# Designing an Adaptive Neural Network Based Controller for Automatic Landing Maneuver

M. Salem, M.A. Shahi Ashtiani, S. H. Sadati

**Abstract**— This paper introduces an adaptive controller based on Neural Networks (NN) and dynamic inversion use for a nonlinear six-degrees-of freedom of a modern aircraft in presence of wind patterns. Adaptive single hidden layer NN are employed for on-line learning and modeling error compensation. The approach utilizes an observer type adaptive neural network loop for an estimation of the correct aircraft model. The network weight adaptation rule is derived from Lyapunov stability analysis that guarantees boundedness of the NN weights and the system tracking errors. Performance is verified through numerical simulations. This paper deals the issue of aircraft landing maneuvers from the neural net-based adaptive control perspective. Generally this part of flight needs to be strongly assisted by human pilot. The results of simulation show that the designed controller can operate and satisfy the related task in landing maneuver.

**Index Terms**— Neural Network, Flight Control, Adaptive Control, Lyapanov Stability.

---

## 1 INTRODUCTION

THE development and application of most today systems and control theory where spurred on by the need to resolve aerospace problems. This is roughly the problem of analyzing and designing flight control systems for modern aircraft. The control laws used in current aircraft are mainly based on classical control design techniques. These control laws were developed in the 1950s and have evolved into fairly standard design procedures [1].

Current automatic flight control system design processes contain time and resource consuming trial-and-error approaches. Especially late changes in the flight control laws contribute to high cost and delay of first delivery. The automatic landing mode development is a good example of a process with trial-and-error design phases, because many parameters of the system (i.e. different runway, terrain, weather characteristics, and from aircraft uncertainties such as aerodynamic parameters, configuration, weight, thrust and actuator model) has to be robust against. The autonomous aircraft landing is an issue that implies three main aspects: a) the performance of equipment, b) the process models and c) the ethics. Generally, landing maneuver is not a standard flight task as it could be thinking, because it has high sensitivity versus environment perturbation and to the psychological factors. The other complementary problems are related of measurement precision of equipment and the reliability of the systems (hardware and software). The above-mentioned aspects reveal the complexity of the automatization of aircraft landing maneuver.

It is also known from the optimal control theory that a straightforward solution to the optimal trajectory shaping problem leads to a two point boundary-value problem [2], which is too complex for real-time onboard implementation. Traditional controller design usually involves complex and extensive mathematical analysis, which implies high cost and cannot guarantee a good performance level in the whole flight envelope. One of the best ways to solve this problem is to approach the artificial intelligence modeling technology based on fuzzy logic [3, 4] and neural network [5 to 7].

Intelligent control is a control technology that replaces the human mind in decisions making, planning control strategies, and learning new functions whenever the environment does not allow or does not the presence of a human operator. Artificial NN and fuzzy logic are two potential tools for use in applications in intelligent control engineering. Artificial NN offer the advantage of performance improvement through learning by means of parallel and distributed processing. Many NN control schemes with back propagation training algorithms, which have been proposed to solve the problems of identification and control of complex nonlinear systems, exploit the nonlinear mapping abilities of NN. Recently, adaptive NN algorithms have also been used to solve highly nonlinear flight control problems. The NN based approach incorporates direct adaptive control with dynamic inversion to provide consistent handling qualities without requiring extensive gain-scheduling or explicit system identification [8, 9]. This particular architecture use on-line learning NN, and reference models to specify desired handling qualities. On-line learning NN is used to compensate for errors and adapt to changes in aircraft dynamics and control allocation schemes.

This paper suggests a design paradigm by exploiting past results in the area of NN adaptive flight control system. This paper lies in a detailed application to design of adaptive command augmentation system that treats the six-degree-of-freedom nonlinear dynamics of aircraft landing maneuver including glide and flare. This paper is organized as follows. A nonlinear flight model is described in section 2. The dynamic Inversion Controller is described in detailed in section 3. The NN adaptive controller is then designed in section 4 when aerodynamic modeling error is present. A numerical simulation of a six-degree-of-freedom of modern aircraft such as F-18, is performed to verify the effectiveness of the proposed

- *PhD. Candidate, Dept of Aerospace Engineering, Space Research Institue , MUT, Islamic Republic of Iran, E-mail: mhm1120060@yahoo.com*
- *Assistant Professor, Space Research Institue, MUT, Islamic Republic of Iran.*
- *Assistant Professor, Space Research Institue, MUT, Islamic Republic of Iran.*

algorithm in section 5.

## 2 Aircraft Model

Aircraft landing maneuver enhanced several phases that define the so-called standard landing trajectory. The landing operation concerning two controlled maneuvers: first for guiding the aircraft in the horizontal plane (in order to align it onto the axe of runway) and the second, for aircraft guiding in the vertical plane (in order to do the approaching of runway).
Basically, the Automatic Landing Systems (ALS) provides the information for instrument navigation along the standard trajectory.However, the decisions in aircraft command should take by human pilot. The very high precise and reliable controller could be able to done this task, but human supervision on board is still required. With this restriction, the equations of motion describing the aircraft take the form [10]:

$$\dot{h} = H \qquad (1)$$

$$\dot{H} = b_0\dot{V} + b_1\dot{\beta} + b_2\dot{\alpha} + a_0 p + a_1 q + a_2 r \qquad (2)$$

where

$a_0 = b_4$ ; $a_1 = b_3\cos\phi + b_4\sin\phi\tan\theta$

$a_2 = b_4\cos\phi\tan\theta - b_3\sin\phi$

$b_0 = (\cos\beta\cos\alpha\sin\theta - \sin\beta\sin\phi\cos\theta - \cos\beta\sin\alpha\cos\phi\cos\theta]$

$b_1 = V(-\sin\beta\cos\alpha\sin\theta - \cos\beta\sin\phi\cos\theta + \sin\beta\sin\alpha\cos\phi\cos\theta]$

$b_2 = V(-\cos\beta\sin\alpha\sin\theta - \cos\beta\cos\alpha\cos\phi\cos\theta]$

$b_3 = V(\cos\beta\cos\alpha\cos\theta + \sin\beta\sin\phi\sin\theta + \cos\beta\sin\alpha\cos\phi\sin\theta]$

$b_4 = V(-\sin\beta\cos\phi\cos\theta + \cos\beta\sin\alpha\sin\phi\cos\theta]$

$$\dot{V} = c_0 + c_1 T \qquad (3)$$

and

$c_0 = \frac{1}{M}[-D\cos\beta + Y\sin\beta - M\,g\,(\sin\theta\cos\alpha\cos\beta -$
$\cos\theta\sin\phi\sin\beta - \cos\theta\cos\phi\sin\alpha\cos\beta]$

$c_1 = \frac{1}{M}(k_1\cos\alpha\cos\beta + k_3\sin\alpha\cos\beta)$

$$\dot{\alpha} = c_2 + c_3 T + c_4 p + c_5 q + c_6 r \qquad (4)$$

$c_2 = \frac{1}{M\,V\cos\beta}[-L + M\,g\,(\cos\theta\cos\phi\cos\alpha + \sin\theta\sin\alpha)]$

$c_3 = \frac{1}{M\,V\cos\beta}(k_3\cos\alpha - k_1\sin\alpha)$

$c_4 = -\cos\alpha\tan\beta$ ; $c_5 = 1$ ; $c_6 = -\tan\beta\sin\alpha$

$$\dot{\beta} = d_0 + d_1 T + d_2 p + d_3 r \qquad (5)$$

where

$d_0 = \frac{1}{M\,V}[D\sin\beta + Y\cos\beta + M\,g\,(\sin\theta\cos\alpha\sin\beta +$
$\cos\theta\sin\phi\cos\beta - \cos\theta\cos\phi\sin\alpha\sin\beta]$

$d_1 = \frac{1}{M\,V}(-k_1\cos\alpha\sin\beta - k_3\sin\alpha\sin\beta)$

$d_2 = \sin\alpha$ ; $d_3 = -\cos\alpha$

$$\dot{\phi} = d_4 p + d_5 q + d_6 r \qquad (6)$$

$$\dot{\theta} = d_7 q + d_8 r \qquad (7)$$

where

$d_4 = 1$ ; $d_5 = \sin\phi\tan\theta$ ; $d_6 = \cos\phi\tan\theta$ ;

$d_7 = \cos\phi$ ; $d_8 = -\sin\phi$

$$\dot{p} = \frac{I_z l_{aero} + I_{xz} n_{aero}}{I_x I_z - I_{xz}^2} + \frac{I_{xz}(I_x - I_y + I_z)pq + [I_z(I_y - I_z) - I_{xz}^2]qr}{I_x I_z - I_{xz}^2} \qquad (8)$$

$$\dot{q} = \frac{1}{I_y}\left[m_{aero} + pr(I_z - I_x) + I_{xz}(r^2 - p^2)\right] \qquad (9)$$

$$\dot{r} = \frac{I_{xz} l_{aero} + I_x n_{aero}}{I_x I_z - I_{xz}^2} + \frac{I_x(I_x - I_y) + I_{xz}^2]pq - I_{xz}(I_x - I_y + I_z)qr}{I_x I_z - I_{xz}^2} \qquad (10)$$

It is also assumed that the non-dimensional aerodynamic forces and moment coefficients are nonlinear parameterized with the angle of attack and sideslip angle, and linearly parameterized with control surface deflections and angular rates. This assumption results in nonlinear dynamic equations affine to the control variables. To make the ALS more intelligent, reliable wind profiles are necessary. Two spectral turbulence forms modeled by Dryden and von Karman are mostly used for aircraft response studies [10]. In this study the Dryden model was used for its demonstration ease.

## 3 Dynamic Inversion Cotrol Law

A multiple time scale approach for dynamic inversion has been developed for designing a flight control system that regulates $[h \;\; \phi \;\; \beta]$. It assumes that the state dynamics can be decomposed as follows [11]:

- Slow dynamics with state vector: $X_s = [V \;\; \alpha \;\; \beta \;\; \phi \;\; \theta \;\; h]^T$
- Fast dynamics with state vector: $X_f = [p \;\; q \;\; r]$

In reality the dynamics are not separable according to the above definitions. However, the inverting solution doesn't rely on a separation in dynamics to be valid. Therefore it may be more appropriate to say that the inversion is done in two stages. The overall structure of the inverting control law is displayed in Figure 1.

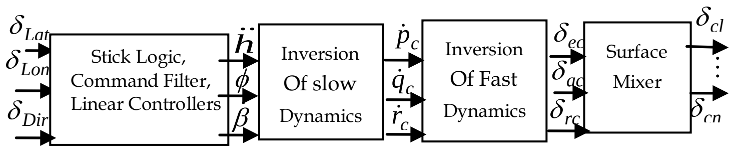


Fig.1. Dynamic Inversion Control Law Structure

In time scale, the equations of motion are expressed in form of $y=c.x$, where y defines the regulated output variables, and $u$ defines the control variables, which are the output variables of the inverting blocks in Figure 1. Note that the control variables for the fast dynamics are effective control displacement commands for roll, pitch and yaw axis:

$$u_f = [\delta_a \quad \delta_e \quad \delta_r]^T \qquad (11)$$

and the control variables for the slow dynamics are the commands to the fast dynamics

$$u_s = [\dot{p} \quad \dot{q} \quad \dot{r}]^T \qquad (12)$$

Assuming $C.b(x)$ is invertible, and then the inverting design in each time scale is based on:

$$\dot{y} = Ca(x) + Cb(x)u = v \qquad (13)$$

where $v$ is the so-called pseudo control. The pseudo control is a linear control law designed to regulate $y$, and corresponds to the inputs to each inverting block in Figure 1. The regulated variables in each time scale are:

$$y_s = [\ddot{h}, \dot{\phi}, \dot{\beta}]^T$$
$$y_f = [P, q, r]^T \quad (14)$$

The variables of interest in landing maneuver control problem can be grouped in to two sets, i.e., 1) $V, h, \alpha, \beta$ or 2) $V, h, \phi, \beta$ either of which can be tracked by the trajectory controller. The remaining task in the slow-time scale control problem is that of converting the three pseudo control into the three real controls. Thus, from equations 2, 5, 6, one has:

$$\begin{bmatrix} a_0 & a_2 & a_2 \\ d_4 & d_5 & d_6 \\ d_2 & 0 & d_3 \end{bmatrix} \begin{bmatrix} p_c \\ q_c \\ r_c \end{bmatrix} = \begin{bmatrix} \ddot{h} - f_h \\ \dot{\phi}_d - f_\phi \\ \dot{\beta}_d - f_\beta \end{bmatrix} \quad (15)$$

where

$$f_h = b_0\dot{V} + b_1\dot{\beta} + b_2\dot{\alpha} \quad ; \quad f_\phi = 0 \quad ; \quad f_\beta = d_0 + d_1 T$$

Once the equation 16 has been solved, the $p, q, r$ values can be substituted in equations 8 to 10 to compute the control surface deflections $\delta_a, \delta_e\, \delta_r$ along the "outer" solution. Three independent control loops can be designed in pseudo control $u_s$ such that equation 12 has a much faster time constant the slow-time scale system of equations 1 to 7. The real controls can be obtained from pseudo controls using

$$\begin{bmatrix} \dot{p}_d \\ \dot{q}_d \\ \dot{r}_d \end{bmatrix} = \begin{bmatrix} f_p(x) \\ f_q(x) \\ f_r(x) \end{bmatrix} + \begin{bmatrix} L_{\delta a} & 0 & N_{\delta a} \\ 0 & M_{\delta e} & 0 \\ L_{\delta r} & 0 & N_{\delta r} \end{bmatrix} \begin{bmatrix} \delta_{ac} \\ \delta_{ec} \\ \delta_{rc} \end{bmatrix} \quad (16)$$

where

$$\begin{bmatrix} f_p(\bar{x}) \\ f_q(\bar{x}) \\ f_r(\bar{x}) \end{bmatrix} = \begin{bmatrix} \dfrac{I_z\hat{l}_{aero} + I_{xz}\hat{n}_{aero}}{I_x I_z - I_{xz}^2} + \dfrac{I_{xz}(I_x - I_y + I_z)pq + [I_z(I_y - I_z) - I_{xz}^2]qr}{I_x I_z - I_{xz}^2} \\ \dfrac{1}{I_y}\left[\hat{m}_{aero} + pr(I_z - I_x) + I_{xz}(r^2 - p^2)\right] \\ \dfrac{I_{xz}\hat{l}_{aero} + I_x\hat{n}_{aero}}{I_x I_z - I_{xz}^2} + \dfrac{I_x(I_x - I_y) + I_{xz}^2]pq - I_{xz}(I_x - I_y + I_z)qr}{I_x I_z - I_{xz}^2} \end{bmatrix} \quad (17)$$

and

$$\hat{l}_{aero} = 1/2\rho V^2 Sb \begin{bmatrix} C_{l\beta}(\alpha)\beta + C_{lp}(\alpha)\dfrac{pb}{2V} + \\ C_{lr}(\alpha)\dfrac{rb}{2V} \end{bmatrix} \quad (18)$$

$$\hat{m}_{aero} = 1/2\rho V^2 S\bar{c}\left[C_m(\alpha)\beta + C_{mq}\frac{\bar{c}q}{2V}\right] \quad (19)$$

$$\hat{n}_{aero} = 1/2\rho V^2 S\bar{b}\left[C_{n_\beta}(\alpha)\beta + C_{n_p}(\alpha)\frac{pb}{2V} + C_{nr}\frac{rb}{2V}\right] \quad (20)$$

The set algebraic equation 16 can be solved for $\delta_a, \delta_e\, \delta_r$ .Note that the matrix multiplying these quantities has full rank everywhere on the flight envelope and a unique solution always exists.

## 4 Intelligent Flight Control System Components

Intelligent control achieves automation via the emulation of biological intelligence. Artificial NN are circuits, computer algorithms, or mathematical representations loosely inspired by the massively connected set of neurons that form biological NN. The application of NN has attracted significant attention in several disciplines, such as signal processing, identification and control. The success of NN is mainly attributed to their unique features:

(1) Parallel structures with distributed storage and processing of massive amounts of information.
(2) Learning ability made possible by adjusting the network interconnection weights and biases based on certain learning algorithms.

The first feature enables NN to process large amounts of dimensional information in real-time (e.g. matrix computations), hundreds of times faster than the numerically serial computation performed by a computer. The implication of the second feature is that the nonlinear dynamics of a system can be learned and identified directly by an artificial NN. The network can also adapt to changes in the environment and make decisions despite uncertainty in operating conditions.

Artifical NN constitute a promising new generation of information processing systems that demonstrate the ability to learn, recall, and generalize from training patterns or data. This specific feature offers the advantage of performance improvement for ill-defined flight dynamics through learning by means of parallel and distributed processing. Rapid adaptation to environment change makes them appropriate for guidance and control systems because they can cope with aerodynamic changes during flight. The feedforward multilayer perceptron is the most popular NN in control system applications and so we limit our discussion to it.

### 4.1 Adaptive System Linerization

One of the common methods for controlling nonlinear dynamical systems is based on approximate feedback linearization [12]. The form that is employed in each control channel depends on the relative degree of the controlled variable. To simplify our discussion, we assume that the system has full relative degree, where each controlled variable (element of the state vector x) has a relative degree of two

$$\ddot{x} = f(x, \dot{x}, \delta) \quad (22)$$

In the case of aircraft, typically $x, \delta \in R^n$, where the elements of x correspond to the roll, pitch and yaw attitude angles. A variant of this form arises in which angular rate is controlled. Here, the equation of motion for that degree of freedom is expressed in first order form. A pseudo-control $\nu(t) \in R^n$ is defined such that the dynamic relation between it and the system state is linear

$$\ddot{x} = \nu = f(x, \dot{x}, \delta) \quad (23)$$

If $f(...)$ is invertible and $x(t), \dot{x}(t)$ are measurable, equation 23 can provide the linearization for the control variable. Ideally, the actual controls $\delta$ are obtained by inverting equation 23. Since the function $f(x, \dot{x}, \delta)$ is not known exactly, an approximation is defined as:

$$\nu = \hat{f}(x, \dot{x}, \delta) \quad (24)$$

which results in

$$y^{(r)} = \ddot{x} = \nu + \Delta(x, \dot{x}, \delta) \quad (25)$$

where the modeling error is represented by:

$$\Delta(x,\dot{x},\delta) = f(x,\dot{x},\delta) - \hat{f}(x,\dot{x},\delta) \quad (26)$$

The approximation, f^, is chosen such that an inverse with respect to δ is computable. The actual control input to be computed by:

$$\delta = \hat{f}^{-1}(x,\dot{x},v) \quad (27)$$

### 4.2 Model Tracking Error Dynamics

The dynamic model inversion, and thus the nonlinear system linearization are in general not exact, mainly because the exact nonlinear model is not known or too complex to be implemented. Clearly, simplified inversion functions are advantageous from real-time implementation perspective and thus are often adopted when adequate feedback linearization error compensation is incorporated in the controller design. In this study, a nonlinear Multilayer Neural Networks (MNN) is used to compensate for the inversion error. The MNN was chosen because of its universal approximation property [13], and its effectiveness in relation to nonlinearly-parametered adaptive control (including nonlinearly parameterized NNs) has been demonstrated for flight control applications [9]. Because of the inversion error $\Delta$, tracking performance may be degraded severely with Proportional Differentiation (PD) control alone, and may be unstable. Thus an application of an additional control methodology like adaptive control component is needed to insure a specified level of tracking performance. The adaptive control structure for each channel is chosen as:

$$v = v_{rm} + v_{pd} - v_{ad} \quad (28)$$

where $v_{rm}$ is the pseudo-control component generated by the reference model, $v_{pd}$ is the output of the linear controller, $v_{ad}$ is generated by the adaptive element introduced to compensate for the model inversion error. A linear compensator is designed for each degree of freedom assuming perfect inversion ($\hat{f} = f$). The linear compensator is designed so that the error dynamics are stabilized. This is most often achieved using standard PD controllers, although additional integral action can be incorporated to improve steady state performance. The PD compensation is expressed by:

$$v_{pd} = \begin{bmatrix} K_P & K_D \end{bmatrix} e \quad (29)$$

where the tracking error vector is defined by:

$$e = \begin{bmatrix} x_{rm} - x \\ \dot{x}_{rm} - \dot{x} \end{bmatrix} \quad (30)$$

Therefore the proportional and derivative control gains are as follows:

$$K_d = 2\zeta\omega_n, \qquad K_p = \omega_n^2, \quad (31)$$

where $\zeta$ is the damping ratio, $\omega_n$ is the undamped natural frequency. By substituting (28) into (25) the error dynamics become:

$$\ddot{\tilde{x}} + K_d\dot{\tilde{x}} + K_p\tilde{x} = v_{ad} - \Delta \quad (32)$$

where $\tilde{x}(t) = x_c(t) - x(t)$ are the error states. The model tracking error dynamics are now found by:

$$\dot{e} = Ae + B[v_{ad} - \Delta] \quad (33)$$

where

$$A = \begin{bmatrix} 0 & I \\ -K_P & -K_D \end{bmatrix},\ B = \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (34)$$

are stable. It is evident from equation 33 that the role of the adaptive component, $v_{ad}$, is to cancel $\Delta$. The linear PD compensator gain matrices $K_P, K_D$ are chosen such that A is Hurwitz.

### 4.3 Neural Network for Inversion Error Compensation

The dynamic model inversion, and thus the nonlinear system linearization are in general not exact, mainly because the exact nonlinear model is not known or too complex to be implemented. Clearly, simplified inversion functions are advantageous from real-time implementation perspective and thus are often adopted when adequate feedback linearization error compensation is incorporated in the controller design. In this study, a nonlinear single hidden layer (SHL) NN is used to compensate for the inversion error. The SHLNN was chosen because of its universal approximation property [14]. For an input vector x, which is constructed of the measured states, the reference model outputs and the pseudo control signal, the output of the SHLNN is given by

$$v_{ad} = W^T\sigma(V^T\bar{x}) \quad (35)$$

where V and W are the input and output weighting matrices, respectively, and σ is a sigmoid activation function. The NN may be used to approximate a nonlinear function, such as $\Delta$. The universal approximation property [15] of NN's ensures that given an $\bar{\varepsilon} > 0$, then $\forall\ \bar{x} \in D$, where D is a compact set, there exists an $n_2$ and an ideal set of weights ($V^*, W^*$), that brings the output of the NN to within an $\varepsilon$-neighborhood of the function approximation error. This $\varepsilon$ is bounded by $\bar{\varepsilon}$ which is defined by

$$\sup_{\bar{x}\in D}\left\| W^T\sigma(V^T\bar{x}) - \Delta \right\| < \bar{\varepsilon} \quad (36)$$

The weights $V^*, W^*$ may be viewed as optimal values of (V, W) in the sense that they minimize $\bar{\varepsilon}$ on D. These values are not necessarily unique. The universal approximation property thus implies that if the NN inputs $x_{in}$ are chosen to reflect the functional dependency of $\Delta$, then $\bar{\varepsilon}$ may be made arbitrarily small given a sufficient number of hidden layer neurons, $n_2$. Although ideal weighting matrices are unknown and usually cannot be computed, they can be adapted in real time using the following NN weights training rules [14, 15]:

$$\dot{\hat{W}} = -\Gamma_w\left[(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\bar{x})\eta + k_w\|e\|\hat{W}\right] \quad (37)$$

$$\dot{\hat{V}} = -\Gamma_v\left[\bar{x}\eta\hat{W}^T\hat{\sigma}' + k_v\|e\|\hat{V}\right] \quad (38)$$

In the above, first terms of the right-hand sides of (37) and (38) are the modified backpropagation algorithms and the last terms correspond to a combination of $\sigma$-modification and $e$-modification used to improve the robustness in the presence of errors [18], $\Gamma_W$ and $\Gamma_V$ are positive definite learning rate matrices (typically chosen as a scalar times an identity matrix), $\sigma'$ is the partial derivative of the sigmoid $\sigma$ with respect to the NN inputs, and $\kappa$ is the known as $e$-modification gain.

Since $A$ is Hurwitz, there exists a unique and positive definite matrix $P > 0$ for an arbitrary matrix $Q > 0$ satisfying the Lyapunov equation:

$$A^T P + PA + Q = 0 \quad (39)$$

$A$ And $B$ in the above equations are the tracking error dynamics matrices defined in (34). $\eta$ is defined by

$$\eta = e^T PB \quad (40)$$

The following assumptions are used in the design and analysis of the adaptive control law presented in this paper.

**Assumption 1**: All command signals are bounded.

**Assumption 2**: The ideal weight matrices are bounded as: $\|Z\| \le \bar{Z}$, where $\|\ldots\|$ denotes the Frobenius norm of a matrix.

**Assumption 3**: A in Eq. (34) is Hurwitz.

Then we can state that the system in (22) together with the inverting controller in (27) subject to assumptions 1and 3, then all signals in the resulting closed loop system described in (33) and the NN adaptation rule in equation (37 to 38) remain bounded. The tracking error dynamics can be written as:

$$\dot{e} = Ae + B\left[\hat{W}\hat{\sigma}(\hat{V}^T\bar{x}) - W^*\sigma(V^{*T}\bar{x}) - \varepsilon\right] \quad (41)$$

where and $\varepsilon$ is the instantaneous residual network approximation error. Utilizing a Taylor-series expansion for the sigmoid with respect to $\hat{V}$, this can be rewritten as:

$$\dot{e} = Ae + B\begin{Bmatrix}\tilde{W}^T\left[\hat{\sigma}(\hat{V}^T\bar{x}) - \hat{\sigma}'(V^T\bar{x})\hat{V}^T\bar{x}\right] \\ +\hat{W}^T\hat{\sigma}'(V^T\bar{x})\tilde{V}^T\bar{x} - \varepsilon + w\end{Bmatrix} \quad (42)$$

where

$$w = \tilde{W}^T\hat{\sigma}'(V^T\bar{x})V^{*T}\bar{x} - W^{*T}\mathrm{O}(\tilde{V}^T\bar{x})^2 \quad (43)$$

and $\tilde{W} = W - W^*$ & $\tilde{V} = V - V^*$. The weights $V^*, W^*$ may be viewed as optimal values of (V, W) in the sense that they minimize $\varepsilon$ on D. The term $\mathrm{O}(\tilde{V}^T\bar{x})^2$ represents higher terms of the Taylor-series expansion. An upper bound on the norm of $w$ can be written as:

$$\|w - \varepsilon\| \le c_0 + c_1\|\tilde{Z}\| \quad (44)$$

where $c_0$, $c_1$ depends on the size of the NN and the assumed over-bound on the weights, $\bar{Z}$. Boundedness of weight error signals is shown employing a Lyapunov analysis, and then this result is used to show boundedness of the tracking error signals. By considering the following Lyapunov function candidate:

$$L = \frac{1}{2}\left(e^T pe + tr(\tilde{W}^T\Gamma_W^{-1}\tilde{W}) + tr(\tilde{V}^T\Gamma_V^{-1}\tilde{V})\right) \quad (45)$$

and differentiating (45) along (33), we have:

$$\dot{L} = -\frac{1}{2}e^T Qe + e^T pb(w - \varepsilon) - k\|e\| tr(\tilde{Z}^T\hat{Z}) \quad (46)$$

Using $2tr\{\tilde{Z}^T\hat{Z}\} = \|\tilde{Z}\|_F^2 + \|\hat{Z}\|_F^2 - \|Z\|_F^2 \ge \|\tilde{Z}\|_F^2 - \|Z\|_F^2$, we obtain

$$\dot{L} \le -\|e\|\left[\frac{1}{2}\lambda_{\min}(Q)\|e\| + \frac{k}{2}\|\tilde{Z}\|_F^2 - \frac{k}{2}\|Z\|_F^2 - c_0\|pb\| - c_1\|pb\|\|\tilde{Z}\|_F\right] \quad (47)$$

If we define $\bar{Z} = \frac{k}{2}\|Z\|_F^2$ then:

$$\dot{L} \le -\|e\|\left[\frac{1}{2}\lambda_{\min}(Q)\|e\| + \frac{k}{2}\|\tilde{Z}\|_F^2 - c_1\|pb\|\|\tilde{Z}\|_F - c_0\|pb\| - \bar{Z}\right] \quad (48)$$

By selecting $\lambda_{\min}(Q)$ and learning rates ($\Gamma_W, \Gamma_V$), $\dot{L} \le 0$ everywhere outside a compact set that is entirely within the largest level set of $L$, which in turn lies entirely within the compact set $D$. Either of the following two conditions renders $\dot{L} \le 0$.

$$\|e\| > \frac{2\bar{Z} + 2c_0\|pb\|}{\lambda_{\min}(Q)} = b_e \quad (49)$$

Or

$$\|\tilde{Z}\|_F > \frac{2c_1\|pb\|}{k} + \sqrt{\frac{2c_0\|pB\| + 2\bar{Z}}{k}} = b_Z \quad (50)$$

Thus for initial conditions within $D$, the tracking error e, and NN weights $\tilde{W}, \tilde{V}$ are uniformly ultimately bounded, with the tracking error bound given by (49) treated as an equality.

Process above, represents the design procedure of the adaptive controller to track the $h$, $\phi$, $\beta$ commands when the modeling errors exist. This shows that, if the controller is applied, the tracking errors and the parameter estimation error of the NN converge to a compact set and also shows that the size of the set is adjustable by tuning the design parameters.

## 5 Simulation Results and Disscussion

A nonlinear simulation of an F-18 aircraft [16] was used to investigate the proposed control methodology. The aircraft model consists of 6 degree of freedom kinematics, linearized aerodynamics, and linearized propulsion. Simulations are performed at sea level; airspeed of 230 ft/s, corresponding to the landing maneuver of the F-18 with applying the Dryden wind patterns (Figure 2), the performance of the controllers has been discussed. The primary (nominal) control system utilizes simple linear PD controller, while the secondary control systems use Adaptive Neural Network control systems. The simulation results are presented in Figures 3 to 10. The actual aircraft response is presented in Figure 2. Time histories of the controls are shown in Figures 3 and 4, which depict the flight speed variation, demonstrates that the engines can regulate slight speed until that is compromised for attitude rate control. The time response of roll, pitch, and yaw rates are show in Figure 5. Time histories of the angle of attack and pitch angle are shows in Figures 6 and 7. The bank angle and sideslip angle starts out and ends at zero pointing in the desired direction. The sideslip angle is presented in Figure 8. Neural Network adaptation signal $v_{ad}$ for compensate inversion error is presented in Figiure 9. A time history of the wind is show in Figure 10. Summarizing the results presented so far, the nonlinear controller performance for landing maneuver has been found very good.

## 6 Conclusion

Current developments of artificial intelligence enable an appropriate approach of high precision control tasks. This paper reveals some aspects of NN based adaptive control engineering. Our particular goal was to demonstrate the potential Adative Neural Network systems for high precision maneuvers required by aircraft landing. The proposed model reveals the functional aspect for realistic simulation data. The method does not require the existing controller to be designed based on a linear model. A SHLNN is used to approximate the interconnection effects and modeling errors on-line.

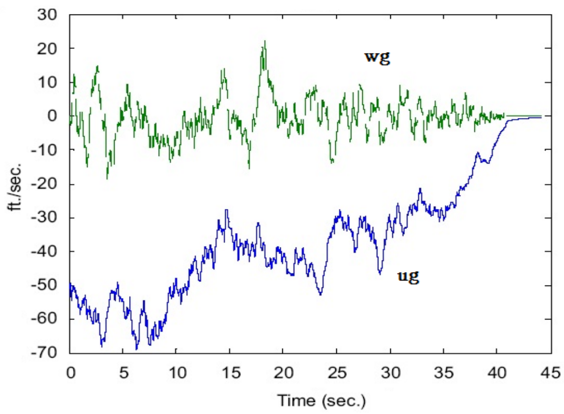


Fig.2. Wind turbulence Model

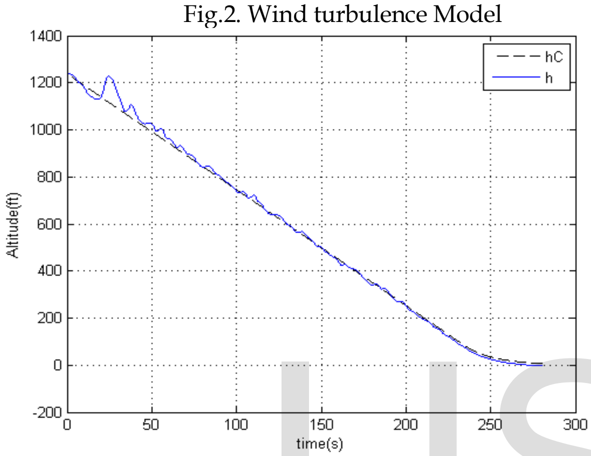


Fig.3 Time response of Desired Landing Trajectories

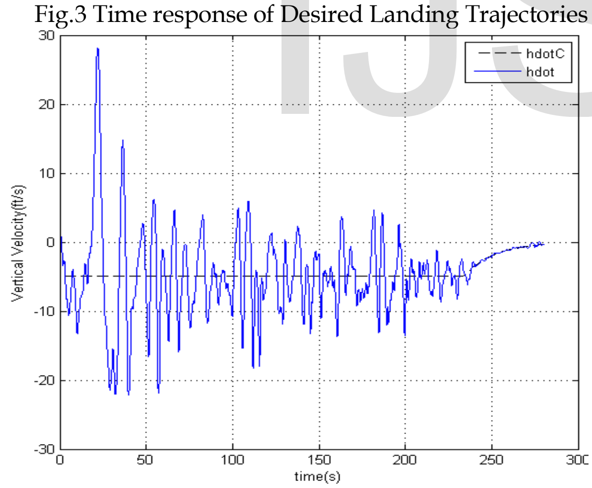


Fig. 4. Time response of the airspeed

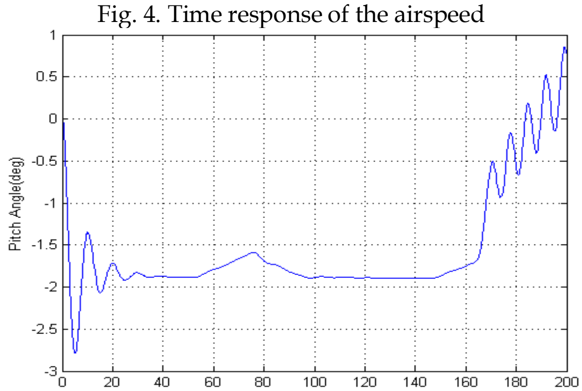


Fig.5 Time response of pitch,angle

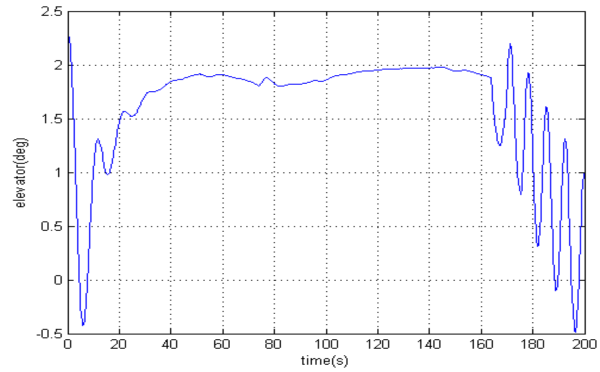


Fig.6. Time response of the elevator



## REFERENCES


[1] Ching-Fang Lin, "*Modern Navigation, Guidance and Control Processing*" Volume *II,* Prentice Hall, Englewood Cliffs, New Jersey 07632, 2002.

[2] Bryson, A. E., Jr. "*Applied Optimal Control*. Published by the press Syndicate of the University of Cambridge, 2002.

[3] Li-xin Wang, "*A Course in Fuzzy Systems and Control*", Prentice-Hall International Inc., 2006.

[4] Sefer Kurnaz, Okyay Kaynak, Omer Cetin, "*Fuzzy Logic Based Approach to Design of Flight Control and Navigation Tasks for Autonomous Unmanned Air Vehicles*", Journal Intell Robot Syst, 2009 (54:229-244).

[5] Rokhsaz, K., Steck, J.E., "*Use of Neural Network in Control of High-Alpha Maneuvers*" AIAA Journal of Guidance, Control, and Dynamics, Vol.16. 5. Sept-Oct. 1993 (934-939).

[6] Ryota Mori, Shinji Suzuki, "*Neural Network Analysis of Pilot Landing Control under Real Flight Condition*", Jan. 2008, Nevada, AIAA 2008-880.

[7] Waseem Aslam BUTT, Lin YAN, Amezquita S. Kendrick, "*Dynamic Surface Control for Nonlinear Hypersonic Air Vehicle using Neural Networks*", 29th Chinese Control Conference, July 2010, Beijing, China.

[8] Y.Q. Jin *et.al*, "*Adaptive Neural Networks Controller design for missile systems with Unmodeled Dynamics*", IMACS Conference, October 2006, Beijing, China.

[9] Fola Soares, "*Neural Network Application in Advanced Aircraft Flight Control System*", AIAA Conference, California, 2007-2943.

[10] J. Roskam, " *Airplane Flight Dynamics and Automatic Flight Controls, Part I, II*", Published by: Roskam Aviation and Engineering Corporation, 1986.

[11] Snell, S.A., Enns, D.F., and Garrard, W.L, Jr., "*Nonlinear Inversion Flight Control for a Super maneuverable Aircraft*" Journal of Guidance, Control and Dynamics, Vol, 15, No 4, 1992, pp 976-984.

[12] P. A. Ioannou and J. Sun, "*Robust Adaptive Control*". Englewood Cliffs, NJ: Prentice-Hall, 1995.

[13] HLewis, F.L., Yesildirek, A, and Liu, K "*Multilayer Neural-Net Robot Controller with Guaranteed Tracking Performance*," IEEE Trans. Neural Networks, vol. 7, no. 2, pp. 388–399, 1996.

[14] K. S. Narendra and A. M. Annaswamy, "*A new adaptive law for robust adaptation without persistent excitation*," IEEE Trans. Automat. Control, vol. AC-32, pp. 134–145, Feb. 1987.

[15] Sefer Kumaz, Omer Ceitn, Okyay Kaynak, "*Adaptive Neuo-Fuzzy inference system based on autonomous flight control of UAV*", Journal Expert Systems with Applications, 2010 (1229-1234).

[16] Fan, Y., Luze, H.F. and Cliff, M."*Time-Optimal Lateral Maneuvers of an Aircraft*" Journal of Guidance, Control and Dynamics, Vol 18, No.5, 1995, pp.1106-1112.