Engineering Applications of Artificial Intelligence 22 (2009) 181–191

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A direct adaptive neural command controller design for an unstable helicopter

M. Vijaya Kumar [a], S. Suresh [b], S.N. Omkar [b], Ranjan Ganguli [b,*], Prasad Sampath [a]

[a] Rotary Wing Research and Design Centre, Hindustan Aeronautics Limited, Bangalore 560017, India
[b] Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560012, India

ARTICLE INFO

Article history:
Received 27 July 2007
Received in revised form
27 May 2008
Accepted 19 July 2008
Available online 23 October 2008

Keywords:
Neural network
Helicopter
Aircraft control
Neural controller
Backpropagation

ABSTRACT

This paper presents an off-line (finite time interval) and on-line learning direct adaptive neural controller for an unstable helicopter. The neural controller is designed to track pitch rate command signal generated using the reference model. A helicopter having a soft inplane four-bladed hingeless main rotor and a four-bladed tail rotor with conventional mechanical controls is used for the simulation studies. For the simulation study, a linearized helicopter model at different straight and level flight conditions is considered. A neural network with a linear filter architecture trained using back-propagation through time is used to approximate the control law. The controller network parameters are adapted using updated rules Lyapunov synthesis. The off-line trained (for finite time interval) network provides the necessary stability and tracking performance. The on-line learning is used to adapt the network under varying flight conditions. The on-line learning ability is demonstrated through parameter uncertainties. The performance of the proposed direct adaptive neural controller (DANC) is compared with feedback error learning neural controller (FENC).

1. Introduction

Traditional flight control design for helicopters involves linearizing the vehicle dynamics about several operating conditions throughout the flight envelope, tuning the gains of the linear controllers for each condition, and scheduling these gains with an interpolation scheme. Although gain scheduling has historically proven successful in a variety of applications (Apkarian et al., 1995; Leith and Leithead, 2000), the present helicopter control requires flight control schemes which explicitly account for the intrinsic nonlinearities of the system. The limitation of the classical gain scheduling techniques is that they exploit the behavior only in the vicinity of the equilibrium operating points and it generally imposes as inherent slow variation requirement on the system to ensue that the state remains close to equilibrium. A neural network approach attempt to relax restrictions to operate near equilibrium point (Leith and Leithead, 2000). In addition, tactical military flight requirements for agile helicopters and stringent day-visual civil flight regulations demand more accurate track control through out the flight envelope (Aeronautical Design Standard, 1994; Tischler, 1996). The complex gain scheduling which requires time consuming flight tests and stringent requirements can be circumvented using nonlinear adaptive control system design. In addition, variation in the helicopter model may occur due to battle damage or component failure, requiring rapid on-line reconfiguration of the control system to maintain stable flight and reasonable levels of handling qualities (Calise and Rysdyk, 1998). Therefore, there is presently a strong interest in the development of on-line adaptive control methods that are applicable to flight control problems.

The emergence of the neural network paradigm as a powerful tool for learning complex mappings from a set of examples has generated a great deal of interest in using neural network models in various applications in aeronautics (Faller and Schreck, 2000; Melin and Castillo, 2003). Neural networks have been used in the identification and control of dynamical systems (Uraikul et al., 2007; Peng et al., 2007; Wang et al., 2006; Pepijn et al., 2005; Castillo and Melin, 2003; Kiong et al., 2003). Due to their approximation capabilities as well as their inherent adaptive features, artificial neural networks present a potentially appealing alternative to modeling of nonlinear systems (Roy and Ganguli, 2006). Furthermore, from a practical perspective, the massive parallelism and fast adaptability of neural network implementations provide more incentive for investigating the connectionist approach in problems involving dynamical systems with unknown nonlinearities. Among various adaptive control schemes, model reference adaptive control, dynamic inversion, adaptive critic and feedback error learning controls are widely used

* Corresponding author.
E-mail address: ganguli@aero.iisc.ernet.in (R. Ganguli).

0952-1976/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2008.07.004

(Narendra, 1996; Anon, 1990; Narendra and Parthasarathy, 1990; Polycarpou, 1996; Li et al., 2001; Werbos, 1995; Leitner et al., 1997). Investigations have been done to study the on-line learning ability of these control schemes in the presence aerodynamic uncertainty, damage to the helicopter components and loss of control surface. Neural network based control schemes for helicopters have been investigated in Leitner et al. (1997), Saade et al. (1996), Narendra and Mukhopadhyay (1992). A complete survey of adaptive neural control systems for various applications is presented in Ng (1997).

Most of the applications using neural network as control architecture, use a conventional controller in the inner loop to stabilize the system dynamics, while the neural controller acts as an aid to the conventional controller by compensating for the nonlinearity. This control scheme is commonly known as feedback error learning control (FENC) (Li et al., 2001). The necessary bounded signal requirement for neural network training is satisfied using an inner conventional state feedback controller (SFC). Assuming the inner conventional controller provides necessary bounded signals in fault and non-nominal conditions, the neural network is further adapted to provide the required tracking performance.

Even though the tracking error is less, the control effort required to follow the command is very high. To overcome these problems, we present in this paper, a direct adaptive neural controller (DANC) scheme to track the pitch rate command signal. The lateral and directional axes are stabilized using state feedback design and the longitudinal axis is controlled using a neural network. This control approach can as well be used to control other axes such as lateral and directional, however, this investigation is not subject of this paper. Since the helicopter model considered is unstable, off-line (finite time interval) and an on-line learning strategy presented in Suresh et al. (2004) is used to adapt the neural controller. The proposed learning scheme is based on the assumption that the states and outputs of the system do not escape to infinity in a finite time interval. Using the above assumption, a neural network with linear filter is trained off-line using the backpropagation through time learning algorithm to approximate the unknown control law. The off-line trained neural network is used as the starting point for on-line adaptation under variations in aerodynamic coefficients or control effectiveness deficiencies caused by control surface damage.

Performance of the proposed control scheme is evaluated for a flight controller design based on a linear model of a helicopter. To investigate the on-line learning ability of the proposed neural controller, different fault scenarios representing large model error and control surface loss are considered. The performances of the proposed DANC scheme are compared with the FENC scheme.

2. Mathematical model

Before presenting a mathematical statement of the proposed neuro-control strategy, we shall describe first quantitatively the class of problem which is encountered in practical situations. A plant to be controlled (Σ) is unstable for a given class of bounded input signal ($\mathbf{u}$), and the question is raised whether a neuro-controller can be designed to stabilize the plant and follow the reference model or command signal. A finite data set from the reference model is provided for the purpose of designing a neuro-controller. The objective is therefore to estimate the controller parameters such that the given plant tracks the reference model or command signal accurately.

Neural controller used in this paper is direct adaptive neural controller as shown in Fig. 1. In this control strategy, a reference model that meets the requirements of ADS-33 is implemented in the scheme. The feedback error is the difference between helicopter model output and the reference block and is feed back to the neural controller obtained from a reference model.

The input to the system is the sum of pilot input $r(t)$ and neural controller output $\mathbf{u}^*(t)$

$$\mathbf{u}_{\mathrm{danc}}(t) = \mathbf{u}^*(t) + \mathbf{r}(t) \tag{1}$$

The neural controller provides stabilization and compensates for nonlinearities and parameter uncertainty. The on-line learning neural controller augments the performance of the baseline controller in order to achieve better performance. The basic building block in the neuro-controller is a Nonlinear Auto-Regressive eXogenous (NARX) input model. The controller parameters are updated using a Lyapunov based synthesis.

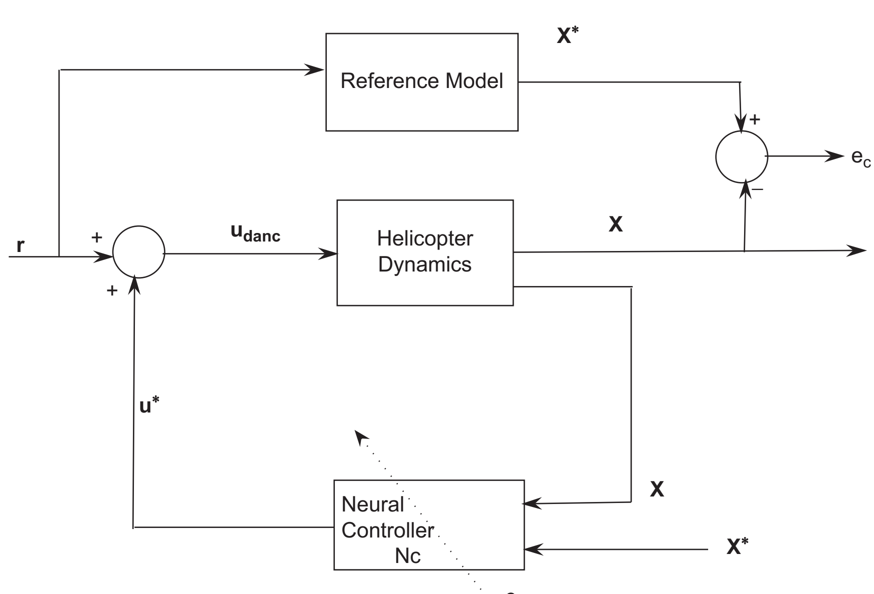

Fig. 1. Schematic diagram of DANC scheme.

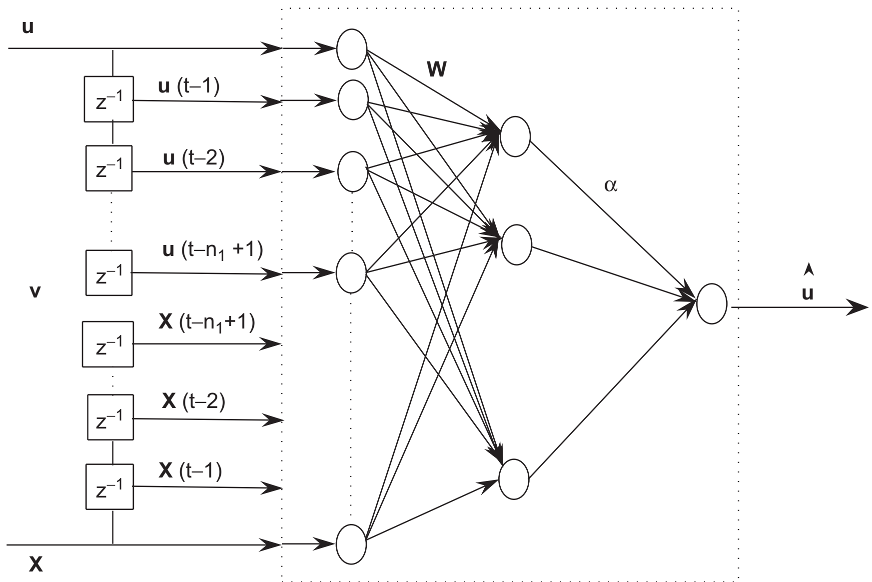

Fig. 2. Nonlinear Auto-Regressive eXogeneous (NARX) input network architecture.

2.1. NARX controller

Fig. 2 shows the architecture of the NARX model (Narendra and Parthasarathy, 1990; Haykin, 1999). The neural network used for the control is the NARX model. The basic building blocks in the NARX model are tapped-delay-lines (TDL) and MLP. This is the model explored by Narendra (Narendra and Parthasarathy, 1990). This model can be used for system identification as well. The considerations for selecting number of inputs, number of hidden nodes for the NARX model are given in VijayaKumar et al. (2006).

The basic building blocks in the NARX model are TDL and MLP. The MLP approximates the nonlinear function and TDL introduces the dynamics into the NARX model. The control inputs and the response of the helicopter model are applied to a TDL of n_1 units. Typically n_1 is the order of the plant or the system. The inputs to the NARX model consist of the delayed control inputs and responses and the present control input. Thus, the inputs applied to the NARX model $\mathbf{v}$ consist of the following:

(a) Present and past values of the input, namely, $\mathbf{u}(t), \mathbf{u}(t-1)\ldots \mathbf{u}(t-n_1+1)$, which represent exogenous inputs originating from outside the network.
(b) Delayed values of the output, namely, $\mathbf{X}(t), \mathbf{X}(t-1)\ldots \mathbf{X}(t-n_1+1)$, on which the model output $(t+1)$ is regressed.

Therefore input $\mathbf{v}$ is

$$\mathbf{v} = (\mathbf{u}(t), \ldots, \mathbf{u}(t-n_1+1), \mathbf{X}(t), \ldots, \mathbf{X}(t-n_1+1)) \tag{2}$$

The network output is then given by

$$\hat{\mathbf{u}}(t+1) = N_F(\mathbf{v}) \tag{3}$$

where, $N_F(.)$ is the network approximation for function $\mathbf{u}^*$ in Eq. (1).

Consider a specific architecture used for this study shown in Fig. 2. This neural network has an input layer, one hidden layer and one output layer. $\mathbf{v}$ is the input vector. The input is passed through a sigmoidal activation function σ, and weight vector is $\mathbf{W}$. The output layer has a weight vector α and the output of the network is $\mathbf{u}$. Mathematically, the output of the neural network is given by

$$\hat{\mathbf{u}} = \sum_{i=1}^{h^*} \alpha_i \sigma \left(\sum_{j=1}^{l} w_{ij} v_j \right) \tag{4}$$

where l is the number of input nodes, h^* is the number of hidden nodes, $\mathbf{W}, \alpha$ are the weight matrices in the hidden and output layers and $\mathbf{v} \in \Re^{l\times 1}$ is the input to the controller, the nonlinear function $\sigma : \Re^{h^*} \to \Re^{h^*}$ is continuous with respect to its arguments for all finite $(\mathbf{W}, \mathbf{v})$, and the adjustable parameters of the controller are elements of $\alpha \in \Re^{m\times h^*}$ and $\mathbf{W} \in \Re^{h^*\times l}$.

The controller parameters are updated using a Lyapunov-based synthesis as described below. The readers attention is drawn to similar studies reported to generate Lyapunov-based update algorithms (Li et al., 2001; Rovithakis and Christodoulou, 1994).

2.2. Adaptive control law formulation

Let the helicopter dynamics be represented as a continuous system of the form

$$\dot{\mathbf{X}} = f_1(\mathbf{X}, \mathbf{u}) \quad \text{given } \mathbf{X}_0 = \mathbf{X}(t_0) \tag{5}$$

where $\mathbf{u} \in \Re^p$ is the $p \times 1$ bounded input vector, $\mathbf{X} \in \Re^n$ is a bounded $n \times 1$ state vector. Assume that the zero dynamics $f_1(0,0)$ is asymptotically stable.

The objective of the problem is to determine the bounded control input $\mathbf{u}$ such that plant output follows the reference signal accurately. The $\mathbf{X}$ output follows the desired response $\mathbf{X}_d$ accurately, i.e.,

$$\|\mathbf{X} - \mathbf{X}_d\| \leqslant \varepsilon \tag{6}$$

where ε is a small positive constant. It will also be assumed that the function f_1 has continuous bounded partial derivatives in a certain neighborhood of all points along the desired response. Determining the conditions under which the desired control $\mathbf{u}^*$ exists forms the theoretical control problem. Under the above assumptions, it follows from implicit function theorem (Narendra, 1996), that there exists a desired control force $\mathbf{u}^*$ of the form:

$$\mathbf{u}^*(t) = \bar{\mathbf{G}}((\mathbf{X}(t), \mathbf{X}(t-1), \ldots, \mathbf{X}(t-n_1), \\ \mathbf{u}(t), \mathbf{u}(t-1), \ldots, \mathbf{u}(t-n_1), \mathbf{y}_d(t+1)) \tag{7}$$

where $\bar{G}$ is a smooth nonlinear map and n_1 is the order of the plant or the system. The above form of control law is known to exist and is unique (Narendra, 1996). Eq. (7) can be simplified further if it is assumed that the system output follows the desired trajectory and the desired responses are assumed to be zero. The simplified control form then becomes

$$\mathbf{u}^*(t) = \bar{\mathbf{G}}(\mathbf{v}) \tag{8}$$

where $\mathbf{v}$ consists of past measurements and present and past values of the state variables and control inputs. Eq. (7) means, if the mapping $\bar{\mathbf{G}}$ is known, then the desired control inputs $\mathbf{u}^*$ can be calculated using n past values of the response measurements, and $n+1$ current and past values of the state variables. Since the function map $\bar{\mathbf{G}}$ is unknown, estimating control input $\mathbf{u}^*$ for a given desired response $\mathbf{y}_\mathrm{d}$ is not possible. However, since the relationship given by Eq. (7) exists, $\bar{\mathbf{G}}$ can be modelled using a nonlinearly parameterized controller. The control law given in Eq. (7) is approximated using a linear combination of h^* nonlinearly parameterized functions, each represented by the symbol σ. Neural network as mathematically represented in Eq. (4) can be used

$$\mathbf{u}^* = \sum_{i=1}^{h^*} \alpha_i^* \sigma\left(\sum_{j=1}^{l} w_{ij}^* v_j\right) \tag{9}$$

where $\mathbf{v} \in \Re^{l\times 1}$ is the input to the controller, the nonlinear function $\sigma : \Re^{h^*} \to \Re^{h^*}$ is continuous with respect to its arguments for all finite $(\mathbf{W}, \mathbf{v})$, and the adjustable parameters of the controller are elements of $\alpha \in \Re^{m\times h^*}$ and $\mathbf{W} \in \Re^{h^*\times l}$. Eq. (4) can be written in matrix form as

$$\mathbf{u}^* = \alpha^* \sigma(\mathbf{W}^* \mathbf{v}) \tag{10}$$

Theoretically, the nonlinearly parameterized controller can model any sufficiently smooth function on a compact set, with arbitrarily bounded modelling error ε_M, by a sufficiently large number of neurons (h^*). That is, for a given positive modelling error bound ε_M and a continuous function σ, there exists adaptable parameters $\alpha = \alpha^*$ and $\mathbf{W} = \mathbf{W}^*$, such that the output of the controller satisfies

$$\max_{\mathbf{v}\in\Omega_v} \|\hat{\mathbf{G}}(\mathbf{v}, \alpha, \mathbf{W}) - \bar{\mathbf{G}}(\mathbf{v})\| \leqslant \varepsilon_\mathrm{M} \tag{11}$$

where $\|\cdot\|$ is any suitable norm. That is,

$$\bar{\mathbf{G}}(\mathbf{v}) = \alpha_i^* \sigma(\mathbf{W}^* \mathbf{v}) + \varepsilon \tag{12}$$

where $\|\varepsilon\| \leqslant \varepsilon_\mathrm{M}$, α^* and W^* are the ideal values of the controller parameters which are defined as

$$(\alpha^*, W^*) := arg \min_{(\alpha,\mathbf{W})\in\Omega_w} \left\{ \sup_{\mathbf{v}\in\Omega_v} \|\alpha^\mathrm{T}\sigma(\mathbf{W}\mathbf{v}) - \bar{\mathbf{G}}(\mathbf{v})\| \right\} \tag{13}$$

in which $\Omega_w := \{(\alpha, \mathbf{W}) \|\alpha\| \leqslant M_\alpha, \|\mathbf{W}\|_\mathrm{F} \leqslant M_w\}$, M_α and M_w are positive constants and $\|\|_\mathrm{F}$ is the Frobenius matrix norm, defined as

$$\|\mathbf{W}\|_\mathrm{F} = \mathrm{tr}(\mathbf{W}^\mathrm{T}\mathbf{W}) = \sum_{i,j} w_{ij}^2 \tag{14}$$

where tr(.) denotes the trace of a matrix.

The value of modelling error (ε_M) depends on many factors, such as type of nonlinear function σ, number of functions (h^*), elements of α and $\mathbf{W}$, as well as size of compact sets Ω_v and Ω_w. Practically, the optimal parameters $(\boldsymbol{\alpha}^*, \mathbf{W}^*)$ of the nonlinearly parameterized controller are unknown. Hence, the nonlinear function $\bar{\mathbf{G}}$ can only be approximated. The approximated adaptive control force is then given by

$$\hat{\mathbf{u}} = \boldsymbol{\alpha}\boldsymbol{\sigma}(\mathbf{W}\mathbf{v}) \tag{15}$$

In the following discussion, a stable parameter update law for the controller will be derived, and it will be shown that the estimated parameters remain bounded.

2.3. Stable parameter update law

By substituting the control law given in Eq. (12) into Eq. (5) one gets

$$\dot{\mathbf{X}} = f_1(\mathbf{X}, \mathbf{u}^*) + \varepsilon \tag{16}$$

Now, substituting the approximate control law given in Eq. (15) into Eq. (5), one gets

$$\dot{\hat{\mathbf{X}}} = f_1(\hat{\mathbf{X}}, \hat{\mathbf{u}}) \tag{17}$$

If the error is defined as $\mathbf{e} = \hat{\mathbf{X}} - \mathbf{X}$, then the error dynamics of the system can be written as

$$\dot{\mathbf{e}} = f_1(\hat{\mathbf{X}}, \hat{\mathbf{u}}) - f_1(\mathbf{X}, \mathbf{u}^*) \tag{18}$$

Linearizing Eq. (18) around $x(0)$ where A is $\partial f_1/\partial \mathbf{X}^\mathrm{T}$ and B is $\partial f_1/\partial \mathbf{u}^\mathrm{T}$ at time $t - t_0$, and the error dynamics can be written as

$$\dot{\mathbf{e}} = \mathbf{A}_1(\hat{\mathbf{X}}) - \mathbf{A}_1(\mathbf{X}) + f_0(\hat{\mathbf{X}}) - f_0(\mathbf{X}) + \mathbf{B}[\alpha\sigma(\mathbf{W}\mathbf{v}) - \alpha^*\sigma(\mathbf{W}^*\mathbf{v}) - \varepsilon] \tag{19}$$

where f_0 are the higher-order terms in a Taylor series expansion. Since the higher-order terms are small, and the first-order term dominates, they are neglected from the equation. The higher-order terms around the equilibrium point are assumed to be small and bounded for the update law derivation. During the simulation, these terms are not neglected. The neural controller weights are adapted such that actual response (including the higher-order terms) tends to follow the reference model. Now, simplified error dynamics can be written as

$$\dot{\mathbf{e}} = \mathbf{A}_1\mathbf{e}_1 + \mathbf{B}[\alpha\sigma(\mathbf{W}\mathbf{v}) - \alpha^*\sigma(\mathbf{W}^*\mathbf{v}) - \varepsilon] \tag{20}$$

where $\mathbf{e}_1 = \hat{\mathbf{X}} - \mathbf{X}$, and $\mathbf{A}_1 = f_1'|_0$.

If the parameter errors are defined as, $\tilde{\alpha} := \alpha - \alpha^*$, $\tilde{\mathbf{W}} := \mathbf{W} - \mathbf{W}^*$ and $\tilde{\sigma} := \sigma(\mathbf{W}\mathbf{v}) - \sigma(W^*\mathbf{v})$, then, by neglecting the higher-order terms, the error dynamics can be written as

$$\dot{\mathbf{e}} = \mathbf{A}_1\mathbf{e}_1 + \mathbf{B}[(\tilde{\alpha} + \alpha^*)\sigma(\mathbf{W}\mathbf{v}) - \alpha^*\sigma(\mathbf{W}^*\mathbf{v}) - \varepsilon]$$

$$\dot{\mathbf{e}} = \mathbf{A}_1\mathbf{e}_1 + \mathbf{B}[\tilde{\alpha}\sigma(\mathbf{W}\mathbf{v}) + \alpha^*\tilde{\sigma} - \varepsilon] \tag{21}$$

To facilitate the derivation of the parameter update laws, $\tilde{\sigma}$ is expanded in a Taylor series around the optimal value W^* as

$$\tilde{\sigma} = \sigma(\mathbf{W}\mathbf{v}) - \sigma(\mathbf{W}^*\mathbf{v}) = \dot{\sigma}(\mathbf{W}\mathbf{v})(\mathbf{W} - \mathbf{W}^*)\mathbf{v} - \sigma_0(\mathbf{W}\mathbf{v}) \tag{22}$$

where σ_0 are the higher-order terms of the Taylor expansion, which can be written as

$$\sigma_0(\mathbf{W}\mathbf{v}) = \sigma(\mathbf{W}\mathbf{v}) - \sigma(\mathbf{W}^*\mathbf{v}) - \dot{\sigma}(\mathbf{W}\mathbf{v})\tilde{\mathbf{W}}\mathbf{v} \tag{23}$$

The higher-order terms in Eq. (23) can be shown to be bounded using the mean-value theorem as follows.

Using mean-value theorem, $\exists\bar{\mathbf{W}} \in [\mathbf{W}, \mathbf{W}^*]$, i.e. $\bar{\mathbf{W}} = \lambda\mathbf{W} + (1-\lambda)\mathbf{W}^*$, for some $\lambda \in [0.1]$, such that

$$\begin{aligned}\sigma_0(\mathbf{W}\mathbf{v}) &= \dot{\sigma}(\mathbf{W}\mathbf{v})|_{\mathbf{W}=\bar{\mathbf{W}}}\tilde{\mathbf{W}}\mathbf{v} - \dot{\sigma}(\mathbf{W}\mathbf{v})\tilde{\mathbf{W}}\mathbf{v} \\ &= [\dot{\sigma}(\mathbf{W}\mathbf{v})|_{\mathbf{W}=\bar{\mathbf{W}}} - \dot{\sigma}(\mathbf{W}\mathbf{v})]\tilde{\mathbf{W}}\mathbf{v}\end{aligned}$$

Let

$$\delta := \sup_{\mathbf{v}\in\Omega_v, \mathbf{W}\in\Omega_v, \mathbf{W}\in[\mathbf{W},\mathbf{W}^*]} \|\dot{\sigma}(\mathbf{W}\mathbf{v})|_{\mathbf{W}=\bar{\mathbf{W}}} - \dot{\sigma}(\mathbf{W}\mathbf{v})\|$$

Then, the higher-order terms are bounded by

$$|\sigma_0(\mathbf{W}\mathbf{v})| \leqslant \delta\|\tilde{\mathbf{W}}\mathbf{v}\|$$

By substituting $\tilde{\sigma}$ from Eq. (22) into the error dynamics Eq. (21), one gets

$$\dot{\mathbf{e}} = \mathbf{A}_1\mathbf{e}_1 + \mathbf{B}[\tilde{\alpha}\sigma(\mathbf{Wv}) + \alpha^*\dot{\sigma}(\mathbf{Wv})\tilde{\mathbf{W}}\mathbf{v} + \alpha^*\sigma_0(\mathbf{Wv}) - \varepsilon] \tag{24}$$

Now, consider the following positive definite Lyapunov function,

$$V = 1/2[\mathbf{e}_1^{\mathrm{T}}\mathbf{P}\mathbf{e}_1 + \mathrm{tr}(\tilde{\alpha}^{\mathrm{T}}\mathbf{F}_1\tilde{\alpha}) + \mathrm{tr}(\tilde{\mathbf{W}}^{\mathrm{T}}\mathbf{F}_2\tilde{\mathbf{W}})] \tag{25}$$

where $\mathbf{F}_1$ and $\mathbf{F}_2$ are constant matrices that satisfy $\mathbf{F}_1 = \mathbf{F}_1^{\mathrm{T}} > 0$ and $\mathbf{F}_2 = \mathbf{F}_2^{\mathrm{T}} > 0$. The matrix $\mathbf{P}$ is a positive-definite symmetric solution obtained from, $\mathbf{A}_1^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{A}_1 = -\mathbf{Q}$ where $\mathbf{Q}$ is a positive definite matrix. The derivative of the Lyapunov function can then be written as

$$\mathrm{Term1} = -\mathbf{e}_1^{\mathrm{T}}\mathbf{Q}\mathbf{e}_1 - \mathbf{e}_1^{\mathrm{T}}\mathbf{P}\mathbf{B}\varepsilon + \mathbf{e}_1^{\mathrm{T}}\mathbf{P}\mathbf{B}\alpha\sigma_0(\mathbf{Wv})$$

$$\mathrm{Term2} = \mathrm{tr}\,\tilde{\mathbf{W}}(\mathbf{v}\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B}\boldsymbol{\alpha}\dot{\sigma} + \mathbf{F}_2\dot{\tilde{\mathbf{W}}}^{\mathrm{T}}) + \mathrm{tr}\,\tilde{\alpha}(\sigma\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B} + \mathbf{F}_1\dot{\tilde{\alpha}}^{\mathrm{T}})$$

$$\dot{V} = \mathrm{Term1} + \mathrm{Term2} \tag{26}$$

Eq. (26) allows us to derive the parameter update laws as follows. Let us assume $\mathbf{F}_2\dot{\tilde{\mathbf{W}}}^{\mathrm{T}} = -\mathbf{v}\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B}\boldsymbol{\alpha}\dot{\sigma}$ and $\mathbf{F}_1\dot{\tilde{\alpha}}^{\mathrm{T}} = -\sigma e_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B}$

Then Eq. (26) reduces to

$$\dot{V} = -\mathbf{e}_1^{\mathrm{T}}\mathbf{Q}\mathbf{e}_1 - \mathbf{e}_1^{\mathrm{T}}\mathbf{P}\mathbf{B}\varepsilon + \mathbf{e}_1^{\mathrm{T}}\mathbf{P}\mathbf{B}\alpha\sigma_0(\mathbf{Wv}) \tag{27}$$

Using standard inequalities and norms, one can re-write Eq. (27) as

$$\begin{aligned}\dot{V} \leqslant {} & \|\mathbf{e}_1\|\lambda_{\min}(\mathbf{Q})\|\mathbf{e}_1\| + \|\mathbf{e}_1\|\lambda_{\max}(\mathbf{P})\|\mathbf{B}\|_{\mathbf{F}}\varepsilon_{\mathrm{M}} \\ & + \|\mathbf{e}_1\|\lambda_{\max}(\mathbf{P})\|\mathbf{B}\|_{\mathbf{F}}\|\alpha\|\delta\end{aligned} \tag{28}$$

where δ is the upper bound magnitude of the higher order terms in Eq. (23). The ideal parameters are constant $(\alpha^*, \mathbf{W}^*)$ for a given condition. Hence, their derivatives are zero. The adaptive update laws for the parameters can be written as

$$\dot{\alpha} = -\mathbf{F}_1^{-1}\sigma\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B} \tag{29}$$

$$\dot{\mathbf{W}} = -\mathbf{F}_2^{-1}\mathbf{v}\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B}\alpha\dot{\sigma} \tag{30}$$

Using the dead zone in tuning rule as proposed in Fabri and Kadirkamanathan (1996) and the dead zone provides boundedness of the controller weights.

$$\begin{aligned}\dot{\alpha} &= -\mathbf{F}_1^{-1}\sigma\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B} + \alpha \\ &= \alpha \quad \text{if } e \geqslant E_a \quad \text{and} \quad |\alpha| \leqslant \alpha_a\end{aligned}$$

$$\begin{aligned}\dot{\mathbf{W}} &= -\mathbf{F}_2^{-1}\mathbf{v}\mathbf{e}_1^{\mathrm{T}}\mathbf{P}^{\mathrm{T}}\mathbf{B}\alpha\dot{\sigma} + \mathbf{W} \\ &= \mathbf{W} \quad \text{if } e \geqslant E_a \quad \text{and} \quad |\mathbf{W}| \leqslant \mathbf{W}_a\end{aligned}$$

where α_a and $\mathbf{W}_a$ are upper bound on the weights.

The error bound condition ensures the negative definiteness of the Lyapunov derivative in Eq. (26) can be written as

$$\|\mathbf{e}_1\| \geqslant \frac{\lambda_{\max}(\mathbf{P}\|\mathbf{B}\|_{\mathrm{F}})(\varepsilon_{\mathrm{M}} + \|\alpha\|\delta)}{\lambda_{\min}(\mathbf{Q})} \tag{31}$$

The nonlinear functions used in the controller are smooth and continuous with respect to their arguments. Since δ represents higher-order terms in the Taylor series expansion, they can be assumed to be small, and hence neglected. To prevent large parameter errors, the parameters are estimated off-line using finite time samples generated from an approximate model. In this approach, the controller parameters are initialized using off-line training with finite time sampling, instead of using random values. The random selection might leads to parameter drift if the error bound is not within the limit. This off-line finite time sample training is required to bring the error within the bound E_a such that the controller parameters does not drift. Also, the initialization is done once and the controller parameters are adapted online to accommodate the variations due to nonlinearity, and so on.

The estimated parameters so obtained are used as a starting point for the on-line adaptation. This strategy is called off-line finite time training and online adaptation. Hence, the error bound condition is reduced to

$$\|\mathbf{e}_1\| \geqslant \frac{\lambda_{\max}(\mathbf{P}\|\mathbf{B}\|_{\mathrm{F}})(\varepsilon_{\mathrm{M}})}{\lambda_{\min}(\mathbf{Q})} = \mathbf{E}_a \tag{32}$$

If there is no modelling error (i.e., $\varepsilon_{\mathrm{M}} = 0$), from Eq. (32), $\dot{V}$ is negative definite; hence the stability of overall system is guaranteed. However, in the presence of modelling error and $\|\mathbf{e}_1\| < E_a$ it is possible that the first derivative of Lyapunov function is greater than zero ($\dot{V} > 0$), which implies that the controller parameters may drift to infinity with time. To avoid such parameter drifts, the controller parameters are confined to a bounded set (as stated in Eq. (13)).

The proposed approach can handle the sudden failure in helicopter. In helicopter, even under failure, the objective is to follow the reference response. Unlike in other systems, the reference model is constant and the aircraft should follow the reference trajectory even under failures (Pashilkar et al., 2006). Under these circumstances, the assumption on constant ideal weights are acceptable and hence, the proposed adaptive control scheme can handle the sudden failures.

3. Feedback error learning neural control scheme

In this section, we present the feedback error learning neural control scheme (FENC) for an unstable helicopter. The block diagram of the FENC scheme is shown in Fig. 3. In the FENC scheme (Li et al., 2001), the conventional SFC in the inner loop is used to stabilize the helicopter and the neural controller in the outer loop approximates the unknown nonlinearity and provides the necessary tracking performance. The neural controller is trained to minimize the deviation between the reference signal and actual output of the helicopter. The control effort applied to the helicopter is the sum of the conventional and neural controller signals,

$$\mathbf{u}_{\mathrm{fenc}}(t) = \mathbf{u}_{\mathrm{nn}}(t) + \mathbf{u}_{\mathrm{con}}(t) + \mathbf{r}(t) \tag{33}$$

where u_{nn} is neural network output and u_{con} is the control input from the SFC.

The conventional controller is designed based on the linearized model at straight and level flight condition at a forward speed of 100 Kmph. The controller gain matrix $K = [K_u, K_v, K_w, K_p, K_q, K_r, K_\phi, K_\theta]$ where K is the feedback gain and the suffix indicates the state variable. The feedback gain is obtained using linear model analysis based on LQR method and validated by flight testing. The feedback gain matrix K is $[0, 0, 0, -0.047, -0.28, 0.69, -0.35, -0.49]$. The conventional controller is able to stabilize the helicopter at various level flight conditions using gain scheduling. A neural network with linear filter architecture is used to approximate the unknown nonlinearity. The stability and convergence of above approach is discussed in Li et al. (2001).

4. Simulation study

The objective of the neuro-controller design is to obtain stable dynamics with good tracking ability. The performance of the neuro-controller is evaluated for pitch rate (q) command under nominal and fault conditions. A brief description of helicopter dynamics and the reference model is given in the following sections.

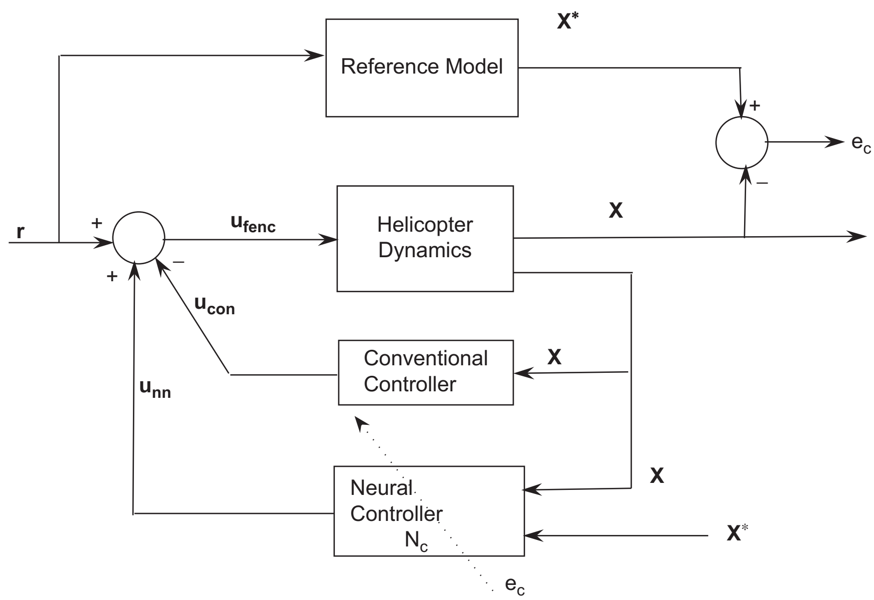

Fig. 3. Schematic diagram of FENC scheme.

4.1. Helicopter model

A helicopter having a soft inplane four-bladed hingeless main rotor and a four-bladed tail rotor with conventional mechanical controls is used for the simulation studies. The mathematical model used in this work was derived from a simplified generic non-linear helicopter model (Padfield, 1996, 1981). The main rotor is modelled using a center-spring approximation. The tail rotor is modelled by a coning disk. The fuselage aerodynamics are represented by a combination of look-up tables and polynomial functions of incidence and sideslip angles. The airfoil surfaces are represented using a constant aerodynamic lift co-efficient, assuming a two-dimensional flow. The main and tail rotor airfoils have a simple drag model comprising zero-lift and lift dependent coefficients. The forces and moment components contributed by each subsystem of the helicopter are calculated and transformed to the helicopter center of gravity reference system. The detailed assumptions and simplifications in force and moment calculations and 6-dof equations are discussed in Padfield (1981) and Padfield (1996).

The nonlinear model is linearized about the trim condition at hover, low speeds and straight and level flight at various speed conditions using analytical methods. The linearized helicopter dynamics can be written in the state space form as

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + Bu(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t) \tag{34}$$

where $\mathbf{x} = [u, v, w, p, q, r, \phi, \theta]^{\mathrm{T}}$ and $\mathbf{u} = [\delta_{\mathrm{col}}, \delta_{\mathrm{lat}}, \delta_{\mathrm{long}}, \delta_{\mathrm{dir}}]^{\mathrm{T}}$ where δ_{col} is the collective input, δ_{lat} is the lateral cyclic input, δ_{long} is the longitudinal cyclic input and δ_{dir} is the pedal input. The system at 100 Kmph is unstable and has one complex conjugate pole $(0.033 \pm 0.35i)$ corresponding to the pitch axis. The system is also unstable in the roll and directional axis. For the purpose of simulation, the continuous time linear model at 100 Kmph is converted to discrete time model using zero-order-hold with 20 ms sampling period. The system matrices are given in the Appendix.

4.2. Handling qualities requirements

The qualities or characteristics of a helicopter that govern the ease and precision with which a pilot is able to perform the tasks required to support the helicopter mission are known as handling qualities. Handling qualities encompass all aspects of the man–machine interface. This includes the cockpit ergonomics, the choice of inceptor and display, the presentation and update rate of the display for the digital systems, the feel or force feedback from the stick, the field of view, the available autopilot functions and response types and the vehicle response to small, moderate and large amplitude inputs.

It is customary to rate handling qualities in terms of Cooper–Harper Levels (Cooper and Harper, 1969). A system is Level 1 if essentially it is satisfactory without improvement. Level 2 implies that improvement is warranted. With a Level 2 system, adequate performance is attainable, though the pilot may face very objectionable deficiencies. A Level 3 system is at least controllable but has major deficiencies, possibly requiring intense pilot compensation. The three main levels are further subdivided into handling quality ratings (HQRs), which range from 1 to 10. HQRs of 1–3.5 equate overall to Level 1. HQRs of 3.5–6.5 equate to Level 2. HQRs of 6.5–9 equate to Level 3. For details and a discussion of pilot ratings, the reader is referred to Padfield (1996) and Cooper and Harper (1969). The metrics presented in the handling qualities specification Aeronautical Design Standard ADS-33 (Aeronautical Design Standard, 1994) is used for the design of controller.

4.3. Reference model

For helicopter control, the most commonly used command control systems for pitch axis is attitude command attitude hold (ACAH). Also, command filters for ACAH is designed such that the command filter response satisfies the ADS-33 handling quality specifications. The command filter serves as a model for desired response when helicopter is subjected to the pilot input signal.

4.4. Attitude command and attitude hold system

For ACAH system in the longitudinal axis, the ADS-33 handling quality specifications require the following characteristics for the helicopter. (a) Pitch attitude shall return to within 10% of peak or 1°, whichever is greater, following a pulse input, in less than 10 s.

(b) A step input to longitudinal cyclic shall produce a proportional pitch attitude change within 6 s. (c) The pitch attitude shall remain constant between 6 and 12 s following the pilot input.

The command filter is designed such that it satisfies the attitude quickness criterion (ratio of peak pitch rate to change in pitch attitude $q_{pk}/\Delta\theta_{pk}$ for Level 1 flying quality requirement). The damping ratio (τ) of the filter is between 0.6 and 0.8 and the natural frequency is 5 rad/s.

The pilot longitudinal cyclic deflection (δ_{long}) versus pitch rate command (q_{com}) serves as the reference model. This transfer function is selected as

$$\frac{q_{\text{com}}}{\delta_{\text{long}}} = \frac{64s}{s^3 + 11.4s^2 + 74.4s + 64} \tag{35}$$

As we can see from the above transfer function, the reference model satisfies Level 1 handling quality requirements and hence it is acceptable.

5. Simulation results and discussions

The schematic diagram of the DANC scheme for pitch rate command control is given in Fig. 4. Since the simulation is carried out for pitch rate command track controller, the roll and directional axis is stabilized using state feedback control as shown in Fig. 4. No damping is provided in the collective axis in this simulation.

The initial network weights are selected randomly (close to zero). The network parameters are estimated off-line using finite time samples generated from reference model. In order to train off-line, a sequence of training data is generated by feeding input signals to the reference model described above. The input signals are a series of step inputs of $[0, 1, -2, 1, -1, 0]$ with varied time step and amplitude -5 to 1 in steps of one. A test data are also generated with a input sequence different from training data. The response of the system is obtained with various initial conditions. The network is trained using these sequence until the error is less than a small value (ε) and then tested with a test data sequence. The estimated network parameters so obtained are used as a starting point for the on-line adaptation.

These data are used to train the controller network off-line for 100 Kmph forward speed with a learning rate of 0.05. The neural network has one input layer, a single hidden layer and one output layer. The controller network architecture is $N^{11,45,1}$, where 11-input nodes (1 present input, 2 past input and 4 past pitch rate and attitude), 45-hidden nodes and 1-output nodes.

The performance capability of the neural control scheme is tested with a reference pulse input of 0.05 rad.

The performances of the controllers (FENC and DANC) are measured using mean square error (MSE) in pitch rate, maximum control deflection in longitudinal cyclic and control effort (area under δ_{long} curve) and are given in Table 1. A low MSE in pitch rate indicates that both controllers track the command reasonably well. From Table 1 we observe that MSE are small for both FENC and DANC schemes and tracking of pitch rate is good. From Table 1 we can clearly observe that the performance of the DANC scheme in terms of control effort is better than the FENC scheme. The control effort in FENC scheme is higher due to the fact that the outer loop track control (neural) has to fight against and overcome the inner loop conventional stability augmentation. The control effort in DANC is less as there is no inner loop conventional control in this scheme.

The longitudinal cyclic displacement, the reference pitch rate command and the pitch rate responses and pitch attitude response of the helicopter for the FENC and DANC schemes are given in Fig. 5 under nominal condition. From this figure, we can see that the controllers track the command reasonably well.

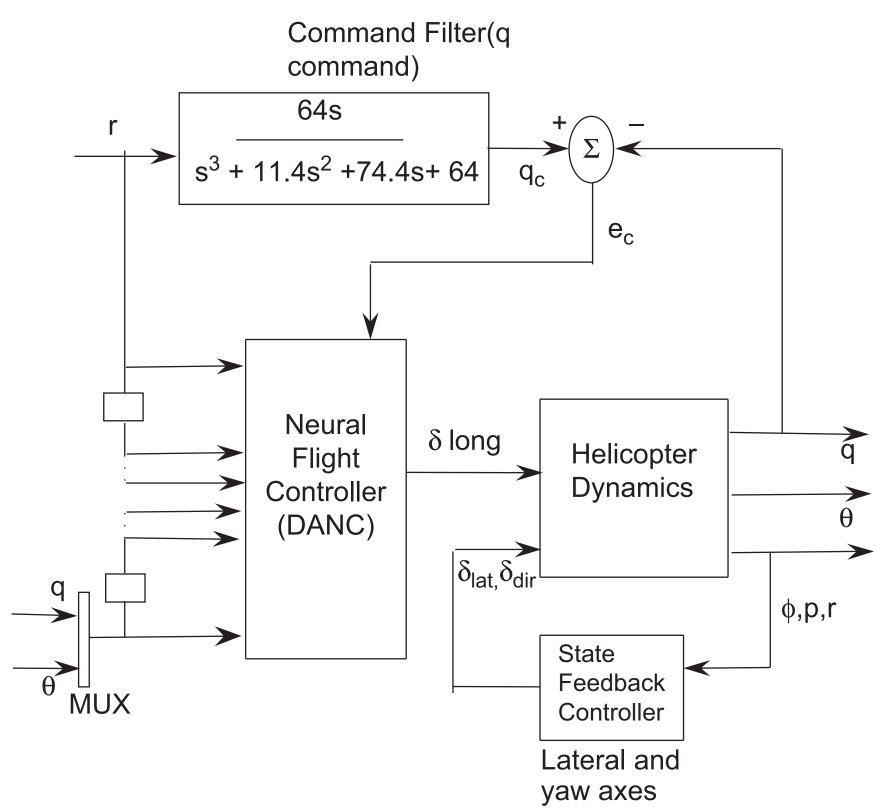

Fig. 4. Schematic diagram of DANC implementation scheme.

Table 1
Performance measures

Condition	Models	Forward speed (Kmph)	FENC			DANC		
			MSE q (deg/s)	Max. δ_{long} (deg)	Control effort	MSE q	Max. δ_{long} (deg/s)	Control effort (deg)
Trim I	Nominal	100	0.014	1.42	0.30	0.016	0.51	0.08
	AD 150%	100	0.029	2.19	0.54	0.034	0.56	0.17
	CS 50%	100	0.024	1.53	0.33	0.027	0.01	0.14
Trim II	High cruise	290	0.017	1.37	0.29	0.020	0.32	0.092

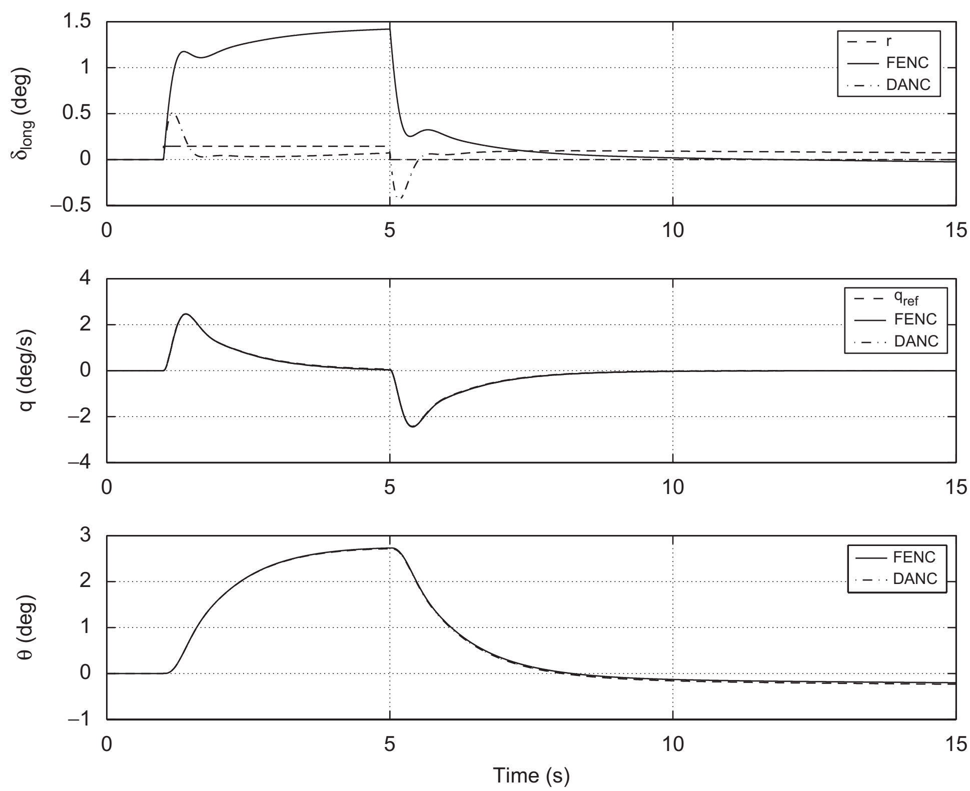

Fig. 5. Response of the helicopter under nominal condition.

The controller which is thus trained at 100 Kmph is presented with a helicopter system trimmed at 290 Kmph. After on-line training, the neuro-controller provides track control and the results are given in Table 1 for FENC and DANC, respectively. We use two metric to measure the performance. The tacking performance with respect to the reference signal and the control effort which is the area under the control surface deflection. From Table 1, we observe that both FENC and DANC adapt to the new trim condition and provide good tracking performance. We observe that control effort of DANC is marginally higher at 290 Kmph, when compared to the control effort at 100 Kmph.

5.1. Aerodynamic and control surface faults

Though linear models serve as valuable system representations for controller design, they suffer from modeling error which might prove critical in actual flight conditions. We examine the effectiveness of the controller performance in the face of aerodynamic fault/uncertainty (AD) and the control surface loss (CS). The on-line learning ability of the neural network is then verified for this perturbed system. The neural controller is trained on-line until the difference between the helicopter response and command signal is less than a certain prescribed value. During this period a pseudo-random pulse input signal is applied as the reference input. The reference input and command signal are sampled at 0.02 s duration. After on-line adaptation of the controller weight matrix, same pulse input signal is used to test the neuro-controller.

5.1.1. Aerodynamic fault (AD)

The performance of the FENC and DANC in the presence of 150% change in the A matrix is studied. After on-line adaptation as explained above, the controller is tested with the same pulse input signal. The performance measures are given in Table 1. The MSE between the command and q response is 0.029 for the FENC. Maximum longitudinal cyclic deflection for the FENC is 2.19° as compared to 1.42° in case of the nominal condition. The control effort is also increased by 176% to account for the aerodynamic uncertainty.

The MSE between the command and q response is 0.034 for the DANC. Maximum longitudinal cyclic deflection for the DANC is 0.56° as compared to 0.51° in case of the nominal condition. The control effort is also increased by 225% to account for the aerodynamic fault.

It is clear from the performance measures for the FENC and DANC that both controllers are able to adapt to the AD fault. The response of the system for AD is shown in Fig. 6. We can observe from the figure that the helicopter follows the pitch rate command well and the control effort required is with in the limit. In terms of the maximum control deflection and control effort, DANC does better than FENC.

5.1.2. Control surface fault (CS)

The performance of the FENC and DANC in the presence of 50% change in the B matrix is studied and the performance measures are given in Table 1.

The MSE between the command and q response is 0.024 for the FENC. Maximum longitudinal cyclic deflection for the FENC is 1.53° as compared to 1.42° in case of nominal condition. The control effort is also increased by 107% to account for the control surface faults.

The MSE between the command and q response is 0.027 for the DANC. Maximum longitudinal cyclic deflection for the DANC is 1.01° as compared to 0.51° in case of the nominal condition. The control effort is also increased by 198% to account for the control surface faults.

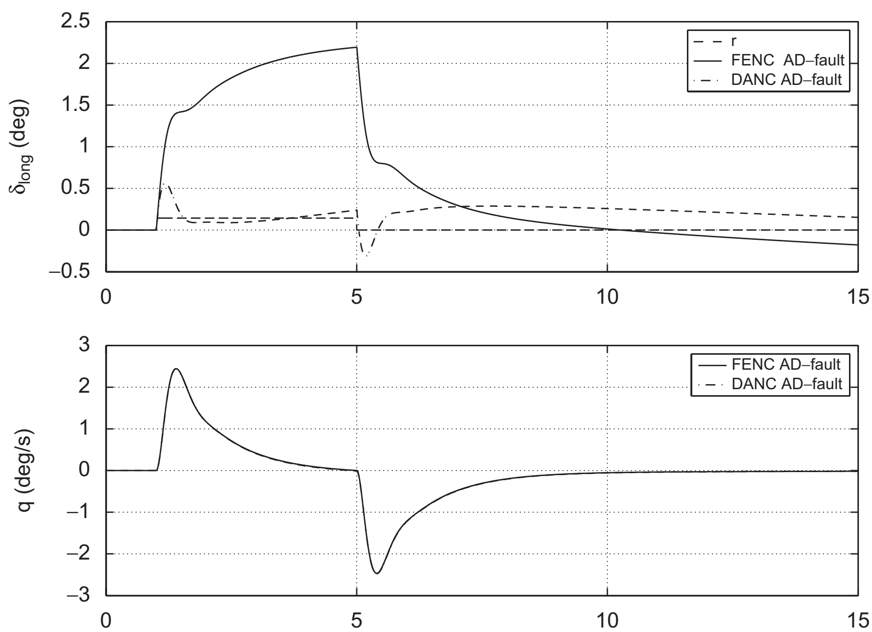

Fig. 6. Response of the helicopter under aerodynamic fault.

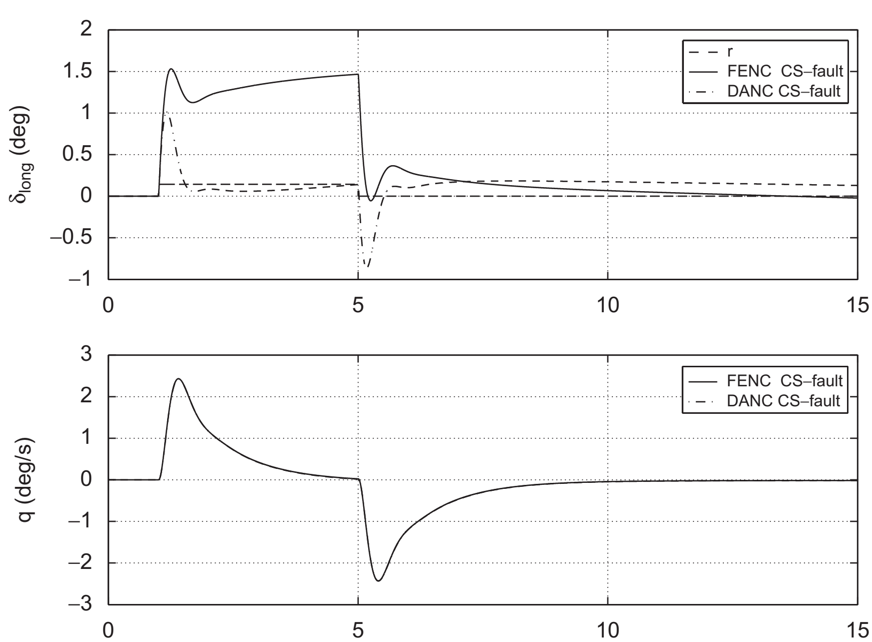

Fig. 7. Response of the helicopter under control surface fault.

It is clear from the performance measures for the FENC and DANC that both controllers are able to adapt to the control surface faults. The response of the system for CS fault is shown in Fig. 7. We can observe from these performance measures that the helicopter follows the pitch rate command well and the control effort required is with in the limit. In terms of the maximum control deflection and control effort, DANC does better than FENC.

These results demonstrate the on-line learning ability of the neural network for adapting to parameter uncertainty.

Results using the nonlinear helicopter model for simulation studies is not presented in this paper. This could be the subject for further investigation and will be presented in future paper. Comparisons with other neural networks based controller are under study and will be subject for further investigation.

6. Conclusions

In this paper, the feasibility of using a neural network based controller for an unstable helicopter system is presented. A neural network with linear filter architecture and backpropagation through time learning algorithm is used to design the neuro-controller for a helicopter model. A direct adaptive neural control (DANC) using off-line (finite interval of time) and on-line learning strategy is used to stabilize the helicopter and track the pitch rate command signal. The on-line learning ability is demonstrated through parameter uncertainties. The controller network parameters are adapted using updated rules Lyapunov synthesis. The laws to update the controller parameters have been derived. The performance of this neuro-controller is compared with the feedback error learning neural controller. The proposed control strategy using neuro-controller exhibits good tracking performance for an unstable helicopter system.

Appendix

The system matrices A, B, C and D matrices are given below for continuous time domain for straight and level flight at forward speed of 100 Kmph.

$$[A] = \begin{bmatrix} -0.038 & -0.22E-02 & 0.96 & 0.15E-02 & -0.024 & -0.84E-03 & 0.0 & -0.17 \\ -0.59E-02 & -0.16 & -0.28E-02 & 0.022 & -0.67E-02 & -0.47 & 0.17 & 0.19E-03 \\ 0.1 & -0.038 & -1.2 & -0.57E-02 & 0.49 & 0.012 & 0.27E-02 & -0.012 \\ -0.15E+01 & -0.74E+01 & 0.33E+01 & -0.74E+01 & -0.74 & -0.38 & -0.49 & 0.0 \\ 0.12E+01 & -0.51 & 0.13E+01 & -0.10 & -0.22E+01 & -0.25E-01 & 0.0 & 0.0 \\ 0.15E & 0.50E+01 & -0.16E+01 & -0.17E+01 & 0.11 & -0.13E+01 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.1E+01 & -0.12E-02 & 0.73E-01 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.99 & 0.16E-01 & 0.0 & 0.0 \end{bmatrix}$$

$$[B] = \begin{bmatrix} 0.19E+00 & -0.11E+00 & -0.11E+00 & 0.47E-01 \\ -0.29E-01 & 0.15E+00 & 0.13E-01 & 0.19E+00 \\ -0.28E+01 & 0.21E-01 & -0.62E+00 & 0.21E-01 \\ 0.10E+02 & 0.95E+02 & -0.32E+01 & 0.52E+01 \\ 0.12E+02 & 0.54E+01 & 0.25E+02 & -0.63E+00 \\ 0.67E+01 & 0.23E+02 & -0.35E+01 & -0.16E+02 \\ 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$[x] = [u\ \ v\ \ w\ \ p\ \ q\ \ r\ \ \phi\ \ \theta]$$

$$[C] = \mathrm{diag}[0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1]$$

$$[D] = \mathrm{diag}[0]$$

References

Aeronautical Design Standard, 1994. Handling Qualities Requirements for Military Rotorcraft. ADS-33D, U.S. Army, St. Louis, MO.

Anon, 1990. Special issue on neural networks in control systems. IEEE Control Systems Magazine 10 (3), 433–456.

Apkarian, P., Gahinet, P., Becker, G., 1995. Self-scheduled H_∞ control of linear parameter-varying systems: a design example. Automatica 31 (9), 1251–1261.

Calise, A.J., Rysdyk, R.T., 1998. Nonlinear adaptive flight control using neural networks. IEEE Control Systems Magazine 18 (6), 14–25.

Castillo, O., Melin, P., 2003. Intelligent adaptive model-based control of robotic dynamic systems with a hybrid fuzzy-neural approach. Applied Soft Computing 3 (4), 363–378.

Cooper, G.E., Harper, R.P., 1969. The use of pilot rating in the evaluation of aircraft handling qualities. National Aeronautics and Space Administration Technical Note TN D-5153, Washington, DC.

Fabri, S., Kadirkamanathan, V., 1996. Dynamic structure neural networks for stable adaptive control of nonlinear systems. IEEE Transactions on Neural Networks 7 (5), 1151–1167.

Faller, W.E., Schreck, S.J., 2000. Neural networks: applications and opportunities in aeronautics. Progress in Aerospace Sciences 32, 433–456.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Kiong, L.C., Rajeshwari, M., Rao, M.V.C., 2003. Neural networks: applications and opportunities in aeronautics. Applied Soft Computing 3 (3), 237–257.

Leith, D.J., Leithead, W.E., 2000. Survey of gain-scheduling analysis and design. International Journal of Control 73 (11), 1001–1025.

Leitner, J., Calise, A.J., Prasad, J.V.R., 1997. Analysis of adaptive neural networks for helicopter flight controls. AIAA Journal of Guidance, Control, and Dynamics 20 (5), 972–979.

Li, Y., Sundararajan, N., Sarachandran, P., 2001. Stable neuro flight controller using fully tuned radial basis function neural networks. Journal of Guidance, Control and Dynamics 24 (4), 665–674.

Melin, P., Castillo, O., 2003. Adaptive intelligent control of aircraft systems with a hybrid approach combining neural networks, fuzzy logic and fractal theory. Applied Soft Computing 3 (4), 353–362.

Narendra, K.S., 1996. Neural networks for control: theory and practice. Proceedings of the IEEE 84 (10), 1385–1406.

Narendra, K.S., Mukhopadhyay, S., 1992. Intelligent control using neural networks. IEEE Control Systems Magazine 5555, 11–18.

Narendra, K.S., Parthasarathy, K., 1990. Identification and control of dynamical system using neural networks. IEEE Transactions on Neural Networks 1 (1), 4–27.

Ng, G.W., 1997. Application of Neural Networks to Adaptive Control of Nonlinear Systems. Research Studies Press, Tauton, UK.

Padfield, G.D., 1981. A theoretical model of helicopter flight mechanics for application to piloted simulation. Royal Aircraft Establishment Technical Report RAE TR 81048 ADS-33D, Farnborough.

Padfield, G.D., 1996. Helicopter Flight Dynamics. Blackwell Science Limited, Oxford, UK.

Pashilkar, A.A., Sundararajan, N., Saratchandran, P., 2006. A fault-tolerant neural aided controller for aircraft auto-landing. Aerospace Science and Technology 10 (1), 49–61.

Peng, H., Yang, Z.J., Gui, W., Wu, M., Shioya, H., Nakano, K., 2007. Nonlinear system modeling and robust predictive control based on RBF-ARX model. Engineering Applications of Artificial Intelligence 20 (1), 1–9.

Pepijn, W.J., van de Ven, Flanagan, C., Toal, D., 2005. Neural network control of underwater vehicles. Engineering Applications of Artificial Intelligence 18 (5), 533–547.

Polycarpou, M.M., 1996. Stable adaptive neural control scheme for nonlinear systems. IEEE Transactions on Automatic Control 41 (3), 447–451.

Rovithakis, G.A., Christodoulou, M.A., 1994. Adaptive control of unknown plants using dynamical neural networks. IEEE Transactions on Systems, Man, and Cybernetics 24 (3), 400–412.

Roy, N., Ganguli, R., 2006. Filter design using radial basis function neural network and genetic algorithm for improved operational health monitoring. Applied Soft Computing 6 (2), 154–169.

Saade, C., Zinsch, F., Toulouse, E., 1996. Neural control of helicopter flight. Proceedings of the European Rotorcraft Forum 22, 107.1–107.11.

Suresh, S., Omkar, S.N., Mani, V., 2004. Nonlinear neural flight control system design. In: AIAA 1st Intelligence System Conference, Paper No: 2004-6219.

Tischler, M.B., 1996. Advances in Aircraft Flight Control. Taylor and Francis, London, UK.

Uraikul, V., Chan, C.W., Tontiwachwuthikul, P., 2007. Artificial intelligence for monitoring and supervisory control of process systems. Engineering Applications of Artificial Intelligence 20 (2), 115–131.

VijayaKumar, M., Omkar, S.N., Ganguli, R., Sampath, P., Suresh, S., 2006. Identification of helicopter dynamics using recurrent neural networks and flight data. Journal of the American Helicopter Society 51 (2), 164–174.

Wang, S.W., Yu, D.L., Gomm, J.B., Page, G.F., Douglas, S.S., 2006. Adaptive neural network model based predictive control for aircraft fuel ratio of SI engines. Engineering Applications of Artificial Intelligence 19 (2), 189–200.

Werbos, P.J., 1995. Neural networks and flight control: overview of capabilities and emerging applications. In: Proceedings of the AIAA Guidance, Navigation, and Control Conference, pp. 912–919.