SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

September 2024, Vol. 67, Iss. 9, 192202:1–192202:20
https://doi.org/10.1007/s11432-023-3996-3

Deterministic learning-based neural output-feedback control for a class of nonlinear sampled-data systems

Yu ZENG[1], Fukai ZHANG[2*], Tianrui CHEN[2] & Cong WANG[2]

[1]*School of Automation Science and Engineering, South China University of Technology, Guangzhou* 510641*, China;*
[2]*School of Control Science and Engineering, Shandong University, Jinan* 250061*, China*

Received 9 August 2023/Revised 27 October 2023/Accepted 19 December 2023/Published online 19 August 2024

Abstract This study investigates the deterministic learning (DL)-based output-feedback neural control for a class of nonlinear sampled-data systems with prescribed performance (PP). Specifically, first, a sampled-data observer is employed to estimate the unavailable system states for the Euler discretization model of the transformed system dynamics. Then, based on the observations and backstepping method, a discrete neural network (NN) controller is constructed to ensure system stability and achieve the desired tracking performance. The noncausal problem encountered during the controller deduction process is resolved using a command filter. Moreover, the regression characteristics of the NN input signals are demonstrated with the observed states. This ensures that the radial basis function NN, based on DL theory, meets the partial persistent excitation condition. Subsequently, a class of discrete linear time-varying systems is proven to be exponentially stable, achieving partial convergence of neural weights to their optimal/actual values. Consequently, accurate modeling of unknown closed-loop dynamics is achieved along the system trajectory from the output-feedback control. Finally, a knowledge-based controller is developed using the modeling results. This controller not only enhances the control performance but also ensures the PP of the tracking error. The effectiveness of the scheme is illustrated through simulation results.

Keywords deterministic learning, neural networks, adaptive neural control, predefined performance, nonlinear sampled-data systems

1 Introduction

Throughout the previous decades, it has been demonstrated that intelligent controllers such as neural network (NN) controllers and fuzzy logic-based controllers can effectively deal with system uncertainties or unmodeled system dynamics [1, 2]. Among various intelligent control techniques, the focus on NN-based adaptive control has increased, whether in theoretical studies [3,4] or practical applications, such as proton exchange membrane fuel cell systems [5], autonomous surface vehicles [6], and robot systems [7].

In modern practical scenarios, control systems must be implemented on digital hardware, such as microcontrollers or programmable logic controllers. This requirement makes discrete-time/sampled-data controllers indispensable for the effective management and control of various dynamic processes [8]. However, in comparison with the extensive literature available on continuous-time systems (e.g., [9–11]), designing controllers for nonlinear discrete-time/sampled-data systems poses significant challenges, primarily because of the complexities associated with the difference of the Lyapunov function [8] and the noncausal problem [12, 13]. Nevertheless, considerable effort has been made to publish some interesting studies, such as [12–16]. The study conducted in [12] focused on adaptive NN predictive control for a specific category of discrete nonlinear pure-feedback systems. In [13], an adaptive NN controller was constructed for a specific category of discrete-time nonlinear systems with multiple dead-zone inputs and multiple outputs. In [14–16], an event-triggered output feedback control method was introduced for nonlinear discrete systems with constrained communication and nonlinear uncertainties.

It is noteworthy that for an extended period, the emphasis on the tracking performance of NN control systems has primarily been on ensuring the steady-state convergence of tracking errors [17]. However,

* Corresponding author (email: zhangfukai@sdu.edu.cn)

certain transient performances such as the response speed and convergence rate are also critical and have received increasing attention [18, 19]. In practice, controlled systems need to achieve a certain performance in both transient and steady states. Regarding the aforementioned problems, prescribed performance control (PPC) was originally proposed based on adaptive control in [20]. It makes a specific category of nonlinear systems (which is feedback linearizable) satisfy certain prescribed performance (PP). Later, PPC was investigated for various control systems [21–25] and practical applications [18,19,26,27]. The key point of PPC is to transform the tracking errors from the constrained forms into equivalent unconstrained forms. Thus, the problem changes to design controllers to make the unconstrained tracking errors bounded [28]. To date, most current PPC methods have been specifically designed for continuous-time dynamical systems, rendering them unsuitable for direct applications to discrete-time/sampled-data systems [17]. In recent studies [29–32], initial investigations were conducted on discrete-time dynamical systems with PP based on sliding mode control. More recently, an NN-embedded adaptive controller based on the backstepping technique was introduced in [33] for a specific category of nonlinear sampled-data systems. However, the above-mentioned studies mainly focused on addressing the issues of system stability and tracking control while lacking learning capabilities.

Learning ability requires knowledge learning and reuse in dynamic processes and is important in intelligent control systems [34]. Recently, deterministic learning (DL) theory was developed by Wang and Hill in [35, 36], which can achieve knowledge acquisition, storage, and reuse in uncertain dynamical systems. According to [35, 36], by using radial basis function (RBF) networks, when the input signals are recurrent, the RBF NNs will satisfy the partial persistent excitation (PE) condition along the recurrent trajectories. This process enables the corresponding NN weights to converge to their true values, thereby achieving locally accurate modeling of the uncertain dynamics. By employing DL method, the control performance in continuous-time systems is enhanced through learning control [25, 37–39]. Subsequently, based on the sampled-data DL method proposed in [40] for nonlinear sampled-data systems, for example, affine nonlinear systems [41] and strict-feedback nonlinear systems [42], learning control schemes were investigated. Existing controller construction based on sampled-data DL theory primarily concentrates on dynamical systems with the full state available. However, in numerous real-world applications, it is common to obtain only specific outputs. This leads to a noncausal problem, as stated in [12,13,43], in the design process of the backstepping-based discrete controller. This makes the analysis of learning control in systems with only outputs available more challenging than in systems with the full state available.

To solve these problems and inspired by previous studies, this study aims to investigate the issue of output-feedback control based on DL theory for a class of nonlinear sampled-data systems with PP. To facilitate the controller design, the Euler discretization model of the transformed system dynamics is used with a sampled-data observer to obtain unmeasurable system states. Based on the estimated states, an adaptive NN controller is developed using the backstepping technique to ensure the PP of the tracking error, and the NN is embedded to approximate the unknown sampled-data closed-loop dynamics. The noncausal problem faced in the design process is addressed using a command filter. Then, the transformed unconstrained error is proven to converge exponentially to the origin based on Lyapunov's stability theorem. Based on the transformation rule, the original constrained tracking error is guaranteed to converge. Subsequently, based on the convergence of signals toward their recurrent reference trajectories, the NN satisfies the partial PE condition. Furthermore, a discrete linear time-varying (DLTV) system, which consists of the tracking error and NN weight estimation errors, is proven to be exponentially convergent based on satisfying the partial PE condition. When the NN weights converge, an accurate modeling of the sampled-data closed-loop dynamics is achieved. Finally, a knowledge-based controller is constructed, which not only enhances the control performance, but also guarantees the PP of the tracking error.

The contributions of this study are as follows:

(1) By employing the Euler discretization model of the transformed system dynamics and based on a sampled-data observer, the discrete NN output-feedback controller design can be implemented while avoiding the noncausal problem using a discrete-time command filter. Moreover, the prescribed tracking performance is proven to be achieved in the transient and steady states.

(2) Based on the estimation of the unmeasurable states by the sampled-data observer and the convergence of the output signal, the regression property of the NN input signals can be proven. This result ensures that the partial PE condition of the RBF NN is satisfied. Furthermore, a class of DLTV systems is guaranteed to be exponentially stable based on the satisfaction of the partial PE condition.

(3) After the learning process, a knowledge-based output-feedback controller is designed, which can

enhance control performance without readaptation of the NN weights online, while the tracking error adheres to the PP.

The structure of the subsequent sections is as follows. Section 2 presents the system description along with some preliminary preparations. In Section 3, the controller design process with the PP is introduced. Section 4 presents the learning control for a class of output-feedback nonlinear sampled-data systems. In Section 5, a knowledge-based controller is proposed. Section 6 provides a simulation analysis, and Section 7 presents the conclusions.

2 Problem statement & preliminary preparation

2.1 Dynamical systems

Consider the nonlinear uncertain closed-loop system:

$$\begin{cases} \dot{x}(t) = Ax(t) + B\left(f(x(t)) + g(x(t))u(t)\right), \\ y(t) = Cx(t), \end{cases} \tag{1}$$

where

$$A = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{bmatrix}, \ B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \ C = \begin{bmatrix} 1, 0, \ldots, 0 \end{bmatrix},$$

$x(t) = [x_1(t), \ldots, x_n(t)]^{\mathrm{T}} \in \mathbb{R}^n$ is the vector of the system states, $u(t) \in \mathbb{R}$ is the control signal, $y(t) \in \mathbb{R}$ is the output, $f(\cdot)$, $g(\cdot)$: $\mathbb{R}^n \to \mathbb{R}$ are two unmodeled functions, which are nonlinear and smooth. In this article, we assume that the measurable variable is only the output $y(t)$ and the sign of $g(\cdot)$ remains fixed. With no loss of generality, we suppose $0 < \underline{g} \leqslant g(\cdot) \leqslant \bar{g}$, where $\underline{g}$ and $\bar{g}$ are constants.

Remark 1. There exist a lot of real-world industrial systems, such as robot manipulators [44], unmanned aerial vehicles [45], and permanent magnet motors [46], whose dynamics take the form of (1) or are equivalent to (1) through diffeomorphic coordinate transformations. Therefore, studying this class of nonlinear systems holds essential practical significance.

The reference signal of $y(t)$ is defined as $y^d(t)$, and it is presumed to be recurrent.

2.2 Introduction of the PP

Define the output tracking error as

$$e_1(t) = y(t) - y^d(t). \tag{2}$$

The PP of $e_1(t)$ establishes the constrained boundaries that require compliance under both transient and steady-state conditions and requires the following inequality to be satisfied

$$-\rho_1 \xi(t) < e_1(t) < \rho_2 \xi(t), \tag{3}$$

where $\rho_1, \rho_2 > 0$ are constants, and $\xi(t)$ is the predefined performance function, which is positive, bounded, and decreasing. We consider the following smooth function $\xi(t)$ as that in [23]:

$$\xi(t) = (\xi(0) - \xi_\infty)\, \mathrm{e}^{-\varphi t} + \xi_\infty, \tag{4}$$

where $\xi(t)$ satisfies $\lim_{t\to\infty} \xi(t) = \xi_\infty$, with the initial value $\xi(0) > \xi_\infty > 0$, and $\varphi > 0$ is a constant.

Considering the Euler discretization model (16) with the predefined tracking performance described in (17), in this study, a discrete-time output-feedback NN control law is constructed to achieve tracking control and accurate modeling of the unknown sampled-data closed-loop dynamics. Furthermore, a knowledge-based output-feedback controller is constructed, which can improve control performance, with the PP of the tracking error satisfied.

2.3 Preliminaries

The general RBF NNs, denoted by $W^{\mathrm{T}}S(Z)$, have been explored in the available literature thoroughly (e.g., [47, 48]) for their ability of approximation to any continuous function $f(Z)$ within a given domain $\Omega_Z \to R$. The RBF vector $S(Z)$ is represented as $[s_1(\|Z-p_1\|), \ldots, s_m(\|Z-p_m\|)]^{\mathrm{T}}$, where $s_i(\|Z-p_i\|) = \exp[-(Z-p_i)^{\mathrm{T}}(Z-p_i)/q_i^2]$ $(i \in \{1, \ldots, m\})$ is an RBF with the center $p_i = [p_{i1}, \ldots, p_{iN}]$, the width q_i, and the input vector $Z \in \Omega_Z \subset \mathbb{R}^N$ (Ω_Z is a compact set). $W \in \mathbb{R}^m$ is the weight vector corresponding to $S(Z)$. As mentioned in [36], the Gaussian RBF vector is bounded as $\|S(Z)\| \leqslant s^*$, with $s^* > 0$ being a constant. Additionally, as stated in [49], $f(Z)$ is approximated as $f(Z) = W^{*\mathrm{T}}S(Z) + \epsilon(Z)$, where W^* denotes the ideal weight vector, $\epsilon(Z)$ is the approximation error which satisfies $\|\epsilon(Z)\| \leqslant \epsilon^*$, with $\epsilon^* > 0$ being the acceptable error.

Alternatively, considering the localization property of the RBFs, $f(Z)$ is approximated by using neurons positioned within a local region along the trajectory of Z (see [35, 50]) as $f(Z) = W_\zeta^{*\mathrm{T}}S_\zeta(Z) + \epsilon_\zeta$, where $S_\zeta(\cdot) \in \mathbb{R}^{m_\zeta}$ $(m_\zeta < m)$ is the subvector of $S(Z)$, $W_\zeta^* \in \mathbb{R}^{m_\zeta}$ denotes the corresponding ideal weight subvector, and ϵ_ζ is the approximation error which satisfies $|\epsilon_\zeta| < \epsilon^*$.

In the field of nonlinear system identification and adaptive control, the PE condition is of vital importance, since if it is not satisfied, accurate approximation of unknown functions is often not guaranteed [51]. However, it is difficult to characterize and verify the PE condition in practical applications. To solve this problem, the partial PE condition has been studied by Wang and Hill [35, 36] and forms a fundamental basis of the DL theory.

Lemma 1 ([35, 36])**.** For a recurrent trajectory $Z(t_k) \in \Omega_Z \subset \mathbb{R}^m$ ($k \in Z_+$ is the sampling number), the subvector $S_\zeta(Z)$, which consists of RBFs whose centers are placed within a local region along $Z(t_k)$, satisfies the PE condition almost always.

Remark 2. The recurrent trajectory in this paper, which can be a periodic, quasi-periodic, or chaotic trajectory [35, 36], etc., means that regardless of the initial point, there exists a $t_k(\wp) > 0$ with $\wp > 0$ such that the trajectory returns to the $\wp$-neighborhood of any point visited by the trajectory in a finite time not greater than $t_k(\wp)$.

Since the only available variable is the output $y = x_1$, to estimate the unknown states $x_2, \ldots, x_n$, the following high gain observer is employed.

Lemma 2 ([52])**.** Consider the following sampled-data observer:

$$\begin{cases} \hat{x}(t_{k+1}) = \hat{x}(t_k) + T_s A\hat{x}(t_k) + \Lambda\Xi(\hat{y}(t_k) - y(t_k)), \\ \hat{y}(t_k) = C\hat{x}(t_k), \end{cases} \tag{5}$$

where $T_s > 0$ is the sampling period, $\hat{x}(t_k) = [\hat{x}_1(t_k), \ldots, \hat{x}_n(t_k)]^{\mathrm{T}} \in \mathbb{R}^n$ is the vector of the observer states, $\hat{y}(t_k) \in \mathbb{R}$ is the observer output, $\Lambda = \mathrm{diag}\{1, 1/\varepsilon, \ldots, 1/\varepsilon^{n-1}\}$, and the vector $\Xi = [r_1, \ldots, r_n]^{\mathrm{T}}$ is designed such that $I + T_s\varepsilon^{-1}A + \Xi C$ is stable, i.e., the matrix has eigenvalues within the unit circle. Let $F(x(t_k), u(t_k)) = f(x(t_k)) + g(x(t_k))u(t_k)$, and $\tilde{F}(t_k) = F(\hat{x}(t_k), u(t_k)) - F(x(t_k), u(t_k))$. According to [53], suppose the nonlinear function $F(\cdot, u)$ is bounded and locally Lipschitz in the operation set of the system, i.e.,

$$|\tilde{F}(t_k)| = |f(\hat{x}(t_k)) + g(\hat{x}(t_k))u(t_k) - f(x(t_k)) - g(x(t_k))u(t_k)| \leqslant L\|\tilde{e}(t_k)\|, \tag{6}$$

where $\tilde{e}(t_k) = \hat{x}(t_k) - x(t_k) = [\tilde{e}_1(t_k), \ldots, \tilde{e}_n(t_k)]^{\mathrm{T}} \in \mathbb{R}^n$, $L > 0$ is the Lipschitz constant. Then, for $t_k \geqslant t_k^*$, with $t_k^* > 0$ being a constant, the following inequality holds:

$$|\tilde{e}_i(t_k)| \leqslant \ell T_s \varepsilon^{n-i}\bar{F},\ i = 1, \ldots, n, \tag{7}$$

where $\ell > 0$ is a constant, and $\bar{F} = \sup_x |F(x(t_k), u(t_k))|$.

It can be deduced from Lemma 2 that once the gain of the observer is chosen large and the sampling period is small enough, then, within a finite time, $\tilde{e}(t_k)$ reaches a small region around the origin.

To circumvent the noncausal problem faced during the controller deduction process, the command filter is employed.

Lemma 3 ([43])**.** Define the following command filter:

$$\begin{cases} \alpha_1^f(t_{k+1}) = \alpha_1^f(t_k) + T_s\omega_n\alpha_2^f(t_k), \\ \alpha_2^f(t_{k+1}) = \alpha_2^f(t_k) - T_s\omega_n(2\zeta_1\alpha_2^f(t_k) + \alpha_1^f(t_k) - \alpha(t_k)), \end{cases} \tag{8}$$

where $\alpha(t_k)$ and $\alpha_1^f(t_k)$ are the filter input and output, respectively, $\zeta_1 \in (0,1]$, and $\omega_n > 0$. If for any $t_k \geqslant 0$, $\alpha(t_k)$ fulfills the conditions $|\alpha(t_{k+1}) - \alpha(t_k)| \leqslant \pi_1$ and $|\alpha(t_{k+2}) - 2\alpha(t_{k+1}) + \alpha(t_k)| \leqslant \pi_2$, where $\pi_1, \pi_2 > 0$ are constants, and $\alpha_1^f(0) = \alpha(0)$, $\alpha_2^f(0) = 0$, then there exist ζ_1 and ω_n such that $|\alpha_1^f(t_k) - \alpha(t_k)| \leqslant \mu$ $(\forall \mu > 0)$, and $|\alpha_1^f(t_{k+1}) - \alpha_1^f(t_k)|$ is bounded.

3 Controller design with PP

To make the system stable and the tracking error converge with PP, in this section, based on the sampled-data observer, an adaptive neural learning control law is formulated.

3.1 Tracking error transformation with PP

To address the issue of constrained tracking control stated above, a nonlinear function $\Psi(z_1(t))$ (which is smooth and increasing) is employed for the transformation of $e_1(t)$ to an unconstrained equivalent form $z_1(t)$. The transfer function $\Psi(z_1(t))$ satisfies

$$\begin{cases} -\rho_1 < \Psi(z_1(t)) < \rho_2, \\ \lim\limits_{z_1(t)\to-\infty} \Psi(z_1(t)) = -\rho_1, \lim\limits_{z_1(t)\to\infty} \Psi(z_1(t)) = \rho_2. \end{cases} \tag{9}$$

Based on (3) and (9), the constrained error is equal to

$$e_1(t) = \xi(t)\Psi(z_1(t)). \tag{10}$$

To obtain the unconstrained error $z_1(t)$, first, we analyze the invertibility of the function $\Psi(z_1(t))$. Note that $\Psi(z_1(t))$ is strictly increasing and $\xi(t) \neq 0$, so the inverse of $\Psi(z_1(t))$ always exists and $z_1(t)$ is formulated as

$$z_1(t) = \Psi^{-1}\left(e_1(t)/\xi(t)\right). \tag{11}$$

The transfer function in this paper is chosen as that in [28]:

$$\Psi(z_1(t)) = \frac{\rho_2 \mathrm{e}^{(z_1(t)+\tau)} - \rho_1 \mathrm{e}^{-(z_1(t)+\tau)}}{\mathrm{e}^{(z_1(t)+\tau)} + \mathrm{e}^{-(z_1(t)+\tau)}}, \tag{12}$$

where $\tau = \frac{1}{2}\ln\frac{\rho_1}{\rho_2}$. Note that $\Psi(z_1(t))$ in (12) not only satisfies the properties in (9) but also satisfies $\Psi(0) = 0$. Combining with (10), this means that $e_1(t) = 0$ when $z_1(t) = 0$. Then, based on (10) and (12), $z_1(t)$ is acquired by

$$z_1(t) = \frac{1}{2}\ln\left(\frac{\rho_1 + e_1(t)/\xi(t)}{\rho_2 - e_1(t)/\xi(t)}\right) - \frac{1}{2}\ln\frac{\rho_1}{\rho_2}. \tag{13}$$

From (11) and (13), it is deduced that

$$\dot{z}_1(t) = \varrho(t)\left(\dot{e}_1(t) - \frac{e_1(t)\dot{\xi}(t)}{\xi(t)}\right), \tag{14}$$

where $\varrho(t) = \frac{1}{2}(\frac{1}{\rho_1\xi(t)+e_1(t)} + \frac{1}{\rho_2\xi(t)-e_1(t)})$. It can be deduced from the PP inequality (3) that $\varrho(t) > 0$ for all $t > 0$.

Then, from (14) and (1), we have the transformed system dynamics as that in [33]:

$$\begin{cases} \dot{z}_1(t) = \varrho(t)\Big(\dot{e}_1(t) - \dfrac{e_1(t)\dot{\xi}(t)}{\xi(t)}\Big), \\ \dot{x}_i(t) = x_{i+1}(t),\ i = 2, \ldots, n-1, \\ \dot{x}_n(t) = f(x(t)) + g(x(t))u(t). \end{cases} \tag{15}$$

Note that from (9) and (10), when $z_1(t)$ is bounded, $e_1(t)$ satisfies the PP inequality (3). So, after the transformation, the original problem of tracking control with PP is converted into a problem of making the unconstrained error $z_1(t)$ bounded.

3.2 Discrete neural controller design

This subsection introduces the design process of an adaptive discrete-time output feedback NN controller to solve the tracking problem with PP. At first, the Euler model of (15) is given by employing the Euler discretization technique [33],

$$\begin{cases} z_1(t_{k+1}) =z_1(t_k) + \varrho(t_k)\left(e_1(t_{k+1}) - e_1(t_k) - \dfrac{e_1(t_k)\left(\xi(t_{k+1}) - \xi(t_k)\right)}{\xi(t_k)}\right) \\ \qquad\quad =z_1(t_k) + \varrho(t_k)\left(e_1(t_{k+1}) - \dfrac{e_1(t_k)\xi(t_{k+1})}{\xi(t_k)}\right), \\ x_i(t_{k+1}) =x_i(t_k) + T_s x_{i+1}(t_k),\ i = 2,\ldots,n-1, \\ x_n(t_{k+1}) =x_n(t_k) + T_s\left(f(x(t_k)) + g(x(t_k))u(t_k)\right), \end{cases} \tag{16}$$

where $x(t_k) = [x_1(t_k),\ldots,x_n(t_k)]^{\mathrm{T}} \in \mathbb{R}^n$, $\varrho(t_k) = \frac{1}{2}(\frac{1}{\rho_1\xi(t_k)+e_1(t_k)} + \frac{1}{\rho_2\xi(t_k)-e_1(t_k)})$, and $\xi(t_k) = (\xi(0)-\xi_\infty)\mathrm{e}^{-\varphi t_k}+\xi_\infty$. The tracking error $e_1(t_k)$ is required to satisfy the discrete-time PP inequality as follows:

$$-\rho_1\xi(t_k) < e_1(t_k) < \rho_2\xi(t_k). \tag{17}$$

In what follows, the deduction of the discrete-time NN control law is introduced by employing the backstepping technique.

Step 1. Since the reference signal is known, $y^d(t_{k+1})$ can be applied at the kth sampling instant. Based on the Euler discretization technique, by combining with $x_1(t_{k+1}) = x_1(t_k)+T_s x_2(t_k)$ and $\tilde{e}_2(t_k) = \hat{x}_2(t_k) - x_2(t_k)$, it is deduced that

$$e_1(t_{k+1}) = x_1(t_{k+1}) - y^d(t_{k+1}) = x_1(t_k) + T_s(\hat{x}_2(t_k) - \tilde{e}_2(t_k)) - y^d(t_{k+1}). \tag{18}$$

From (16) and (18), we have

$$z_1(t_{k+1}) =z_1(t_k) + \varrho(t_k)\Big(T_s(\hat{x}_2(t_k) - \tilde{e}_2(t_k)) + \delta_1(t_k)\Big), \tag{19}$$

where $\delta_1(t_k) = x_1(t_k) - y^d(t_{k+1}) - \frac{e_1(t_k)\xi(t_{k+1})}{\xi(t_k)}$.

Let $z_2(t_k) = \hat{x}_2(t_k) - \alpha_{1,1}^f(t_k)$ and $\Delta\alpha_1(t_k) = \alpha_{1,1}^f(t_k) - \alpha_1(t_k)$, where $\alpha_1(t_k)$ is the virtual control law, $\alpha_{1,1}^f(t_k)$ is the command filter output described in Lemma 3 with input $\alpha_1(t_k)$, and there exists an arbitrarily small μ_1 such that $|\Delta\alpha_1(t_k)| \leqslant \mu_1$. Then, it follows from (19) that

$$z_1(t_{k+1}) = z_1(t_k) + \varrho(t_k)(T_s(z_2(t_k) + \Delta\alpha_1(t_k) + \alpha_1(t_k) - \tilde{e}_2(t_k)) + \delta_1(t_k)). \tag{20}$$

We construct $\alpha_1(t_k)$ as

$$\alpha_1(t_k) = -\lambda_1 z_1(t_k) - \delta_1(t_k)/T_s, \tag{21}$$

where $\lambda_1 > 0$ and is to be designed. Then, by substituting (21) into (20), we have

$$z_1(t_{k+1}) =(1 - T_s\lambda_1\varrho(t_k))z_1(t_k) + T_s\varrho(t_k)(z_2(t_k) + \iota(t_k)), \tag{22}$$

where $\iota(t_k) = \Delta\alpha_1(t_k) - \tilde{e}_2(t_k)$, λ_1 is chosen such that $0 < T_s\lambda_1\varrho(t_k) < 1$. From Lemma 2 and 3, $|\iota(t_k)| \leqslant \bar{\iota} := \ell T_s\varepsilon^{n-2}\bar{F} + \mu_1$ for all $t_k \geqslant t_k^*$.

Next, we construct the candidate for the Lyapunov function as follows:

$$V_1(t_k) = z_1^2(t_k). \tag{23}$$

Let $\Delta V_1(t_k) = V_1(t_{k+1}) - V_1(t_k)$. Based on (23) and (22), we have

$$\begin{aligned} \frac{\Delta V_1(t_k)}{\varrho^2(t_k)} =&\frac{z_1^2(t_{k+1}) - z_1^2(t_k)}{\varrho^2(t_k)} \\ =&\frac{T_s\lambda_1(-2 + T_s\lambda_1\varrho(t_k))}{\varrho(t_k)}z_1^2(t_k) + T_s^2\left(z_2^2(t_k) + \iota^2(t_k)\right) \end{aligned}$$

$$+\frac{2T_s(1-T_s\lambda_1\varrho(t_k))}{\varrho(t_k)}z_1(t_k)\left(z_2(t_k)+\iota(t_k)\right)+2T_s^2z_2(t_k)\iota(t_k). \tag{24}$$

From Young's inequality and $\varrho(t_k)>0$, it is deduced that

$$\frac{2T_s(1-T_s\lambda_1\varrho(t_k))}{\varrho(t_k)}z_1(t_k)z_2(t_k)\leqslant\frac{T_s(1-T_s\lambda_1\varrho(t_k))}{\varrho(t_k)}\left(\lambda_1z_1^2(t_k)+\frac{1}{\lambda_1}z_2^2(t_k)\right), \tag{25}$$

$$\frac{2T_s(1-T_s\lambda_1\varrho(t_k))}{\varrho(k)}z_1(t_k)\iota(t_k)\leqslant\frac{T_s(1-T_s\lambda_1\varrho(t_k))}{\varrho(t_k)}\left(\lambda_1z_1^2(t_k)+\frac{1}{\lambda_1}\iota^2(t_k)\right), \tag{26}$$

$$2T_s^2z_2(t_k)\iota(t_k)\leqslant T_s\left(\frac{z_2^2(t_k)}{\lambda_1\varrho(t_k)}+\lambda_1T_s^2\varrho(t_k)\iota^2(t_k)\right). \tag{27}$$

So by substituting (25)–(27), and $|\iota(t_k)|\leqslant\bar{\iota}$ into (24), we have

$$\Delta V_1(t_k)\leqslant\left(-T_s^2\lambda_1^2z_1^2(t_k)+\frac{2T_s}{\lambda_1\varrho(t_k)}z_2^2(t_k)+T_s\frac{1+\lambda_1^2T_s^2\varrho^2(t_k)}{\lambda_1\varrho(t_k)}\bar{\iota}^2\right)\varrho^2(t_k). \tag{28}$$

Step 2. From $z_2(t_k)=\hat{x}_2(t_k)-\alpha_{1,1}^f(t_k)$ and the observer (5), it follows

$$\begin{aligned}z_2(t_{k+1})=&\hat{x}_2(t_k)+T_s\hat{x}_3(t_k)+\frac{r_2}{\varepsilon}\left(\hat{y}(t_k)-y(t_k)\right)-\alpha_{1,1}^f(t_{k+1})\\=&z_2(t_k)+T_s\hat{x}_3(t_k)+\delta_2(t_k),\end{aligned} \tag{29}$$

where $\delta_2(t_k)=\frac{r_2}{\varepsilon}\left(\hat{y}(t_k)-y(t_k)\right)+\alpha_{1,1}^f(t_k)-\alpha_{1,1}^f(t_{k+1})$. Note that by employing the command filter, $\alpha_{1,1}^f(t_{k+1})$, which represents the value of $\alpha_{1,1}^f(t_k)$ in the future, is available at the current sampling instant. This avoids the noncausal problem encountered when using $\alpha_1(t_{k+1})$, as stated in [42].

Let $z_3(t_k)=\hat{x}_3(t_k)-\alpha_{2,1}^f(t_k)$ and $\Delta\alpha_2(t_k)=\alpha_{2,1}^f(t_k)-\alpha_2(t_k)$, where $\alpha_{2,1}^f(t_k)$ is the command filter output in Lemma 3 with input $\alpha_2(t_k)$, and $\alpha_2(t_k)$ is a virtual control law. From Lemma 3, there exists an arbitrarily small μ_2 such that $|\Delta\alpha_2(t_k)|\leqslant\mu_2$. If $\alpha_2(t_k)$ is constructed as

$$\alpha_2(t_k)=-\lambda_2z_2(t_k)-\delta_2(t_k)/T_s, \tag{30}$$

where $\lambda_2>0$ represents a design constant. Then Eq. (29) becomes

$$z_2(t_{k+1})=(1-T_s\lambda_2)z_2(t_k)+T_s\left(z_3(t_k)+\Delta\alpha_2(t_k)\right), \tag{31}$$

where the parameter λ_2 is chosen such that $0<T_s\lambda_2<1$.

By constructing the candidate for the Lyapunov function as follows:

$$V_2(t_k)=z_2^2(t_k) \tag{32}$$

and considering (31), we can obtain

$$\begin{aligned}\Delta V_2(t_k)=&V_2(t_{k+1})-V_2(t_k)=z_2^2(t_{k+1})-z_2^2(t_k)\\=&T_s\lambda_2\left(-2+T_s\lambda_2\right)z_2^2(t_k)+T_s^2\left(z_3^2(t_k)+\left(\Delta\alpha_2(t_k)\right)^2\right)\\&+2T_s(1-T_s\lambda_2)z_2(t_k)\left(z_3(t_k)+\Delta\alpha_2(t_k)\right)+2T_s^2z_3(t_k)\Delta\alpha_2(t_k).\end{aligned} \tag{33}$$

From Young's inequality, we have

$$2T_s(1-T_s\lambda_2)z_2(t_k)z_3(t_k)\leqslant T_s(1-T_s\lambda_2)\left(\lambda_2z_2^2(t_k)+\frac{1}{\lambda_2}z_3^2(t_k)\right), \tag{34}$$

$$2T_s(1-T_s\lambda_2)z_2(t_k)\Delta\alpha_2(t_k)\leqslant T_s(1-T_s\lambda_2)\left(\lambda_2z_2^2(t_k)+\frac{1}{\lambda_2}(\Delta\alpha_2(t_k))^2\right), \tag{35}$$

$$2T_s^2z_3(t_k)\Delta\alpha_2(t_k)\leqslant T_s\left(\frac{z_3^2(t_k)}{\lambda_2}+\lambda_2T_s^2(\Delta\alpha_2(t_k))^2\right). \tag{36}$$

Then, it follows from (33)–(36) and $|\Delta\alpha_2(t_k)| \leqslant \mu_2$ that

$$\Delta V_2(t_k) \leqslant -T_s^2\lambda_2^2 z_2^2(t_k) + \frac{2T_s}{\lambda_2} z_3^2(t_k) + T_s\frac{1+T_s^2\lambda_2^2}{\lambda_2}\mu_2^2. \tag{37}$$

Step i $(3 \leqslant i \leqslant n-1)$**.** Let $z_i(t_k) = \hat{x}_i(t_k) - \alpha_{i-1,1}^f(t_k)$. Then, by combining with (5), we have

$$\begin{aligned} z_i(t_{k+1}) =& \hat{x}_i(t_k) + T_s\hat{x}_{i+1}(t_k) + \frac{r_i}{\varepsilon^{i-1}}\left(\hat{y}(t_k) - y(t_k)\right) - \alpha_{i-1,1}^f(t_{k+1}) \\ =& z_i(t_k) + T_s\hat{x}_{i+1}(t_k) + \delta_i(t_k), \end{aligned} \tag{38}$$

where $\delta_i(t_k) = \frac{r_i}{\varepsilon^{i-1}}\left(\hat{y}(t_k) - y(t_k)\right) + \alpha_{i-1,1}^f(t_k) - \alpha_{i-1,1}^f(t_{k+1})$.

Let $z_{i+1}(t_k) = \hat{x}_{i+1}(t_k) - \alpha_{i,1}^f(t_k)$ and $\Delta\alpha_i(t_k) = \alpha_{i,1}^f(t_k) - \alpha_i(t_k)$, where $\alpha_i(t_k)$ is a virtual controller, $\alpha_{i,1}^f(t_k)$ is the command filter output, and the input is $\alpha_i(t_k)$. From Lemma 3, we have $|\Delta\alpha_i(t_k)| \leqslant \mu_i$, where $\mu_i > 0$ is small. By combining with (38), we construct $\alpha_i(t_k)$ as

$$\alpha_i(t_k) = -\lambda_i z_i(t_k) - \delta_i(t_k)/T_s, \tag{39}$$

where $\lambda_i > 0$ is a constant to be designed. Then Eq. (38) becomes

$$z_i(t_{k+1}) = (1 - T_s\lambda_i)z_i(t_k) + T_s\left(z_{i+1}(t_k) + \Delta\alpha_i(t_k)\right), \tag{40}$$

where the parameter λ_i is designed so that $0 < T_s\lambda_i < 1$.

The Lyapunov function candidate in the ith step is constructed as

$$V_i(t_k) = z_i^2(t_k). \tag{41}$$

Similar to (33)–(37), we have

$$\begin{aligned} \Delta V_i(t_k) =& V_i(t_{k+1}) - V_i(t_k) \\ \leqslant& -T_s^2\lambda_i^2 z_i^2(t_k) + \frac{2T_s}{\lambda_i} z_{i+1}^2(t_k) + T_s\frac{1+T_s^2\lambda_i^2}{\lambda_i}\mu_i^2. \end{aligned} \tag{42}$$

Step n**.** From $z_n(t_k) = \hat{x}_n(t_k) - \alpha_{n-1,1}^f(t_k)$, and by combining with (16) and $\tilde{e}_n(t_k) = \hat{x}_n(t_k) - x_n(t_k)$, it follows that

$$\begin{aligned} z_n(t_{k+1}) =& x_n(t_k) + T_s\left(f(x(t_k)) + g(x(t_k))u(t_k)\right) + \tilde{e}_n(t_{k+1}) - \alpha_{n-1,1}^f(t_{k+1}) \\ =& z_n(t_k) + \alpha_{n-1,1}^f(t_k) - \tilde{e}_n(t_k) + T_s\left(f(\hat{x}(t_k)) + g(\hat{x}(t_k))u(t_k)\right) \\ &+ T_s\left(f(x(t_k)) + g(x(t_k))u(t_k) - f(\hat{x}(t_k)) - g(\hat{x}(t_k))u(t_k)\right) \\ &+ \tilde{e}_n(t_{k+1}) - \alpha_{n-1,1}^f(t_{k+1}) \\ =& z_n(t_k) + T_s g(\hat{x}(t_k))\left(h(Z(t_k) + u(t_k))\right) - T_s\tilde{F}(t_k) - \tilde{e}_n(t_k) + \tilde{e}_n(t_{k+1}), \end{aligned} \tag{43}$$

where $h(Z(t_k)) = \frac{f(\hat{x}(t_k)) + \frac{\alpha_{n-1,1}^f(t_k) - \alpha_{n-1,1}^f(t_{k+1})}{T_s}}{g(\hat{x}(t_k))}$ is the unknown sampled-data closed-loop dynamics, $Z(t_k) = [\hat{x}^{\mathrm{T}}(t_k), \frac{\alpha_{n-1,1}^f(t_k) - \alpha_{n-1,1}^f(t_{k+1})}{T_s}]^{\mathrm{T}} \in \mathbb{R}^{n+1}$. To ensure system stability in the presence of the unknown functions $f(\cdot)$ and $g(\cdot)$, we formulate the following control law:

$$u(t_k) = -\lambda_n z_n(t_k) - \hat{W}^{\mathrm{T}}(t_k)S(Z(t_k)), \tag{44}$$

where $\lambda_n > 0$ is a constant to be designed, $\hat{W}^{\mathrm{T}}(t_k)S(Z(t_k))$ is an RBF NN used to approximate $h(Z(t_k))$. The corresponding ideal RBF NN is given as

$$h(Z(t_k)) = W^{*\mathrm{T}}S(Z(t_k)) + \epsilon(Z(t_k)), \tag{45}$$

where W^* is the optimal value of $\hat{W}(t_k)$, $\epsilon(Z(t_k))$ is the approximation error, and $|\epsilon(Z(t_k))| < \epsilon^*$, with $\epsilon^* > 0$ being the acceptable error.

Substituting (44) and (45) into (43), we have

$$z_n(t_{k+1}) = \beta_1(t_k) z_n(t_k) + T_s g(\hat{x}(t_k)) \cdot \left(\tilde{\epsilon}(Z(t_k)) - \tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)) \right), \tag{46}$$

where $\beta_1(t_k) = 1 - T_s \lambda_n g(\hat{x}(t_k))$, the parameter λ_n is designed so that $0 < \underline{\beta}_1 \leqslant \bar{\beta}_1 < 1$, $\underline{\beta}_1 = 1 - T_s \lambda_n \bar{g}$, $\bar{\beta}_1 = 1 - T_s \lambda_n \underline{g}$, $\tilde{W}(t_k) = \hat{W}(t_k) - W^*$ represents the weight estimation error, and $\tilde{\epsilon}(Z(t_k)) = \epsilon(Z(t_k)) - \frac{1}{T_s g(\hat{x}(t_k))}(T_s \tilde{F}(t_k) + \tilde{e}_n(t_k) - \tilde{e}_n(t_{k+1}))$. From (6) and (7), we have $|\tilde{F}(t_k)| \leqslant L\|\tilde{e}(t_k)\| \leqslant l_1 L \|\tilde{e}(t_k)\|_1 \leqslant l_1 L \sum_{i=1}^{n} \ell T_s \varepsilon^{n-i} \bar{F} < \frac{l_1 L \ell T_s \bar{F}}{1-\varepsilon}$, where $l_1 > 0$ is a constant. By combining with the boundedness of $g(\cdot)$, it can be deduced that $|\tilde{\epsilon}(Z(t_k))| < \tilde{\epsilon}^* := \epsilon^* + 2\ell \bar{F}/\underline{g} + \frac{l_1 L \ell T_s \bar{F}}{\underline{g}(1-\varepsilon)}$. The NN weight updating law is chosen as

$$\hat{W}(t_{k+1}) = \hat{W}(t_k) + T_s \Gamma \left[S(Z(t_k)) z_n(t_{k+1}) - \sigma \hat{W}(t_k) \right], \tag{47}$$

where $\Gamma > 0$ is a symmetric matrix, $\sigma > 0$ is a constant to be designed. Since $\tilde{W}(t_k) = \hat{W}(t_k) - W^*$, it follows that

$$\tilde{W}(t_{k+1}) = \gamma_1 \tilde{W}(t_k) + T_s \Gamma S(Z(t_k)) z_n(t_{k+1}) - \gamma_2 W^*, \tag{48}$$

where $\gamma_1 = I - \sigma T_s \Gamma$, $\gamma_2 = \sigma T_s \Gamma$, I is an identity matrix, and σ is chosen such that the eigenvalues of γ_1 are within $(0, 1)$.

The following Lyapunov function candidate in this step is considered:

$$V_n(t_k) = z_n^2(t_k) + \tilde{W}^{\mathrm{T}}(t_k) \Gamma^{-1} \tilde{W}(t_k). \tag{49}$$

Then

$$\begin{aligned} \Delta V_n(t_k) =& V_n(t_{k+1}) - V_n(t_k) \\ =& z_n^2(t_{k+1}) - z_n^2(t_k) + \tilde{W}^{\mathrm{T}}(t_{k+1}) \Gamma^{-1} \tilde{W}(t_{k+1}) - \tilde{W}^{\mathrm{T}}(t_k) \Gamma^{-1} \tilde{W}(t_k). \end{aligned} \tag{50}$$

From (46), it can be deduced that

$$\begin{aligned} z_n^2(t_{k+1}) - z_n^2(t_k) =& - T_s \lambda_n g(\hat{x}(t_k))(1 + \beta_1(t_k)) z_n^2(t_k) + T_s^2 g^2(\hat{x}(t_k))(\tilde{\epsilon}^2(Z(t_k)) + (\tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)))^2) \\ &+ 2T_s \beta_1(t_k) g(\hat{x}(t_k)) z_n(t_k)(\tilde{\epsilon}(Z(t_k)) - \tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k))) \\ &- 2T_s^2 g^2(\hat{x}(t_k)) \tilde{\epsilon}(Z(t_k)) \tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)). \end{aligned} \tag{51}$$

Since

$$2T_s \beta_1(t_k) g(\hat{x}(t_k)) z_n(t_k) \tilde{\epsilon}(Z(t_k)) \leqslant T_s \beta_1(t_k) g(\hat{x}(t_k)) \left(\lambda_n z_n^2(t_k) + \frac{\tilde{\epsilon}^2(Z(t_k))}{\lambda_n} \right), \tag{52}$$

$$- 2T_s \beta_1(t_k) g(\hat{x}(t_k)) z_n(t_k) \tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)) \leqslant T_s \beta_1(t_k) g(\hat{x}(t_k)) \left(\lambda_n z_n^2(t_k) + \frac{(\tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)))^2}{\lambda_n} \right), \tag{53}$$

$$- 2T_s^2 g^2(\hat{x}(t_k)) \tilde{\epsilon}(Z(t_k)) \tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)) \leqslant T_s^2 g^2(\hat{x}(t_k))(\tilde{\epsilon}^2(Z(t_k)) + (\tilde{W}^{\mathrm{T}}(t_k) S(Z(t_k)))^2). \tag{54}$$

By substituting (52)–(54) into (51), it follows that

$$\begin{aligned} & z_n^2(t_{k+1}) - z_n^2(t_k) \\ & \leqslant -T_s \lambda_n g(\hat{x}(t_k)) \left(1 - \beta_1(t_k)\right) z_n^2(t_k) + T_s^2 g(\hat{x}(t_k)) \left(2g(\hat{x}(t_k)) + \frac{\beta_1(t_k)}{T_s \lambda_n} \right) (\|\tilde{W}(t_k)\|^2 s^{*2} + \tilde{\epsilon}^{*2}) \\ & = -\left(1 - \beta_1(t_k)\right)^2 z_n^2(t_k) + T_s^2 g(\hat{x}(t_k)) \left(2g(\hat{x}(t_k)) + \frac{\beta_1(t_k)}{T_s \lambda_n} \right) (\|\tilde{W}(t_k)\|^2 s^{*2} + \tilde{\epsilon}^{*2}), \end{aligned} \tag{55}$$

where s^* is the bound of $\|S(Z(t_k))\|$, i.e., $\|S(Z(t_k))\| < s^*$.

On the other hand, from (48), we have

$$\tilde{W}^{\mathrm{T}}(t_{k+1}) \Gamma^{-1} \tilde{W}(t_{k+1}) - \tilde{W}^{\mathrm{T}}(t_k) \Gamma^{-1} \tilde{W}(t_k)$$

$$= -\sigma T_s \tilde{W}^{\mathrm{T}}(t_k)(I+\gamma_1)\tilde{W}(t_k) + \sigma T_s W^{*\mathrm{T}}\gamma_2 W^* + \frac{T_s}{\sigma}S^{\mathrm{T}}(Z(t_k))\gamma_2 S(Z(t_k))z_n^2(t_{k+1}) + 2T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1 S(Z(t_k))z_n(t_{k+1}) - 2\sigma T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1 W^* - 2T_s S^{\mathrm{T}}(Z(t_k))\gamma_2 W^* z_n(t_{k+1}). \quad (56)$$

Since

$$2T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1 S(Z(t_k))z_n(t_{k+1}) \leqslant \sigma T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1\tilde{W}(t_k) + \frac{T_s}{\sigma}S^{\mathrm{T}}(Z(t_k))\gamma_1 S(Z(t_k))z_n^2(t_{k+1}), \quad (57)$$

$$-2\sigma T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1 W^* \leqslant \sigma T_s\tilde{W}^{\mathrm{T}}(t_k)\gamma_1\tilde{W}(t_k) + \sigma T_s W^{*\mathrm{T}}\gamma_1 W^*, \quad (58)$$

$$-2T_s S^{\mathrm{T}}(Z(t_k))\gamma_2 W^* z_n(t_{k+1}) \leqslant \sigma T_s W^{*\mathrm{T}}\gamma_2 W^* + \frac{T_s}{\sigma}S^{\mathrm{T}}(Z(t_k))\gamma_2 S(Z(t_k))z_n^2(t_{k+1}). \quad (59)$$

By substituting (57)–(59) into (56), it follows that

$$\begin{aligned}&\tilde{W}^{\mathrm{T}}(t_{k+1})\Gamma^{-1}\tilde{W}(t_{k+1}) - \tilde{W}^{\mathrm{T}}(t_k)\Gamma^{-1}\tilde{W}(t_k)\\ &\leqslant -\sigma T_s\tilde{W}^{\mathrm{T}}(t_k)(I-\gamma_1)\tilde{W}(t_k) + \sigma T_s W^{*\mathrm{T}}\gamma_3 W^* + \frac{T_s}{\sigma}S^{\mathrm{T}}(Z(t_k))\gamma_3 S(Z(t_k))z_n^2(t_{k+1})\\ &\leqslant \sigma T_s(\bar{\lambda}_{\gamma_1}-1)\|\tilde{W}(t_k)\|^2 + \sigma T_s\bar{\lambda}_{\gamma_3}\|W^*\|^2 + \varsigma_1 z_n^2(t_{k+1}),\end{aligned} \quad (60)$$

where $\gamma_3 = 2\gamma_2+\gamma_1 = I+\sigma T_s\Gamma$, $\bar{\lambda}_{\gamma_1}$ and $\bar{\lambda}_{\gamma_3}$ represent the maximum eigenvalues of γ_1 and γ_3, respectively, and $\varsigma_1 = \frac{T_s}{\sigma}\bar{\lambda}_{\gamma_3}s^{*2}$.

By substituting (55) and (60) into (50), we have

$$\begin{aligned}\Delta V_n(t_k) \leqslant& \sigma T_s(\bar{\lambda}_{\gamma_1}-1)\|\tilde{W}(t_k)\|^2 + \sigma T_s\bar{\lambda}_{\gamma_3}\|W^*\|^2 + (\varsigma_1+1)\,z_n^2(t_{k+1}) - z_n^2(t_k)\\ \leqslant& \sigma T_s(\bar{\lambda}_{\gamma_1}-1)\|\tilde{W}(t_k)\|^2 + \sigma T_s\bar{\lambda}_{\gamma_3}\|W^*\|^2 + (\varsigma_1+1)\Bigg[-(1-\beta_1(t_k))^2 z_n^2(t_k)\\ &+ T_s^2 g(\hat{x}(t_k))\left(2g(\hat{x}(t_k)) + \frac{\beta_1(t_k)}{T_s\lambda_n}\right)(\|\tilde{W}(t_k)\|^2 s^{*2} + \tilde{\epsilon}^{*2})\Bigg] + \varsigma_1 z_n^2(t_k)\\ \leqslant& -\varsigma_2 z_n^2(t_k) - \varsigma_3\|\tilde{W}(t_k)\|^2 + \varsigma_4,\end{aligned} \quad (61)$$

where $\varsigma_2 = \left(1-\bar{\beta}_1\right)^2(\varsigma_1+1) - \varsigma_1$, $\varsigma_3 = -T_s^2\bar{g}(2\bar{g} + \frac{\bar{\beta}_1}{T_s\lambda_n})(\varsigma_1+1)\,s^{*2} - \sigma T_s(\bar{\lambda}_{\gamma_1}-1)$, $\varsigma_4 = T_s^2\bar{g}(2\bar{g} + \frac{\bar{\beta}_1}{T_s\lambda_n})(\varsigma_1+1)\,\tilde{\epsilon}^{*2} + \sigma T_s\bar{\lambda}_{\gamma_3}\|W^*\|^2$.

Theorem 1. Consider the closed-loop system consisting of the Euler discretization model (16), the sampled-data observer (5), the discrete-time command filter (8), the control law (44), and the NN weight updating law (47). For the initial values satisfying (17), we have (i) all the variables in the Euler discretization model (16) are bounded; (ii) within a finite time T_1, the tracking error e_1 will converge exponentially, with the PP described in (17), to a small region around the origin.

Proof. (i) First, consider the candidate for the Lyapunov function as follows:

$$V(t_k) = \sum_{i=1}^{n} V_i(t_k) = \sum_{i=1}^{n} z_i^2(t_k) + \tilde{W}^{\mathrm{T}}(t_k)\Gamma^{-1}\tilde{W}(t_k). \quad (62)$$

Then, from (28), (37), (42), and (61), it can be deduced that

$$\begin{aligned}\Delta V(t_k) = \sum_{i=1}^{n}\Delta V_i(t_k) \leqslant& -T_s^2\lambda_1^2\varrho^2(t_k)z_1^2(t_k) - \left(T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1}\right)z_2^2(t_k) - \sum_{i=3}^{n-1}\left(T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}\right)z_i^2(t_k)\\ &- \left(\varsigma_2 - \frac{2T_s}{\lambda_{n-1}}\right)z_n^2(t_k) - \varsigma_3\|\tilde{W}(t_k)\|^2 + \varsigma_4 + \bar{\mu},\end{aligned} \quad (63)$$

where $\bar{\mu} = T_s\frac{1+\lambda_1^2T_s^2\varrho^2(t_k)}{\lambda_1}\bar{\iota}^2\varrho(t_k) + \sum_{i=2}^{n-1}T_s\frac{1+T_s^2\lambda_i^2}{\lambda_i}\mu_i^2$. If the parameters are chosen such that

$$\begin{cases} T_s^2\lambda_2^2 - \dfrac{2T_s\varrho(t_k)}{\lambda_1} > 0, \\ T_s^2\lambda_i^2 - \dfrac{2T_s}{\lambda_{i-1}} > 0,\ i = 3,\ldots,n-1, \\ \varsigma_2 - \dfrac{2T_s}{\lambda_{n-1}} > 0, \\ \varsigma_3 > 0, \end{cases} \tag{64}$$

then, we have from (63) that $\Delta V(t_k) < 0$ if any of the following conditions holds

$$\begin{aligned} &|z_1(t_k)| > \sqrt{(\varsigma_4+\bar{\mu})}/\left(T_s\lambda_1\varrho(t_k)\right), \\ &|z_2(t_k)| > \sqrt{(\varsigma_4+\bar{\mu})\,/\,(T_s^2\lambda_2^2 - 2T_s\varrho(t_k)/\lambda_1)}, \\ &|z_i(t_k)| > \sqrt{(\varsigma_4+\bar{\mu})\,/\,(T_s^2\lambda_i^2 - 2T_s/\lambda_{i-1})},\ i = 3,\ldots,n-1, \\ &|z_n(t_k)| > \sqrt{(\varsigma_4+\bar{\mu})\,/\,(\varsigma_2 - 2T_s/\lambda_{n-1})}, \\ &\|\tilde{W}(t_k)\| > \sqrt{(\varsigma_4+\bar{\mu})\,/\varsigma_3}. \end{aligned} \tag{65}$$

According to Lyapunov's stability theorem [8], it can be obtained from the analysis above that $z_i(t_k)$ $(i = 1,\ldots,n)$ and $\tilde{W}(t_k)$ are bounded. From the boundedness of W^* and the equation $\tilde{W}(t_k) = \hat{W}(t_k) - W^*$, we have that $\hat{W}(t_k)$ is bounded. According to the analysis presented in [54], when employing the forward difference technique, the stability of the continuous-time signal is achieved if its corresponding sampling-time signal is stable. Thus, from the boundedness and stability of the sampling-time signal $z_1(t_k)$, we can obtain the boundedness of the continuous-time signal $z_1(t)$. Based on the error transfer equation (10), $e_1(t)$ and thus $e_1(t_k)$ are bounded. Since $y^d(t_k)$ is bounded and $x_1(t_k) = e_1(t_k) + y^d(t_k)$, the boundedness of $x_1(t_k)$ can be deduced. On the other hand, since $y^d(t_k)$ and the PP function $\xi(t_k)$ are bounded, from (21), we have that $\alpha_1(t_k)$ is bounded, and then the filtered value $\alpha_{1,1}^f(t_k)$ is bounded from Lemma 3. By combining with the boundedness of $z_2(t_k)$ and $\alpha_{1,1}^f(t_k)$, $\hat{x}_2(t_k)$ is bounded. From Lemma 2, $\hat{x}_2(t_k)$ converges to $x_2(t_k)$ in a finite time, so $x_2(t_k)$ is bounded. Recursively, $x_i(t_k)$ $(i = 3,\ldots,n)$ are bounded, and $u(t_k)$ is bounded by combining with the boundedness of $S(Z(t_k))$. Consequently, the variables in the Euler discretization model (16) are all bounded.

(ii) Consider the following candidate for the Lyapunov function:

$$V_z(t_k) = \sum_{i=1}^{n} z_i^2(t_k). \tag{66}$$

Following the similar steps in Subsection 3.2, from (28), (37), (42), and (55), we have

$$\begin{aligned} \Delta V_z(t_k) =& V_z(t_{k+1}) - V_z(t_k) = \sum_{i=1}^{n-1}\Delta V_i(t_k) + z_n^2(t_{k+1}) - z_n^2(t_k) \\ =& -T_s^2\lambda_1^2\varrho^2(t_k)z_1^2(t_k) - \left(T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1}\right)z_2^2(t_k) - \sum_{i=3}^{n-1}\left(T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}\right)z_i^2(t_k) \\ &- \left((1-\beta_1(t_k))^2 - \frac{2T_s}{\lambda_{n-1}}\right)z_n^2(t_k) + T_s^2g(\hat{x}(t_k))\left(2g(\hat{x}(t_k)) + \frac{\beta_1(t_k)}{T_s\lambda_n}\right)(\|\tilde{W}(t_k)\|^2s^{*2} + \tilde{\epsilon}^{*2}) + \bar{\mu} \\ \leqslant& -\theta_1V_z(t_k) + \theta_2(t_k), \end{aligned} \tag{67}$$

where $\theta_1 = \min\{T_s^2\lambda_1^2\varrho^2(t_k),\ T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1},\ T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}\ (i = 3,\ldots,n-1),\ (1-\bar{\beta}_1)^2 - \frac{2T_s}{\lambda_{n-1}}\}$, $\theta_2(t_k) = \bar{\mu} + T_s^2\bar{g}(2\bar{g} + \frac{\bar{\beta}_1}{T_s\lambda_n})(\|\tilde{W}(t_k)\|^2s^{*2} + \tilde{\epsilon}^{*2})$. It can be deduced from the analysis in (i) of this proof that $\theta_2(t_k)$ is bounded, so there is a constant $\bar{\theta}_2 > 0$ such that $|\theta_2(t_k)| < \bar{\theta}_2$. Then, from (67), we have

$$V_z(t_{k+1}) \leqslant (1-\theta_1)V_z(t_k) + \bar{\theta}_2. \tag{68}$$

From the analysis in the backstepping steps, we have $0 < T_s\lambda_1\varrho(t_k) < 1$, $0 < T_s\lambda_i < 1$ $(i = 2, \ldots, n-1)$. From (64), $0 < T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1}$ and $0 < T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}$ $(i = 3, \ldots, n-1)$ hold. So, $0 < T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1} < T_s^2\lambda_2^2 < 1$, $0 < T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}} < T_s^2\lambda_i^2 < 1$ $(i = 3, \ldots, n-1)$. Furthermore, from (46), we have $0 < \beta_1(t_k) < 1$. Then, it can be deduced from (61) and $\varsigma_1 > 0$ that $\varsigma_2 = (1-\bar{\beta}_1)^2 + ((1-\bar{\beta}_1)^2 - 1)\varsigma_1 < (1-\bar{\beta}_1)^2$. So, by combining with (64), we have $0 < \varsigma_2 - \frac{2T_s}{\lambda_{n-1}} < (1-\bar{\beta}_1)^2 - \frac{2T_s}{\lambda_{n-1}} < (1-\bar{\beta}_1)^2 < 1$. Therefore, it can be concluded that $0 < \theta_1 < 1$. From (68), we have $V_z(t_k) \leqslant (1-\theta_1)^k V_z(0) + \frac{\bar{\theta}_2(1-(1-\theta_1)^k)}{1-(1-\theta_1)} < (1-\theta_1)^k V_z(0) + \frac{\bar{\theta}_2}{\theta_1}$. So given any $c_1 > \frac{\bar{\theta}_2}{\theta_1}$, we have $z_i(t_k) < \sqrt{c_1}$, $\forall t_k > T_1$ $(i = 1, \ldots, n)$, with T_1 being a finite time instant. Since c_1 is arbitrarily small when the parameters are chosen appropriately, $z_i(t_k)$ $(i \in \{1, \ldots, n\})$ will converge to a small region around the origin exponentially. Therefore, the convergence of $z_i(t)$ is guaranteed from [54], and with the transformation (10), within a finite time T_1, the tracking error $e_1(t)$ and thus $e_1(t_k)$ will converge to a small region around the origin exponentially.

4 Closed-loop learning control

Based on the convergence of system signals shown in Theorem 1, the exponential convergence of a DLTV system will be obtained in this section. Therefore, closed-loop learning is realized by using the adaptive neural controller, and the learned knowledge can then be reused in the construction of the knowledge-based controller.

Theorem 2. Consider the closed-loop system consisting of the Euler discretization model (16), the sampled-data observer (5), the discrete-time command filter (8), the control law (44), and the NN weight updating law (47) with the initial value $\hat{W}(0) = 0$. Then, for initial conditions satisfying (17), along the recurrent system trajectory, the NN weight estimation error $\tilde{W}(t_k)$ converges to a small region around the origin and the unknown sampled-data closed-loop dynamics $h(Z(t_k))$ can be locally accurately approximated around the recurrent signal $Z(t_k)$ by $\hat{W}^{\mathrm{T}}(t_k)S(Z(t_k))$ and the constant NN $\bar{W}^{\mathrm{T}}S(Z(t_k))$, with $\bar{W}$ being

$$\bar{W} = \mathrm{mean}_{t_k \in [t_{k_a}, t_{k_b}]}\hat{W}(t_k), \tag{69}$$

where $t_{k_b} > t_{k_a} > T_1$, and $[t_{k_a}, t_{k_b}]$ is a time segment after $\hat{W}(t_k)$ converges.

Proof. From the analysis in Theorem 1, when $t_k > T_1$, e_1 will converge exponentially to a small neighborhood around the origin, with the PP described in (17). So $x_1(t_k)$ tracks the recurrent orbit of $y^d(t_k)$. Besides, since $z_1(t_k)$ and $e_1(t_k)$ converge exponentially and $\xi(t_k)$ is bounded, we have from (21) that $\alpha_1(t_k)$ is recurrent, and the filtered value $\alpha_{1,1}^f(t_k)$ is also recurrent from Lemma 3. Then, from $z_2(t_k) = \hat{x}_2(t_k) - \alpha_{1,1}^f(t_k)$ and the convergence of $z_2(t_k)$, it is deduced that $\hat{x}_2(t_k)$ is recurrent. Recursively, the signals $\hat{x}_i(t_k)$ $(i = 3, \ldots, n)$ and $\alpha_{n-1,1}^f(t_k)$ are all recurrent. Thus, $Z(t_k) = [\hat{x}^{\mathrm{T}}(t_k), \frac{\alpha_{n-1,1}^f(t_{k+1}) - \alpha_{n-1,1}^f(t_k)}{T_s}]^{\mathrm{T}}$ is recurrent for $t_k > T_1$. Consequently, by dividing the vector $S(Z(t_k))$ into $S_\zeta(Z(t_k))$ and $S_{\bar{\zeta}}(Z(t_k))$, which contain the neurons near and far from the orbit of $Z(t_k)$, respectively, we have from Lemma 1 that for $t_k > T_1$, the PE condition of $S_\zeta(Z(t_k))$ is ensured.

From the analysis above, Eqs. (46) and (48) are separated as

$$\begin{cases} z_n(t_{k+1}) = \beta_1(t_k)z_n(t_k) + T_s g(\hat{x}(t_k)) \cdot (\tilde{\epsilon}_\zeta(Z(t_k)) - \tilde{W}_\zeta^{\mathrm{T}}(t_k)S_\zeta(Z(t_k))), \\ \tilde{W}_\zeta(t_{k+1}) = \tilde{W}_\zeta(t_k) + T_s\Gamma[S_\zeta(Z(t_k))z_n(t_{k+1}) - \sigma\hat{W}_\zeta(t_k)], \end{cases} \tag{70}$$

$$\tilde{W}_{\bar{\zeta}}(t_{k+1}) = \tilde{W}_{\bar{\zeta}}(t_k) + T_s\Gamma[S_{\bar{\zeta}}(Z(t_k))z_n(t_{k+1}) - \sigma\hat{W}_{\bar{\zeta}}(t_k)], \tag{71}$$

where $\tilde{\epsilon}_\zeta(Z(t_k)) = \tilde{\epsilon}(Z(t_k)) - \tilde{W}_{\bar{\zeta}}^{\mathrm{T}}(t_k)S_{\bar{\zeta}}(Z(t_k))$, $\tilde{W}_\zeta(t_k)$ and $\tilde{W}_{\bar{\zeta}}(t_k)$ represent the NN weight subvectors corresponding to $S_\zeta(Z(t_k))$ and $S_{\bar{\zeta}}(Z(t_k))$, respectively. It is evident that $S_{\bar{\zeta}}(Z(t_k))$ is small because of the localization property of RBFs. Moreover, since $\tilde{W}(t_k)$ is bounded according to Theorem 1, so $\tilde{W}_{\bar{\zeta}}^{\mathrm{T}}(t_k)S_{\bar{\zeta}}(Z(t_k))$ is small. Thus $\tilde{\epsilon}_\zeta(Z(t_k))$ is near to $\tilde{\epsilon}(Z(t_k))$. On the other hand, it is noticed that $g(\hat{x}(t_k))$ in (70) may enlarge the NN approximation error $\tilde{\epsilon}_\zeta(Z(t_k))$. To avoid this situation, we introduce a state transformation $\vartheta(t_k) = z_n(t_k)/\bar{g}$. Then, Eq. (70) becomes

$$\begin{bmatrix} \vartheta(t_{k+1}) \\ \tilde{W}_\zeta(t_{k+1}) \end{bmatrix} = \begin{bmatrix} \beta_1(t_k) & -T_s\frac{g(\hat{x}(t_k))}{\bar{g}}S_\zeta^{\mathrm{T}}(Z(t_k)) \\ \nu_1 & \nu_2 \end{bmatrix} \begin{bmatrix} \vartheta(t_k) \\ \tilde{W}_\zeta(t_k) \end{bmatrix} + \begin{bmatrix} T_s\frac{g(\hat{x}(t_k))}{\bar{g}}\tilde{\epsilon}_\zeta(Z(t_k)) \\ \nu_3 \end{bmatrix}, \tag{72}$$

where $\nu_1 = \beta_1(t_k)\bar{g}T_s\Gamma S_\zeta(Z(t_k))$, $\nu_2 = I - T_s^2 g(\hat{x}(t_k))\Gamma S_\zeta(Z(t_k))S_\zeta^{\mathrm{T}}(Z(t_k))$, $\nu_3 = T_s\Gamma(T_s g(\hat{x}(t_k)) \cdot S_\zeta(Z(t_k))\tilde{\epsilon}_\zeta(Z(t_k)) - \sigma\hat{W}_\zeta(t_k))$. According to Theorem 1 in [40], if Γ is chosen such that $0 < T_s^2\bar{g}\bar{\lambda}_\Gamma s^{*2} < \frac{1}{2}$, where $\bar{\lambda}_\Gamma$ is the maximum eigenvalue of Γ, and λ_n is designed such that $\beta_1(t_k) < \frac{(1-\varpi^2)^2}{2(1-\varpi^2)+1}$, with ϖ denoting the convergence speed of the unforced estimation error system of NN weights. Then, the nominal part of (72) will converge to the origin exponentially. Since $\tilde{\epsilon}_\zeta(Z(t_k))$ is bounded and $0 < \frac{g(\hat{x}(t_k))}{\bar{g}} < 1$, by choosing T_s small, we have that $T_s\frac{g(\hat{x}(t_k))}{\bar{g}}\tilde{\epsilon}_\zeta(Z(t_k))$ is small. Furthermore, since $g(\hat{x}(t_k))$, $S(Z(t_k))$, and $\hat{W}(t_k)$ are bounded, by choosing σ and T_s small enough, ν_3 can be made small. According to [55], $\vartheta(t_k)$ and $\tilde{W}_\zeta(t_k)$ can converge to small neighborhoods of the origin, with the neighbourhood bounds depending on the bounds of $T_s\frac{g(\hat{x}(t_k))}{\bar{g}}\tilde{\epsilon}_\zeta(Z(t_k))$ and ν_3, respectively. Since $S_{\bar{\zeta}}(Z(t_k))$ and σ are small, $\hat{W}(0) = 0$, and $z_n(t_k)$ converges exponentially to a small region around the origin, from (71), we have that $\tilde{W}_{\bar{\zeta}}(t_k)$ is slightly updated, and so is $\hat{W}_{\bar{\zeta}}(t_k)$. As a result, after the convergence of $\hat{W}_\zeta(t_k)$ to W_ζ^*, around the trajectory of $Z(t_k)$, the function $h(Z(t_k))$ can be approximated by

$$\begin{aligned} h(Z(t_k)) =& \hat{W}_\zeta^{\mathrm{T}}(t_k)S_\zeta(Z(t_k)) + \epsilon'(Z(t_k)) \\ =& \hat{W}^{\mathrm{T}}(t_k)S(Z(t_k)) + \epsilon''(Z(t_k)), \end{aligned} \tag{73}$$

where $\epsilon''(Z(t_k)) = \epsilon'(Z(t_k)) - \hat{W}_{\bar{\zeta}}^{\mathrm{T}}(t_k)S_{\bar{\zeta}}(Z(t_k))$, $|\epsilon'(Z(t_k))|$ and $|\epsilon''(Z(t_k))|$ are close to $|\epsilon(Z(t_k))|$ in (45). To reduce the impact of noises, we take the average value of $\hat{W}(t_k)$ as that in (69), and $h(Z(t_k))$ can be approximated by

$$h(Z(t_k)) = \bar{W}^{\mathrm{T}}S(Z(t_k)) + \epsilon'''(Z(t_k)), \tag{74}$$

where $|\epsilon'''(Z(t_k))|$ is close to $|\epsilon(Z(t_k))|$.

Therefore, within a local region around $Z(t_k)$, $\bar{W}^{\mathrm{T}}S(Z(t_k))$ can approximate $h(Z(t_k))$ accurately with a small error. The definition of the local region is given as

$$\begin{aligned} \Omega_Z := \{Z'(t_k) | & \mathrm{dist}(Z'(t_k), \psi_Z) < d^* \\ & \Rightarrow |\bar{W}^{\mathrm{T}}S(Z'(t_k)) - h(Z'(t_k))| < \eta^*\}, \end{aligned} \tag{75}$$

where ψ_Z denotes the the experienced orbit in the learning phase, $d^* > 0$ and $\eta^* > 0$ are constants, and η^* is close to ϵ^* within Ω_Z.

From the analysis above, it is deduced that for a trained NN $\bar{W}^{\mathrm{T}}S(Z(t_k))$, when an input $Z'(t_k)$ is within the region Ω_Z, $\bar{W}^{\mathrm{T}}S(Z'(t_k))$ can approximate $h(Z'(t_k))$ rapidly and accurately.

5 Knowledge-based controller design

It has been shown in Section 4 that the unknown sampled-data closed-loop dynamics $h(Z(t_k))$ is accurately modeled, and the knowledge can be acquired as $\bar{W}^{\mathrm{T}}S(Z(t_k))$. This section will demonstrate the achievement of knowledge-based control. With this objective, the following knowledge-based controller is designed

$$u(t_k) = -\lambda_n z_n(t_k) - \bar{W}^{\mathrm{T}}S(Z(t_k)). \tag{76}$$

Meanwhile, the aforementioned virtual controllers (those in (21), (30), and (39)) are employed in this section. With this knowledge-based controller, there is no need for online updating of the NN weights.

Theorem 3. Consider the closed-loop system consisting of the Euler discretization model (16), the sampled-data observer (5), the discrete-time command filter (8), and the control law (76). Then, for initial conditions satisfying (17) and in a local region of the experienced trajectory ψ_Z, (i) system (16) is stable with all the variables bounded; (ii) the tracking error e_1 with the PP inequality (17) converges exponentially to a small region around the origin.

Proof. Following the steps similar to those in Section 3, the error system is concluded as

$$\begin{cases} z_1(t_{k+1}) = (1 - T_s\lambda_1\varrho(t_k))z_1(t_k) + T_s\varrho(t_k)(z_2(t_k) + \iota(t_k)), \\ z_i(t_{k+1}) = (1 - T_s\lambda_i)z_i(t_k) + T_s\left(z_{i+1}(t_k) + \Delta\alpha_i(t_k)\right), \ i = 2, \ldots, n-1, \\ z_n(t_{k+1}) = \beta_1(t_k)z_n(t_k) + T_s g(\hat{x}(t_k))\tilde{h}(Z(t_k)), \end{cases} \tag{77}$$

where $\tilde{h}(Z(t_k)) = h(Z(t_k)) - \bar{W}^{\mathrm{T}} S(Z(t_k)) - \frac{1}{T_s g(\hat{x}(t_k))}(T_s \tilde{F}(t_k) + \tilde{e}_n(t_k) - \tilde{e}_n(t_{k+1}))$. Combining with (75) and the analysis in (46), it is deduced that $|\tilde{h}(Z(t_k))| < \tilde{h}^* := \eta^* + 2\ell\bar{F}/\underline{g} + \frac{l_1 L \ell T_s \bar{F}}{\underline{g}(1-\varepsilon)}$.

Consider the following candidate for the Lyapunov function:

$$V_{\tilde{z}}(t_k) = \sum_{i=1}^{n-1} z_i^2(t_k) + \frac{z_n^2(t_k)}{\bar{g}}. \tag{78}$$

From (77) and $\beta_1(t_k) = 1 - T_s \lambda_n g(\hat{x}(t_k))$, we have

$$\begin{aligned}\frac{z_n^2(t_{k+1}) - z_n^2(t_k)}{\bar{g}} =& -\frac{T_s \lambda_n g(\hat{x}(t_k))}{\bar{g}} \left(1 + \beta_1(t_k)\right) z_n^2(t_k) + \frac{T_s^2 g^2(\hat{x}(t_k))}{\bar{g}} \tilde{h}^2(Z(t_k)) \\ &+ \frac{2T_s \beta_1(t_k) g(\hat{x}(t_k)) z_n(t_k)}{\bar{g}} \tilde{h}(Z(t_k)).\end{aligned} \tag{79}$$

It is obvious that

$$\frac{2T_s \beta_1(t_k) g(\hat{x}(t_k)) z_n(t_k)}{\bar{g}} \tilde{h}(Z(t_k)) \leqslant \frac{T_s \beta_1(t_k) g(\hat{x}(t_k))}{\bar{g}} \cdot \left(\lambda_n z_n^2(t_k) + \frac{\tilde{h}^2(Z(t_k))}{\lambda_n}\right). \tag{80}$$

Then, by substituting (80) into (79), and combining with $|\tilde{h}(Z(t_k))| < \tilde{h}^*$, it is deduced that

$$\frac{z_n^2(t_{k+1}) - z_n^2(t_k)}{\bar{g}} \leqslant \frac{T_s g(\hat{x}(t_k))}{\bar{g}} \left(-\lambda_n z_n^2(t_k) + \frac{\tilde{h}^{*2}}{\lambda_n}\right). \tag{81}$$

From (28), (37), (42), and (81), it is deduced that

$$\begin{aligned}\Delta V_{\tilde{z}}(t_k) =& \sum_{i=1}^{n-1} \Delta V_i(t_k) + \frac{z_n^2(t_{k+1}) - z_n^2(t_k)}{\bar{g}} \\ =& -T_s^2 \lambda_1^2 \varrho^2(t_k) z_1^2(t_k) - \left(T_s^2 \lambda_2^2 - \frac{2T_s \varrho(t_k)}{\lambda_1}\right) z_2^2(t_k) - \sum_{i=3}^{n-1} \left(T_s^2 \lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}\right) z_i^2(t_k) \\ &- \left(\frac{T_s \lambda_n g(\hat{x}(t_k))}{\bar{g}} - \frac{2T_s}{\lambda_{n-1}}\right) z_n^2(t_k) + \bar{\mu} + \frac{T_s g(\hat{x}(t_k))}{\lambda_n \bar{g}} \tilde{h}^{*2} \\ \leqslant& -T_s^2 \lambda_1^2 \varrho^2(t_k) z_1^2(t_k) - \left(T_s^2 \lambda_2^2 - \frac{2T_s \varrho(t_k)}{\lambda_1}\right) z_2^2(t_k) \\ &- \sum_{i=3}^{n-1} \left(T_s^2 \lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}\right) z_i^2(t_k) - \left(\frac{T_s \lambda_n \underline{g}}{\bar{g}} - \frac{2T_s}{\lambda_{n-1}}\right) z_n^2(t_k) + \theta_4,\end{aligned} \tag{82}$$

where $\theta_4 = \bar{\mu} + \frac{T_s}{\lambda_n} \tilde{h}^{*2}$. If the parameters are chosen such that

$$\begin{cases} T_s^2 \lambda_2^2 - \dfrac{2T_s \varrho(t_k)}{\lambda_1} > 0, \\ T_s^2 \lambda_i^2 - \dfrac{2T_s}{\lambda_{i-1}} > 0,\ i = 3, \ldots, n-1, \\ \dfrac{T_s \lambda_n \underline{g}}{\bar{g}} - \dfrac{2T_s}{\lambda_{n-1}} > 0, \end{cases} \tag{83}$$

then, when any of the following conditions holds:

$$\begin{aligned} &|z_1(t_k)| > \sqrt{\theta_4} / \left(T_s \lambda_1 \varrho(t_k)\right), \\ &|z_2(t_k)| > \sqrt{\theta_4 / \left(T_s^2 \lambda_2^2 - 2T_s \varrho(t_k)/\lambda_1\right)}, \\ &|z_i(t_k)| > \sqrt{\theta_4 / \left(T_s^2 \lambda_i^2 - 2T_s/\lambda_{i-1}\right)},\ i = 3, \ldots, n-1, \end{aligned}$$

$$|z_n(t_k)| > \sqrt{\theta_4/\left(T_s\lambda_n\underline{g}/\bar{g} - 2T_s/\lambda_{n-1}\right)}, \tag{84}$$

we have that $\Delta V_{\bar{z}}(t_k) < 0$ from (82). So the signals $z_i(t_k)$ $(i = 1, \ldots, n)$ are bounded based on Lyapunov's theorem. Since $e_1(t_k)$ is bounded from similar analysis in Theorem 1, by combining with the boundedness of the RBF vector $S(Z(t_k))$, the constant NN weight vector $\bar{W}$, the reference signal $y^d(t_k)$, and the PP function $\xi(t_k)$, we can conclude that $x_i(t_k)$ $(i = 1, \ldots, n)$ and the knowledge-based controller $u(t_k)$ are bounded. Furthermore, Eq. (82) is simplified as

$$V_{\bar{z}}(t_{k+1}) - V_{\bar{z}}(t_k) \leqslant -\theta_3 V_{\bar{z}}(t_k) + \bar{\theta}_4, \tag{85}$$

where $\theta_3 = \min\{T_s^2\lambda_1^2\varrho^2(t_k),\ T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1},\ T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}}(i = 3, \ldots, n-1),\ T_s\lambda_n\underline{g}/\bar{g} - \frac{2T_s}{\lambda_{n-1}}\}$, and $\bar{\theta}_4 > 0$ is the upper bound of $|\theta_4|$. Then, we have from (85) that

$$V_{\bar{z}}(t_{k+1}) \leqslant (1 - \theta_3)V_{\bar{z}}(t_k) + \bar{\theta}_4. \tag{86}$$

From the analysis of (68), $0 < T_s^2\lambda_1^2\varrho^2(t_k) < 1$, $0 < T_s^2\lambda_2^2 - \frac{2T_s\varrho(t_k)}{\lambda_1} < 1$, and $0 < T_s^2\lambda_i^2 - \frac{2T_s}{\lambda_{i-1}} < 1$ $(i = 3, \ldots, n-1)$. From (83), $0 < T_s\lambda_n\underline{g}/\bar{g} - \frac{2T_s}{\lambda_{n-1}}$. If the parameters are chosen such that $0 < T_s\lambda_n - \frac{2T_s}{\lambda_{n-1}} < 1$, then $0 < T_s\lambda_n\underline{g}/\bar{g} - \frac{2T_s}{\lambda_{n-1}} < T_s\lambda_n - \frac{2T_s}{\lambda_{n-1}} < 1$. Thus, it can be concluded that $0 < \theta_3 < 1$. From (86), we have $V_{\bar{z}}(t_k) \leqslant (1 - \theta_3)^k V_{\bar{z}}(0) + \frac{\bar{\theta}_4(1-(1-\theta_3)^k)}{1-(1-\theta_3)} < (1 - \theta_3)^k V_{\bar{z}}(0) + \frac{\bar{\theta}_4}{\theta_3}$. So, for any $c_2 > \frac{\bar{\theta}_4}{\theta_3}$, we have $z_i(t_k) < \sqrt{c_2}$, $\forall t_k > T_2$ $(i = 1, \ldots, n)$, with T_2 being a finite time instant. Since c_2 is arbitrarily small when the parameters are chosen appropriately, $z_i(t_k)$ $(i \in \{1, \ldots, n\})$ converges to a small region around the origin exponentially. As with the analysis in Theorem 1, $e_1(t_k)$ will converge to a small region around the origin exponentially.

Remark 3. Note that the knowledge-based controller (76) is embedded with the learned constant NN $\bar{W}^{\mathrm{T}}S(Z(t_k))$, and the weight updating law (47) in the previous adaptive process is no longer needed. Thus, the computational load caused by online adaptive adjustment of NN weights is eliminated. Furthermore, since an accurate NN approximation of the unknown sampled-data closed-loop dynamics $h(Z(t_k))$ is obtained by $\bar{W}^{\mathrm{T}}S(Z(t_k))$, this constant NN can recall the learned knowledge and provide an accurate approximation of $h(Z(t_k))$ rapidly within a local region Ω_Z around $Z(t_k)$. Therefore, the enhancement of control performance is achieved with quicker response time and reduced computational costs.

Remark 4. From Subsection 3.1, we know that in order to make $e_1(t)$ adhere to the PP, it is sufficient to ensure that $z_1(t)$ is bounded. According to [54], in the case of forward difference, once the corresponding discrete-time variable is stable, the continuous-time variable remains in a stable region. Since the discrete-time controllers designed in this paper can achieve tracking control of the Euler approximation nonlinear sampled-data system (16), from the boundedness of the discrete-time variable $z_1(t_k)$, we can obtain the boundedness of the continuous-time variable $z_1(t)$ [33], which guarantees the PP of $e_1(t)$.

6 Simulation studies

In this section, a pendulum system in [56] is chosen as the controlled system with PP. The pendulum system is as follows:

$$\begin{cases} \dot{\theta}(t) = \omega(t), \\ \dot{\omega}(t) = \dfrac{3}{4ml^2}\left(u(t) + mgl\sin\left(\theta(t)\right)\right), \end{cases} \tag{87}$$

where $\theta(t)$ represents the angular position, $\omega(t)$ denotes the velocity, $u(t)$ represents the control signal and also the angular acceleration, m and l are the mass and length, and g is the gravitational acceleration. The values of the physical parameters are chosen as those in [56]: $m = 1/3$, $l = 3/2$, $g = 0.98$ m/s^2.

From (16), the Euler discretization of the transformed pendulum system is as follows:

$$\begin{cases} z_1(t_{k+1}) = z_1(t_k) + \varrho(t_k)\left(e_1(t_{k+1}) - \dfrac{e_1(t_k)\xi(t_{k+1})}{\xi(t_k)}\right), \\ x_2(t_{k+1}) = x_2(t_k) + \dfrac{3T_s}{4ml^2}\left(u(t_k) + mgl\sin\left(x_1(t_k)\right)\right), \end{cases} \tag{88}$$

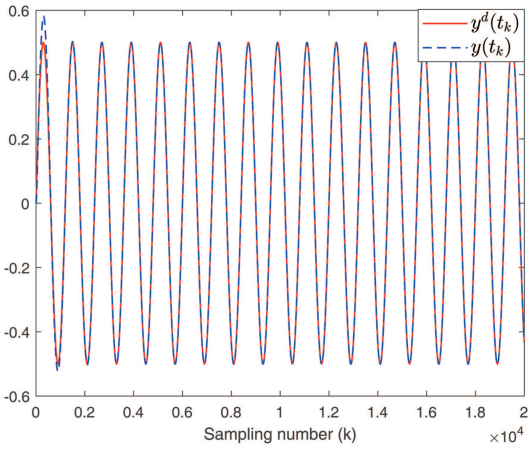

Figure 1 (Color online) Tracking of $y(t_k)$ to $y^d(t_k)$.

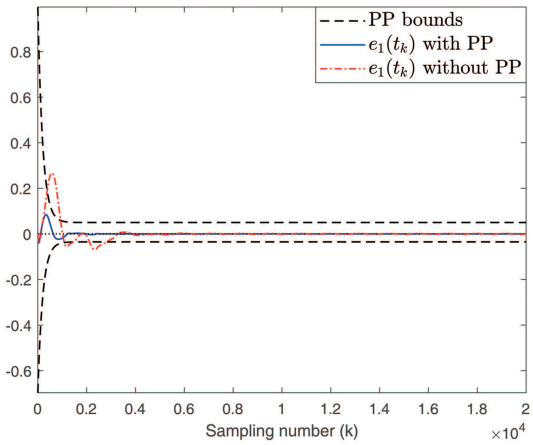

Figure 2 (Color online) Convergence of the tracking error $e_1(t_k)$ with PP in this paper and without PP in [41].

where $e_1(t_k) = x_1(t_k) - y^d(t_k)$, $x_1(t_k) = \theta(t_k)$, $x_2(t_k) = \omega(t_k)$, $y^d(t_k)$ is the reference signal and is produced by $y^d(t_k) = 0.5\sin(\pi t_k/3)$, and the sampling period $T_s = 0.005$. The functions in (16) satisfy $f(t_k) = \frac{3g}{4l}\sin(x_1(t_k))$ and $g(t_k) = \frac{3}{4ml^2}$. The initial values of $x_1(0)$ and $x_2(0)$ are set to 0, which indicates that the pendulum is in the upward position with an angular velocity of 0.

For the PP inequality in (17), we take $\rho_1 = 0.7$, $\rho_2 = 1$, and $\xi(t_k)$ satisfies $\xi(t_{k+1}) = (\xi(0) - \xi_\infty)\mathrm{e}^{-\varphi t_k} + \xi_\infty$ with $\xi(0) = 1$, $\varphi = 1$, and $\xi_\infty = 0.05$.

Since $x_2(t_k)$ is unavailable, the sampled-data observer (5) is employed as follows:

$$\begin{cases} \hat{x}_1(t_{k+1}) = \hat{x}_1(t_k) + T_s\hat{x}_2(t_k) + r_1(\hat{y}(t_k) - y(t_k)), \\ \hat{x}_2(t_{k+1}) = \hat{x}_2(t_k) + \dfrac{r_2}{\varepsilon}(\hat{y}(t_k) - y(t_k)), \\ \hat{y}(t_k) = \hat{x}_1(t_k), \end{cases} \tag{89}$$

where the parameters are chosen as $r_1 = -2$, $r_2 = -8$, $\varepsilon = 1/45$. For the initial values, we set $\hat{x}_1(0)$ and $\hat{x}_2(0)$ to zero.

6.1 Control with PP

In this subsection, the controller in (44) with the neural adaptive law in (47) is employed as

$$u(t_k) = -\lambda_2 z_2(t_k) - \hat{W}^{\mathrm{T}}(t_k)S(Z(t_k)), \tag{90}$$

$$\hat{W}(t_{k+1}) = \hat{W}(t_k) + T_s\Gamma[S(Z(t_k))z_2(t_{k+1}) - \sigma\hat{W}(t_k)], \tag{91}$$

where $Z(t_k) = [\hat{x}_1(t_k), \hat{x}_2(t_k), \frac{\alpha^f_{1,1}(t_k) - \alpha^f_{1,1}(t_{k+1})}{T_s}]^{\mathrm{T}}$, $\alpha^f_{1,1}(t_k)$ is the command filter output in (8). For the command filter, we choose $\omega_n = 110$, $\zeta_1 = 0.3$, the input $\alpha_1(t_k)$ is given in (21), and the values of $\alpha^f_{1,1}(0)$ and $\alpha^f_{1,2}(0)$ are set to $\alpha_1(0)$ and 0, respectively. The parameters of the virtual controller $\alpha_1(t_k)$ and controller $u(t_k)$ are chosen as $\lambda_1 = 0.3$, $\lambda_2 = 10$, $\Gamma = 10$, and $\sigma = 0.0001$. For the NN $\hat{W}^{\mathrm{T}}(t_k)S(Z(t_k))$, there are 245 neurons evenly spaced on $[-0.6, 0.6] \times [-0.6, 0.6] \times [-1, 1]$, with widths 0.2, 0.2, and 0.5, respectively, and the NN weights are initialized as $\hat{W}(0) = 0$.

Remark 5. The PE condition plays an important role in nonlinear identification and adaptive control. As shown in [57], when using the RBF NN, the PE condition of the NN is affected by the speed of the NN input trajectory and the network structure. In general, the higher the speed of the input signal, the lower the PE level, and vice versa. On the other hand, given a dynamical trajectory, the PE level will increase when the NN separation distance increases, and decreases when the NN separation distance decreases. However, there is a tradeoff between the higher PE level and the better approximation capability of the NN. Therefore, an appropriate network structure should be selected to guarantee a relatively high PE level and good NN approximation ability.

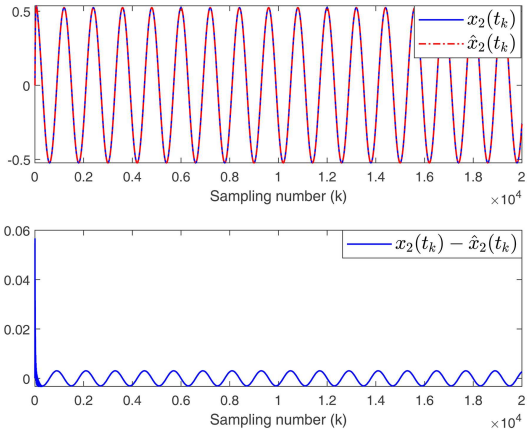

Figure 3 (Color online) Estimation results of the state $x_2(t_k)$ by $\hat{x}_2(t_k)$.

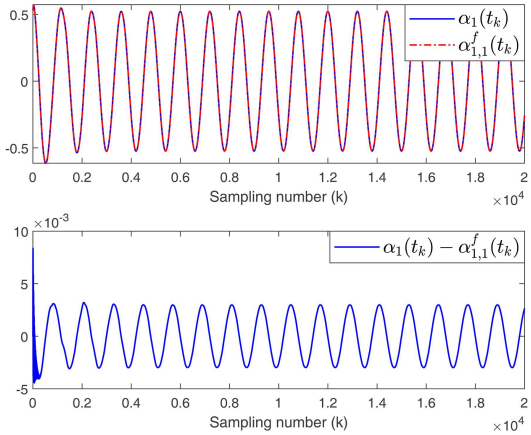

Figure 4 (Color online) Virtual control $\alpha_1(t_k)$ and the output of the command filter $\alpha^f_{1,1}(t_k)$.

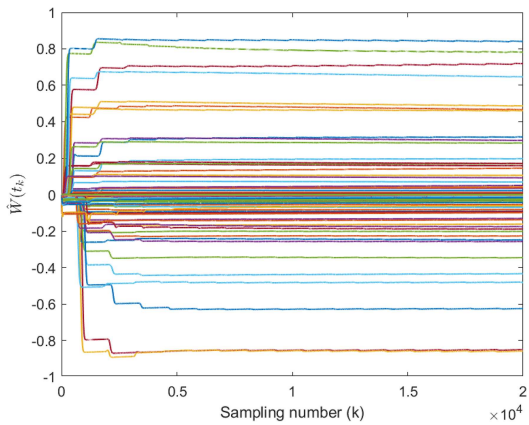

Figure 5 (Color online) Partial convergence of the NN weights $\hat{W}(t_k)$.

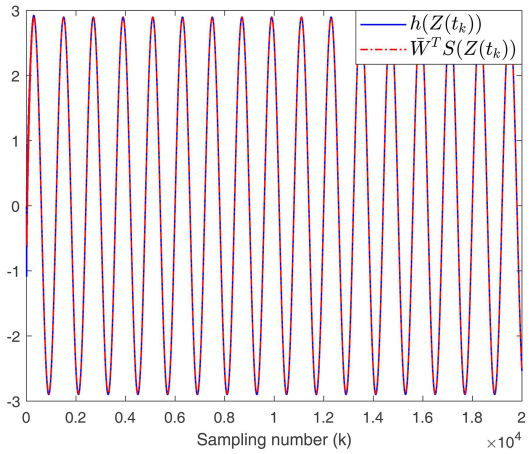

Figure 6 (Color online) Approximation of $h(Z(t_k))$ using the constant NN $\bar{W}^{\mathrm{T}} S(Z(t_k))$.

Figures 1–6 depict the results. As presented in Figure 1, the tracking performance of $y(t_k)$ to the desired signal $y^d(t_k)$ is satisfactory. Figure 2 shows that under the controller (90), $e_1(t_k)$ converges to zero with an overshoot less than 0.1, a convergence speed faster than that of the performance function $\xi(t_k)$, and a steady-state within $[-0.035, 0.05]$. Figure 3 depicts the estimation results of $x_2(t_k)$ by the observer (89), and well estimation is achieved. Figure 4 shows the command filter output $\alpha^f_{1,1}(t_k)$ can provide an accurate estimation of the virtual control law $\alpha_1(t_k)$. After the convergence of $e_1(t_k)$, $x_1(t_k)$ tracks the recurrent orbit of $y^d(t_k)$, and all the signals are recurrent according to the analysis in Theorem 2. So, the partial PE condition is satisfied, and Figure 5 depicts the partial convergence of the NN weights $\hat{W}(t_k)$. After the convergence of $\hat{W}(t_k)$, the unknown sampled-data closed-loop dynamics $h(Z(t_k)) = \frac{1}{g(\hat{x}(t_k))}(f(\hat{x}(t_k)) + \frac{\alpha^f_{1,1}(t_k) - \alpha^f_{1,1}(t_{k+1})}{T_s})$ can be approximated by $\bar{W}^{\mathrm{T}} S(Z(t_k))$, as depicted in Figure 6.

Remark 6. In the simulation, $y^d(t_k) = 0.5\sin(\pi t_k/3)$ is chosen as the recurrent reference signal, and the pendulum system in [56] is chosen as the controlled system. To achieve tracking control with PP, the adaptive controller $u(t_k) = -\lambda_2 z_2(t_k) - \hat{W}^{\mathrm{T}}(t_k) S(Z(t_k))$ with the NN weight updating law $\hat{W}(t_{k+1}) = \hat{W}(t_k) + T_s \Gamma[S(Z(t_k)) z_2(t_{k+1}) - \sigma \hat{W}(t_k)]$ is constructed, where $Z(t_k) = [\hat{x}_1(t_k), \hat{x}_2(t_k), \frac{\alpha^f_{1,1}(t_k) - \alpha^f_{1,1}(t_{k+1})}{T_s}]^{\mathrm{T}}$ is the NN input vector, $\hat{x}_1(t_k)$ and $\hat{x}_2(t_k)$ are the observed values of the states $x_1(t_k)$ and $x_2(t_k)$, respectively, and $\alpha^f_{1,1}(t_k)$ is the filtered value of the virtual control law. It is proven in Theorem 1 that with the adaptive controller $u(t_k)$, when the initial conditions satisfy the PP, the tracking error $e_1(t_k) = y(t_k) - y^d(t_k)$ will converge exponentially with PP, and the results are shown in Figure 2. Since

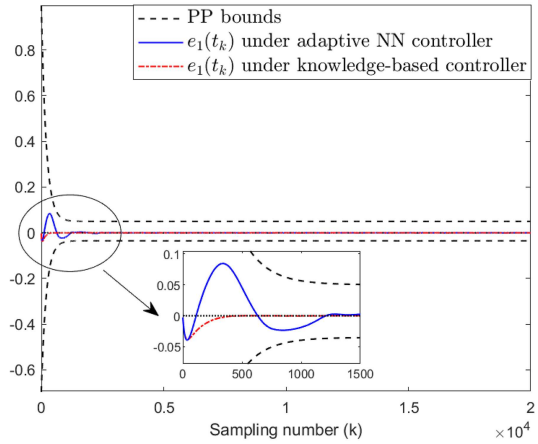

Figure 7 (Color online) Convergence of the tracking error $e_1(t_k)$ under the adaptive NN and knowledge-based controllers with PP.

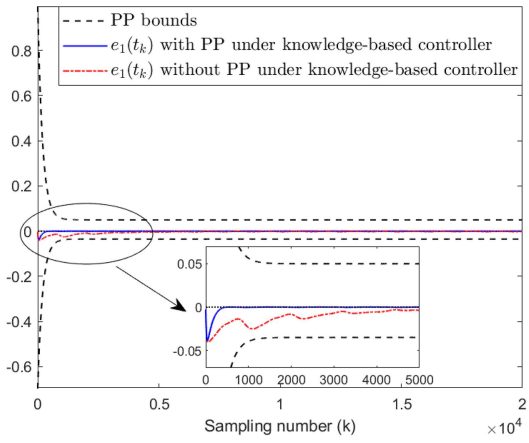

Figure 8 (Color online) Convergence of the tracking error $e_1(t_k)$ under knowledge-based controllers with PP in this paper and without PP in [41].

$y^d(t_k)$ is recurrent, after $e_1(t_k)$ converges, the output $y(t_k)$ of the system is also recurrent, which can be seen in Figure 1. At the same time, as analyzed in Theorem 2, the signals $\hat{x}_1(t_k)$, $\hat{x}_2(t_k)$, and $\alpha_{1,1}^f(t_k)$ are all recurrent, and the simulation results are shown in Figures 3 and 4, where $\hat{x}_1(t_k)$ is omitted due to the limitation of space. Then, along the recurrent NN input signals, the partial PE condition of the RBF NN can be established, and partial convergence of the NN weights is achieved, which can be seen in Figure 5. Moreover, as shown in Figure 6, after the weights converge, accurate modeling of the unknown function is achieved by the constant NN along the recurrent input signals.

6.2 Knowledge-based control with PP

In this subsection, the controller in (76) is employed as

$$u(t_k) = -\lambda_2 z_2(t_k) - \bar{W}^{\mathrm{T}} S(Z(t_k)), \tag{92}$$

where $\bar{W}$ is obtained from (69), and $S(Z(t_k))$ is the same as that in Subsection 6.1. The other parameters and initial values are set the same as those in Subsection 6.1. The simulation results depicted in Figure 7 demonstrate that by using the knowledge-based controller (92), better control performances such as quicker convergence speed and reduced overshoot are achieved with the PP satisfied.

Finally, the method in this paper is compared with the method proposed in [41] (which is designed without considering PP) to showcase its effectiveness. The parameter values and initial conditions for both methods are selected to be identical, and the NN configuration is also the same. Figures 2 and 8 present the simulation results. From Figure 2, when using an adaptive neural controller designed with PP, the tracking error $e_1(t_k)$ consistently adheres to the prescribed bounds in (17), while under the method proposed without PP in [41], $e_1(t_k)$ does not adhere to the desired performance bounds. When using the knowledge-based controller, from Figure 8, we can see that the method with PP in this paper demonstrates an advancement in tracking performance over the approach without PP presented in [41].

Remark 7. The purpose of this paper is to realize output-feedback tracking control with PP for a class of nonlinear sampled-data systems. The proposed two-stage scheme includes the phases of learning from closed-loop control with PP and knowledge-based control with PP. In the learning phase, an adaptive NN controller is constructed to realize stable control and achieve convergence of the tracking error with PP. The results can be seen in Figures 1–4. Based on the convergence of signals, the partial PE condition of the RBF NN can be established, which ensures the partial convergence of the NN weights, as shown in Figure 5. After the NN weights converge, an accurate approximation of $h(Z(t_k))$ can be achieved by the constant NN $\bar{W}^{\mathrm{T}} S(Z(t_k))$, as shown in Figure 6. In the knowledge-based control phase, a constant NN controller is constructed. It can be seen from Figures 7 and 8 that the stability of the system and the tracking control with PP can also be realized. Furthermore, better transient performance is achieved since no online adaptation is needed.

7 Conclusion

This study introduces a DL-based output-feedback control scheme for a class of nonlinear sampled-data systems with PP. For the Euler discretization model of the transformed system dynamics, a discrete neural output-feedback controller based on a sampled-data observer and a discrete-time command filter is proposed using the backstepping technique. System stability and the output tracking performance are ensured according to Lyapunov's theorem. Moreover, an accurate NN approximation of the unknown sampled-data closed-loop dynamics can be obtained according to the sampled-data DL theory and exponential stability of a DLTV system. After the learning process, knowledge can be acquired in the form of a constant NN that is reused in the construction process of a knowledge-based controller. By using the knowledge-based controller, an improved control performance is attained, and the PP of the tracking error is also satisfied in both the transient and steady states. In the future, we plan to investigate various control environments including faults and other circumstances.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant Nos. 62203262, 62350083) and Program of Shandong Provincial Natural Science Foundation (Grant No. ZR2022QF124).

References

1 Krstić M, Kanellakopoulos I, Kokotović P V. Nonlinear and Adaptive Control Design. Boca Raton: CRC Press, 2006
2 Wang L X, Mendel J M. Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. IEEE Trans Neural Netw, 1992, 3: 807–814
3 Sanner R M, Slotine J J E. Gaussian networks for direct adaptive control. IEEE Trans Neural Netw, 1992, 3: 837–863
4 Ge S S, Wang C. Direct adaptive NN control of a class of nonlinear systems. IEEE Trans Neural Netw, 2002, 13: 214–221
5 Lin-Kwong-Chon C, Grondin-Pérez B, Kadjo J J A, et al. A review of adaptive neural control applied to proton exchange membrane fuel cell systems. Annu Rev Control, 2019, 47: 133–154
6 Peng Z, Wang D, Chen Z, et al. Adaptive dynamic surface control for formations of autonomous surface vehicles with uncertain dynamics. IEEE Trans Contr Syst Technol, 2013, 21: 513–520
7 Yang C, Chen C, He W, et al. Robot learning system based on adaptive neural control and dynamic movement primitives. IEEE Trans Neural Netw Learn Syst, 2019, 30: 777–787
8 Sarangapani J. Neural Network Control of Nonlinear Discrete-Time Systems. Boca Raton: CRC Press, 2006
9 Liu Y-J, Tong S-C, Wang D, et al. Adaptive neural output feedback controller design with reduced-order observer for a class of uncertain nonlinear SISO systems. IEEE Trans Neural Netw, 2011, 22: 1328–1334
10 Xu B, Gao D X, Wang S X. Adaptive neural control based on HGO for hypersonic flight vehicles. Sci China Inf Sci, 2011, 54: 511–520
11 Wang T, Li Y. Neural-network adaptive output-feedback saturation control for uncertain active suspension systems. IEEE Trans Cybern, 2022, 52: 1881–1890
12 Ge S S, Yang C G, Lee T H. Adaptive predictive control using neural network for a class of pure-feedback systems in discrete time. IEEE Trans Neural Netw, 2008, 19: 1599–1614
13 Liu Y J, Tang L, Tong S, et al. Adaptive NN controller design for a class of nonlinear MIMO discrete-time systems. IEEE Trans Neural Netw Learn Syst, 2015, 26: 1007–1018
14 Wang M, Wang K, Huang L, et al. Observer-based event-triggered tracking control for discrete-time nonlinear systems using adaptive critic design. IEEE Trans Syst Man Cybern Syst, 2023, 53: 5393–5403
15 Zhang P, Yuan Y, Guo L. Fault-tolerant optimal control for discrete-time nonlinear system subjected to input saturation: a dynamic event-triggered approach. IEEE Trans Cybern, 2021, 51: 2956–2968
16 Wang X H, Wang Z, Xia J W, et al. Adaptive event-trigger-based sampled-data stabilization of complex-valued neural networks: a real and complex LMI approach. Sci China Inf Sci, 2023, 66: 149203
17 Bu X. Prescribed performance control approaches, applications and challenges: a comprehensive survey. Asian J Control, 2023, 25: 241–261
18 Huang H, He W, Li J, et al. Disturbance observer-based fault-tolerant control for robotic systems with guaranteed prescribed performance. IEEE Trans Cybern, 2022, 52: 772–783
19 Sun Y, Liu J, Gao Y, et al. Adaptive neural tracking control for manipulators with prescribed performance under input saturation. IEEE ASME Trans Mechatron, 2023, 28: 1037–1046
20 Bechlioulis C P, Rovithakis G A. Robust adaptive control of feedback linearizable MIMO nonlinear systems with prescribed performance. IEEE Trans Automat Contr, 2008, 53: 2090–2099
21 Wang M, Wang C, Shi P, et al. Dynamic learning from neural control for strict-feedback systems with guaranteed predefined performance. IEEE Trans Neural Netw Learn Syst, 2016, 27: 2564–2576
22 Fan Q-Y, Xu S H, Xu B, et al. Simplified prescribed performance tracking control of uncertain nonlinear systems. Sci China Inf Sci, 2022, 65: 189204
23 Li Z, Yue D, Ma Y, et al. Neural-networks-based prescribed tracking for nonaffine switched nonlinear time-delay systems. IEEE Trans Cybern, 2022, 52: 6579–6590
24 Wang X, Wang Q, Sun C. Prescribed performance fault-tolerant control for uncertain nonlinear MIMO system using actor-critic learning structure. IEEE Trans Neural Netw Learn Syst, 2022, 33: 4479–4490
25 Zhang F K, Wu W M, Wang C. Pattern-based learning and control of nonlinear pure-feedback systems with prescribed performance. Sci China Inf Sci, 2023, 66: 112202
26 He S D, Dai S L, Dong C. Adaptive synchronization control of uncertain multiple USVs with prescribed performance and preserved connectivity. Sci China Inf Sci, 2022, 65: 199201
27 Wang X, Niu B, Wang H, et al. Prescribed performance-based finite-time consensus technology of nonlinear multiagent systems and application to FDPs. IEEE Trans Circuits Syst II, 2023, 70: 591–595
28 Zheng Z, Feroskhan M. Path following of a surface vessel with prescribed performance in the presence of input saturation and external disturbances. IEEE ASME Trans Mechatron, 2017, 22: 2564–2575

29 Liu D, Yang G H. Prescribed performance model-free adaptive integral sliding mode control for discrete-time nonlinear systems. IEEE Trans Neural Netw Learn Syst, 2019, 30: 2222–2230
30 Nguyen M L, Chen X, Yang F. Discrete-time quasi-sliding-mode control with prescribed performance function and its application to piezo-actuated positioning systems. IEEE Trans Ind Electron, 2018, 65: 942–950
31 Treesatayapun C. Prescribed performance of discrete-time controller based on the dynamic equivalent data model. Appl Math Model, 2020, 78: 366–382
32 Hou M, Wang Y. Data-driven adaptive terminal sliding mode control with prescribed performance. Asian J Control, 2021, 23: 774–785
33 Shao S, Chen M. Adaptive neural discrete-time fractional-order control for a UAV system with prescribed performance using disturbance observer. IEEE Trans Syst Man Cybern Syst, 2021, 51: 742–754
34 Antsaklis P J. Intelligent learning control. IEEE Control Syst, 1995, 15: 5–7
35 Wang C, Hill D J. Learning from neural control. IEEE Trans Neural Netw, 2006, 17: 130–146
36 Wang C, Hill D J. Deterministic Learning Theory for Identification, Recognition, and Control. Boca Raton: CRC Press, 2009
37 Yang F, Wang C. Pattern-based NN control of a class of uncertain nonlinear systems. IEEE Trans Neural Netw Learn Syst, 2018, 29: 1108–1119
38 Wang M, Wang C, Liu X. Dynamic learning from adaptive neural control with predefined performance for a class of nonlinear systems. Inf Sci, 2014, 279: 874–888
39 Yuan C, Licht S, He H. Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics. IEEE Trans Cybern, 2018, 48: 2920–2934
40 Wu W M, Wang C, Yuan C Z. Deterministic learning from sampling data. Neurocomputing, 2019, 358: 456–466
41 Zhang F K, Wu W M, Hu J T, et al. Deterministic learning from neural control for a class of sampled-data nonlinear systems. Inf Sci, 2022, 595: 159–178
42 Zhang F K, Wu W M, Wang C. Dynamic learning from neural network-based control for sampled-data strict-feedback nonlinear systems. Intl J Robust Nonlinear, 2022, 32: 8397–8420
43 Zhou Z, Yu J, Yu H, et al. Neural network-based discrete-time command filtered adaptive position tracking control for induction motors via backstepping. Neurocomputing, 2017, 260: 203–210
44 Huang D Y, Yang C G, Pan Y P, et al. Composite learning enhanced neural control for robot manipulator with output error constraints. IEEE Trans Ind Inf, 2021, 17: 209–218
45 Shao S Y, Chen M, Hou J, et al. Event-triggered-based discrete-time neural control for a quadrotor UAV using disturbance observer. IEEE ASME Trans Mechatron, 2021, 26: 689–699
46 Song J, Wang Y K, Zheng W X, et al. Adaptive terminal sliding mode speed regulation for PMSM under neural-network-based disturbance estimation: a dynamic-event-triggered approach. IEEE Trans Ind Electron, 2023, 70: 8446–8456
47 Park J, Sandberg I W. Universal approximation using radial-basis-function networks. Neural Computation, 1991, 3: 246–257
48 Schilling R J, Carroll J J, Al-Ajlouni A F. Approximation of nonlinear systems with radial basis function neural networks. IEEE Trans Neural Netw, 2001, 12: 1–15
49 Powell M J D. The Theory of Radial Basis Function Approximation in 1990. Oxford: Oxford University Press, 1992
50 Wang C, Hill D J. Deterministic learning and rapid dynamical pattern recognition. IEEE Trans Neural Netw, 2007, 18: 617–630
51 Narendra K S, Annaswamy A M. Stable Adaptive Systems. Upper Saddle River: Prentice Hall, 1989
52 Hu J T, Wu W M, Zhang F K, et al. Observer-based dynamical pattern recognition via deterministic learning. Neural Networks, 2023, 159: 161–174
53 Liu D, Wang D, Wang F Y, et al. Neural-network-based online HJB solution for optimal robust guaranteed cost control of continuous-time uncertain nonlinear systems. IEEE Trans Cybern, 2014, 44: 2834–2847
54 Franklin G F, Powell J D, Workman M L. Digital Control of Dynamic Systems. 3rd ed. Half Moon Bay: Ellis-Kagle Press, 1998
55 Weiss A, Mitra D. Digital adaptive filters: conditions for convergence, rates of convergence, effects of noise and errors arising from the implementation. IEEE Trans Inform Theor, 1979, 25: 637–652
56 Yang Q M, Jagannathan S. Reinforcement learning controller design for affine nonlinear discrete-time systems using online approximators. IEEE Trans Syst Man Cybern B, 2012, 42: 377–390
57 Zheng T J, Wang C. Relationship between persistent excitation levels and RBF network structures, with application to performance analysis of deterministic learning. IEEE Trans Cybern, 2017, 47: 3380–3392