Design of Learning Flight Control System via Input Matching

Shigeru Uchikado*, Kimio Kanai, Yasuhiro Osa*** and Kanya Tanaka*****

* Fuculty of Science and Engineering, Tokyo Denki University, Ishizaka, Hatoyama-cho, Hiki-county, Saitama Prefecture, 350-03 Japan
** Department of Aerospace Engineering, The National Defense Academy, 1-10-20 Hashirimizu, Yokosuka, Kanagawa Prefecture, 239 Japan
*** Department of Mechanical Engineering, Kobe City College of Technology, 8-3 Gakuen Higashi-cho, Nishi-Ku, Kobe, Hyogo Prefecture, 651-21 Japan
**** Fuculty of Engineering, Ehime University, 3 Bunkyo-cho, Matsuyama, Ehime Prefecture, 790 Japan

Abstracts: In this paper, a design method of learning flight control system via input matching is proposed. The proposed learning Control system is a simple structure which has an artificial neural network and feedback mechanism, and it is a useful method to control nonlinear systems.

Keywords Neural Network, Perceptron, Nonlinear System, Flight Control, Adaptive Control

1. Introduction

Adaptive control system and neural control[1)2)] can be given as control system having a learning function. Generally, adaptive control system is desigened for a linear system, until now adaptive control systems having robustness are mainly researched.

On the other hand, neural control[3)] which has neural networks considering a mechanism of human's brain is dealed for a nonlinear system. Neural control can be classified into Multilayer Network or Recurrent Network by a method of combination of neuron. Also, neural control can be divided into General Learning Architecture, Specialized Learning Architecture, Forward and Inverse Modeling Architecture and Feedback-error Learning Architecture. But, Forward and Inverse Modeling Architecture approach including two neural networks needs two learning laws and this becomes complicated. And Feedback-error Learning Architecture has a fixed feedback system, but its neural network does not have a feedback architecture and it becomes a open loop control system.

In this study, we propose a neural control system which improves the above two problems. That is, ① Using only one neural network and ② Giving a feed back function to the neural netework.

Next, it is said that the control for a short period approximated system causes the deterioration of control performance. Considering a current situation that the requirements to the control performance of aircrafts, which need high performance and can fly wide range, will be strict, an aircraft motion[5)] must be dealed as a nonlinear system. Also, as a result of constructing control system considering this nonlinear system as the plant, it is seemed that the progress of control performance is expected much. Then, we will give a design method of a flight control system based on our proposed neural control learning law and show the feasibility of our approach by numerical simulations.

In the second paragraph, the basic theory about neural network is given.

In the third paragraph, a flight control system based on the new neural control system is proposed. First, the formulation of problem is described. Next a learning flight control system with neural control is constructed.

At the end of paper, a nonlinear equation of aircraft motion is given, and the feasibility of proposed learning flight control system with numerical simulations using parameters of a high speed and small sized aircraft is shown.

2. Back Propagation Algorithm[4)]

In this paragraph, the basic theory about newral network is described.

2.1 Basic Perceptron[4)]

The baic perceptron can be drawed as Fig.1.

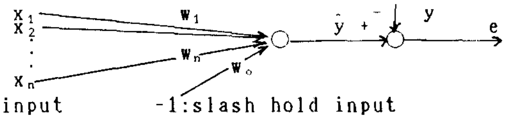

Fig.1 Structure of Basic Perceptron

where,

$x_i \in R$: input signal, $\bar{y} \in R$: output signal
$w_i \in R$: synaptic weight, $y \in R$: model signal

$$z \overset{def}{:=} \sum_{i=1}^{n} w_i x_i - w_0 = W^T X$$
$$= (-w_0 \ w_1 \ \cdots \ w_n)(1 \ x_1 \ x_2 \ \cdots \ x_n)^T$$

$\bar{y} = f(z)$: output function

Generally, the following functions can be considered as output function.

$$f(z)\begin{cases} \bar{y} = f(z) = z & : \text{Identity Function} \\ \bar{y} = 1/[1 + \exp(-z)] & : \text{Sigmoid Function} \end{cases}$$

Defining the renewal value of synaptic weight, ΔW, the following delta rule can be considered as a learning law.

$$\Delta W = -\varepsilon(\bar{y} - y)X$$

However, the application of this basic perceptron is limited. Then, generally the famous three layered neuralnetwork which is described in the next section is often used.

2.2 Three Layerd Perceptron[4)] and Generalized Delta Rule

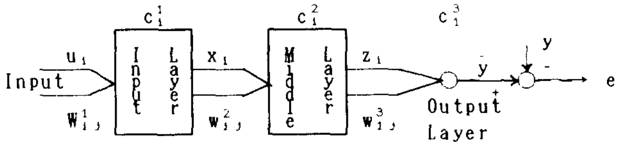

FIG.2 Three Layered Perceptron

Each signal in this figure is defined as follows.

$$\bar{x}_j = \sum_{c_i^1} w_{ji}^1 u_i \qquad \bar{z}_j = \sum_{c_j^2} w_{ji}^2 x_i \qquad \bar{y} = \sum_{c_i^3} w_{1i}^3 z_i$$

$$\bar{x}_j = f(x_j) \qquad \bar{z}_j = f(z_j) \qquad \hat{y} = f(\bar{y})$$
$$i = 1, 2, \cdots, k1 \quad i = 1, 2, \cdots, k2$$
$$j = 1, 2, \cdots, k2 \quad j = 1, 2, \cdots, k3$$

$e = \hat{y} - y$ (y : model signal)

c_i^j : input pattern of i-th output of j-th layer

The above neural network is defined as follows.

$$\hat{y} = ANN[x_i, z_i, w^1_{ij}, w^2_{ij}, w^3_{ij}](u_i)$$

Also,

$$\dot{y} = ANN[X, Z, W^1, W^2, W^3](U)$$

Where,

$$U^T = (u_1, u_2, \cdots, u_{k1})$$
$$X^T = (x_1, x_2, \cdots, x_{k2})$$
$$Z^T = (z_1, z_2, \cdots, z_{k3})$$

$$W^1 = \begin{bmatrix} w^1_{1,1} & w^1_{1,2} & \cdots & w^1_{1,k1} \\ \vdots & & & \vdots \\ w^1_{k2,1} & w^1_{k2,2} & \cdots & w^1_{k2,k1} \end{bmatrix}$$

$$W^2 = \begin{bmatrix} w^2_{1,1} & w^2_{1,2} & \cdots & w^2_{1,k2} \\ \vdots & & & \vdots \\ w^2_{k3,1} & w^2_{k3,2} & \cdots & w^2_{k3,k2} \end{bmatrix}$$

$$W^3 = (w^3_{11}, w^3_{12}, \cdots, w^3_{1,k3})$$

Now, defining the learning performance function E as following,

$$E \overset{def}{:=} (1/2)e^2$$

The learning law for the synaptic weight w^k_{ji} can be described as follows.

Learning Law: Generalized Delta Rule

$$\Delta w^k_{ji} = -\varepsilon \frac{\partial E}{\partial W^k_{ji}} \quad (\varepsilon > 0)$$

$$\frac{\partial E}{\partial W^3_{1i}} = \frac{\partial E}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \bar{y}} \cdot \frac{\partial \bar{y}}{\partial W^3_{1i}} = ef'(\bar{y})z_i$$

$$\frac{\partial E}{\partial W^2_{ji}} = \frac{\partial E}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \bar{y}} \cdot \frac{\partial \bar{y}}{\partial W^2_{ji}}$$

$$\frac{\partial \bar{y}}{\partial W^2_{ji}} = \sum_{c_i^3} \frac{\partial (W^3_{1i} z_i)}{\partial W^2_{ji}} = \sum_{c_i^3} W^3_{1i} \frac{\partial z_i}{\partial \bar{z}_i} \cdot \frac{\partial \bar{z}_i}{\partial W^2_{ji}}$$

$$\frac{\partial E}{\partial W^1_{ji}} = \frac{\partial E}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \bar{y}} \cdot \frac{\partial \bar{y}}{\partial W^1_{ji}}$$

$$\frac{\partial \bar{y}}{\partial W^1_{ji}} = \sum_{c_i^3} W^3_{1i} \frac{\partial z_i}{\partial \bar{z}_i} \cdot \frac{\partial \bar{z}_i}{\partial W^1_{ji}}$$

$$\frac{\partial \bar{z}_i}{\partial W^1_{ji}} = \sum_{c_i^2} W^2_{ij} \frac{\partial x_j}{\partial \bar{x}_j} \cdot \frac{\partial \bar{x}_j}{\partial W^1_{ji}}$$

3. Construction of Learning Flight Control System via Input Matching

In this paragraph, to get the easy understanding for the proposed method, first the learning flight control system to the linear system, next to the nonlinear system are given.

3.1 Formulation of Problem

Considering the next nonlinear system.

$$x(k+1) = f(x(k), u(k), k), \quad x(0) = x_0 \in R^n$$
$$y(k) = g(x(k), u(k), k) \qquad (1)$$

Where, the output variable $y(k) \in R$ and the input variable $u(k) \in R$ are observed signals. And $x(k) \in R^n$ is the state variable, the nonlinear elements have the following dimensions.

$$f(x(k), u(k), k) : R^n \times R \times R \rightarrow R^n$$
$$g(x(k), u(k), k) : R^n \times R \times R \rightarrow R$$

Also, using the shift operator q^{-1}, Eq.(1) can be expressed as following. (2)

$$P(q^{-1})y(k+d) = R(q^{-1})u(k) + f'(u(k), y(k), k)$$

The polynomials $P(q^{-1})$, $R(q^{-1})$ and the nonlinear element $f'(u(k), y(k), k)$ are defined as follows.

$$P(q^{-1}) \overset{def}{:=} 1 + p_1 q^{-1} + p_2 q^{-2} + \cdots + p_n q^{-n}$$
$$R(q^{-1}) \overset{def}{:=} r_0 + r_1 q^{-1} + \cdots + r_m q^{-m} \quad (r_0 \neq 0) \quad (n \geqq m)$$
$$f'(u(k), y(k), k) : R \times R \times R \rightarrow R$$

On the other hand, the reference model is given as following.

$$P_m(q^{-1})y_m(k+d) = R_m(q^{-1})u_m(k) \qquad (3)$$

Where, $u_m(k) \in R$ is the bounded reference input, $y_m(k+d) \in R$ is the bounded reference output, $P_m(q^{-1})$ and $R_m(q^{-1})$ are the arbitrary stable polynomials.

The objective of this study is to develop the learning flight control system which force the output signal $y(k)$ to track the reference model output signal $y_m(k)$ asymtotically with only the observable signals of the nonlinear system.

$$\lim_{k \to \infty} y(k) \rightarrow y_m(k) \qquad \text{for } k \rightarrow \infty$$

3.2 In the case of Linear System

First, Considering the plant as the linear system which is the nominal part of the nonlinear system Eq.(2).

$$P(q^{-1})y(k+d) = R(q^{-1})u(k) \qquad (4)$$

<Subtheory 1>

The following equation can be concluded for the arbitorary stable polynomials $Q(q^{-1})$ and $D(q^{-1})$.

$$P(q^{-1}) \text{ and } R(q^{-1}) \text{ are coprime} \rightarrow$$
$$\exists A(q^{-1}),\ \exists B(q^{-1}) \text{ such that} \qquad (5)$$
$$A(q^{-1})q^{-1}P(q^{-1}) + B(q^{-1})q^{-d-1}R(q^{-1}) = Q(q^{-1})[P(q^{-1}) - k_p D(q^{-1})R(q^{-1})]$$
$$\text{for } \forall Q(q^{-1}),\ \forall D(q^{-1}) : \text{stable polynomials}$$

Where,

$$Q(q^{-1}) \overset{def}{:=} 1 + q_1 q^{-1} + q_2 q^{-2} + \cdots + q_n q^{-n}$$
$$D(q^{-1}) \overset{def}{:=} 1 + d_1 q^{-1} + d_2 q^{-2} + \cdots + d_{n-m} q^{-n+m}$$
$$A(q^{-1}) \overset{def}{:=} a_0 + a_1 q^{-1} + \cdots + a_{n-1} q^{-n+1}$$
$$B(q^{-1}) \overset{def}{:=} b_0 + b_1 q^{-1} + \cdots + b_{n-1} q^{-n+1}$$
$$k_p \neq 0$$

Generally the identity equation[6] Eq.(5) can be conculded in the case of $Q(q^{-1}) = 1$ and $D(q^{-1}) = 1$. Therefore, for the easy explanation of following theory, $Q(q^{-1})$ and $D(q^{-1})$ can be set to 1 respectively.

<Theory 1>

If the polynomials of the plant $P(q^{-1})$ and $R(q^{-1})$ are known for the plant Eq.(4) and the reference model Eq.(3), and defining the control input $u(k)$ as following equation, the control objective can be obtained. (6)

$$u(k) = A(q^{-1})u(k-1) + B(q^{-1})y(k-1) + k_p y_m(k+d)$$

(Proof) Using the partial state of the plant $Z(k)$, Eq.(4) can be expressed as follows.

$$P(q^{-1})Z(k) = u(k)$$
$$R(q^{-1})Z(k) = y(k+d) \qquad (7)$$

Also using the above equation and the identity equation Eq.(5), the following nonminimal expression of the linear system can be obtained. (8)

$$u(k) = A(q^{-1})u(k-1) + B(q^{-1})y(k-1) + k_p y(k+d)$$

Here, defining the output error as following equation.

$$e(k) \overset{def}{:=} y_m(k) - y(k)$$

Also considering Eq.(6) and Eq.(8), the following output error equation can be obtained. It is easily understood that the contorl objective can be achieved.

$$k_p e(k) = 0 \qquad \nabla\nabla\nabla$$

Now, the dynamics of the linear system $P(q^{-1})$ and $R(q^{-1})$ are unknown, the control polynomials $A(q^{-1})$ and $B(q^{-1})$ are unknown and can not be decided in advance.

Then, the structures of $A(q^{-1})$ and $B(q^{-1})$ are known, we show a design method of learning flight control system using artificial neural network based on the nonminimal expression of system Eq. (8).

The nonminimal expression[7] Eq. (8) can be expressed as follows.

$$\hat{y}(k) \overset{def}{:=} y(k) = W_1^T X_1(k) \tag{9}$$

Also,

$$\hat{y}(k) \overset{def}{:=} \hat{u}(k-d) = W^T X(k) \tag{10}$$

Where,

$$W_1^T \overset{def}{:=} -(a_0, a_1, \cdots, a_{n-1}, b_0, b_1, \cdots, b_{n-1}, -1)(1/k_p)$$

$$X_1^T \overset{def}{:=} (u(k-d-1), \cdots, u(k-d-n), y(k-d-1), \cdots, y(k-d-n), u(k-d))$$

$$W^T \overset{def}{:=} (a_0, a_1, \cdots, a_{n-1}, b_0, b_1, \cdots, b_{n-1}, k_p)$$

$$X^T \overset{def}{:=} (u(k-d-1), \cdots, u(k-d-n), y(k-d-1), \cdots, y(k-d-n), y(k)) = (\ U_1^T(k), U_2(k)\ )$$

$$U_1^T(k) \overset{def}{:=} (u(k-d-1), \cdots, u(k-d-n), y(k-d-1), \cdots, y(k-d-n))$$

$$U_2(k) = y(k)$$

Each learning flight control systems based on Eq. (9) and Eq. (10) can be developed equally, in this study considering the easy develop of the control input, we show a learning flight control system based on the input matching using Eq. (10).

First, defining the neural network as follows.

$$\hat{y}(k) = ANN[X, Z, W^1, W^2, W^3](U_1(k)) + ANN[W^4](U_2(k))$$

Where, $U_1 \in R^{k4}, U_2 \in R,\ X \in R^{k4},\ Z \in R^{K5}$ (11)

$W^1 \in R^{k4\times1},\ W^2 \in R^{K5\times k4},\ W^3 \in R^{1\times k5},\ W^4 \in R$

Also, defining the learing performance function E as follows,

$$E = (1/2)e_{\hat{u}}^2(k),\quad e_u(k) \overset{def}{:=} \hat{u}(k-d) - u(k-d) \tag{12}$$

the following theory about the leaning flight control system can be obtained.

<Theory 2>

For the plant Eq. (4), the reference model Eq. (3), the neural network Eq. (11) and the performance function Eq. (12), if the learning law and the learning control law are developed as follows, the control objective can be achieved.

Learning law: (13)

$$\begin{cases} W^j(k+1) = W^j(k) + \Delta W^j(k+1),\ j = 1, 2, 3, 4 \\ \Delta W^j(k+1) = -\varepsilon\ [\partial E/\partial W^j(k)] \end{cases}$$

Learning Control Law: (14)

$$u(k) = ANN[X, Z, W^1, W^2, W^3](U_1(k+d)) + ANN[W^4](y_m(k+d))$$

(Proof) Eq. (14) which is constructed by each elements of $U_1(k+d)$ in Eq. (10) and the reference model Eq. (3) is causal.

Using the learning law Eq. (13), the following relation can be obatained by the convergent theory.

$$\hat{u}(k) \rightarrow u(k) \qquad \text{for } k \rightarrow \infty$$

$$W^j \rightarrow W^{j*}\text{: a constant value} \qquad \text{for } k \rightarrow \infty$$

Using the above relation, Eq. (11) and Eq. (14) become as follows.

$$\hat{u}(k) = ANN[X, Z, W^{1*}, W^{2*}, W^{3*}](U_1(k+d)) + ANN[W^{4*}](y(k+d))$$

$$u(k) = ANN[X, Z, W^{1*}, W^{2*}, W^{3*}](U_1(k+d)) + ANN[W^{4*}](y_m(k+d))$$

Also, using above equations, the following equation can be obtained.

$$y(k) \rightarrow y_m(k) \qquad \text{for } k \rightarrow \infty$$

Then, the control obajective can be achieved. $\nabla\nabla\nabla$

3.3 In the case of Nonlinear System

In this case, first of all, it is assumed that the nonlinear term of Eq. (2) $f'(u(k), y(k), k)$ can be expressed by the three layered artificial neural network which has the time delay signals for the input-output signal of the system $[u(k), y(k)]$ as input. Generally, this assumption can be satisfied by increasing the number of neuron in the middle layer.

Using the above assumption and the design method of the learning flight control system for the linear system in the above section, the neural network by input matching can be defined as follows. (15)

$$\hat{y}(k) = ANN[X, Z, W^5, W^6, W^7](U_3(k)) + ANN[W^8](U_2(k))$$

Where, $U_3 \in R^{k6}, U_2 \in R,\ X \in R^{k6},\ Z \in R^{K7}$

$W^5 \in R^{k6\times1},\ W^6 \in R^{K7\times k6},\ W^7 \in R^{1\times k7},\ W^8 \in R$

$$U_3^T(k) \overset{def}{:=} (u(k-d-1), \cdots, u(k-d-k_6/2), y(k-d-1), \cdots, y(k-d-k_6/2))$$

$$U_2(k) = y(k)$$

The learning performance function and the learning law can be developed as same as Eq. (12) and Eq. (13) of the above section.

Also, if the learning control input can be developed as follows, the control objective can be achieved.

$$u(k) = ANN[X, Z, W^5, W^6, W^7](U_3(k+d)) + ANN[W^8](y_m(k+d)) \tag{16}$$

The block diagram of this learning control system is drawed as following figure.

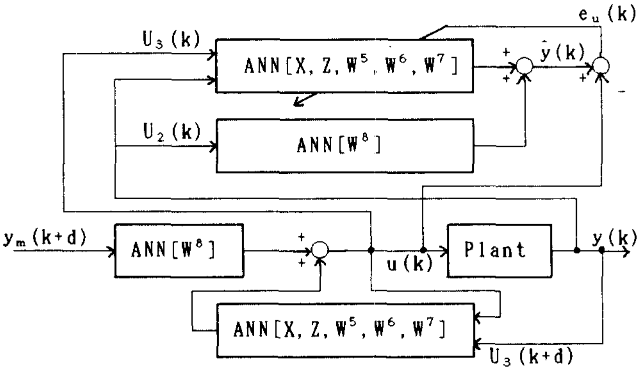

Fig. 3 Learning Control System via Input Matching

4. Application to Small-size High Speed aircraft

4.1 Nonlinear Equation of Aircraft Motion

Recently, most of airplanes can fly over a wide drange with its improved performance. Moreover such airplains require to fly at a nonlinear range, for example, at a range of more high angle of attack by thrust vector control and flying by discontinuity, etc..

Then, in this study the single input and single output system using the elevator angle as input and the pitch angle as output is considered.

The nonlinear longitudinal equations of aircraft motion can be described as follows.

$$\dot{V} = (\rho S/2m)V_T^2(C_x + C_{xdh}\Delta_h) - g\sin\Theta - WQ + T/m$$

$$\dot{W} = (\rho S/2m)V_T^2(C_z + C_{zdh}\Delta_h) + g\cos\Theta + VQ$$

$$\dot{Q} = (\rho S\bar{C}/2I_y)V_T^2(C_m + C_{mdh}\Delta_h) + (\rho S\bar{C}^2/4I_y)V_T C_{mq}Q$$

$$\dot{\Theta} = Q \qquad (16)$$

Where,

V : forward velocity(m/s), Q : pitch rate(rad/s)
W : vertical velocity(m/s), Θ : pitch angle(rad)
Δ_h : elevator angle(rad), T : thrust(N)
m : mass(kg), ρ : air density(kg/m³)
I_y : inertial moment around y axis(kg·m²)
S : area of the main plane(m²)
$\bar{C}$: mean aerodynamic wing span of main plane(m)
g : gravity force(m/s)
V_T : air speed(m/s): $V_T^2 = V^2 + W^2$
A : angle of attack(rad): $A = \tan^{-1}(W/U)$

C_x, C_z, C_m, C_{mq}, C_{xdh}, C_{zdh} and C_{mdh} are the nondimensional aerodynamic derivatives, etc..

On the other hand, the reference model is selected as following first order system.

$$\Theta_m = a/(s+a)\Theta_r$$

Where $\Theta_m \in R$ and $\Theta_r \in R$ are the bounded reference output and input.

4.2 Numerical Simulation

In this simulation, the single input and single output system with the elevator angle as input and the pitch angle as output is consiered. First the values of Appendix are used to system Eq.(9), Next the values of the reference model are selected as, $a = 0.5$, $\Theta_r = -10$ and $+30$ deg. Also the initial values of both input are 0.09483 deg.

The neural network has 6 neurons in input layer, 12 neurons in middle layer and 1 neuron in output layer respectively. Also, sigmoid functions are used for input layer and middle layer, identity function is used for output layer. Moreover the weight of learning law is $\varepsilon = 0.03$ and the initial value of synaptic weight is 0.1.

The simulation results are shown in Fig.1～4.

Fig.1 shows that pitch angle as system output tracks the model output -10 deg at the begging and 30 deg after 20 sec.

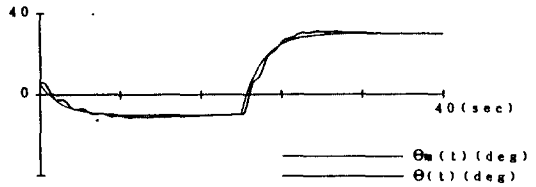

FIG.1 The Response of Output Θ (pitch angle)

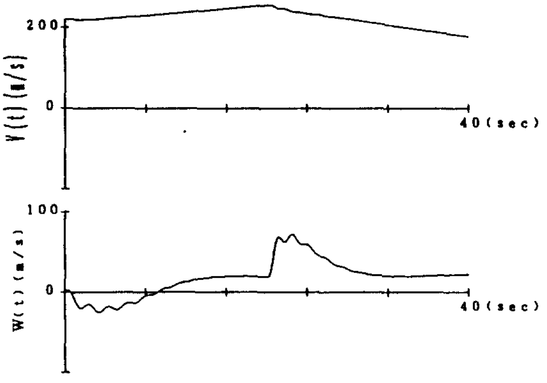

FIG.2 The Responses of V (forward velocity) and W (vertical velocity)

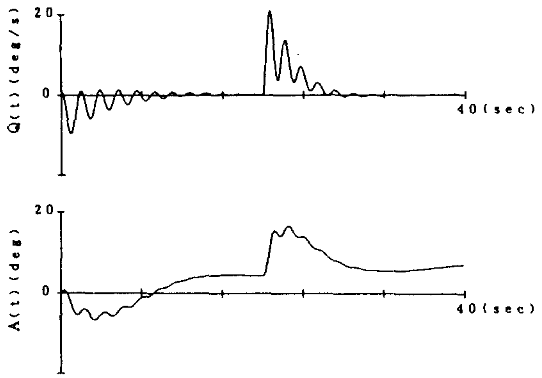

FIG.3 The Response of Q (pitch rate) and A (angle of attack)

FIG.4 The response of Δ_h (elevator angle)

6. Conclusions

We propose a design method of learning flight control system via input matching. The proposed method has two benefit, that is, ① Using only one neural network and ② Giving a feed back function to the neural netework. And we show the feasibility of the proposed method with numerical simulations.

References

1) Kumpati, S.Narendra:Adaptive Control Using Neural Networks, Neural Networks for Control, A Bradford Book, The MIT Press,115/142(1990)
2) Omatu: Neural Network and Adaptive Control, Computrol No. 32, Colona Co.,(1990)
3) Kanai, Ochi, Kato: Design of Flight Cotnrol System using Nerual Network, The proceeding of 29th Aircraft Symposium, 294/297, (1991)
4) Nakano: The Basis of Neuro Computer, Colona Co., (1990)
5) Kanai,K: Flight Control, Maki Publishing Co., Tokyo, (1985)
6) W. A. Wolovich: Linear Multivariable Systems, Springer-Verlag, (1974)

Appendix

Data of the flight condition of a small sized and high speed aircraft flying at altitude 7,600m

U_o = 220(m/s), m = 22695(kg)
I_y = 427348(kg·m²), ρ = 0.5495(kg/m³)
S = 49.77(m²), $\bar{C}$ = 2.76(m)
$C_x = 2.429A^2 - 0.1959A - 0.03$
$C_z = -11.16A^2 - 1.776A - 0.08$
$C_m = -0.7221A^2 - 1.083A + 0.063$
$C_{mq} = -25.61A^2 + 23.09A - 26.04$
C_{xdh} = 0.2464(1/rad), C_{xzh} = 0.8480(1/rad)
C_{mdh} = 1.719(1/rad)