Radial Basic Function Neural Controller for Pitch Control of an Aircraft

P S Khuntia* and Debjani Mitra**

** Department of Electronics and Communication Engineering,
Durgapur Institute of Advanced Technology and Management, Rajbandh-713212, Durgapur, West Bengal, INDIA
Email: parthsarathi_k@yahoo.com*

*** Department of Electronics and Communication Engineering
Indian School of Mines,Dhanbad-826004, Jharkhand, INDIA, Email: debjani7@yahoo.com*

Abstract

In this paper Radial Basic Function Neural Controller (RBFNC) is designed for the pitch controller of an aircraft to obtain the desired pitch angel as required by the pilot. This controller utilizes a learning mechanism, which observes the plant outputs and adjusts the weights of RBFNC, so that the overall system behaves like a "reference model" which characterizes the desired behavior. The performance of the RBFNC is demonstrated by simulation for various conditions with change in the aircraft dynamics caused due to change in speed of the aircraft and sensor noise.

Keywords: *RBFNC, Reference Model, Fuzzy Inverse Model, Learning Mechanism Pitch Control System*

1. Introduction

The conventional design methods of a control system often require a mathematical model describing the dynamic behavior of the plant to be controlled. When such a mathematical model is difficult to obtain due to uncertainty or complexity of systems the conventional techniques based on a mathematical model are not well suited. Artificial Neural Networks (ANN) in last decade has become popular for plant identification and control [1-2]. A advantage of the ANN is its ability to handle the nonlinear mapping of the input–output space It is well known that backpropagation based ANN suffers from local minima and over fitting problems which is difficult to be implemented in real time due to a large number of neurons in the hidden layer in comparison to the RBFNC[3-5]. Locally tuned and overlapping receptive fields[6] have been found cerebral cortex visual cortex and in other parts of the brain. The concept of localized information processing in the form of receptive fields suggests that such local learning offers alternative computational opportunities to learning with global basis functions.

2. Radial Basic Function Neural Network (RBFNN)

The proposed RBFNN model with single neuron output y is presented in figure-1 consists of three-layers. Each input values are assigned to a node and passed directly to the hidden layer without weights. The hidden layer nodes are called Radial Basic Function (RBF) units which are determined by a parameter vector called center and a scalar called width. The Gaussian density function is used as an activation function for the hidden neurons. The RBFNN shown in Figure 1 has inputs $x_i, i = 1,2,3\ldots n$.

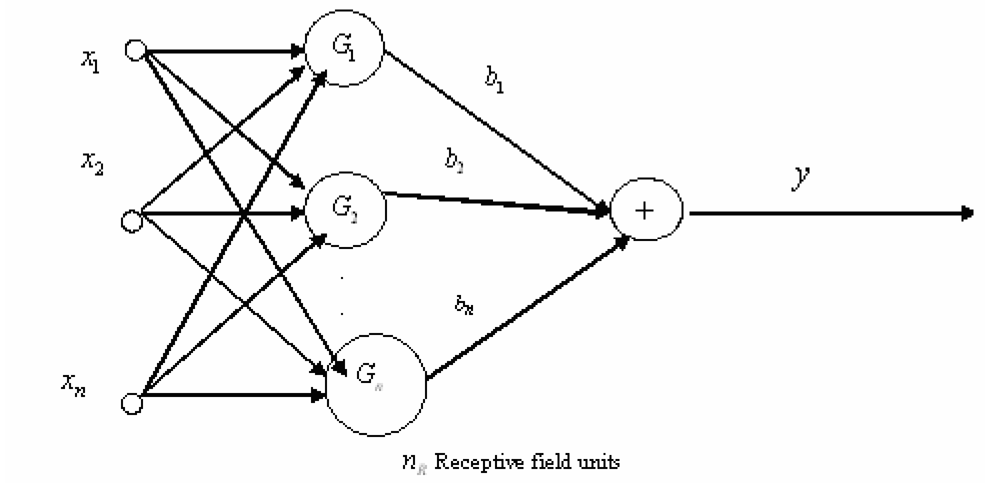


Figure 1: Radial Basic Function Neural Network

and output $y = F_{rbf}(x)$. $x = [x_1, x_2, x_3, ... x_n]^T$ is the input and $R_i(x)$ is the output of the i^{th} receptive field with strength denoted by b_i .Assuming n_R receptive fields present in the RBFNN, the output y can be written as

$$y = F_{rbf}(x, \theta) = \sum_{i=1}^{n_R} b_i R_i(x) \tag{1}$$

where θ holds the parameters of the receptive field units which consist of the parameters b_i and possibly the parameters of the $R_i(x)$. The Gaussian-shaped functions are preferred for analytical convenience i.e.

$$R_i(x) = exp\left[\left|x - c^i\right|^2 \Big/ (\sigma^i)^2\right] \tag{2}$$

where $c_i = [c_1{}^i, c_2{}^i, ... c_n{}^i]^T$ parameterize the locations and σ decides the spreading of the receptive fields in the input space

The weighted average output of the RBF neural network can be written as

$$y = F_{rbf}(x, \theta) = \sum_{i=1}^{n_R} b_i R_i(x) \Big/ \sum_{i=1}^{n_R} R_i(x) \tag{3}$$

3. Problem Formulation

Here the pitch control system of the Bravo fighter aircraft [7] is taken as plant whose pitch angle is to be controlled. The input to the plant is the elevator deflection (δ) and the output is the pitch angle (θ) .The longitudinal dynamics[7] of a aircraft can be represented with following set of equations.

$$\dot{u} = X_u u + X_w w - g\cos\gamma_0\theta$$

$$\dot{w} = Z_u u + Z_w w + U_o q - g\sin\gamma_0\theta + Z_{\delta E}\ \delta_E$$

$$\dot{q} = M_u u + M_w w + M_{\dot{w}}\dot{w} + M_q q + M_{\delta E}\ \delta_E$$

$$\dot{\theta} = q \tag{4}$$

Substituting the values of stability derivatives(Z_W , $M_{\dot{W}}$, M_q , M_W , U_o , $Z_{\delta E}$, $M_{\delta E}$) of the aircraft[7] for flight condition-3 and 4 given the following transfer functions are obtained as follows

Flight Condition-3

$$\frac{\delta(s)}{\theta(s)} = \frac{-0.4500(1+1.6094s)}{[1+(0.0319 + 0.1844i)s]\ [1+(0.0319 -0.1844i)s]} \qquad (5)$$

Flight Condition-4

$$\frac{\delta(s)}{\theta(s)} = \frac{-0.1350(1+2.6045\ s)}{[1+(0.0170 + 0.1469)s][1+(0.0170 - 0.1469)s]} \qquad (6)$$

4. Design of RBFNC

The RBFNC for aircraft pitch control system is shown in the figure 2 tracks the desired pitch angle $\theta_{ref}(kT)$ .The closed loop system has a reference model with input $\theta_{ref}(kT)$ and output $\theta_m(kT)$.

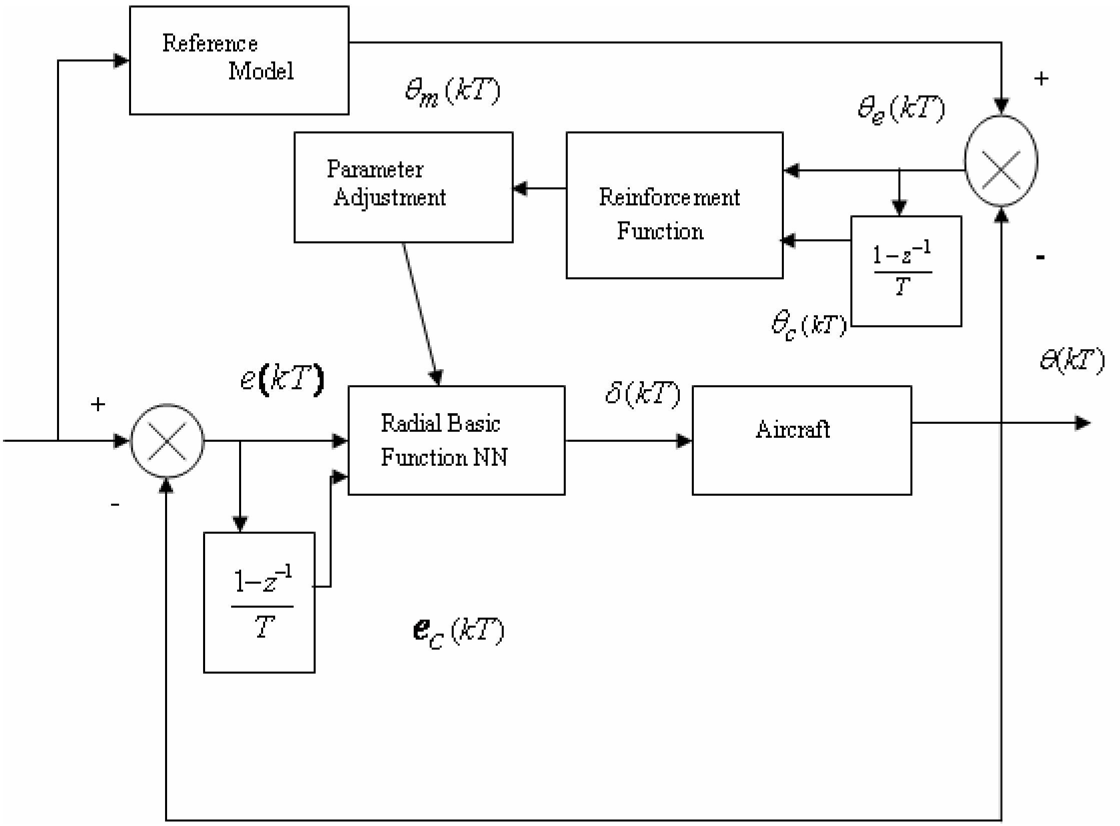


Figure 2: RBFNC for Aircraft Pitch Control

.As shown in Figure 2 the RBFNC attempts $\theta(kT)$ to match the reference model output asymptotically. In this reinforcement learning control[8-9] the error between the plant and the reference model outputs is used to adjust the weights of the neural controller i.e. b_i .

The error $e(kT)$,change of error $e_c(kT)$ are the inputs to the RBF neural network. Where

$$e(kT) = \theta_{ref}(kT) - \theta(kT)$$

$$e_c(kT) == \frac{e(kT) - e(kT-T)}{T}$$

and T is the sampling time.. The output of the RBF neural controller $\delta(k)$ is computed by taking $e(k)$ and $e_c(k)$ as the argument to the radial basic function,

$$\delta(k) = F_{rbf}(e(k), e_c(k)).$$

It is decided in the designing of pitch controller that elevator should not to exceed more than $\pi/2$ radian in either upward or downward direction and the change of error should not be more than 0.01 radian/sec. It concludes range of error $e(kT)$ and change of error $e_c(kT)$ are $e(k) \in [-\pi/2, \pi/2]$ and $e_c(k) \in [-0.01, 0.01]$.

A uniformly grid is created by taking the error and the change of error with the corners of the grid are placed at $(-\pi/2, -0.01)$, $(-\pi/2, 0.01)$, $(\pi/2, 0.01)$ and $(\pi/2, -0.01)$.Each point on the grid contains a receptive field which is a Gaussian function . The error and change of the error's spreading (σ) are taken differently as

$$\sigma_C = 0.7 \frac{0.02}{\sqrt{n_G}}$$

Where n_G is the no of partitions on the grid (n_G =11 here).

$$\sigma_e = 0.7 \frac{\pi}{\sqrt{n_R}}$$

Where n_R is the no of receptive field units in RBFNC which is equal to , $n_R = nG^2$ (n_R =121).

Each center which represents a RBF is represented by circle shown in the Figure 3.

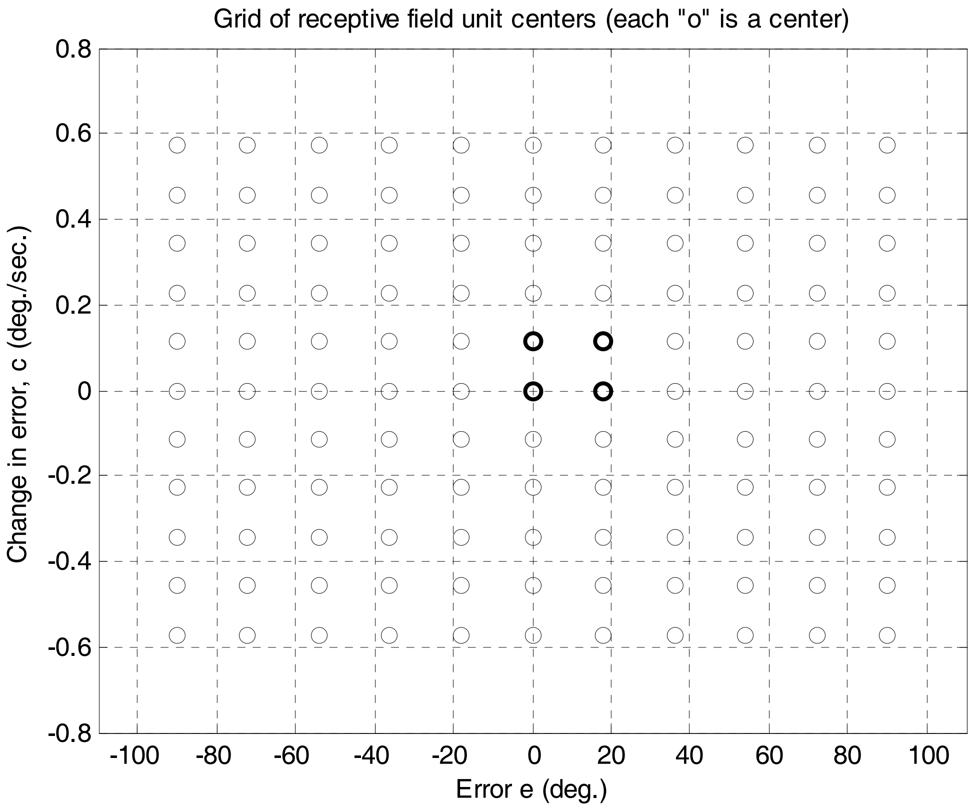


Figure 3: Receptive Field Unit Centers

The left most bottom circle $(\pi/2, -0.01)$ is counted as 1 and the counting increases by 1 making the left most top circle $(-\pi/2, 0.01)$ to be numbered as 11.Next counting starts from bottom circle of the next column with number as 12 and so on.

The input and output mapping of the radial basis function neural network is shaped by choice of scaling parameters b_i .For example an unscaled receptive field R_{73} is plotted in figure 4. Similarly a scaling and summation of the receptive fields shown in the figure 3 as the four dark shaded circles i.e. $2R_{61}(e,c)+3R_{62}(e,c)+R_{72}(e,c)+2R_{73}(e,c)$ is plotted in the Figure 5.

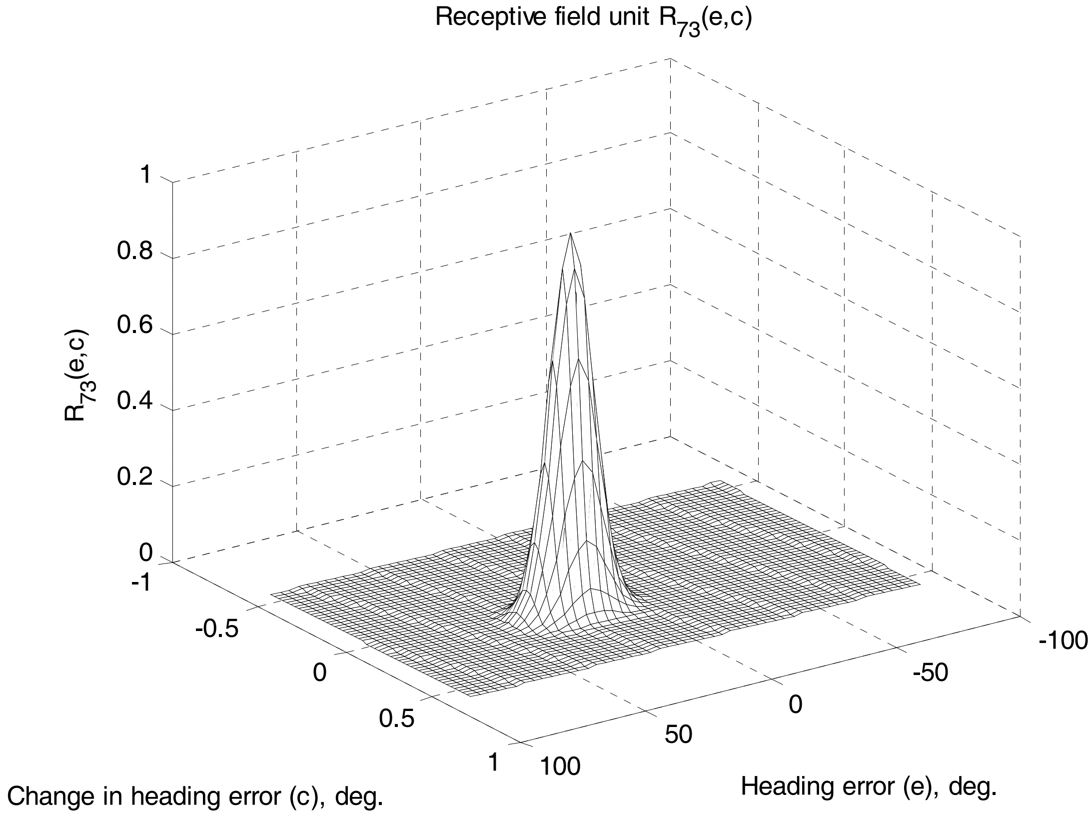


Figure 4: Receptive Fields $R_{73}(e,e_c)$

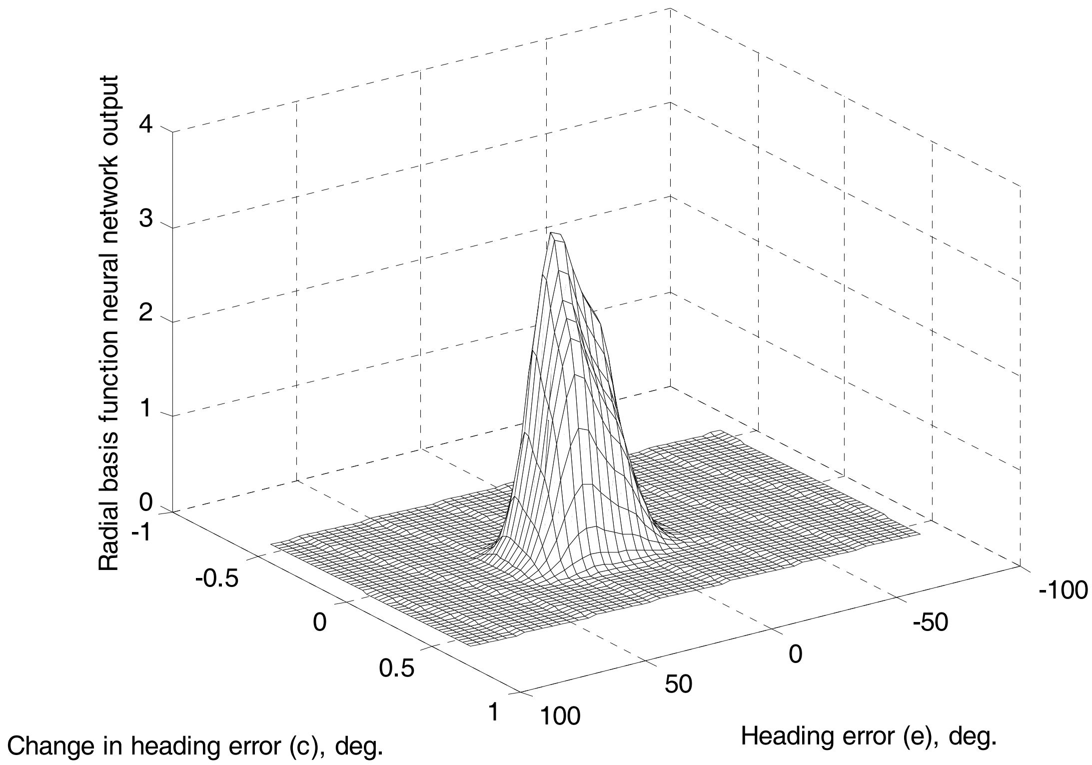


Figure 5: Scaling and Addition of many Receptive Fields
$2R_{61}(e,c)+3R_{62}(e,c)+R_{72}(e,c)+2R_{73}(e,c)$

The purpose of reinforcement function is to modify the field strength as shown in RBF network, i.e. b_i .

The reinforcement function [10] is defined below as

$$J_R(y_e(kT), y_C(kT)) = \eta(-\eta_e y_e(kT) - \eta_C yc(kT)) \quad (7)$$

where

$\eta =$ Adoption gain

$\eta_e =$ Adoption gain $e(k)$

$\eta_C =$ Adoption gain $e_c(k)$

η_e, η_c are so adjusted to indicate the performance of tracking and change in tracking respectively. The smaller value of η indicate slow adoption and large value of η indicates faster rate of adoption of RBF Neural Controller.

The threshold value 'α' beyond which the adoption takes place is chosen here 0.005.

$$J_R(y_e(kT), y_C(kT)) = \begin{cases} J_{1_R}(y_e(kT), y_C(kT)) & \text{if } \left|J_{1_R}(y_e(kT), y_C(kT))\right| \geq \alpha \\ 0 & \text{if } \left|J_{1_R}(y_e(kT), y_C(kT))\right| < \alpha \end{cases}$$

First b_i is initialized to be zero for all values of i to indicate the neural network knows little how to control the pitch of the aircraft and later it is modified. with following equation.

$$b_i(kT) = b_i(kT - T) + J_R(kT) R_i(kT - T) \quad (8)$$

Where $R_i(kT)$ is the output of the i^{th} receptive field unit.

4.1 Reference Model

A first order reference model is proposed here because the pitch angle should be smooth and to follow the reference value without damping or overshoot. So the reference model is given below as

$$\frac{\theta_m(s)}{\theta_{ref}(s)} = \frac{K}{s + a} \quad (9)$$

The input to the reference model are $\theta e(kT)$ and $\theta c(kT)$

Where

$$\theta e(kT) = \theta m(kT) - \theta(kT)$$

$$\theta c(kT) = \frac{\theta e(kT) - \theta e(kT - T)}{T}$$

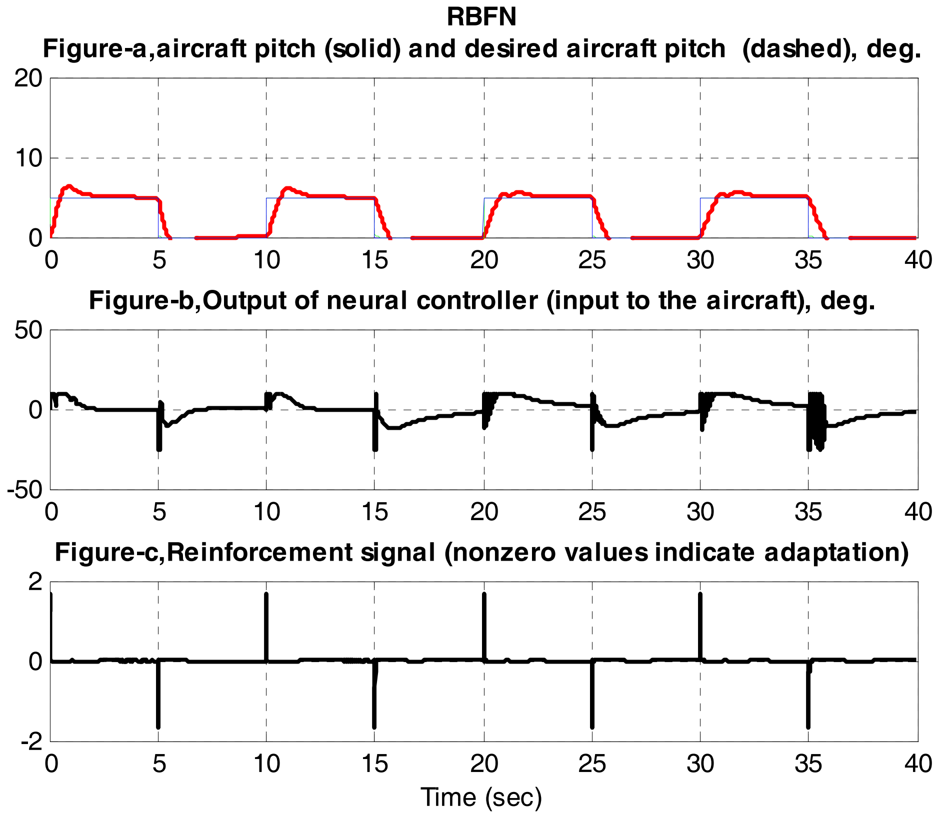
Bilinear Transform is used to find the discrete equivalent of the equation (9) by substituting $s = \frac{2}{T}\left(\frac{z-1}{z+1}\right)$ .With the values of K = 0.1, a = 0.1 and sampling time T = 1 milli second

$$\theta_m(KT + T) = 0.9999\ \theta_m(KT) + 4.9995(10^{-5})\ [\theta_{ref}(KT + T) + \theta_{ref}(KT)] \quad (10)$$

5. Simulation Results

Case 1: (No sensor noise)

The reference signal here is a pulse of duration 40 seconds. From t=0 to 15 seconds the flight travels with flight condition-3(350 m/sec) and after 15 seconds the flight travels with flight condition-4(650 m/sec). It is noted in Figure 6(a) the reference pitch angle $\theta_{ref}(kT)$ and actual pitch angle $\theta(kT)$ are different at initial stage of time due to which the oscillation presents in $\theta(kT)$. It is so because the RBFNC has no idea of adoption to control the aircraft. pitch angle. As the time elapses the controller gets adopted. The out put of the RBFNN $\delta(kT)$ shown in figure 6(b) indicates the controller output is significant when the pitch angle is not equal to the reference value. The reinforcement signal $\delta(kT)$ is shown in figure 6(c). The error $e(kT)$ and change of error $e_c(kT)$ are shown in figure 6(d) and figure 6(e) respectively. The value of $\theta_e(kT)$ and $\theta_c(kT)$ are shown in figure 6(f) and figure 6(g).



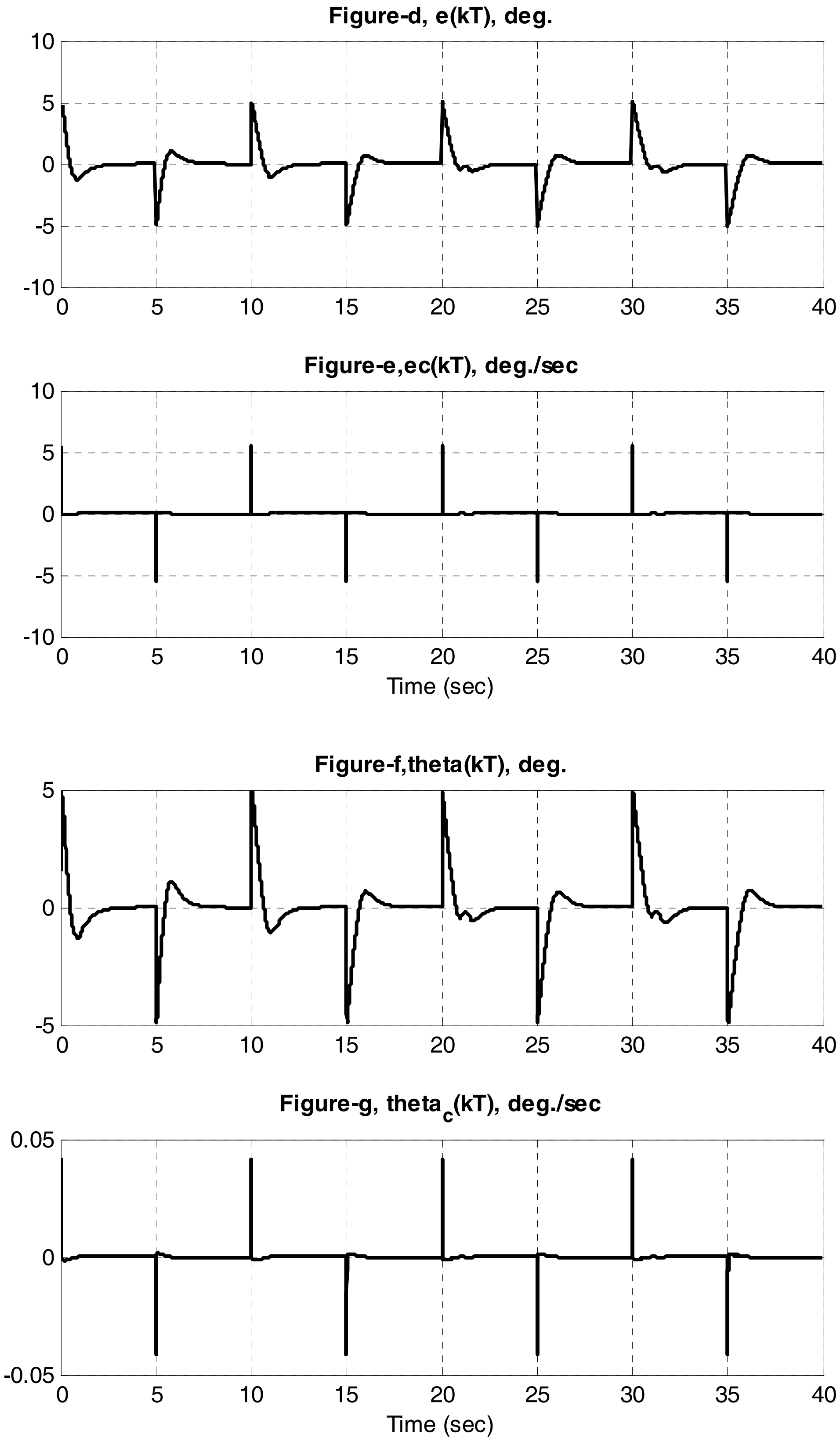


Figure 6: Response of Simulation-(case-1)

Case 2 :(Sensor Noise Presents)

The reference signal is a pulse of duration 40 seconds .From t=0 to 15 seconds the flight travels with flight condition-3 and after 15 seconds the flight travels with flight condition-4 . It may so happens the sensor measuring the pitch angle may be added with noise So a random noise is added uniformly with pitch angle θ with help of a random function i.e. $0.01\frac{\pi}{180}(2\,rand\,\text{-}1)$. The θ and θ_{ref} are plotted in figure

7(a).In Figure 7(b) neural controller output $\delta(kT)$ is found to be continuous because the adoption takes place continuously due to continuous random noise present in the sensor. The continuous output present here because $\theta(kT)$ continuously follow the reference pitch angle $\theta_{ref}(kT)$.Thus the controller is noise adoptive

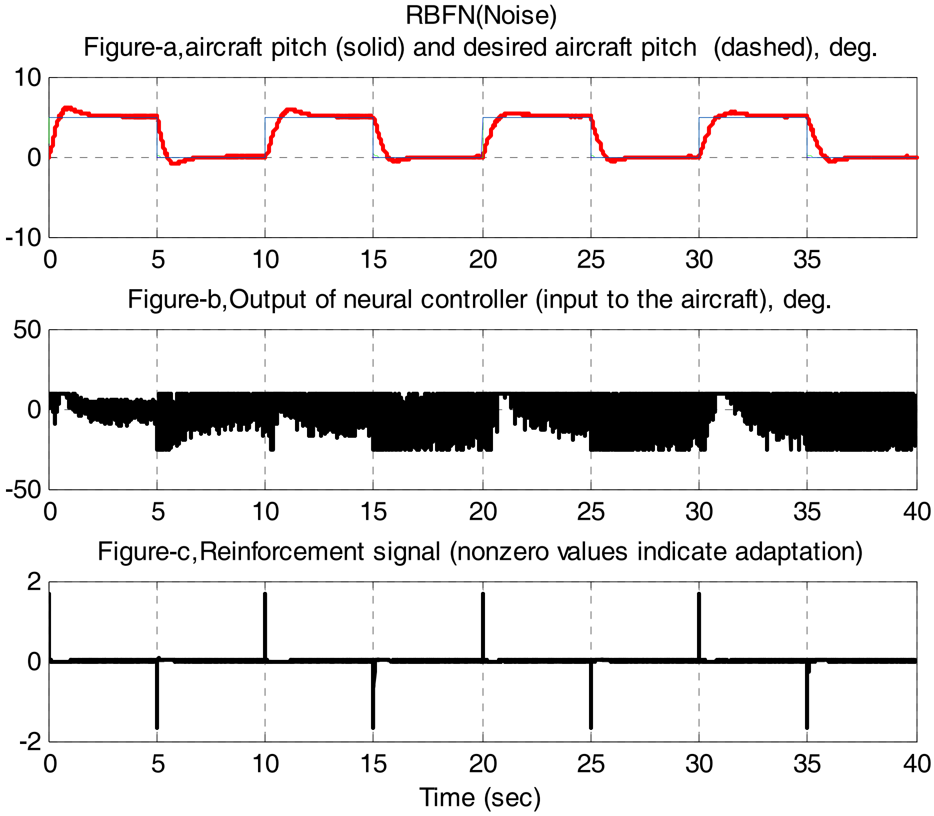


Figure 7: Response of Simulation (case-2)

6. Control Surface

The control surfaces at different time instants are plotted in the following figures. The figure 8 and figure 9 describes the surface at t=2 seconds and t=10 seconds respectively which goes on changing as the simulation proceeds. Figure 10 describes the control surfaces at t=14.999 second which is the time where the aircraft is in flight condition-3 for the last sampling time. Just after that at t=15second the aircraft changes the speed from 350 meter/second to 650 meter/second whose response is shown in figure 11.The controller adopts to the situation and aircraft becomes easier to control to obtain the desired pitch angle. The mapping changes after the simulation ends at t=40 second is shown in Figure 12.It is interesting to see the mapping difference between t=15 second (figure 11) second and t=40 second (figure 12) is shown in figure-13. The positive peak indicates that the mapping has increased at the end of the simulation. In the contour map shown in figure 14 dark color lines indicates decrease and light color lines indicate increase in the mapping. It can be noted that changes are local not global where the system is operating. The Figures 8 to 12 shown below establishes the change in control surface is continuously happening due to the continuous adoption of RBFNC controller. The change is local which means the controller tunes the area in control surface where the error in $\theta_{ref}(kT)$ occurs.

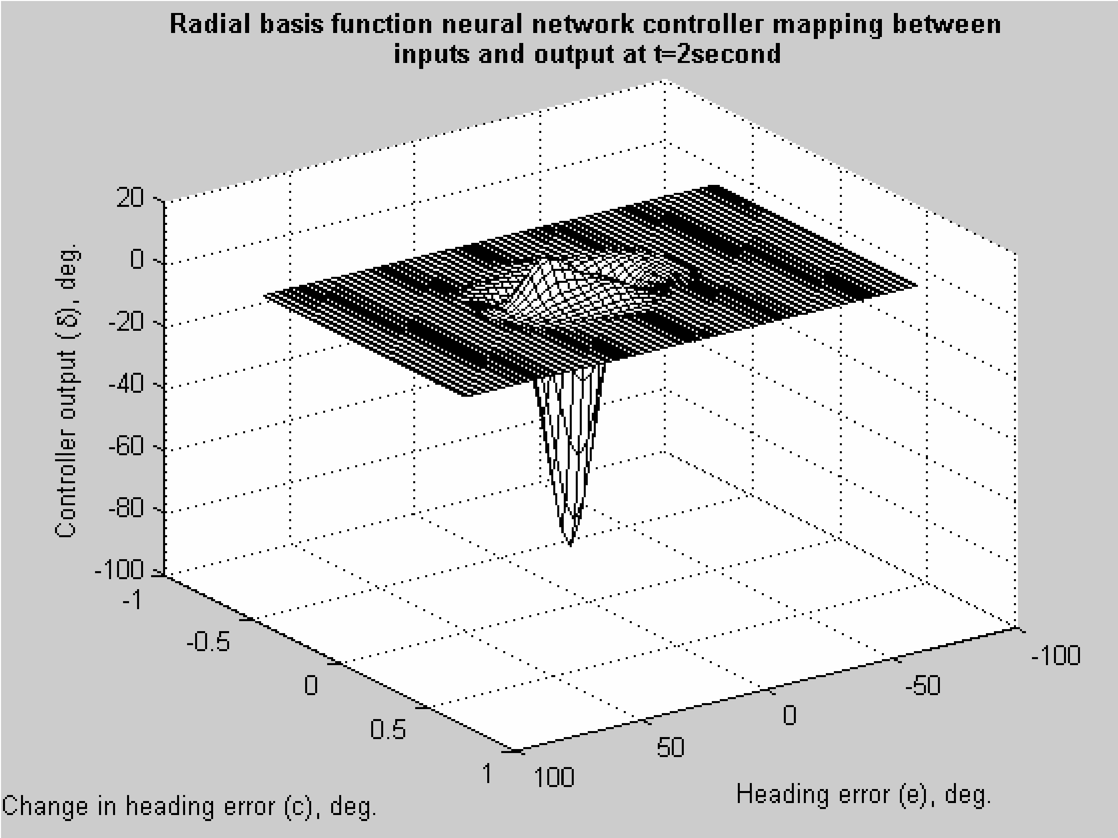


Figure 8: Control Surface at t=2 seconds

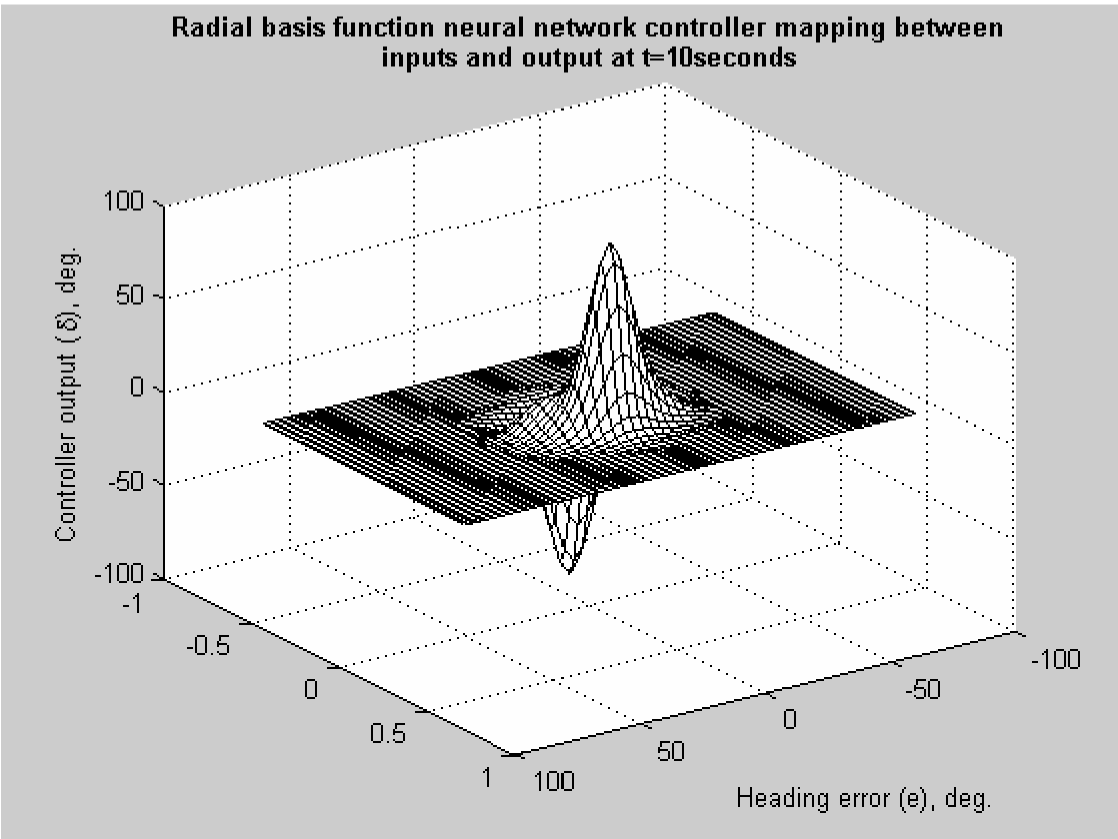


Figure 9: Control Surface at t=10 seconds

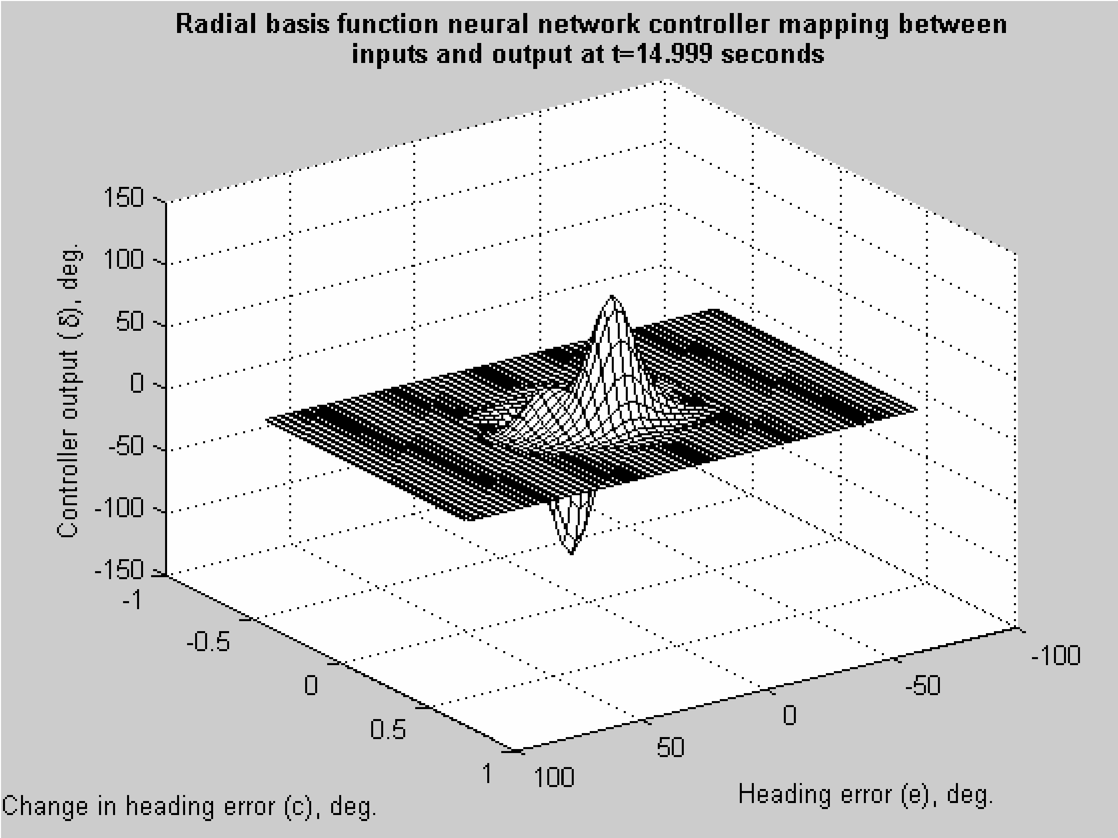


Figure 10: Control Surface at t=14.999 seconds

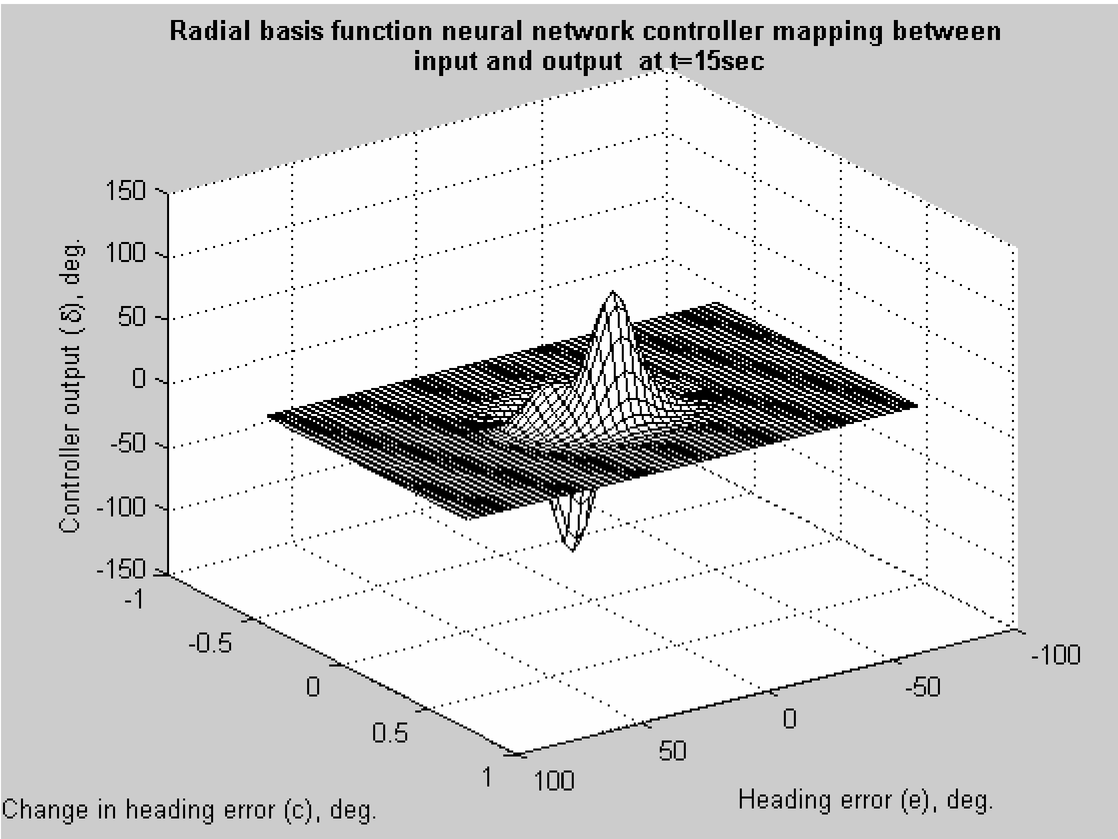


Figure 11: Control Surface at t=15 seconds

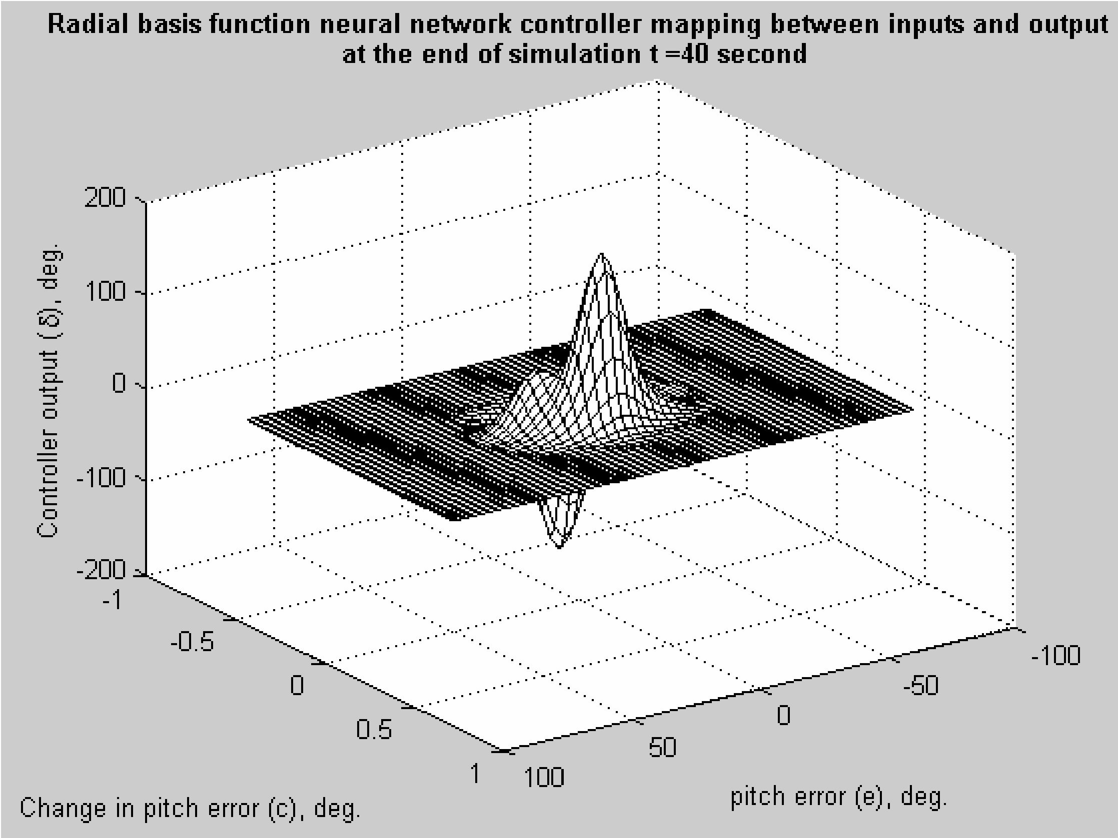


Figure 12: Control Surface at t=40 seconds at the end of Simulation

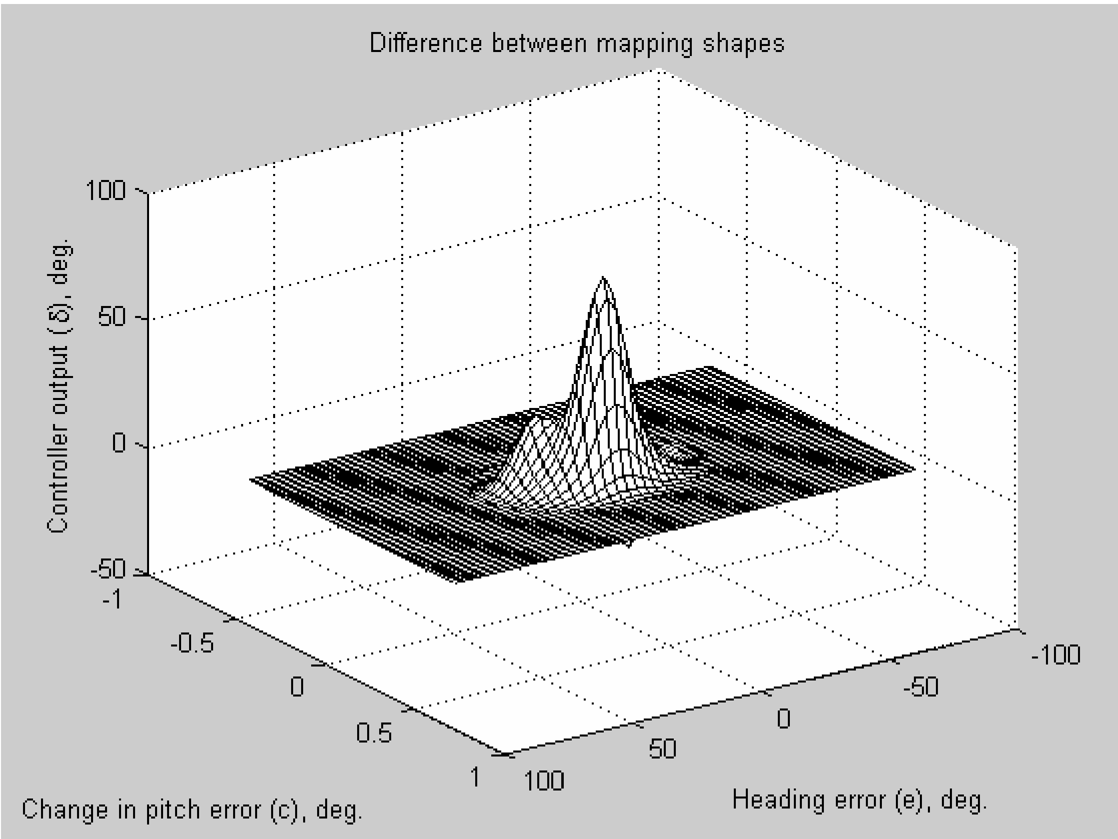


Figure 13: Control Surface difference between t=40 seconds and t= 15 second

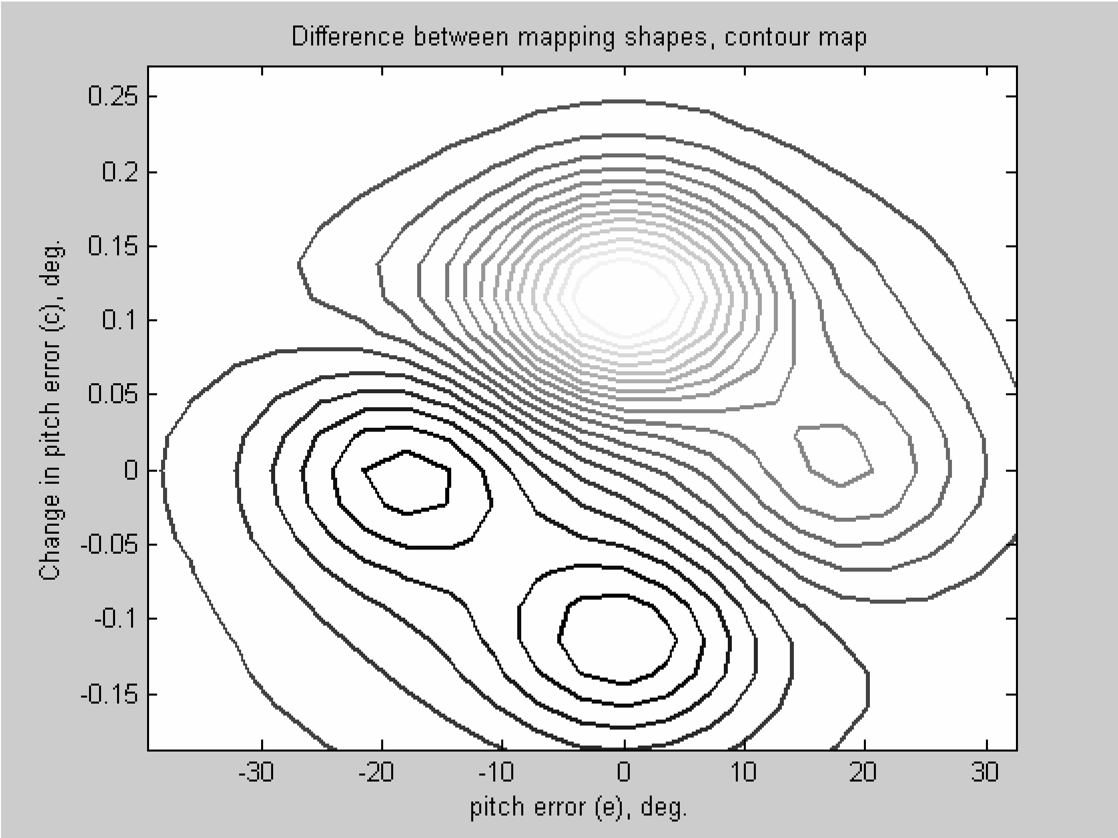


Figure 14: Contour Map

7. Conclusion

The non zero value of the output of RBFNC exhibits it's adoptive nature when ever the actual pitch angle differs from it's reference value and at the time of transition of the reference signal. When the speed of the aircraft is changed the control signal to the pitch control system also changes to cope up with the speed change. It is also shown in this simulation the sensor noise does not affect the out put of the RBFNC because the controller output continuously changes to nullify the effect of this noise. In conclusion the reinforcement learning and adoptive mechanism dynamically tunes the weights of the RBF neural network. The changes in the control surface are local because the controller tunes the area in control surface where the error in desired pitch angle occurs. RBFNC designed in this case is found to be adoptive and robust with strong learning ability.

References

[1] K. S. Narendra and K. Parthasarathy, “Identification and Control of Dynamical Systems using Neural Networks,” *IEEE Trans. on Neural Networks*,vol. 1, no. 1, pp. 4–27, Mar. 1990.

[2] S. Suresh, N. Kannan, S. N. Omkar and V. Mani “Nonlinear Lateral Command Control Using Neural Network for F-16 Aircraft” ,American Control Conference, 2005. June 8-10 Portland, , USA

[3] Li, Y., Sundararajan, N., and Saratchandran, P., “Neuro-Controller Design For Nonlinear Fighter Aircraft Maneuver Using Fully Tuned RBF Networks," *Automatica*, Vol. 37, No. 8, 2001

[4] Ming-guang Zhang, Xing-gui Wang, Man-qiang Liu” Adaptive PID Control Based on RBF Neural Network Identification” Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence,2005, IEEE Computer Society

[5] Wedding, D.K., Eltimsahy, A. “Flexible Link Control using Multiple Forward Paths, Multiple RBF Neural Networks in a Direct Control Application”,IEEE International Conference on Systems, Man, and Cybernetics, 2000 . Volume 4, Issue , Page(s):2619 - 2624 vol.4

[6] Schaal, S., and Atkeson, C., 1998, Constructive Incremental Learning from Only Local Information, Neural Computation, 10, 2047–2084

[7] Automatic Flight Control Systems, Mclean Donald, Prentice Hall International (UK)Ltd, 1990

[8] M. Sedighizadeh, and A. Rezazadeh. “Adaptive PID Controller based on Reinforcement Learning for Wind Turbine Control” Proceedings of World Academy of Science, Engineering And Technology Volume 27 February 2008

[9] 0. Grigore & 0. Grigore“Reinforcement Learning Neural Network Used in a Tracking System Controller” IEEE International workshop on Robot and Human Interactive Communication, Osaka. Japan - September 27-19 2000

[10] Biomimicry for Optimization, Control and Optimization, Kelvino M Passino, Springer-verlag, London Limited-2005.

Article received: 2008-07-20