Research Article

Distributed Cooperative Sliding Mode Fault-Tolerant Control for Multiple High-Speed Trains Based on Actor-Critic Neural Network

Xiangyu Kong and Tong Zhang

1 College of Electrical and Information Engineering, Dalian Jiaotong University, Dalian 116028, China
2 College of Locomotive and Rolling Stock Engineering, Dalian Jiaotong University, Dalian 116028, China

Correspondence should be addressed to Tong Zhang; zhang_tong66@126.com

Received 31 March 2021; Revised 20 April 2021; Accepted 28 April 2021; Published 11 May 2021

Copyright © 2021 Xiangyu Kong and Tong Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article investigates the cooperative fault-tolerant control problem for multiple high-speed trains (MHSTs) with actuator faults and communication delays. Based on the actor-critic neural network, a distributed sliding mode fault-tolerant controller is designed for MHSTs to solve the problem of actuator faults. To eliminate the negative effects of unknown disturbances and time delay on train control system, a distributed radial basis function neural network (RBFNN) with adaptive compensation term of the error is designed to approximate the nonlinear disturbances and predict the time delay, respectively. By calculating the tracking error online, an actor-critic structure with RBFNN is used to estimate the switching gain of the distributed controller, which reduces the chattering phenomenon caused by sliding mode control. The global stability and ultimate bounded of all signals of the closed-loop system are proposed with strict mathematic proof. Simulations show that the proposed method has superior effectiveness and robustness compared with other fault-tolerant control methods, which ensures the safe operation of MHSTs under moving block conditions.

1. Introduction

In recent years, high-speed trains (HSTs) are increasingly popular because of the characteristics of high speed, high efficiency, and low energy consumption [1]. With the rapid development of HSTs, the requirements for reliability and safety of systems are becoming higher. In the course of the train operation, actuators in traction and braking systems are frequently affected by high temperature and violent vibration so that it may fail after long time high load operation. If the failures are not detected and treated in time, it may cause time delay or stop of trains; in critical condition, it will even lead to derailment, overturning, and other major accidents [2]. Thus, it is an urgent demand to design a fault-tolerant control scheme to guarantee the safe operation of train control system.

During the past years, there were various fault-tolerant control schemes developed for single train, including back stepping control [3], neural network control [4], and sliding mode control (SMC) [5]. In [3], an adaptive backstepping fault-tolerant control scheme was designed for HSTs with unknown parameters, actuator faults, and disturbances, and a piecewise time-varying indicator function was used to describe the train motion model. In [4], a neuroadaptive fault-tolerant control method was proposed for HSTs with actuation notches and antiskid constraints, and the radial basis function neural network (RBFNN) was used to study the nonlinear parameters of the system. In [5], an adaptive sliding mode fault-tolerant control (SMFTC) scheme was designed to solve the actuator uncertainties and faults of HSTs, simultaneously, and a dynamic model with input distribution matrix uncertainty is established to describe the properties of the train system. Nowadays, with the increasing pressure of urban traffic, the number of HSTs is gradually increasing, and the methods and ways for single-train operation control have not been enough to meet the efficiency and safety requirements so that there are frequent incidents of train delays and passenger dissatisfaction [6]. To improve the operation efficiency and ability of the rail transit system, the cooperative control among multiple high-speed
trains (MHSTs) has aroused widespread attention of many scholars [7]. For MHSTs, the purpose of cooperative control is to make decisions to coordinate train behavior based on more information and use the intelligent control algorithm to coordinate their respective state, speed, and the next step, which guarantees there are no accidents between two neighboring trains happening [8]. In [9], an adaptive fuzzy fault-tolerant control scheme with proportional and integral-based sliding mode technique was proposed for MHSTs with actuator faults, and a novel fuzzy system with minimal computational burden was designed to approximate the unknown disturbances. In [10], a distributed cooperative control method was proposed for MHSTs under moving block mode, and a virtual marshalling topology was used to establish the cooperative control model. In [11], a robust block mode, and a virtual marshalling topology was used to solve the cruise control problem of MHSTs under external disturbances, and the observer topology was designed to approximate the unknown disturbances. In addition to the design of controller, the cooperative control for MHSTs needs high communication requirements, such as packet losses and time delays during data transmission [12]. It is noted that the difficulties may worsen seriously with the increase in speed in real engineering [13]. To overcome the above shortcomings, in [14], a recursive filter was designed for train control systems with measurement noise and packet dropouts, and the upper bound of the filtering error was obtained by deriving the filter gain. In [15], a T-S fuzzy model was established to describe the nonlinear train networked control system, and the least mean square algorithm was used to predict the network delays to improve the control performance. Until now, there are still few works on the problems of cooperative control for MHSTs by considering actuator fault and communication delays, comprehensively. Therefore, it is of vital importance to design an appropriate distributed fault-tolerant control scheme to deal with the actuator faults and time delays for MHSTs to improve work efficiency and operation safety. In the field of control theory, due to its strong robustness, SMC has been widely applied to the cooperative control for complex nonlinear systems such as multilateral telerobotics [16], multiple quad rotors [17], and MHSTs [18]. However, the applications of conventional SMC are mainly limited by its chattering phenomenon in practice. For the drawback, some parameter tuning methods, such as fuzzy inference technology [19] and reinforcement learning algorithm [20], have been used to adjust the control gains to reduce the chattering phenomenon caused by switching control term of SMC. Considering the above problems mentioned, our main contributions are summarized as follows:

1. Considering the complexity of the actual operation condition, a method that combines the distributed SMFTC with neural networks is proposed for the cooperative control of MHSTs with actuator faults and communication delays.

2. With the compensation mechanism of the approximation error, a distributed RBFNN is designed to deal with the unknown nonlinear disturbances from train traction and braking, and a time delay prediction model based on RBFNN is established to predict the forward channel delay between controller and actuator.

3. Based on the current observed input-output data, an actor-critic neural network is designed to tune the switching gains of the distributed sliding mode controller in an adaptive way for overcoming the chattering problem.

The structure is presented as follows. In Section 2, the problem formulation for MHSTs with actuator faults is discussed. In Section 3, a distributed collaborative SMFTC scheme with an actor-critic neural network is designed, and the stability proof is analyzed. The main results and the conclusions are presented in Sections 4 and 5, respectively.

2. Problem Formulation

In this section, the dynamic equation of MHSTs is described, the actuator fault model is given, and the constraints on cooperative control of MHSTs are introduced.

2.1. Dynamic Model of MHSTs. Considering the operation process of MHSTs, its dynamic model is described as follows:

$$\begin{align*}
\dot{x}_i(t) &= v_i(t), \\
\dot{v}_i(t) &= \chi(u_i(t) - f_i(v_i(t)) - d_i(t)),
\end{align*}$$

where $i = 1, 2, ..., N$ is the number of trains running on the same railway, $x_i(t)$ and $v_i(t)$ are the position and speed of train $i$, respectively, $\chi$ is the acceleration factor, $u_i(t)$ is the unit control force of train $i$, and $f_i(v_i(t))$ and $d_i(t)$ are the unit basic resistance and additional resistance of train $i$, respectively, which can be described as follows [21]:

$$\begin{align*}
f_i(v_i(t)) &= a_0 + a_1 v_i(t) + a_2 v_i(t)^2, \\
d_i(t) &= 1000 \sin \beta_i(t) \frac{600}{R_i(t)} + 0.00013 L_i(t),
\end{align*}$$

where $a_0$, $a_1$, and $a_2$ are the time-varying resistance coefficients, which are closely related to the operation condition of the train, and $\beta_i(t)$, $R_i(t)$, and $L_i(t)$ are the gradient angle, curved track radius, and tunnel length, respectively.

2.2. Actuator Fault Model. With the train running for a long time, many components in the traction control system will suffer performance degradation to varying degrees and cause various faults [22]. Among them, the most common is the partial failure of the actuator. If not dealt with in time, it is easy to cause the tractive force and braking force to deviate from the expected value, resulting in various train accidents [3]. In general, the dynamic model of MHSTs with actuator partial failures can be described as follows:
\[
\dot{v}_i(t) = \theta_i(t)u_i(t) - a_{0i} - a_{1i}v_i(t) - a_{2i}v_i(t)^2 - 1000 \sin \beta_i(t) - \frac{600}{R(t)} - 0.00013L_i(t),
\]

where \( \theta_i(t) \) is the actuator effective factor of train \( i \), which can be described as follows:

\[
\theta_{ij}(t) = \begin{cases} 
1, & t < t_1, \\
\theta_{i1}(t), & t_1 \leq t < t_2, \\
\theta_{i2}(t), & t_2 \leq t < t_3, \\
\vdots 
\end{cases}
\]

where \( t_1 \) is the fault occurring time instant, \( j = 1, 2, \ldots, l \). It follows from (4), after the actuator failures, the control force will deviate from the expected value, which certainly affects the stability of the train system.

2.3. Constraints on Cooperative Operation of MHSTs. In order to ensure the operation safety of MHSTs on the same line, there must be a certain safety interval between each train [12]. As shown in Figure 1, when the following trains \( i = 1, 2, \ldots, n \) run on the same line, the radio block center (RBC) can collect real-time information such as train speed and position via the global system for mobile communications for railways (GSM-R). Afterwards, the RBC transmits the information to the vehicle equipment of the neighboring train to activate the distance monitoring module so that the multiple trains can calculate the minimum safe distance \( L_{\text{min}}(t) \) and track the preceding trains under moving block mode.

Under the moving block condition, the \( L_{\text{min}}(t) \) between neighboring trains is given as follows:

\[
L_{\text{min}}(t) = L_e(t) + L_b(t) + L_s(t) + L_i,
\]

where \( L_e(t) \) is the additional safety distance, \( L_b(t) \) is the braking distance, \( L_s(t) \) is the proper redundant safe distance, and \( L_i \) is the length of the train, as shown in [23].

\[
\begin{align*}
L_e(t) &= t_v v(t), \\
L_b(t) &= \sum \frac{v_1(t) - v_2(t)}{2a}, \\
L_s(t) &= C + 0.5v(t),
\end{align*}
\]

where \( t_v \) is the running time of the train after braking, \( v(t) \) is the actual speed of the train, \( v_1(t) \) and \( v_2(t) \) are the initial and final speeds of the train in the sampling time during braking, respectively, \( a \) is the average acceleration of the train in the sampling process, and \( C \) is the constant coefficient.

In the process of cooperative operation of MHSTs, the constraint conditions required by the train control system can be described as follows [23]:

\[
X_f(t) - X_b(t) > \text{L}_{\text{min}}(v(t)),
\]

where \( X_f(t) \) and \( X_b(t) \) are the actual running distance of the preceding and following trains, respectively.

3. Design of Distributed Collaborative SMFTC Scheme Based on Actor-Critic Neural Network

Under the condition of the actuator faults and communication delays, the control objective is to design a distributed cooperative fault-tolerant control scheme, which enables MHSTs to accurately track the desired speed and position so that the headway distances of each train with its neighbors are maintained in proper ranges. The diagram of the proposed fault-tolerant control scheme is illustrated in Figure 2.

Figure 2 shows that the distributed controller sends the control law \( u_i(t - \tau_{\text{ca}}) \) and forward timestamp \( \tau_{\text{ca}} \) into a process data packet to the train traction system via the multifunction vehicle bus (MVB) network. Under the partial failure conditions, the actuators in the traction system execute the control law and record the actual forward channel time delay \( \tau_{\text{ca}} \). Then, the sensors detect the actual output of the train at regular intervals and send \( v_i(t), x_i(t) \), feedback timestamp \( \tau_{\text{sc}} \), and \( \tau_{\text{ca}} \) into a process data packet to the train control system via the MVB network. Further, the distributed controller calculates the latest control law combined with the key information including the position information of other trains \( x_{i-1}(t) \) from the RBC and GSM-R, the gain parameters \( \lambda_i(t) \) from the actor-critic neural network, the disturbance estimation \( \hat{f}_1(v_i(t)) \) and time delay prediction \( \tau_{\text{ca}} \) from the distributed RBFNN, and the ideal reference input.

3.1. Design of Distributed Sliding Mode Fault-Tolerant Controller. In order to realize the accurate tracking of the first train to the desired position and speed, the tracking error signal can be defined as follows:

\[
\begin{align*}
\epsilon_1(t) &= x_{d1}(t) - x_1(t), \\
\dot{\epsilon}_1(t) &= v_{d1}(t) - v_1(t),
\end{align*}
\]

where \( \epsilon_1(t) \) and \( \dot{\epsilon}_1(t) \) are the position and speed tracking error of the first train, respectively, \( x_{d1}(t) \) and \( v_{d1}(t) \) are the desired position and speed of the first train, respectively, and \( x_1(t) \) and \( v_1(t) \) are the actual position and speed of the first train, respectively.

For tracking error signal (8), the following sliding surface is designed:

\[
s_1(t) = c_1 \epsilon_1(t) + \dot{\epsilon}_1(t),
\]

where \( c_1 > 0 \).

The time derivative of \( s_1(t) \) is derived as follows:

\[
\dot{s}_1(t) = c_1 \dot{\epsilon}_1(t) + \ddot{\epsilon}_1(t) = c_1 \epsilon_1(t) + \dot{v}_{d1}(t) - \dot{v}_1(t).
\]

Taking use of (1) to substitute for \( \dot{v}_1(t) \), (10) can be rewritten as follows:
\[
\dot{s}_i(t) = c\dot{e}_i(t) + \dot{\nu}_d(t) - \chi(\theta_i(t)u_i(t - \tau_{ca}) - f_1(v_1(t)) - d_i(t))
\]
\[
= c\dot{e}_i(t) + \dot{\nu}_d(t) - \delta_i(t)u_i(t - \tau_{ca}) + \chi f_1(v_1(t)) + \chi \delta_i(t),
\]  
(11)

where \(\chi \theta_i(t) = \delta_i(t), f_1(v_1(t))\) is the unknown nonlinear function. As can be seen from (2), with the increase of train speed, the nonlinear characteristic of \(f_1(v_1(t))\) becomes more and more obvious, thus increasing the design difficulty of the controller.

As the advantages of simple structure, fast learning speed, and strong approximation ability, the RBFNN is widely used with real-time control [24]. To solve the problem that the controller depends on the train parameters, we adopt the RBFNN to approximate the \(f_1(v_1(t))\) during the train traction braking process. Based on the Lyapunov theory, we use the train speed information collected by sensors to train the RBF neural network online. The neural network algorithms are described as follows:

\[
\begin{align*}
  f(t) & = w^T \phi(x) + \epsilon^*, \\
  \phi(x) & = e^{-1x - c_i x_i^2/2b_i^2},
\end{align*}
\]  
(12)

where \(x\) is the input of network, \(i = 1, 2, ..., n\) is the number of input nodes, \(j = 1, 2, ..., m\) is the number of hidden layer nodes, \(\phi(x)\) is the output of Gaussian function, \(c\) is the center vector of hidden layer neurons, \(b\) is the width of Gaussian function, \(w^*\) is the ideal weight, and \(\epsilon^*\) is the approximation errors and meets \(\epsilon^* \leq \epsilon_{\text{max}}\).

To achieve this control objective, we define the input of RBFNN as \(x = v_1(t)\), and the actual output of RBFNN is expressed as follows:

\[
\hat{f}(x) = \tilde{w}^T \phi(x),
\]
(13)

where \(\tilde{w}\) is the estimated weight; we define \(\tilde{f}(x) = f(t) - \hat{f}(x)\).

For the multitrain system (3), in terms of (11) and (13), the fault-tolerant controller of the first train is designed in the following form:

\[
u_1(t - \tau_{ca}) = \tilde{p}_1(t)\alpha_1(t),
\]
(14)

where \(\tilde{p}_1(t)\) and \(\alpha_1(t)\) are constructed in the following form:
where $k_i > 0$ and $\lambda_i > 0$.

Substituting (15) into (14), the control law can be rewritten as follows:

$$u_i (t - \tau_{ca}) = \tilde{p}_i (t)\left(c_i \dot{v}_i (t) + \dot{v}_{di} (t) + k_i s_i (t) + \lambda_i \text{sgn} (s_i (t)) + \chi \tilde{f}_i (v_i (t)) + \chi \tilde{e}_i (t)\right),$$

(16)

If the control law is designed as (16), the adaptive laws of the first train will be designed as follows:

$$\begin{cases} 
\dot{\hat{p}}_i (t) = \gamma_i \alpha_i (t) \text{sgn} (\hat{\delta}_i (t)), \\
\dot{\hat{w}}_i (t) = \gamma_i \chi s_i (t) \phi_i (x), \\
\dot{\hat{e}}_i (t) = \gamma_i \chi s_i (t),
\end{cases}$$

(17)

where $\gamma_i > 0$, $\gamma_i > 0$, and $\gamma_i > 0$.

In order to realize the cooperative control of MHSTs, we assume that all states of MHSTs are measurable and define the tracking error signal of the following trains as

$$e_i (t) = x_{ri} (t) - (x_i (t) - x_{i-1} (t)),$$

$$\dot{e}_i (t) = \nu_{di} (t) - v_i (t),$$

(18)

where $x_{ri} (t)$ is the desired distance headway of each train with its neighbors.

Based on the tracking error signal (18), the sliding surface is designed as follows:

$$s_i (t) = c_i \dot{e}_i (t) + \dot{e}_i (t),$$

(19)

where $c_i > 0$.

The derivative of $s_i (t)$ can be expressed as follows:

$$\dot{s}_i (t) = c_i \dot{e}_i (t) + \ddot{e}_i (t) = c_i \dot{e}_i (t) + \dot{v}_{di} (t) - \dot{v}_i (t).$$

(20)

Substituting (1) into (20), the $\dot{s}_i (t)$ can be rewritten as follows:

$$\dot{s}_i (t) = c_i \dot{e}_i (t) + \dot{v}_{di} (t) - \chi \theta_i (t) u_i (t - \tau_{ca}) - f_i (v_i (t)) - d_i (t)$$

$$= c_i \dot{e}_i (t) + \dot{v}_{di} (t) - \delta_i (t) u_i (t - \tau_{ca}) + \chi f_i (v_i (t)) + \chi d_i (t).$$

(21)

For the convenience of calculation, we define the $\tilde{p}_i (t)$ and $\alpha_i (t)$ as follows:

$$\begin{cases} 
\dot{\tilde{p}}_i (t) = \frac{1}{\delta_i (t)}, \\
\alpha_i (t) = c_i \dot{e}_i (t) + \dot{v}_{di} (t) + k_i s_i (t) + \lambda_i \text{sgn} (s_i (t)) + \chi \tilde{f}_i (v_i (t)) + \chi \tilde{e}_i (t),
\end{cases}$$

(22)

where $k_i > 0$ and $\lambda_i > 0$.

Referring to the same design idea as (16), the control law of the following trains is designed as follows:

$$u_i (t - \tau_{ca}) = \tilde{p}_i (t) \alpha_i (t)$$

$$= \tilde{p}_i (t) (c_i \dot{e}_i (t) + \dot{v}_{di} (t) + k_i s_i (t) + \lambda_i \text{sgn} (s_i (t)) + \chi \tilde{f}_i (v_i (t)) + \chi \tilde{e}_i (t)),$$

(23)

and the adaptive parameters are updated as follows:

$$\begin{cases} 
\dot{\hat{p}}_i (t) = \gamma_i \alpha_i (t) \text{sgn} (\hat{\delta}_i (t)), \\
\dot{\hat{w}}_i (t) = \gamma_i \chi s_i (t) \phi_i (x), \\
\dot{\hat{e}}_i (t) = \gamma_i \chi s_i (t),
\end{cases}$$

(24)

where $\gamma_i > 0$, $\gamma_i > 0$, and $\gamma_i > 0$.

3.2. Stability Analysis

**Theorem 1.** Consider the dynamic model (3) and constraint conditions (7) of MHSTs with actuator faults and communication delays, if the sliding surfaces are selected in (9) and (19), the distributed control laws are designed in (16) and (23), and the adaptive parameters are updated in (17) and (24); then all signals of the whole closed-loop system will be bounded, and the position and speed tracking errors will converge to zero. Furthermore, the whole closed-loop system is ultimately stable.

**Proof of Theorem 1.** The Lyapunov function candidate is selected as follows:

$$V (t) = \frac{1}{2} \left( s_i (t)^2 + \frac{\delta_i (t)}{\gamma_i} \tilde{p}_i (t)^2 + \frac{1}{\gamma_{ui}} \tilde{w}_i (t)^T \tilde{w}_i (t) + \frac{1}{\gamma_{ei}} \tilde{e}_i (t)^2 \right),$$

(25)

where $\tilde{P}_i (t) = \tilde{P}_i (t) - p_i (t)$, $\tilde{w} = w^* - \tilde{w}$, and $\tilde{e} = e^* - \tilde{e}$.

The derivative of $V (t)$ can be calculated as follows:

$$V (t) = s_i (t) \dot{s}_i (t) + \frac{\delta_i (t)}{\gamma_i} \tilde{p}_i (t) \dot{\tilde{p}}_i (t) - \frac{1}{\gamma_{ui}} \tilde{w}_i (t)^T \dot{\tilde{w}}_i (t) - \frac{1}{\gamma_{ei}} \tilde{e}_i (t) \dot{\tilde{e}}_i (t)$$

$$= s_i (t) (c_i \dot{e}_i (t) - \delta_i (t) u_i (t - \tau_{ca}) + \dot{v}_{di} (t) + \chi \tilde{f}_i (v_i (t)) + \chi \tilde{d}_i (t))$$

$$+ \frac{\delta_i (t)}{\gamma_i} \tilde{p}_i (t) \dot{\hat{p}}_i (t) - \frac{1}{\gamma_{ui}} \tilde{w}_i (t)^T \dot{\hat{w}}_i (t) - \frac{1}{\gamma_{ei}} \tilde{e}_i (t) \dot{\hat{e}}_i (t).$$

(26)
Substituting (22) into (26), we have the following:

\[
\dot{V}(t) = s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) u_i(t - \tau \tilde{y}_a) + \chi \tilde{f}_i(v_i(t)) - \chi \tilde{e}_i(t) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
+ \frac{\delta_i(t)}{\gamma_i} \frac{\pi_i}{\gamma_i} \tilde{p}_i(t) \tilde{p}_i(t) - \frac{1}{\gamma_i} \tilde{w}_i(t) \tilde{w}_i(t) - \frac{1}{\gamma_i} \tilde{e}_i(t) \tilde{e}_i(t) \\
= s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) u_i(t - \tau \tilde{y}_a) + \chi \tilde{w}_i(t) + \chi \tilde{e}_i(t) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
+ \frac{\delta_i(t)}{\gamma_i} \frac{\pi_i}{\gamma_i} \tilde{p}_i(t) \tilde{p}_i(t) + \tilde{w}_i(t) \tilde{w}_i(t) - \frac{1}{\gamma_i} \tilde{e}_i(t) \tilde{e}_i(t) \\
= s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) u_i(t - \tau \tilde{y}_a) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
+ \frac{\delta_i(t)}{\gamma_i} \frac{\pi_i}{\gamma_i} \tilde{p}_i(t) \tilde{p}_i(t) + \tilde{w}_i(t) \tilde{w}_i(t) - \frac{1}{\gamma_i} \tilde{e}_i(t) \tilde{e}_i(t) \\
\overset{(27)}{=} s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) u_i(t - \tau \tilde{y}_a) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
+ \frac{\delta_i(t)}{\gamma_i} \frac{\pi_i}{\gamma_i} \tilde{p}_i(t) \tilde{p}_i(t) + \tilde{w}_i(t) \tilde{w}_i(t) - \frac{1}{\gamma_i} \tilde{e}_i(t) \tilde{e}_i(t). 
\]

Substitute (23) and (24) into (27), then

\[
\dot{V}(t) = s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) \tilde{p}_i(t) \alpha_i(t) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) + \delta_i(t) \tilde{p}_i(t) \tilde{s}_i(t) \alpha_i(t) \text{sgn}(\delta_i(t)) \\
= s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) \tilde{p}_i(t) \alpha_i(t) + \delta_i(t) \tilde{p}_i(t) \alpha_i(t) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
= s_i(t)(\alpha_i(t) - k_i s_i(t) - \delta_i(t) \tilde{p}_i(t) \alpha_i(t) + \chi d_i(t) - \lambda_i \text{sgn}(s_i(t))) \\
= -k_i s_i(t)^2 + \chi d_i(t) s_i(t) - \lambda_i |s_i(t)|. 
\]

When \( \lambda_i \) meets \( \lambda_i > \chi d_i(t) \), \( \dot{V}(t) \leq -k_i s_i(t)^2 \), then \( \dot{V}(t) \leq 0 \).

As \( \dot{V}(t) > 0 \) and \( \dot{V}(t) \leq 0 \), \( V(t) \) is bounded, then \( s_i(t) \), \( \tilde{w}_i(t) \), \( \tilde{e}_i(t) \), and \( \tilde{p}_i(t) \) are bounded. According to \( V(t) \leq -k_i s_i(t)^2 \), the following inequality can be obtained:

\[
\int_0^t \dot{V}(t) dt \leq -k_i \int_0^t s_i(t)^2 dt. 
\]

From (29), we obtain

\[
V(\infty) - V(0) \leq -k_i \int_0^t s_i(t)^2 dt. 
\]

When \( t \to \infty \), \( \int_0^\infty s_i(t)^2 dt \) is bounded because of the boundedness of \( V(t) \). According to the Barbalat lemma [25], \( s_i(t) \to 0 \) as \( t \to \infty \), then \( \tilde{e}_i(t) \to 0 \), \( \tilde{e}_i(t) \to 0 \). The proof is completed.

3.3. Design of Actor-Critic Neural Network. In the design of the sliding mode controller, in order to ensure the stability of closed-loop system, a large switching gain is needed for large disturbances, resulting in the chattering of the system. In (28), the switching gain \( \tilde{\lambda}_i \) is used to compensate for the uncertain disturbances \( \chi d_i(t) \) to ensure \( \dot{V}(t) < 0 \). Because the unknown disturbances are time-varying, the switching gain should be designed as the time-varying parameter for reducing the chattering phenomenon. Because of its fast convergence and good optimization ability, reinforcement learning algorithms are widely used in the field of artificial intelligence [26]. Among them, the actor-critic learning algorithm is often used for strategic approximation of continuous control problems [27]. To overcome the problem mentioned above, we use the RBFNN to construct a reinforcement learning algorithm with an actor-critic structure to estimate the \( \tilde{\lambda}_i(t) \) online, so as to counteract the unknown disturbances, and the train speed and speed tracking error are applied to the training of the actor-critic neural network.

3.3.1. Design of Critic Neural Network. The state value function \( f_i(t) \) is defined as follows:

\[
f_i(t) = \tilde{w}_{ci}^T(t) \varphi_i(x),
\]

where \( \tilde{w}_{ci}(t) \) is the estimate value of the ideal weight \( w_{ci}(t) \), \( \varphi_i(x) \) is the Gaussian basis function, and \( x \) is the input of network with \( x = v_i(t) \).

Considering the influence of system tracking error on control performance, the utility function is defined as follows:

\[
R_i(t) = \rho_i r_i(t),
\]

where \( \rho_i \) is the enhanced signal coefficient of tracking error and \( r_i(t) \) is the enhanced signal of tracking error, which is given as follows [27]:

\[\]


\[ r_i(t) = \begin{cases} 
0, & \hat{e}_i^2(t) \leq \xi_i, \\
1, & \hat{e}_i^2(t) > \xi_i, 
\end{cases} \tag{33} \]

where \( \xi_i \) is the tolerance error. When the error function is less than \( \xi_i \), the tracking performance is ideal. On the contrary, it indicates that the tracking performance is poor.

Time difference error expresses the quality of decisions of actor neural network, which is defined as follows:

\[ \delta_{TDi}(t) = \alpha_i(t)J_i(t) - (J_i(t - 1) - R_i(t)), \tag{34} \]

where \( \alpha_i(t) \) is the discount factor with \( 0 \leq \alpha_i(t) \leq 1 \). If the \( \alpha_i(t) \) is small, it means the agent is more concerned with maximizing the present income. In contrast, as the \( \alpha_i(t) \) approaches 1, the agent will think more about future gains.

Substituting (31) into (34), we obtain the following:

\[ \delta_{TDi}(t) = R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x). \tag{35} \]

The cost function of the system is defined as follows:

\[ E_i(t) = \frac{1}{2}\delta_{TDi}^2(t). \tag{36} \]

The partial derivatives of (36) are obtained as follows:

\[ \frac{\partial E_i(t)}{\partial \delta_{TDi}(t)} = \frac{\partial E_i(t)}{\partial \delta_{TDi}(t)} \frac{\partial \delta_{TDi}(t)}{\partial \hat{w}(t)} = \frac{\partial \delta_{TDi}(t)}{\partial \hat{w}(t)} \delta_{TDi}(t). \tag{37} \]

The partial derivatives of (35) can be obtained as follows:

\[ \frac{\partial \delta_{TDi}(t)}{\partial \delta_{TDi}(t)} = \alpha_i(t)\varphi_i(x). \tag{38} \]

Substituting (35) and (38) into (37), we have the following:

\[ \frac{\partial E_i(t)}{\partial \delta_{TDi}(t)} = \alpha_i(t)\varphi_i(x) \left[ R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x) \right]. \tag{39} \]

According to the gradient descent method, the weight update law of critic neural network is given by

\[ \hat{w}_{ci}(t + 1) = \hat{w}_{ci}(t) - \eta_{ci}\frac{\partial E_i(t)}{\partial \hat{w}_{ci}(t)}, \tag{40} \]

where \( \eta_{ci} \) is the learning rate of \( \hat{w}_{ci}(t) \). Substituting (39) into (40), we get the following weight update law:

\[ \hat{w}_{ci}(t + 1) = \hat{w}_{ci}(t) - \eta_{ci}\alpha_i(t)\varphi_i(x) \left[ R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x) \right]. \tag{41} \]

3.3.2. Design of Actor Neural Network. In order to improve computational efficiency, the same RBFNN is used to learn value function and action function. Therefore, the action function of actor neural network is defined as follows:

\[ \lambda_i(t) = \hat{w}_{ai}^T(t)\varphi_i(x), \tag{42} \]

where \( \hat{w}_{ai}(t) \) is the estimated value of the ideal weight \( w_{ai}(t) \) and \( \lambda_i(t) \) is the estimation of the switching gain.

In general, the output of the actor neural network needs to be superimposed with a Gaussian signal to act on the sliding mode controller; the specific calculation method is as follows:

\[ \hat{\lambda}_i(t) = \lambda_i(t) + \lambda_{ai}(0, \sigma(J_i(t))), \tag{43} \]

where \( \sigma(J_i(t)) = 1/(1 + e^{21(J_i(t))}) \) and \( \hat{\lambda}_i(t) \) is the actual output of the actor neural network.

According to (38), the following approximate strategy gradient estimation algorithm is adopted [20]:

\[ \frac{\partial \hat{\lambda}_i(t)}{\partial \hat{w}_{ai}(t)} = \frac{\partial \hat{\lambda}_i(t) - \lambda_i(t)}{\sigma(J_i(t))} \varphi_i(x) \delta_{TDi}(t). \tag{44} \]

Substituting (44) into (37), we obtain the following:

\[ \frac{\partial E_i(t)}{\partial \hat{w}_{ai}(t)} = \frac{\lambda_i(t) - \lambda_i(t)}{\sigma(J_i(t))} \varphi_i(x) \left[ R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x) \right]. \tag{45} \]

Using (35), (45) can be simplified as follows:

\[ \frac{\partial E_i(t)}{\partial \hat{w}_{ai}(t)} = \frac{\lambda_i(t) - \lambda_i(t)}{\sigma(J_i(t))} \varphi_i(x) \left[ R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x) \right]. \tag{46} \]

Based on the gradient descent algorithm, the weight updating mode of the actor neural network can be obtained as follows:

\[ \hat{w}_{ai}(t + 1) = \hat{w}_{ai}(t) - \eta_{ai}\frac{\partial E_i(t)}{\partial \hat{w}_{ai}(t)}, \tag{47} \]

where \( \eta_{ai} \) is the learning rate of \( \hat{w}_{ai}(t) \). Substituting (46) into (47), we have the following:

\[ \hat{w}_{ai}(t + 1) = \hat{w}_{ai}(t) - \eta_{ai}\frac{\lambda_i(t) - \lambda_i(t)}{\sigma(J_i(t))} \varphi_i(x) \left[ R_i(t) + \alpha_i(t)\hat{w}_{ci}^T(t)\varphi_i(x) - \hat{w}_{ci}^T(t - 1)\varphi_i(x) \right]. \tag{48} \]

4. Simulation Results

In order to analyze the performance of the above distributed cooperative SMFTC method, we select four CRH3 trains as the controlled object. The main parameters of the CRH3 train are as follows [28]: the speed range is 0–350 km/h, the continuous running speed is 300 km/h, the total weight of the train is 400 tons, the train length is 200 m, and the rotary mass coefficient is 0.06. The reference input is set as the actual working conditions of the train including traction, braking, and inertia. Referring to the literature [29], the task cycle is set as 50 ms, the sampling cycle is set as 64 ms, and
the load rate is set as 45%. The change of failure factors of each train actuator is set as (49). Control parameters are shown in Table 1.

\[
\theta_i(t) = \begin{cases} 
1, & t < 12.8s, \\
0.35 + 0.1 \sin(0.3t), & 12.8s \leq t < 76.8s, \\
0.28 + 0.1 \cos(0.25t), & 76.8s \leq t < 128s, \\
0.23 + 0.1e^{-0.1t}, & t \geq 128s.
\end{cases}
\] (49)

4.1. Analysis of Cooperative Control Results of MHSTs. To illustrate the effectiveness of the proposed method, we use (5)–(7) to fully calculate the length of each train and the safety redundancy distance and other factors. The tracking interval distances between two neighboring trains are set as 1000 m. The initial state of each train is set as \( u_1 (0) = 0, v_1 (0) = 0, x_1 (0) = 0, u_2 (0) = 0, v_2 (0) = 0, x_2 (0) = 1000, u_3 (0) = 0, v_3 (0) = 0, x_3 (0) = 2000, u_4 (0) = 0, v_4 (0) = 0, \) and \( x_4 (0) = 3000. \) The speed and position tracking results of MHSTs are shown in Figure 3.

Figure 3 shows that under the same desired speed and line conditions, the actual running speed of all trains is always the same. Thus, the headway distance between two neighboring trains is always maintained at 1000 m, which ensures the coordinated operation of MHSTs with actuator faults and communication delays under the moving block condition.

4.2. Comparison of Different Fault-Tolerant Control Methods under Actual Working Conditions. In this section, both fuzzy logic system and actor-critic neural network can be used to estimate the switching gain of sliding mode controllers. Among them, the design of the fuzzy logic system is more dependent on experience. Thus, with the increase of the parameters to be optimized, the design difficulty of fuzzy

---

**Table 1: Control parameters.**

| Parameters | Value | Unit symbol |
|------------|-------|-------------|
| \( a_{0i}(t) \) | \( a_{0i}(t) \in [0.53, 0.58] \) | N/kg |
| \( a_{1i}(t) \) | \( a_{1i}(t) \in [0.0039, 0.0041] \) | N/s/m-kg |
| \( a_{2i}(t) \) | \( a_{2i}(t) \in [0.000114, 0.0001176] \) | N/s²/m-kg |
| \( \chi \) | \( \chi = 1/(1 + 0.06) \) | — |
| \( \beta_i(t) \) | \( \beta_i(t) = [\pi \times \text{rand}]/180 \) | — |
| \( R_i(t) \) | \( R_i(t) \in [5, 6] \) | km |
| \( L_i(t) \) | \( L_i(t) \in [4, 4.3] \) | km |
| \( c_i \) | \( c_1 = 6.49, c_2 = 8.74, c_3 = 1.35, c_4 = 6.75 \) | — |
| \( k_i \) | \( k_1 = 5.78, k_2 = 2.83, k_3 = 16.11, k_4 = 7.52 \) | — |
| \( \gamma_{pi} \) | \( \gamma_{pi} = 0.63, \gamma_{p2} = 1.94, \gamma_{p3} = 0.94, \gamma_{p4} = 0.8 \) | — |
| \( \gamma_{ui} \) | \( \gamma_{ui} = 0.53, \gamma_{u2} = 0.95, \gamma_{u3} = 1.94, \gamma_{u4} = 1.07 \) | — |
| \( \xi_i \) | \( \xi_1 = 0.78, \xi_2 = 1.61, \xi_3 = 1.05, \xi_4 = 0.58 \) | — |
| \( \alpha_i \) | \( \alpha_i = 0.9, \alpha_2 = 0.9, \alpha_3 = 0.9, \alpha_4 = 0.9 \) | — |
| \( \eta_i \) | \( \eta_1 = 0.5, \eta_2 = 0.5, \eta_3 = 0.5, \eta_4 = 0.5 \) | — |
| \( \rho_i \) | \( \rho_1 = 1, \rho_2 = 1, \rho_3 = 1, \rho_4 = 1 \) | — |
| \( w_i(0) \) | \( w_1(0) = \text{rands}(11, 1), w_2(0) = \text{rands}(11, 1) \) | — |
| \( w_{ci}(0) \) | \( w_{c1}(0) = \text{rands}(11, 1), w_{c2}(0) = \text{rands}(11, 1) \) | — |
| \( w_{ci}(0) \) | \( w_{ci}(0) = \text{rands}(11, 1), w_{ci}(0) = \text{rands}(11, 1) \) | — |
| \( w_c(0) \) | \( w_1(0) = \text{rands}(11, 1), w_2(0) = \text{rands}(11, 1) \) | — |
| \( w_{ci}(0) \) | \( w_{ci}(0) = \text{rands}(11, 1), w_{ci}(0) = \text{rands}(11, 1) \) | — |

![Figure 3: The speed and position tracking results of MHSTs using the proposed method.](image-url)
logic systems will be increased. In order to verify the advantages of the proposed method, we choose the adaptive fuzzy sliding mode fault-tolerant control (AFSMFTC) method [30] with $c_1 = 0.8$, $c_2 = 0.8$, $c_3 = 0.3$, $c_4 = 0.8$, $k_1 = 0.8$, $k_2 = 0.5$, $k_3 = 1$, $k_4 = 0.5$, $\gamma_1 = 0.5$, $\gamma_2 = 0.3$, $\gamma_3 = 0.3$, $\gamma_4 = 0.5$, $G_1 = 0.4$, $G_2 = 0.4$, $G_3 = 0.3$, and $G_4 = 0.4$ and the adaptive sliding mode fault-tolerant control (ASMFTC) method [31] with $c_1 = 2.2$, $c_2 = 3.3$, $c_3 = 5.3$, $c_4 = 1.8$, $k_1 = 0.3$, $k_2 = 0.5$, $k_3 = 0.3$, $k_4 = 0.6$, $\gamma_1 = 0.8$, $\gamma_2 = 0.8$, $\gamma_3 = 1$, $\gamma_4 = 0.4$, $\eta_1 = 12$, $\eta_2 = 12$, $\eta_3 = 12$, and $\eta_4 = 12$ as a comparison. The initial state of each train is set the same as in Section 4.1. Figure 4 shows the control effects of the first train using different fault-tolerant control methods. Figure 5 shows the speed tracking errors of MHSTs using different fault-tolerant control methods. Figure 6 shows the displacement interval errors of MHSTs using different fault-tolerant control methods. Table 2 shows the position interval error and speed tracking error of each train under different control methods recorded by root mean square error (RMSE) and mean absolute error (MAE). The RMSE and MAE are defined as follows:

$$e_{\text{RMSE}} = \sqrt{\frac{1}{L_i} \sum_{i=1}^{L_i} (x_{ri}(t) - \Delta x_i(t))^2},$$

$$e_{\text{MAE}} = \frac{1}{L_i} \sum_{i=1}^{L_i} |x_{ri}(t) - \Delta x_i(t)|,$$

where $L_i$ is the total running time of the train $i$, $\Delta x_i(t) = x_i(t) - x_{i-1}(t)$.

As shown in Figure 4, in the traction stage, the speed tracking result of the proposed method is close to that of the

Figure 4: The control effect of the first train using different fault-tolerant control methods. (a) The speed tracking results. (b) The position tracking results. (c) The unit control force input results.
ASMFTC method, but the AFSMFTC method has a large speed tracking error, which indicates that, with the increase of the number of trains, the design difficulty of the fuzzy logic rule base increases, the number of learning parameters increases sharply, and the dynamic performance of the system is seriously affected. In the braking and inertia stages, compared with the ASMFTC and AFSMFTC methods, the proposed method can achieve smooth switching at the steady-state working point and has the advantages of accurate tracking and fast response. In addition, the proposed method uses the actor-critic neural network to estimate the switching gain of the sliding mode controller, which can effectively reduce the chattering of the system, and the output fluctuation range of the control force is obviously reduced compared with other methods.

It can be seen from Figure 5 that with the partial failure of actuators, the proposed method uses the actor-critic neural network and RBFNN with error compensation to deal with the problems of unknown disturbances and controller tuning of the system, respectively. Thus, the speed tracking error can be quickly adjusted to a minimum range, which guarantees the smooth tracking of desired speed. However, due to its poor ability of fault suppression, the tracking error of other control methods frequently oscillates, leading to a poor speed tracking effect.

Figure 6 shows that, compared with other control methods, the proposed method always keeps the position interval error between each train within a small range, which ensures that the following train can track the front train in real time and accurately under the moving block mode, thus
Figure 6: The position interval errors of MHSTs using different fault-tolerant control methods. (a) The position interval error of train 1. (b) The position interval error of train 2. (c) The position interval error of train 3. (d) The position interval error of train 4.

Table 2: The comparison of position interval and speed tracking errors.

| Evaluation index | ASMFTC | AFSMFTC | Proposed method |
|------------------|--------|---------|-----------------|
| $e_1(t)$         | RMSE   | 0.1038  | 0.2078          | 0.0213          |
|                  | MAE    | 0.0282  | 0.0631          | 0.0102          |
| $\dot{e}_1(t)$   | RMSE   | 0.8275  | 1.0825          | 0.2535          |
|                  | MAE    | 0.3059  | 0.2856          | 0.0354          |
| $e_2(t)$         | RMSE   | 0.0653  | 0.2542          | 0.0098          |
|                  | MAE    | 0.0209  | 0.0801          | 0.0036          |
| $\dot{e}_2(t)$   | RMSE   | 0.8984  | 1.443           | 0.2292          |
|                  | MAE    | 0.3116  | 0.4355          | 0.0368          |
| $e_3(t)$         | RMSE   | 0.0751  | 0.2679          | 0.0468          |
|                  | MAE    | 0.0187  | 0.1233          | 0.0351          |
| $\dot{e}_3(t)$   | RMSE   | 0.8598  | 0.959           | 0.2263          |
|                  | MAE    | 0.3417  | 0.2865          | 0.0563          |
| $e_4(t)$         | RMSE   | 0.0859  | 0.242           | 0.0126          |
|                  | MAE    | 0.0339  | 0.0845          | 0.0076          |
| $\dot{e}_4(t)$   | RMSE   | 1.0305  | 1.6455          | 0.2584          |
|                  | MAE    | 0.3188  | 0.4911          | 0.0532          |
solving the cooperative fault-tolerant control problems of MHSTs.

From the data in Table 2, under actual working conditions, the proposed method significantly reduces the position interval and speed tracking errors of each train and has more ideal robustness and stability compared with other control methods, which can meet the complex nonlinearities and time variations of MHSTs.

5. Discussions

With the wide application of modern communication technology in train control systems, the design of a cooperative control scheme for multiple trains has become the best choice to improve the train running efficiency and system safety. Considering that there are few research methods for fault-tolerant control of multiple trains at present, we extend the research idea of fault-tolerant control for a single train to multiple trains and propose a distributed SMFTC method for multiple trains. The proposed method can solve the unknown actuator faults and the system uncertainties at the same time by designing the adaptive compensation law. Compared with reference [21], this method does not need the neural network fault observer and cannot be affected by fault diagnosis error, which can well deal with the uncertainty caused by fault and simplifies the design complexity of the fault-tolerant controller in [21]. It should be noted that on the basis of existing cooperative fault-tolerant control methods for multiple trains [9], the proposed method further considers communication delays during signal transmission, and the distributed neural network prediction model is introduced into the control scheme to compensate for the influence of network delay on the train control system.

6. Conclusions

In this paper, we have proposed a distributed SMFTC method based on the actor-critic neural network for MHSTs with actuator faults and communication delays. An adaptive compensation control law is designed to eliminate the influence of unknown actuator faults on the train control system. The distributed RBFNN with the compensation mechanism of approximate error is designed to deal with the external disturbance and time delay of the system, respectively. By estimating the switching gain in the distributed sliding mode controller using the actor-critic neural network, the precise control quantity is obtained to reduce the damages caused by the chattering of the system. Simulations indicate that the proposed method can effectively reduce the chattering phenomenon and track the change of reference input quickly and accurately, which ensures the safe operation of MHSTs. In addition, under the same fault conditions and operating environment, the parameter of the controller can be optimized by using the online adaptive parameter tuning mechanism of an actor-critic neural network, which is not available in other general RBFNNs. Future research will focus on the problems of sensor failure and cyber security of cooperative control for MHSTs.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This research was funded by the Natural Science Foundation of Liaoning Province (grant no. 20180551003).

References

[1] X. Yao, J. H. Park, H. Dong, L. Guo, and X. Lin, “Robust adaptive nonsingular terminal sliding mode control for automatic train operation,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 12, pp. 2406–2415, 2019.

[2] Z. Mao, G. Tao, B. Jiang, X.-G. Yan, and M. Zhong, “Adaptive position tracking compensation for high-speed trains with actuator failures,” IFAC-PapersOnLine, vol. 50, no. 1, pp. 14266–14271, 2017.

[3] S. K. Liu, B. Jiang, Z. H. Mao, and S. X. Ding, "Adaptive backstepping based fault-tolerant control for high-speed trains with actuator faults," International Journal of Control, Automation and Systems, vol. 17, no. 4, pp. 1408–1420, 2019.

[4] D.-Y. Li, P. Li, W.-C. Cai, X.-P. Ma, B. Liu, and H.-H. Dong, "Neural adaptive fault tolerant control for high speed trains considering actuation notches and antiskid constraints," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 5, pp. 1706–1718, 2019.

[5] Z. Mao, X.-G. Yan, B. Jiang, and M. Chen, "adaptive fault-tolerant sliding-mode control for high-speed trains with actuator faults and uncertainties," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 6, pp. 2449–2460, 2020.

[6] P. Xiang, X. Wang, and L. Mo, "Cooperative $\mathcal{SH}_\infty$ control of multiple high-speed trains with saturation constraints," IEEE Access, vol. 7, pp. 129437–129442, 2019.

[7] S. Li, L. Yang, and Z. Gao, "Distributed optimal control for multiple high-speed train movement: an alternating direction method of multipliers," Automatica, vol. 112, Article ID 108646, 2020.

[8] W. Bai, Z. Lin, and H. Dong, "Coordinated control in the presence of actuator saturation for multiple high-speed trains in the moving block signaling system mode," IEEE Transactions on Vehicular Technology, vol. 69, no. 8, pp. 8054–8064, 2020.

[9] X. G. Guo, J. L. Wang, and F. Liao, "Adaptive fuzzy fault-tolerant control for multiple high-speed trains with PI-based sliding mode," IET Control Theory and Applications, vol. 11, no. 8, pp. 1234–1244, 2016.

[10] B. Ning, H. R. Dong, S. G. Gao, T. Tang, and W. Zhang, "Distributed cooperative control of multiple high-speed trains under a moving block system by nonlinear mapping-based feedback," Science China (Information Sciences), vol. 61, no. 12, pp. 22–33, 2018.

[11] X. Wang, L. Zhu, H. Wang, T. Tang, and K. Li, "Robust distributed cruise control of multiple high-speed trains based on neural network with compensation law. Compared with reference [21], this method does not need the neural network fault observer and cannot be affected by fault diagnosis error, which can well deal with the uncertainty caused by fault and simplifies the design complexity of the fault-tolerant controller in [21]. It should be noted that on the basis of existing cooperative fault-tolerant control methods for multiple trains [9], the proposed method further considers communication delays during signal transmission, and the distributed neural network prediction model is introduced into the control scheme to compensate for the influence of network delay on the train control system.

6. Conclusions

In this paper, we have proposed a distributed SMFTC method based on the actor-critic neural network for MHSTs with actuator faults and communication delays. An adaptive compensation control law is designed to eliminate the influence of unknown actuator faults on the train control system. The distributed RBFNN with the compensation mechanism of approximate error is designed to deal with the external disturbance and time delay of the system, respectively. By estimating the switching gain in the distributed sliding mode controller using the actor-critic neural network, the precise control quantity is obtained to reduce the damages caused by the chattering of the system. Simulations indicate that the proposed method can effectively reduce the chattering phenomenon and track the change of reference input quickly and accurately, which ensures the safe operation of MHSTs. In addition, under the same fault conditions and operating environment, the parameter of the controller can be optimized by using the online adaptive parameter tuning mechanism of an actor-critic neural network, which is not available in other general RBFNNs. Future research will focus on the problems of sensor failure and cyber security of cooperative control for MHSTs.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This research was funded by the Natural Science Foundation of Liaoning Province (grant no. 20180551003).

References

[1] X. Yao, J. H. Park, H. Dong, L. Guo, and X. Lin, “Robust adaptive nonsingular terminal sliding mode control for automatic train operation,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 12, pp. 2406–2415, 2019.

[2] Z. Mao, G. Tao, B. Jiang, X.-G. Yan, and M. Zhong, “Adaptive position tracking compensation for high-speed trains with actuator failures,” IFAC-PapersOnLine, vol. 50, no. 1, pp. 14266–14271, 2017.

[3] S. K. Liu, B. Jiang, Z. H. Mao, and S. X. Ding, "Adaptive backstepping based fault-tolerant control for high-speed trains with actuator faults," International Journal of Control, Automation and Systems, vol. 17, no. 4, pp. 1408–1420, 2019.

[4] D.-Y. Li, P. Li, W.-C. Cai, X.-P. Ma, B. Liu, and H.-H. Dong, "Neural adaptive fault tolerant control for high speed trains considering actuation notches and antiskid constraints," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 5, pp. 1706–1718, 2019.

[5] Z. Mao, X.-G. Yan, B. Jiang, and M. Chen, "adaptive fault-tolerant sliding-mode control for high-speed trains with actuator faults and uncertainties," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 6, pp. 2449–2460, 2020.

[6] P. Xiang, X. Wang, and L. Mo, "Cooperative $\mathcal{SH}_\infty$ control of multiple high-speed trains with saturation constraints," IEEE Access, vol. 7, pp. 129437–129442, 2019.

[7] S. Li, L. Yang, and Z. Gao, "Distributed optimal control for multiple high-speed train movement: an alternating direction method of multipliers," Automatica, vol. 112, Article ID 108646, 2020.

[8] W. Bai, Z. Lin, and H. Dong, "Coordinated control in the presence of actuator saturation for multiple high-speed trains in the moving block signaling system mode," IEEE Transactions on Vehicular Technology, vol. 69, no. 8, pp. 8054–8064, 2020.

[9] X. G. Guo, J. L. Wang, and F. Liao, "Adaptive fuzzy fault-tolerant control for multiple high-speed trains with PI-based sliding mode," IET Control Theory and Applications, vol. 11, no. 8, pp. 1234–1244, 2016.

[10] B. Ning, H. R. Dong, S. G. Gao, T. Tang, and W. Zhang, "Distributed cooperative control of multiple high-speed trains under a moving block system by nonlinear mapping-based feedback," Science China (Information Sciences), vol. 61, no. 12, pp. 22–33, 2018.

[11] X. Wang, L. Zhu, H. Wang, T. Tang, and K. Li, "Robust distributed cruise control of multiple high-speed trains based
on disturbance observer,” IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 1, pp. 267–279, 2021.
[12] H. Zhao and X.-W. Dai, “Event-triggered adaptive control for multiple high-speed trains with deception attacks in bottleneck sections,” Information Sciences, vol. 547, pp. 470–481, 2021.
[13] B. Chen, “Development and reference of the TCMS on power centralized EMU abroad,” Railway Locomotive & Car, vol. 39, no. 1, pp. 7–14, 2019.
[14] T. Wen, L. Zou, J. Liang, and C. Roberts, “Recursive filtering for communication-based train control systems with packet dropouts,” Neurocomputing, vol. 275, pp. 948–957, 2018.
[15] T. Zhang, “Real-time control method for communication network of high-speed EMU based on T-S fuzzy model,” China Railway Science, vol. 39, no. 3, pp. 93–99, 2018.
[16] Z. Chen, F. Huang, W. Chen et al., “RBFNN-based adaptive sliding mode control design for delayed nonlinear multilateral telerobotic system with cooperative manipulation,” IEEE Transactions on Industrial Informatics, vol. 16, no. 2, pp. 1236–1247, 2020.
[17] Y. Yue, H. Yang, F. Liu, and H. Zang, “Cooperative control for multiple quadropters under position deviations and aerodynamic drag,” Mechanical Systems and Signal Processing, vol. 147, Article ID 107096, 2021.
[18] S. Gao, H. Dong, B. Ning, C. Roberts, and L. Chen, “Neural adaptive coordination control of multiple trains under bidirectional communication topology,” Neural Computing and Applications, vol. 27, no. 8, pp. 2497–2507, 2016.
[19] E. Liu, Y. Yang, and Y. Yan, “Spacecraft attitude tracking for space debris removal using adaptive fuzzy sliding mode control,” Aerospace Science and Technology, vol. 107, Article ID 106310, 2020.
[20] S. Y. Song, J. B. Hu, Y. Y. Wang, and X. L. Han, "Actor-critic learning algorithm for parameter tuning of sliding mode controller,” Electronics Optics & Control, vol. 27, no. 9, pp. 24–27, 2020.
[21] T. Zhang and X. Kong, "Adaptive fault-tolerant sliding mode control for high-speed trains with actuator faults under strong winds,” IEEE Access, vol. 8, pp. 143902–143919, 2020.
[22] B. Jang, Y. K. Wu, N. Y. Lu, and Z. H. Mao, "Review of fault diagnosis and prognosis techniques for high-speed railway traction system,” Control and Decision, vol. 33, no. 5, pp. 841–855, 2018.
[23] D. C. Li, Research on control strategy of high-speed train safety running under crosswind condition, Ph.D. dissertation, Dept. Mechatronic Eng., Lanzhou Jiao Tong Univ., Lanzhou, China, 2019.
[24] P. Deng, L. Zeng, and Y. Liu, "RBF neural network backstepping sliding mode adaptive control for dynamic pressure cylinder electrohydraulic servo pressure system,” Complexity, vol. 2018, Article ID 4159639, 16 pages, 2018.
[25] Y. Y. Min and Y. G. Liu, "Barbalat Lemma and its application in analysis of system stability,” Journal of Shandong University (Engineering Science), vol. 37, no. 1, pp. 51–55, 2007.
[26] X. Long, Z. He, and Z. Wang, "Online optimal control of robotic systems with single critic NN-based reinforcement learning,” Complexity, vol. 2021, Article ID 8893931, 7 pages, 2021.
[27] K. Zhang, H. G. Zhang, Y. L. Cai, and R. Su, "Parallel optimal tracking control schemes for mode-dependent control of coupled markov jump systems via integral RL method,” IEEE Transactions on Automation Ence and Engineering, vol. 17, no. 3, pp. 1332–1342, 2020.