

Nonlinear Adaptive Flight Control Using Backstepping and Neural Networks Controller

Taeyoung Lee* and Youdan Kim†

Seoul National University, Seoul 151-742, Republic of Korea

A nonlinear adaptive flight control system is proposed using a backstepping and neural networks controller. The backstepping controller is used to stabilize all state variables simultaneously without the two-timescale assumption that separates the fast dynamics, involving the angular rates of the aircraft, from the slow dynamics, which includes angle of attack, sideslip angle, and bank angle. It is assumed that the aerodynamic coefficients include uncertainty, and an adaptive controller based on neural networks is used to compensate for the effect of the aerodynamic modeling error. Under mild assumptions on the aerodynamic uncertainties and nonlinearities, it is shown by the Lyapunov stability theorem that the tracking errors and the weights of neural networks exponentially converge to a compact set. Finally, nonlinear six-degree-of-freedom simulation results for an F-16 aircraft model are presented to demonstrate the effectiveness of the proposed control law.

Nomenclature

| | | |
|--------------------------------|---|--|
| F | = | aerodynamic force about the body-fixed frame |
| I | = | moment of inertia |
| L, M, N | = | aerodynamic rolling, pitching, yawing moment |
| p, q, r | = | roll, pitch, yaw rate about the body-fixed frame |
| \bar{q} | = | dynamic pressure |
| T | = | thrust |
| V | = | velocity |
| α, β | = | angle of attack, sideslip angle |
| $\delta_e, \delta_a, \delta_r$ | = | elevator, aileron, rudder angle |
| ϕ, θ, ψ | = | roll, pitch, yaw angle |

I. Introduction

Feedback linearization is a theoretically established and widely used method in controlling nonlinear systems. Exact input–output linearization is often used to control specific output variable sets in nonlinear flight control problems.¹ Unfortunately, this direct application of feedback linearization requires the second or third derivatives of uncertain aerodynamic coefficients and does not guarantee internal stability for nonminimum phase systems.

Another approach to designing flight control laws with feedback linearization is to separate the flight dynamics into fast and slow dynamics by using timescale properties.^{2,3} This method allows the controller design process to be performed without state transformation, because each separated subsystem is square: The number of control inputs is equal to the number of states. The design process of this method can be divided into two steps. In the outer loop, the controller for the slow states α , β , and ϕ is designed to facilitate tracking of the given commands by assuming that the fast states p , q , and r are control inputs, which achieve their commanded values instantaneously. With the slow states controller designed in the outer loop, a separated inner-loop controller is designed to make the fast states p , q , and r follow the outer loop's control input trajectories using the real control inputs: aileron, rudder, and elevator.

This method can be justified only if there is sufficient timescale separation between the inner- and outer-loop dynamics because the fast states p , q , and r are used as control inputs in the outer-loop system. Hence, the states p , q , and r in the inner loop should be much faster than the states α , β , and ϕ in the outer loop. The stability of this timescale separation approach may be analyzed by the singular perturbation theory. However, in most nonlinear flight control re-

search, the gain of the inner-loop controller is set much larger than that of the outer-loop controller, and it is assumed that the aircraft dynamics satisfies this property. This, therefore, does not guarantee closed-loop stability.

Schumacher and Khargonekar analyzed theoretically the stability of the flight control system with the two-timescale separation assumption.⁴ Using the Lyapunov theory, they determined the minimal inner-loop gain guaranteeing closed-loop stability. However, this approach is so complicated and conservative that the calculated value of the minimal inner-loop gain is too large to be applied in the flight controller. It may excite unmodeled dynamics or saturate the control inputs and, therefore, cause robustness problems.

Another difficulty related to the application of feedback linearization to a flight control system is that a complete and accurate aircraft dynamic model including aerodynamic coefficients is required. It is difficult to identify accurately aerodynamic coefficients because they are nonlinear functions of several physical variables. Gain scheduling with a linear \mathcal{H}_∞ design is a traditional approach to overcome this problem; however, it can guarantee the desired performance only when conditions comprise a small perturbation and slow variance.

Neural networks have been proposed recently as an adaptive controller for nonlinear systems.^{5,6} By the use of their universal approximation capability, the adaptive controller based on neural networks can be designed without significant prior knowledge of the system dynamics. In flight control problems, the applications of adaptive neural networks can be found in Refs. 7 and 8. Single-layer neural networks are used to compensate for unknown dynamics⁷ and inversion error.⁸

This paper proposes a backstepping and neural networks controller for a nonlinear flight dynamic system and analyzes the stability of the proposed control system using the Lyapunov theory. The controller is designed by using the backstepping approach⁹ with the assumption that all aerodynamic coefficients are fully understood. The proposed method also intermediately uses the fast states p , q , and r as control inputs. However, it considers the transient responses of the fast states and does not require the two-timescale assumption. Therefore, it is not necessary to make the controller gain impractically large to guarantee the closed-loop stability because the timescale separation assumption is not used in the design or analysis.

The effects of the modeling error in some aerodynamic coefficients are also considered and are compensated for by multilayer neural networks. The parameters of the neural networks are adjusted to offset the term generated by the modeling error. The adaptive controller based on multilayer neural networks is an extension of the work described in Ref. 6. In that paper, only the state variables can be used as input for the neural networks. This paper generalizes it by adding a robust control term, which allows the use of neural network inputs that do not belong to the states. The main contribution of

Received 29 November 1999; revision received 1 July 2000; accepted for publication 2 October 2000. Copyright © 2000 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Graduate Student, Department of Aerospace Engineering.

†Associate Professor, Department of Aerospace Engineering; ydkim@snu.ac.kr. Senior Member AIAA.

this paper is that the stability of nonlinear flight dynamics is proven mathematically without unrealistic restrictions.

This paper is organized as follows. A nonlinear flight model is described in Sec. II. In Sec. III, the backstepping controller is designed when modeling error does not exist. The neural networks adaptive controller is then designed in Sec. IV when aerodynamic modeling error is present. Finally, a numerical simulation of a six-degree-of-freedom F-16 aircraft model is performed to verify the effectiveness of the proposed algorithm in Sec. V.

II. Problem Definition

This paper presents a controller for a nonlinear aircraft. The task of the controller is to track the commands of α , β , and ϕ when aerodynamic model uncertainties exist.

The body-fixed axes, nonlinear equations of motion for an aircraft over a flat Earth are given by¹⁰

$$\begin{aligned}\dot{V} = & \frac{\cos \alpha \cos \beta}{m} [T + F_x] + \frac{\sin \beta}{m} [F_y] + \frac{\sin \alpha \cos \beta}{m} [F_z] \\ & + g[-\cos \alpha \cos \beta \sin \theta + \sin \beta \sin \phi \cos \theta \\ & + \sin \alpha \cos \beta \cos \phi \cos \theta]\end{aligned}\quad (1)$$

$$\begin{aligned}\dot{\alpha} = & -\cos \alpha \tan \beta p + q - \sin \alpha \tan \beta r - \frac{\sin \alpha}{mV \cos \beta} [T + F_x] \\ & + \frac{\cos \alpha}{mV \cos \beta} F_z + \frac{g}{V \cos \beta} [\sin \alpha \sin \theta + \cos \alpha \cos \phi \cos \theta]\end{aligned}\quad (2)$$

$$\begin{aligned}\dot{\beta} = & \sin \alpha p - \cos \alpha r - \frac{\cos \alpha \sin \beta}{mV} [T + F_x] + \frac{\cos \beta}{mV} F_y \\ & - \frac{\sin \alpha \sin \beta}{mV} F_z + \frac{g}{V} [\cos \alpha \sin \beta \sin \theta + \cos \beta \cos \theta \sin \phi \\ & - \sin \alpha \sin \beta \cos \phi \cos \theta]\end{aligned}\quad (3)$$

$$\dot{p} = I_2 pq + I_1 qr + I_3 L + I_4 N \quad (4)$$

$$\dot{q} = I_5 pr - I_6 (p^2 - r^2) + I_7 M \quad (5)$$

$$\dot{r} = -I_2 qr + I_8 pq + I_4 L + I_9 N \quad (6)$$

$$\dot{\phi} = p + \tan \theta (\sin \phi q + \cos \phi r) \quad (7)$$

$$\dot{\theta} = \cos \phi q - \sin \phi r \quad (8)$$

$$\dot{\psi} = \frac{\sin \phi q + \cos \phi r}{\cos \theta} \quad (9)$$

where the moments of inertia I_i , $i = 1, 2, \dots, 9$, are defined as follows:

$$\begin{aligned}I_1 = & -\frac{I_z(I_z - I_y) + I_{xz}^2}{I_x I_z - I_{xz}^2}, & I_2 = & \frac{I_{xz}(I_x - I_y + I_z)}{I_x I_z - I_{xz}^2} \\ I_3 = & \frac{I_z}{I_x I_z - I_{xz}^2}, & I_4 = & \frac{I_{xz}}{I_x I_z - I_{xz}^2}, & I_5 = & \frac{I_z - I_x}{I_y}, & I_6 = & \frac{I_{xz}}{I_y} \\ I_7 = & \frac{1}{I_y}, & I_8 = & \frac{I_x(I_x - I_y) + I_{xz}^2}{I_x I_z - I_{xz}^2}, & I_9 = & \frac{I_x}{I_x I_z - I_{xz}^2}\end{aligned}$$

Definitions of state and control variables, forces, and moments in the preceding equations are described in the Nomenclature. It is assumed that the aerodynamic forces and moments are expressed as functions of angle of attack, sideslip angle, angular rates, and control surface deflection.¹¹ For example, F_x and L are expressed as follows:

$$F_x = [C_x(\alpha) + C_{x_{\delta_e}}(\alpha)\delta_e + (\bar{c}q/2V)C_{x_q}(\alpha)]\bar{q}S$$

$$\begin{aligned}L = & [C_l(\alpha, \beta) + C_{l_{\delta_a}}(\alpha, \beta)\delta_a + C_{l_{\delta_r}}(\alpha, \beta)\delta_r + (bp/2V)C_{l_p}(\alpha) \\ & + (br/2V)C_{l_r}(\alpha)]\bar{q}Sb\end{aligned}$$

Substituting the aerodynamic coefficients into the flight dynamic equations yields

$$\begin{aligned}\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\phi} \end{bmatrix} = & \frac{1}{mV} \begin{bmatrix} -(\sin \alpha / \cos \beta)[T + C_x(\alpha)\bar{q}S] + (\cos \alpha / \cos \beta)C_z(\alpha)\bar{q}S \\ -\cos \alpha \sin \beta [T + C_x(\alpha)\bar{q}S] + \cos \beta C_y(\beta)\bar{q}S - \sin \alpha \sin \beta C_z(\alpha, \beta)\bar{q}S \\ 0 \end{bmatrix} + \begin{bmatrix} -\cos \alpha \tan \beta & 1 & -\sin \alpha \tan \beta \\ \sin \alpha & 0 & -\cos \alpha \\ 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \\ & + \frac{\rho S}{4m} \begin{bmatrix} 0 & -(\sin \alpha / \cos \beta)C_{x_q}(\alpha)\bar{c} + (\cos \alpha / \cos \beta)C_{z_q}(\alpha)\bar{c} & 0 \\ \cos \beta C_{y_p}(\alpha)b & -\cos \alpha \sin \beta C_{x_q}(\alpha)\bar{c} - \sin \alpha \sin \beta C_{z_q}(\alpha)\bar{c} & \cos \beta C_{y_r}(\alpha)b \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \\ & + \frac{\rho VS}{2m} \begin{bmatrix} -(\sin \alpha / \cos \beta)C_{x_{\delta_e}}(\alpha) + (\cos \alpha / \cos \beta)C_{z_{\delta_e}}(\alpha, \beta) & 0 & 0 \\ -\cos \alpha \sin \beta C_{x_{\delta_e}}(\alpha) - \sin \alpha \sin \beta C_{z_{\delta_e}}(\alpha, \beta) & \cos \beta C_{y_{\delta_a}}(\beta) & \cos \beta C_{y_{\delta_r}}(\beta) \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_a \\ \delta_r \end{bmatrix} \\ & + \frac{g}{V} \begin{bmatrix} (1/\cos \beta)(\sin \alpha \sin \theta + \cos \alpha \cos \phi \cos \theta) \\ \cos \alpha \sin \beta \sin \theta + \cos \beta \cos \theta \sin \phi - \sin \alpha \sin \beta \cos \phi \cos \theta \\ 0 \end{bmatrix}\end{aligned}\quad (10)$$

$$\begin{aligned}\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = & \begin{bmatrix} I_2 pq + I_1 qr \\ I_5 pr - I_6 (p^2 - r^2) \\ -I_2 qr + I_8 pq \end{bmatrix} + \begin{bmatrix} I_3 C_l(\alpha, \beta)\bar{q}Sb + I_4 C_n(\alpha, \beta)\bar{q}Sb \\ I_7 C_m(\alpha)\bar{q}S\bar{c} \\ I_4 C_l(\alpha, \beta)\bar{q}Sb + I_9 C_n(\alpha, \beta)\bar{q}Sb \end{bmatrix} \\ & + \frac{\rho VS}{4} \begin{bmatrix} I_3 C_{l_p}(\alpha)b + I_4 C_{n_p}(\alpha)b & 0 & I_3 C_{l_r}(\alpha)b + I_4 C_{n_r}(\alpha)b \\ 0 & I_7 C_{m_q}(\alpha)\bar{c} & 0 \\ I_4 C_{l_p}(\alpha)b + I_9 C_{n_p}(\alpha)b & 0 & I_4 C_{l_r}(\alpha)b + I_9 C_{n_r}(\alpha)b \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \\ & + \bar{q}S \begin{bmatrix} 0 & I_3 C_{l_{\delta_a}}(\alpha, \beta)b + I_4 C_{n_{\delta_a}}(\alpha, \beta)b & I_3 C_{l_{\delta_r}}(\alpha, \beta)b + I_4 C_{n_{\delta_r}}(\alpha, \beta)b \\ I_7 C_{m_{\delta_e}}(\alpha)\bar{c} & 0 & 0 \\ 0 & I_4 C_{l_{\delta_a}}(\alpha, \beta)b + I_9 C_{n_{\delta_a}}(\alpha, \beta)b & I_4 C_{l_{\delta_r}}(\alpha, \beta)b + I_9 C_{n_{\delta_r}}(\alpha, \beta)b \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_a \\ \delta_r \end{bmatrix}\end{aligned}\quad (11)$$

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (12)$$

Let us define the states $x_1, x_2 \in \mathcal{R}^3$, $x_3 \in \mathcal{R}^2$, and the control inputs $u \in \mathcal{R}^3$ as follows: $x_1 = [\alpha, \beta, \phi]^T$, $x_2 = [p, q, r]^T$, $x_3 = [\theta, \psi]^T$, and $u = [\delta_e, \delta_a, \delta_r]^T$. With the choices of x_1, x_2, x_3 , and u , the flight dynamic Eqs. (10–12) can be rearranged as

$$\begin{aligned} \dot{x}_1 &= f_1(\alpha, \beta) + g_1(\alpha, \beta, \phi, \theta)x_2 + g_{1a}(\alpha, \beta)x_2 \\ &\quad + h_1(\alpha, \beta)u + f_{1g}(\alpha, \beta, \phi, \theta) \end{aligned} \quad (13)$$

$$\dot{x}_2 = f_2(\alpha, \beta, p, q, r) + f_{2a}(\alpha, \beta)x_2 + g_2(\alpha, \beta)u \quad (14)$$

$$\dot{x}_3 = f_3(\phi, \theta)x_2 \quad (15)$$

where f, g , and h represent the terms of Eqs. (10–12) in order. The preceding equations are mainly used for the controller design and stability analysis processes of the following sections.

III. Controller Design and Stability Analysis

When designing a flight control system with the two-timescale assumption, the inner-loop controller is designed to control the fast states x_2 using the control input u , where the desired values of the fast states x_2^d are given by the outer loop. In the outer loop, the controller is designed to control the slow states x_1 using the fast states x_2 as control inputs. The inner-loop controller neglects the transient responses of the fast states x_2 . It assumes that the fast states track their commanded values instantaneously and that the control surface deflection has no effect on the outer-loop dynamics. The structure of the two-timescale controller is shown in Fig. 1. In each feedback loop, control laws x_2^d and u are designed separately.

In this paper, the backstepping method is used to design a controller.⁹ The backstepping design procedure can be viewed as the two-timescale approach because the fast states x_2 are used as control inputs for the slow states x_1 intermediately. However, this methodology considers the transient responses of the fast states and, therefore, does not require the timescale separation assumption. The following assumptions are used in the design and analysis processes.

Assumption 1: The desired trajectories $x_1^d = [\alpha^d, \beta^d, \phi^d]^T$ are bounded as

$$\| [x_1^d, \dot{x}_1^d, \ddot{x}_1^d] \| \leq c_d \quad (16)$$

where $c_d \in \mathcal{R}$ is a known positive constant and $\| \cdot \|$ denotes the 2-norm of a vector or a matrix.

Assumption 2: The total velocity and the dynamic pressure are constant.

$$\dot{V} = 0, \quad \dot{q} = 0 \quad (17)$$

Assumption 3: There exist positive constants α_m and $\beta_m \in \mathcal{R}$ such that the magnitudes and derivatives of f_1, f_2, f_{2a}, g_{1a} , and g_2 are bounded for all α and $\beta \in \mathcal{R}$ satisfying $|\alpha| \leq \alpha_m$ and $|\beta| \leq \beta_m$.

Assumption 4: The magnitude of θ is bounded as

$$|\theta| \leq \theta_m < \pi/2 \quad (18)$$

where $\theta_m \in \mathcal{R}$ is a positive constant.

The following lemma is used in the design and analysis processes.

Lemma 1: There exist positive constants α_m, β_m , and $\theta_m \in \mathcal{R}$ such that $g_1(\alpha, \beta, \phi, \theta)$ is invertible for all $\phi \in \mathcal{R}$ and all α, β , and $\theta \in \mathcal{R}$ satisfying $|\alpha| \leq \alpha_m, |\beta| \leq \beta_m$, and $|\theta| \leq \theta_m$.

Lemma 1 can be proved by a sufficient condition that the set of rows of g_1 is linearly independent.

According to Assumption 3 and Lemma 1, the size of g_1^{-1} and g_{1a} is bounded by some positive constants $c_{g_{1a}}, c_{g_1^{-1}}$:

$$\| g_1(\alpha, \beta, \phi, \theta)^{-1} \| \leq c_{g_1^{-1}} \quad (19)$$

$$\| g_{1a}(\alpha, \beta) \| \leq c_{g_{1a}} \quad (20)$$

Assumption 5: The following inequalities are satisfied for constants $c_{g_{1a}}$ and $c_{g_1^{-1}}$ in Eqs. (19) and (20):

$$c_{g_{1a}} c_{g_1^{-1}} < 1 \quad (21)$$

Note that g_{1a} is composed of the aerodynamic coefficient terms related to the angular rate and multiplied by a very small quantity ρ/m , and therefore, the magnitude of g_{1a} is very small. Also note that the norm of $c_{g_1^{-1}}$ is mainly influenced by the pitch angle θ , and therefore Assumption 5 is closely related to the maximum pitch angle in Lemma 1. For the aerodynamic model considered in this study, the numerical value of $c_{g_{1a}} c_{g_1^{-1}}$ is less than 0.13.

Assumption 6: There exist positive constants α_m and $\beta_m \in \mathcal{R}$ such that $g_2(\alpha, \beta)$ is invertible for all α and $\beta \in \mathcal{R}$ satisfying $|\alpha| \leq \alpha_m$ and $|\beta| \leq \beta_m$.

Assumption 7: The control surface deflection has no effects on the aerodynamic force component:

$$h_1(\alpha, \beta) = 0 \quad (22)$$

Note that g_2 in Eq. (14) represents the input matrix of the control u to the dynamics of the angular rates p, q , and r . Also, h_1 in Eq. (13) represents the aerodynamic force component caused by the control surface deflection. Because the control surfaces of the aircraft are designed to control each axes' angular rate of aircraft independently, the input matrix g_2 is invertible for all cases, and the magnitude of h_1 is very small compared to other aerodynamic terms in the dynamic equation. Therefore, it can be assumed that g_2 is invertible, and $h_1 = 0$. Numerical studies for the aerodynamic model considered in this study also show that g_2 is always invertible, and the size of h_1 is negligible.

Let us introduce the error state variables z_1 and $z_2 \in \mathcal{R}^3$ as follows:

$$z_1 = x_1 - x_1^d \quad (23)$$

$$z_2 = x_2 - x_2^d \quad (24)$$

where x_1^d and x_2^d are the desired trajectories of x_1 and x_2 , respectively. Note that x_1^d is given by command signals and x_2^d will be defined later.

Using Eqs. (13) and (14), and Assumption 7, the dynamic equations of the error states are given as follows:

$$\dot{z}_1 = \dot{x}_1 - \dot{x}_1^d = f_1 + g_1 x_2 + g_{1a} x_2 + f_{1g} - \dot{x}_1^d \quad (25)$$

$$\dot{z}_2 = \dot{x}_2 - \dot{x}_2^d = f_2 + f_{2a} x_2 + g_2 u - \dot{x}_2^d \quad (26)$$

Theorem 1: Consider the system in Eqs. (25) and (26), where the control input u is defined in Eq. (27). Then, the solutions of the system are locally uniformly ultimately bounded:

$$u = g_2^{-1} [-k_2 z_2 - g_{1a}^T z_1 - g_1^T z_1 - A] \quad (27)$$

where x_2^d and $A \in \mathcal{R}^{3 \times 1}$ are defined in Eqs. (28) and (29) and k_1 and $k_2 \in \mathcal{R}$ are positive design parameters:

$$x_2^d = g_1^{-1} [-k_1 z_1 - f_1 - f_{1g} + \dot{x}_1^d] \quad (28)$$

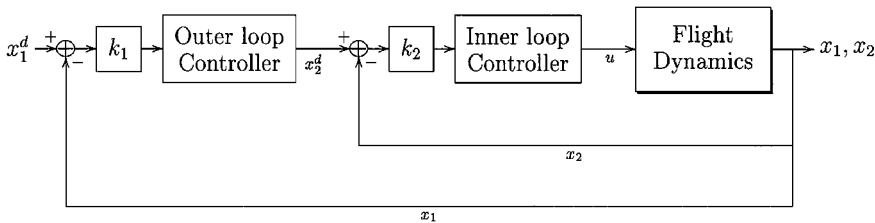


Fig. 1 Structure of the two-timescale controller.

$$A = f_2 + f_{2a}x_2 - \frac{\partial x_2^d}{\partial x_1} [f_1 + g_1x_2 + g_{1a}x_2 + f_{1g}] - \frac{\partial x_2^d}{\partial x_3} f_3x_2 - g_1^{-1} [k_1\dot{x}_1^d + \ddot{x}_1^d] \quad (29)$$

Furthermore, the bound of the tracking error may be kept as small as desired by adjusting the design parameters.

Proof: By Assumptions 1 and 3, the following inequalities are satisfied for certain positive constants c_{f_1} , $c_{f_{1g}}$, and $c_{\dot{x}_1^d}$:

$$\|f_1(\alpha, \beta)\| \leq c_{f_1} \quad (30)$$

$$\|f_{1g}(\alpha, \beta, \phi, \theta)\| \leq c_{f_{1g}} \quad (31)$$

$$\|\dot{x}_1^d\| \leq c_{\dot{x}_1^d} \quad (32)$$

The bound of $\|x_2^d\|$ can be computed using Eqs. (19), (30), (31), and (32) as

$$\|x_2^d\| \leq c_{g_1^{-1}} [k_1\|z_1\| + c_{f_1} + c_{f_{1g}} + c_{\dot{x}_1^d}] \quad (33)$$

Let us consider the Lyapunov function candidate,

$$V_1 = \frac{1}{2}z_1^T z_1 + \frac{1}{2}z_2^T z_2 \quad (34)$$

The derivative of the Lyapunov function V_1 along the trajectories of Eqs. (25) and (26) is given by

$$\dot{V}_1 = \frac{\partial V_1}{\partial z_1} \dot{z}_1 + \frac{\partial V_1}{\partial z_2} \dot{z}_2 \quad (35)$$

Substitution of Eqs. (25) and (28) into Eq. (35) yields the following equation:

$$\begin{aligned} \dot{V}_1 &= \frac{\partial V_1}{\partial z_1} [-k_1z_1 + g_{1a}x_2] + \frac{\partial V_1}{\partial z_1} g_1[x_2 - x_2^d] + \frac{\partial V_1}{\partial z_2} \dot{z}_2 \\ &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2 + z_1^T g_1z_2 + z_2^T \dot{z}_2 \\ &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2^d + z_1^T g_{1a}z_2 + z_1^T g_1z_2 + z_2^T \dot{z}_2 \end{aligned} \quad (36)$$

Substituting Eqs. (26) and (29) into Eq. (36) yields

$$\begin{aligned} \dot{V}_1 &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2^d + z_1^T g_{1a}z_2 + z_1^T g_1z_2 + z_2^T \left[f_2 + f_{2a}x_2 \right. \\ &\quad \left. + g_2u - \frac{\partial x_2^d}{\partial x_1} \dot{x}_1 - \frac{\partial x_2^d}{\partial x_3} \dot{x}_3 - \frac{\partial x_2^d}{\partial x_1^d} \dot{x}_1^d - \frac{\partial x_2^d}{\partial x_1^d} \ddot{x}_1^d \right] \\ \dot{V}_1 &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2^d + z_1^T g_{1a}z_2 + z_1^T g_1z_2 + z_2^T [A + g_2u] \\ &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2^d + z_2^T [g_{1a}^T z_1 + g_1^T z_1 + A + g_2u] \end{aligned} \quad (37)$$

Because f_1 in Eq. (28) is dependent on V , the term $-z_2^T (\partial x_2^d / \partial V) \dot{V}$ must be included in Eq. (37); however, it is neglected by using Assumption 2 in this study. Substituting Eq. (27) into Eq. (38) and using Eqs. (20) and (33) give the following equations:

$$\begin{aligned} \dot{V}_1 &= -k_1\|z_1\|^2 - k_2\|z_2\|^2 + z_1^T g_{1a}x_2^d \leq -k_1\|z_1\|^2 - k_2\|z_2\|^2 \\ &\quad + \|g_{1a}\| \|z_1\| \|x_2^d\| \leq -k_1(1 - c_{g_{1a}} c_{g_1^{-1}}) \|z_1\|^2 + c_{g_{1a}} c_{g_1^{-1}} (c_{f_1} \\ &\quad + c_{f_{1g}} + c_{\dot{x}_1^d}) \|z_1\| - k_2\|z_2\|^2 = -k_1(1 - c_1) \|z_1\|^2 + c_1 c_2 \|z_1\| \\ &\quad - k_2\|z_2\|^2 = -\frac{k_1}{2}(1 - c_1) \|z_1\|^2 - \frac{k_1}{2}(1 - c_1) \left[\|z_1\| \right. \\ &\quad \left. - \frac{c_1 c_2}{k_1(1 - c_1)} \right]^2 + \frac{c_1^2 c_2^2}{2k_1(1 - c_1)} - k_2\|z_2\|^2 \\ &\leq -\frac{k_1}{2}(1 - c_1) \|z_1\|^2 - k_2\|z_2\|^2 + \frac{c_1^2 c_2^2}{2k_1(1 - c_1)} \end{aligned} \quad (39)$$

$$\dot{V}_1 \leq -2\mu_1 V_1 + \frac{c_1^2 c_2^2}{2k_1(1 - c_1)} \quad (40)$$

where the constants c_1 , c_2 , and μ_1 are defined as follows:

$$\begin{aligned} c_1 &= c_{g_{1a}} c_{g_1^{-1}}, & c_2 &= c_{f_1} + c_{f_{1g}} + c_{\dot{x}_1^d} \\ \mu_1 &= \min[(k_1/2)(1 - c_1), k_2] \end{aligned} \quad (41)$$

Note that μ_1 is positive because c_1 is less than 1 by Assumption 5. Equation (40) implies that $\dot{V}_1 < 0$ when $V_1 > c_1^2 c_2^2 / [4\mu_1 k_1(1 - c_1)]$. Therefore, the error states z_1 and z_2 are bounded and converge exponentially to the residual set \mathcal{D}_1 :

$$\mathcal{D}_1 = \left\{ z_1, z_2 \in \mathcal{R}^3 \mid \|z_1\|^2 + \|z_2\|^2 \leq \frac{c_1^2 c_2^2}{2\mu_1 k_1(1 - c_1)} \right\} \quad (42)$$

Because c_1 and c_2 do not depend on k_1 and k_2 , the size of the set \mathcal{D}_1 can be made arbitrarily small by adjusting the design parameters k_1 and k_2 .

Theorem 1 represents the design procedure of the controller to track the α , β , and ϕ commands, and also shows that the tracking error of the control system converges to a compact set whose size is adjustable by the design parameters. Because the timescale separation assumption is not used in this study, it is not necessary to make the control gain large to guarantee closed-loop stability.

IV. Neural Networks Adaptive Controller Design

In the preceding section, the backstepping controller is designed with the assumption that the aerodynamic characteristics are fully known. Because aerodynamic coefficients are highly nonlinear and dependent on lots of physical variables, it is very difficult to identify them exactly. The difference between the mathematical model and the real system may cause performance degradation. To overcome this drawback, multilayer neural networks are used in this study. The weights of the neural networks are adjusted to compensate for the effect of the modeling error. The adaptive controller based on multilayer neural networks is an extension of the work described in Ref. 6. In this paper, it is generalized such that the variables that do not belong to the states can be used as neural networks' inputs.

A. Effect of Modeling Error

In this paper, the modeling error in the body-fixed angular rates dynamics is considered. The identified values of f_2 , f_{2a} , and g_2 in Eq. (14) are defined as \hat{f}_2 , \hat{f}_{2a} , and \hat{g}_2 , respectively. Then, the control input is expressed as follows:

$$\hat{u} = \hat{g}_2^{-1} [-k_2 z_2 - g_{1a}^T z_1 - g_1^T z_1 - \hat{A}] \quad (43)$$

$$\begin{aligned} \hat{A} &= \hat{f}_2 + \hat{f}_{2a}x_2 - \frac{\partial x_2^d}{\partial x_1} [f_1 + g_1x_2 + g_{1a}x_2 + f_{1g}] \\ &\quad - \frac{\partial x_2^d}{\partial x_3} f_3x_2 - g_1^{-1} [k_1\dot{x}_1^d + \ddot{x}_1^d] \end{aligned} \quad (44)$$

Substituting Eqs. (43) and (44) into Eq. (38) gives

$$\begin{aligned} \dot{V}_1 &= -k_1\|z_1\|^2 + z_1^T g_{1a}x_2^d + z_2^T [g_{1a}^T z_1 + g_1^T z_1 + A + g_2\hat{u} \\ &\quad + g_2u - g_2\hat{u}] = -k_1\|z_1\|^2 - k_2\|z_2\|^2 + z_1^T g_{1a}x_2^d \\ &\quad + z_2^T g_2[u - \hat{u}] \triangleq -k_1\|z_1\|^2 - k_2\|z_2\|^2 + z_1^T g_{1a}x_2^d - z_2^T \Delta_1 \end{aligned}$$

where $\Delta_1 \equiv g_2[u - \hat{u}]$ represents the term caused by the modeling error, that is, the modeling error of f_2 , f_{2a} , and g_2 adds the term $z_2^T \Delta_1$ to the derivative of the Lyapunov function.

B. Neural Networks Structure

Given an input $x_{nn}^0 \in \mathcal{R}^{N_1}$, the three-layer neural networks as shown in Fig. 2 has an output $y_{nn} \in \mathcal{R}^{N_3}$ as

$$y_{nni} = \sum_{j=1}^{N_2} \left[w_{ij} \sigma \left(\sum_{k=1}^{N_1} v_{jk} x_{nnk}^0 + \theta_{vj} \right) + \theta_{wi} \right], \quad i = 1, 2, \dots, N_3 \quad (45)$$

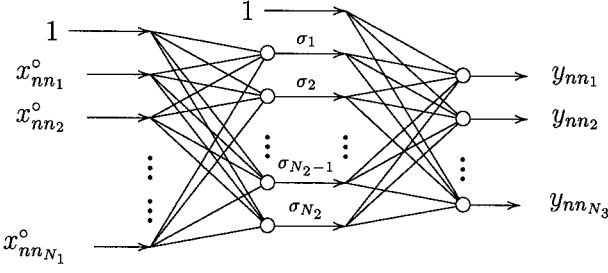


Fig. 2 Structure of three-layer neural networks.

where v_{jk} is the weight connecting the first layer to the second layer, w_{ij} is the weight connecting the second layer to the third layer, θ is a bias, and N_i is the number of neurons in the i th layer. The sigmoid activation function $\sigma(\cdot)$ is defined by

$$\sigma(z) = 1/(1 + e^{-z}) \quad (46)$$

The neural networks input-output mapping equation (45) can be expressed in matrix form as follows:

$$y_{nn} = W^T \sigma(V^T x_{nn}) \quad (47)$$

where

$W \in \mathcal{R}^{N_2+1 \times N_3}$, $V \in \mathcal{R}^{N_1+1 \times N_2}$, $x_{nn} \in \mathcal{R}^{N_1+1}$, and $\sigma: \mathcal{R}^{N_2} \mapsto \mathcal{R}^{N_2+1}$ are defined as follows:

$$W^T = \begin{bmatrix} \theta_{w1} & w_{11} & w_{12} & \cdots \\ \theta_{w2} & w_{21} & w_{22} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \quad V^T = \begin{bmatrix} \theta_{v1} & v_{11} & v_{12} & \cdots \\ \theta_{v2} & v_{21} & v_{22} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$x_{nn} = [1, x_{nn1}^0, x_{nn2}^0, \dots, x_{nnN_1}^0]^T$$

$$\sigma(z) = [1, \sigma(z_1), \sigma(z_2), \dots, \sigma(z_{N_2})]^T$$

Multilayer neural networks can approximate a nonlinear function to any desired accuracy. This is known as the universal approximation capability.^{12,13} That is, for a continuous function $\Delta: \mathcal{R}^{N_1} \mapsto \mathcal{R}^{N_3}$ and an arbitrary constant $\epsilon_N > 0$, there exist an integer N_2 , the number of neurons in the hidden layer, and ideal constant weight matrices $W \in \mathcal{R}^{N_2+1 \times N_3}$ and $V \in \mathcal{R}^{N_1+1 \times N_2}$ such that

$$\Delta = W^T \sigma(V^T x_{nn}) + \epsilon(x_{nn}) \quad (48)$$

where $\epsilon(x_{nn})$ is the approximation error satisfying $\|\epsilon(x_{nn})\| \leq \epsilon_N$ for all x_{nn} in some input space.

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

Assumption 8: Here, $x_{nn}^0 = [x_1^d, x_1, x_2]^T$ is defined as the input vector of the neural networks, and this input satisfies Eq. (48) for some ϵ_N .

Assumption 9: The ideal weights are bounded in the sense that

$$\|W\|_F \leq W_M, \quad \|V\|_F \leq V_M \quad (49)$$

where W_M and $V_M \in \mathcal{R}$ are known positive constants and $\|\cdot\|_F$ denotes the Frobenious norm of a matrix.

The ideal weight matrices W and V satisfying Eq. (48) cannot be determined in anticipation because we have no information on the error term Δ . Instead, the estimated values of the ideal weights, \hat{W} and \hat{V} , are used in the controller, and they are adjusted by the adaptive laws. Consequently, there exists an effect caused by the difference between the ideal weights W and V , and the estimated weights \hat{W} and \hat{V} . This effect is stated in the following lemma.

Lemma 2: Let us define the weight estimation errors as $\tilde{W} = W - \hat{W}$, $\tilde{V} = V - \hat{V}$, and $Z = \text{diag}[W, V]$. Given a neural networks' input x_{nn} , the output error is expressed as

$$\begin{aligned} \hat{W}^T \sigma(\hat{V}^T x_{nn}) - \Delta = & -\tilde{W}^T [\sigma(\hat{V}^T x_{nn}) - \sigma'(\hat{V}^T x_{nn}) \hat{V}^T x_{nn}] \\ & - \hat{W}^T \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} + w \end{aligned} \quad (50)$$

where

$$\sigma'(\hat{z}) = \left. \frac{d\sigma}{dz} \right|_{z=\hat{z}}$$

and $w \in \mathcal{R}^3$ is defined as

$$w(t) = -\tilde{W}^T \sigma'(\hat{V}^T x_{nn}) V^T x_{nn} - W^T \mathcal{O}(\tilde{V}^T x_{nn}) - \epsilon(x_{nn}) \quad (51)$$

Furthermore, $\|w\|$ satisfies the following inequality for some positive constants C_i , $i = 1, 2, 3, 4$:

$$\|w\| \leq C_1 + C_2 \|\tilde{Z}\|_F + C_3 \|\tilde{Z}\|_F \|x_1\| + C_4 \|\tilde{Z}\|_F \|x_2\| \quad (52)$$

Proof: The output error of the hidden layer for the input x_{nn} is defined as

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(V^T x_{nn}) - \sigma(\hat{V}^T x_{nn}) \quad (53)$$

The Taylor series expansion of σ in Eq. (53) may be written as

$$\sigma(V^T x_{nn}) = \sigma(\hat{V}^T x_{nn}) + \left. \frac{d\sigma}{dz} \right|_{z=\hat{V}^T x_{nn}} \tilde{V}^T x_{nn} + \mathcal{O}(\tilde{V}^T x_{nn}) \quad (54)$$

By the use of Eqs. (53) and (54), the hidden layer output error equation can be written as follows:

$$\tilde{\sigma} = \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} + \mathcal{O}(\tilde{V}^T x_{nn}) \quad (55)$$

Then, the error of the output layer is expressed as follows:

$$\begin{aligned} \hat{W}^T \sigma(\hat{V}^T x_{nn}) - \Delta = & \hat{W}^T \sigma(\hat{V}^T x_{nn}) - W^T \sigma(V^T x_{nn}) - \epsilon(x_{nn}) \\ = & -\tilde{W}^T \sigma(\hat{V}^T x_{nn}) - W^T [\sigma(V^T x_{nn}) - \sigma(\hat{V}^T x_{nn})] - \epsilon(x_{nn}) \end{aligned} \quad (56)$$

Substituting Eq. (55) and $W = \tilde{W} + \hat{W}$ and $\tilde{V} = V - \hat{V}$ into Eq. (56) yields

$$\begin{aligned} \hat{W}^T \sigma(\hat{V}^T x_{nn}) - \Delta = & -\tilde{W}^T \sigma(\hat{V}^T x_{nn}) - \tilde{W}^T \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} \\ & - \hat{W}^T \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} - W^T \mathcal{O}(\tilde{V}^T x_{nn}) \\ & - \epsilon(x_{nn}) \\ = & -\tilde{W}^T [\sigma(\hat{V}^T x_{nn}) - \sigma'(\hat{V}^T x_{nn}) \hat{V}^T x_{nn}] \\ & - \hat{W}^T \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} + w \end{aligned} \quad (57)$$

Therefore, Eq. (50) has been proved.

When the sigmoid function $\sigma(\cdot)$ and its derivative $d\sigma(z)/dz$ are bounded by some constants and Assumption 8 is used, it can be shown that the high-order term \mathcal{O} in Eq. (54) satisfies the following inequality:

$$\begin{aligned} \|\mathcal{O}(\tilde{V}^T x_{nn})\| \leq & \|\sigma(V^T x_{nn})\| + \|\sigma(\hat{V}^T x_{nn})\| \\ & + \|\sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn}\| \leq c + c + c \|\tilde{V}\|_F \|x_{nn}\| \\ \leq & c + c \|\tilde{V}\|_F + c \|x_1\| \|\tilde{V}\|_F + c \|x_2\| \|\tilde{V}\|_F \end{aligned} \quad (58)$$

where c is a generic symbol used to denote any finite constant. Note that the property of $\|Ax\| \leq \|A\|_F \|x\|$ is used in the preceding equations.

Finally, Eq. (52) is proved by using Eqs. (51) and (58) and hbox-Assumption 8 as follows:

$$\begin{aligned} \|w(t)\| \leq & \|\tilde{W}\|_F c V_M (c + c \|x_1\| + c \|x_2\|) + W_M (c + c \|\tilde{V}\|_F \\ & + c \|x_1\| \|\tilde{V}\|_F + c \|x_2\| \|\tilde{V}\|_F) + \epsilon_N \leq C_1 + C_2 \|\tilde{Z}\|_F \\ & + C_3 \|\tilde{Z}\|_F \|x_1\| + C_4 \|\tilde{Z}\|_F \|x_2\| \end{aligned} \quad (59)$$

□

This lemma shows that the neural networks output error caused by the weight estimation error can be expressed as Eq. (50) and that the size of the term w in Eq. (50) is bounded by Eq. (52).

C. Design of the Adaptive Controller

The following lemma defines a robust term v and shows a property for the stability analysis.

Lemma 3: Consider $w \in \mathcal{R}^3$ defined as in Eq. (51) and $v \in \mathcal{R}^3$, ξ , and $k_v \in \mathcal{R}$ defined as

$$v = -\frac{z_2 \xi}{\|z_2\| \xi + \epsilon} \xi \quad (60)$$

$$\xi = k_v (Z_M + \|\hat{Z}\|_F) (\|x_1\| + \|x_2\|) \quad (61)$$

$$k_v \geq \max\{C_3, C_4\} \quad (62)$$

where ϵ is a positive constant. Then, the following inequality is satisfied:

$$z_2^T (w + v) \leq \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F] + \epsilon \quad (63)$$

Proof: With the choice of v in Eq. (60), $z_2^T (w + v)$ is expressed as

$$z_2^T (w + v) = z_2^T w - \frac{(\|z_2\| \xi)^2}{\|z_2\| \xi + \epsilon} \quad (64)$$

Substituting Eq. (52) into the preceding equation gives

$$z_2^T (w + v) \leq \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F + C_3 \|\tilde{Z}\|_F \|x_1\| + C_4 \|\tilde{Z}\|_F \|x_2\|] - \frac{(\|z_2\| \xi)^2}{\|z_2\| \xi + \epsilon} \quad (65)$$

Furthermore, from the definition $\tilde{Z} = Z - \hat{Z}$ and Eqs. (61) and (62), we have

$$C_3 \|\tilde{Z}\|_F \|x_1\| + C_4 \|\tilde{Z}\|_F \|x_2\| \leq k_v \|Z - \hat{Z}\|_F (\|x_1\| + \|x_2\|) \leq \xi \quad (66)$$

When Eqs. (65) and (66) are used, Eq. (63) can be proved as follows:

$$z_2^T (w + v) \leq \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F + \xi] - \frac{(\|z_2\| \xi)^2}{\|z_2\| \xi + \epsilon} = \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F] + \frac{\|z_2\| \xi}{\|z_2\| \xi + \epsilon} \epsilon \leq \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F] + \epsilon \quad (67)$$

The design process of the adaptive controller and stability analysis are stated in the following theorem.

Theorem 2: Consider the system Eqs. (25) and (26), where the control input u is defined in Eq. (68) and the adaptive laws are defined in Eqs. (69) and (70). Then, the trajectories of the system as well as the neural networks' weights are locally uniformly ultimately bounded:

$$u = \hat{g}_2^{-1} [-k_2 z_2 - g_{1a}^T z_1 - g_1^T z_1 - \hat{A}] + \hat{W}^T \sigma(\hat{V}^T x_{nn}) + v \quad (68)$$

where \hat{A} and v are defined as in Eqs. (44) and (60), respectively. \hat{W} and \hat{V} are computed by the following adaptive laws:

$$\dot{\hat{W}} = -\gamma_w [\sigma(\hat{V}^T x_{nn}) z_2^T - \sigma'(\hat{V}^T x_{nn}) \hat{V}^T x_{nn} z_2^T] - \kappa \gamma_w \hat{W} \quad (69)$$

$$\dot{\hat{V}} = -\gamma_v x_{nn} [\sigma'(\hat{V}^T x_{nn})^T \hat{W} z_2] - \kappa \gamma_v \hat{V} \quad (70)$$

where κ , γ_w , and $\gamma_v \in \mathcal{R}$ are some positive design parameters. Furthermore, the bound of the tracking error and the neural networks parameter estimation error may be kept as small as desired by adjusting the design parameters.

Proof: Let us consider the Lyapunov function candidate,

$$V_2 = \frac{1}{2} z_1^T z_1 + \frac{1}{2} z_2^T z_2 + (1/2\gamma_w) \text{tr} [\tilde{W}^T \tilde{W}] + (1/2\gamma_v) \text{tr} [\tilde{V}^T \tilde{V}] \quad (71)$$

When Eq. (38) is used the derivative of the Lyapunov function V_2 along the trajectories of Eqs. (25) and (26) with the adaptive laws, Eqs. (69) and (70), are computed as

$$\dot{V}_2 = -k_1 \|z_1\|^2 + z_1^T g_{1a} x_2^d + z_2^T [g_{1a}^T z_1 + g_1^T z_1 + A + g_2 u + g_2 u^* - g_2 u^*] + (1/\gamma_w) \text{tr} [\tilde{W}^T \dot{\tilde{W}}] + (1/\gamma_v) \text{tr} [\tilde{V}^T \dot{\tilde{V}}] \quad (72)$$

where u^* is the ideal control input when the exact aerodynamic model is available and is expressed as follows:

$$u^* = g_2^{-1} [-k_2 z_2 - g_{1a}^T z_1 - g_1^T z_1 - A] + \hat{W}^T \sigma(\hat{V}^T x_{nn}) + v \quad (73)$$

With the definition of u^* in Eq. (73), Eq. (72) becomes

$$\dot{V}_2 = -k_1 \|z_1\|^2 - k_2 \|z_2\|^2 + z_1^T g_{1a} x_2^d - z_2^T \Delta_2 + z_2^T \hat{W}^T \sigma(\hat{V}^T x_{nn}) + z_2^T v + (1/\gamma_w) \text{tr} [\tilde{W}^T \dot{\tilde{W}}] + (1/\gamma_v) \text{tr} [\tilde{V}^T \dot{\tilde{V}}] \quad (74)$$

where Δ_2 is defined as $g_2[u^* - u]$ and represents the term caused by the modeling error. Substituting Eqs. (50), (69), and (70) into the preceding equation gives the following equation.

$$\begin{aligned} \dot{V}_2 = & -k_1 \|z_1\|^2 - k_2 \|z_2\|^2 + z_1^T g_{1a} x_2^d + z_2^T \{-\tilde{W}^T [\sigma(\hat{V}^T x_{nn}) \\ & - \sigma'(\hat{V}^T x_{nn}) \hat{V}^T x_{nn}] - \hat{W}^T \sigma'(\hat{V}^T x_{nn}) \tilde{V}^T x_{nn} + w + v\} \\ & + \text{tr} \{ \tilde{W}^T [\sigma(\hat{V}^T x_{nn}) z_2^T - \sigma'(\hat{V}^T x_{nn}) \hat{V}^T x_{nn} z_2^T + \kappa \hat{W}] \} \\ & + \text{tr} (\tilde{V}^T \{ x_{nn} [\sigma'(\hat{V}^T x_{nn})^T \hat{W} z_2]^T + \kappa \hat{V} \}) \end{aligned} \quad (75)$$

When the property of the trace, $\text{tr}[xy^T] = x^T y$, is used, the preceding equation is simplified as follows:

$$\dot{V}_2 = -k_1 \|z_1\|^2 - k_2 \|z_2\|^2 + z_1^T g_{1a} x_2^d + \kappa \text{tr} [\tilde{Z}^T \hat{Z}] + z_2^T (w + v) \quad (76)$$

Using the property $\text{tr} [\tilde{Z}^T \hat{Z}] = \text{tr} [\tilde{Z}^T Z] - \text{tr} [\tilde{Z}^T \tilde{Z}] \leq \|\tilde{Z}\|_F Z_M - \|\tilde{Z}\|_F^2$ and Eqs. (39) and (63), we have

$$\begin{aligned} \dot{V}_2 \leq & -\frac{k_1}{2} (1 - c_1) \|z_1\|^2 - k_2 \|z_2\|^2 + \frac{c_1^2 c_2^2}{2k_1 (1 - c_1)} + \kappa [\|\tilde{Z}\|_F Z_M \\ & - \|\tilde{Z}\|_F^2] + \|z_2\| [C_1 + C_2 \|\tilde{Z}\|_F] + \epsilon = -\frac{k_1}{2} (1 - c_1) \|z_1\|^2 \\ & - \frac{k_2}{2} \|z_2\|^2 - \frac{k_2}{2} \left[\|z_2\| - \frac{C_1}{k_2} \right]^2 - \frac{\kappa}{2} \|\tilde{Z}\|_F^2 - \frac{\kappa}{2} [\|\tilde{Z}\|_F - Z_M]^2 \\ & + C_2 \|z_2\| \|\tilde{Z}\|_F + \frac{c_1^2 c_2^2}{2k_1 (1 - c_1)} + \frac{C_1^2}{2k_2} + \epsilon + \frac{\kappa Z_M^2}{2} \end{aligned} \quad (77)$$

With the definition $C_5 = [c_1^2 c_2^2 / 2k_1 (1 - c_1)] + (C_1^2 / 2k_2) + (\kappa Z_M^2 / 2) + \epsilon$, the preceding equation becomes

$$\begin{aligned} \dot{V}_2 \leq & -\frac{k_1}{2} (1 - c_1) \|z_1\|^2 - \frac{k_2}{2} \|z_2\|^2 + C_2 \|z_2\| \|\tilde{Z}\|_F - \frac{\kappa}{2} \|\tilde{Z}\|_F^2 \\ & + C_5 = -\frac{k_1}{2} (1 - c_1) \|z_1\|^2 - \frac{k_2}{4} \|z_2\|^2 - \frac{\kappa}{4} \|\tilde{Z}\|_F^2 \\ & - \frac{1}{2} \left[\begin{array}{cc} \|z_2\| \\ \|\tilde{Z}\|_F \end{array} \right]^T \left[\begin{array}{cc} \frac{k_2}{2} & -C_2 \\ -C_2 & \frac{\kappa}{2} \end{array} \right] \left[\begin{array}{c} \|z_2\| \\ \|\tilde{Z}\|_F \end{array} \right] + C_5 \end{aligned} \quad (78)$$

If k_2 and κ are chosen to satisfy $k_2 \kappa - 4C_2^2 > 0$, the last matrix in the preceding equation becomes positive definite. Therefore, the following inequality is satisfied:

$$\dot{V}_2 \leq -(k_1/2)(1 - c_1) \|z_1\|^2 - (k_2/4) \|z_2\|^2 - (\kappa/4) \|\tilde{Z}\|_F^2 + C_5 \quad (79)$$

For a positive constant μ_2 satisfying $0 < \mu_2 < \min\{(k_1/2)(1 - c_1), k_2/4, \kappa/4 \min\{\gamma_w, \gamma_v\}\}$, finally, the following equation is obtained for the derivative of the chosen Lyapunov candidate function:

$$\dot{V}_2 \leq -2\mu_2 V_2 + C_5 \quad (80)$$

Equation (80) implies that $\dot{V}_2 < 0$ when $V_2 > C_5/2\mu_2$. Hence, the error states z_1 and z_2 and the weight estimation error Z are bounded and converge exponentially to the residual set \mathcal{D}_2 :

$$\mathcal{D}_2 = \{z_1, z_2 \in \mathcal{R}^3, \tilde{Z}_F \in \mathcal{R}^{N_1 + N_2 + 2 \times N_2 + N_3} \mid \|z_1\|^2 + \|z_2\|^2 + (1/\max\{\gamma_w, \gamma_v\})\|\tilde{Z}_F\|_F^2 \leq C_5/\mu_2\}$$

Because c_1, c_2, C_1, Z_M , and ϵ are independent of k_1 and k_2 , the size of the set \mathcal{D}_2 can be made arbitrarily small by adjusting the design parameters k_1 and k_2 . \square

Theorem 2 represents the design procedure of the adaptive controller to track the α, β , and ϕ commands when the modeling errors exist. This shows that, if the controller is applied, the tracking errors and the parameter estimation error of the neural networks converge to a compact set and also shows that the size of the set is adjustable by tuning the design parameters.

The universal approximation theorem only guarantees the existence of the ideal weight and the ideal number of the hidden layer neurons. In this paper, the weight of neural networks are adjusted by the adaptive laws; however, the size of hidden layer neuron N_2 is fixed. It also affects the approximation capacity of the neural networks. If too small a value of N_2 is chosen, the neural networks may not compensate for the effect of the modeling error properly. Therefore, the value of N_2 should be chosen carefully in consideration of the complexity and the size of the modeling error.

V. Numerical Simulation

In Sec. III, the design methodology of the flight controller to track the α, β , and ϕ commands is proposed when full knowledge of the aerodynamic characteristics is available, and in Sec. IV, the adaptive controller based on neural networks is designed to eliminate the effect caused by the aerodynamic modeling error. This section presents numerical simulation results for each controller to demonstrate the performance of the proposed nonlinear control laws.

The F-16 aircraft model is used in this paper,¹¹ and the following command values of α, β , and ϕ are applied to the aircraft in a steady-state level flight of $V = 500$ ft/s, $h = 10,000$ ft:

$$\begin{aligned} \alpha_d &= 2.659, & \beta_d &= 0, & \phi_d &= 0 \text{ deg}, & 0 \leq t \leq 1 \text{ s} \\ \alpha_d &= 10, & \beta_d &= 0, & \phi_d &= 50 \text{ deg}, & 1 \leq t \leq 10 \text{ s} \\ \alpha_d &= -2, & \beta_d &= 0, & \phi_d &= 0 \text{ deg}, & 10 \leq t \leq 20 \text{ s} \end{aligned}$$

To obtain differentiable commands satisfying Assumption 1, the third-order linear command filter is used. The aerodynamic modeling error is made arbitrarily, and the average error for each coefficient is listed in Table 1. The controller design parameters are chosen as follows: $k_1 = 3, k_2 = 8, \kappa = 0.2, k_v = 0.153, \epsilon = 0.001, Z_M = 0.6142$, and $\gamma_w = \gamma_v = 30$.

Figure 3 represents the simulation results of the backstepping controller described in Sec. III when the exact aerodynamic model is available. The solid line represents the simulation result of the backstepping controller proposed in Sec. II, and the dotted line represents the command signal. As expected, the proposed backstepping controller makes the α, β , and ϕ follow the command values in a satisfactory way. Figure 4 shows the simulation results when the modeling error exists. The solid lines represent the simulation result of the adaptive controller based on neural networks in Sec. IV, and the dashed lines represent the result of the controller in Sec. III. The dotted lines represent the command signals. It is shown that the system output of the adaptive controller tracks the command quite well, even if large modeling uncertainty exists. It can be said that the performance of the system is not degraded in the case of the neural networks adaptive controller.

Table 1 Average modeling errors of the aerodynamic coefficients, %

| Coefficients | Error | Coefficients | Error | Coefficients | Error |
|--------------------|-------|--------------------|-------|--------------------|-------|
| C_l | 79.6 | C_m | 207 | C_n | 180.1 |
| C_{l_p} | 16.0 | C_{m_q} | 77.6 | C_{n_p} | 86.7 |
| C_{l_r} | 148.9 | $C_{m_{\delta_e}}$ | 146.2 | C_{n_r} | 94.9 |
| $C_{l_{\delta_a}}$ | 141.8 | — | — | $C_{n_{\delta_a}}$ | 48.3 |
| $C_{l_{\delta_r}}$ | 69.8 | — | — | $C_{n_{\delta_r}}$ | 228.5 |

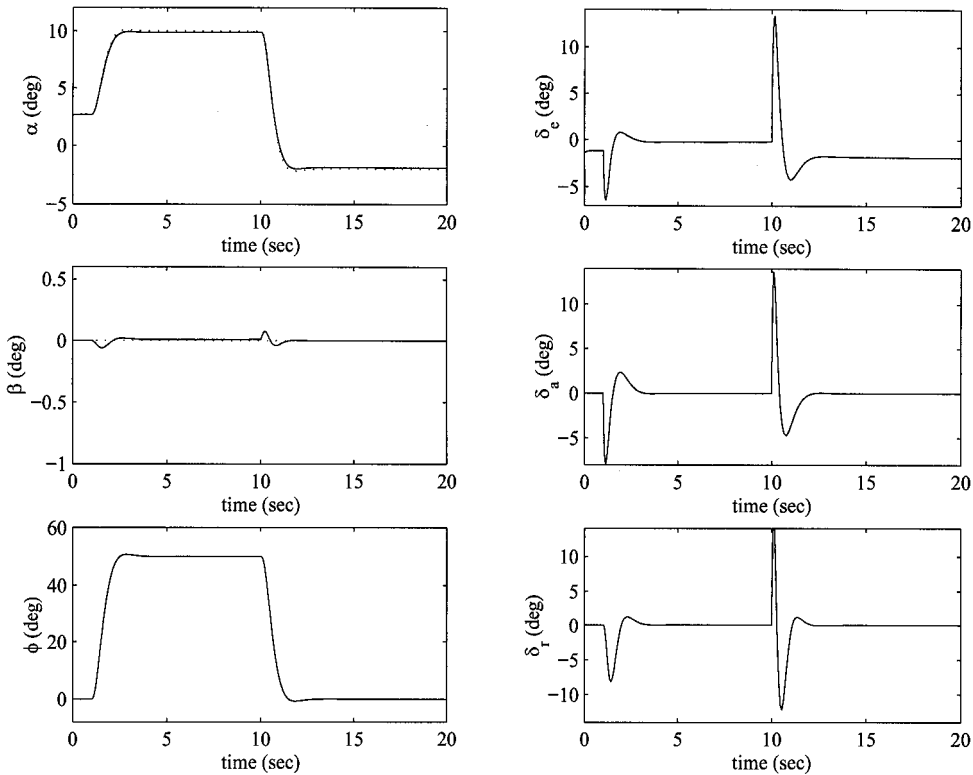


Fig. 3 Time response of x_1 and u (without modeling error): backstepping controller in Sec. III —, and command ····.

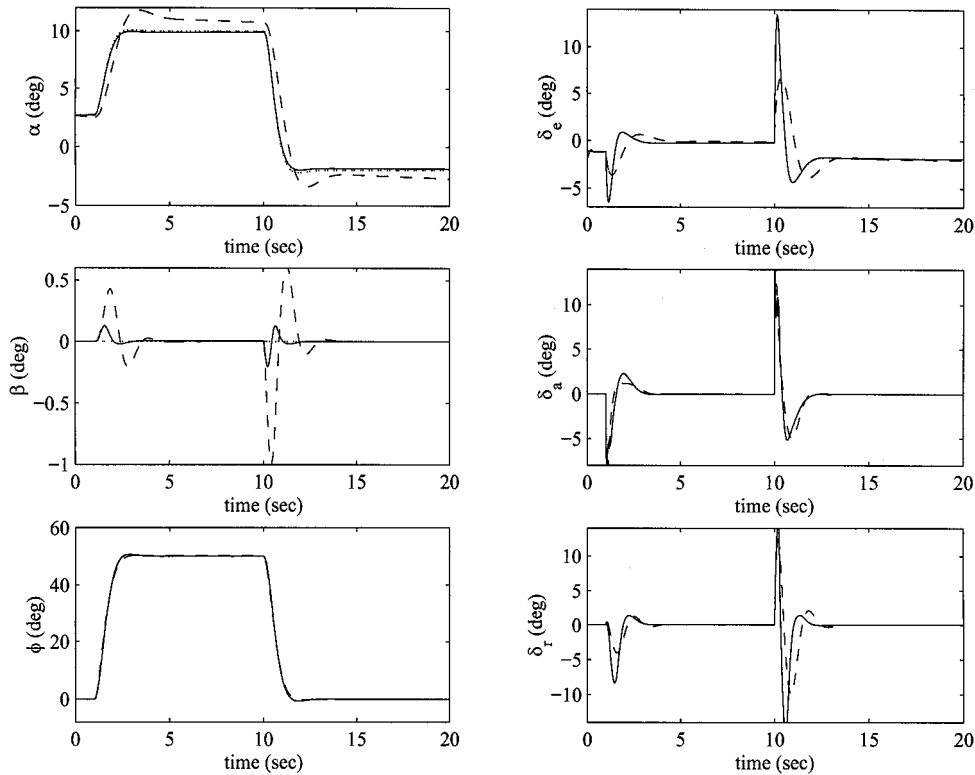


Fig. 4 Time response of x_1 and u (with modeling error): adaptive controller in Sec. IV — , backstepping controller in Sec. III - - -, and command

VI. Conclusions

A controller for a six-degree-of-freedom nonlinear flight model is proposed, and its stability is analyzed by using the Lyapunov theory. The backstepping controller is used to track the α , β , and ϕ commands with the assumption that the aerodynamic characteristics are fully understood. It is shown that, if the controller is applied, the tracking error exponentially converges to a compact set and the size of the set can be made arbitrarily small by tuning the design parameters. It is not necessary to make the controller gain excessively large to guarantee stability because the timescale separation assumption is not used.

An adaptive controller based on neural networks is used to compensate for the effects of the aerodynamic modeling errors. The neural networks' parameters are adjusted to offset the error term. The closed-loop stability of the error states and the parameters of the neural networks are examined by the Lyapunov theory, and it is shown that the error states and the parameter estimation errors exponentially converge to a compact set whose size is adjustable by the design parameters. Finally, a nonlinear simulation of an aircraft maneuver is performed to demonstrate the performance of the proposed control laws.

Acknowledgment

This research was supported by the Korea Science and Engineering Foundation Grant 1999-2-305-004-3.

References

¹Lane, S. H., and Stengel, R. F., "Flight Control Design Using Non-Linear Inverse Dynamics," *Automatica*, Vol. 24, No. 4, 1988, pp. 471-483.
²Menon, P., Badgett, M., and Walker, R., "Nonlinear Flight Test Trajec-

tory Controllers for Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 1, 1987, pp. 67-72.
³Snell, S. A., Enns, D. F., and Garrard, W. L., Jr., "Nonlinear Inversion Flight Control for a Supermaneuverable Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 976-984.
⁴Schumacher, C., and Khargonekar, P. P., "Stability Analysis of a Missile Control System with a Dynamic Inversion Controller," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 3, 1998, pp. 508-515.
⁵Farrell, J. A., "Stability and Approximator Convergence in Nonparametric Nonlinear Adaptive Control," *IEEE Transactions on Neural Networks*, Vol. 9, No. 5, 1998, pp. 1008-1020.
⁶Lewis, F. L., Yesildirek, A., and Liu, K., "Multilayer Neural-Net Robot Controller with Guaranteed Tracking Performance," *IEEE Transactions on Neural Networks*, Vol. 7, No. 2, 1996, pp. 388-399.
⁷Singh, S. N., Yim, W., and Wells, W. R., "Direct Adaptive and Neural Control of Wing-Rock Motion of Slender Delta Wings," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 1, 1995, pp. 25-30.
⁸Kim, B. S., and Calise, A. J., "Nonlinear Flight Control Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 26-33.
⁹Krstić, M., Kanellakopoulos, I., and Kokotović, P., *Nonlinear and Adaptive Control Design*, Wiley, New York, 1995, Chap. 2.
¹⁰Stevens, B. L., and Lewis, F. L., *Aircraft Control and Simulation*, Wiley, New York, 1992, Chap. 2.
¹¹Morelli, E. A., "Global Nonlinear Parametric Modeling with Application to F-16 Aerodynamics," *Proceedings of the 1998 American Control Conference*, IEEE Publications, Piscataway, NJ, 1998, pp. 997-1001.
¹²Hornik, K., Stinchcombe, M., and White, H., "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks*, Vol. 2, No. 5, 1989, pp. 359-366.
¹³Funahashi, K., "On the Approximate Realization of Continuous Mapping by Neural Networks," *Neural Networks*, Vol. 2, No. 3, 1989, pp. 183-192.