

Research Article

Robust Adaptive Neural Control of Morphing Aircraft with Prescribed Performance

Zhonghua Wu,¹ Jingchao Lu,¹ Jingping Shi,¹ Yang Liu,¹ and Qing Zhou²

¹School of Automation, Northwestern Polytechnical University, Xi'an 710072, China

²Xi'an Aeronautics Computing Technique Research Institute, AVIC, Xi'an 710068, China

Correspondence should be addressed to Zhonghua Wu; 463897575@qq.com

Received 14 October 2016; Revised 28 March 2017; Accepted 11 April 2017; Published 17 May 2017

Academic Editor: Asier Ibeas

Copyright © 2017 Zhonghua Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study proposes a low-computational composite adaptive neural control scheme for the longitudinal dynamics of a swept-back wing aircraft subject to parameter uncertainties. To efficiently release the constraint often existing in conventional neural designs, whose closed-loop stability analysis always necessitates that neural networks (NNs) be confined in the active regions, a smooth switching function is presented to conquer this issue. By integrating minimal learning parameter (MLP) technique, prescribed performance control, and a kind of smooth switching strategy into back-stepping design, a new composite switching adaptive neural prescribed performance control scheme is proposed and a new type of adaptive laws is constructed for the altitude subsystem. Compared with previous neural control scheme for flight vehicle, the remarkable feature is that the proposed controller not only achieves the prescribed performance including transient and steady property but also addresses the constraint on NN. Two comparative simulations are presented to verify the effectiveness of the proposed controller.

1. Introduction

Morphing aircraft has received considerable interest, since it possesses distinct advantages, which is capable of altering autonomously its aerodynamic configuration to obtain optimal flight performance, adapting different flight environments and high efficiency executing multiple types of missions [1]. As regards the control issue of morphing vehicle, the key point is to design a flight control system that has capability to guarantee the stability of the aircraft [2, 3]. The difficulty associated with the control design of such system arises from the fact that morphing aircraft manifests time-varying characteristic of aerodynamic forces, moments, and mass distribution as well as strong nonlinear nature [4, 5]. In the literature, several effective methods have been presented to tackle the control problem of folding-wing and swept-back wing aircraft. A multiloop control structure comprised of linear inner-loop and outer-loop controller is proposed for a kind of folding-wing aircraft [4]; similarly, the idea can also be found in [1] depending on gain self-scheduled H_∞ technique. Subsequently, a finite-time boundedness control approach [2]

and a switching linear parameter varying method [6] are investigated for swept-back wing aircraft, respectively. The common feature in [1, 2, 4, 6] is that the controllers are capable of ensuring the aircraft flight steady subject to the wing shape changes, but those designs are highly dependent on the precise prior knowledge of the dynamic model. However, the aerodynamic forces and moments are also quite difficult to model accurately. Moreover, general aircraft dynamics possess strong nonlinearities and uncertainties, which have necessitated the use of nonlinear control methods. Therefore, designing a nonlinear control method independent of prior knowledge of the aerodynamic model for the morphing aircraft is still an interesting yet challenging problem.

Adaptive back-stepping method has been widely studied in tracking control designs for nonlinear systems in strict-feedback or pure-feedback form, because it owns capability of systematically manipulating mismatched uncertainties [7–10]. Lately, some significant works regarding adaptive neural/fuzzy control for nonlinear systems with totally unknown or nonlinearly parameterized nonlinearities are

investigated [11–13], and the noticeable problem of “explosion of items” is elegantly avoided by virtue of dynamic surface control or nonlinear differentiator technique [7, 14–16]. More specifically, adaptive neural/fuzzy approaches are also extensively utilized to the control problem of flight vehicles accompanied with aerodynamic parameters uncertainty in [17–20]. Although the previous adaptive neural control methods have the ability to guarantee the steady-state performance converging to a small residual set, few results consider the transient performance related to overshoot and convergence rate [21]. More recently, the prescribed performance control (PPC) scheme which denotes that the tracking error should converge to a predefined bound with convergence rate no less than a certain value has been an active research area [21–30]. Based on an error transformation technique that incorporates the desirable performance, Bechlioulis and Rovithakis firstly presented an adaptive PPC method for strict-feedback nonlinear systems [22]. Afterwards, an observer based fuzzy adaptive prescribed performance tracking approach is investigated for nonlinear stochastic systems subject to input saturation [26]. Subsequently, the PPC approach has been extended to deal with MIMO state and output feedback nonlinear control problems [27–30]. Moreover, this tool has also been successfully applied to flight control area [24, 31].

Despite the recent progress in the neural networks control of unknown nonlinear systems, certain issues still remain open. In practice, even though the stability analysis of aforementioned adaptive neural control schemes is proven, it relies on the condition that the approximation ability of NN must be effective all the time (i.e., the NN should be permanently working in the neural active region). Therefore, the deterioration of the tracking performance or even instability may happen provided that the transient states overstep the neural active region. Additionally, such a condition is also difficult to verify beforehand in real applications [32]. This difficulty can be naturally eliminated in case the invariance of the neural active region is guaranteed. In [33], by introducing n error transformation functions into the back-stepping design, a priori guaranteed evolution within the NN approximation set methodology is proposed to conquer this issue. However, the method developed in [33] constitutes an overparameterized solution which is hard to implement in practical application. Another alternative approach is designing a smooth switching function, which has the capability of switching the normal neural controller to a robust controller for pulling back the escaped transient into the neural effective regions. Several smooth switching function based neural adaptive back-stepping control schemes are proposed to solve aforementioned problem, where each virtual controller contains a neural controller and a robust controller effectively working inside or outside the active region, respectively [32, 34–37]. However, numerous adaptive parameters still need to be updated online and the transient performance problem is also omitted in those methods.

Motivated by the aforementioned discussion, a switching strategy based composite adaptive neural control scheme is proposed for a swept-wing morphing aircraft. NNs are employed to approximate unknown functions; thus a priori knowledge of the aerodynamic parameters is not necessary.

It is proven that all the signals in the closed-loop systems are bounded. The main contributions of this work are shown as follows:

- (1) Different from the works [32, 34–37] which completely neglect the transient performance related to overshoot and convergence rate, by introducing an error transformation, the proposed controller is capable of allowing attributes such as a lower bound on the convergence rate and steady error to be specified. The MLP and FOSD are incorporated into the neural back-stepping design to reduce the online updating parameters of NN and to conquer the problem of “explosion of the items,” thus deriving a low-computational scheme.
- (2) In contrast to traditional neural control schemes [21–24, 26–30] whose stability analysis relies on the condition that the NNs should always be kept within the neural active region, a smooth switching function based composite control scheme, presented to manipulate the exchange of control authorities between normal neural controller working in the active region and a robust controller out of this scope, is constructed to relax this constraint, and a new kind of adaptive laws is proposed. Note that it is also the first low-computational neural control scheme which can efficiently address the prescribed performance issue and relax the constraint on NN simultaneously.

2. Model Dynamics and Problem Formulation

2.1. Morphing Aircraft Model. The longitudinal dynamics of a morphing aircraft considered in this study are derived from [5, 38]. This model includes state variables $(V, h, \alpha, \gamma, q)$ and control inputs (δ_e, T) , where V denotes the velocity, h is the altitude, α denotes angle of attack, γ represents the flight path angle (FPA), and q is the pitch rate; δ_e and T represent elevator deflection and thrust, respectively.

$$\dot{V} = \frac{-D + T \cos \alpha - mg \sin \gamma + F_{Ix}}{m}, \quad (1)$$

$$\dot{h} = V \sin \gamma, \quad (2)$$

$$\dot{\gamma} = \frac{L + T \sin \alpha - mg \cos \gamma - F_{IKz}}{mV}, \quad (3)$$

$$\dot{\alpha} = \frac{-L - T \sin \alpha + mg \cos \gamma + F_{Iz}}{mV} + q, \quad (4)$$

$$\dot{q} = \frac{-\dot{I}_y q - S_x g \cos \theta + M_A + TZ_T + M_{Iy}}{I_y}, \quad (5)$$

$$\begin{aligned} F_{Ix} &= S_x (\dot{q} \sin \alpha + q^2 \cos \alpha) + 2\dot{S}_x q \sin \alpha - \ddot{S}_x \cos \alpha, \\ F_{Iz} &= F_{IKz} \\ &= S_x (\dot{q} \cos \alpha - q^2 \sin \alpha) + 2\dot{S}_x q \cos \alpha + \ddot{S}_x \sin \alpha, \end{aligned} \quad (6)$$

$$M_{Iy} = S_x (\dot{V} \sin \alpha + V \dot{\alpha} \cos \alpha - Vq \cos \alpha),$$

where D , L , and M_A denote drag force, lift force, and pitch moment, respectively. m , I_y , and g denote the mass of aircraft, moment of inertia about pitch axis, and gravity constant. F_{I_x} , F_{I_z} , $F_{I_{kz}}$, and M_{I_y} are the inertial force and moment caused by morphing process. Z_T is the position of engine in the body axis. S_x is the static moment caused by wing sweep. The related definitions are given as follows:

$$\begin{aligned}
C_D(\zeta) &= C_{D0}(\zeta) + C_{D\alpha}(\zeta)\alpha + C_{D\alpha^2}(\zeta)\alpha^2, \\
S_x(\zeta) &\approx [2m_1r_{1x} + m_3r_{3x}], \\
L &= C_L(\zeta)QS_w(\zeta), \\
D &= C_D(\zeta)QS_w(\zeta), \\
C_m(\zeta) &= C_{m0}(\zeta) + C_{m\alpha}(\zeta)\alpha + C_{m\delta_e}(\zeta)\delta_e \\
&\quad + \frac{C_{mq}qc_A(\zeta)}{2V}, \\
Q &= \frac{1}{2}\rho_h V^2; \\
C_L(\zeta) &\approx C_{L0}(\zeta) + C_{L\alpha}(\zeta)\alpha, \\
M_A &= C_m(\zeta)QS_w(\zeta)c_A(\zeta),
\end{aligned} \tag{7}$$

where ζ is the sweep angle; the detailed explanation of the other parameters can be found in [5].

2.2. Model Transformation and Control Objective. Thrust T mainly affects velocity V , and elevator deflection δ_e has a dominant contribution to altitude h change; thus the dynamics model is reasonably decomposed into two subsystems including altitude and velocity subsystems.

2.2.1. Altitude Subsystem. Define $x_1 = h$, $x_2 = \gamma$, $x_3 = \theta$, and $x_4 = q$, where $\theta = \alpha + \gamma$ and $\bar{x} = (x_2, x_3, x_4)$. Therefore, the altitude subsystem can be converted into the following formulation:

$$\begin{aligned}
\dot{x}_1 &= Vx_2, \\
\dot{x}_2 &= f_2(x_2, x_3) + x_3, \\
\dot{x}_3 &= x_4, \\
\dot{x}_4 &= f_4(\bar{x}, \delta_e) + u, \\
y &= x_1, \\
u &= -\delta_e,
\end{aligned} \tag{8}$$

where y is the output signal of altitude subsystem (8); $f_2(x_2, x_3) = [L + T \sin(\theta - \gamma) - mg \cos \gamma]/(mV) - \theta$ and $f_4(\bar{x}, u) = [-\dot{I}_y q + (-S_x g \cos \theta + M_A + TZ_T + M_{I_y})]/I_y + \delta_e$ are unknown functions.

2.2.2. Velocity Subsystem. Velocity subsystem is transformed into the following formulation:

$$\dot{V} = f_V(x_V, T) + T, \tag{9}$$

where $f_V(x_V, T) = (-D + T \cos \alpha - mg \sin \gamma + F_{I_x})/m - T$ is an unknown function.

Remark 1. In order to transform the altitude system into pure-feedback system, $F_{I_{kz}}$ in (3) is regarded as an unmodeled term. Since we only consider the cruise phase in this paper, γ is quite small and we can take $\sin \gamma \approx \gamma$ in (2) to simplify the system.

Control Objective. The control objective in this study is to design adaptive controllers δ_e and T such that

- (1) the altitude and velocity can track the desired trajectory h_d and V_d , while guaranteeing that all the signals in the closed loop are bounded;
- (2) the corresponding altitude and velocity tracking errors achieve prescribed transient and steady-state performance.

2.3. Some Preliminaries

2.3.1. Prescribed Performance. To achieve the control objective, the tracking error $z_i(t)$, $i = 1, V$ should satisfy the following prescribed performance bounds [22, 39]:

$$-\lambda_i(t) < z_i(t) < \lambda_i(t), \tag{10}$$

where $\lambda_i(t) > 0$ named performance function is defined as

$$\lambda_i(t) = (\lambda_{i0} - \lambda_{i\infty}) \exp(-l_i t) + \lambda_{i\infty}, \tag{11}$$

where λ_{i0} , $\lambda_{i\infty}$, and l_i are positive constants; $\lambda_{i0} = \lambda_i(0)$ and $\lambda_{i\infty} = \lim_{t \rightarrow \infty} \lambda_i(t)$; l_i denotes the minimum speed of convergence and $\lambda_{i\infty}$ is the maximum steady-state error.

To transform the constrained tracking error condition (10) into an equivalent unconstrained one, the following state transformation is employed. So we have

$$\mu_i(t) = R_i \left(\frac{z_i(t)}{\lambda_i(t)} \right), \tag{12}$$

where $\mu_i(t)$ is the transformed error and $R_i(\cdot)$ is an increasing transformation function shown as follows:

$$R_i \left(\frac{z(t)}{\lambda(t)} \right) = \ln \left(\frac{(1 + z_i(t)/\lambda_i(t))}{1 - z_i(t)/\lambda_i(t)} \right). \tag{13}$$

The derivative of (12) is shown as

$$\dot{\mu}_i(t) = r_i \left(\dot{z}_i(t) - \frac{\dot{\lambda}_i(t)}{\lambda_i(t)} z_i(t) \right), \tag{14}$$

where $r_i = (\partial R_i / \partial (z_i(t)/\lambda_i(t))) (1/\lambda_i(t))$.

Using (13), the following inequalities can be obtained:

$$r_i > r_{i\min} > 0, \tag{15}$$

where $r_{i\min} = 2/(\lambda_i(0))$.

2.3.2. Useful Function and Key Lemmas

Definition 2 (see [35]). The boundaries of the compact subsets Ω_i are defined by several prescribed constants $0 < r_{i1} < r_{i2}$, $i = 2, 4$; meanwhile some useful switching functions are described as

$$m_i(\bar{x}_i) \triangleq \prod_{k=1}^i B_k(x_k),$$

$$B_k(x_k) \triangleq \begin{cases} 1, & \text{if } |x_k| < r_{k1}, \\ \frac{r_{k2}^2 - x_k^2}{r_{k2}^2 - r_{k1}^2} e^{-((r_{k2}^2 - x_k^2)/((r_{k2}^2 - r_{k1}^2)\bar{\omega}_k))^{2b_1}}, & \text{if } r_{k1} \leq |x_k| \leq r_{k2}, \\ 0, & \text{if } |x_k| > r_{k2}, \end{cases} \quad (16)$$

where $\bar{\omega}_k > 0$ and $b_1 > 0$ are positive constants.

Lemma 3 (see [40]). *The following inequality holds for any $\omega_0 > 0$ and $\eta \in \mathbb{R}$:*

$$0 \leq |\eta| - \eta \tanh\left(\frac{\eta}{\omega_0}\right) \leq \kappa_0 \omega_0, \quad (17)$$

where κ_0 is a constant satisfying $\kappa_0 = e^{-(\kappa_0+1)}$; that is, $\kappa_0 = 0.2785$.

Lemma 4 (see [41]). *The “first-order sliding mode differentiator (FOSD)” is designed as*

$$\begin{aligned} \dot{c}_0 &= -\bar{\mu}_0 |c_0 - l(t)|^{0.5} \text{sign}(c_0 - l(t)) + c_1, \\ \dot{c}_1 &= -\bar{\mu}_1 \text{sign}(c_1 - c_0), \end{aligned} \quad (18)$$

where c_0 and c_1 are the states of system (18), $\bar{\mu}_0$ and $\bar{\mu}_1$ are the designed parameters of FOSD, and $l(t)$ is an input function. \dot{c}_0 can estimate $\dot{l}(t)$ to an arbitrary precision in case the initial values $c_0 - l(t_0)$ and $\dot{c}_0 - \dot{l}(t_0)$ are bounded.

3. Controller Design

In order to process the derivation, motivated by [8, 12], filtered signals are used to circumvent algebraic loop problems encountered in the following design; thus we define

$$\begin{aligned} \Delta f_2 &= f_2(x_2, x_3) - f_2(x_2, x_{3f}), \\ \Delta f_4 &= f_4(\bar{x}, u) - f_4(\bar{x}, u_f), \\ \Delta f_V &= f_V(x_V, T) - f_V(x_V, T_f), \end{aligned} \quad (19)$$

where x_{3f} , u_f , and T_f are the filtered signals defined by [8]:

$$\begin{aligned} x_{3f} &= H_L(s) x_3 \approx x_3, \\ u_f &= H_L(s) u \approx u, \\ T_f &= H_L(s) T \approx T, \end{aligned} \quad (20)$$

where $H_L(s)$ is a Butterworth low-pass filter. The corresponding filter parameters of Butterworth filters can be obtained in [8].

Assumption 5. In this paper, we assume that all of the system states are measurable.

Assumption 6. The functions $f_i(\cdot)$, $i = 2, 4$, are unknown and are bounded by $|f_i(\cdot)| \leq f_i^u(\cdot)$, where $f_i^u(\cdot)$ are known nonnegative smooth functions. Meanwhile, it is also assumed that Δf_i are bounded.

Obviously, there exist ideal weight vectors W_2^* , W_4^* , and W_V^* such that

$$\begin{aligned} f_2(x_2, x_{3f}) &= W_2^{*T} \Phi_2(x_2, x_{3f}) + \varepsilon_2, \quad |\varepsilon_2| \leq \varepsilon_{2M}, \\ f_4(\bar{x}, u_f) &= W_4^{*T} \Phi_4(\bar{x}, u_f) + \varepsilon_4, \quad |\varepsilon_4| \leq \varepsilon_{4M}, \\ f_V(x_V, T_f) &= W_V^{*T} \Phi_V(x_V, T_f) + \varepsilon_V, \quad |\varepsilon_V| \leq \varepsilon_{VM}, \end{aligned} \quad (21)$$

where ε_i and ε_{iM} denote the approximation errors and their upper bounds, respectively. W_i^* is the weight of NN. $\Phi_i(\cdot) = [\phi_1(\cdot), \dots, \phi_n(\cdot)]$ is the basis function vector with $\phi_i(\cdot) = \exp(-(\cdot - c_i)^T(\cdot - c_i)/b_i^2)$, wherein c_i and b_i are the centers and widths of $\phi_i(\cdot)$. Obviously, the ideal weights W_2^* , W_4^* , and W_V^* are completely unknown. Thus, the MLP technique is employed to estimate the norm of W_2^* , W_4^* , and W_V^* to reduce the computation burden. Those parameters are defined as $\varphi_i = \|W_i^*\|^2$ ($i = 2, 4, V$). In the following, we replace $\varphi_i(\cdot)$ with φ_i to simplify the expression.

3.1. Velocity Controller Design. Define velocity tracking error as

$$z_V = V - V_d. \quad (22)$$

The time derivative of z_V can be described as

$$\dot{z}_V = f_V + T - \dot{V}_d. \quad (23)$$

According to (14) and (23), the time derivation of the transformed error $\mu_V(t)$ is shown as

$$\begin{aligned} \dot{\mu}_V(t) &= r_V \left(\dot{z}_V - \frac{\dot{\lambda}_V}{\lambda_V} z_V \right) \\ &= r_V \left(f_V + T - \dot{V}_d - \frac{\dot{\lambda}_V}{\lambda_V} z_V \right), \end{aligned} \quad (24)$$

where $r_V = (\partial R_V / \partial (z_V / \lambda_V)) (1 / \lambda_V) > r_{V\min} > 0$ and $\lambda_V(t) = (\lambda_{V0} - \lambda_{V\infty}) \exp(-l_V t) + \lambda_{V\infty}$.

By employing MLP technique, the controller T_d is designed as

$$\begin{aligned} T_d &= - \left(k_{V1} - \frac{\hat{r}_V}{2r_V^2} \right) \mu_V - \frac{1}{2} \mu_V \hat{\varphi}_V \Phi_V^T \Phi_V \\ &\quad - \hat{d}_V \tanh\left(\frac{\mu_V}{\omega_{V1}}\right) + \dot{V}_d + \frac{\dot{\lambda}_V}{\lambda_V} z_V, \end{aligned} \quad (25)$$

where k_{V1} and ω_{V1} are positive design parameters. $\hat{\varphi}_V$ and \hat{d}_V denote the estimation of φ_V and d_{VM} , respectively. $d_V = \Delta f_V + \varepsilon_V$ is the lump approximation error with $|d_V| \leq d_{VM}$

and \hat{r}_V denotes the estimation of \dot{r}_V by means of sliding mode differentiator. According to Lemma 4, we can easily obtain $|\hat{r}_V - \dot{r}_V| \leq \bar{l}$ with $\bar{l} > 0$.

Consider the following adaptive laws for $\hat{\varphi}_V$ and \hat{d}_V :

$$\begin{aligned}\dot{\hat{\varphi}}_V &= \frac{\rho_{V1}}{2} (\mu_V^2 \Phi_V^T \Phi_V - 2\sigma_{V1} \hat{\varphi}_V), \\ \dot{\hat{d}}_V &= \rho_{V2} \left[\mu_V \tanh\left(\frac{\mu_V}{\omega_{V1}}\right) - \sigma_{V2} \hat{d}_V \right],\end{aligned}\quad (26)$$

where ρ_{V1} , ρ_{V2} , σ_{V1} , and σ_{V2} denote positive design parameters.

Theorem 7. *Suppose that the velocity subsystem (9) satisfies Assumption 5; if the adaptive controller is selected as (25) and updating laws are selected as (26), the signals including μ_V , $\hat{\varphi}_V$, and \hat{d}_V are ensured to be bounded.*

Remark 8. The velocity design is partially derived from [24]. Note that the FOSD is used to estimate unknown item \dot{r}_V . By introducing $0.5\hat{r}_V\mu_V/r_V^2$ in (25), the stability analysis problem in [24] is overcome.

3.2. Altitude Controller Design. The following coordinate change is constructed to facilitate the control design:

$$\begin{aligned}z_1 &= x_1 - y_d, \\ z_2 &= x_2 - \alpha_1, \\ z_3 &= x_3 - \alpha_2, \\ z_4 &= x_4 - \alpha_3,\end{aligned}\quad (27)$$

where α_1 , α_2 , and α_3 are the virtual controllers to be designed at Steps 1, 2, and 3, respectively. $y_d = h_d$ is the reference signal. The control scheme for the altitude subsystem is developed in the framework of back-stepping technique, which contains 4-step recursive design procedure.

Step 1. The time derivative of $z_1 = x_1 - y_d$ is expressed as

$$\dot{z}_1 = \dot{x}_1 - \dot{y}_d = Vx_2 - \dot{y}_d = V(z_2 + \alpha_1) - \dot{y}_d. \quad (28)$$

By using (15) and (28), the time derivative of the transformed altitude error $\mu_1(t)$ is shown as follows:

$$\dot{\mu}_1(t) = r_1 \left(\dot{z}_1 - \frac{\dot{\lambda}_1}{\lambda_1} z_1 \right) = r_1 \left(Vx_2 - \dot{y}_d - \frac{\dot{\lambda}_1}{\lambda_1} z_1 \right), \quad (29)$$

where $r_1 = (\partial R_1 / \partial (z_1 / \lambda_1)) (1 / \lambda_1) > r_{1\min} > 0$ and $\lambda_1(t) = (\lambda_{10} - \lambda_{1\infty}) \exp(-l_1 t) + \lambda_{1\infty}$.

The virtual controller α_1 is designed as

$$\alpha_1 = \frac{-(k_1 - \dot{r}_1 / 2r_1^2) \mu_1 + \dot{y}_d + (\dot{\lambda}_1 / \lambda_1) z_1}{V}, \quad (30)$$

where k_1 is positive parameter. It is worth noticing that \dot{r}_1 can be easily obtained via system states.

Invoking (28) and (30), one has

$$\dot{\mu}_1 = r_1 \left(Vz_2 - \left(k_1 - \frac{\dot{r}_1}{2r_1^2} \right) \mu_1 \right). \quad (31)$$

In order to avoid the tedious computation of $\dot{\alpha}_1$, the following FOSD is adopted to estimate it:

$$\begin{aligned}\dot{\varsigma}_{10} &= -\mu_{10} |\varsigma_{10} - \alpha_1|^{0.5} \text{sign}(\varsigma_{10} - \alpha_1) + \varsigma_{11}, \\ \dot{\varsigma}_{11} &= -\mu_{11} \text{sign}(\varsigma_{11} - \varsigma_{10}),\end{aligned}\quad (32)$$

where ς_{10} and ς_{11} are the states of FOSD (32) and μ_{10} and μ_{11} are the positive design constants.

Then, we have

$$\dot{\alpha}_1 = \dot{\varsigma}_{10} + \tau_1, \quad (33)$$

where τ_1 is the estimation error of the FOSD with $|\tau_1| \leq \bar{\tau}_1$.

Step 2. The differentiation of z_2 is obtained as follows:

$$\dot{z}_2 = \dot{x}_2 - \dot{\alpha}_1 = f_2(x_2, x_3) + z_3 + \alpha_2 - \dot{\alpha}_1. \quad (34)$$

The virtual controller α_2 is designed as

$$\alpha_2 = -k_2 z_2 + \dot{\varsigma}_{10} - m_2 u_2^N - (1 - m_2) u_2^r \quad (35)$$

with

$$\begin{aligned}u_2^N &= \frac{1}{2} z_2 \hat{\varphi}_2 \Phi_2^T \Phi_2 + \hat{d}_2 \tanh\left(\frac{z_2}{\omega_{21}}\right), \\ u_2^r &= f_2^u \tanh\left(\frac{z_2 f_2^u}{\omega_{22}}\right),\end{aligned}\quad (36)$$

where k_2 , ω_{21} , and ω_{22} are positive design parameters. $d_2 = \Delta f_2 + \varepsilon_2$ is bounded with $|d_2| \leq d_{2M}$. $\hat{\varphi}_2$ and \hat{d}_2 denote the estimations of φ_2 and d_{2M} , respectively.

The structure of adaptive control laws is expressed as follows:

$$\begin{aligned}\dot{\hat{\varphi}}_2 &= \frac{\rho_{21}}{2} (m_2 z_2^2 \Phi_2^T \Phi_2 - 2\sigma_{21} \hat{\varphi}_2), \\ \dot{\hat{d}}_2 &= \rho_{22} \left[m_2 z_2 \tanh\left(\frac{z_2}{\omega_{21}}\right) - \sigma_{22} \hat{d}_2 \right].\end{aligned}\quad (37)$$

Substituting (35) into (34), (34) can be rewritten as

$$\begin{aligned}\dot{z}_2 &= z_3 - k_2 z_2 + \dot{\varsigma}_{10} - \dot{\alpha}_1 + m_2 (f_2 - u_2^N) + (1 - m_2) \\ &\quad \cdot (f_2 - u_2^r) = z_3 - k_2 z_2 + \dot{\varsigma}_{10} - \dot{\alpha}_1 + m_2 \left(W_2^{*T} \Phi_2 \right. \\ &\quad \left. - \frac{1}{2} z_2 \hat{\varphi}_2 \Phi_2^T \Phi_2 + d_2 - \hat{d}_2 \tanh\left(\frac{z_2}{\omega_{21}}\right) \right) + (1 \\ &\quad - m_2) \left(f_2 - f_2^u \tanh\left(\frac{z_2 f_2^u}{\omega_{22}}\right) \right).\end{aligned}\quad (38)$$

The following FOSD is adopted to estimate $\dot{\alpha}_2$:

$$\begin{aligned}\dot{\varsigma}_{20} &= -\mu_{20} |\varsigma_{20} - \alpha_2|^{0.5} \text{sign}(\varsigma_{20} - \alpha_2) + \varsigma_{21}, \\ \dot{\varsigma}_{21} &= -\mu_{21} \text{sign}(\varsigma_{21} - \varsigma_{20}),\end{aligned}\quad (39)$$

where ς_{20} and ς_{21} are the states of system (32) and μ_{20} and μ_{21} are positive design constants.

From (39) and Lemma 3, we have

$$\dot{\alpha}_2 = \dot{\varsigma}_{20} + \tau_2, \quad (40)$$

where τ_2 is the estimation error with $|\tau_2| \leq \bar{\tau}_2$.

Step 3. The differentiation of z_3 is obtained as follows:

$$\dot{z}_3 = \dot{x}_3 - \dot{\alpha}_2 = z_4 + \alpha_3 - \dot{\alpha}_2. \quad (41)$$

The virtual control law α_3 is designed as

$$\alpha_3 = -k_3 z_3 + \dot{\varsigma}_{20} - z_2, \quad (42)$$

where k_3 is a positive design parameter.

Substituting (42) into (41) yields

$$\dot{z}_3 = z_4 - z_2 - k_3 z_3 + \dot{\varsigma}_{20} - \dot{\alpha}_2 = z_4 - z_2 - k_3 z_3 - \tau_2. \quad (43)$$

As done previously, the following FOSD is employed to estimate $\dot{\alpha}_3$:

$$\begin{aligned}\dot{\varsigma}_{30} &= -\mu_{30} |\varsigma_{30} - \alpha_3|^{0.5} \text{sign}(\varsigma_{30} - \alpha_3) + \varsigma_{31}, \\ \dot{\varsigma}_{31} &= -\mu_{31} \text{sign}(\varsigma_{31} - \varsigma_{30}),\end{aligned}\quad (44)$$

where ς_{30} and ς_{31} are the states of the system and μ_{30} and μ_{31} are the positive design constants.

Thus, we have

$$\dot{\alpha}_3 = \dot{\varsigma}_{30} + \tau_3, \quad (45)$$

where τ_3 is an estimation error with $|\tau_3| \leq \bar{\tau}_3$.

Step 4. In this step, the actual controller u will be developed. The differentiation of z_4 can be obtained as follows:

$$\begin{aligned}\dot{z}_4 &= f_4(\bar{x}, u) + u - \dot{\alpha}_3 \\ &= m_4 f_4(\bar{x}, u) + (1 - m_4) f_4(\bar{x}, u) + u - \dot{\alpha}_3.\end{aligned}\quad (46)$$

The controller u is designed as

$$u = -k_4 z_4 + \dot{\varsigma}_{30} - z_3 - m_4 u_4^N - (1 - m_4) u_4^r \quad (47)$$

with

$$\begin{aligned}u_4^N &= \frac{1}{2} z_4 \hat{\varphi}_4 \Phi_4^T \Phi_4 + \hat{d}_4 \tanh\left(\frac{z_4}{\omega_{41}}\right), \\ u_4^r &= f_4^u \tanh\left(\frac{z_4 f_4^u}{\omega_{42}}\right),\end{aligned}\quad (48)$$

where k_4 , ω_{41} , and ω_{42} are the positive design constants. $d_4 = \Delta f_4 + \varepsilon_4$ is the lump approximation error with $|d_4| \leq d_{4M}$. $\hat{\varphi}_4$

and \hat{d}_4 denote the estimations of φ_4 and d_{4M} , respectively. $\hat{\varphi}_4$ and \hat{d}_4 are updated as

$$\begin{aligned}\dot{\hat{\varphi}}_4 &= \frac{\rho_{41}}{2} (m_4 z_4^2 \Phi_4^T \Phi_4 - 2\sigma_{41} \hat{\varphi}_4), \\ \dot{\hat{d}}_4 &= \rho_{42} \left(m_4 z_4 \tanh\left(\frac{z_4}{\omega_{41}}\right) - \sigma_{42} \hat{d}_4 \right).\end{aligned}\quad (49)$$

Thus, (46) can be rewritten as

$$\begin{aligned}\dot{z}_4 &= -k_4 z_4 - z_3 - \tau_3 + m_4 \left(W_4^{*T} \Phi_4 - \frac{1}{2} z_4 \hat{\varphi}_4 \Phi_4^T \Phi_4 \right. \\ &\quad \left. + d_4 - \hat{d}_4 \tanh\left(\frac{z_4}{\omega_{41}}\right) \right) + (1 - m_4) \left(f_4 \right. \\ &\quad \left. - f_4^u \tanh\left(\frac{z_4 f_4^u}{\omega_{42}}\right) \right).\end{aligned}\quad (50)$$

Theorem 9. Consider the altitude subsystem (8) with Assumptions 5 and 6; if the switching adaptive neural prescribed performance control scheme is selected as (30), (35), (42), and (47), adaptive laws are selected as (37) and (49), and FOSD is selected as (32), (39), and (44), the signals μ_1 , $z_{i=2,3,4}$, $\tilde{\varphi}_{i=2,4}$, and $\tilde{d}_{i=2,4}$ in the closed-loop system are bounded.

Remark 10. The altitude controller, composed of a normal adaptive neural controller working in the neural active region, a robust controller being in charge outside the neural approximation region, and a switching strategy supervising the exchange of the former two controllers, is constructed.

Remark 11. In this paper, in order to estimate the derivative of virtual controllers α_1 , α_2 , and α_3 , the FOSD (first-order sliding mode differentiator) is employed. Using (33) as an example, $\dot{\varsigma}_{10}$ is the estimation of \dot{a}_1 and τ_1 is the estimation error between actual \dot{a}_1 and $\dot{\varsigma}_{10}$. It must be noted that τ_1 is not used in the controller design but is just employed for stability analysis (please see (B.6)).

4. Simulations

In this section, two comparative cases are presented to illustrate the effectiveness of the switching functions based adaptive neural control for longitudinal model of the morphing aircraft. The aerodynamic coefficients and model parameters are the same as [5]. The initial conditions are set as $X_0 = [\gamma_0, \alpha_0, q_0, h_0, V_0] = [0^\circ, 0.99512^\circ, 0^\circ/\text{s}, 1000 \text{ m}, 30 \text{ m/s}]$. The control parameters are selected as $k_1 = 0.3$, $k_2 = 0.3$, $k_3 = 0.05$, $k_4 = 3.75$, and $k_{V1} = 3$. Gains for the adaptive laws are set as $\rho_{21} = 40$, $\sigma_{21} = 0.01$, $\rho_{22} = 10$, $\sigma_{22} = 0.1$, $\rho_{41} = 250$, $\sigma_{41} = 0.01$, $\rho_{42} = 100$, $\sigma_{42} = 0.01$, $\rho_{V1} = 10$, $\sigma_{V1} = 0.1$, $\rho_{V2} = 100$, $\sigma_{V2} = 0.1$, $\omega_{21} = 1$, $\omega_{22} = 1$, $\omega_{41} = 1$, and $\omega_{42} = 5$. The aforementioned transient and steady output error bounds are prescribed by the performance functions $\lambda_i(t) = (\lambda_{i0} - \lambda_{i\infty}) \exp(-l_i t) + \lambda_{i\infty}$, $i = 1, V$, where $\lambda_{10} = 0.5$, $\lambda_{1\infty} = 0.2$, $\lambda_{V0} = 0.35$, $\lambda_{V\infty} = 0.1$, and $l_i = 0.05$. The corresponding neural active regions are defined as $V \in [30 \text{ m/s } 40 \text{ m/s}]$, $x_2 \in [-4 \text{ deg } 4 \text{ deg}]$, $x_3 \in [-6 \text{ deg } 6 \text{ deg}]$,

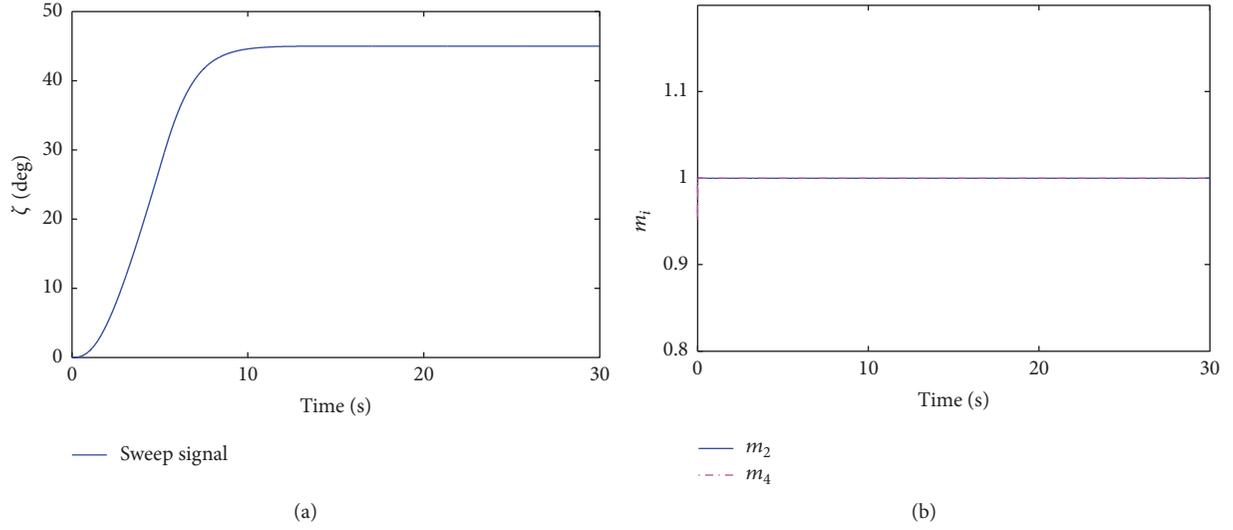


FIGURE 1: Sweep reference and switch signal.

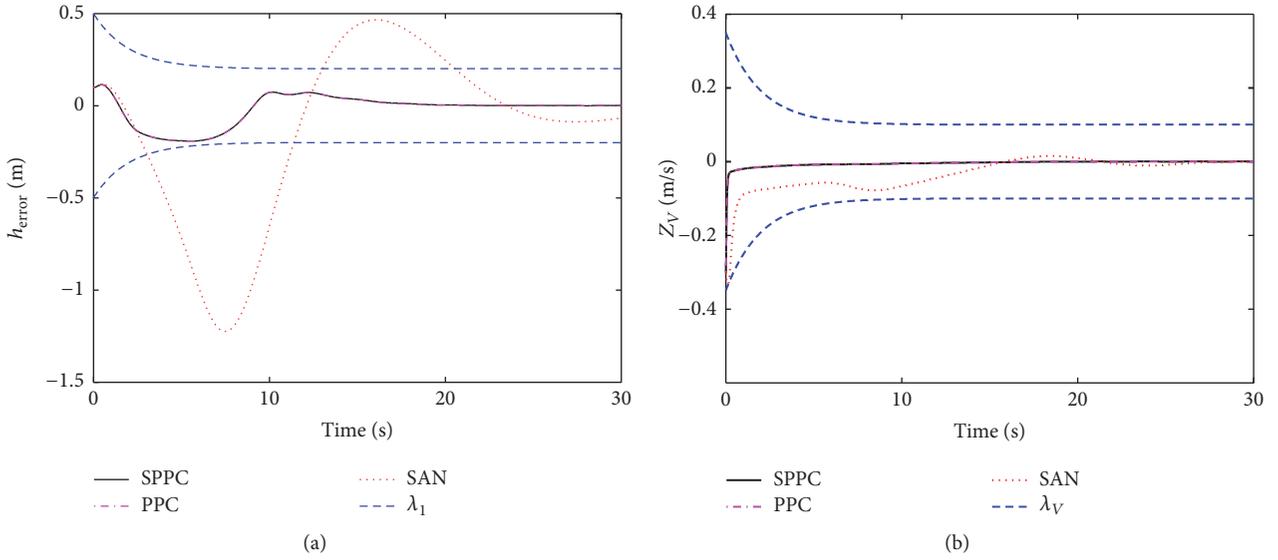


FIGURE 2: Altitude and velocity tracking errors.

and $x_4 \in [-6 \text{ deg/s } 6 \text{ deg/s}]$. The centers such as c_2 , c_4 , and c_V including 50 nodes are evenly spaced in their bounds. The widths of Gaussian functions are chosen as $b_{i2} = 2$, $b_{i4} = 1.5$, and $b_{iV} = 4$. The parameters of switching function are selected as $r_{21} = 4 \text{ deg}$, $r_{22} = 7 \text{ deg}$, $r_{31} = 6 \text{ deg}$, $r_{32} = 10 \text{ deg}$, $r_{41} = 5 \text{ deg/s}$, $r_{42} = 10 \text{ deg/s}$, $\bar{\omega}_{i=1,2,3} = 10$, and $b_1 = 1$. Reference commands are smoothed via several second-order filters which are given in (51).

$$\begin{aligned}
 \frac{h_d}{h_{d0}} &= \frac{0.04}{s^2 + 0.4s + 0.04}, \\
 \frac{V_d}{V_{d0}} &= \frac{0.04}{s^2 + 0.4s + 0.04}, \\
 \frac{\zeta_d}{\zeta_{d0}} &= \frac{1}{s^2 + 2s + 1}.
 \end{aligned} \tag{51}$$

Case 1. In this simulation, we assume that the aircraft is cruising at trim states, and only the morphing process is considered. The initial tracking errors are assumed to be $z_1(0) = 0.1 \text{ m}$ and $z_V(0) = 0.3 \text{ m/s}$. For comparison purposes, the switching function based adaptive neural control (SAN which means the PPC technique is not employed in the control design) and adaptive neural prescribed performance control (PPC described in Remark 10) are utilized; meanwhile the control gains are kept fixed to the values used in the proposed control scheme (SPPC). Simulation results are presented in Figures 1–4. Specifically, the output tracking errors are presented in Figure 2. Moreover, the required control input δ_e , T , system states, and the evolution of NNs' weight are provided in Figures 3 and 4. Notice that during the morphing process the input states (x_2, x_3, x_4)

of NNs always stay in the active region which can be explained in Figure 1(b); thus the proposed SPPC scheme is equivalent to PPC scheme. As expected, output tracking with prescribed performance as well as states boundedness is achieved with reasonable control effort. In contrast to PPC or SPPC approach, although the SAN control scheme can keep the stability of the aircraft, it owns relatively poor system performance as shown in Figure 2. Moreover, the superiority of the proposed SPPC scheme will be further revealed in Case 2 simulation.

Case 2. In this simulation, the control gains are kept the same as Case 1; meanwhile the SAN and PPC approaches are also used as comparison. The upper bounds of f_2^u and f_4^u are set as $f_2^u = 0.5$ and $f_4^u = 1$. Simulation results are presented in Figures 5–9. The output tracking for SAN and SPPC is presented in Figures 5(a) and 5(c) and the corresponding tracking errors are presented in Figures 5(b) and 5(d) along with their performance bounds. Particularly, the control inputs and system states as well as the value of the switch functions are provided in Figures 6, 7, and 8. As expected, output tracking with prescribed performance and states boundedness are both achieved by using the proposed SPPC scheme. Unfortunately, the altitude and velocity tracking errors of SAN transcend the prescribed bounds λ_1 and λ_V . It is worth noting that, different from Case 1, the switch values are not always equal to one as depicted in Figure 8, which means that the NNs are out of the neural approximation regions. The curve of δ_e is obtained as shown in Figure 7(a). After about 11 seconds, the robust controller pulls back the escaped transient to the neural effective regions. Some corresponding responses for PPC scheme are pictured in Figure 9. From this figure, we can conclude that the altitude error oversteps the prescribed bounds without the utilization of switching strategy, thus leading to the instability of the system. In all, compared with simulation results, the superiority of SPPC scheme is obvious.

5. Conclusion

A composite switching neural prescribed performance control scheme has been proposed for the longitudinal dynamic model of the morphing aircraft. In the control design, by using neural networks to approximate the unknown functions, the prior information of the aerodynamic parameters is unnecessary. By introducing the performance function, the proposed controller is able to permit attributes such as a lower bound on the convergence rate and maximum allowable steady error to be specified. A switching mechanism supervising the exchange of control authorities between the normal neural controller and a robust controller is used to relax the constraint that NN should be kept in the active regions all the time. Two comparative simulations have revealed the superiority of this control scheme.

Appendix

A. Proof of Theorem 7

Proof. Invoking (24) and (25) yields

$$\begin{aligned} \dot{\mu}_V(t) = r_V \left[- \left(k_{V1} - \frac{\dot{r}_V}{2r_V^2} \right) \mu_V + W_V^{*T} \Phi_V \right. \\ \left. - \frac{1}{2} \mu_V \hat{\varphi}_V \Phi_V^T \Phi_V + d_V - \hat{d}_V \tanh \left(\frac{\mu_V}{\omega_{V1}} \right) \right]. \end{aligned} \quad (\text{A.1})$$

Consider the following candidate Lyapunov function:

$$L_V = \frac{1}{2r_V} \mu_V^2 + \frac{1}{2\rho_{V1}} \tilde{\varphi}_V^2 + \frac{1}{2\rho_{V2}} \tilde{d}_V^2, \quad (\text{A.2})$$

where $\tilde{\varphi}_V = \varphi_V - \hat{\varphi}_V$ and $\tilde{d}_V = d_{VM} - \hat{d}_V$.

Based on (26) and (A.1), the time derivative of L_V is given by

$$\begin{aligned} \dot{L}_V &= \frac{1}{r_V} \mu_V \dot{\mu}_V - \frac{\dot{r}_V}{2r_V^2} \mu_V^2 - \frac{1}{\rho_{V1}} \tilde{\varphi}_V \dot{\tilde{\varphi}}_V - \frac{1}{\rho_{V2}} \tilde{d}_V \dot{\tilde{d}}_V \\ &= - \left(k_{V1} - \frac{\dot{r}_V - \hat{r}_V}{2r_V^2} \right) \mu_V^2 + \mu_V W_V^{*T} \Phi_V \\ &\quad - \frac{1}{2} \mu_V^2 \varphi_V \Phi_V^T \Phi_V + d_V \mu_V \\ &\quad - d_{VM} \mu_V \tanh \left(\frac{\mu_V}{\omega_{V1}} \right) + \frac{1}{2} \sigma_{V1} \tilde{\varphi}_V \hat{\varphi}_V \\ &\quad + \sigma_{V2} \tilde{d}_V \hat{d}_V. \end{aligned} \quad (\text{A.3})$$

Note that the following inequalities hold:

$$\begin{aligned} \mu_V W_V^{*T} \Phi_V &\leq \frac{1}{2} \mu_V^2 \varphi_V \Phi_V^T \Phi_V + \frac{1}{2}, \\ \sigma_{V2} \tilde{d}_V \hat{d}_V &\leq \frac{\sigma_{V2}}{2} (d_{VM}^2 - \tilde{d}_V^2), \\ d_V \mu_V - d_{VM} \mu_V \tanh \left(\frac{\mu_V}{\omega_{V1}} \right) \\ &\leq d_{VM} |\mu_V| - d_{VM} \mu_V \tanh \left(\frac{\mu_V}{\omega_{V1}} \right) \leq \kappa_0 \omega_{V1} d_{VM}, \\ \sigma_{V1} \tilde{\varphi}_V \hat{\varphi}_V &= \frac{\sigma_{V1}}{2} (\varphi_V^2 - \tilde{\varphi}_V^2 - \hat{\varphi}_V^2) \leq \frac{\sigma_{V1}}{2} (\varphi_V^2 - \tilde{\varphi}_V^2). \end{aligned} \quad (\text{A.4})$$

By considering (A.4), \dot{L}_V can be reformulated as

$$\begin{aligned} \dot{L}_V &\leq - \left(k_{V1} - \frac{\bar{l}_V}{2r_{V\min}^2} \right) \mu_V^2 - \frac{\sigma_{V1}}{2} \tilde{\varphi}_V^2 - \frac{\sigma_{V2}}{2} \tilde{d}_V^2 \\ &\quad + C_V, \end{aligned} \quad (\text{A.5})$$

where $C_V = 0.5 + \kappa_0 \omega_{V1} d_{VM} + (\sigma_{V1}/2) \varphi_V^2 + (\sigma_{V2}/2) d_{VM}^2$.

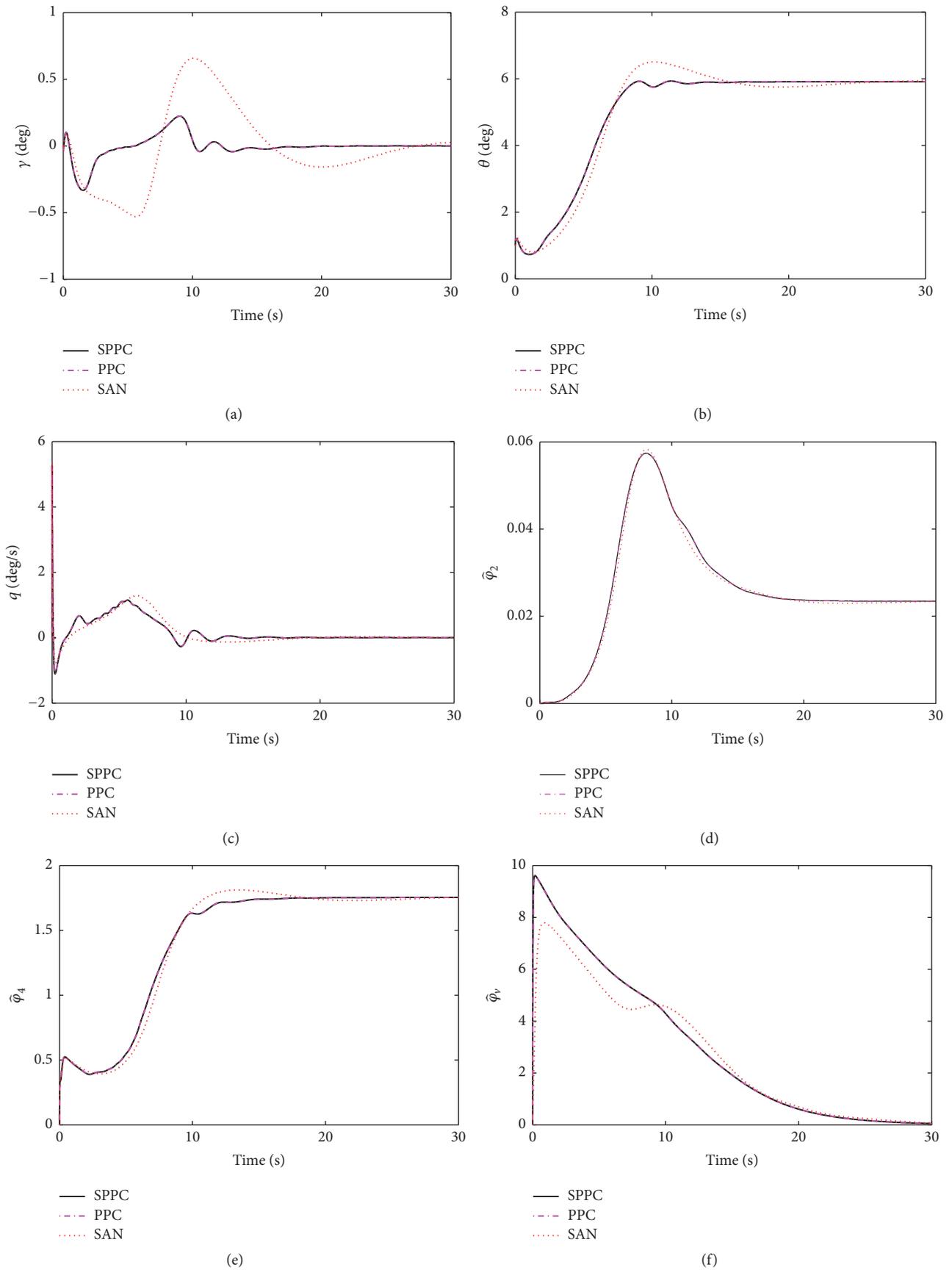


FIGURE 3: System states and NN weights.

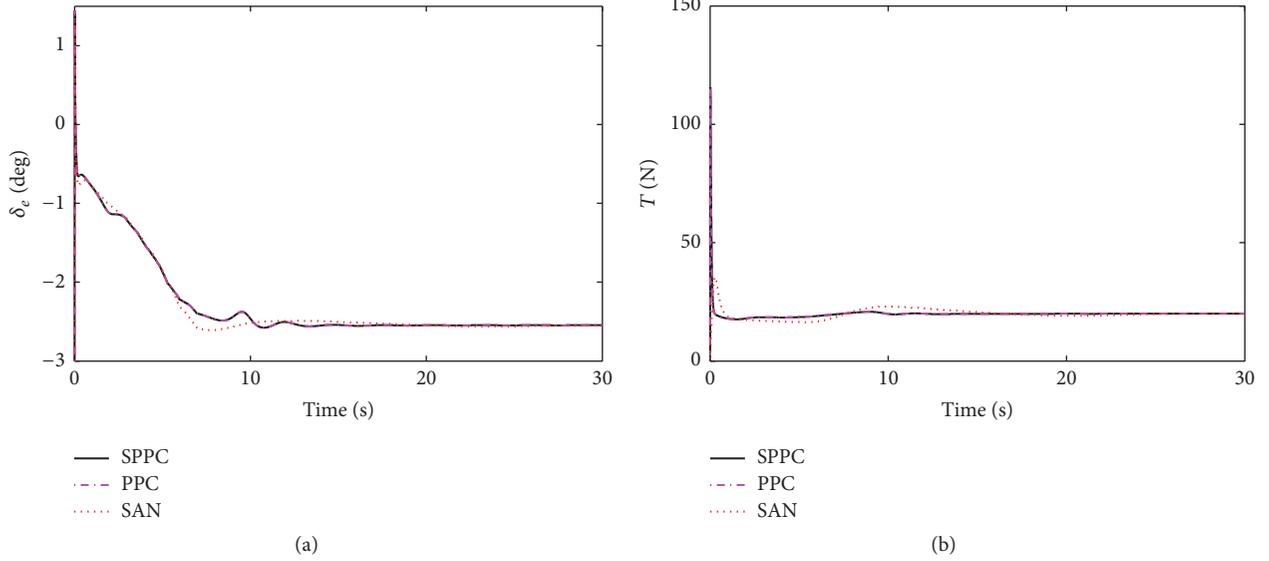


FIGURE 4: Control inputs.

If $k_{V1} - 0.5\bar{l}_V/r_{V\min}^2 > 0$, define the following compact sets:

$$\begin{aligned}\Omega_{\mu_V} &= \left\{ \mu_V \mid |\mu_V| \leq \sqrt{\frac{C_V}{(k_{V1} - 0.5\bar{l}_V/r_{V\min}^2)}} \right\}, \\ \Omega_{\tilde{\varphi}_V} &= \left\{ \tilde{\varphi}_V \mid |\tilde{\varphi}_V| \leq \sqrt{\frac{C_V}{(0.5\sigma_{V1})}} \right\}, \\ \Omega_{\tilde{d}_V} &= \left\{ \tilde{d}_V \mid |\tilde{d}_V| \leq \sqrt{\frac{C_V}{(0.5\sigma_{V2})}} \right\}.\end{aligned}\quad (\text{A.6})$$

It is obvious that \dot{L}_V is negative if $\mu_V \notin \Omega_{\mu_V}$, $\tilde{\varphi}_V \notin \Omega_{\tilde{\varphi}_V}$, and $\tilde{d}_V \notin \Omega_{\tilde{d}_V}$. Therefore, the signals in μ_V , $\tilde{\varphi}_V$, and \tilde{d}_V in the closed-loop system are bounded. \square

B. Proof of Theorem 9

Proof. Select the candidate Lyapunov function as follows:

$$L = L_1 + L_2 + L_3 + L_4, \quad (\text{B.1})$$

where $L_1 = (1/2\bar{V}^2 r_1)\mu_1^2$, $L_2 = (1/2)z_2^2 + \tilde{\varphi}_2^2/2\rho_{21} + \tilde{d}_2^2/2\rho_{22}$, $L_3 = (1/2)z_3^2$, $L_4 = (1/2)z_4^2 + \tilde{\varphi}_4^2/2\rho_{41} + \tilde{d}_4^2/2\rho_{42}$, $\tilde{\varphi}_2 = \varphi_2 - \hat{\varphi}_2$, $\tilde{d}_2 = d_{2M} - \hat{d}_2$, $\tilde{\varphi}_4 = \varphi_4 - \hat{\varphi}_4$, and $\tilde{d}_4 = d_{4M} - \hat{d}_4$ and \bar{V} denotes the upper bound of V .

On the basis of (31), the time derivative of L_1 is given by

$$\dot{L}_1 = \frac{1}{\bar{V}^2 r_1} \mu_1 \dot{\mu}_1 - \frac{\dot{r}_1}{2\bar{V}^2 r_1^2} \mu_1^2 = \frac{-k_1}{\bar{V}^2} \mu_1^2 + \frac{V}{\bar{V}^2} \mu_1 z_2. \quad (\text{B.2})$$

Differentiating L_2 with respect to time, invoking (37) and (38), we have

$$\begin{aligned}\dot{L}_2 &= z_2 \dot{z}_2 - \frac{1}{\rho_{21}} \tilde{\varphi}_2 \dot{\tilde{\varphi}}_2 - \frac{1}{\rho_{22}} \tilde{d}_2 \dot{\tilde{d}}_2 \leq -k_2 z_2^2 + z_2 z_3 \\ &\quad - z_2 \tau_1 + m_2 \left(z_2 W_2^{*T} \Phi_2 - \frac{1}{2} z_2^2 \tilde{\varphi}_2 \Phi_2^T \Phi_2 \right. \\ &\quad \left. + d_{2M} |z_2| - d_{2M} z_2 \tanh\left(\frac{z_2}{\omega_{21}}\right) \right) + (1 - m_2) \\ &\quad \cdot \left(|z_2 f_2^u| - z_2 f_2^u \tanh\left(\frac{z_2 f_2^u}{\omega_{22}}\right) \right) + \sigma_{21} \tilde{\varphi}_2 \hat{\varphi}_2 \\ &\quad + \sigma_{22} \tilde{d}_2 \hat{d}_2.\end{aligned}\quad (\text{B.3})$$

Employing (43), the time derivative of L_3 is obtained as

$$\begin{aligned}\dot{L}_3 &= z_3 (z_4 - z_2 - k_3 z_3 - \tau_2) \\ &\leq -\left(k_3 - \frac{1}{2k_{12}}\right) z_3^2 - z_2 z_3 + z_3 z_4 + \frac{k_{12} \bar{\tau}_2^2}{2}.\end{aligned}\quad (\text{B.4})$$

Using (49) and (50) results in the time derivative of L_4 :

$$\begin{aligned}\dot{L}_4 &= z_4 \dot{z}_4 - \frac{1}{\rho_{41}} \tilde{\varphi}_4 \dot{\tilde{\varphi}}_4 - \frac{1}{\rho_{42}} \tilde{d}_4 \dot{\tilde{d}}_4 \leq -k_4 z_4^2 - z_3 z_4 \\ &\quad - z_4 \tau_3 + m_4 \left(z_4 W_4^{*T} \Phi_4 - \frac{1}{2} z_4^2 \tilde{\varphi}_4 \Phi_4^T \Phi_4 \right. \\ &\quad \left. + |z_4| d_{4M} - z_4 d_{4M} \tanh\left(\frac{z_4}{\omega_{41}}\right) \right) + (1 - m_4) \\ &\quad \cdot \left(|z_4 f_4^u| - z_4 f_4^u \tanh\left(\frac{z_4 f_4^u}{\omega_{42}}\right) \right) + \sigma_{41} \tilde{\varphi}_4 \hat{\varphi}_4 \\ &\quad + \sigma_{42} \tilde{d}_4 \hat{d}_4.\end{aligned}\quad (\text{B.5})$$

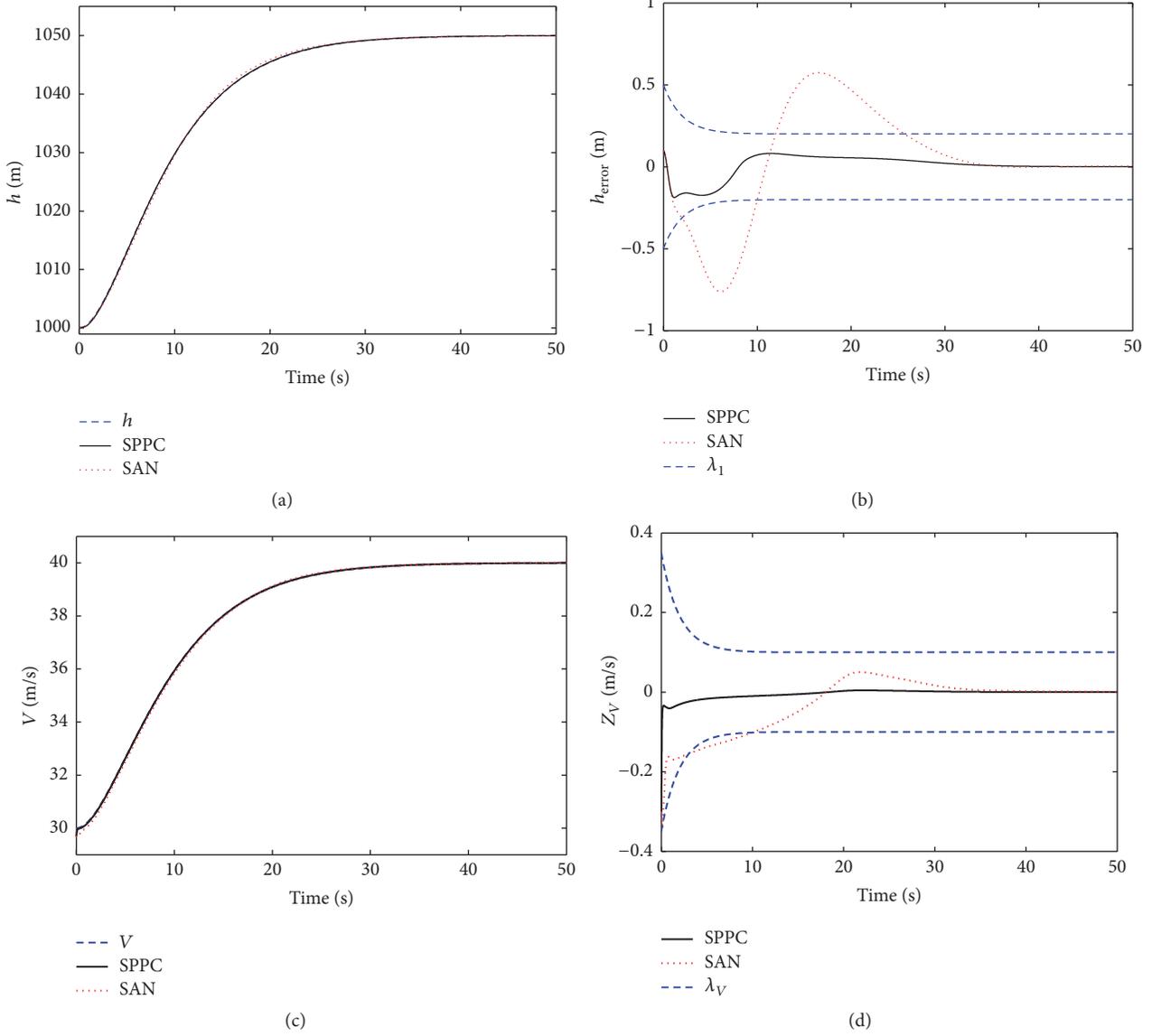


FIGURE 5: Altitude and velocity tracking.

Consider the following facts:

$$V\mu_1 z_2 \leq \frac{1}{2k_{11}}\mu_1^2 + \frac{k_{11}}{2}V^2 z_2^2,$$

$$\sigma_{21}\tilde{\varphi}_2\hat{\varphi}_2 = \frac{\sigma_{21}}{2}(\varphi_2^2 - \tilde{\varphi}_2^2 - \hat{\varphi}_2^2) \leq \frac{\sigma_{21}}{2}(\varphi_2^2 - \tilde{\varphi}_2^2),$$

$$z_2 W_2^{*T} \Phi_2 \leq \frac{1}{2}z_2^2 \Phi_2^T \Phi_2 + \frac{1}{2},$$

$$\sigma_{22}\tilde{d}_2\hat{d}_2 \leq \frac{1}{2}(\sigma_{22}d_{2M}^2 - \sigma_{22}\tilde{d}_2^2),$$

$$-z_2\tau_1 \leq \frac{1}{2}z_2^2 + \frac{1}{2}\tau_1^2,$$

$$|z_2|d_2 - z_3d_{2M} \tanh\left(\frac{z_3}{w_2}\right) \leq \kappa_0 d_{2M}\omega_{21},$$

$$|z_2 f_2^u| - z_2 f_2^u \tanh\left(\frac{z_2 f_2^u}{\omega_{22}}\right) \leq \kappa_0 \omega_{22},$$

$$-\tau_2 z_3 \leq \frac{1}{2k_{12}}z_3^2 + \frac{k_{12}}{2}\tau_2^2,$$

$$\sigma_{41}\tilde{\varphi}_4\hat{\varphi}_4 = \frac{\sigma_{41}}{2}(\varphi_4^2 - \tilde{\varphi}_4^2 - \hat{\varphi}_4^2) \leq \frac{\sigma_{41}}{2}(\varphi_4^2 - \tilde{\varphi}_4^2),$$

$$\sigma_{42}\tilde{d}_4\hat{d}_4 \leq \frac{1}{2}\sigma_{42}(d_{4M}^2 - \tilde{d}_4^2),$$

$$|z_4 d_4| - z_4 d_{4M} \tanh\left(\frac{z_4}{\omega_{41}}\right) \leq \kappa_0 d_{4M}\omega_{41},$$

$$-z_4\tau_3 \leq \frac{1}{2}(z_4^2 + \tau_3^2),$$

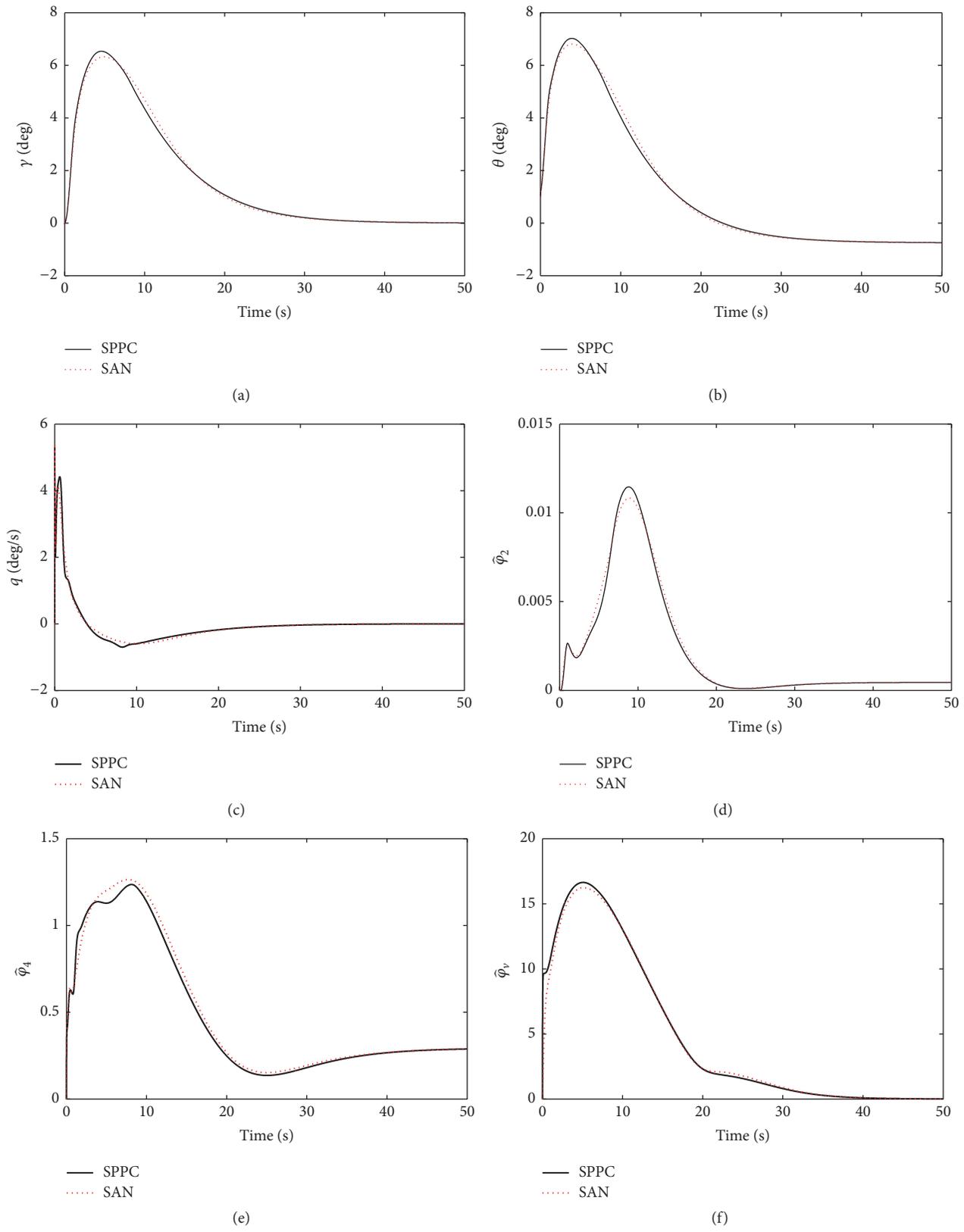


FIGURE 6: System states and NN weights.

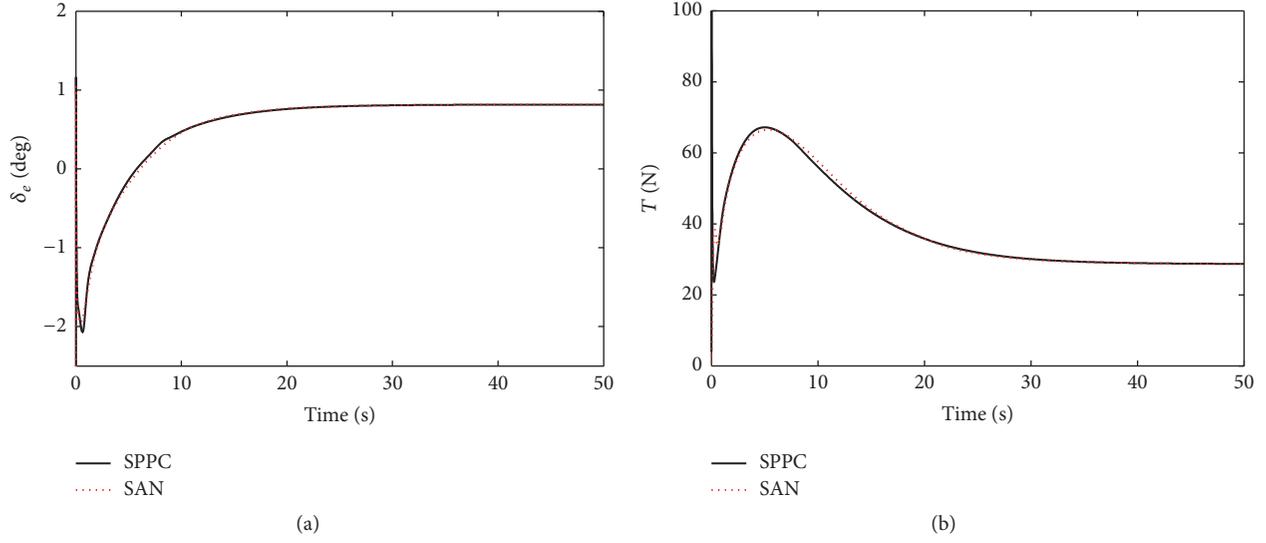


FIGURE 7: Control inputs.

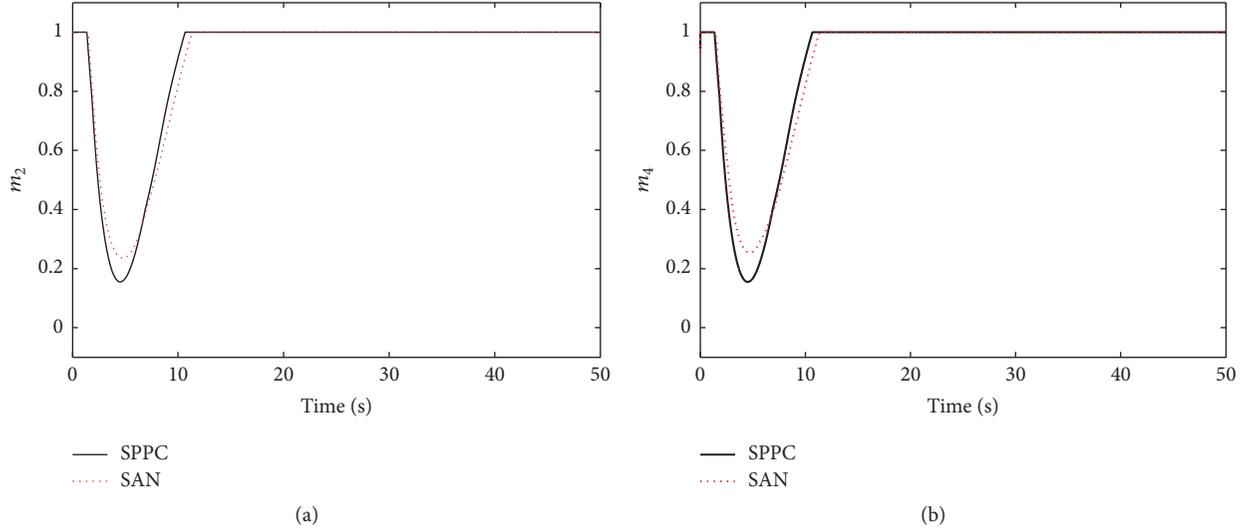


FIGURE 8: Switching functions.

$$z_4 W_4^{*T} \Phi \leq \frac{1}{2} z_4^2 \varphi_4 \Phi_4^T \Phi_4 + \frac{1}{2},$$

$$|z_4 f_4| - z_4 f_4^u \tanh\left(\frac{z_4 f_4^u}{\omega_{42}}\right) \leq \kappa_0 \omega_{42}.$$

(B.6)

We have

$$\begin{aligned} \dot{L} \leq & -\frac{1}{V^2} \left(k_1 - \frac{1}{2k_{11}}\right) \mu_1^2 - (k_2 - 0.5 - 0.5k_{11}) z_2^2 \\ & - \frac{1}{2} \sigma_{21} \bar{\varphi}_2^2 - \frac{1}{2} \sigma_{22} \bar{d}_2^2 - \left(k_3 - \frac{1}{2k_{12}}\right) z_3^2 \\ & - (k_4 - 0.5) z_4^2 - \frac{1}{2} \sigma_{41} \bar{\varphi}_4^2 - \frac{1}{2} \sigma_{42} \bar{d}_4^2 + C_2, \end{aligned} \quad (\text{B.7})$$

where the corresponding design parameters should be chosen such that $k_1 - 0.5/k_{11} > 0$, $k_2 - 0.5 - 0.5k_{11} > 0$, $k_3 - 0.5/k_{12} > 0$, $(k_4 - 0.5) > 0$, and $\sigma_{ij} > 0$, $i = 2, 4$, $j = 1, 2$.

$$\begin{aligned} C_2 = & 0.5 \left(\sigma_{21} \varphi_2^2 + \sigma_{22} d_{2M}^2 + m_2 + \bar{\tau}_1^2 + k_{12} \bar{\tau}_2^2 \right. \\ & \left. + \sigma_{42} d_{4M}^2 + \sigma_{41} \varphi_4^2 + \bar{\tau}_3^2 + m_4 \right) + m_2 \kappa_0 d_{2M} \omega_{21} \\ & + (1 - m_2) \kappa_0 \omega_{22} + m_4 \kappa_0 d_{4M} \omega_{41}. \end{aligned} \quad (\text{B.8})$$

Define the following compact sets:

$$\begin{aligned} \Omega_{\mu_1} &= \left\{ \mu_1 \mid |\mu_1| \leq \sqrt{\frac{C_2}{(k_1 - 0.5/k_{11}) V^2}} \right\}, \\ \Omega_{z_2} &= \left\{ z_2 \mid |z_2| \leq \sqrt{\frac{C_2}{(k_2 - 0.5 - 0.5k_{11})}} \right\}, \end{aligned}$$

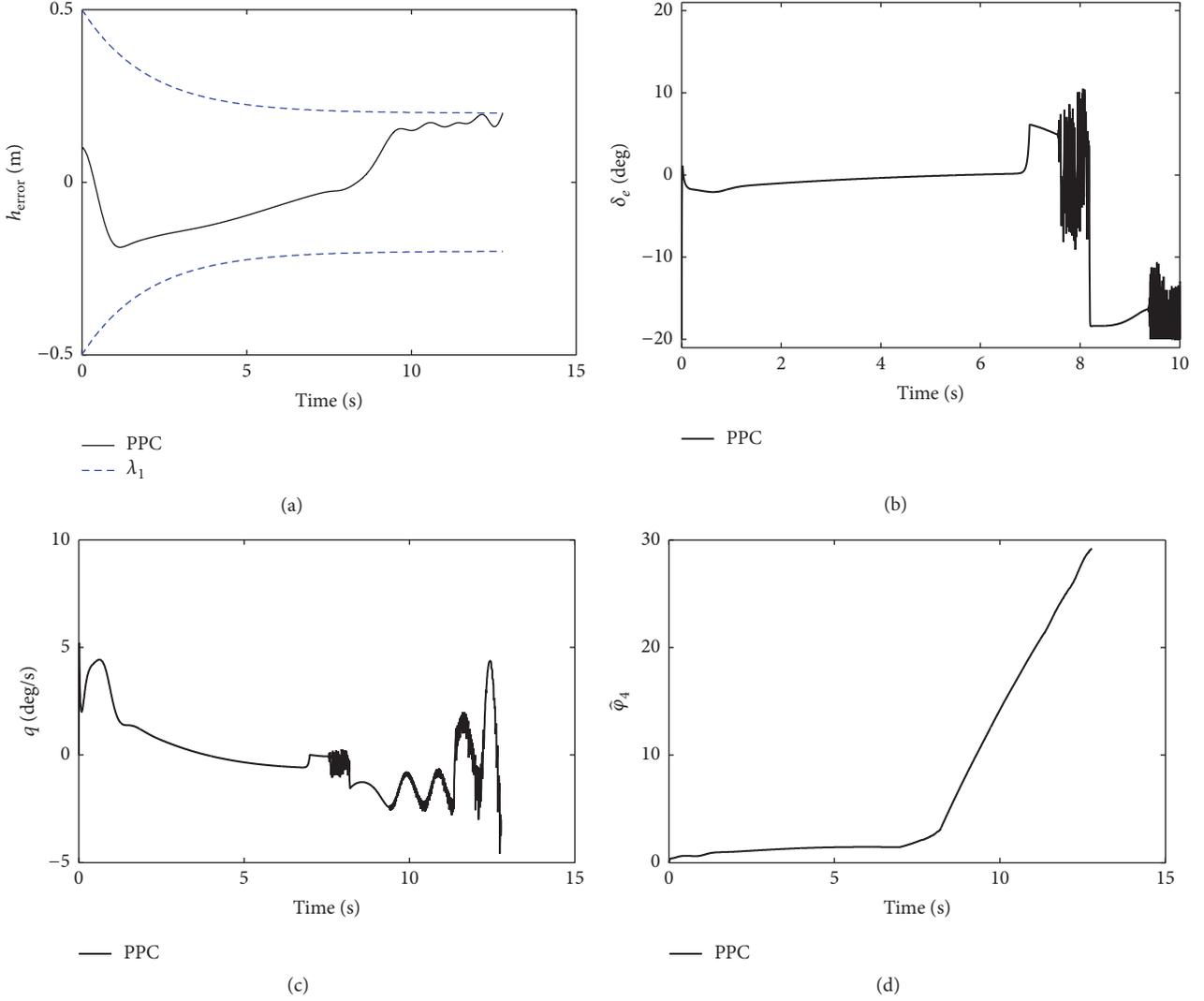


FIGURE 9: Partial response of PPC scheme.

$$\Omega_{z_3} = \left\{ z_3 \mid |z_3| \leq \sqrt{\frac{C_2}{(k_3 - 0.5/k_{12})}} \right\},$$

$$\Omega_{z_4} = \left\{ z_4 \mid |z_4| \leq \sqrt{\frac{C_2}{(k_4 - 0.5)}} \right\},$$

$$\Omega_{\tilde{\varphi}_2} = \left\{ \tilde{\varphi}_2 \mid |\tilde{\varphi}_2| \leq \sqrt{\frac{2C_2}{\sigma_{21}}} \right\},$$

$$\Omega_{\tilde{\varphi}_4} = \left\{ \tilde{\varphi}_4 \mid |\tilde{\varphi}_4| \leq \sqrt{\frac{2C_2}{\sigma_{41}}} \right\},$$

$$\Omega_{\tilde{d}_2} = \left\{ \tilde{d}_2 \mid |\tilde{d}_2| \leq \sqrt{\frac{2C_2}{\sigma_{22}}} \right\},$$

$$\Omega_{\tilde{d}_4} = \left\{ \tilde{d}_4 \mid |\tilde{d}_4| \leq \sqrt{\frac{2C_2}{\sigma_{42}}} \right\}.$$

(B.9)

If $\mu_1 \notin \Omega_{\mu_1}$, $z_2 \notin \Omega_{z_2}$, $z_3 \notin \Omega_{z_3}$, $z_4 \notin \Omega_{z_4}$, $\tilde{\varphi}_2 \notin \Omega_{\tilde{\varphi}_2}$, $\tilde{\varphi}_4 \notin \Omega_{\tilde{\varphi}_4}$, $\tilde{d}_2 \notin \Omega_{\tilde{d}_2}$, and $\tilde{d}_4 \notin \Omega_{\tilde{d}_4}$, we know that \dot{L} will be negative. Therefore, the signals μ_1 , $z_{i=2,3,4}$, $\tilde{\varphi}_{i=2,4}$, and $\tilde{d}_{i=2,4}$ in the closed-loop system are bounded. \square

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

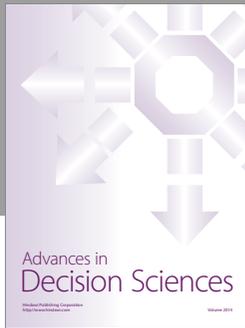
This work is partially supported by the National Natural Science Foundation of China (Grant nos. 61374032 and 61573286).

References

- [1] T. Yue, L. Wang, and J. Ai, "Gain self-scheduled H_∞ control for morphing aircraft in the wing transition process based on an

- LPV model,” *Chinese Journal of Aeronautics*, vol. 26, no. 4, pp. 909–917, 2013.
- [2] T. Wang, C. Dong, and Q. Wang, “Finite-time boundedness control of morphing aircraft based on switched systems approach,” *Optik*, vol. 126, no. 23, pp. 4436–4445, 2015.
 - [3] Z. Wu, J. Lu, Q. Zhou, and J. Shi, “Modified adaptive neural dynamic surface control for morphing aircraft with input and output constraints,” *Nonlinear Dynamics*, vol. 87, no. 4, pp. 2367–2383, 2017.
 - [4] D. H. Baldelli, D.-H. Lee, R. S. Sánchez Peñal, and B. Cannon, “Modeling and control of an aeroelastic morphing vehicle,” *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 6, pp. 1687–1699, 2008.
 - [5] Z. Wu, J. Lu, J. Rajput, J. Shi, and W. Ma, “Adaptive neural control based on high order integral chained differentiator for morphing aircraft,” *Mathematical Problems in Engineering*, vol. 2015, pp. 1–12, 2015.
 - [6] W. Jiang, C. Dong, and Q. Wang, “A systematic method of smooth switching LPV controllers design for a morphing aircraft,” *Chinese Journal of Aeronautics*, vol. 28, no. 6, pp. 1640–1649, 2015.
 - [7] D. Wang and J. Huang, “Neural network-based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form,” *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 195–202, 2005.
 - [8] A.-M. Zou, Z.-G. Hou, and M. Tan, “Adaptive control of a class of nonlinear pure-feedback systems using fuzzy backstepping approach,” *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 4, pp. 886–897, 2008.
 - [9] B. S. Kim and S. J. Yoo, “Approximation-based adaptive control of uncertain non-linear pure-feedback systems with full state constraints,” *IET Control Theory & Applications*, vol. 8, no. 17, pp. 2070–2081, 2014.
 - [10] Q. Shen, B. Jiang, and V. Cocquempot, “Fuzzy logic system-based adaptive fault-tolerant control for near-space vehicle attitude dynamics with actuator faults,” *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 2, pp. 289–300, 2013.
 - [11] Y. Li, S. Tong, and T. Li, “Adaptive fuzzy output-feedback control for output constrained nonlinear systems in the presence of input saturation,” *Fuzzy Sets and Systems*, vol. 248, pp. 138–155, 2014.
 - [12] Y. M. Li and S. C. Tong, “Adaptive fuzzy output-feedback control of pure-feedback uncertain nonlinear systems with unknown dead-zone,” *IEEE Transactions on Fuzzy Systems*, vol. 22, pp. 1341–1347, 2014.
 - [13] K. P. Tee, S. S. Ge, and E. H. Tay, “Barrier Lyapunov functions for the control of output-constrained nonlinear systems,” *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.
 - [14] M. Chen, G. Tao, and B. Jiang, “Dynamic surface control using neural networks for a class of uncertain nonlinear systems with input saturation,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 9, pp. 2086–2097, 2015.
 - [15] B. Xu, Z. Shi, C. Yang, and F. Sun, “Composite neural dynamic surface control of a class of uncertain nonlinear systems in strict-feedback form,” *IEEE Transactions on Cybernetics*, vol. 44, pp. 2626–2634, 2014.
 - [16] Q. Shen, B. Jiang, and V. Cocquempot, “Adaptive fuzzy observer-based active fault-tolerant dynamic surface control for a class of nonlinear systems with actuator faults,” *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 2, pp. 338–349, 2014.
 - [17] B. Xu, Y. Fan, and S. Zhang, “Minimal-learning-parameter technique based adaptive neural control of hypersonic flight dynamics without back-stepping,” *Neurocomputing*, vol. 164, no. 1–2, pp. 201–209, 2015.
 - [18] B. Xu, Z. Shi, C. Yang, and S. Wang, “Neural control of hypersonic flight vehicle model via time-scale decomposition with throttle setting constraint,” *Nonlinear Dynamics*, vol. 73, no. 3, pp. 1849–1861, 2013.
 - [19] X. Bu, X. Wu, D. Wei, and J. Huang, “Neural-approximation-based robust adaptive control of flexible air-breathing hypersonic vehicles with parametric uncertainties and control input constraints,” *Information Sciences*, vol. 346–347, pp. 29–43, 2016.
 - [20] Q. Zong, F. Wang, B. Tian, and R. Su, “Robust adaptive dynamic surface control design for a flexible air-breathing hypersonic vehicle with input constraints and uncertainty,” *Nonlinear Dynamics*, vol. 78, no. 1, pp. 289–315, 2014.
 - [21] L. Zhang, S. Sui, Y. Li, and S. Tong, “Adaptive fuzzy output feedback tracking control with prescribed performance for chemical reactor of MIMO nonlinear systems,” *Nonlinear Dynamics*, vol. 80, no. 1–2, pp. 945–957, 2015.
 - [22] C. P. Bechlioulis and G. A. Rovithakis, “Adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems,” *Automatica*, vol. 45, no. 2, pp. 532–538, 2009.
 - [23] J. Na, Q. Chen, X. Ren, and Y. Guo, “Adaptive prescribed performance motion control of servo mechanisms with friction compensation,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 486–494, 2014.
 - [24] X. Bu, X. Wu, F. Zhu, J. Huang, Z. Ma, and R. Zhang, “Novel prescribed performance neural control of a flexible air-breathing hypersonic vehicle with unknown initial errors,” *ISA Transactions*, vol. 59, pp. 149–159, 2015.
 - [25] Y. Huang, J. Na, X. Wu, X. Liu, and Y. Guo, “Adaptive control of nonlinear uncertain active suspension systems with prescribed performance,” *ISA Transactions*, vol. 54, pp. 145–155, 2015.
 - [26] S. Sui, S. Tong, and Y. Li, “Observer-based fuzzy adaptive prescribed performance tracking control for nonlinear stochastic systems with input saturation,” *Neurocomputing*, vol. 158, pp. 100–108, 2015.
 - [27] Y. Li and S. Tong, “Prescribed performance adaptive fuzzy output-feedback dynamic surface control for nonlinear large-scale systems with time delays,” *Information Sciences*, vol. 292, pp. 125–142, 2015.
 - [28] S. I. Han and J. M. Lee, “Partial tracking error constrained fuzzy dynamic surface control for a strict feedback nonlinear dynamic system,” *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 5, pp. 1049–1061, 2014.
 - [29] C. P. Bechlioulis and G. A. Rovithakis, “Prescribed performance adaptive control for multi-input multi-output affine in the control nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1220–1226, 2010.
 - [30] S. Tong, S. Sui, and Y. Li, “Fuzzy adaptive output feedback control of MIMO nonlinear systems with partial tracking errors constrained,” *IEEE Transactions on Fuzzy Systems*, vol. 23, pp. 729–742, 2015.
 - [31] Z. Wu, J. Lu, J. Shi, Q. Zhou, and X. Qu, “Tracking error constrained robust adaptive neural prescribed performance control for flexible hypersonic flight vehicle,” *International Journal of Advanced Robotic Systems*, vol. 14, no. 1, 2017.
 - [32] J.-T. Huang, “Global tracking control of strict-feedback systems using neural networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 11, pp. 1714–1725, 2012.

- [33] C. P. Bechlioulis and G. A. Rovithakis, "A priori guaranteed evolution within the neural network approximation set and robustness expansion via prescribed performance control," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 4, pp. 669–675, 2012.
- [34] J. Wu, W. Chen, D. Zhao, and J. Li, "Globally stable direct adaptive backstepping NN control for uncertain nonlinear strict-feedback systems," *Neurocomputing*, vol. 122, pp. 134–147, 2013.
- [35] B. Xu, C. Yang, and Y. Pan, "Global neural dynamic surface tracking control of strict-feedback systems with application to hypersonic flight vehicle," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2563–2575, 2015.
- [36] J.-T. Huang, "Global adaptive neural dynamic surface control of strict-feedback systems," *Neurocomputing*, vol. 165, pp. 403–413, 2015.
- [37] J. Wu, W. Chen, F. Yang, J. Li, and Q. Zhu, "Global adaptive neural control for strict-feedback time-delay systems with predefined output accuracy," *Information Sciences*, vol. 301, pp. 27–43, 2015.
- [38] W. Chen, J. Lu, X. Wang, and W. Zhang, "Design of a controller for morphing aircraft based on backstepping/RHO," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 40, no. 8, pp. 1060–1065, 2014.
- [39] Y. Yang, C. Ge, H. Wang, X. Li, and C. Hua, "Adaptive neural network based prescribed performance control for teleoperation system under input saturation," *Journal of the Franklin Institute. Engineering and Applied Mathematics*, vol. 352, no. 5, pp. 1850–1866, 2015.
- [40] B. Ren, S. S. Ge, K. P. Tee, and T. H. Lee, "Adaptive neural control for output feedback nonlinear systems using a barrier lyapunov function," *IEEE Transactions on Neural Networks*, vol. 21, no. 8, pp. 1339–1345, 2010.
- [41] M. Chen and S. Ge, "Adaptive neural output feedback control of uncertain nonlinear systems with unknown hysteresis using disturbance observer," *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 7706–7716, 2015.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

