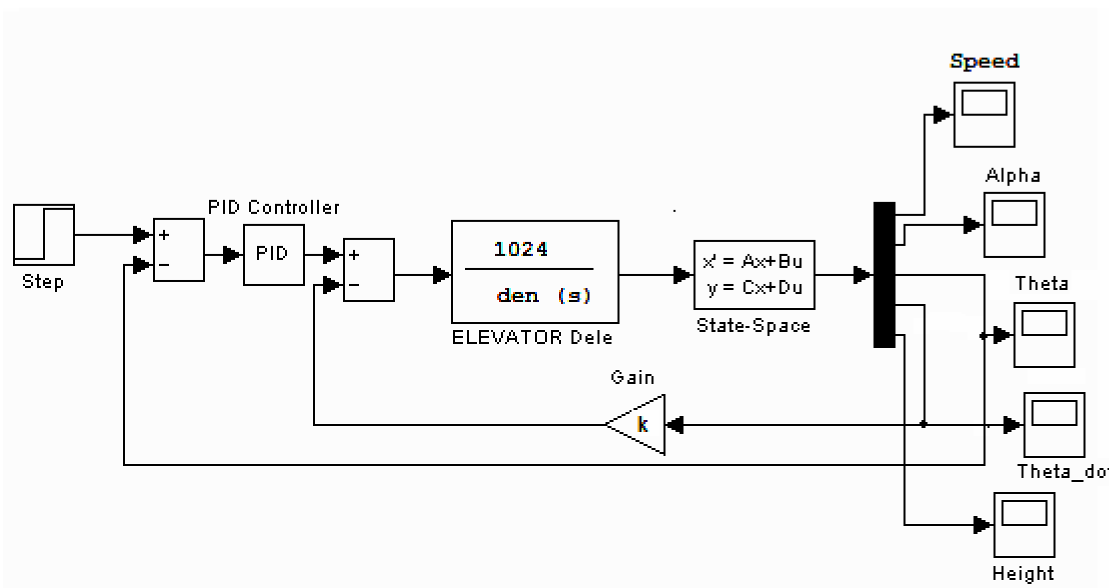


Fig-7.2. MATLAB simulation diagram of Pitch Autopilot

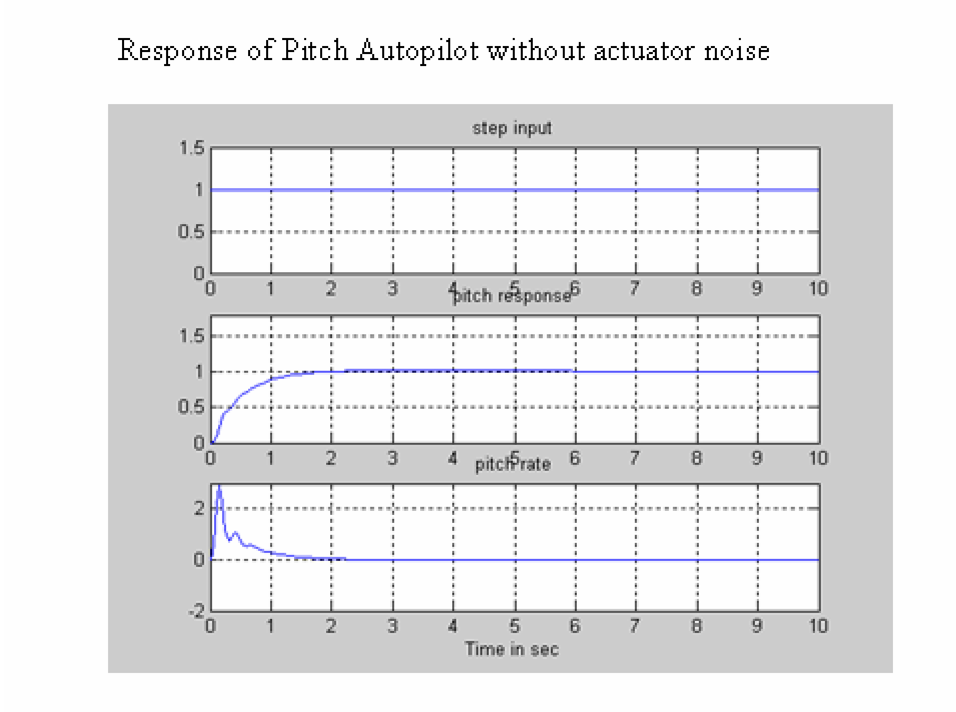


Fig 7.3. Response Plots for Pitch Autopilot

7.3. STUDY OF AUTOPILOT ROLL RESPONSE

The ailerons differential deflection is the input to roll attitude control system. An attitude gyroscope which provides roll information is used as reference sensor in y, z plane. Here also a roll rate inner feedback loop and high sensitivity outer loop are employed for optimum stability maneuverability. The design requirements are

- Overshoot : Less than 5%
- Rise Time : Less than 4 seconds
- Settling time : Less than 8 seconds
- Steady – State error : Less than 2%

One of the most important function of AFCS in roll axis is to attain high degree of spiral damping. The Functional Block diagram, Simulation diagram and model response plots are shown in Figure 7.4, Figure 7.5 and Figure 7.6 respectively.

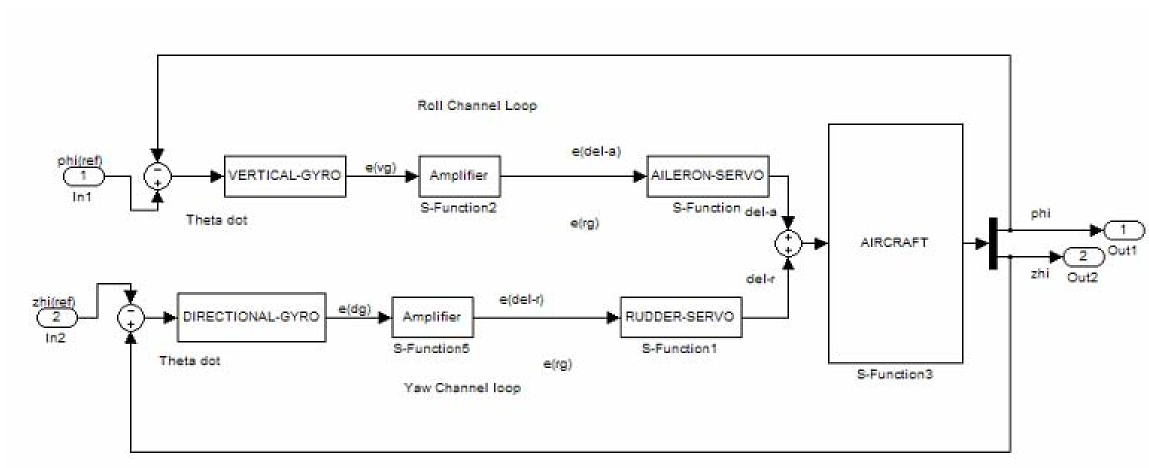


Fig-7.4. Functional Block Diagram Of Roll Autopilot

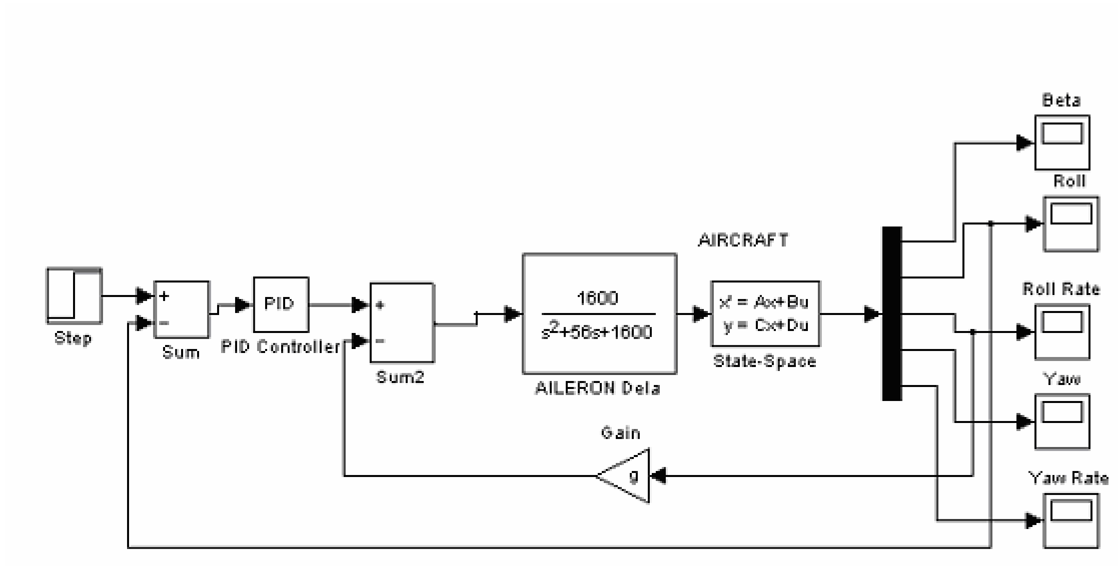


Fig-7.5. MATLAB Simulation diagram of Roll Autopilot

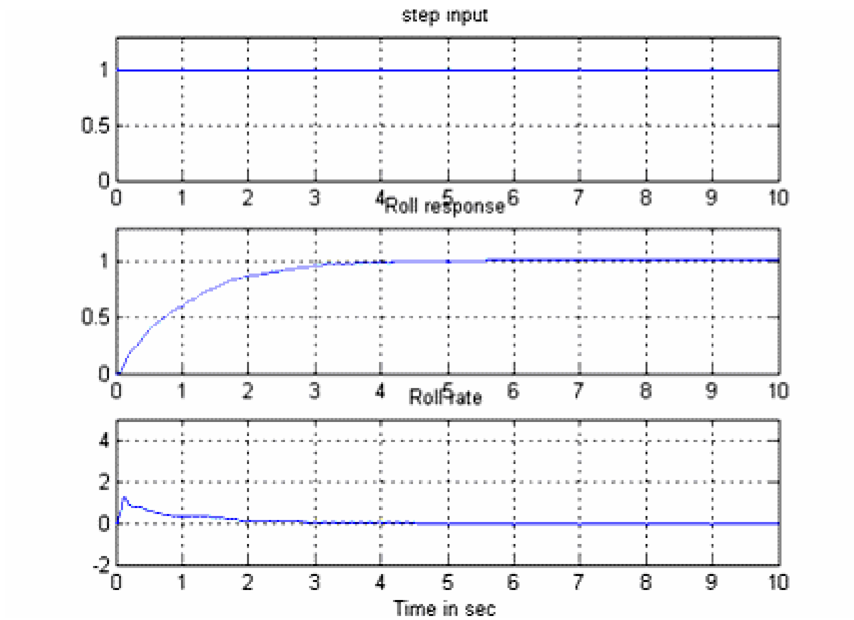


Fig-7.6. Response Plots for Roll Autopilot

7.4. STUDY OF AUTOPILOT RESPONSE IN LANDING CONTROL.

The most important requirement of aircraft is to be controlled for stability during landing phase where weather disturbances prevail to a large degree and the speed of the craft is low due to drag and landing speed limitations. The basic pitch and roll autopilot controllers are employed in outer control loops, with reference signals provided by radio – navigation controllers. The basic landing controls are glide slope localizer and flare control. These are investigated in the following sections.

7.4.1 Design considerations

The following are the design considerations:

1.Landing Strips are usually located at low altitudes where wind element is a strong influencing factor in the model involved. The glide path trajectory is to be controlled at 2.5° under no-wind as well as under wind conditions.

2. Pitch and roll feed-back loops to be engaged in the respective modes for system redundancy.

3.In our simulation models we have omitted Engine throttle control. The models always return linear response for a range of throttle settings. The engine throttles are assumed to be at constant idle setting at the start of approach and speed control loop is kept open.

4. The deviation of UAV in the Y-Plane shall be within $\pm$ 4.5 m and Shall be within

$\pm$ 2 m in Z-axis.

The limits on control variables in our design study are as follows:

1) $-7° \leq \alpha \leq +7°$ → Angle-of –attack limits.

2) $-16° \leq \theta \leq +16°$ → Limits on pitch attitude

3) $-20° \leq (\delta e, \delta a, \delta r) \leq +20°$ → Limits on Actuator deflections.

4) Servo Rate $\leq$ 100° per second → Limits on Actuator speed.

7.4.2. Landing Simulation

The important feature of the landing system is coupling glide-slope and localizer radio signals to autopilot controller using appropriate instrumentation. Now let us study the design features of the landing system. In practice capture of localizer is executed using an on-board navigation computer coupled to auto-pilot. Once captured, the localizer, the airplane is steered to airspace boundary along the localizer to a lower to a lower altitude so that it intercepts the glide-slope signal which is used to provide reference for landing. At the threshold of touching down on a runway. Another control loop called flare-path control executes the landing phase. On touchdown the aircraft pitch is increased gradually according to an exponential function called flare.

7.4.3. Glide Path Landing Simulation.

The geometry of glide slope situation is shown in the figure. 7.7.

The glide path receiver of error between the glide slope reference and the actual aircraft trajectory.

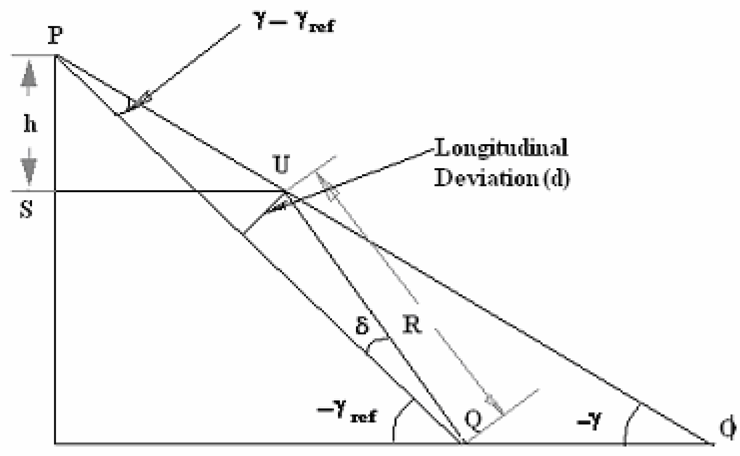


Fig-7.7. Glide Path trajectory.

In terms of Longitudinal displacement, this error or deviation in the longitudinal axis is given by:

$$\mathbf{d = R \sin(\delta)},$$

The controller function is to achieve deviation , d=0

d_dot = U∗ sin (γ-γref)

for small angles : d_dot = U∗(γ-γref)

Integration of the above equation, gives the information about the longitudinal deviations. Once the controller is set on glide-slope mode the closed- loop control components steer the aircraft along the glide-path using error computation and corrections to achieve the desired trajectory.

The glide path control system block diagram, MATLAB simulation diagram and response plots are given in fig 7.8., 7.9. and 7.10 respectively

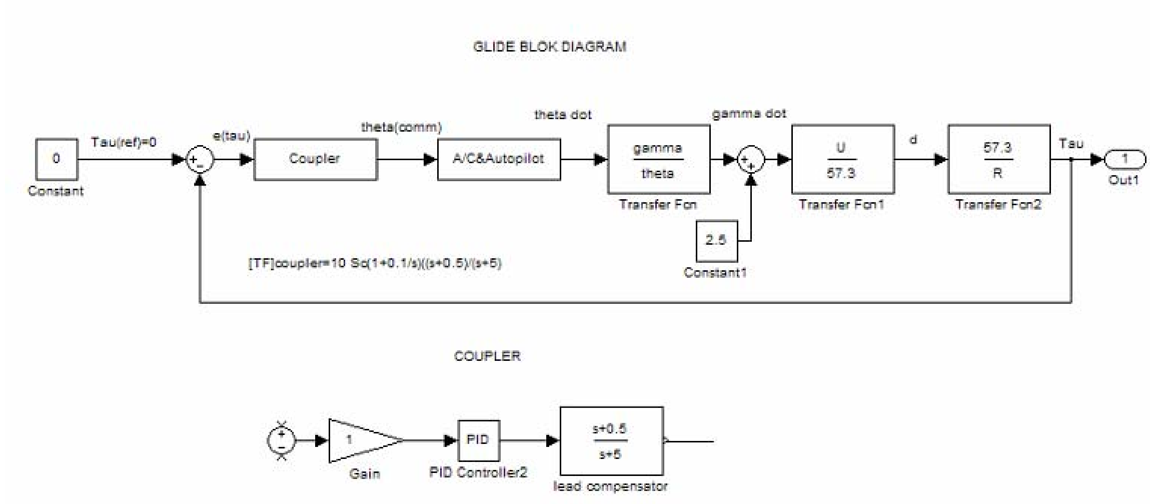


Fig-7.8. Functional lock Diagram of Glide Path Control System

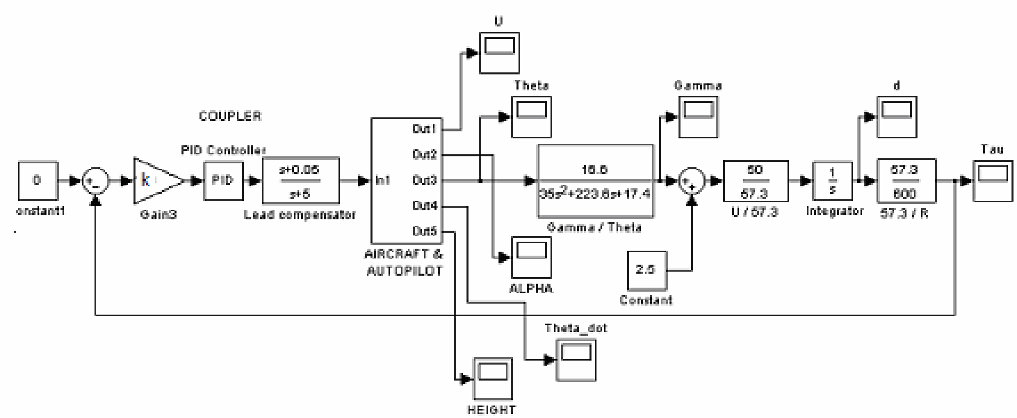


Fig-7.9. MATLAB simulation model for the glide path control .

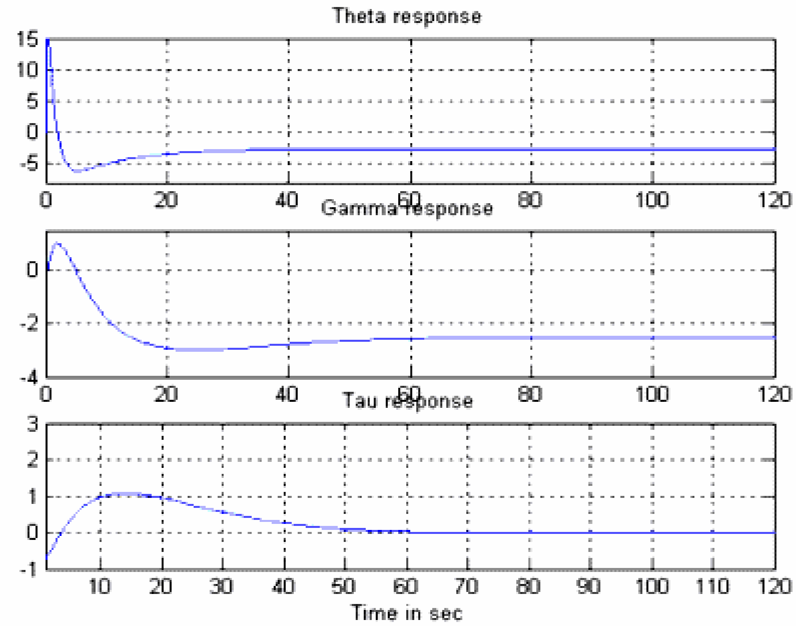


Fig-7.10. Response Plots for Glide path simulation

7.5 . Flare Control Simulation.

The final phase of the landing is the transition from the glide slope to the actual touchdown, generally referred to as the flare. At predetermined height before touch-down, the flare control begins to modify flight path angle from -2.5° to a positive value ,

according to an exponential trajectory given by,

$$h = h0\, e^{-(t/\tau)}$$

The flare path trajectory and the runway approach scheme are shown in Fig- 7.11, 7.12 respectively.

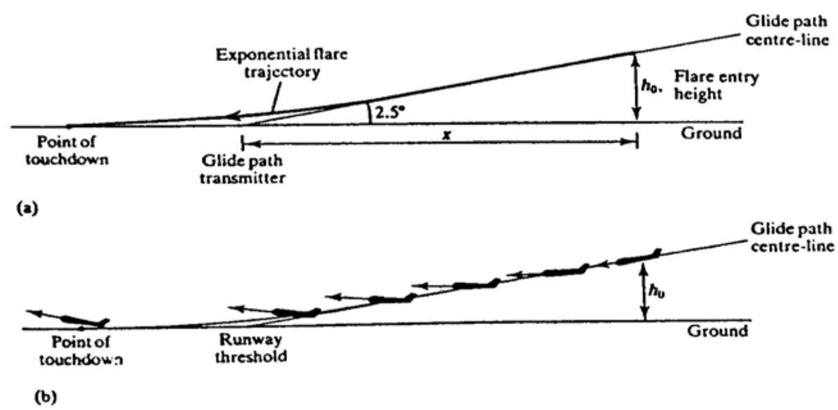


Fig. 7.11 .Flare path trajectory

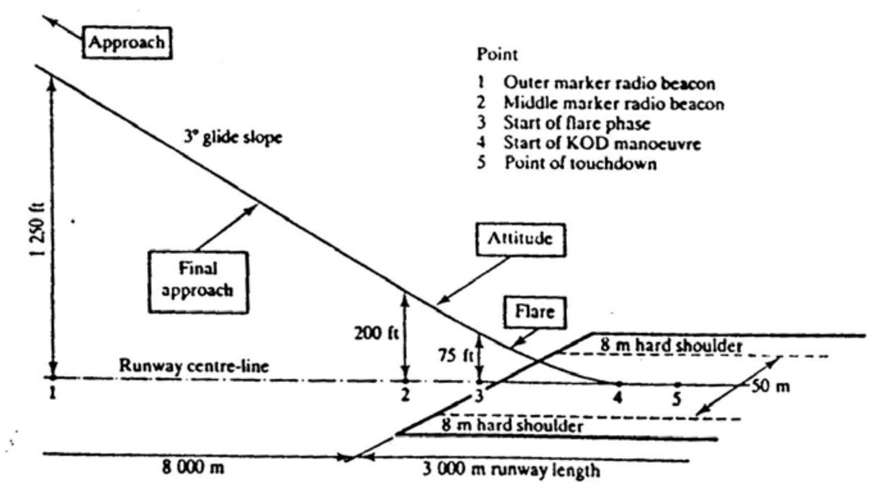


Fig- 7.12. Runway approach path.

The functional schematic, MATLAB simulation diagram and response plots of the flare control system are shown in Fig- 7.13 ,Fig-7.14.and Fig-7.15 respectively. The outer loop supplies the rate-of-descend command hr_dot. The pitch hold autopilot mode is activated at flare entry height h0. The second lead network in the coupler is used to obtain a higher value of coupler sensitivity, thus preventing the aircraft from flying into runway too soon.

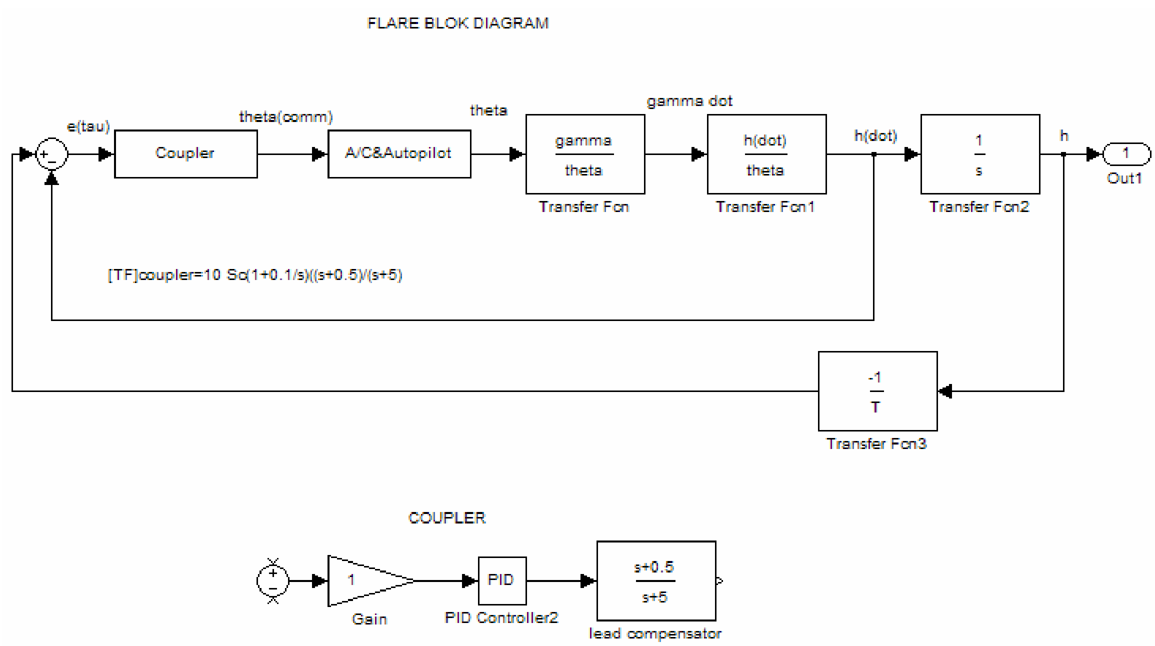


Fig-7.13. Functional Diagram of Flare control system.

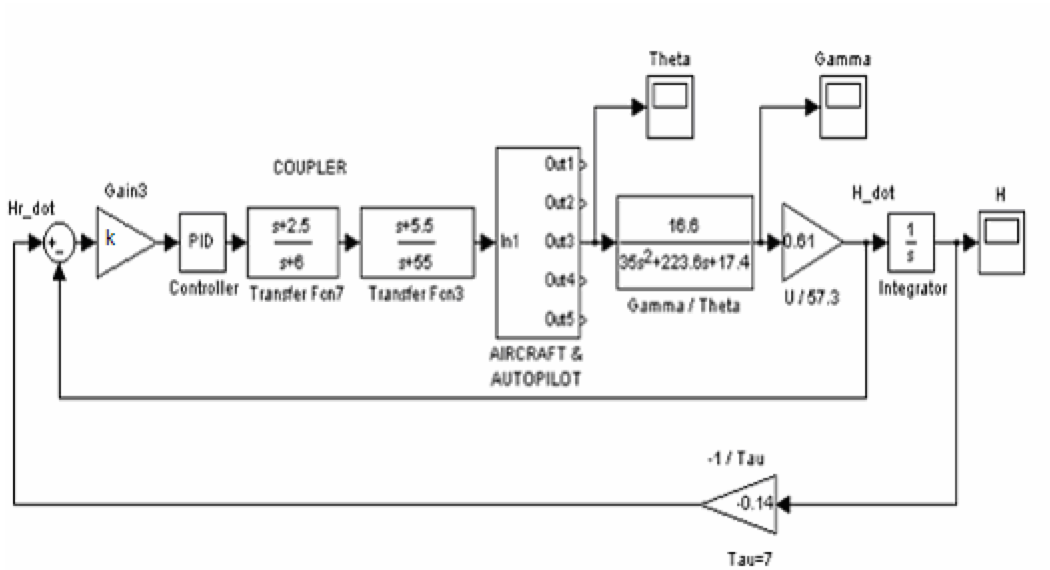


Fig-7.14. MATLAB simulation model for Flare control system.

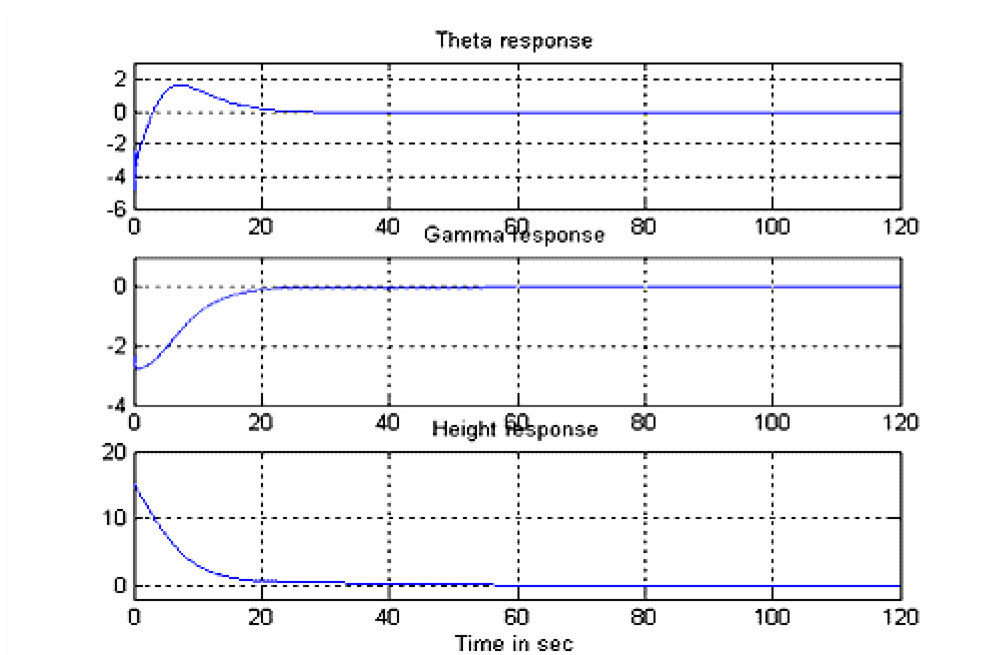


Fig-7.15. Response plots of Flare Control System.

CHAPTER 8
NEURAL NETWORKS

8.1. INTRODUCTION

In the 1950s, requirements for adaptive and optimal control system design were more demanding and a need for 'expert systems' were in demand in engineering, defense and several other areas . In the beginning, technical papers were published on neural network preliminaries and earlier concepts evolved on primitive ideas based on limited understanding of brain neuron cell architecture and their functioning in brain. Based on the investigations neuron models were published in technical publications related to diverse disciplines like medical journals , IEEE transactions on control systems , science journals etc. Due to a very high number of neurons and their complex physical and chemical properties, the complete understanding of brain functions like memory, learning and intelligence are still under exploration.

However, the ideas and discoveries published in the past have contributed to the development of neural network theory and their applications in diverse disciplines like control system, instrumentation, aerospace, artificial intelligence, robotics, medicine, commerce , market analysis and so on. Today , more powerful and still evolving software tools like MATLAB® on neural network principles are available to engineers and scientists .

8.2 BASICS OF ARTIFICIAL NEURAL NETWORKS

Artificial Neural Network is an information-processing system. The elements called *neurons* process the information. The signals are transmitted by means of connection links. The links posses an associated weight, which is multiplied along with the incoming signal (net input) for any typical neural net. The output signal is obtained by applying activations to net input.

A single artificial neuron element is shown below:

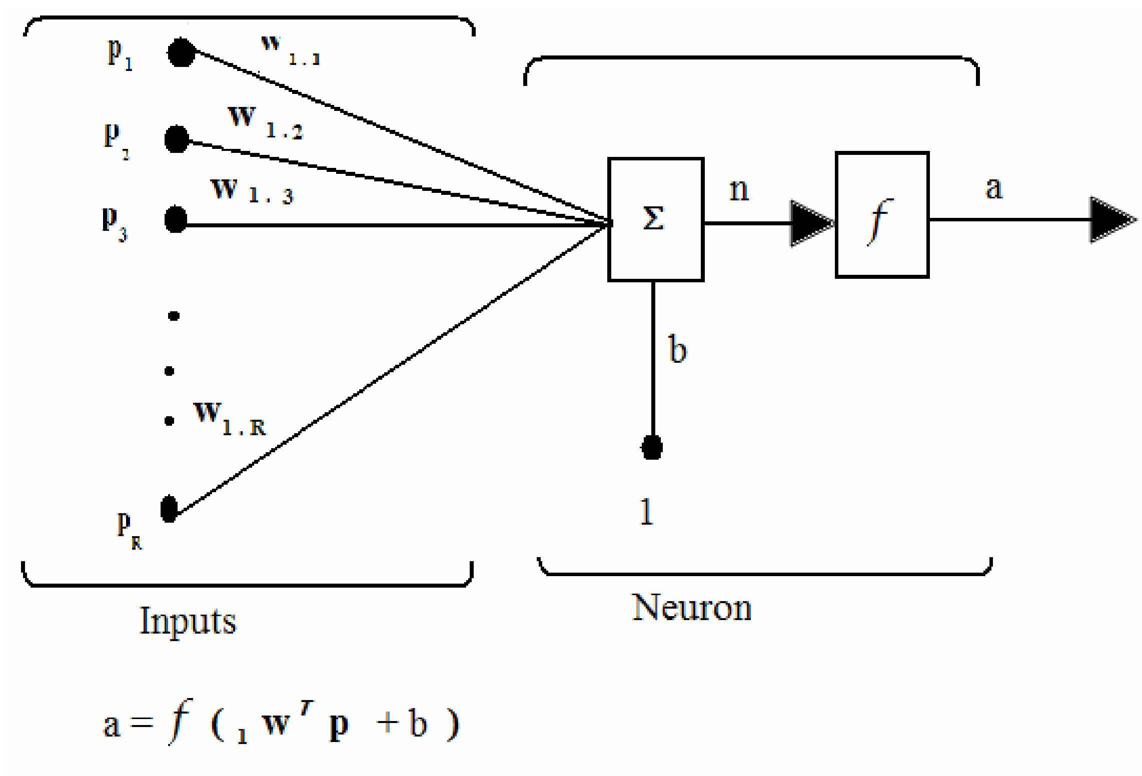


Fig : 8.1. Artificial Neuron

a = output

w = scalar weight

p = scalar input

b = bias

n = Σ(w p + b)

a = f(Σ (w p + b))

f = Transfer Function

The function a = f(Σ (w p + b)) is the basic data process in an artificial neural network. The function 'f' is decided by the application.

For example, a threshold function also called as hardlim is employed in solving digital logic functions and linear algebraic expressions. Control systems involving non-linear and linear differential equations employ neurons with differentiable transfer function like pure-linear, log-sigmoid and tan-sigmoid.

8.2.1 Methodology

Several neuron elements connected in parallel series combination with suitable interconnection is called a neural network. Several neural network architectures and algorithms have been developed independently. These are specialized for different type of applications. For solving control system problems, multilayer neural network with log sigmoid , tan-sigmoid and pure-linear transfer functions and advanced training algorithms are used. Neural Networks are adjusted or trained to get specific target output for particular input. Training algorithms based on supervised learning or incremental training are used. The general multilayer neural network is shown in figure 8.2.

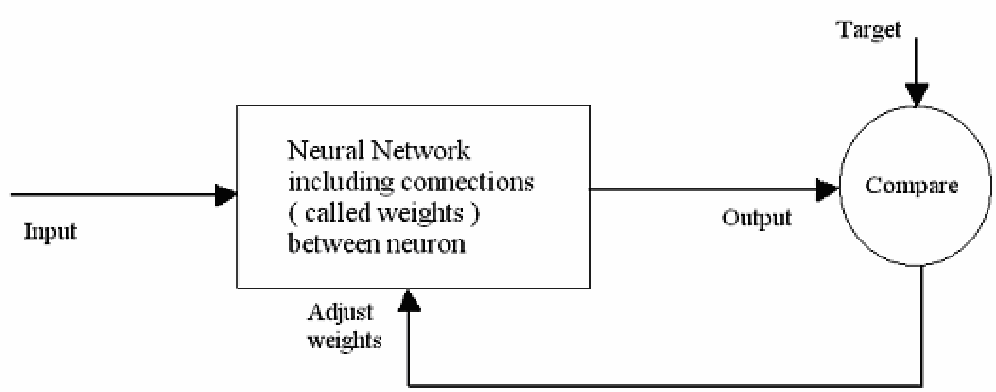


Fig : 8.2. Basic Methodology in Neural Network

8.2.2.Learning and recall

Neural Networks perform two major functions namely *Learning* and ***Recall*.** Learning is the process of adapting the connecting weights in an ANN to produce the desired output vector in response to stimulus vector presented the input vector. Recalling is the process of accepting an input stimulus and producing an output response in accordance with the network weight structure.

The set of well defined rules for the solution of a learning problem is called learning algorithm. Learning encodes patterns information into inter-neuronal connection strengths. Different types of learning are Supervised Learning, Unsupervised Learning, Graded Learning, Hebbian Learning, Associative Learning, Competitive Learning, and Reinforcement Learning.

8.3.ADVANTAGES OF NEURAL NETWORK DESIGN

The main advantage of ANN are

- ANN develops model through learning, unlike the set of rules in a computer program, that process data and produce response.
- ANN is flexible in changing environment. Same model 'learns' and adapts to new requirements in system response. They need only information (or 'learning data') on changes.
- ANN is capable of learning complex dynamics with uncertainty. Statistical learning algorithms are very powerful tools to handle uncertainties.
- ANN provides fault tolerance, since, by design, damage to few interconnection links within the network will not impair the overall performance. The remaining network elements 'learn' to adapt and produce the same result as before.
- ANN performs massive parallel processing in contrast to conventional digital computers in which computations are executed sequentially.

8.4. NEURAL NETWORK CLASSIFICATION

Neural Networks can be classified as either Feed Forward network or Feed Backward (Recurrent) network. Feed Forward network has no feedback between layers. The output of one neuron multiplied by a weight becomes the input of an adjacent neuron of the next layer e.g. Back Propagation Network, Radial basis function. Feed Backward (Recurrent) network is one in which each neuron receives as input a weighted output from all other neurons, including itself as Feedback signal e.g. Simulated annealing, Boltzmann machine, Hopfield net etc. In the design of Neural controller in this project multi layer Feed Forward network is considered.

8.5. MULTILAYER FEED FORWARD NEURAL NETWORK

8.5.1.Architecture

In a Feed Forward Neural Network, neurons in a given layer receive input only from the previous layer. A multilayer Neural Network consist of an input layer, a hidden layer and an output layer with bias units. A multilayer Feed Forward Back Propagation network with one hidden layer is shown in figure. The input to the first layer is the input to the network, called *Input Layer*. The output of the intermediate layer are not observable and hence this layer is called *hidden layer* of the network. The outputs of the third layer are observable , called *output layer*. A typical Multilayer Feed-forward Neural Network is shown in Fig-8.3.

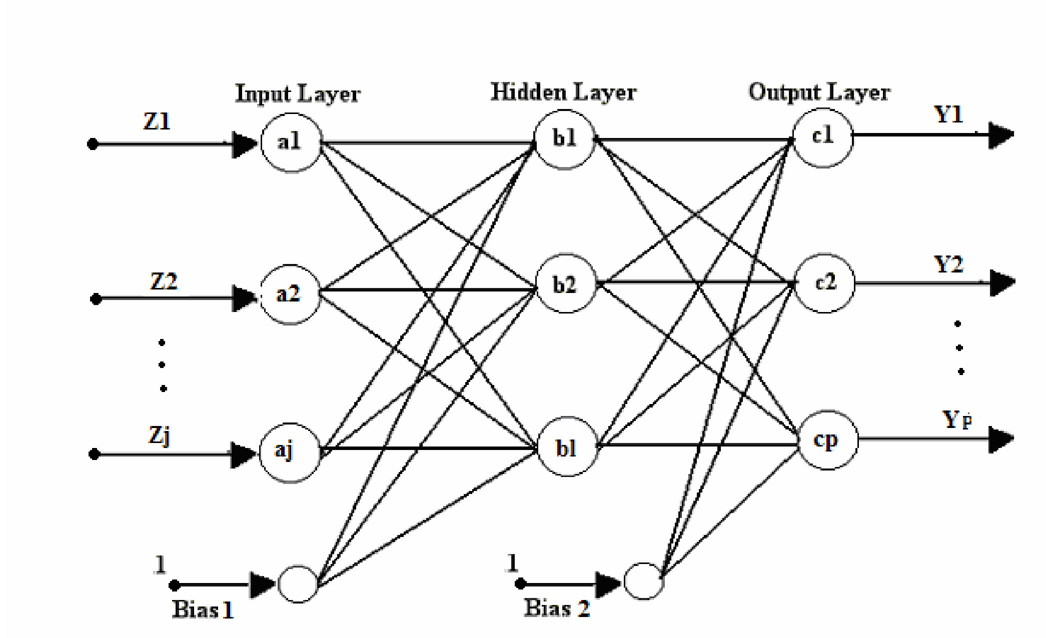


Fig : 8.3. Multilayer Feed-Forward Neural Network.

8.5.2. Feed Forward Neural Network Equations

Input layer:

$$Z_j = f(Z_j),$$
$$= \Sigma \beta_{l1} u_l \qquad l = 1, 2, \ldots n_u$$

Hidden layer:

$$V_1 = f(V_1)$$
$$= \Sigma \alpha_{1j} z_j \qquad j = 1, 2, \ldots n_1$$

Output layer:

$$Y_p = f(Y_p), \qquad p = 1,2, \ldots n_y$$
$$= \Sigma \gamma_{p1} V_1 \qquad I = 1,2, \ldots n_2$$

Where n_u is the number of inputs, n_1 is the number of neurons in the hidden layer, n_2 is the number of neurons in the output layer, n_y is the number of outputs and β_{11}, α_{1j}, γ_{p1} are interconnected weights. Linear activation function is used for input layer, tangent sigmoid function is used for hidden and output layer.

8.6. BACK PROPAGATION NETWORK

Back propagation network is a multi layer feed forward network. It uses extended gradient-descent based delta learning rule, called Back propagation (of errors) rule. Back propagation adjusts the weights and biases of the network with differentiable activation function units, to learn a training set of input-output. Gradient-descent used to reduce or minimize the sum squared error of the NN at the output layer. The network is trained by supervised learning method. Back propagation network is train the net to achieve a balance between the ability to respond correctly to the input patterns that are used for training and the ability to provide good response to the input that are similar.

8.6.1. Back propagation rule

The total squared error of the output computed by net is minimized by a gradient-descent method known as Back propagation rule or generalized delta rule.

8.6.2. Back propagation algorithm

1. Initialize the weights to small random values.
2. Choose the input pattern from the data generated.(feed forward)
3. Propagate the signal forward through the network.
4. Calculate output of the network.
5. Compute the δ_i^L in the output layer.($o_i=y_i^L$)

$$\delta_i^L = f'(S_i^L)\,[d_i^L - y_i^L)$$

Where S_i^L represents the net input to the i-th unit in the L-th layer is the derivative of the activation function f.

6. Compute the deltas for the preceding layer by propagating the errors.

$$\delta_i^k = f'(s_i^k)\ \Sigma w_{ij}^{k+1}\ \delta_i^{k+1},\ \text{for } k = (L-1),\ldots,1.$$

7.Update weights using

$$\Delta w_{ji}^k = \mu\ \delta_i^k\ y_j^{k-1}$$

$$Wi,new = wi,old + \Delta wi$$

$$\Delta wi = \eta\ \delta\ out\ I + \alpha\ \Delta wi\ (old)$$

$$\delta = (d(X)-y(X))\ out\ i(1- out\ i)$$

For hidden layer,

$$\delta = outi(1 - outi)$$

where η is the learning rate and α is a designer parameter which affects the speed of convergence.

8. Go to step 2 and repeat for the next pattern until the error in the output layer is below a pre-specified threshold or a maximum number of iterations is reached.

8.6.3. Selection of parameters

Initial weights *:*

Initial weights and biases are set to random values between -0.5 to 0.5 or -1 to 1.

Selection of learning rate:

1. Start with higher learning rate and steadily decrease it.
2. Increase the learning rate to improve performance and decrease to worsen the performance.

3. Double the learning rate until the error value worsens.

Types of learning:

1. Sequential learning or pre-pattern method.
2. Batch learning or pre-epoch method.

Selection of the hidden layer:

Generally two hidden layers are used. Neurons in the first hidden layer learn the local features that characterize specific regions of the input space. Global features are extracted in the second hidden layer.

Selection of neurons in the hidden layer

1.Pruning method

Process of removing the redundant hidden unit's produces the smallest neural network capable of performing a desired task is called *pruning.*

The neurons to be used in hidden layers are estimated by trail and error approach. One method is to construct with excessive number of hidden units and then some redundant units are removed during learning process. To find which redundant units to be removed the output of all hidden units is monitored and analyzed across all the training examples after the network achieves convergence. If the output of a certain units are approximately constant for all training examples this unit can be removed since it does not contribute essentially to the solution because it is acting as an additional bias to all neurons. After removing some hidden units which do not contribute to the solution the weights of the reduced network must be trained once again to ensure the desired performance.

2. Cross validation method

Divide the data set into a training set T –training and a test set T-test. Subdivide T–training into two subsets: one to train the network T-learning and one to validate the network T-validation. Train different network architectures on T-training and evaluate their performance on T-validation. Select the best network. Finally retrain the network architecture on T –training .Test the generalization ability using T-test.

8.7. NEURAL NETWORK TRAINING

Training process involves four steps:

1. Assemble the training data .
2. Create the network object.
3. Train the network.
4. Simulate the network response to new inputs.

8.8 . NEURAL NETWORK CONTROL STRUCTURE

Model reference control is used for the design .The desired performance of the closed-loop system is specified through a stable reference model. The control system attempts to make the plant output $y_p(k)$ match the reference model output asymptotically, i.e.

$$\text{Lim} \| y_r(k) - y_p(k) \| <= \varepsilon ,$$

For some specified constant $\varepsilon >= 0$.

The MRC training structure is shown in fig.8.4.

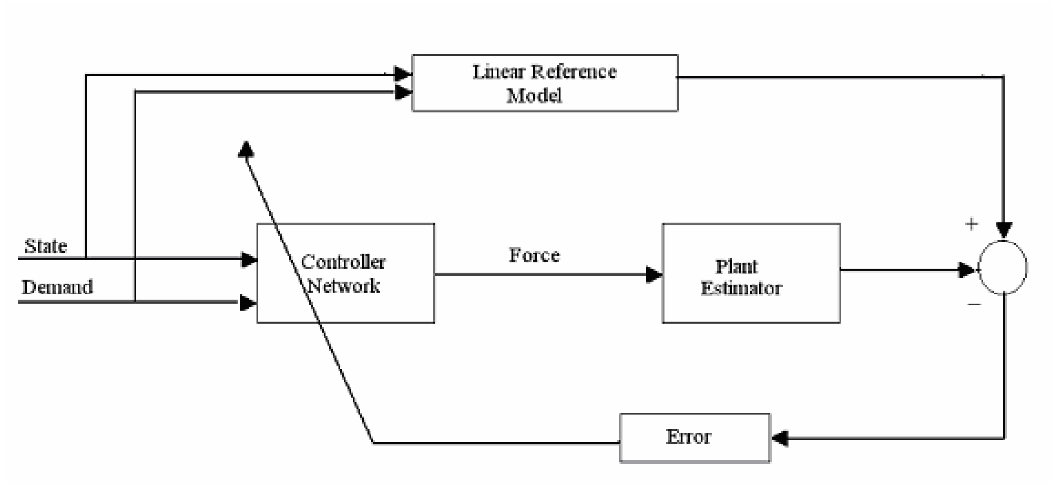


Fig : 8.4. Model Reference control.

The plant is required to respond with target states. The plant estimator is replaced with its Neural model (called neural estimator),for the purpose of training the controller. So before training the Neural net controller, a separate Neural net is trained to behave like a plant. Here specifically , the Neural net is trained to emulate the aircraft open loop response. Training the estimator is similar to plant identification in control theory. The error occurs at the output of the model network. The derivatives of the error can be back propagated through the controller and used to adjust its weights and biases. Thus the control network must learn how to control the plant so that it behaves like the linear reference model.

8.8.1.Training the Neural Network Controller

The given estimator now closely matches the plant dynamics, use it for the purpose of training the controller. The controller learns to derive the plant estimator from an initial state to the desired state. Learning takes place during many trails or runs each starting from an initial state terminating at a final state. The objective of the learning process is to find a set of controller weights that minimizes the error function J , where J is the averaged over the set of initial states.

$$J = E(\| y_r(k) - y_p(k) \|)$$

Then the weights in the controller network have to be modified so that the square error will be less at the end of next run. To train the controller , we need to know the error in the controller output for each of the next run. To train the controller , output for each time step k .Only the error in the final plant state , i.e.$(y_r(k) - y_p(k))$ is available. since the plant estimator is a Neural network, we can back propagate the final plant error through the plant estimator to get an equivalent error for the controller in the k-th state.

8.8.2 Training Data Generation

The data that is generated (to be used as training signal)to train the network should contain all the relevant information about the dynamics of the control system. The required training data generations are given in Appendix.

CHAPTER 9
SIMULATION STUDY RESULTS OF NEURAL NETWORKS

9.1. INTRODUCTION

As outlined in the methodology, initially a choice of multilayer Neural Network is made. Since higher order system are involved in pitch and roll, differentiable transfer functions are required in all the network layers. Hence tansig , logsig and purelin transfer functions are used in the net layers. To produce training data the following method is adopted. Using state-space models in pitch and roll and transfer functions of actuator , model response data are generated for unit step input. Target data (θ_t) are generated using AAA standard models in pitch and roll respectively with same unit step signals. These data are used to train respective Neural Network controllers in pitch and roll. Using back propagation algorithm the Neural network controllers are trained iteratively for different learning rates, until the mean square error between actual controller estimate and target fall at or below a specified error figure.

After training, Neural Network Controller response for pitch and roll channels are Plotted. The trained controller sub-system is then built into glide-path and flare-path control systems and the responses are also plotted. Finally, a comparative analysis of Conventional and Neural Network Controller responses are done.

9.2. NEURAL NETWORKS DESIGN

9.2.1. Neural controller Design

The general structure of the estimator for ***Pitch autopilot***

NN input : u , α , θ ,q ,h.

NN output : θ

Number of hidden layer neurons : 12
Neural architecture : 9, 12, 1.
Training data length : 201

The general structure of the estimator for ***Roll autopilot***

NN input : β , Φ , p , Ψ , r .
NN output : Φ
Number of hidden layer neurons : 12
Neural architecture : 9, 12, 1.
Training data length : 201

9.2.2. Design of Pitch Attitude Control System

Based on the MIL-STD-8785C specifications , the following reference model is selected for training the neural network

$$\theta_{ref}(s) \;/\; \delta\, e(s) \quad = \quad 5.6 \,/\, (s\text{^}2 + 3.5\, s + 5.6)$$

Where s represents a laplace operator for a sampling time of 50 msec.

9.2.3. Neural controller architecture

NN input : δe , u , α , θ ,q ,h.
NN output : θ ref
Number of hidden layer neurons : 12
Neural architecture : 9, 12, 1
Training data length : 201

9.2.4. Design of Roll Control System

Based on the MIL-STD-1797A specifications , the following reference model is selected for training the neural network

$$\Phi_{ref}(s) \;/\; \delta a(s) \quad = \quad 16 \;/\; (s^2 + 5.6\, s + 16)$$

Where s represents a laplace operator for a sampling time of 50 m.sec.

9.2.5. Neural controller architecture

NN input	: δa , β , Φ , p ,Ψ , r .
NN output	: Φ ref
Number of hidden layer neurons	: 12
Neural architecture	: 9, 12, 1
Training data length	: 201

9.3. NEURAL NETWORK SIMULATION RESULTS.

Using the above specifications, Neural Network Controller designs for pitch and roll channels were created and their response in glide slope and flare path control systems were simulated. The response plots are shown in Figures 9.1, 9.2, 9.3 and 9.4. In order to find out the initial weights and biases, the neural network is initialized with randomly chosen weights and biases and trained. This is done with the controller not connected to the system (Off-line). Now the network is initialized with new weights and biases , and the network learns fast comparatively when disturbances are introduced. So this network is capable of changing its weights and biases if there is any disturbance or flight conditions are changed. For the pitch hold systems, the network was trained for a large number

of epochs and the performance index (mean square error) is reached. For roll control systems the
network was also trained for a large number of epochs and the mean square error is reached.

For landing system, the neural network is trained with two hidden layers and each layer is having 5 neurons. With suitable learning rate and momentum constant the network was trained for several epochs till performance index (mean square error) is reached.

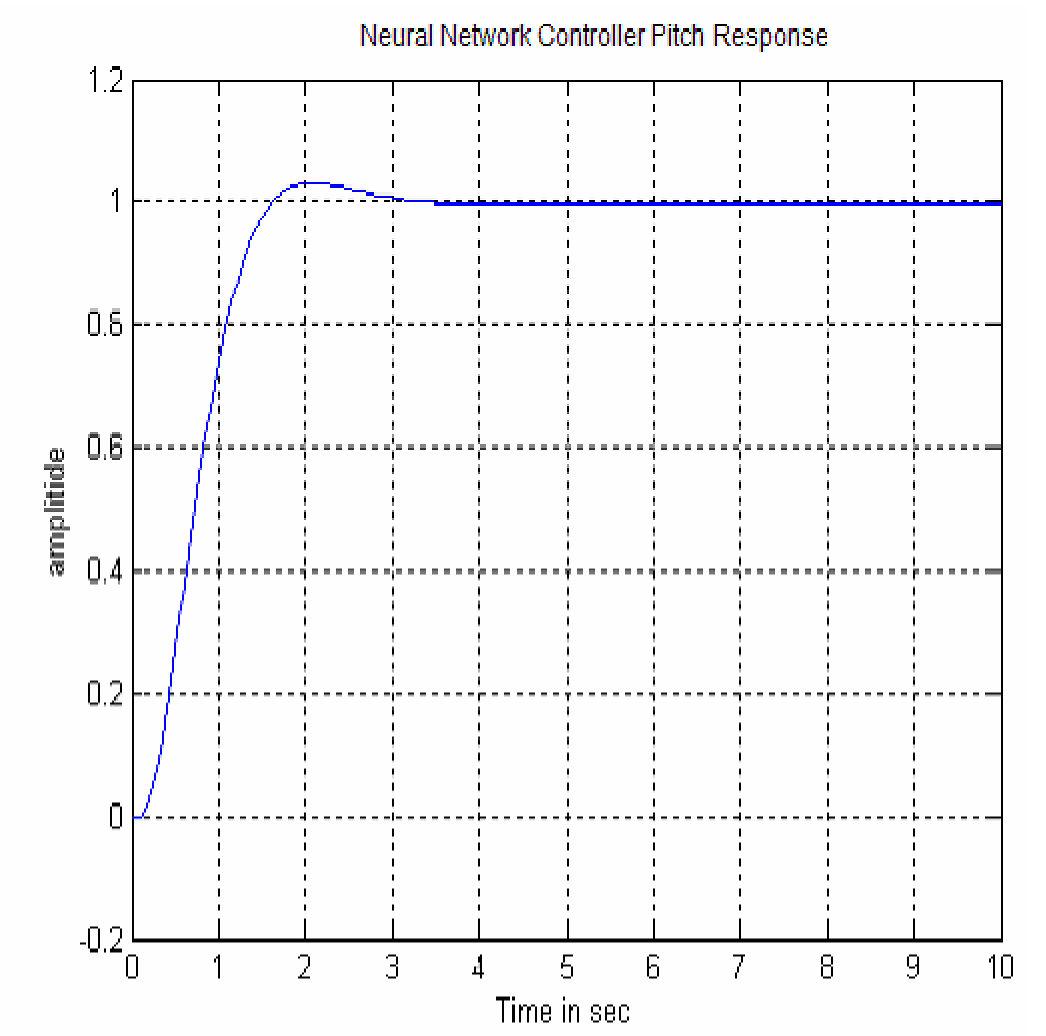


Figure 9.1. Neural Autopilot Response of pitch control.

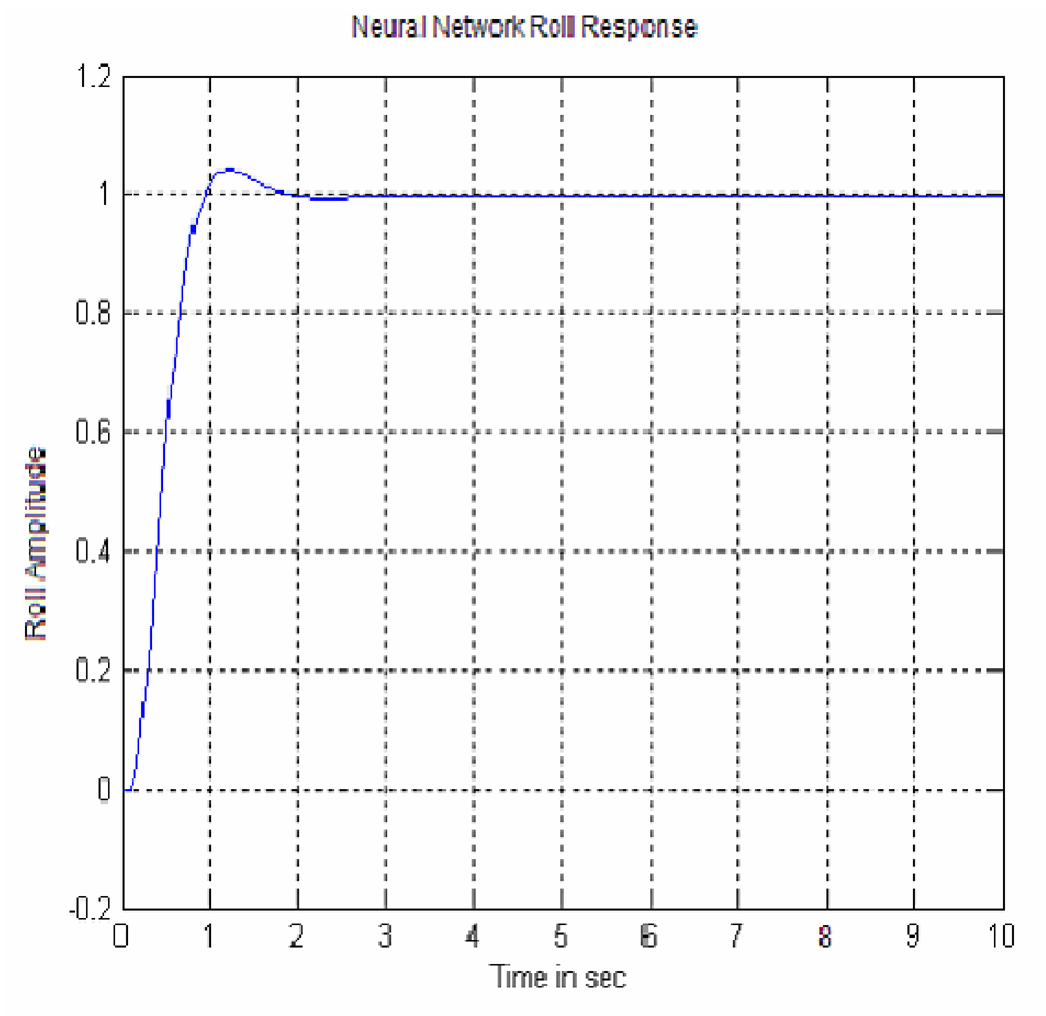


Figure 9.2. Neural Autopilot Response of Roll Control

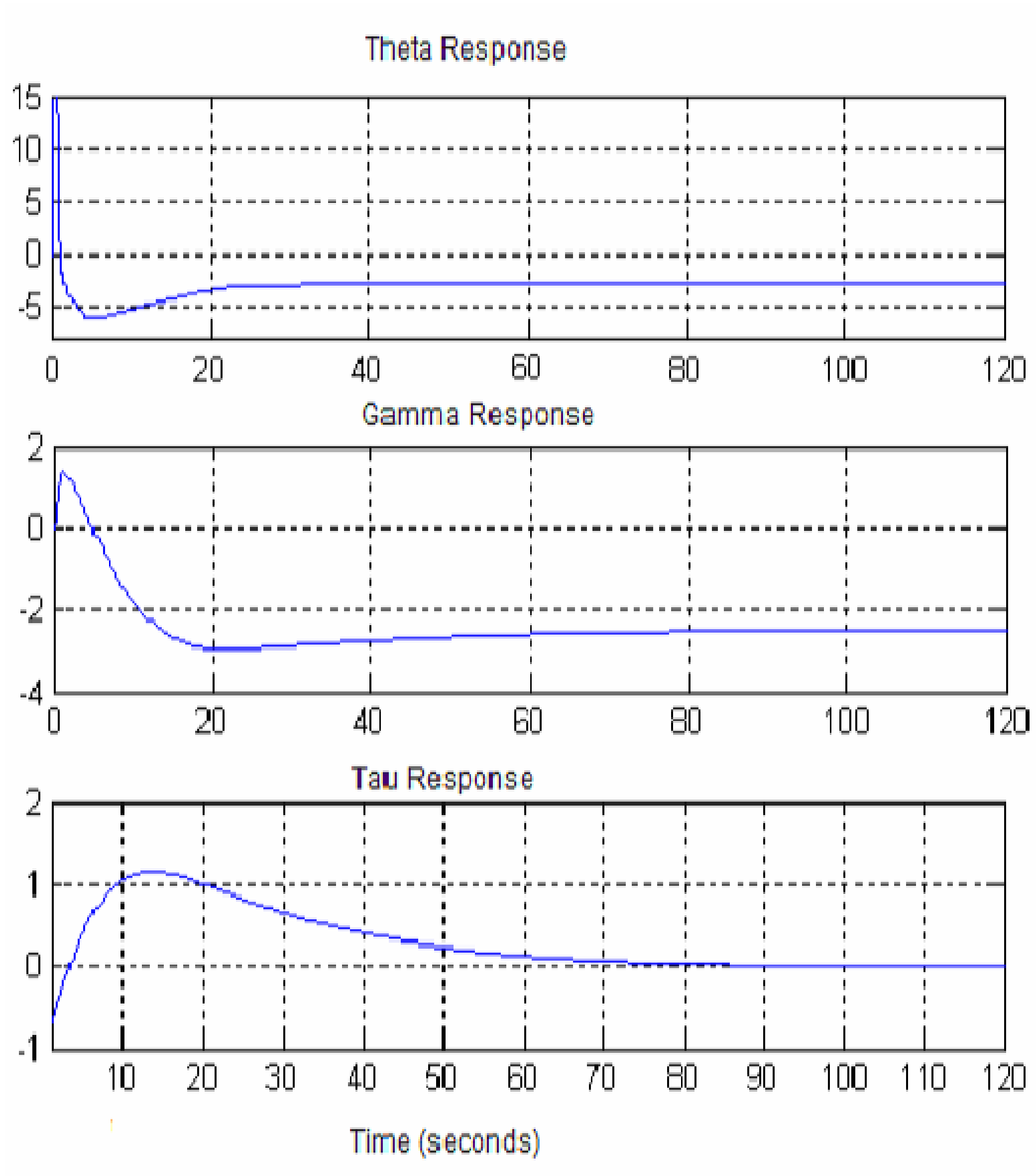


Figure 9.3. Neural Autopilot Response of Glide path Control.

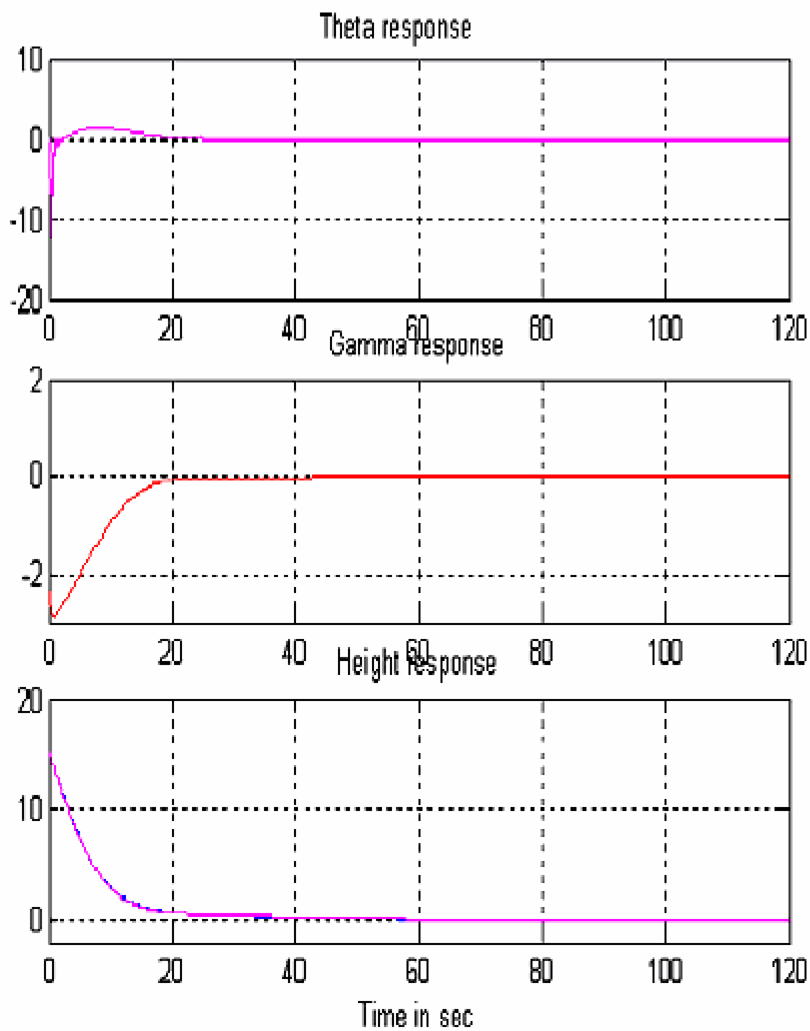


Figure 9.4. Neural Autopilot Response of Flare path control.

9.4. SIMULATION FOR COMPARATIVE STUDY ANALYSIS

The performance comparison of Neural and Conventional Autopilots are studied using parallel simulation and the response figures for Pitch, Roll and Landing control are given the following figures.

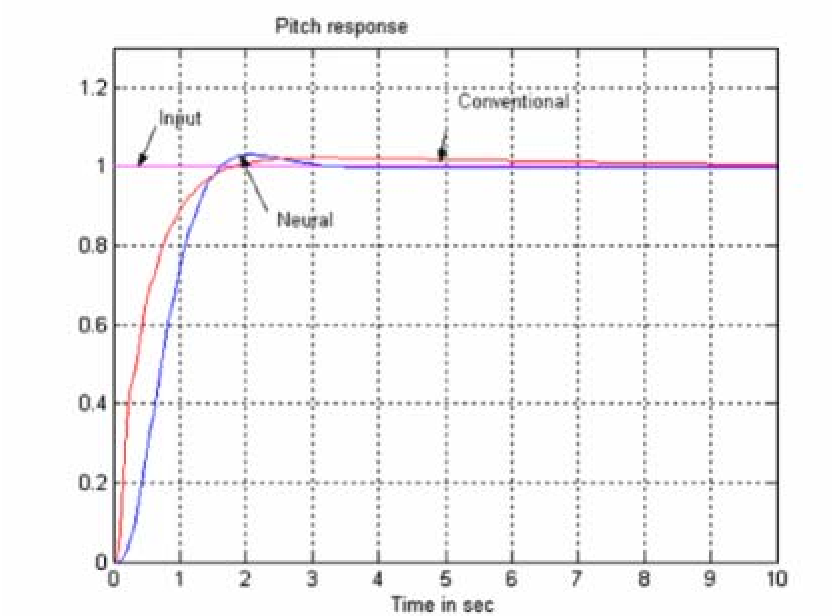


Fig 9.5 Comparison of Pitch Response

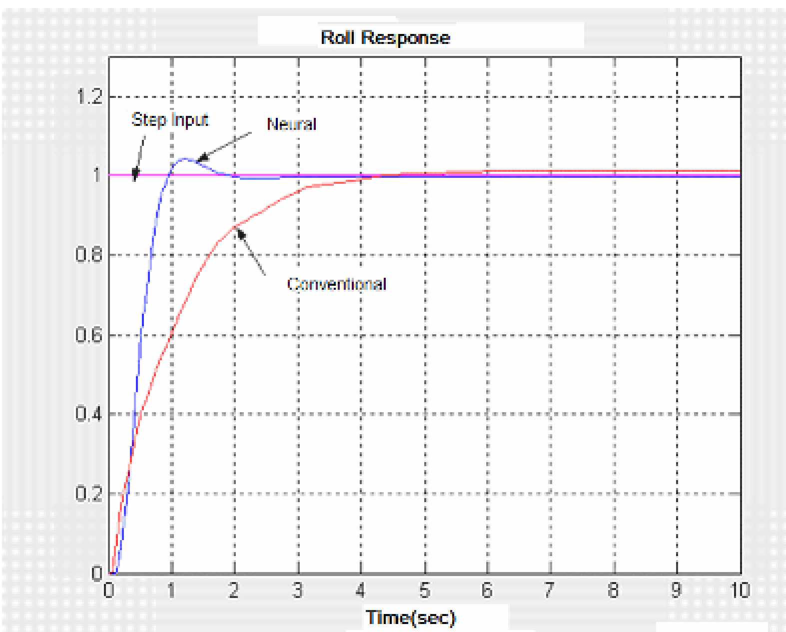


Fig 9.6 Comparison of Roll Response

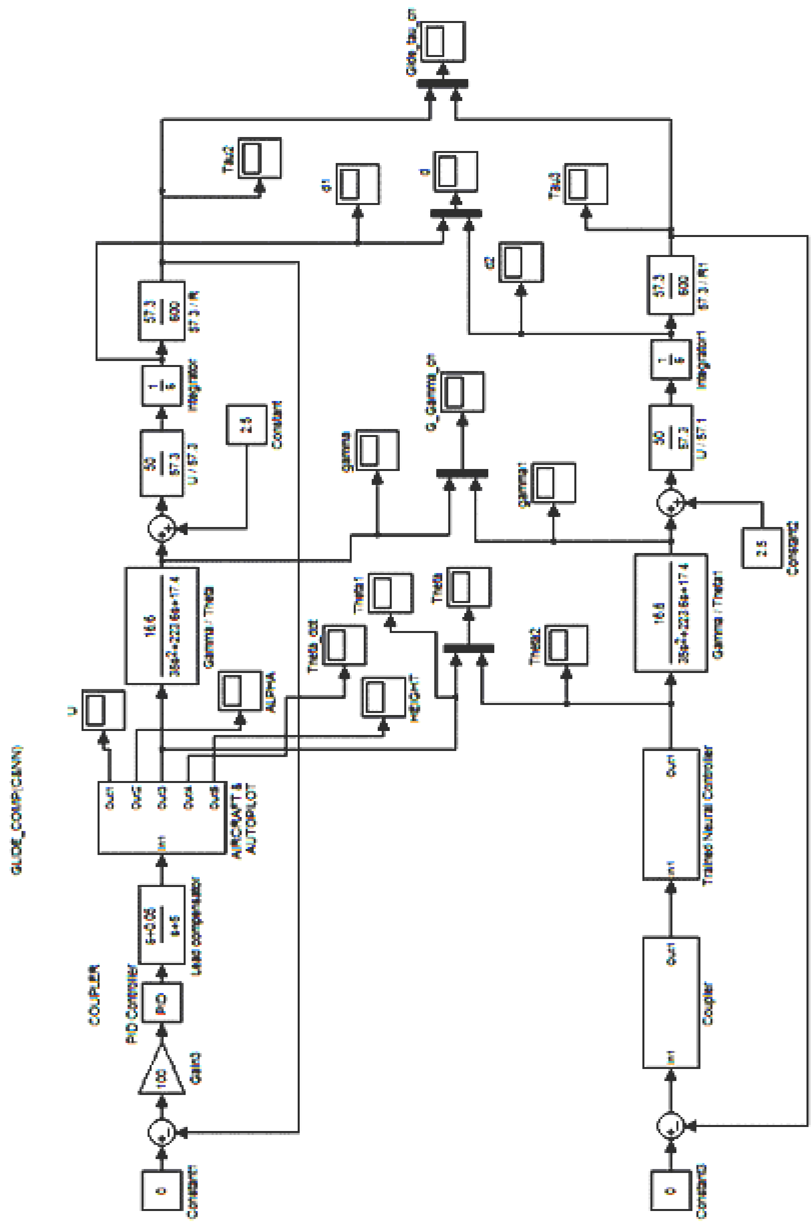


Fig 9.7 MATLAB Simulation Diagram for comparison of Glide path

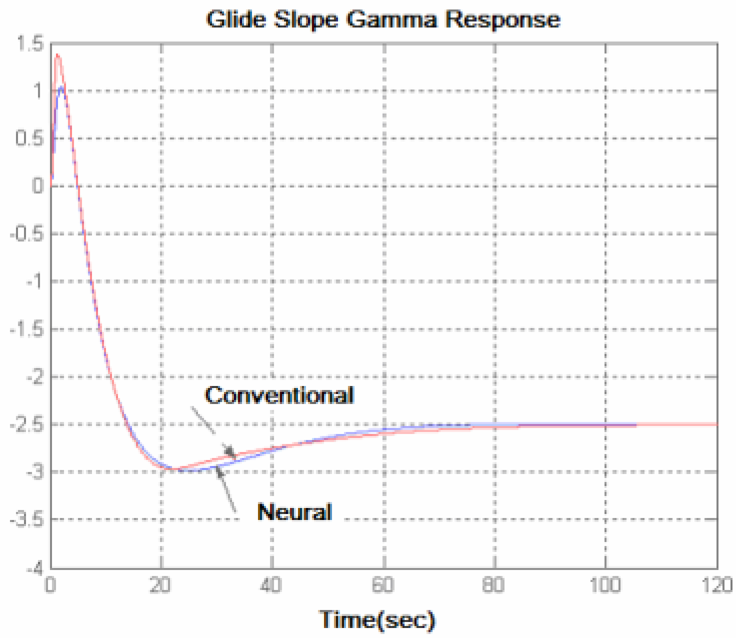


Fig 9.8 Comparison of Glide slope Gamma Response

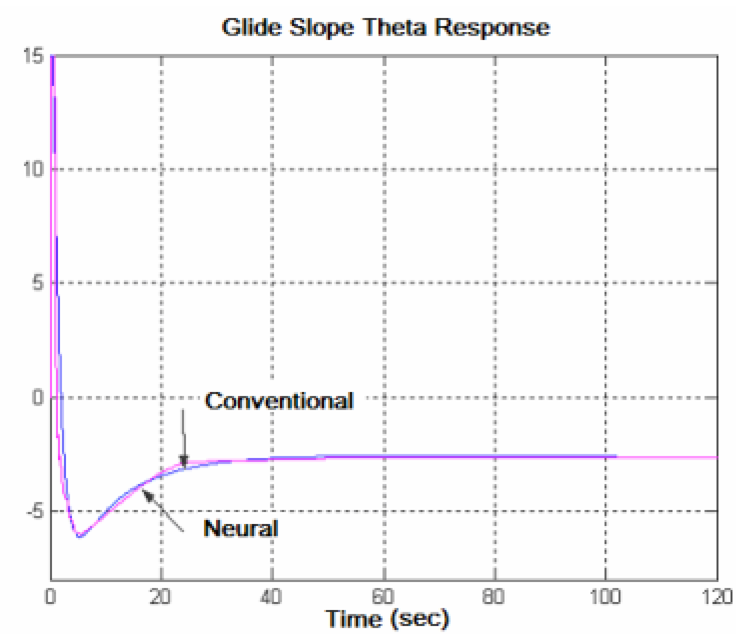


Fig 9.9 Comparison of Glide slope Theta Response

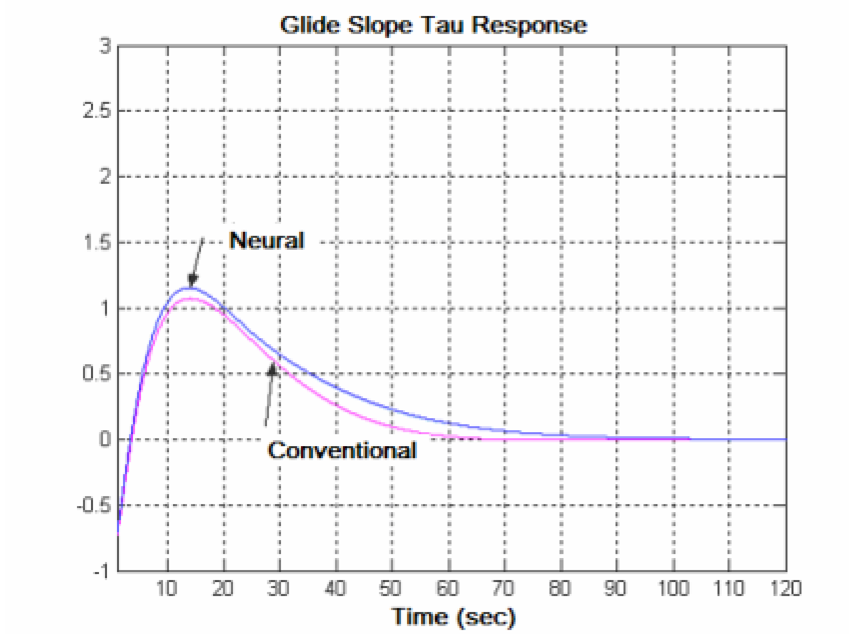


Fig 9.10 Comparison of Glide slope Tau Response

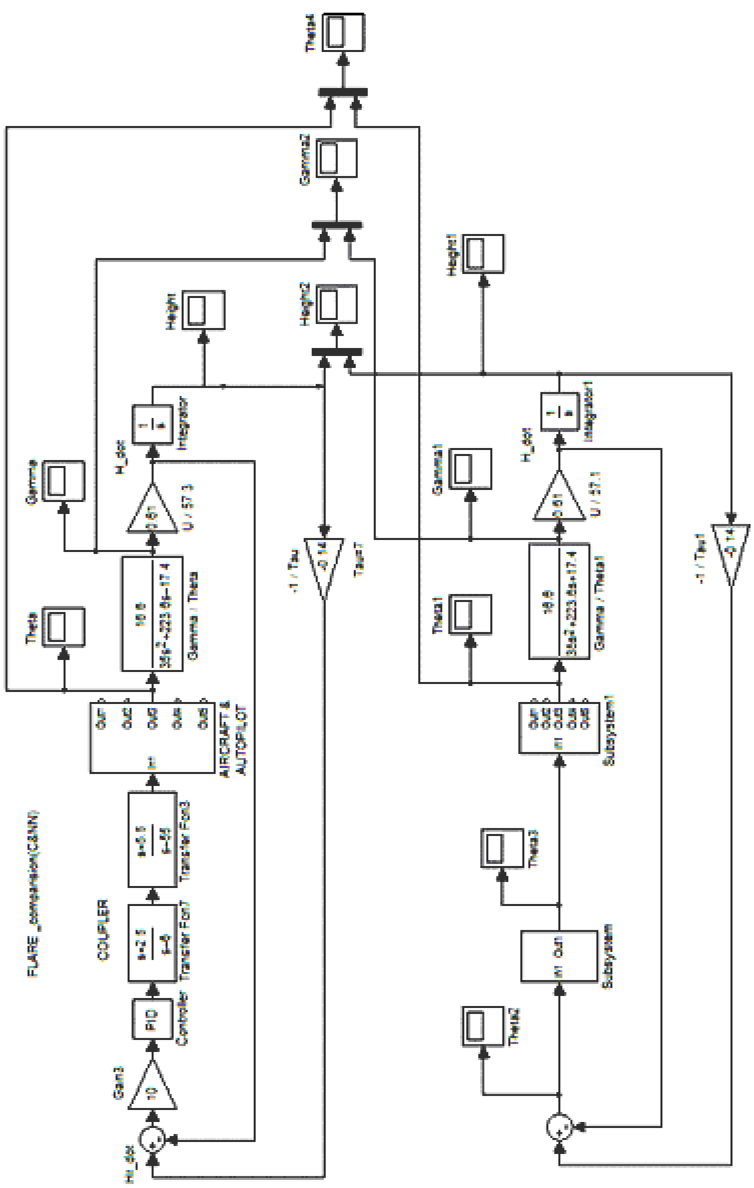


Fig 9.11 MATLAB Simulation Diagram for comparison of Flare Path

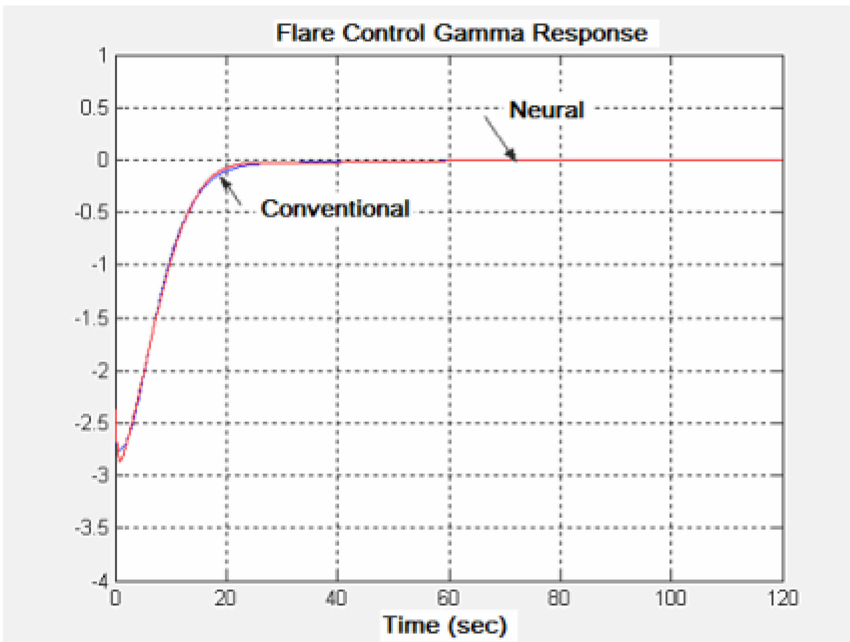


Fig 9.12 Comparison of Flare Path Gamma Response

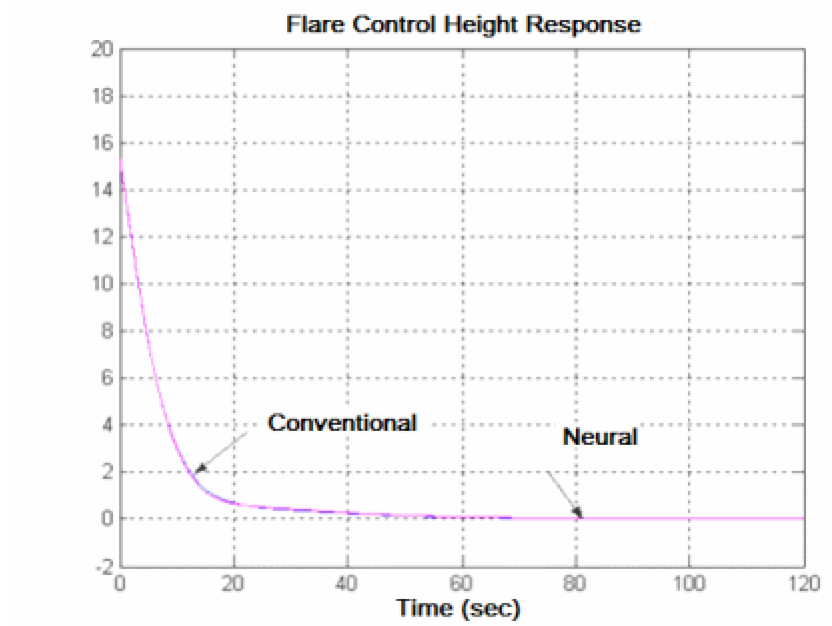


Fig 9.13 Comparison of Flare path Height Response

CONCLUSION

The work demonstrates the feasibility of developing a neural network application in place of conventional controller design. Instead of using conventional control system methods to design autopilot controllers, neural network controllers have been designed and trained with quantitative experimental data and actual model open-loop responses. This largely takes care of most of non-linearities in aircraft characteristics and it is sufficient to define only a target response characteristic. Due to parallel architecture of Neural Design, the controller response is faster. The model response behavior is found to be accurate enough for practical design applications. This is evident from the Comparative simulation study response plots.

FUTURE SCOPE

A vast resource of aircraft design data is available from the past experience by leading aerospace organizations. Advanced navigational aids like GPS and solid state inertial reference systems are available to the designer. The size, weight and power consumption of aircraft hardware has been reduced to a large extent. Combined with other control designs using fuzzy logic and genetic algorithm, more elaborate control laws can be incorporated in the autopilot design to provide more precise and intelligent control response for difficult weather conditions and non-linearities in aircraft characteristics. These developments make future scope for designing fully autonomous autopilots with more robustness, higher degree of controllability and better failure tolerance.

APPENDIX

A.I. List of Matlab Programs for Simulation Studies

1.Programs for Training Data generation

A.1.Reference Model Pitch transfer function.

```
%tf for training ref milstd theta to del-e

num=[5.6 0];

den=[1 3.5 5.6];

sysc1_r=tf(num,den);-<--------- Laplace Transfer function.

sysd1_r=c2d(sysc1_r,.05);-<---- Discrete form.
```

Laplace Transfer function:

$$\frac{5.6\ s}{s^2 + 3.5\ s + 5.6}$$

Discrete Transfer function:

$$\frac{0.2563\ z - 0.2563}{z^2 - 1.827\ z + 0.8395}$$

Sampling time: 0.05

A.2.Elevator actuator Transfer function

```
%tf for training ref milstd theta to del-e

%tf for aileron actuator

num=[20];

den=[1 20];
```

```
sysc_a=tf(num,den);

sysd_a=c2d(sysc_a,.05);


Laplace Transfer function:
      20
   ------------
    s + 20

Discrete Transfer function:

   0.6321
----------------
 z - 0.3679

Sampling time: 0.05
```

A.3. Aircraft State-Space Model and Transfer Functions

A.3.1 Longitudinal mode

```
A =  [-0.077 5.6691 -9.81 0 0;-0.0022 -5.8734 0 0.930  0;
        0 0 0 1 0;0.0004 -41.8834  0 -2.8536 0;0 -1.6579 1.6579 0 0]
A =  [u  α  θ  q  h]

B  =  [0;-0.3662;0;35.0018;0]
B  =  [δe]

C  =  [1 0 0 0 0;0 1 0 0 0 ;0 0 1 0 0;0 0 0 1 0;0 0 0 0 1]
C  = [u  α  θ  q  h]

D  =  [0;0;0;0;0]
D  =  [δe]

iu  =  1;

[num,den]  =  ss2tf(A,B,C,D,iu);

sys =  ss(A,B,C,D);
```

HAHS_ss = c2d(sys,.05)

printsys[num,den]

[zz,pp,kk] = ss2zp(A,B,C,D)

A =

	u	**α**	**θ**	**q**	**h**
u	-0.077	5.669	-9.81	0	0
α	-0.0022	-5.873	0	0.93	0
θ	0	0	0	1	0
q	0.0004	-41.88	0	-2.854	0
h	0	-1.658	1.658	0	0

B =

	δe
u	0
α	-0.3662
θ	0
q	35
h	0

C =

	u	**α**	**θ**	**q**	**h**
u	1	0	0	0	0
α	0	1	0	0	0
θ	0	0	1	0	0
q	0	0	0	1	0
h	0	0	0	0	1

D =

	δe
u	0
α	0
θ	0
q	0
h	0

Continuous-time model.

A =

	x1	x2	x3	x4	x5
x1	0.9961	0.249	-0.4896	-0.00594	0
x2	-9.325e-005	0.7076	2.423e-005	0.03682	0
x3	2.187e-006	-0.04499	1	0.0459	0
x4	0.0001172	-1.658	-2.145e-005	0.8272	0
x5	4.145e-006	-0.07201	0.08289	0.000306	1

b =

	u1
x1	-0.005808
x2	0.01935
x3	0.04171
x4	1.623
x5	0.0008473

c =

	x1	x2	x3	x4	x5
y1	1	0	0	0	0
y2	0	1	0	0	0
y3	0	0	1	0	0
y4	0	0	0	1	0
y5	0	0	0	0	1

d =

	u1
y1	0
y2	0
y3	0
y4	0
y5	0

Sampling time: 0.05
Discrete-time model.

```
num =

  1.0e+003 *

  Columns 1 through 5

         0   -0.0000   -0.0021   -0.1648   -2.1672
         0   -0.0004    0.0315    0.0024    0.0008
         0   -0.0000    0.0350    0.2236    0.0174
         0    0.0350    0.2236    0.0174    0.0000
         0   -0.0000    0.0006    0.0058    0.3667

  Column 6

         0
         0
         0
         0
    0.0277


den =

  Columns 1 through 5

    1.0000    8.8040   56.3963    4.3272    0.9270

  Column 6

         0
```

Speed to Elevator (u/δe) Transfer Function

num(1)/den =

$$\frac{-1.7764e\text{-}015\ s^4 - 2.076\ s^3 - 164.7531\ s^2 - 2167.1985\ s}{s^5 + 8.804\ s^4 + 56.3963\ s^3 + 4.3272\ s^2 + 0.92697\ s}$$

Angle of Attack to Elevator (α/δe) Transfer Function

num(2)/den =

$$\frac{-0.3662\ s^4 + 31.4785\ s^3 + 2.426\ s^2 + 0.75397\ s}{s^5 + 8.804\ s^4 + 56.3963\ s^3 + 4.3272\ s^2 + 0.92697\ s}$$

Pitch to Elevator (θ/δe) Transfer Function
num(3)/den =

$$\frac{-1.2434e\text{-}014\ s^4 + 35.0018\ s^3 + 223.6124\ s^2 + 17.4463\ s}{s^5 + 8.804\ s^4 + 56.3963\ s^3 + 4.3272\ s^2 + 0.92697\ s}$$

Pitch rate to Elevator (p/δe) Transfer Function
num(4)/den =

$$\frac{35.0018\ s^4 + 223.6124\ s^3 + 17.4463\ s^2 + 8.1046e\text{-}015\ s}{s^5 + 8.804\ s^4 + 56.3963\ s^3 + 4.3272\ s^2 + 0.92697\ s}$$

Height to Elevator (h/δe) Transfer Function
num(5)/den =

$$\frac{-7.1054e\text{-}015\ s^4 + 0.60712\ s^3 + 5.8413\ s^2 + 366.7049\ s + 27.6743}{s^5 + 8.804\ s^4 + 56.3963\ s^3 + 4.3272\ s^2 + 0.92697\ s}$$

```
ZZ =

  Columns 1 through 3

 -62.7143            86.0371            -0.0790
 -16.6456            -0.0386 + 0.1498i  -6.3096
   0.0000           -0.0386 - 0.1498i        0
      Inf                  0                Inf



 Columns 4 through 5

       0           -4.7729 +24.0936i
  -6.3096            -4.7729 -24.0936i
  -0.0790            -0.0756
       0               Inf


PP =

       0
  -4.3645 + 6.0561i
  -4.3645 - 6.0561i
  -0.0375 + 0.1234i
  -0.0375 - 0.1234i

KK =

   -2.0760
   -0.3662
   35.0018
   35.0018
    0.6071
```

A.3.2 Lateral Mode

A = [**β ϕ Ψ p r]**
B = **[δa]**
C = [**β ϕ Ψ p r]**
D = **[δa]**

A =

-0.2954	0.1509	0.0001	0	-0.9930
0	0	1.0000	0	0
-36.2243	0	-8.9888	0	2.4260
0	0	0	0	1.0000
10.9816	0	-3.1858	0	-0.4554

B =

0
0
34.3115
0
7.7691

C =

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

D =

0
0
0
0
0

ZZ =

Columns 1 through 3

```
   5.8700            -0.6501 + 4.3501i        0
  -0.0000            -0.6501 - 4.3501i  -0.6501 + 4.3501i
  -0.1149                  0            -0.6501 - 4.3501i
     Inf                Inf                  0
```

Columns 4 through 5

```
   4.8822             4.4822
  -1.5440            -1.5440
   1.8474             1.8474
     Inf                 0
```

PP =

```
        0
  -9.5082
  -0.1191 + 4.8278i
  -0.1191 - 4.8278i
   0.0069
```

KK =

```
  -7.7113
  34.3115
  34.3115
   7.7691
   7.7691
```

num =

Columns 1 through 5

```
      0   -0.0000   -7.7113   44.3794    5.2020
      0    0.0000   34.3115   44.6089  663.8012
      0   34.3115   44.6089  663.8012   -0.0000
      0    0.0000    7.7691  -37.1797  -11.5950
      0    7.7691  -37.1797  -11.5950   99.32
```

```
  Column 6

         0
         0
         0
   99.3262
         0


den =

  Columns 1 through 5

    1.0000    9.7396   25.5204  221.5735   -1.5309

  Column 6

         0
```

Sideslip to aileron ($\beta/\delta a$) Transfer Function

```
num(1)/den =

   -1.2434e-014 s^4 - 7.7113 s^3 + 44.3794 s^2 + 5.202 s
  --------------------------------------------------------
  s^5 + 9.7396 s^4 + 25.5204 s^3 + 221.5735 s^2 - 1.5309 s
```

Roll angle to aileron ($\phi/\delta a$) Transfer Function

```
num(2)/den =

  2.3093e-014 s^4 + 34.3115 s^3 + 44.6089 s^2 + 663.8012 s
  --------------------------------------------------------
  s^5 + 9.7396 s^4 + 25.5204 s^3 + 221.5735 s^2 - 1.5309 s
```

Roll Rate to aileron ($p/\delta a$) Transfer Function

```
num(3)/den =

  34.3115 s^4 + 44.6089 s^3 + 663.8012 s^2 - 1.6209e-013 s
  --------------------------------------------------------
  s^5 + 9.7396 s^4 + 25.5204 s^3 + 221.5735 s^2 - 1.5309 s
```

Yaw to aileron ($\Psi/\delta a$) Transfer Function

```
num(4)/den =

  7.1054e-015 s^4 + 7.7691 s^3 - 37.1797 s^2 - 11.595 s + 99.3262
  ---------------------------------------------------------------
     s^5 + 9.7396 s^4 + 25.5204 s^3 + 221.5735 s^2 - 1.5309 s
```

Yaw Rate to aileron ($r/\delta a$) Transfer Function

```
num(5)/den =

     7.7691 s^4 - 37.1797 s^3 - 11.595 s^2 + 99.3262 s
  --------------------------------------------------------
  s^5 + 9.7396 s^4 + 25.5204 s^3 + 221.5735 s^2 - 1.5309 s
```

II Training data Generation – Post Training Analysis

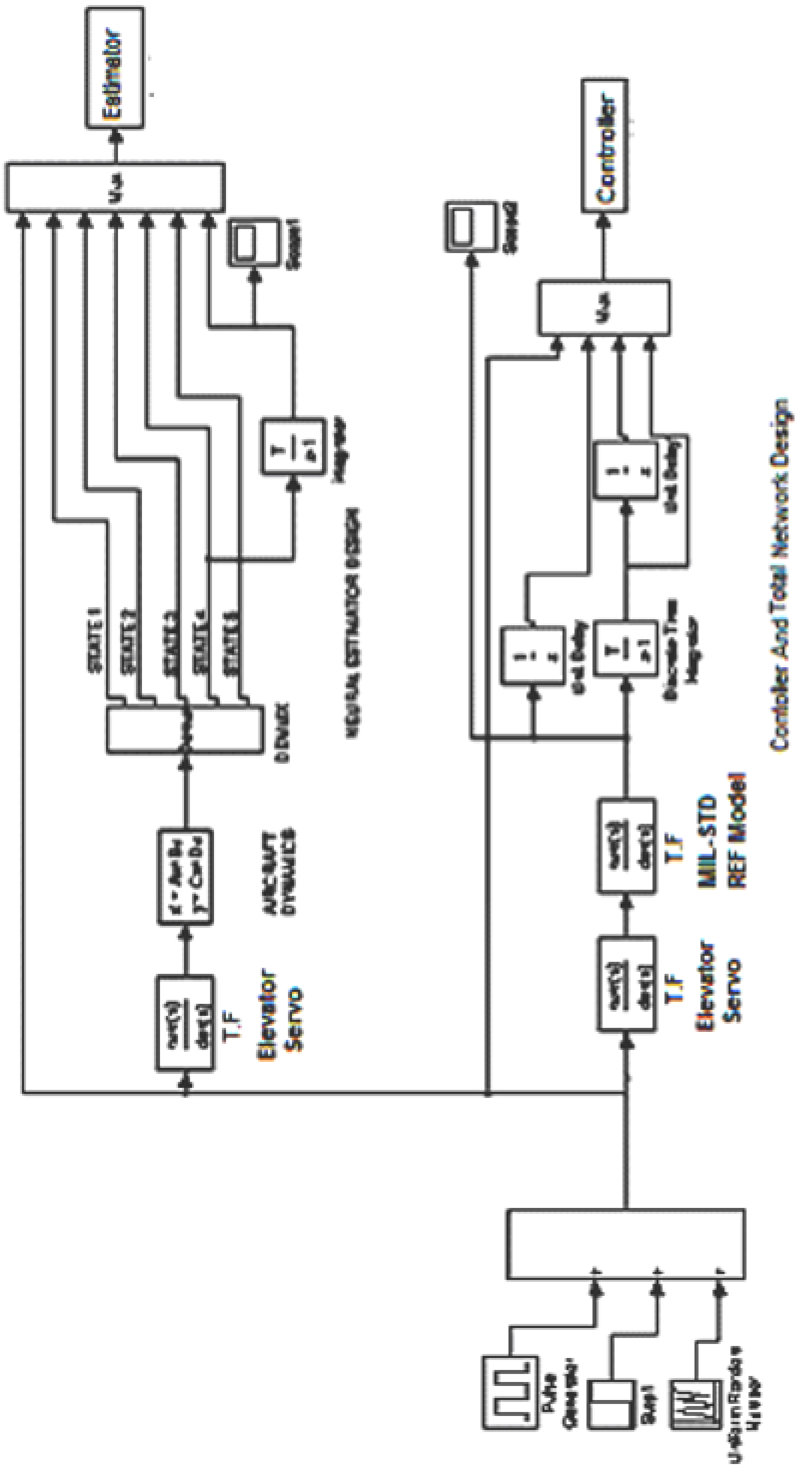


Fig A.1   Basic Block Diagram of Training Network

III Model training session

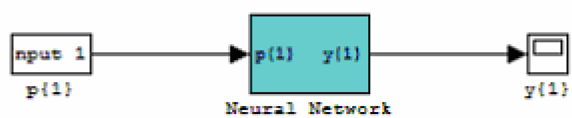
TRAINLM, Epoch 0/500, MSE 1.07304e+006/1e-01, Gradient 9.90309e+009/1e-010
TRAINLM, Epoch 5/500, MSE 0.00193682/1e-010, Gradient 241257/1e-010
TRAINLM, Epoch 10/500, MSE 2.89731e-006/1e-010, Gradient 3853.49/1e-010
TRAINLM, Epoch 15/500, MSE 3.7775e-007/1e-010, Gradient 98.1025/1e-010
TRAINLM, Epoch 20/500, MSE 2.96953e-007/1e-010, Gradient 85.797/1e-010
TRAINLM, Epoch 25/500, MSE 2.40323e-007/1e-010, Gradient 39.9062/1e-010
TRAINLM, Epoch 30/500, MSE 2.05305e-007/1e-010, Gradient 2.9601/1e-010
TRAINLM, Epoch 35/500, MSE 1.99814e-007/1e-010, Gradient 3.73385/1e-010
TRAINLM, Epoch 40/500, MSE 1.95443e-007/1e-010, Gradient 2.82125/1e-010
TRAINLM, Epoch 45/500, MSE 1.86932e-007/1e-010, Gradient 14.669/1e-010
TRAINLM, Epoch 50/500, MSE 1.78222e-007/1e-010, Gradient 57.2569/1e-010
TRAINLM, Epoch 55/500, MSE 1.60421e-007/1e-010, Gradient 20.6774/1e-010
TRAINLM, Epoch 60/500, MSE 1.47161e-007/1e-010, Gradient 23.9514/1e-010
TRAINLM, Epoch 65/500, MSE 1.37778e-007/1e-010, Gradient 16.9432/1e-010
TRAINLM, Epoch 70/500, MSE 1.29673e-007/1e-010, Gradient 13.0583/1e-010
TRAINLM, Epoch 75/500, MSE 1.22521e-007/1e-010, Gradient 10.7753/1e-010
TRAINLM, Epoch 80/500, MSE 1.16145e-007/1e-010, Gradient 9.27313/1e-010
TRAINLM, Epoch 85/500, MSE 9.95819e-008/1e-010, Gradient 513.254/1e-010
TRAINLM, Epoch 90/500, MSE 7.31175e-008/1e-010, Gradient 155.243/1e-010
TRAINLM, Epoch 95/500, MSE 5.98047e-008/1e-010, Gradient 75.0702/1e-010
TRAINLM, Epoch 100/500, MSE 5.10544e-008/1e-010, Gradient 41.0905/1e-010
TRAINLM, Epoch 105/500, MSE 4.50858e-008/1e-010, Gradient 24.2643/1e-010
TRAINLM, Epoch 110/500, MSE 4.08806e-008/1e-010, Gradient 15.0486/1e-010
TRAINLM, Epoch 115/500, MSE 3.77993e-008/1e-010, Gradient 9.93836/1e-010
TRAINLM, Epoch 120/500, MSE 3.54409e-008/1e-010, Gradient 7.17603/1e-010
TRAINLM, Epoch 125/500, MSE 3.35554e-008/1e-010, Gradient 5.75861/1e-010
TRAINLM, Epoch 130/500, MSE 3.19859e-008/1e-010, Gradient 5.12003/1e-010
TRAINLM, Epoch 135/500, MSE 3.06331e-008/1e-010, Gradient 4.95535/1e-010
TRAINLM, Epoch 140/500, MSE 2.94335e-008/1e-010, Gradient 5.11601/1e-010
TRAINLM, Epoch 145/500, MSE 2.83458e-008/1e-010, Gradient 5.55455/1e-010
TRAINLM, Epoch 150/500, MSE 2.73432e-008/1e-010, Gradient 6.3081/1e-010
TRAINLM, Epoch 155/500, MSE 2.64071e-008/1e-010, Gradient 7.51889/1e-010
TRAINLM, Epoch 160/500, MSE 2.55225e-008/1e-010, Gradient 9.5121/1e-010
TRAINLM, Epoch 165/500, MSE 2.46704e-008/1e-010, Gradient 13.0001/1e-010
TRAINLM, Epoch 170/500, MSE 2.38125e-008/1e-010, Gradient 19.6427/1e-010
TRAINLM, Epoch 175/500, MSE 2.28333e-008/1e-010, Gradient 33.9545/1e-010
TRAINLM, Epoch 180/500, MSE 2.13066e-008/1e-010, Gradient 76.7027/1e-010
TRAINLM, Epoch 185/500, MSE 1.93879e-008/1e-010, Gradient 226.328/1e-010
TRAINLM, Epoch 190/500, MSE 1.6393e-008/1e-010, Gradient 149.383/1e-010
TRAINLM, Epoch 195/500, MSE 1.45281e-008/1e-010, Gradient 59.1147/1e-010

TRAINLM, Epoch 200/500, MSE 1.34057e-008/1e-010, Gradient 48.7485/1e-010
TRAINLM, Epoch 205/500, MSE 1.32052e-008/1e-010, Gradient 6.8286/1e-010
TRAINLM, Epoch 210/500, MSE 1.30405e-008/1e-010, Gradient 46.0211/1e-010
TRAINLM, Epoch 215/500, MSE 1.2453e-008/1e-010, Gradient 5.33685/1e-010
TRAINLM, Epoch 220/500, MSE 1.18148e-008/1e-010, Gradient 8.38775/1e-010
TRAINLM, Epoch 225/500, MSE 1.10531e-008/1e-010, Gradient 4.89203/1e-010
TRAINLM, Epoch 230/500, MSE 1.07301e-008/1e-010, Gradient 6.80886/1e-010
TRAINLM, Epoch 235/500, MSE 1.0507e-008/1e-010, Gradient 1.81323/1e-010
TRAINLM, Epoch 240/500, MSE 9.5942e-009/1e-010, Gradient 132.353/1e-010
TRAINLM, Epoch 245/500, MSE 8.52874e-009/1e-010, Gradient 6.30973/1e-010
TRAINLM, Epoch 250/500, MSE 8.05285e-009/1e-010, Gradient 7.09839/1e-010
TRAINLM, Epoch 255/500, MSE 7.61701e-009/1e-010, Gradient 7.57393/1e-010
TRAINLM, Epoch 260/500, MSE 7.21169e-009/1e-010, Gradient 7.96109/1e-010
TRAINLM, Epoch 265/500, MSE 6.83068e-009/1e-010, Gradient 8.27607/1e-010
TRAINLM, Epoch 270/500, MSE 6.46988e-009/1e-010, Gradient 8.5136/1e-010
TRAINLM, Epoch 275/500, MSE 6.12654e-009/1e-010, Gradient 8.66811/1e-010
TRAINLM, Epoch 280/500, MSE 5.79876e-009/1e-010, Gradient 8.73458/1e-010
TRAINLM, Epoch 285/500, MSE 5.48528e-009/1e-010, Gradient 8.70898/1e-010
TRAINLM, Epoch 290/500, MSE 5.18524e-009/1e-010, Gradient 8.58797/1e-010
TRAINLM, Epoch 295/500, MSE 4.89815e-009/1e-010, Gradient 8.37048/1e-010
TRAINLM, Epoch 300/500, MSE 4.62371e-009/1e-010, Gradient 8.05859/1e-010
TRAINLM, Epoch 305/500, MSE 4.3618e-009/1e-010, Gradient 7.65852/1e-010
TRAINLM, Epoch 310/500, MSE 4.1124e-009/1e-010, Gradient 7.18041/1e-010
TRAINLM, Epoch 315/500, MSE 3.87552e-009/1e-010, Gradient 6.63825/1e-010
TRAINLM, Epoch 320/500, MSE 3.65122e-009/1e-010, Gradient 6.04861/1e-010
TRAINLM, Epoch 325/500, MSE 3.43951e-009/1e-010, Gradient 5.4294/1e-010
TRAINLM, Epoch 330/500, MSE 3.24037e-009/1e-010, Gradient 4.79851/1e-010
TRAINLM, Epoch 335/500, MSE 3.05374e-009/1e-010, Gradient 4.17272/1e-010
TRAINLM, Epoch 340/500, MSE 2.87947e-009/1e-010, Gradient 3.56704/1e-010
TRAINLM, Epoch 345/500, MSE 2.71734e-009/1e-010, Gradient 2.99374/1e-010
TRAINLM, Epoch 350/500, MSE 2.56702e-009/1e-010, Gradient 2.46238/1e-010
TRAINLM, Epoch 355/500, MSE 1.65054e-009/1e-010, Gradient 13.1254/1e-010
TRAINLM, Epoch 360/500, MSE 1.22363e-009/1e-010, Gradient 23.2065/1e-010
TRAINLM, Epoch 365/500, MSE 9.8263e-010/1e-010, Gradient 11.6357/1e-010
TRAINLM, Epoch 370/500, MSE 8.35506e-010/1e-010, Gradient 5.28806/1e-010
TRAINLM, Epoch 375/500, MSE 7.36023e-010/1e-010, Gradient 3.55964/1e-010
TRAINLM, Epoch 380/500, MSE 6.63978e-010/1e-010, Gradient 2.49682/1e-010
TRAINLM, Epoch 385/500, MSE 6.0993e-010/1e-010, Gradient 1.84617/1e-010
TRAINLM, Epoch 390/500, MSE 5.67718e-010/1e-010, Gradient 0.26092/1e-010
TRAINLM, Epoch 395/500, MSE 5.35127e-010/1e-010, Gradient 2.75723/1e-010
TRAINLM, Epoch 400/500, MSE 5.0882e-010/1e-010, Gradient 6.35681/1e-010
TRAINLM, Epoch 405/500, MSE 4.89506e-010/1e-010, Gradient 9.42287/1e-010
TRAINLM, Epoch 410/500, MSE 4.69659e-010/1e-010, Gradient 11.0706/1e-010
TRAINLM, Epoch 415/500, MSE 4.52532e-010/1e-010, Gradient 10.1864/1e-010
TRAINLM, Epoch 420/500, MSE 4.33189e-010/1e-010, Gradient 8.15139/1e-010
TRAINLM, Epoch 425/500, MSE 4.17039e-010/1e-010, Gradient 4.20092/1e-010

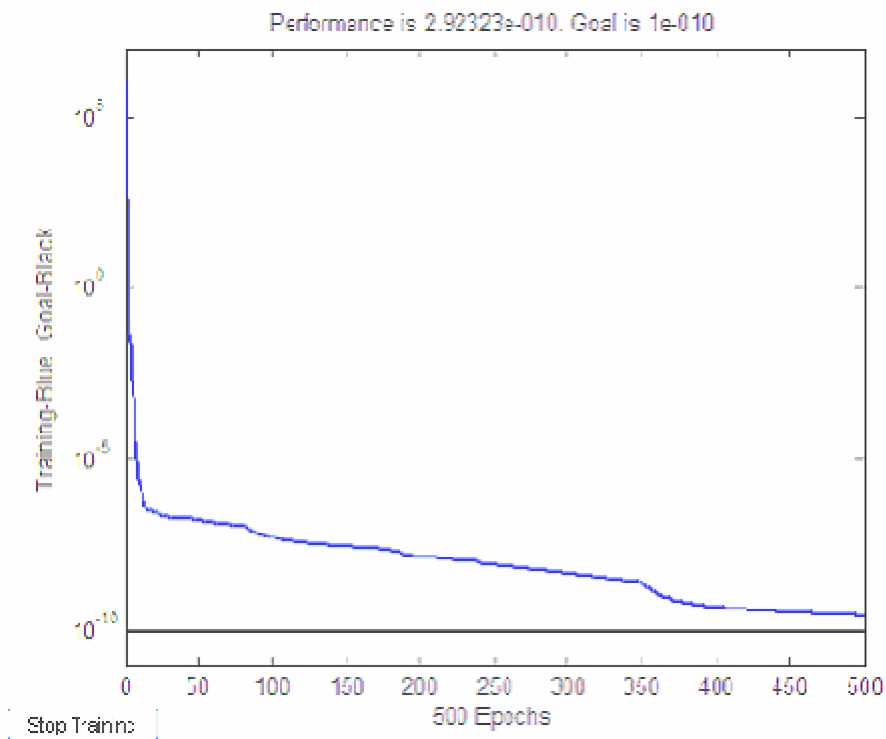
TRAINLM, Epoch 430/500, MSE 4.02685e-010/1e-010, Gradient 1.87239/1e-010
TRAINLM, Epoch 435/500, MSE 3.90754e-010/1e-010, Gradient 1.20416/1e-010

TRAINLM, Epoch 440/500, MSE 3.79998e-010/1e-010, Gradient 1.96774/1e-010
TRAINLM, Epoch 445/500, MSE 3.70269e-010/1e-010, Gradient 3.02196/1e-010
TRAINLM, Epoch 450/500, MSE 3.61095e-010/1e-010, Gradient 3.03228/1e-010
TRAINLM, Epoch 455/500, MSE 3.52543e-010/1e-010, Gradient 3.22549/1e-010
TRAINLM, Epoch 460/500, MSE 3.44456e-010/1e-010, Gradient 3.16928/1e-010
TRAINLM, Epoch 465/500, MSE 3.36823e-010/1e-010, Gradient 3.18188/1e-010
TRAINLM, Epoch 470/500, MSE 3.29574e-010/1e-010, Gradient 3.15426/1e-010
TRAINLM, Epoch 475/500, MSE 3.22675e-010/1e-010, Gradient 3.13781/1e-010
TRAINLM, Epoch 480/500, MSE 3.16088e-010/1e-010, Gradient 3.11539/1e-010
TRAINLM, Epoch 485/500, MSE 3.09782e-010/1e-010, Gradient 3.09239/1e-010
TRAINLM, Epoch 490/500, MSE 3.03733e-010/1e-010, Gradient 3.06691/1e-010
TRAINLM, Epoch 495/500, MSE 2.97919e-010/1e-010, Gradient 3.03973/1e-010
TRAINLM, Epoch 500/500, MSE 2.92323e-010/1e-010, Gradient 3.00974/1e-010
TRAINLM, Maximum epoch reached, performance goal was not met.

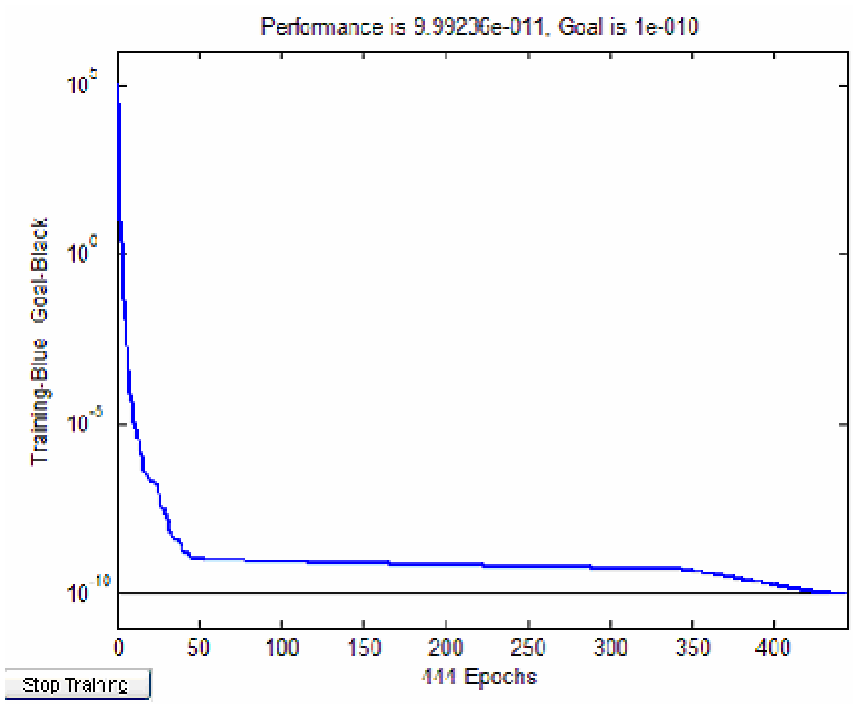
Evaluating callback 'LoadFcn' for neural/Control Systems/X(2Y) Graph
Callback: sfunxy2([],[],[],'LoadBlock')



A.2 Neural Network



A.3 Network Convergence



A.4 Network Convergence

REFERENCES

[1] J.H.Blakelock (1990), ‘ Automatic Control of Aircraft and Missiles.’, John-Wiley Sons-New York.

[2] Donald McLean(1990),’Automatic Flight Control Systems’, Prentice-Hall Publications.

[3] B.L.Stevens & Franc L Lewis(1992), ‘Aircraft Control and Simulation’, a Wley Inter Science publications.

[4] John Roskam (1979) ‘ Airplane flight dynamic and Automatic flight controls- part I & II, Roskam Aviation and Engineering Corporation.

[5] A.C .Kermode (1996) ‘Mechanics of Flight’ Himalayan Publications,.

[6] Willium Branch (1998) ‘Auto-pilot design rules and guidelines’. AIAA

[7] Simulink Users’ Manual / Users’ guide.1994-The Math Works Inc.

[8] Matlab Users’ Manual / Users’ guide.1997-The Math Works Inc.

[9] Control System Tool box Users’ Manual / Users’ guide.1994-The Math Works Inc.

[10] K.V. Srinivasan, Satendra Singh, BVL Narayana, Atit Misra and Ravi N. Bangar 2002) ,Controller Synthesis and real-time simulation of the net-recovery phase of a remotely Piloted Vehicle’, IEEE International conference on control applications,

[11] Satish Kumar-'Neural Networks-A Classroom Approach' –Tata McGraw Hill Publishing Company

[12] Martin T. Hagan, Howard B. Demuth and Mark Beale—'Neural Network Design' –Thomson Learning, Vikas Publishing House.

[13] Stephen J. Chapman—' Matlab Programming For Engineers-Third Edition' --Thomson Learning.

[14] Rudra Pratap-- 'Getting Started With Matlab'—Oxford University Press.

[15] S.N.Sivanandam, S.Sumathi S.N.Deepa—' Introduction To Neural Networks Using Matlab 6.0'—Tata McGraw Hill Publishing Company.

[16] Dr. Robert C. Nelson—'Flight Stability And Automatic Control'—McGraw Hill Book Company.

[17] Madan M. Gupta & Dandina H. Rao "Neuro Control Systems '_Theory and Applications.

[18] Patricia Melin and Oscar Castillino – "Intelligent control of aircraft dynamic systems with anew hybrid neuro – fuzzy – fractal approach."

[19] Neural Network User's guide, The Math Works Inc.

[20] Neural Networks in flight control, www.chipcenter.com

[21] Laurence Fausett (1994), ' Fundamentals of Neural Networks – architecture, Algorithms and Applications', Prentice – Hall Publication.

[22] Ha C. M. (1995), ' Neural Networks Approach to AIAA Aircraft Control Design Challenge', Journal of guidance, Control and dynamics.