Some new results on the finite-time control and its application to a chemical reactor system

Ziteng Guo¹, Caisheng Wei²*

¹School of Textile and Material Engineering, Dalian Polytechnic University
²School of Astronautics, Northwestern Polytechnical University
*corresponding author’s e-mail: 1097853117@qq.com, yundiqiuyu@163.com

Abstract. In this paper, a novel finite-time control scheme is proposed for a class of non-strict feedback systems with guaranteed preassigned output tracking performance. First, a finite-time convergent performance function is proposed in the prescribed performance control structure. Then, based on the proposed performance function, a finite-time stable controller is devised. Compared with the existing finite-time control schemes, the fractional state or output information and discontinuous phenomenon is avoided totally. Finally, application to a chemical reactor system is organized to validate the effectiveness of the proposed control scheme.

1. Introduction
The past few decades have witnessed the fast development of finite-time control theory due to its widely potential applications in real systems like robotic system, chemical reaction system and etc (e.g., see [1]-[3] and references therein). The main way to achieve finite-time stability is via sliding mode control (SMC) based technique. Namely, fractional power state or output information and symbolic functions are widely used to construct the relevant control schemes. Although effective, the usage of fractional power state or output information makes the relevant controller pretty complex due to the highly computational burden. Moreover, using symbolic function used will make the controller discontinuous, which is not easily achievable in real systems. To conquer the foregoing two inherent limitations, SMC-based technique should be avoided.

In recent years, prescribed performance control (PPC) has gained considerable attention due to its prominent advantage in quantitatively characterizing the transient and steady-state performance of the controlled systems (e.g., see [4]-[5] and reference therein). In the existing works, the performance function used in the PPC structure is in an exponential form, which means the controlled system will converge to its equilibrium point exponentially. However, if the performance function is finite-time convergent, the relevant controlled system will be finite-time stable. By following this idea, in this paper, we first propose a novel finite-time convergent performance function. Then, based on the newly proposed performance function, finite-time convergent controller is devised in the PPC structure. Compared with the SMC-based finite-time control schemes, the foregoing inherent limitations will be avoided totally.

The rest of this paper is organized as follows. In Section 2, the problem formulation is stated. Section 3 shows the finite-time controller design with its stability analysis. In Section 4, application to a chemical reactor system is organized to validate the effectiveness of the proposed control scheme. Some conclusions are drawn in Section 5.
2. Problem formulation
The nonstrict feedback system considered in this paper is expressed by

\[
\begin{align*}
\dot{x}_i &= x_2 + f_i(x) \\
&\vdots \\
\dot{x}_n &= bu + f_n(x) + d \\
y &= x_1
\end{align*}
\]  

(1)

where \( x = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^n \), \( y \in \mathbb{R} \) are the system state vector and output, respectively. \( f_i(x) \in \mathbb{R} \) is a nonlinear function, which is unknown. \( u, d \in \mathbb{R} \) denote the control input and unknown bounded external disturbance, respectively. \( b \neq 0 \) is the control gain, which is known.

For system (1), the control objective is twofold: (i). The expected output reference \( r_y \) can be tracked under the designed controller with guaranteed preassigned tracking performance; (ii). The tracking error system is finite-time convergent in the presence of unknown nonlinearities and external disturbance.

Remark 1. As presented in Section 1, one can find that the finite-time stability can be achieved via using SMC-based technique in the existing works. However, the relevant controller is pretty tedious owing to the usage of fractional power state or output feedback and symbolic function. Thus, in this paper, a different way to obtain the finite-time stability is proposed, which conquers the inherent drawbacks of SMC-based technique.

Prior to showing the controller design, some preliminary knowledge used in this paper is shown as follows.

2.1. Preliminary knowledge of radial basis function neural network (RBFNN)
Radial basis function neural network (RBFNN) has been widely applied to approximate smooth nonlinear functions with arbitrary approximation accuracy [7]. Thus, for an unknown nonlinear function \( f(X) : \mathbb{R}^N \rightarrow \mathbb{R} \), it can be approximated by

\[
f(X) = \mathcal{W}^T \varphi(X) + \delta(X)
\]

(2)

where \( X = [X_1, X_2, \ldots, X_N] \in \mathbb{R}^N \) is the input of a RBFNN. \( \mathcal{W}^* = [\mathcal{W}_1^*, \mathcal{W}_2^*, \ldots, \mathcal{W}_N^*]^T \in \mathbb{R}^\ell \) is the optimal weight vector of a RBFNN, which is obtained by

\[
\mathcal{W}^* = \arg \min_{\mathcal{W} \in \Omega_{\mathcal{W}}} \left[ \sup_{X \in \Omega_X} \left| f(X|\mathcal{W}) - f(X) \right| \right]
\]

(3)

where \( \Omega_{\mathcal{W}}, \Omega_X \) are the relevant compact sets of \( \mathcal{W}, X \), respectively. \( \delta(X) \) is the approximation error, which can be made sufficiently small via using sufficient nodes of RBFNN. \( \varphi(X) = [\varphi_1(X), \varphi_2(X), \ldots, \varphi_\ell(X)]^T \in \mathbb{R}^\ell \) is the kernel function ( \( \ell \) is the number of nodes in the RBFNN), which is usually chosen in the Gaussian form, i.e.,

\[
\varphi_i(X) = \exp \left( -\frac{(X - \zeta_i)^T(X - \zeta_i)}{2\xi_i^2} \right)
\]

(4)

where \( \zeta_i = [\zeta_{i,1}, \zeta_{i,2}, \ldots, \zeta_{i,N}] \in \mathbb{R}^N, \xi_i \in \mathbb{R} \) is the respective center and width of the Gaussian function.

Remark 2. Widespread applications indicate that the optimal weight vector \( \mathcal{W}^* \) is bounded.

3. Novel finite-time convergent controller with guaranteed tracking performance
In this section, the finite-time convergent controller design is divided into three parts, namely, finite-time convergent performance function design, finite-time convergent controller design and stability.
analysis. To achieve finite-time convergence, a novel practically finite-time convergent performance function is first proposed in the following.

3.1 Finite-time convergent performance function design

In the PPC structure, performance function is a vital one to characterize the transient and steady-state performance of the controlled system. In this paper, the performance function \( \rho(t) \) is derived by

\[
\dot{\rho}(t) = \begin{cases} 
-\eta_0 (\rho(t) - \rho_\alpha)^{\frac{m}{n}} - \eta_0 (\rho(t) - \rho_\alpha)^{\frac{n}{n}}, & (\rho(t) - \rho_\alpha) > 1 \\
0, & (\rho(t) - \rho_\alpha) = 1 \\
-\eta_0 (\rho(t) - \rho_\alpha)^{\frac{m}{n}} - \eta_0 (\rho(t) - \rho_\alpha)^{\frac{n}{n}}, & (\rho(t) - \rho_\alpha) < 1
\end{cases}
\]

where \( \rho_\alpha = \rho(0) > \rho_\alpha > 0 \) are the initial and final states of the performance function \( \rho(t) \).

Example 1. For brevity, \( m_1 = n_1, p_1 < q_1 \) are positive odd integers. \( \eta_0 \) is a positive constant. From Eq. (5), one can find that

\[
\lim_{|\rho(t)| > \rho_\alpha} \dot{\rho}(t) = \lim_{|\rho(t)| > \rho_\alpha} \dot{\rho}(t) = \lim_{|\rho(t)| > \rho_\alpha} \dot{\rho}(t) = 0.
\]

Thus, the performance function \( \rho(t) \) is continuous in the time domain. Accordingly, one can obtain the following property for the performance function.

**Property 1.** Performance function \( \rho(t) \) is positive and will converge to \( \rho_\alpha \) within finite time.

**Proof.** The proof of **Property 1** is organized as follows. Firstly, define a new variable \( \sigma = (\rho(t) - \rho_\alpha)^{\frac{1}{n}} \), then, the first and third parts of Eq. (5) can be rewritten as

\[
S_1 : \sigma + \frac{q_1 - p_1}{q_1} \eta_0 \left( \frac{m_0 - n_0}{n_0 (n - n_0)} \right) = 0; \quad S_2 : \sigma + \frac{q_1 - p_1}{q_1} \eta_0 (\sigma + 1) = 0
\]

For brevity, define \( h = [(m_1 - n_1) q_1] / [(q_1 - p_1) n_1] \). Inspired by [8], solving Eq. (6) yields the relevant convergence time \( T_o \), i.e.,

\[
\lim_{\sigma \rightarrow \sigma_o} T_o(S_\sigma) = \lim_{\sigma \rightarrow \sigma_o} \left( \frac{q_1}{q_1 - p_1} \eta_0 \left( \int_{\sigma_o}^{\sigma} \frac{1}{\sigma + 1} d\sigma + \int_{\sigma_o}^{\sigma_0} \frac{1}{\sigma^{n_0 + 1} + 1} d\sigma \right) \right) < \lim_{\sigma \rightarrow \sigma_o} \frac{q_1}{(q_1 - p_1) \eta_0} \left( \int \frac{1}{\sigma^{n_0 + 1} + 1} d\sigma \right)
\]

Based on Eqs. (6) and (7), one can easily obtain that there exists an upper bound for the convergence of system (5). Namely, \( \lim_{t \rightarrow \infty} \rho(t) = \rho_\alpha > 0 \). Consequently, **Property 1** is proved.

**Remark 3.** From Eqs. (5)-(7), one can find that the proposed performance function is a continuous one only with respect to time. Thus, it can be predefined by the users easily.

3.2 Finite-time convergent controller design with guaranteed output tracking performance

For system (1), the output tracking error is defined as \( e = y - y_\gamma \), wherein \( y_\gamma \) is the desired output reference command. Before moving on, the following two assumptions are given.

**Assumption 1.** The desired output reference command \( y_\gamma \) is smooth and its first derivative is known.

**Assumption 2.** The state variables of system (1) are available for measurement.

**Remark 4.** With consideration of that the reference command is designed by the user, thus **Assumption 1** is easily satisfied. As for **Assumption 2**, there are many effective measurement devices and techniques available. Thus, it is reasonable.

To guarantee the output tracking performance, based on the proposed performance function in subsection 3.1, the following prescribed performance is given
\[-\rho(t) < e < \rho(t) \quad (8)\]

By applying backstepping control technique, three steps are involved in the following controller design.

**Step 1.** When \( i = 1 \), we choose the following Lyapunov function

\[ V_i = \frac{1}{2} z_i^2 + \frac{1}{2} \hat{W}_i^T \dot{W}_i \]  

where \( z_i = e / \rho(t) \in (-1, 1) \) is the standard tracking error. \( \hat{W}_i = \mathbf{W}_i - \hat{\mathbf{W}}_i \) is the estimated weight vector error to be determined later (\( \mathbf{W}_i^*, \hat{\mathbf{W}}_i^* \) denote the optimal and estimated weight vectors of RBFNN, respectively). Based on system (1), the derivative of \( V_i \) equals to

\[ \dot{V}_i = -\frac{z_i}{\rho(t)(1-z_i^2)} \left( z_i + \frac{1}{\rho(t)} \rho(t) e \right) + \hat{W}_i^T \dot{W}_i \]  

Define the coordinate transformation \( z_2 = x_2 - s_i \), wherein, \( s_i \) is the approximation for the first virtual controller \( \alpha_i \), which is given later. Based on the preliminary knowledge in subsection 2.1, the unknown nonlinear function \( f_i(x) \) can be approximated by a RBFNN. Thus, Eq. (10) equals to

\[ \dot{V}_1 = \frac{z_i}{\rho(t)(1-z_i^2)} \left( z_1 + \alpha_1 - \dot{y}_i + \hat{\mathbf{W}}_1^T \phi_i(x) + \delta_1 + s_1 - \alpha_1 \right) + \hat{W}_1^T \dot{W}_1 \]  

where \( \delta_1 = \delta_1 + s_1 - \alpha_1 \). For Eq. (11), the following inequality holds

\[ \frac{z_i}{\rho(t)(1-z_i^2)} \left( z_2 + \delta_1 \right) \leq \frac{2z_i^2}{\rho^2(t)(1-z_i^2)} + \frac{1}{4} z_i^2 + \frac{1}{4} \delta_1^2 \]  

Thus, the first virtual controller \( \alpha_1 \) is devised as

\[ \alpha_1 = -k_1 \dot{z}_i - \hat{W}_1^T \phi_i(x) + \dot{y}_i + \frac{\rho(t)}{\rho(t)} e - \frac{2z_i}{\rho(t)(1-z_i^2)} \]  

where \( k_1 \) is the positive control gain. The corresponding adaptive scheme for the estimated weight vector is

\[ \dot{\hat{W}}_i = -\mu_i \hat{W}_i + \frac{z_i}{\rho(t)(1-z_i^2)} \phi_i(x) \]  

where \( \mu_i > 0 \) is a constant. By considering \( \dot{\hat{W}}_i = \dot{W}_i^* - \dot{\hat{W}}_i = -\dot{\hat{W}}_i \), thus, substituting Eqs. (13) and (14) into (11) gets

\[ \dot{V}_i \leq -\frac{k_1 z_i^2}{\rho(t)(1-z_i^2)} + \frac{1}{4} z_i^2 + \frac{1}{4} \delta_1^2 + \frac{z_i}{\rho(t)(1-z_i^2)} \hat{W}_i^T \phi_i(x) + \mu_i \hat{W}_i - \frac{z_i}{\rho(t)(1-z_i^2)} \phi_i(x) \]  

**Step 2.** When \( 2 \leq i \leq n-1 \), we define the relevant coordinate transformation as \( z_i = x_i - s_{i-1} \), wherein, \( s_{i-1} \) is output of the a low-pass filter used to approximate the derivative of the virtual controller.
This is also referred to as dynamic surface control in the existing works like [5]. \( s_j (j = 1, \ldots, n-2) \) is derived from

\[
\varepsilon_j \dot{s}_j + s_j = \alpha_j \left( s_j(0) = \alpha_j(0) \right)
\]  

(16)

The filter estimation error is defined as \( \chi_j = \alpha_j - s_j \). The relevant Lyapunov function is given by

\[
V_j = V_{j+1} + \frac{1}{2} z_i^2 + \frac{1}{2} \mathbf{W}_j^T \mathbf{W}_j
\]

(17)

where \( \mathbf{W}_j = \mathbf{W}_j^* - \mathbf{W}_j \) is the estimated weight vector error to be determined later ( \( \mathbf{W}_j^* \), \( \mathbf{W}_j \) denote the optimal and estimated weight vectors of RBFNN, respectively). Based on Eq. (1) and \textit{Step 1}, taking the derivative of \( V_j \) yields

\[
\dot{V}_j = \dot{V}_{j+1} + z_i \dot{z}_i + \mathbf{W}_j^T \dot{\mathbf{W}}_j = z_i \left( \dot{x}_i - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j
\]

\[
= z_i \left( x_i + f_i(x) - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j = z_i \left( \dot{z}_{i+1} + s_i + f_i(x) - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j
\]

(18)

Thus, the relevant virtual controller \( \alpha_j \) and the adaptive scheme for \( \mathbf{W}_j \) are expressed by

\[
\alpha_j = -k_i z_i - \frac{1}{4} z_i - \frac{1}{4} z_i - \mathbf{W}_j^T \varphi_i(x) + \dot{s}_i, \quad \dot{\mathbf{W}}_j = -\mu \dot{\mathbf{W}}_j + z_i \varphi_i(x)
\]

(19)

Substituting Eq. (19) into (18) yields

\[
\dot{V}_j = \dot{V}_{j+1} + z_i \dot{z}_i + \mathbf{W}_j^T \dot{\mathbf{W}}_j = z_i \left( \dot{x}_i - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j
\]

\[
= z_i \left( x_i + f_i(x) - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j = z_i \left( \dot{z}_{i+1} + s_i + f_i(x) - \dot{s}_i \right) + \dot{V}_{j+1} + \mathbf{W}_j^T \dot{\mathbf{W}}_j
\]

(20)

\[
\leq -\frac{k_i z_i^2}{\rho(t)\left(1 - z_i^2\right)^2} \sum_{j=2}^{J} k_j z_j^2 - \frac{1}{2} \sum_{j=1}^{J} \mu_j \mathbf{W}_j^T \mathbf{W}_j + \frac{1}{4} z_i^2 + \frac{1}{4} \sum_{j=1}^{J} \delta_j^2 + \frac{1}{2} \sum_{j=1}^{J} \mu_j \mathbf{W}_j^T \mathbf{W}_j
\]

where \( \delta_j = \chi_j + \dot{\delta}_j \) is the lumped error.

\textit{Step 3.} When \( i = n \), we define the relevant coordinate transformation as \( z_n = x_n - s_{n-1} \), wherein, \( s_{n-1} \) is given in Eq. (16). Then, the relevant Lyapunov function is chosen as

\[
V_n = V_{n+1} + \frac{1}{2} z_n^2 + \frac{1}{2} \mathbf{W}_n^T \mathbf{W}_n
\]

(21)

Similar to \textit{Steps 2} and \textit{3}, the derivative of \( V_n \) equals to

\[
\dot{V}_n = \dot{V}_{n+1} + z_n \dot{z}_n + \mathbf{W}_n^T \dot{\mathbf{W}}_n = z_n \left( bu + f_n(x) + d - \dot{s}_{n-1} \right) + \dot{V}_{n+1} - \mathbf{W}_n^T \dot{\mathbf{W}}_n
\]

(22)

The relevant actual control input \( u \) and the adaptive scheme for \( \mathbf{W}_n \) are

\[
u = \frac{1}{b} \left( -k_n z_n - \frac{1}{4} z_n - \frac{1}{4} z_n - \mathbf{W}_n^T \varphi_n(x) + \dot{s}_{n-1} \right), \quad \dot{\mathbf{W}}_n = -\mu_n \dot{\mathbf{W}}_n + z_n \varphi_n(x)
\]

(23)

Substituting Eqs. (20) and (23) into (22) gets
\[ \dot{V}_a \leq -\frac{k_i z_i^2}{\rho(t)(1-z_i^2)^2} - \frac{1}{2} \sum_{j=1}^{n} \mu_j \dot{W}_j^T \dot{W}_j + \frac{1}{4} \sum_{j=1}^{n} \delta_j^2 \dot{W}_j^T \dot{W}_j \]  

(24)

where \( \delta_j = d + \dot{\delta}_j \).

3.3. Stability Analysis

Based on the recursive controller design in subsection 3.2, the following theorem is attained.

**Theorem 1.** The output reference command can be tracked within finite time under the devised controller in Eq. (23). And the prescribed output tracking performance can be achieved. All the closed-loop signals are uniformly ultimately bounded.

**Proof.** Based on Eq. (24), one can obtain that there exists two positive constants \( \theta_0, \omega_0 \) such that the following inequality holds

\[ \frac{1}{2} \frac{z_i^2}{1-z_i^2} \leq \frac{\theta_0}{\omega_0} \Rightarrow |z_i| \leq \sqrt{\frac{2\omega_0}{\theta_0} + \theta_0} \Rightarrow |\epsilon| \leq \sqrt{\frac{2\omega_0}{\theta_0} + \theta_0} \rho(t) < \rho(t) \]  

(26)

As shown in Eq. (26), one can find that the output tracking error will be kept into the performance envelope in the whole time domain. Due to the finite-time convergence of the proposed performance function, it is easy to obtain that the output tracking error will converge to the final tracking error bound within finite time. Thereby, **Theorem 1** is proved.

4. Numerical Simulations

To validate the effectiveness of the proposed control scheme, the application to a chemical reactor system- continuously stirred tank reactor (CSTR) is organized, wherein, the CSTR system is given by

\[ \begin{align*}
\dot{x}_i &= -x_i + D_a (1-x_i) \exp\left(\frac{x_2}{1+x_2/k_0}\right) \\
\dot{x}_2 &= bu - (1+b)x_2 + B_0 D_a (1-x_2) \exp\left(\frac{x_2}{1+x_2/k_0}\right) + d
\end{align*} \]  

(27)

where \( D_a, k_0, B_0 \) represent Damköhler number, activated energy and heat of reaction [9]. In the simulation, \( D_a = 0.072, k_0 = 20, B_0 = 8, b = 0.3 \). The relevant simulation parameters are chosen as: \( k_i = 10, k_z = 100, \eta_0 = 0.15, m_i = q_i = 5, n_i = p_i = 3, \rho(0) = 5, \rho_\epsilon = 0.2, \epsilon_0 = 0.01, x_i(0) = 0.15, x_z(0) = -0.3 \), \( \mu_i = \mu_z = 2 \). The initial RBFNN weights \( \hat{W}_j (j=1,2) \) are randomly chosen in the interval [-1.5, 1.5]. The total nodes of the RBFNN are 20, the parameters involved in the Gaussian function \( \zeta_{ij}, \xi_{ij} (j=1,2, i=1,2,...,20) \) are randomly chosen in the interval [-1, 1]. The expected output reference is \( y_r = 0.5 \). The corresponding simulation results are given in Figs. 1 and 2.
As shown in Figs. 1 and 2, one can find that the desired output reference command can be tracked with guaranteed prescribed performance within 10 seconds. Thus, the proposed control scheme is effective in dealing with the output tracking problem. Figure 2 shows that the RBFNN weights will converge to the relevant optimal ones quickly. So the proposed adaptive scheme is stable. To sum up, the proposed control scheme and the adaptive scheme are effective.

5. Conclusion
In this paper, a novel finite-time control scheme is proposed in the prescribed performance control structure. Compared with the existing works, the fractional power state or output feedback is avoided, which makes the relevant controller easily achievable online. Application to a chemical reactor system validates the effectiveness of the proposed control scheme.

References
[1] Yu, S., Yu, X., Shirinzadeh, B., & Man, Z. (2005). Continuous finite-time control for robotic manipulators with terminal sliding mode. Automatica, 41(11), 1957-1964.
[2] Amato, F., Ariola, M., & Cosentino, C. (2010). Finite-time control of discrete-time linear systems: analysis and design conditions. Automatica, 46(5), 919-924.
[3] Xu, Y. (2017). Robust finite-time control for autonomous operation of an inverter-based microgrid. IEEE Transactions on Industrial Informatics, 13(5), 2717-2725.
[4] Bechlioulis, C. P., & Rovithakis, G. A. (2008). Robust adaptive control of feedback linearizable MIMO nonlinear systems with prescribed performance. IEEE Transactions on Automatic Control, 53(9), 2090-2099.
[5] Li, Y., & Tong, S. (2015). Prescribed performance adaptive fuzzy output-feedback dynamic surface control for nonlinear large-scale systems with time delays. Information Sciences, 292, 125-142.
[6] Wei, C., Luo, J., Dai, H., Yin, Z., & Yuan, J. (2017). Low-complexity differentiator-based decentralized fault-tolerant control of uncertain large-scale nonlinear systems with unknown dead zone. Nonlinear Dynamics, 89(4), 2573-2592.
[7] Song, Y., & Guo, J. (2017). Neuro-adaptive fault-tolerant tracking control of Lagrange systems pursuing targets with unknown trajectory. IEEE Transactions on Industrial Electronics, 64(5), 3913-3920.
[8] Ni, J., Liu, L., Liu, C., Hu, X., & Li, S. (2017). Fast fixed-time nonsingular terminal sliding mode control and its application to chaos suppression in power system. IEEE Transactions on Circuits and Systems-II: Express Briefs, CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, 64(2), 151-155.
[9] Chang, W. D. (2013). Nonlinear CSTR control system design using an artificial bee colony algorithm. Simulation Modelling Practice and Theory, 31, 1-9.