JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 29, No. 2, March–April 2006

Direct Adaptive Neural Flight Controller for F-8 Fighter Aircraft

S. Suresh,* S. N. Omkar,† and V. Mani‡
Indian Institute of Science, Bangalore 560 012, India
and
N. Sundararajan§
Nanyang Technological University, Singapore 639798, Republic of Singapore

A novel neural network approach based on model-following direct adaptive control system design is proposed to improve damping and also to follow pilot commands accurately. The control law is derived using system theory and an implicit function theorem. A neural network with a linear filter is used to approximate the control law. The neural controller is trained using offline (finite interval of time) and online learning strategies. The neural controller was trained offline using the error signal between the aircraft response and pilot command (reference model). The offline-trained neural controller provides the necessary damping and tracking performances. The neural controller is adapted online for variations in aerodynamic coefficients or control surface deficiencies caused by control surface damage. A discrete time linear dynamic longitudinal model of a high-performance aircraft (F-8) is considered to demonstrate the effectiveness of the proposed control scheme. The performance results of the proposed control scheme are compared with the recently developed dynamic inversion technique and fully tuned radial basis function network. The neural controller performance is also evaluated for steady climb-and-hold maneuver in a nonlinear six-degree-of-freedom aircraft model.

I. Introduction

ADAPTIVE identification and control of nonlinear dynamic systems with parameter uncertainties has been an active area of research for the last three decades. The fundamental issues of adaptive control for linear systems such as selection of controller architecture, assumptions on a priori system knowledge, parameterization of adaptive systems, establishment of error models, adaptive law, and analysis of stability have been extensively addressed.[1−3] However, most practical systems are nonlinear in nature. Adaptive control of such nonlinear systems is, therefore, an intense area of research. Novel techniques in adaptive control of nonlinear systems are facilitated through advances in geometric nonlinear control theory and, in particular, feedback linearization methods[2] and backstepping methods.[3] A key assumption in these methods is that the system nonlinearities are known a priori. Most of the work in the area of nonlinear adaptive control deals with systems where the uncertainty is due to unknown parameters that appear linearly with respect to known nonlinearities.

During the last decade, a large amount of research work has been carried out in neural control theory, almost independently from adaptive nonlinear control research.[4,5] A complete survey on adaptive control system design using neural network is presented in Ref. 6. The neural network paradigm emerged as a powerful tool for learning complex mappings from a set of examples. This has generated a great deal of interest in using neural network models for identification and control of dynamic systems with unknown nonlinearities.[6,7] In these studies, neural networks are mostly used as approximate models for unknown nonlinearities, thus removing the need for a priori knowledge of system nonlinearities. The approximating capabilities and inherent adaptive features of neural networks are appealing for modeling and control of nonlinear dynamic systems. Furthermore, from a practical perspective, the massive parallelism and fast adaptability of neural network implementations provide more incentive for further investigation in problems involving dynamic systems with unknown nonlinearities.

The neural networks in identification and control of nonlinear dynamic systems are studied extensively in Refs. 6–10. The neural adaptive control problem was formulated first in Ref. 8, which instigated further research in this area. For most part, these studies are based on first replacing unknown functions in a difference equation by static neural networks and then, according to a quadratic cost function, deriving update laws using optimization methods. A particular optimization procedure that has attracted considerable attention is the steepest descent, or gradient method, which leads to the static backpropagation or dynamic backpropagation algorithms.[11−13] Among the various other neural network paradigms, radial basis function networks[14−16] and recurrent networks[12] are used for identification and control of dynamic systems.

Adaptive controllers based on different neural network architectures are widely used in flight control systems,[15−23] robotics,[24,25] missile control systems,[26] and helicopter control.[27] A detailed survey of the use of neural networks for nonlinear flight control is presented in Ref. 28. In Refs. 16–18, the online learning ability of a neural network is demonstrated for modeling uncertainties in aircraft dynamics. However, most of the applications using neural network as control architecture use a conventional controller in the inner loop to stabilize the system dynamics, whereas the neural controller acts as an aid to the conventional controller by compensating for the nonlinearity. The necessary bounded signal requirement for neural network training is satisfied using an inner conventional controller. Under the assumption that the inner conventional controller provides necessary bounded signals in fault and nonnominal conditions, the neural network is further adapted to provide the required tracking performance. The adaptation of a neural controller alone under fault and nonnominal conditions leads to excessive control effort. To overcome excessive control effort, researchers have attempted to use reconfigurable flight controllers to stabilize the system and improve tracking performance.[29−31]

Received 2 July 2004; revision received 14 February 2005; accepted for publication 25 February 2005.

*Research Scholar, Department of Aerospace Engineering. Member AIAA.
†Senior Scientific Officer, Department of Aerospace Engineering.
‡Professor, Department of Aerospace Engineering.
§Professor, School of Electrical and Electronic Engineering. Associate Fellow AIAA.

Recently, in Ref. 29, a reconfigurable flight controller for unstable aircraft based on an offline (finite interval of time) and online training procedure was proposed. The offline (finite interval of time) training scheme ensures the stability requirement (bounded signal requirement) using a mild assumption on the system. (In finite time, the aircraft does not escape to infinity.) The flight controller is adapted using the input–output data generated between $[0, T]$. In Ref. 29, an indirect adaptive neural controller scheme with an offline and online training procedure is used to stabilize a linear dynamic model of a research aircraft and also to improve the tracking performance. In an indirect adaptive control scheme, the convergence of neural controller depends on the accuracy of the identification model. Under nonnominal and fault conditions, the identifier and controller network are adapted online. If the identification model is not accurate, then the error estimation at the controller network will not be accurate. This will affect the controller performance. Similarly, under plant uncertainties, the rate at which the identifier and controller network are adapted will affect the convergence. A direct adaptive control scheme can be used to overcome the aforementioned drawbacks.

In this paper, a model-reference direct adaptive neural control design procedure using an offline (finite time interval) and online training scheme is presented. The proposed learning scheme is based on the assumption that the states and outputs of the system do not escape to infinity in the finite time interval. By the use of the preceding assumption, a neural network with a linear filter is trained offline using backpropagation through a time learning algorithm to approximate the unknown control law. The offline-trained neural network is used as starting point for online adaptation under variations in aerodynamic coefficients or control effectiveness deficiencies caused by control surface damage.

The performance of the proposed control scheme, along with the direct adaptive control law, is evaluated for a flight controller design based on an F-8 longitudinal aircraft model as described in Ref. 18. To investigate the online learning ability of the proposed neural controller, different fault scenarios representing large model error and control surface loss are considered. The performance of the proposed scheme is compared with the dynamic inversion technique[18] and a fully tuned radial basis function network (RBFN).[15] The results demonstrate that the proposed scheme has better tracking ability than the dynamic inversion method[18] and RBFN[15] for a linear system. To study the effectiveness of the proposed control scheme for nonlinear systems, a steady climb-and-hold maneuver in a nonlinear six-degree-of-freedom model is also evaluated.

The paper is organized as follows: Sec. II gives a brief description of the statement of the problem, and in Sec. III we derive the direct adaptive neural controller. The aircraft controller design is presented in Sec. IV. The simulation results and comparative study are presented in Sec. V. Section VI provides a summary of the conclusions from this study.

II. Problem Formulation

The aircraft dynamics is represented in discrete time framework as,

$$\sum : \boldsymbol{x}(k+1) = f(\boldsymbol{x}, \boldsymbol{u}, k), \qquad \boldsymbol{y}(k) = g(\boldsymbol{x}, \boldsymbol{u}, k) \quad (1)$$

where the functions $f()$ and $g()$ are smooth and continuous functions, $\boldsymbol{x}$ is the state vector in $\Re^n$, u is the control input to the system in $\Re^p$, and $\boldsymbol{y}$ is the output vector in $\Re^m$. The control input u is bounded, that is, $u_i := \{\|u_i(t)\| < \delta_i,\ \forall\, u_i \in U_i\}$, where $\delta_i \in \Re^+$, $i = 1, 2, \ldots, p$. Without loss of generality, let us assume that the values of the functions and the inputs at equilibrium states are $f(0, 0) = 0$ and $g(0, 0) = 0$ at $x(0) = 0$ and $u(0) = 0$.

The problem here is to design a neural flight controller (NFC) such that the aircraft response y follows the arbitrary bounded reference signal (pilot command) y_d accurately; that is, determine the control input $u(k)\,\forall\,, k > 0$ such that

$$\lim_{k\to\infty} |y_d(k) - y(k)| \leq \epsilon \quad (2)$$

for some small specified constant $\epsilon \geq 0$. Let us assume that the functions $f()$ and $g()$ have bounded partial derivatives in a certain neighborhood of all of the points along the reference trajectories y_d and that their derivatives with respect to u are nonsingular. It follows from the implicit function theorem that the control input $u^*(k)$ can be expressed in terms of past inputs and outputs of the system[32]:

$$u^*(k) = \bar{g}[u(k-1), \ldots, u(k-n), y(k-1), \ldots, y(k-n), y_d(k+1)] \quad (3)$$

where $\bar{g}()$ is a smooth continuous function and n is the delay. The reference input (pilot stick deflection) r to desired responses y_d belongs to a class of dynamic systems $\sum$ and is expressed as

$$x_d(k+1) = A_d x_d(k) + B_d r(k), \qquad y_d(k) = C_d x_d(k) \quad (4)$$

where A_d, B_d, and C_d are the system matrices. This system matrices are calculated based on the flying quality requirements.[33]

The reference model given in Eq. (4) is controllable and observable. Then the reference output can be expressed using nonlinear autoregressive moving average model[8] as

$$y_d(k+1) = h[r(k), \ldots, r(k-n_1), y_d(k-1), \ldots, y_d(k-n_1)] \quad (5)$$

where $h()$ is a smooth and continuous function and n_1 is the order of the system.

By substituting Eq. (5) into Eq. (3), we will get the desired function to estimate the control input u^* to follow the reference command signal,

$$u^*(k) = \bar{g}[u(k-1), \ldots, u(k-n), y(k-1), \ldots, y(k-n), h(., .)] \quad (6)$$

If the aircraft follows the reference command exactly, then the signal from $u(k-1)$ to $u(k-n)$ can be expressed in terms of pilot stick deflections and aircraft responses,

$$u^*(k) = \bar{g_1}[r(k), \ldots, r(k-n_2), y(k-1), \ldots, y(k-n_2)] \quad (7)$$

where $\bar{g_1}()$ is a smooth unknown function and n_2 is the delay. The preceding controller equation (7) is unique in the neighborhood of the equilibrium state defined earlier. Note that the algebraic relationship given in Eq. (7) is valid for any bounded reference output sequence y_d and arbitrary values of inputs and outputs in the domain in which the system is controllable and observable.[32] If the function $\bar{g_1}(., .)$ is known, then the neural controller weights can be updated directly using deviation between the network output and the algebraic solution of Eq. (7). Because the function $\bar{g_1}(., .)$ is not known exactly, it is difficult to estimate the control input u^* for given reference pilot command. Hence, the function $\bar{g_1}(., .)$, given in Eq. (7), is approximated using a direct adaptive neural control scheme with an update law based on the deviation between aircraft response and desired pilot command. The details of controller architecture and the update law are described in the next section.

III. Neural Flight Controller

In this section, we briefly describe the neural network architecture required to approximate the control law given in Eq. (7) and the corresponding learning algorithm.

A. Neural Network Architecture

Because the function $\bar{g_1}()$ is not known exactly, it is difficult to estimate the control input u^* for a given reference command. Hence, a direct adaptive control scheme is used to approximate the function with a learning law based on the tracking error $e_c(k)$. The objective of the direct adaptive control system is to find the control input u^* such that the aircraft response follows the reference output in some sense. Let us assume that the number of signals to be tracked is equal to the number of control variables. The block diagram for the structure of a direct adaptive neural controller is shown in the Fig. 1.

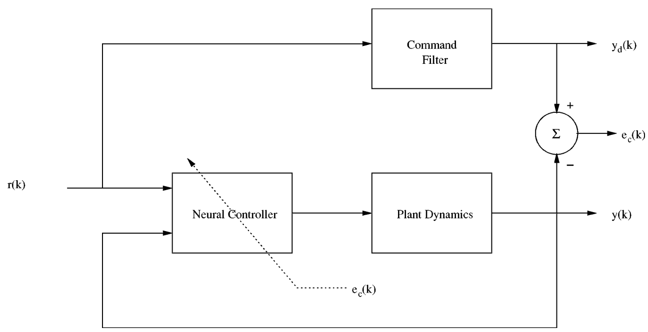

Fig. 1 Direct adaptive control block diagram.

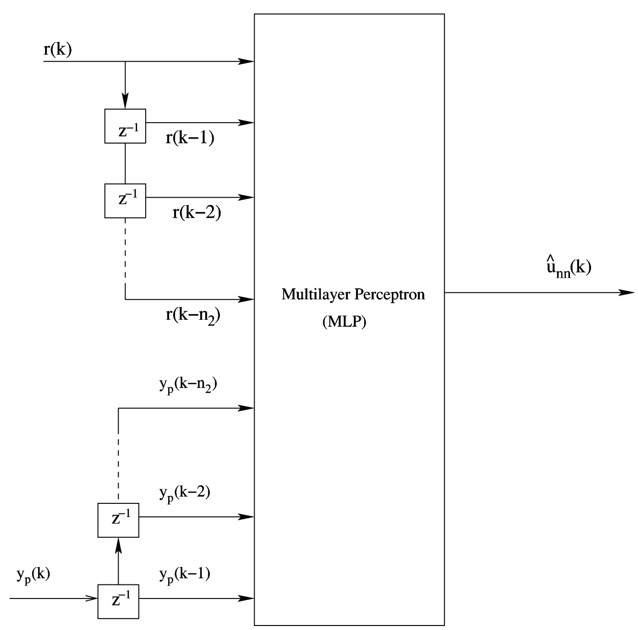

Fig. 2 Neural network architecture.

The tracking error $e_c(k)$ is used to adjust the network parameters along the negative gradient of the performance index J,

$$J(k) = \tfrac{1}{2} e_c(k)^2 \tag{8}$$

where

$$e_c(k) = y_d(k) - y(k)$$

Let neural network N_c with a three-layer network structure $N_c^{m,h,p}$ (m number of input neurons, h number of hidden neurons, and p number of output neurons) and a linear filter be used to approximate the control law given in Eq. (7). The architecture of the network is given in Fig. 2, with input $Z \in \Re^m$ and output $u \in \Re^p$,

$$u_i = \sum_{j=1}^{h} \left[w_{ij} \sigma \left(\sum_{k=1}^{m} v_{jk} z_k \right) \right], \qquad i = 1, 2, \ldots, p \tag{9}$$

with $\sigma()$ being the activation function, v_{jk} the first-to-second layer interconnection weights, and w_{ij} the second-to-third layer interconnection weights. From Eq. (7), we can observe that the input to the neural network N_c is past n_2 reference inputs and aircraft responses and the present reference input

$$Z = [r(k), r(k-1), \ldots, r(k-n_2), y_p(k-1), \ldots, y_p(k-n_2)]$$

The neural network dynamics can be conveniently expressed in matrix form by defining $Z = [z_1, z_2, \ldots, z_m]$, $u = [u_1, u_2, \ldots, u_p]$, and weight matrices $W^T = [w_{ij}]$ and $V^T[v_{ij}]$. Let W^* and V^* be the optimal weights, then

$$u_{nn}^* = (W^*)^T \sigma[(V^*)^T Z] \tag{10}$$

If the input Z and output u^* of the network belongs to a compact set, then there exists a sufficient number of hidden neurons h and optimal weight Θ^* ($\Theta^* = [W^* \;\; V^*]^T$) such that $e_c(k) = 0, \forall, k > 0$. The optimal weight vector Θ^* belongs to convex compact set $B(\Theta)$. Let us assume that the norm of the optimal neural network weights is bounded by a known positive constant Δ. By the use of the preceding assumption for all inputs $Z \in \Re^m$, the optimal neural network weight is defined as

$$\Theta^* := \arg \min_{\Theta \in B(\Theta)} \left\{ \sup_{Z \in \Re^m} \left\| \bar{g}_1(.,) - u_{nn}^* \right\| \right\}$$

The preceding equation is valid only when the input to the neural network belongs to compact set ($Z \in \Re^m$). Hence, we make a small assumption on the aircraft model.

Assumption: Given a class U of admissible (piecewise continuous and bounded) inputs, for any bounded u and any finite initial condition x_0, the state and output trajectories do not escape to infinity in finite time,

$$\sup_{t \in [0, T_c]} \|y(t)\| \leq \Delta \tag{11}$$

where Δ is a known real positive number. The time T_c is referred to as the critical time.

Based on the preceding assumption on the aircraft to be controlled, let us define a compact set S as

$$S := \left\{ (y, r) \in \Re^{m+p} : \|y(t) - y_0\| \leq \epsilon, u \in U \right\}, \qquad \forall t \in [0, T]$$

where ϵ is some positive constant and y_0 is initial output at time $t = 0$. Hence, the input Z to the network N_c belongs to a compact set inside the finite time interval $[0, T]$. By the universal approximation property of neural networks, it is possible to approximate any function to desired accuracy if the inputs and outputs belong to compact sets. Hence, the neural network weight vector will converge close to optimal value.[34–37] Now, we state the following definition.

Definition: A neural network is said to satisfy the universal approximation property if, on any compact subset $C \subset \Re^n$ of the input space, it is always possible to find an appropriate number of hidden neurons and optimal weight vectors (not necessarily unique), such that any continuous function can be approximated to any arbitrary level of accuracy.

Note that the preceding assumption and the definition proves the existence of the optimal network parameters. Let $\hat{W}$ and $\hat{V}$ be the estimated network weights. Then the neural flight controller response is given as

$$\hat{u}_{nn} = \hat{W}^T \sigma(\hat{V}^T Z) + \epsilon(Z) \tag{12}$$

and the error function $\epsilon(Z) < \delta$ (δ being a small positive number). The details on convergence of the static backpropagation learning algorithm is given in Ref. 36.

B. Learning Algorithm

Based on the preceding discussions, we propose an offline (finite time interval) and online training strategy for approximating the unknown control law using neural network architecture. First the neural network weights are updated using a backpropagation learning algorithm[8] using the data generated in a finite time interval. The offline-trained neural network weight matrices are further adapted online for modeling uncertainties and fault conditions.

In the proposed learning scheme, the network weights are adapted in the local direction of greatest error reduction such that aircraft follows the pilot command. Let w_{ij} be the weight connection between jth neuron in output layer and ith neuron in the hidden layer. Similarly, let v_{ij} be the weight connection between the jth neuron in hidden layer and the ith neuron in the input layer. The network weight update laws are given as follows:

$$\Delta w_{ij}(k) = \eta e_c^i(k)\sigma\left[\sum_{l=1}^{m} v_{jl}(k-1)z_l(k)\right]$$

$$i = 1, 2, \ldots, p, \qquad j = 1, 2, \ldots, h \quad (13)$$

$$\Delta v_{ij}(k) = \eta\delta_i z_j(k), \qquad i = 1, 2, \ldots, h, \qquad j = 1, 2, \ldots, m \quad (14)$$

where η is the learning rate and

$$\delta_i = \sigma'\left[\sum_{l=1}^{m} v_{il}(k-1)z_l(k)\right]\sum_{l=1}^{p} w_{li}(k-1)e_c^i(k)$$

where σ' is the first derivative of the activation function $\sigma()$.

IV. Design of Neural Flight Control System

The proposed neural controller design is used for longitudinal control of F-8 aircraft model as described in Ref. 18. The NFC is designed to follow the velocity and pitch rate command signals. The block diagram of the NFC is shown in Fig. 3. The performance of the NFC is studied under parameter uncertainty and control surface loss. To asses the simulation performances, the parameters of system matrix A and control matrix B are varied by 70%. The simulation results are compared with dynamic inversion in Ref. 18. RBFN and traditional RBFN are presented in Ref. 15. The following section describes the designing the neural longitudinal flight control system for an F-8 aircraft.

A. Aircraft Model

The aircraft is linearized about the equilibrium condition, and the aircraft response can be expressed in terms of linear equations as

$$\dot{x} = Ax + Bu, \qquad y = Cx + Du \quad (15)$$

where the signal x represents $x(t)$, and so on. For the purpose of illustration, the aircraft response is decoupled, and only the longitudinal response of the aircraft is considered for simulation study. Let A, B, C, and D be (4×4), (2×1), (4×4), and (2×1) matrices, respectively. The state vector comprises velocity v_0, angle of attack α, pitch rate q, and pitch attitude θ. The control inputs $\boldsymbol{u}$ include the elevator deflection δ_e and throttle position δ_t. Physical constraints on the elevator are defined as $-26.5 \le \delta_e \le 6.75$ deg. The elevator and throttle actuator are approximated using a first-order transfer function with time constants 0.1 and 0.2 s, respectively. The aircraft is trimmed at $v_0 = 198$ m/s, $\alpha_0 = 4.467$ deg, and altitude $h = 6096$ m. The system and output matrices are same as in Ref. 18. The roots of the characteristic equation are $-0.6656 \pm j2.82$ and $-0.0069 \pm j0.0765$. Though stable, the phugoid mode needs stability augmentation to improve its damping and settling time. The NFC is designed to improve the stability and tracking response of the system.

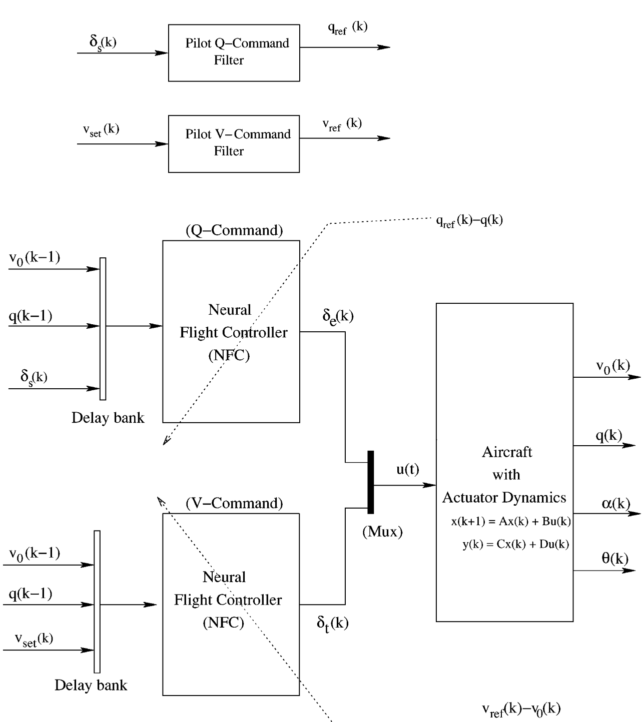

Fig. 3 Block diagram of NFC.

B. Command Generation

The pilot stick deflection (reference input) to the command signal transfer function is selected based on the flying quality requirement (level 1 handling quality). The NFC is designed to track the pilot desired velocity v_d and pitch rate q_d command signals. The desired response functions are given as follows[18]:

$$\frac{v_d}{v_{\text{set}}} = \frac{0.0400(s+3.13)}{s^2+0.6408s+0.1296}, \qquad \frac{q_d}{\delta_s} = \frac{0.6130(s+0.5)}{s^2+3.987s+5.018}$$

and $v_d/\delta_s = 0$ and $q_d/v_{\text{set}} = 0$. Pseudorandom pilot command signals with different amplitude are generated between 0 and 10 s for various initial conditions. The responses are sampled at 0.05, and these finite time data are used to train the NFC offline. The trained NFC is tested with a velocity step command of 15 ft/s, and a pitch rate command is generated using pulse pilot stick input (width 4 s and amplitude of 0.5 in.) as described in Ref. 18.

C. NFC Training

In this simulation, a study of two different neural controllers, one for velocity command v_d and other for pitch rate command q_d, are designed to follow the pilot commands. The optimal number of hidden nodes required to approximate the control law is selected using the technique presented in Ref. 38.

1. *Velocity Command*

The inputs to the NFC are past four outputs (v_0 and q), past two present reference inputs, and past one bias. The output of the flight controller is throttle position δ_t. The network architecture is represented as $N_c^{12,25,1}$ (12 input neurons, 25 hidden neurons, and 1 output neuron).

2. *Pitch Rate Command*

The inputs to the NFC are past four outputs (v_0 and q), past two present reference inputs and past one bias. The output of the NFC is elevator deflection δ_e. The NFC architecture is $N_c^{12,25,1}$ (12 input neurons, 25 hidden neurons, and 1 output neuron).

The NFC is trained using a pseudorandom signal for 100 epochs, or a training error less than 0.002 with a learning rate equal to 0.03. The offline-trained NFC are adapted online during parameter uncertainty and control surface loss. Bipolar sigmoidal function is used as an activation function in the hidden layer. During the online learning process, the weights of the NFC are adapted for three windows of samples (approximately 30 s) with a learning rate of 0.3.

V. Results and Discussion

In this section, we present the simulation study of a command control system for a linear F-8 aircraft model. The performance of the proposed NFC is compared with flight controllers developed based on the dynamic inversion scheme[18] and the RBFN scheme.[15] First, we present the results for the proposed NFC control scheme and then compare the simulation results with the other two methods. The pitch rate response of the F-8 aircraft under nominal conditions is shown in Fig. 4c. From Fig. 4c, we can clearly observe that the proposed controller alone is sufficient to improve the damping of the system and the tracking ability of the aircraft. The quantitative measures such as maximum absolute error E_q(max) and mean

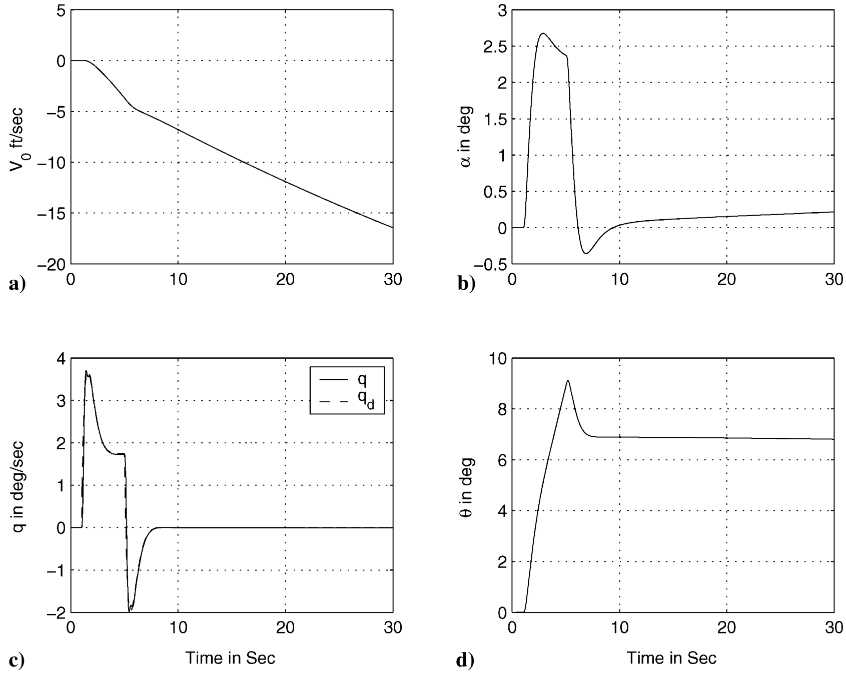

Fig. 4 Response of aircraft under pitch rate command.

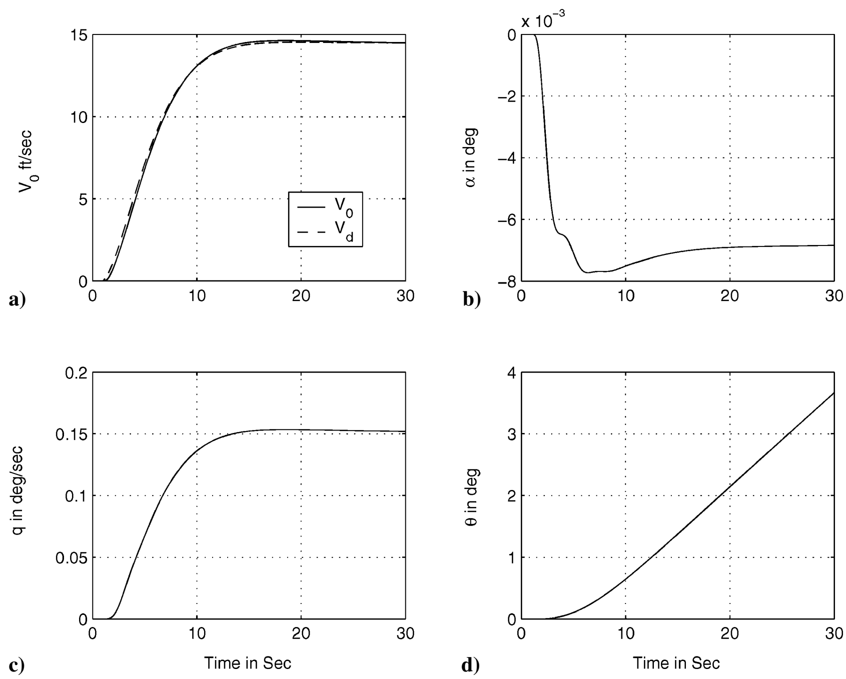

Fig. 5 Response of aircraft under velocity command.

square error $E_q(\text{MSE})$ are measured to study the performance of the controller,

$$E_q(\max) = \max |q_d(k) - q(k)|, \qquad k = 1, 2, \ldots, M$$

$$E_q(\text{MSE}) = \frac{1}{M}\sum_{k=1}^{M}[q_d(k) - q(k)]^2$$

where M is the total number of samples. The maximum and mean square errors are 0.0023 and 0.0016 deg/s. The complete response of all states are presented in Figs. 4a–4d.

Similarly, the complete state response of the aircraft under the velocity command is shown in Figs. 5a–5d. Figure 5a shows the pilot command signal and aircraft response. The maximum E_v(max and mean square error $E_v(\text{MSE})$ for a velocity command are given as

$$E_v(\max) = \max |v_d(k) - v_0(k)|, \qquad k = 1, 2, \ldots, M$$

$$E_v(\text{MSE}) = \frac{1}{M}\sum_{k=1}^{M}[v_d(k) - v_0(k)]^2$$

The $E_v(\max) = 0.0443$ and $E_v(\text{MSE}) = 0.0022$ ft/s. From Figs. 5a–5d, we observe that the proposed NFC has the ability to improve the damping and also provide the necessary tracking performance.

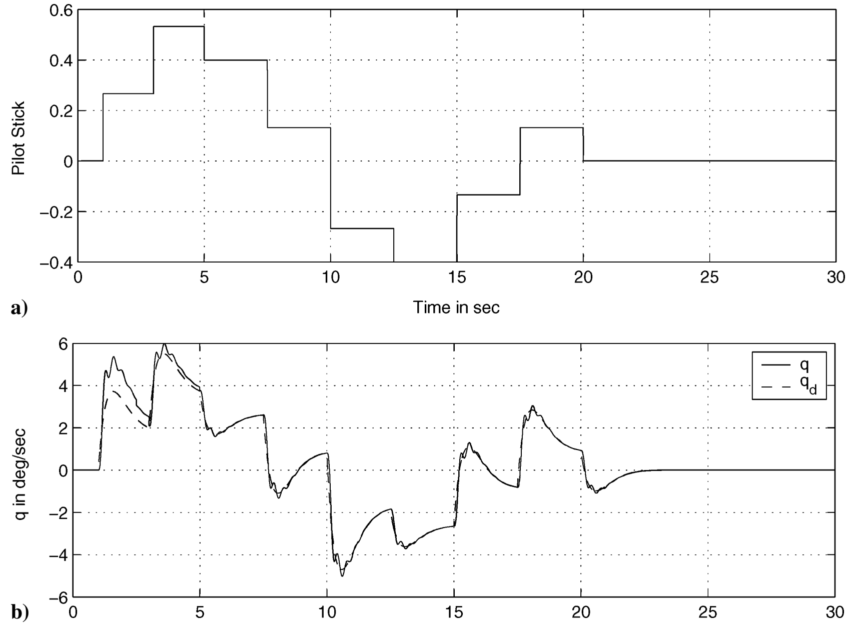

Fig. 6 Pilot stick input and pitch rate response during online training.

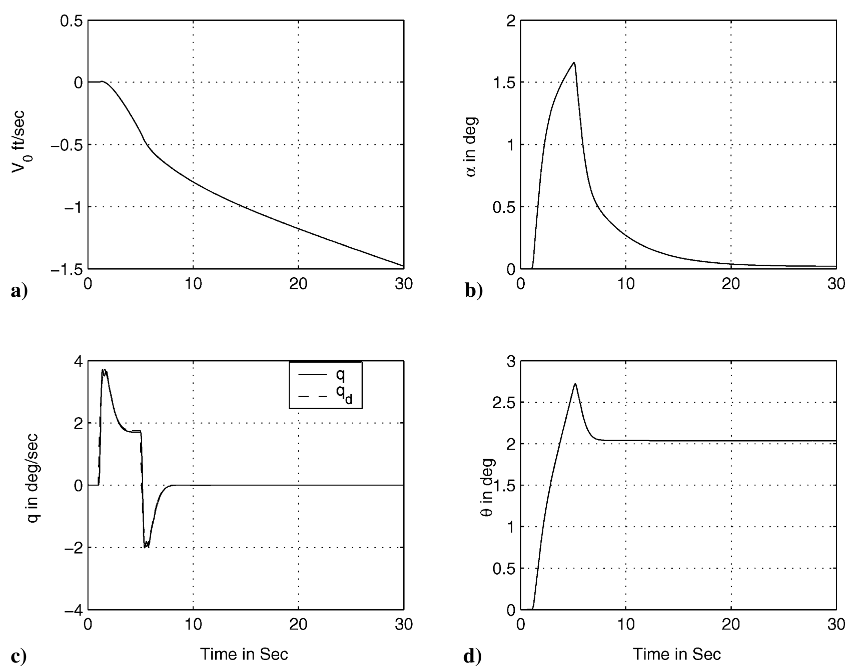

Fig. 7 Response of aircraft under pitch rate command at 70% model error.

A. Model Error

To asses the performance of the proposed NFC, the model errors are simulated by changing all of the elements of the system matrix A by 70%, and control surface losses are simulated by changing control matrix B by 70% as described in Refs. 15 and 18. First, we present the results for the model error condition ($A' = 0.3A$). Under these conditions, the aircraft response is different from the pilot command. Hence, the controller weights are adapted online when the average tracking error is greater than 0.02 deg/s. Figure 6b shows the pitch rate response of the aircraft and the reference command during online adaptation. During this period, a random pulse stick signal (Fig. 6a) is applied to generate the tracking command. From Fig. 6b, we can see that the tracking error is high only for the first 4 s, and afterward the neural controller adapts to the new environment and starts following the pilot commands accurately. After a 30-s duration, the NFC system is tested with the same tracking command signal used to test the nominal aircraft model. The response of the aircraft is shown in Figs. 7a–7d. From Fig. 7c, we observe that the aircraft response follows the pilot command accurately. Figure 7c shows a small oscillation in pitch rate after 8 s. The amplitude of oscillation is less than 0.02 deg/s, which is quite small and can be neglected. A similar study is carried out under control surface loss, and the response of the aircraft after online learning is shown in the Figs. 8a–8d. The control surface requirement under nominal, model error, and control surface loss is shown in Fig. 9. From Fig. 9, note that the control surface deflection under nominal and fault conditions are less than the maximum available limit.

Now, we present the performance results for a velocity command control system under model error and control surface loss. Figures 10a–10d show the response of the aircraft after online learning under 70% model error. The response of aircraft and pilot command is shown in Fig. 10a. Similarly, the response of the aircraft

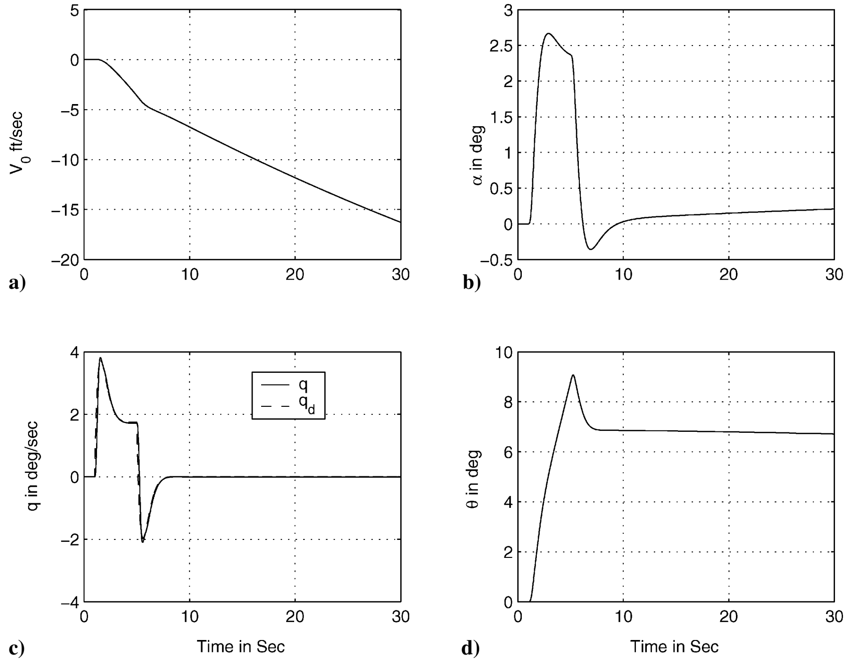

Fig. 8 Response of aircraft under pitch rate command at 70% control surface loss.

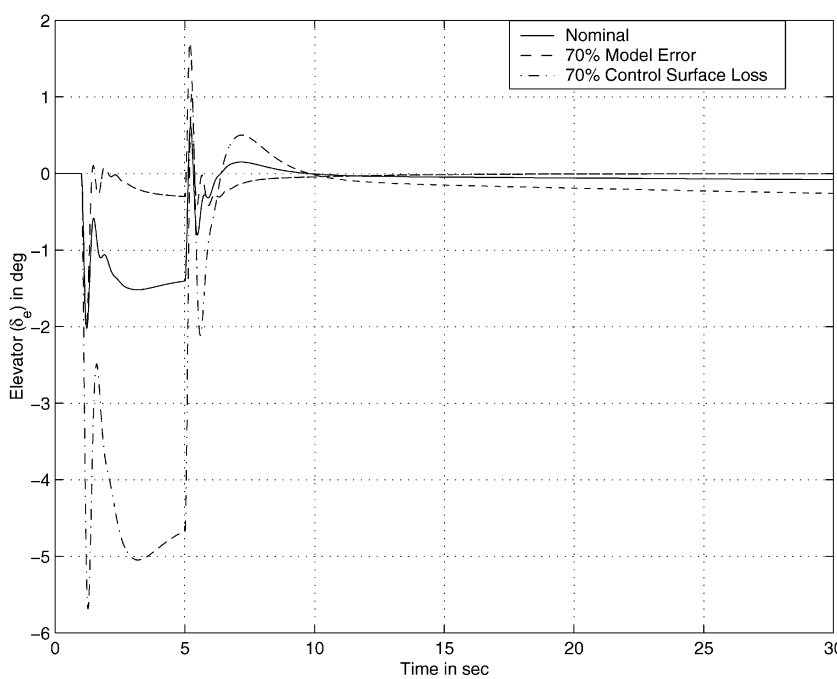

Fig. 9 Elevator requirement under nominal, model error, and control surface loss.

under the control surface (70%) is shown in Figs. 11a–11d (after online learning). The throttle requirement under nominal, model error, and control surface loss are shown in Fig. 12. From Fig. 12, the maximum throttle requirement under fault condition is less than the available limit.

Similar studies are carried out for F-8 aircraft using a dynamic inversion technique in Ref. 18 and RBFN in Ref. 15. The simulation studies of a dynamic inversion technique[18] show that the controller provided stable and near desired response in the presence of modeling uncertainty up to 30%. However, under 40% model error, the velocity oscillates in the first few seconds, though eventually settling.[18] Hence, we do not compare the performance of the dynamic inversion technique with the proposed NFC scheme under severe fault conditions. The control effort and performance measures for proposed scheme and RBFN under nominal and fault consitions are given in Table 1. Although the simulation studies based on RBFN in Ref. 15 show that the neural controller can also be applied to track the pilot command even when the model error is 70%, the control effort required is approximately seven times higher than the proposed NFC under same conditions. From Table 1, note that the control effort required to follow the pitch rate q_d command signal under nominal flight conditions for the RBFN is nearly two times greater than that of the proposed NFC scheme.

To compare the preceding methods, we determine the quantitative performance measure such as average error E(MSE) and maximum error E(max) under nominal and fault conditions. Table 1 shows the performance measure for velocity command, pitch rate command, and maximum control surface deflection under nominal

Table 1 Comparison of different neural flight controller performances

	Proposed NFC			RBFN		
Plant	E(MSE)	E(max)	Absolute max. control effort	E(MSE)	E(max)	Absolute max. control effort
			q_d			
Nominal	0.0016	0.0023	2.02 deg	0.0010	0.0010	4 deg
70%A	0.0014	0.0202	2.13 deg	0.0370	1.7500	15 deg
70%B	0.0021	0.0252	5.69 deg	——	——	——
			v_d			
Nominal	0.0022	0.0443	0.2460	0.0091	0.0100	0.810
70%A	0.0019	0.3803	0.2361	0.0260	0.6900	78.00
70%B	0.0022	0.4391	0.8209	——	——	——

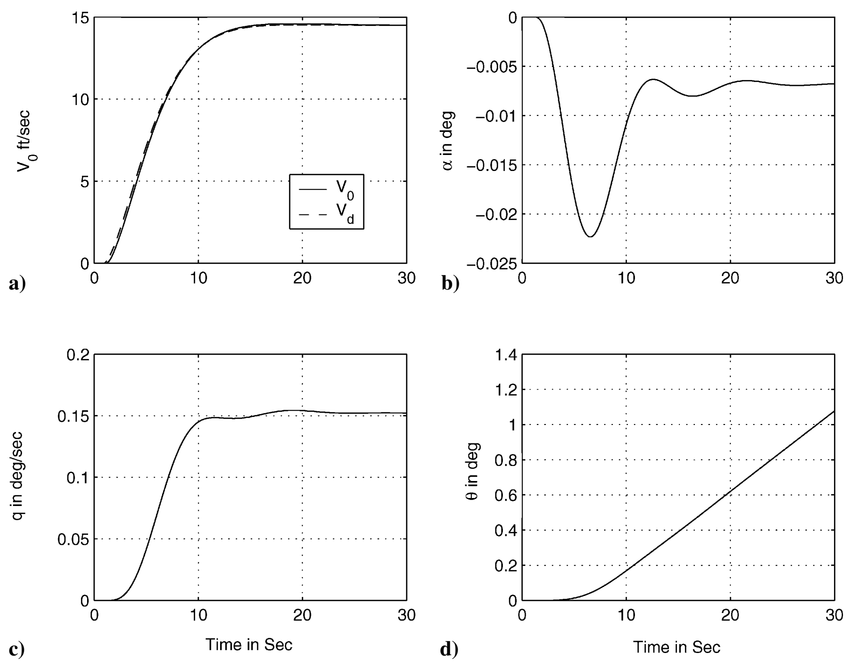

Fig. 10 Response of aircraft under velocity command at 70% model error.

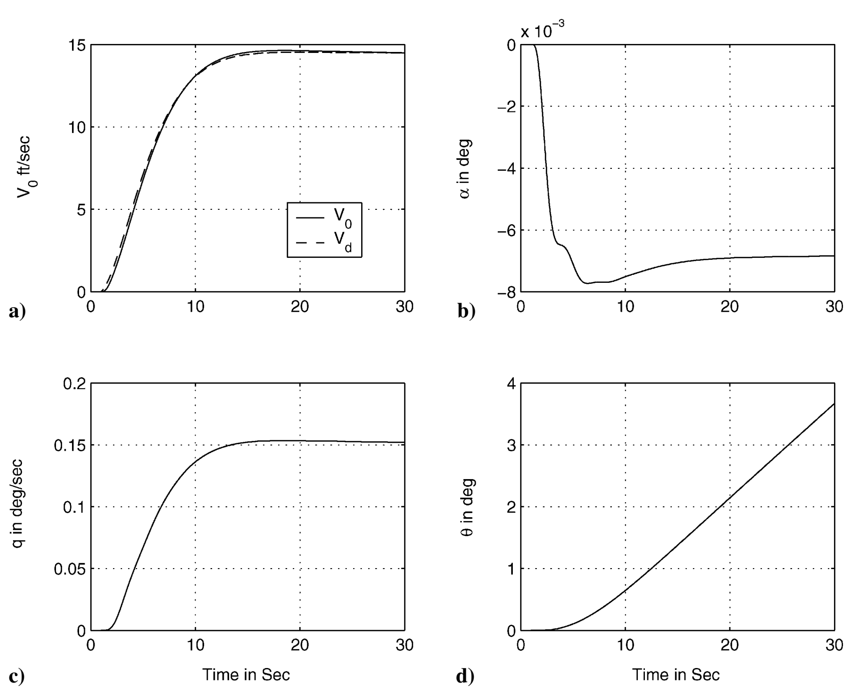

Fig. 11 Response of aircraft under velocity command at 70% control surface loss.

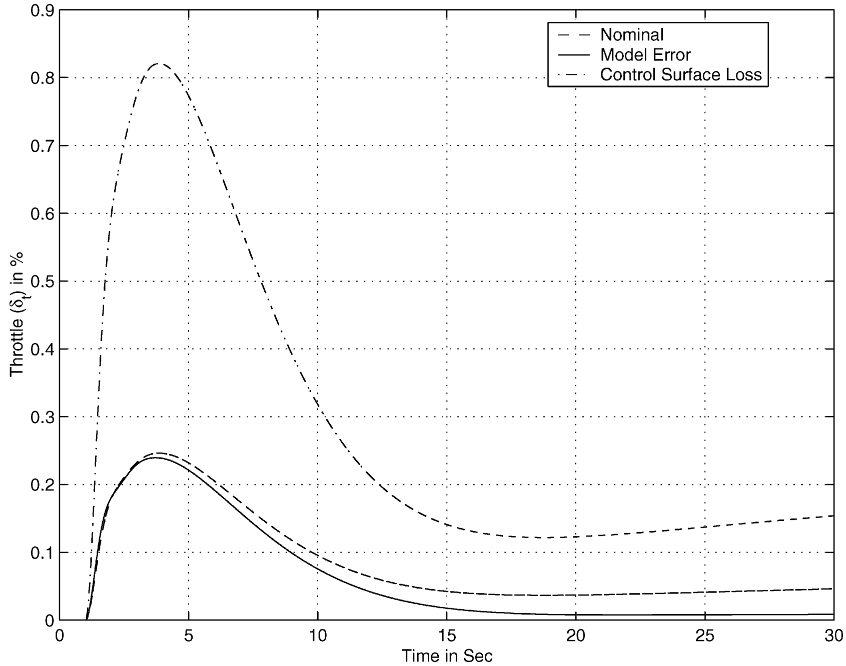

Fig. 12 Throttle requirement under nominal, model error, and control surface loss.

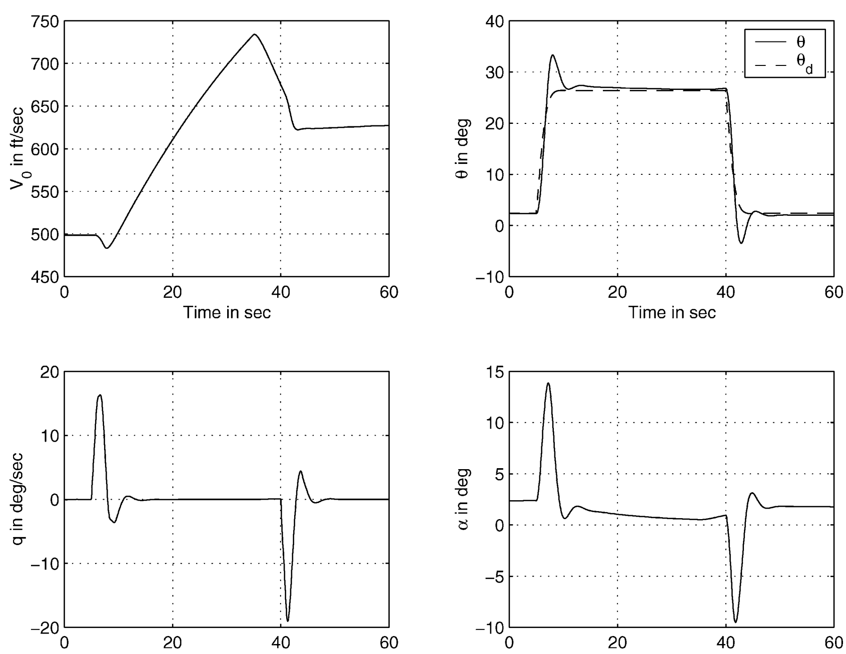

Fig. 13 Steady climb-and-hold maneuver aircraft response.

and nonnominal conditions. For the proposed control scheme, under a 70% model error ($A = 0.3A$) condition, the maximum and average errors in pitch rate command control are $E_q(\max) = 0.0202$ and $E_q(\mathrm{MSE}) = 0.0014$. Similarly, the maximum and average error for RBFN scheme are $E_q(\max) = 1.75$ and $E_q(\mathrm{MSE}) = 0.037$. The performance measures clearly show that the proposed control scheme performs better than the RBFN scheme. From Table 1, note that the proposed scheme has better tracking performance under nonnominal conditions than does the RBFN scheme.[15]

B. Nonlinear F-8 Model

In this section, the performance results of the neural controller developed earlier is presented based on a nonlinear high-performance fighter aircraft model. For this purpose, an aircraft executing a steady climb-and-hold maneuver is considered. The importance of designing an adaptive nonlinear controller to control this maneuver lies in the fact that this maneuver does exercise the aircraft in a very wide dynamic range in a short time and brings out all of the nonlinearities of the aircraft. The simulation study is carried out on a full-fledged six-degree-of-freedom nonlinear aircraft model that is developed using the MATLAB® package. Rigid-body dynamics of the aircraft are described globally (over the full flight envelope) by a set of 12 nonlinear differential equations, two each for six degrees of freedom.[39] The nonlinear functions describing the aircraft behavior known to us reasonably accurately as a mix of analytic expressions and tabular data.[39]

The elements of the body-axes state vector will comprise, respectively, the components of the velocity vector $\boldsymbol{V}_B$, the vector of Euler angles $\boldsymbol{\Phi}$, the angular rate vector $\boldsymbol{\omega}_B$, and the position vector $\boldsymbol{P}_{ned}$. Therefore, we have the state vector

$$X^T = [U \quad V \quad W \quad \phi \quad \theta \quad \psi \quad P \quad Q \quad R \quad p_N \quad p_E \quad h]$$

A number of aerodynamic forces and moment components contain dependence on the control surface deflections, and these deflections form inputs to the model. The control input vector comprises throttle setting, elevator deflection, aileron deflection, and rudder deflection,

$$U^T = [\delta_t \quad \delta_e \quad \delta_a \quad \delta_r]$$

The complete aerodynamic data acquired from wind-tunnel tests cover a wide range of angles of attack, $-20 < \alpha < 90$ deg, and slideslip angles $-30 < \beta < 30$ deg. The aircraft is powered by an afterburning turbofan jet engine. The dynamics of the actuator are modeled as a first-order transfer function.

1. Steady Climb-and-Hold

The steady climb-and-hold maneuver starts with steady straight and level flight at trim condition (trim 1) of $\delta_t = 0.14771$, $\delta_e = -1.9582$ deg, $v_0 = 500$ ft/s, $h = 500$ ft, and $\alpha = 2.368$ deg. A pitch-up command to the elevator is applied at 5 s to increase the pitch attitude θ from its trim value to 26.366 deg in a 4-s duration. The pitch attitude is kept constant for a 30-s duration, and it is decreased to initial trim value at 45 s. Now, the aircraft maintains straight and level flight conditions at trim condition (trim 2) of $\delta_t = 0.2471$, $\delta_e = -1.9322$ deg, $v_0 = 625$ ft/s, $h = 9570$ ft, and $\alpha = 2.368$ deg.

2. Simulation Results

The neural controller is trained offline using the linear plant obtained at two level flight conditions (trim 1 and trim 2). Pseudorandom pulse stick signals are used to generate the training data. The offline-trained controller is tested using the nonlinear aircraft model.

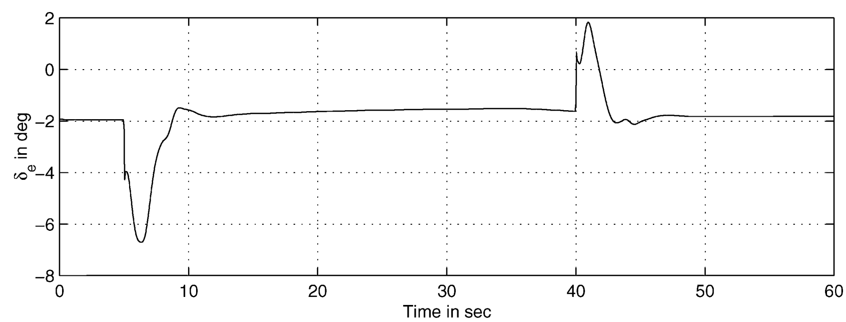

Fig. 14 Control surface deflection.

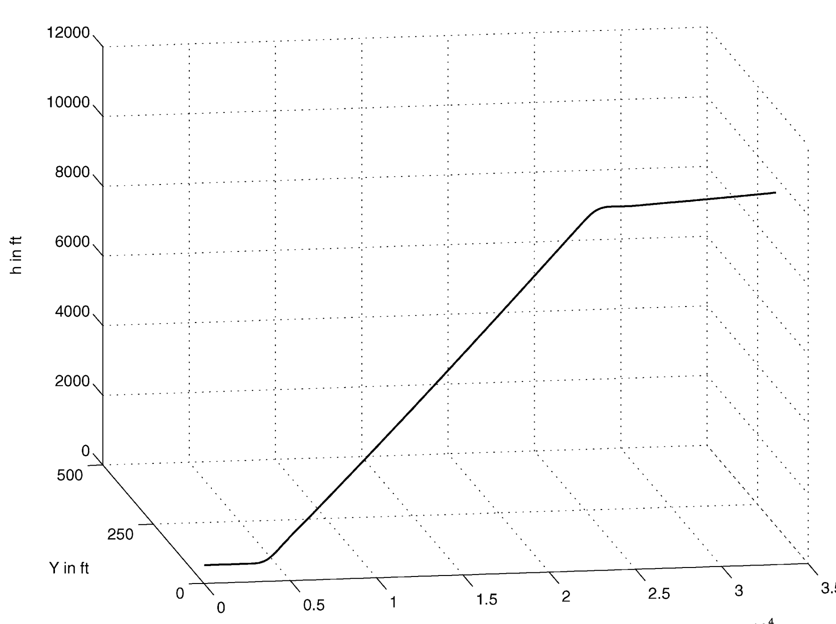

Fig. 15 Trajectory tracked by aircraft.

The controller weights are adjusted for online training for some time to compensate for the nonlinearities. The performance capabilities of the designed controller for executing the steady climb-and-hold maneuver is presented based on the results from this simulation package. Figure 13 presents the response of the nonlinear aircraft model. From Fig. 13, we can clearly observe that the aircraft follows the command accurately. Figure 14 presents the elevator and throttle deflections, and Fig. 15 presents the the trajectory of the maneuver in three dimensions using the proposed NFC. It is clear from Figs. 13–15 that the proposed control scheme provided satisfactory performance for a nonlinear aircraft maneuver.

VI. Conclusions

A stable, direct adaptive neural controller is presented based on offline (finite time interval) and online training schemes to improve damping of the system and to provide the necessary tracking performance. The neural network with linear filter model is used to approximate the unique control law. This paper also demonstrates the ability of proposed neural controller to follow the pitch rate and velocity commands in F-8 longitudinal aircraft model. The simulation studies are carried out under 70% model error and control surface loss. The simulation results show that the proposed NFC has the ability to track the command signal under model error and control surface loss. The performance of the proposed NFC is compared with a recently developed dynamic inversion technique and RBFN. The control effort required and average tracking error by the proposed NFC is less than that of the RBFN and dynamic inversion techniques. The performance of the proposed control scheme is also evaluated under a nonlinear aircraft maneuver using a six-degree-of-freedom aircraft model. Complete simulation histories of all variables are available and are well behaved.

Acknowledgments

The last author acknowledges gratefully the support extended to him by the Departments of Computer Science and Automation and Aerospace Engineering, Indian Institute of Science, Bangalore, which allowed him to visit the Indian Institute of Science as a Prof I. G. Sarma Aerodynamic Research and Development Board Memorial Professor during which this study was undertaken. The authors are thankful to the reviewers for their comments and suggestions, which have improved the quality of presentation.

References

[1]Astrom, K. J., and Wittenmark, B., *Adaptive Control*, Pearson Education, New York, 1995, Chaps. 5–6.

[2]Marino, R., and Tomei, P., *Nonlinear Control Design: Geometric, Adaptive, Robust*, Prentice–Hall, Hertfordshire, UK, 1995.

[3]Isidori, A., *Nonlinear Control Systems*, Springer-Verlag, New York, 1989.

[4]Lewis, F. L., Jagannathan, S., and Yesildirek, A., *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Francis, London, 1998.

[5]Gupta, M. M., and Rao, D. H., *Neuro-Control Systems: Theory and Applications*, IEEE Publications, New York, 1994.

[6]Hunt, A. K. J., Sbarbaro, D., Zbikowski, R., and Gawthrop, P. J., "Neural Networks for Control Systems Survey," *Automatica*, Vol. 28, No. 6, 1992, pp. 1083–1112.

[7]*IEEE Control Systems Magazine*, Special Issue on Neural Networks in Control Systems, Vol. 10, No. 3, 1990, pp. 6–125.

[8]Narendra, K. S., and Parthasarathy, K., "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Transactions on Neural Networks*, Vol. 1, No. 1, 1990, pp. 4–27.

[9]Chen, F. C., and Khalil, H. K., "Adaptive Control of Nonlinear Systems Using Neural Networks," *Proceedings of the IEEE Conference on Decision and Control*, IEEE Publications, Piscataway, NJ, 1990, pp. 1707–1712.

[10]Polycarpou, M. M., "Stable Adaptive Neural Control Scheme for Nonlinear Systems," *IEEE Transactions on Automatic Control*, Vol. 41, No. 3, 1996, pp. 447–451.

[11]Werbos, P. J., "Backpropagation Through Time: What It Does and How to Do It," *Proceedings of the IEEE*, Vol. 78, 1990, pp. 1550–1560.

[12]Puskorius, G. V., and Feldkamp, L. A., "Model Reference Adaptive Control with Recurrent Networks Trained by the Dynamic DEKF Algorithm," *International Joint Conference on Neural Networks*, Vol. 2, IEEE Publications, Piscataway, NJ, 1992, pp. 106–113.

[13]Cabrera, J. B. D., and Narendra, K. S., "On Regulation and Tracking in Nonlinear Discrete Time Systems—Part I: State Variables Accessible," Technical Rept. No. 9515, Center for System Science, Yale Univ., New Haven, CT, 1992.

[14]Sanner, R. M., and Slotine, J. J. E., "Gaussian Networks for Direct Adaptive Control," *IEEE Transactions on Neural Networks*, Vol. 3, No. 6, 1992, pp. 837–863.

[15]Li, Y., Sundararajan, N., and Saratchandran, P., "Stable Neuro-Flight-Controller Using Fully Tuned Radial Basis Function Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, 2001, pp. 665–674.

[16]Li, Y., Sundararajan, N., and Saratchandran, P., "Neuro-Flight Controllers for Aircraft Using Minimal Resource Allocating Networks," *Neural Computing and Applications*, Vol. 10, No. 2, 2001, pp. 172–183.

[17]Shin, D.-H., and Kim, Y., "Reconfigurable Flight Control System Design Using Adaptive Neural Network," *IEEE Transactions on Control System Technology*, Vol. 12, No. 1, 2004, pp. 87–100.

[18]Sadhukhan, D., and Feteih, S., "F8 Neuro-Controller Based on Dynamic Inversion," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 1, 1996, pp. 150–156.

[19]Calise, A. J., and Rysdyk, R. T., "Nonlinear Adaptive Control Using Neural Networks," *IEEE Control Systems Magazine*, Vol. 18, No. 6, 1998, pp. 14–25.

[20]Kim, B., and Calise, A. J., "Nonlinear Adaptive Flight Control Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 26–33.

[21]Gili, P. A., and Battipede, M., "Adaptive Neuro-Controller for a Nonlinear Combat Aircraft Model," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, 2001, pp. 910–917.

[22]Hovakimyan, H., Rysdyk, R., and Calise, A. J., "Dynamic Neural Network for Output Feedback Control," *38th IEEE Conference on Decision Control*, Vol. 2, IEEE Publications, Piscataway, NJ, 1999, pp. 1685–1690.

[23]Li, Y., Sundararajan, N., and Sarachandran, P., "Neuro-Controller Design for Nonlinear Fighter Aircraft Maneuver Using Fully Tuned RBF Networks," *Automatica*, Vol. 37, No. 8, 2001, pp. 1293–1301.

[24]Lewis, F. L., Yesilidrek, A., and Jagannathan, S., *Neural Network Control of Robot Manipulator and Nonlinear Systems*, Taylor and Francis, Philadelphia, 1999.

[25]Lewis, F. L., Yesildirek, A., and Liu, K., "Multilayer Neural-Network Robot Controller with Guaranteed Tracking Performance," *IEEE Transaction on Neural Networks*, Vol. 7, No. 2, 1996, pp. 388–399.

[26]McFarland, M. B., and Calise, A. J., "Multilayer Neural Networks and Adaptive Control of Agile Anti-Air Missile," AIAA Paper 97-3540, 1997.

[27]Leitner, J., Calise, A. J., and Prasad, J. V. R., "Analysis of Adaptive Neural Networks for Helicopter Flight Controls," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 5, 1997, pp. 972–979.

[28]Calise, A. J., "Neural Networks in Nonlinear Aircraft Flight Control," *IEEE Aerospace and Electronic System Magazine*, Vol. 11, No. 7, 1996, pp. 5–10.

[29]Suresh, S., Omkar, S. N., Mani, V., and Sundararajan, N., "Nonlinear Adaptive Neural Controller for Unstable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 6, 2005, pp. 1103–1111.

[30]Shtessel, Y., Buffington, J., and Banda, S., "Multiple Timescale Flight Control Using Reconfigurable Sliding Modes," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 6, 1999, pp. 873–883.

[31]Zhang, Y., and Jiang, J., "Fault Tolerant Control System Design with Explicit Consideration of Performance Degradation," *IEEE Transactions on Aerospace and Electronics Systems*, Vol. 39, No. 3, 2003, pp. 838–848.

[32]Narendra, K. S., "Neural Networks for Control: Theory and Practice," *Proceedings of the IEEE*, Vol. 84, No. 10, 1996, pp. 1385–1406.

[33]Rynaski, E. G., "Flying Qualities in the Time Domain," AIAA Paper 85-1849, Aug. 1985.

[34]Barron, A. R., "Universal Approximation Bounds For Superpositions Of A Sigmoidal Function," *IEEE Transactions on Information Theory*, Vol. 39, No. 3, 1993, pp. 930–945.

[35]Hornik, K., Stinchcombe, M., and White, H., "Universal Approximation Of An Unknown Mapping And Its Derivatives Using Multilayer Feedforward Networks," *Neural Networks*, Vol. 3, No. 5, 1990, pp. 551–560.

[36]Cybenko, G., "Approximation by Superpositions of a Sigmoidal Function," *Mathematics of Control, Signals, and Systems*, Vol. 2, No. 4, 1989, pp. 303–314.

[37]Hornik, K., Stinchombe, M., and White, H., "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks*, Vol. 2, No. 5, 1989, pp. 359–366.

[38]Suresh, S., Omkar, S. N., Mani, V., and GuruPrakash, T. N., "Lift Coefficient Prediction At High Angle Of Attack Using Recurrent Neural Network," *Aerospace Science and Technology*, Vol. 7, No. 8, 2003, pp. 595–602.

[39]Stevens, B. L., and Lewis, F. L., *Aircraft Control and Simulation*, Wiley, New York, 1992.