Research Article

Composite Learning Prescribed Performance Control of Nonlinear Systems

Fang Zhu, Wei Xiang, and Chunzhi Yang

Department of Applied Mathematics, Huainan Normal University, Huainan 232038, China

Correspondence should be addressed to Wei Xiang; xiangwei27@126.com

Received 19 March 2021; Revised 21 April 2021; Accepted 29 April 2021; Published 10 May 2021

1. Introduction

In recent decades, many scholars are studying the stability tracking control of strict-feedback nonlinear systems [1–5], such as the inverted model and the one-link manipulator. Noting that the structures of these nonlinear systems are cascaded, an effective control method, namely, backstepping technology is developed; however, an inherent shortcoming of this method consists in repeated differentiation of virtual control inputs, which leads to “explosion-complexity.” A feasible approach to solve this problem is to estimate the virtual control inputs together with their derivatives. Consequently, filter command control approach was introduced in [1]. And, an adaptive dynamic surface control (ADSC) was proposed in [6]. The design feature of ADSC is that it does not need to calculate the derivative of virtual controller, and it can relax the smoothness requirements of system models and referenced signals. In [7], Yu et al. used a first-order Levant differentiator to discuss the finite-time tracking control problem of a class of nth order strict-feedback systems. However, the uncertain parameter or unknown function does not consider whether it can be accurately estimated. Recently, some composite learning approaches were proposed in [8, 9]. Under an interval excitation (IE) condition, composite learning needs to use online recorded data to construct a prediction error and then use prediction error and tracking error to construct composite learning update law. So, the unknown parameter is estimated accurately. In addition to the accurate estimation of unknown parameters, how to estimate the unknown functions accurately needs further research.

However, most of the methods in the above literatures focus on how to ensure that the tracking error converges to a small neighborhood of zero, which is a steady-state research. However, some practical systems, such as near space vehicles (NSV) [10], need to consider both steady-state performance and transient performance. In order to solve this problem, prescribed performance control (PPC) [11–18] as one of the solutions has received extensive attention. The basic idea of the PPC method is to design the performance function (PPF) and the error transfer function so that the tracking error can be always limited within the performance function boundary (PPB) by the PPF. In [15], an adaptive neural PPC is proposed for NSVs with input nonlinearity. In order to achieve

Copyright © 2021 Fang Zhu 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 paper investigates a composite learning prescribed performance control (PPC) scheme for uncertain strict-feedback system. Firstly, a prescribed performance boundary (PPB) condition is developed for the tracking error, and the original system is transformed into an equivalent one by using a transformation function. In order to ensure that the tracking error satisfies the PPB, a sufficient condition is given. Then, a control scheme of PPC combined with neural network (NN) and backstepping technique is proposed. However, the unknown functions cannot be guaranteed to estimate accurately by this method. To solve this problem, predictive errors are defined by applying online recorded date and instantaneous date. Furthermore, novel composite learning laws are proposed to update NN weights based on a partial persistent excitation (PE) condition. Subsequently, the stability of the closed-loop system is guaranteed and all signals are kept bounded by using composite learning PPC method. Finally, simulation results verify the effectiveness of the proposed methods.
the convergence of tracking error with small overshoot, Bu et al. [16] constructed a new PPF and error transformation function (ETF). Compared with traditional PPC, the control effect of the proposed method has been improved in [16]. Xiang and Liu [17] employed the same PPF and ETF in [16] to design a fuzzy adaptive tracking control for uncertain nonlinear systems with unknown control gain signs. Wang et al. [18] used finite-time performance function to discuss the performance problem of tracking error under actuator faults and external disturbances. Inspired by [18], can we construct a new simple finite-time performance function to achieve the predeterminate performance of tracking error? Moreover, fuzzy logic system (FLS) or neural network (NN) is used to estimate the unknown function, but literatures [11–16] did not further verify whether the unknown function is effectively estimated. Therefore, it is a challenge to effectively estimate the unknown function under the condition of ensuring the steady-state and transient performance of the tracking error. Usually, in order to achieve effective estimation of unknown functions, it is necessary to obtain online recorded data together with instantaneous data to generate prediction errors [19–23]. In this paper, in order to realize the output state to track the recurrent reference signal and the tracking error satisfies PPF, prediction errors are generated by using the partial persistent excitation (PE) condition in [23]. The main contributions of this paper are summarized as follows. (1) A new finite-time performance function is developed. (2) The proposed composite learning PPC method can guaranteed the small convergence overshoot of tracking error. (3) Learning-based parameter adaptive laws constructed with prediction error can be used to accurately estimate unknown nonlinear functions.

The remainder of this paper is organized as follows. Some preliminaries are listed in Section 2. In Section 3, two types of adaptive neural network PPC method are discussed. The simulation results are explained in Section 4. Finally, brief conclusion is presented in Section 5.

2. Preliminaries

2.1. System Description. Consider the following nth-order strict-feedback nonlinear system:

\[
\begin{align*}
\dot{x}_i &= f_i(x) + x_i x_{i+1}, \quad i = 1, 2, \ldots, n - 1, \\
\dot{x}_n &= f_n(x) + u, \\
y &= x_1,
\end{align*}
\]

where \(x = [x_1, x_2, \ldots, x_i]^T \in \mathbb{R}^i \) (\(x_1 = x\) and \(x = x_m\)) are the measurable state vectors, \(u \in \mathbb{R}\) is the control input, \(y \in \mathbb{R}\) is the output, and \(f_i(x)\), \(i = 1, 2, \ldots, n\), are smooth functions.

Assumption 1. The reference signal \(x_d\) is recurrent and bounded. \(x_d\) and its derivative \(\dot{x}_d\) are continuous and available.

Assumption 2. \(f_i(x)\), \(i = 1, 2, \ldots, n\), are unknown and bounded.

Usually, the backstepping technique is used to design the controller \(u\) for system (1). The tedious derivative calculation of the virtual control law is required, which will cause the problem of "explosion-complexity." To avoid such problems, the following first-order filter is employed in this paper.

Lemma 1. For the first-order filter,

\[\tau \dot{\zeta} + \zeta = \alpha,\]

where \(\tau\) is time design constant and \(\zeta(0) = \alpha(0)\). Define an error as \(e = \zeta - \alpha\). According to [5], if the appropriate parameter \(\tau\) is chosen, then there exists positive constant \(\kappa\) such that \(|e| \leq \kappa\).

In this paper, we employed NN to approximate \(f_i(x)\). According to the approximation property of RBFNNs [20], the approximation form of \(f_i(x)\) can be expressed as

\[\tilde{f}_i(x) = \varphi_{f_i}^T(x) \bar{\theta}_{f_i},\]

where \(\bar{\theta}_{f_i} = [\bar{\theta}_{f_1}, \bar{\theta}_{f_2}, \ldots, \bar{\theta}_{f_n}] \in R_{\bar{\theta}_{f_i}}^m\) is weight vector, \(i_t\) is node number, and \(\varphi_{f_i}^T(x) = [\varphi_{f_1}^T(x), \varphi_{f_2}^T(x), \ldots, \varphi_{f_n}^T(x)]^T \in R^m\) is basis function vector. Here, \(\varphi_{f_j}(x)\):

\[
\varphi_{f_j}(x) = \exp\left(\frac{-[x - c_j]^T [x - c_j]}{2b_j^2}\right),
\]

where \(c_j = [c_{j1}, c_{j2}, \ldots, c_{ji}]^T \in R^m\) and \(b_j\) is the center vector and width of the basis function, respectively.

By [20, 21], choosing appropriate \(c_j, b_j\), and some sufficiently large integer \(i\), there exists ideal weight vector \(\bar{\theta}_{f_i}^* \in R_{\bar{\theta}_{f_i}}\) such that

\[f_i(x) = \varphi_{f_i}^T(x) \bar{\theta}_{f_i}^* + \epsilon_{f_j}(x),\]

where \(\epsilon_{f_j}(x)\) is the approximate error, which satisfies \(\|\epsilon_{f_j}(x)\| \leq \epsilon_{f_j}^*\), where \(\epsilon_{f_j}^*\) is a positive constant. So, we can rewrite \(f_i(x)\) in (5) as

\[f_i(x) = \varphi_{f_i}^T(x) \bar{\theta}_{f_i} - \epsilon_{f_j}(x),\]

where \(\bar{\theta}_{f_i} = \bar{\theta}_{f_i}^* - \bar{\theta}_{f_i}\).

Define the tracking error \(e_1 = x_1 - x_d\); substituting (6) into (1), we obtain

\[
\begin{align*}
\dot{e}_1 &= \varphi_{f_i}^T(\bar{x}_1) \bar{\theta}_{f_i} + \varphi_{f_i}^T(\bar{x}_2) \bar{\theta}_{f_2} + x_2 + \epsilon_{f_i}(x) - x_d, \\
\dot{e}_1 &= \varphi_{f_i}^T(\bar{x}_1) \bar{\theta}_{f_i} + \varphi_{f_i}^T(\bar{x}_2) \bar{\theta}_{f_2} + x_2 + \epsilon_{f_i}(x), \\
\dot{x}_1 &= \varphi_{f_1}(\bar{x}_1) \bar{\theta}_{f_1} + \varphi_{f_2}(\bar{x}_1) \bar{\theta}_{f_2} + x_1 + \epsilon_{f_1}(x), \\
\dot{x}_n &= \varphi_{f_n}(\bar{x}_n) \bar{\theta}_{f_n} + \varphi_{f_1}(\bar{x}_n) \bar{\theta}_{f_1} + u + \epsilon_{f_n}(x_n), \\
y &= x_1.
\end{align*}
\]

Remark 1. Basis functions \(\varphi_{f_i}(x), \ i = 1, 2, \ldots, n, \) in (7) satisfy \(\max\{\|\varphi_{f_1}(x)\|, \|\varphi_{f_1}(x)\|\} \leq \chi_i\), where \(\chi_i\) are positive constants independent of \(x_i\) and \(i_t\). For detailed proof, see [6].

Definition 1 (see [23]). A bounded signal \(\varphi(t) \in R^m\) is said to satisfy the persistent excitation (PE) condition if there
exist constants \( \tau_d \) and \( \sigma \) such that \( \int_{t_{\tau_d}}^{t} \phi(s)q^T(s)ds \geq \sigma I \), \( \forall t \in R^* \), where \( I \) is an appropriate identity matrix.

**Lemma 2** (partial PE condition) (see [23]). Consider the RBF network \( \phi_j^T(\mathbf{x}) \theta_j \) (3) with centers placed on a regular lattice to cover \( \Omega_z \). If the trajectory of \( \mathbf{x} \) is recurrent, then there exists a regression subvector \( \phi_j^T(\mathbf{x}) \) that almost always satisfies PE condition.

**Remark 2.** Since the reference signal \( x_d \) is recurrent, it is necessary to prove that the trajectories of \( x_1, x_2, \ldots, x_n \) are recurrent in a finite time \( T \), which mainly avoids the case that learning occurs only when time approaches infinity. Moreover, notice that system states are bounded and NN inputs \( \mathbf{x} \) are recurrent for \( t \geq T \), we need to reset the center vector \( \mathbf{c}_j \), where the value of \( \epsilon \) is the minimum distance between any two center vectors in RBF (3) and construct a regression subvector \( \phi_j^T(\mathbf{x}) \) recurrent trajectory if there exists a positive constant \( \lambda \) such that \( \left| x_1 - x_2 \right| < \lambda \), \( \forall t \in R^* \).

**Lemma 3** (recurrent trajectory, see [23]). \( w(t) \) is called a recurrent trajectory if there exists a positive constant \( \gamma \) such that \( |w(t) - w(t)| < \gamma \), where \( w(t) \) is recurrent signal.

### 2.2. Prescribed Performance (FTPFP)

Firstly, define a finite-time performance function (FTPFP) \( p(t, T, \lambda_0, \lambda_{\infty}) \) as

\[
p(t, T, \lambda_0, \lambda_{\infty}) = \begin{cases} a_1 t^4 + a_2 t^3 + a_3 t^2 + a_4 t + \lambda_0, & t < T, \\ \lambda_{\infty}, & t \geq T, \end{cases}
\]

(8)

where \( T \) is predefined settling time, \( \lambda_0 \) is the initial value of the FTPFP, and \( \lambda_{\infty} \) is the boundary value of the FTPFP after \( T \). To ensure that the FTPFP \( p(t, T, \lambda_0, \lambda_{\infty}) \) is continuous and derivative at \( T \), parameters \( a_1, a_2, a_3, \) and \( a_4 \) are designed as follows:

\[
\begin{align*}
a_1 &= \frac{4(\lambda_{\infty} - \lambda_0)}{T}, \\
a_2 &= \frac{6(\lambda_0 - \lambda_{\infty})}{T^2}, \\
a_3 &= \frac{4(\lambda_{\infty} - \lambda_0)}{T^3}, \\
a_4 &= \frac{\lambda_0 - \lambda_{\infty}}{T^4}.
\end{align*}
\]

(9)

Define the tracking error variable \( e_1 = x_1 - x_d \), and set \( e_1 \) satisfies the following prescribed performance boundary (PPB):

\[
p(t, T, a_0, a_{\infty}) < e_1 < p(t, T, b_0, b_{\infty}),
\]

(10)

where \( p(t, T, a_0, a_{\infty}) \) and \( p(t, T, b_0, b_{\infty}) \) are FTPFPs and \( a_0, b_0, a_{\infty}, b_{\infty} \) satisfy \( a_0 < b_0 \) and \( a_{\infty} < b_{\infty} < b_0 \).

The transformation variable \( z_1 \), which corresponding to PPB (10), is defined as follows:

\[
z_1 = \ln \left( \frac{\rho(t)}{1 - \rho(t)} \right),
\]

(11)

where \( \rho(t) = (e_1 - p(t, T, a_0, a_{\infty})) / (p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty})) \).

**Lemma 4** (see [17]). If \( z_1 \) is bounded, \( e_1 \) can be limited within PPB (10).

The time derivative of \( z_1 \) becomes

\[
\dot{z}_1 = \dot{r}_1 (\dot{\phi}_{\lambda_1}^T(\mathbf{x}_1) \dot{\theta}_{\lambda_1} + \dot{\phi}_{\lambda_1}^T(\mathbf{x}_1) \dot{\theta}_{\lambda_1} + x_2 + \epsilon_{\lambda_1}(\mathbf{x}_1) - x_d + \Pi),
\]

(12)

where

\[
r_1 = \frac{1}{\rho(t) (1 - \rho(t)) (p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty}))} > 0,
\]

(13)

\[
\Pi = \frac{p(t, T, a_0, a_{\infty}) (p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty})) p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty}) p(t, T, b_0, b_{\infty}) - \epsilon (p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty}))}{p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty})}\]

Remark 4. Compared with the traditional predetermined performance function [24], the proposed prescribed performance function (8) in this paper can guarantee that \( e \) can be limited within PPB (10) after \( t > T \). Meanwhile, we can obtain the following results: (i) \( r_1 = (1 / \rho(t) (1 - \rho(t)) (p(t, T, b_0, b_{\infty}) - p(t, T, a_0, a_{\infty})) > 0 \) and (ii) \( (r_1/r_2) \) tends to a small neighborhood of zero. Therefore, \((z_1^2/2r_1)\) will be used in Lyapunov function to explore the boundedness of \( z_1 \).
2.3. Control Aim. The aim of this paper is (i) to design the composite prescribed performance control method so that the tracking error $e_1$ satisfies PPB (10), (ii) to construct prediction errors by using historical data for neural weights update, and (iii) to compare with an adaptive neural network PPC method to show the effectiveness of the proposed control method in this paper.

3. Control Design

3.1. Adaptive Neural Network PPC Method. In order to guarantee that $z_1$ is bounded, backstepping technique will be employed, and the whole control process is divided into $n$ steps.

Step 1: from (12), the virtual control $\alpha_2$ is designed as

$$\alpha_2 = -\varphi_{f_j}^T(\bar{x}_i)\dot{\theta}_{f_j} + \dot{x}_d - \Pi - c_1z_1, \quad (14)$$

where $c_1$ is a positive design constant.

Define $z_2 = x_2 - \zeta_1$, where $\zeta_1$ is used to estimate $\alpha_2$ through the following first-order filter:

$$\tau_1 \dot{\zeta}_1 = -(\zeta_1 - \alpha_2), \quad \zeta_1(0) = \alpha_2(0), \quad (15)$$

where $\tau_1$ is design time constant. By Lemma 1, we know that there exists a positive constant $\kappa_1$ such as $|\zeta_1 - \alpha_2| \leq \kappa_1$.

Define $e_1(t) = \zeta_1 - \alpha_2$; then, $x_2 = z_2 + e_1(t) + \alpha_2$ and (12) can be rewritten as

$$\dot{z}_1 = r_1(\varphi_{f_1}^T(\bar{x}_i)\dot{\theta}_{f_1} + e_{f_1} + z_2 + e_1(t) - c_1z_1), \quad (16)$$

The adaptive law of $\mathcal{\hat{z}}_{f_1}$ is chosen as

$$\dot{\mathcal{\hat{z}}}_{f_1} = \lambda_{f_1}(z_1\varphi_{f_1}(\bar{x}_i) - \delta_{f_1}\mathcal{\hat{z}}_{f_1}), \quad (17)$$

where $\lambda_{f_1}$ and $\delta_{f_1}$ are positive design parameters.

Step $n$: the time derivative of $z_n$ produces

$$\dot{z}_n = \varphi_{f_n}^T(\bar{x}_i)\mathcal{\hat{z}}_{f_n} + e_{f_n} + \delta_{f_n}\mathcal{\hat{z}}_{f_n} + u - \dot{\zeta}_{n-1}, \quad (20)$$

where $\tau_j$ is design time constant. Define $e_j(t) = \zeta_j - \alpha_{j+1}$, and there exists a positive constant $\kappa_j$ such as $|e_j(t)| \leq \kappa_j$ by Lemma 1. Similar to step 1, we can rewrite (18) as follows:

$$\dot{z}_j = \varphi_{f_j}^T(\bar{x}_i)\dot{\theta}_{f_j} + e_{f_j} + z_{j-1} + e_j(t) - c_jz_j - z_{j-1}. \quad (21)$$

And, the parameter adaptive law of $\mathcal{\hat{z}}_{f_j}$ is designed as

$$\dot{\mathcal{\hat{z}}}_{f_j} = \lambda_{f_j}(z_j\varphi_{f_j}(\bar{x}_i) - \delta_{f_j}\mathcal{\hat{z}}_{f_j}), \quad (22)$$

where $\lambda_{f_j}$ and $\delta_{f_j}$ are positive design parameters.

3.2. Complexity

Consider uncertain nth-order strict-feedback nonlinear system (1) with assumptions 1-2. If virtual controllers (14) and (19), real controller (24), and RBFNN updating laws (17), (22), and (26) are designed, then all signals in the closed-loop system (27) are uniformly ultimately bounded, and then, the boundedness of $z_1$ ensures that $e_1$ satisfies PPB (10).

Proof. Consider the following Lyapunov function:

$$V_1 = \frac{1}{2r_1}z_1^2 + \frac{1}{2\lambda_{f_1}}\mathcal{\hat{z}}_{f_1}^T\mathcal{\hat{z}}_{f_1} + \sum_{i=2}^{n-1}\left( \frac{1}{2}z_i^2 + \frac{1}{2\lambda_{f_i}}\mathcal{\hat{z}}_{f_i}^T\mathcal{\hat{z}}_{f_i} \right). \quad (27)$$

According to (16), (21), and (25) and RBFNN updating laws (17), (22), and (26), we have

$$\dot{V}_1 = \frac{-r_1}{2}z_1^2 - \sum_{i=2}^{n}c_iz_i^2 + \sum_{k=1}^{n}(z_k\mathcal{\hat{z}}_{f_k}(\bar{x}_i)) - \sum_{k=1}^{n}\mathcal{\hat{z}}_{f_k}^T(z_k\varphi_{f_k}(\bar{x}_i) - \delta_{f_k}\mathcal{\hat{z}}_{f_k}) + \sum_{j=1}^{n-1}z_je_j(t), \quad (28)$$

where $r_1$ is a positive design constant.
Let $c_j = c_{j1} + 2c_{j2}$, $c_i = c_{i1} + c_{i2}$, and $c_{j1}, c_{j2} > 0$, $j = 1, 2, \ldots, n - 1$. By using Young’s inequality, we obtain
\[
\delta_k \| \tilde{\theta}_{j_k} \| \leq \frac{\delta_k}{2} \| \tilde{\theta}_{j_k} \|^2 + \frac{\delta_k}{2} \| \tilde{\theta}_{j_k} \|^2,
\]
\[
-c_{j2} z_k^2 + z_k \varepsilon_j (k) \leq \frac{\varepsilon_j^2 (k)}{4c_{j2}} - \varepsilon_j^2 (k), \quad k = 1, 2, \ldots, n,
\]
\[
-c_{j2} z_j^2 + z_j \varepsilon_j (t) \leq \frac{\varepsilon_j^2 (t)}{4c_{j2}} - \varepsilon_j^2 (k), \quad j = 1, 2, \ldots, n - 1.
\]
Substituting (29) into (28), we obtain
\[
\dot{V}_1 \leq \left( c_{i1} + \frac{\varepsilon_j^2}{2r_{n1}} \right) z_i^2 - \sum_{i=1}^n \varepsilon_j^2 (k) - \sum_{i=1}^n \frac{\delta_k}{2} \| \tilde{\theta}_{j_k} \|^2 + R_1,
\]
where $R_1 = \sum_{k=1}^n (\delta_k / 2) \| \tilde{\theta}_{j_k} \|^2 + (\varepsilon_j^2 / 4c_{j2}) + \sum_{i=1}^n (\varepsilon_j^2 / 4c_{j2})$. According to 5, we choose $c_{i1} > (r_{n1} / 2r_{n1})$ and define the following compact sets:
\[
\Omega_{z_1} = \left\{ z_1 \left| z_1 \leq \frac{R_1}{c_{i1} + (r_{n1} / 2r_{n1})} \right. \right\},
\]
\[
\Omega_{z_2} = \left\{ z_2 \left| z_2 \leq \frac{R_1}{c_{i1}} \right. \right\},
\]
\[
\Omega_{\tilde{\theta}_{j_k}} = \left\{ \tilde{\theta}_{j_k} \left| \| \tilde{\theta}_{j_k} \| \leq \frac{2R_1}{\delta_k} \right. \right\}.
\]
Obviously, $z_1 \notin \Omega_{z_2}$ or $z_2 \notin \Omega_{z_1}$ or $\tilde{\theta}_{j_k} \notin \Omega_{\tilde{\theta}_{j_k}}$, we have $\dot{V}_1 < 0$. Thus, $z_1, \tilde{z}_2, z_2$, and $\tilde{\theta}_{j_k}$ are all semiglobally uniformly bounded. According to Lemma 4, the boundedness of $z_1$ ensures that $e_1$ meets PPB (10). This concludes the proof.

Remark 5. In fact, Theorem 1 also implies that all system states $x_1, x_2, \ldots, x_n$ are recurrent for $t \geq T$. Because $e_1$ is limited within PPB (10), i.e., $a_{i0} < x_i - x_j < b_{i0}, \quad t \geq T$, so, from Lemma 3, we obtain that $x_1$ is recurrent. And, because $f_j (x)$ is smooth, this leads to $x_j, x_1, \ldots, x_n$ are recurrent in turn. It also implies that there exists a regression subvector $\varphi_j (x)$ such that $\varphi_j (x)$ satisfies partial PE condition by Lemma 2.

Remark 6. It should be pointed out that RBFNN updating laws (17), (22), and (26) only use the instantaneous data, and it may not estimate the unknown function $f (x)$ accurately. Therefore, in the next section, we will use the composite learning control method to ensure the boundedness of $z_1$ and the accurate estimation of $f (x)$.

3.2. Adaptive Neural Network PPC Method with Learning

In this section, a prediction error $\bar{e}_1 (t)$ is constructed by using online data and instantaneous data. According to Remark 5 in Section 3.1, system states $x_1, x_2, \ldots, x_n$ in system (1) will be recurrent after $T$. So, for $x_i, \varphi_j (x)$, there is a regression subvector $\varphi_j (x)$, which satisfies partial PE condition by Lemma 2, i.e., $\int_{t-\tau}^t \sigma_f \varphi_j (x)dt \geq \sigma_i \Pi$, where $\sigma_i > 0$ and $t \geq T$. So, $f_j (x)$ can also be estimated by $\varphi_j (x) \tilde{\theta}_{j_k}$ instead of $\varphi_j (x) \tilde{\theta}_{j_k}$ after $T$. The specific modification is as follows.

Firstly, substituting $f_j (x) = \varphi_j (x) \tilde{\theta}_{j_k} + e_{\tilde{\theta}_{j_k}} (x) (t \geq T)$ into system (1), one obtains
\[
\dot{x}_1 = \varphi_j (x) \tilde{\theta}_{j_k} + e_{\tilde{\theta}_{j_k}} (x) + x_{i+1}, \quad i = 1, 2, \ldots, n - 1,
\]
\[
\dot{x}_n = \varphi_j (x) \tilde{\theta}_{j_k} + e_{\tilde{\theta}_{j_k}} (x) + u,
\]
where $|e_{\tilde{\theta}_{j_k}} (x)| \leq c_{j}^*, c_{j}^*$ is a positive constant. Let $Q_i (t) := \int_{t-\tau}^t \varphi_j (x) \tilde{\theta}_{j_k} \sigma_f \varphi_j (x)dt$ and $\tau_j (t) := \int_{t-\tau}^t \varphi_j (x) \varphi_j (x) \sigma_f \varphi_j (x)dt$. Multiplying both sides of above equalities in (32) by $\varphi_j (x)$ and integrating equalities over $[t, t - \tau_j]$, one obtains
\[
\begin{aligned}
Q_i (t - \tau_j) + \tau_j (t) &= \int_{t-\tau}^t \varphi_j (x) \sigma_f \varphi_j (x) (x_{i+1} - x_i) dt, \\
Q_n (t - \tau_j) + \tau_j (t) &= \int_{t-\tau}^t \varphi_j (x) \sigma_f \varphi_j (x) (u - u) dt.
\end{aligned}
\]
So, the prediction error $\bar{e}_1 (t)$ is defined as
\[
\bar{e}_1 (t) = Q_i (t - \tau_j) + \tau_j (t) - Q_i (t) \tilde{\theta}_{j_k} = Q_i (t) \tilde{\theta}_{j_k} + \tau_j (t).
\]
Obviously, $\| \tau_j (t) \| \leq \tau_d \bar{e}_1^{'*}$; let $\bar{x} = \max \{x_1, x_2, \ldots, x_n\}$ and $\bar{\bar{e}} = \max \{c_{j}^*, c_{j}^*, \ldots, c_{j}^* \}$; then, for $\forall i$, $\| \tau_j (t) \| \leq \tau_d \bar{e}_1^{'*}$ holds.

Then, we replace $\varphi_j^T (x) \tilde{\theta}_{j_k}$ with $\varphi_j^T (x) \tilde{\theta}_{j_k}$ in (12), (18), and (23) after $T$, and one obtains
\[
\begin{aligned}
\dot{z}_1 &= r_i \left( \varphi_j^T (x) \tilde{\theta}_{j_k} + \varphi_j^T (x) \tilde{\theta}_{j_k} + x_{i+1} + \varepsilon_j (x) - \dot{x}_i - \Pi \right), \\
\dot{z}_j &= \varphi_j^T (x) \tilde{\theta}_{j_k} + \varphi_j^T (x) \tilde{\theta}_{j_k} + x_{j+1} + \varepsilon_j (x) - \dot{\tilde{\theta}}_{j_k}, \\
\dot{z}_n &= \varphi_j^T (x) \tilde{\theta}_{j_k} + \varphi_j^T (x) \tilde{\theta}_{j_k} + u + \varepsilon_j (x) - \tilde{\theta}_{j_k}.
\end{aligned}
\]
where $z_j = x_j - \tilde{\theta}_{j_k}, \tilde{\theta}_{j_k}$ is the estimation of $\varphi_j (x)$, which is the same as the one in Section 3.1, $j = 2, 3, \ldots, n$. And, the modified virtual controller $\alpha_j$ is designed as follows.

According to (35), the virtual control $\alpha_j$ in (14) is modified as
\[
\alpha_j = \begin{cases}
-\varphi_j^T (x) \tilde{\theta}_{j_k} + x_{i+1} - \Pi - c_{j} \bar{e}_1^{'}, & t < T, \\
-\varphi_j^T (x) \tilde{\theta}_{j_k} + x_{j+1} - \Pi - c_{j} \bar{e}_1^{'}, & t \geq T.
\end{cases}
\]
\[ \dot{\theta}_{j_1} = \lambda_{j_1}(z_i \varphi_{j_1}(x_i) - \delta_i \dot{\theta}_{j_1}), \quad t < T, \]
\[ \dot{\theta}_{j_1} = \lambda_{j_1}(z_i \varphi_{j_1}(x_i) + \delta_i \omega_i(t)), \quad t \geq T, \]  
where \( \lambda_{j_1} \) and \( \delta_i \) are positive design parameters.

The virtual control \( \alpha_{j_1} \) in (19) is modified as
\[ \alpha_{j_1} = \begin{cases} \dot{-\theta}_{j_1} \varphi_{j_1}(x_i) + \dot{\theta}_{j_1} - c_i z_i - z_i, & t < T, \\ \dot{-\theta}_{j_1} \varphi_{j_1}(x_i) + \dot{\theta}_{j_1} - c_i z_i - z_i, & t \geq T. \end{cases} \]

Similarly, composite learning laws of \( \dot{\theta}_{j_1} \) and \( \dot{\theta}_{j_2} \) are designed as
\[ \dot{\theta}_{j_1} = \lambda_{j_1}(z_i \varphi_{j_1}(x_i) - \delta_i \dot{\theta}_{j_1}), \quad t < T, \]
\[ \dot{\theta}_{j_2} = \lambda_{j_1}(z_i \varphi_{j_1}(x_i) + \delta_i \omega_i(t)), \quad t \leq T, \]
where \( \lambda_{j_1} \) and \( \delta_i \) are positive design parameters.

The controller \( u \) in (24) is modified as
\[ u = \begin{cases} -\dot{-\theta}_{j_1} \varphi_{j_1}(x_i) + \dot{\theta}_{j_1} - c_i z_i - z_i, & t < T, \\ -\dot{-\theta}_{j_2} \varphi_{j_2}(x_i) + \dot{\theta}_{j_2} - c_i z_i - z_i, & t \geq T. \end{cases} \]
and composite learning laws for \( \dot{\theta}_{j_1} \) and \( \dot{\theta}_{j_2} \) are designed as

\[ V_2 = -\left(c_1 + \frac{r_1}{2c_1^2}\right)z_1^2 - \sum_{i=2}^{n} c_i z_i^2 + \sum_{k=1}^{n} \left(z_k \varphi_{j_1}(x_k) + \varphi_{j_2}(x_k) \dot{\theta}_{j_2}\right) - \sum_{k=1}^{n} \dot{\theta}_{j_1}
\begin{align*}
\text{where } &c_i = c_{j_1} + 2c_{j_2}, c_n = c_{j_1} + c_{j_2}, \\
&\text{and } c_{j_1}, c_{j_2} > 0. \text{ By using Young's inequality and Lemma 2, one obtains}
\end{align*}
\[ -c_{j_2} z_j^2 + z_j \varphi_{j_2}(t) \leq \frac{\varphi_{j_2}^2(t)}{4c_{j_2}} \leq \frac{\varphi_{j_2}^2(t)}{4c_{j_2}}, \]
\[ -c_{j_2} z_j^2 + z_j \varphi_{j_2}(x_k) \leq \frac{\varphi_{j_2}^2(x_k)}{4c_{j_2}} \leq \frac{\varphi_{j_2}^2(x_k)}{4c_{j_2}}, \]
\[ -\delta_i \dot{\theta}_{j_2} Q_k(t) \dot{\theta}_{j_2} \leq -\sigma_k \delta_i \dot{\theta}_{j_2} \varphi_{j_2}(x_k), \]
\[ -\frac{\delta_i \sigma_k}{2} (\dot{\theta}_{j_2} \dot{\theta}_{j_2} + 2 \varphi_{j_2}^2(t)) \leq \frac{\delta_i \sigma_k}{2} \varphi_{j_2}^2(t) \leq \frac{\delta_i (\varphi_{j_2}^2)^2}{2\sigma_k}. \]
Substituting (44) into (43), one has
\[ \dot{V}_2 \leq -c_1 z_1^2 - \sum_{i=2}^{n} c_i z_i^2 - \sum_{k=1}^{n} \sigma_k \delta_i \dot{\theta}_{j_2} \varphi_{j_2}(x_k), \quad \forall t \in [T, +\infty). \]

\[ V_2 \leq V_2(T) \exp(-\gamma_2 t) + R_2, \]
where \( R_2 = \sum_{k=1}^{n} \left( (\delta_i (\tau_{d_k} \varphi_{j_2}^2)/2\sigma_k) + (\varphi_{j_2}^2/4c_{j_2}) \right), \) and \( \gamma_2 \) is given by
\[ \gamma_2 = \min \left\{ 2r_1 c_{j_1} + (r_1 / r_1), 2c_{j_2}, \ldots, 2c_{j_n} - \lambda_{j_1} \delta_i, \ldots, -\lambda_{j_2} \delta_n \right\}, \]
which yields
\[ V_2 \leq V_2(T) \exp(-\gamma_2 t) + R_2. \]

Inequality (47) implies that \( z_i \) and \( \dot{\theta}_{j_1} \) exponentially converge to \( \sqrt{(R_2/\gamma_2)} \)-neighborhood of zero. And, then the
boundedness of $z_1$ derives that $e_1$ satisfies PPB (10) by Lemma 4. This concludes the proof. □

Remark 7. The difference between Theorem 1 and Theorem 2 is that Theorem 2 introduces prediction errors $\tilde{e}_i(t)$, and online recorded data and instantaneous data is used for neural weights update. The comparative simulation results show the control effect of the two methods directly and highlight the proposed composite learning PPC method proposed in this paper is better.
Figure 4: Trajectories of (a) $\tilde{\theta}_2$ and (b) $\tilde{\theta}_3$ by using the CLPPC method.

Figure 5: Control trajectories of $u$.

Figure 6: Tracking error $e_1$ by the CLPPC method with (a) $T = 3\text{s}$, (b) $T = 10\text{s}$, and (c) $T = 20\text{s}$.
4. Simulation Studies

In this section of the simulation, the Chua system [25] is considered as an example:

\[
\begin{align*}
\dot{x}_1 &= \beta x_2, \\
\dot{x}_2 &= f_2(\mathbf{x}_2) + x_3, \\
\dot{x}_3 &= f_3(\mathbf{x}_3) + u, \\
y &= x_1,
\end{align*}
\]

where \( f_2(\mathbf{x}_2) = x_1 - x_2 \) and \( f_3(\mathbf{x}_3) = y((x_2 - x_3 - m_1 x_3 - 1)/(2(m_0 - m_1)(|x_3 + 1| - |x_3 - 1|))) \). Denote the method in Section 3.1 as the ANPPC method and the method in Section 3.2 as the CLPPC method. The reference signal \( x_d = \cos(\tau) \).

For the NN design of the ANPPC method, the number of hidden nodes is chosen as \( 3 \times 3 \) and \( 3 \times 3 \times 3 \), respectively, and their centers \( \mathbf{x}_2 \) and \( \mathbf{x}_3 \) are being evenly spaced on \([-2,2] \times [-2,2] \times [-1,1] \) and \([-2,2] \times [-0.5,0.5] \times [-1,1] \). Initial weights are \( \bar{\theta}_f(0) = 0 \) and \( \bar{\theta}_g(0) = 0 \) and the initial states are \( [x_1(0), x_2(0), x_3(0)]^T = [0.03, 0.3, -0.1]^T \). Parameters of system (48) are selected as \( \beta = -25 \), \( \gamma = 15.6 \), \( m_0 = 1.028 \), and \( m_1 = 0.714 \), and design parameters of ANPPC method are chosen as \( \lambda_f = \lambda_g = 10 \), \( c_1 = c_2 = c_3 = 10 \), \( r_1 = r_2 = 0.002 \), and \( \delta_1 = \delta_2 = 0.005 \). FTTPs are selected as \( p(t, T, -4, -0.05) \) and \( p(t, T, -0.5, 0.05) \) with \( T = 5s \). For the parameters in the CLPPC method, the selection of parameters is the same as that in the ANPPC method when \( t < T = 5s \) and the parameters \( \lambda_f = \lambda_g = 10 \), \( \delta_1 = \delta_2 = 0.005 \) and other parameters are the same as those in ANPPC method when \( t \geq T = 5s \). Notice that the system states \( x_1, x_2, \) and \( x_3 \) are recurrent after \( T = 5s \) seconds, so Lemma 2 can be applied to redesign the NN after 5 seconds: the number of hidden nodes is chosen as \( 3 \times 3 \) and \( 3 \times 3 \times 3 \), respectively, and their centers \( \mathbf{x}_2 \) and \( \mathbf{x}_3 \) are being evenly spaced on \([-1,1] \times [-1,1] \times [-0.05,0.05] \) and \([-1,1] \times [-0.5,0.5] \times [-1,1] \). And, initial weights are \( \bar{\theta}_f(0) = 0 \) and \( \bar{\theta}_g(0) = 0 \). The simulation results are shown in Figures 1-6. Figure 1 shows that the tracking error \( e_1 \) can be limited within PPB (10) under the ANPPC method and the CLPPC method, but Figure 1(b) shows that, after 10s, the control effect of \( e_1 \) is greatly improved by using the CLPPC method compared with the ANPPC method. The ANPPC method and CLPPC method are used to estimate unknown functions \( f_2(\mathbf{x}_2) \) and \( f_3(\mathbf{x}_3) \). The estimation effect is shown in Figures 2 and 3, respectively. Obviously, the estimation effect of the CLPPC method is better than that of the ANPPC method after 10s. Figure 4 shows composite learning laws of \( \bar{\theta}_f \) and \( \bar{\theta}_g \) after \( T = 5s \). Figure 5 shows the control inputs \( u \) under above methods. In the whole process of control input, the energy consume is not significant difference, except that the CLPPC method has two jumps at 5 seconds and 10 seconds. In addition, compared with the prescribed performance control method used in [11–16], the proposed CLPPC method in this paper can realize the finite time adjustment of tracking error \( e_1 \), which is shown in Figure 6. In general, the control effect of tracking error and estimation effect of unknown functions by using the CLPPC method are better than that by using the ANPPC method.

5. Conclusion

In this paper, the composite learning prescribed performance control method is investigated for uncertain strict-feedback system. Firstly, a new finite-time performance was proposed and predictive errors were introduced by applying online recorded date and instantaneous date to update NN weights. The boundedness of the closed-loop system was ensured by the proposed CLPPC method. Simulation results show the effectiveness of the proposed method. Because the reference signal may not be recurrent, it will lead to partial PE condition is not tenable. So, the accurate estimation of unknown function under an interval excitation (IE) condition becomes the next research direction.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

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

Acknowledgments

This work was supported in part by the Natural Science Research Projects in Anhui Universities under Grant nos. KJ2020A0644, KJ2019ZD48, and KJ2019A0695 and Anhui Natural Science Foundation under Grant no. 2008085MF200.

References

[1] J. A. Farrell, M. Polycarpou, M. Sharma, and W. Dong, "Command filtered backstepping," IEEE Transactions on Automatic Control, vol. 54, no. 6, pp. 1391–1395, 2009.
[2] O. Aung, Y. Lin, and Q. Zhao, "Fuzzy based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form," in Proceedings of the 2009 9th International Conference on Electronic Measurement & Instruments, pp. 3–499, IEEE, Beijing, China, August 2009.
[3] J. Yu, L. Zhao, H. Yu, C. Lin, and W. Dong, "Fuzzy finite-time command filtered control of nonlinear systems with input saturation," IEEE Transactions on Cybernetics, vol. 48, no. 8, pp. 2378–2387, 2017.
[4] W. S. Chen, "Adaptive backstepping dynamic surface control for systems with periodic disturbances using neural networks," IET Control Theory & Applications, vol. 3, no. 10, pp. 1383–1394, 2009.
[5] B. Xu and F. Sun, "Composite intelligent learning control of strict-feedback systems with disturbance," IEEE Transactions on Cybernetics, vol. 48, no. 2, pp. 730–741, 2017.
[6] 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.
[7] J. Yu, P. Shi, and L. Zhao, "Finite-time command filtered backstepping control for a class of nonlinear systems," Automatica, vol. 92, pp. 173–180, 2018.
[8] Y. Pan, T. Sun, and H. Yu, “Composite adaptive dynamic surface control using online recorded data,” *International Journal of Robust and Nonlinear Control*, vol. 26, no. 18, pp. 3921–3936, 2016.

[9] H. Liu, Y. Pan, S. Li, and Y. Chen, “Adaptive fuzzy backstepping control of fractional-order nonlinear systems,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2209–2217, 2017.

[10] M. Chen, Q. Wu, C. Jiang, and B. Jiang, “Guaranteed transient performance based control with input saturation for near space vehicles,” *Science China Information Sciences*, vol. 57, no. 5, pp. 1–12, 2014.

[11] S.-I. Han and J.-M. Lee, “Recent fuzzy neural network backstepping control for the prescribed output tracking performance of nonlinear dynamic systems,” *ISA Transactions*, vol. 53, no. 1, pp. 33–43, 2014.

[12] C. Cheng, Y. Zhang, and S. Liu, “Neural observer-based adaptive prescribed performance control for uncertain nonlinear systems with input saturation,” *Neurocomputing*, vol. 370, pp. 94–103, 2019.

[13] J. Li, J. Du, and X. Hu, “Robust adaptive prescribed performance control for dynamic positioning of ships under unknown disturbances and input constraints,” *Ocean Engineering*, vol. 206, p. 107254, 2020.

[14] 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.

[15] Q. Yang and M. Chen, “Adaptive neural prescribed performance tracking control for near space vehicles with input nonlinearity,” *Neurocomputing*, vol. 174, pp. 780–789, 2016.

[16] X. Bu, G. He, and D. Wei, “A new prescribed performance control approach for uncertain nonlinear dynamic systems via back-stepping,” *Journal of the Franklin Institute*, vol. 355, no. 17, pp. 8510–8536, 2018.

[17] W. Xiang and H. Liu, “Fuzzy adaptive prescribed performance tracking control for uncertain nonlinear systems with unknown control gain signs,” *IEEE Access*, vol. 7, pp. 149867–149877, 2019.

[18] H. Wang, W. Bai, X. Zhao, and P. X. Liu, “Finite-time-prescribed performance-based adaptive fuzzy control for strict-feedback nonlinear systems with dynamic uncertainty and actuator faults,” *IEEE Transactions on Cybernetics*, vol. 99, pp. 1–13, 2021.

[19] 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, no. 12, pp. 2626–2634, 2014.

[20] Y. Pan and M. J. Er, “Enhanced adaptive fuzzy control with optimal approximation error convergence,” *IEEE Transactions on Fuzzy Systems*, vol. 21, no. 6, pp. 1123–1132, 2013.

[21] Y. Pan, Y. Liu, and H. Yu, “Online data-driven composite adaptive backstepping control with exact differentiators,” *International Journal of Adaptive Control and Signal Processing*, vol. 30, no. 5, pp. 779–789, 2015.

[22] H. Liu, S. Li, J. Cao, G. Li, A. Alsaedi, and F. E. Alsaadi, “Adaptive fuzzy prescribed performance controller design for a class of uncertain fractional-order nonlinear systems with external disturbances,” *Neurocomputing*, vol. 219, pp. 422–430, 2017.

[23] C. Wang and D. J. Hill, “Learning from neural control,” *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 130–146, 2006.