Parameter Self-Tuning of SISO Compact-Form Model-Free Adaptive Controller Based on Long Short-Term Memory Neural Network

YE YANG1, CHEN CHEN1, AND JIANGANG LU1,2
1State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China
2Zhejiang Laboratory, Hangzhou 311121, China
Corresponding author: Jiangang Lu (lujg@zju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61590925 and Grant U1609212, in part by the National Key Research and Development Plan of China under Grant 2017YFC0210403, in part by the Fundamental Research Funds for the Central Universities (Zhejiang University New Generation Industrial Control System (NGICS) Platform), and in part by the Major Scientific Project of Zhejiang Laboratory under Grant 2020MC0AE01.

ABSTRACT Model-free adaptive controller (MFAC) is a novel data-driven control methodology that relies only on input/output (I/O) measurement data instead of classic mathematical models of actual controlled plants. The single-input single-output (SISO) compact-form MFAC (SISO-CFMFAC) is a promising method for controlling SISO nonlinear time-varying systems. The parameters in SISO-CFMFAC must be carefully tuned before use, as inappropriate parameters may lead to poor control performance. However, up to now, parameter tuning has been a time-consuming and laborious task. In this paper, a new approach called SISO-CFMFAC-LSTM is proposed for parameter self-tuning of SISO-CFMFAC based on long short-term memory (LSTM) neural network. To evaluate the performance of the proposed methodology, qualitative and quantitative comparisons with other existing control algorithms are carried out. Six individual performance indices, namely, the root mean square error (RMSE), the integral absolute error (IAE), the integral time-weighted absolute error (ITAE), the integral absolute variation of the control signal (IAVU), the maximum overshoot (MO), and the imprecise control ratio (ICR), are introduced for quantitative comparison. The experimental results demonstrate that the proposed SISO-CFMFAC-LSTM achieves the best performance in all indices, indicating that it is an effective control method for SISO nonlinear time-varying systems.

INDEX TERMS LSTM neural network, model-free adaptive controller, parameter self-tuning, three-tank system.

I. INTRODUCTION

The concept of the state space method proposed by Kalman in 1960 [1], [2] marked the birth of the current control theory and method, which is also called model-based control (MBC). In applications of MBC, the first step is modeling the plant, and then designing the controller based on the plant model obtained using the certainty equivalence principle with the faith that the plant model approaches the true system [3]. After years of development, MBC has achieved theoretical results. Linear–quadratic regulator (LQR) algorithm can obtain the optimal control law of state linear feedback to form a closed-loop optimal control [4]; robust control [5] can achieve robust performance and stability in the presence of bounded modeling errors; Lyapunov-based controller [6] is introduced to help ensure the stability of the adaptive controller; recursive sliding mode control [7] has the advantage of fast response and is sensitive to the physical parameter changes which have been used in practical control scenarios like automotive control [8] and linear motor control [9].

As information technology has developed rapidly, the industrial process has also undergone tremendous changes. It is becoming increasingly difficult to model the mechanism of industrial processes [10] due to the expansion of the production scale and the increasingly complex production process. Production processes generate a large amount of equipment state data, and it is important to design a controller with these data to achieve effective control of the processes. To address the aforementioned challenge, data-driven control (DDC) [11] has become a popular research area.
Various DDC methods, such as the proportional-integral-derivative (PID) control algorithm [12], model-free adaptive controller (MFAC) algorithm [13], iterative feedback tuning (IFT) [14], resilient control [15], adaptive fuzzy output-feedback tracking control [16], can bypass the modeling steps and design the controller by using offline or online input and output data.

The MFAC method [13], which is based only on I/O measurement data, is a type of a DDC method. It was proposed by Hou in 1994 and has undergone subsequent development and improvement. It has now become part of practical control theory and method. MFAC uses a linearization technique, but the linearization error may affect the performance of the data-driven control. Besides, compared to the general discrete-time nonlinear system, MFAC introduces a time-varying pseudo-partial derivative (PPD) [13] or pseudo-gradient (PG) [17] to replace the discrete-time nonlinear system with a series of dynamic linear time-varying models, thus decreasing the above-mentioned error. Therefore, MFAC is gradually used in mechanical systems, industrial production, power systems, and transportation systems [18].

Dynamic linearization methods for MFAC can be divided into three categories: compact-form dynamic linearization (CFDL), partial-form dynamic linearization (PFDL), and full-form dynamic linearization (FFDL) [19]–[21]. Among these three methods, the model structure of the CFDL method is relatively simple. In addition, since PPD is not sensitive to time-varying parameters, time-varying structure and time-varying phase, CFDL has the characteristics of easy operation and strong robustness [13], which is suitable for innovative research. Therefore, CFDL of MFAC used in single-input single-output (SISO) systems called SISO-CFMAC is introduced as the main research object in this paper.

For the controlled object, different parameter settings will lead to different control results, especially when the industrial production status changes. Therefore, parameters in SISO-CFMAC should be carefully tuned before being put into use, as inappropriate parameters may lead to reduced control performance [22]. As a result, parameter self-tuning of the controller has become a popular research area [23]. Zhang and Yang [24] proposed a parameter self-tuning method for PID-based MFAC; however, they only discussed the tuning rules of the parameters and verified them in offline experiments. The IFT method was proposed in [14] but this proposed algorithm has a long calculation time and can easily fall into the local optimum. Jiang and Cong [25] proposed a VRFT-based MFAC parameter tuning method; However, the algorithm is an offline parameter tuning algorithm and requires a large number of computing resources. Chen and Lu [26] proposed a method for achieving MFAC parameter self-tuning online by a back-propagation (BP) neural network; however, it was not compared with other intelligent parameter tuning algorithms. Therefore, the superiority of the algorithm in the evaluation criteria was not demonstrated. Nettari and Harmas [27] proposed a genetic-algorithm-based MFAC parameter tuning method; however, it is also an offline parameter tuning algorithm.

The above-mentioned studies indicate that research on the parameters self-tuning of DDC methods has developed quickly and achieved a few theoretical results. Moreover, neural networks have been introduced to this research area [29]. Zhu and Hou [28] have introduced a basic radial basis function neural network to tune time-varying parameters PPD by using the measured input and output data of the plant. Considering PPD values can be determined by the existing estimation algorithm in MFAC, it can be expected to be meaningful to focus the tuning target that cannot be calculated, which are the tuning objects of the proposed algorithm in this paper.

In practical application scenarios, as the amount of data increases and the model structure becomes more complex, an ordinary neural network cannot quickly adjust controller parameters online because it has difficulty converging [30]. Recurrent neural network (RNN) can be used to address this problem, as it has a stronger information processing ability for these controllers and thus performs better in parameter self-tuning work. However, this approach has not been applied to SISO-CFMAC.

In this work, a parameter self-tuning methodology named SISO-CFMAC-LSTM which is based on the LSTM neural network with the system error set as input is proposed. This method introduces an LSTM neural network [31] to perform parameter self-tuning in SISO-CFMAC to improve control performance.

The contributions of this paper are mainly of the following three-folds:

1) An LSTM neural network is proposed to achieve parameter self-tuning of SISO-CFMAC. As an improved RNN [32], LSTM neural network has three gates for processing information. Each gate has a strong ability to process time-series data feature information, which solves the gradient exploding and vanishing problems in RNNs and improves the robustness of parameter self-tuning.

2) This paper presents qualitative and quantitative comparisons between SISO-CFMAC-LSTM and other intelligent algorithms using different examples. Six evaluation indices are introduced to test the performance of the proposed algorithm, and the results indicate that the proposed method performs best among all tested algorithms.

3) The experimental results demonstrate that the SISO-CFMAC-LSTM algorithm achieves high tracking performance for both SISO discrete nonlinear system and the three-tank water system. Results indicate that the algorithm can realize the online parameter self-tuning of a control model in an actual control setting.

The remainder of the paper is arranged as follows. Section II introduces basic model principles of the SISO-CFMAC control method, while Section III introduces the structure and mathematical principles of the proposed
algorithm, analyzing the mechanism for completing parameters self-tuning work of SISO-CFMFAC. Section IV is the experimental part, this section carries out two typical nonlinear discrete systems to demonstrate the effectiveness and stability of the proposed algorithm. Section V concludes the paper.

II. CONTROL SYSTEM DESIGN

Consider a class of SISO discrete-time nonlinear systems [33]:

\[ y(k + 1) = f \left( y(k), \cdots , y(k - n_y), u(k), \cdots , u(k - n_u) \right) \]  

(1)

where \( y(k) \in R, u(k) \in R \) are defined as the output and input of the system at time \( k \), \( n_y \) and \( n_u \) are two positive integers; \( f \left( \cdots \right) : \mathbb{R}^{n_y + n_u + 2} \rightarrow \mathbb{R} \) is a nonlinear function.

The following assumptions for nonlinear dynamics are given to facilitate our analysis.

**Assumption 1:** In addition to the finite time points, the partial derivatives of the \( f \left( \cdots \right) \) concerning the \((n_y+2)th\) variable are continuous.

**Assumption 2:** Except for finite time points, system (1) satisfies the generalized Lipschitz condition, that is, for any \( k_1 \neq k_2, k_1, k_2 \geq 0 \) and \( u(k_1) \neq u(k_2) \), the inequality \( |y(k_1 + 1) - y(k_2 + 1)| \leq b |u(k_1) - u(k_2)| \) is established, where \( b \) is a constant.

**Remark 1:** From a practical point of view, the above assumptions about the controlled objects are reasonable and acceptable. **Assumption 1** is a typical constraint condition for general nonlinear systems in the design of control systems. **Assumption 2** is a restriction on the upper bound of the system output change rate. From an energy point of view, bounded input energy changes should produce bounded output energy changes within the system. Many practical systems satisfy this assumption, such as the temperature control system, pressure control system, liquid level control system, etc.

When \( u(k_1) \neq 0 \), there exists a time-varying parameter \( \phi_c(k) \in R \) that can transfer system (1) into the following CFDL model:

\[ y(k + 1) = y(k) + \phi_c(k) \Delta u(k) \]  

(2)

where the parameter \( \phi_c(k) \) is called a pseudo partial derivative (PPD) which is a time-varying parameter. Equation (2) is an equivalent dynamic linearization representation of the system (1), it is a linear time-varying model designed for controllers with a simple incremental form, which is in contrast to traditional mechanism models. For the design of the CFDL model, three main research objects are considered: control algorithm, PDD estimation algorithm and control scheme.

A. CONTROL ALGORITHM

For discrete-time systems, a control algorithm that uses minimum system error as a criterion function may produce a steady-state tracking error of the system. This paper considers the following control input criterion function [33]:

\[ J(u(k)) = |y^*(k + 1) - y(k + 1)|^2 + \lambda |u(k) - u(k - 1)|^2 \]  

(3)

where \( \lambda > 0 \) is a weighting factor that limits the change in the control input, and \( y^*(k + 1) \) is the desired output signal. The criterion function \( J \left( \cdots \right) \) consists of two parts, the term \( |y^*(k + 1) - y(k + 1)|^2 \) is introduced to minimize system error and the term \( \lambda |u(k) - u(k - 1)|^2 \) is introduced to prevent excessive changes in control input. By substituting (2) into (3) and deriving the parameter \( u(k) \) and setting it to 0, the following control algorithm can be obtained:

\[ u(k) = u(k - 1) + \frac{\rho \hat{\phi}_c(k)}{\lambda + |\hat{\phi}_c(k)|^2} \left( y^*(k + 1) - y(k) \right) \]  

(4)

where \( \rho \) is an important parameter introduced as a penalty factor for a more general and flexible control rule between 0 and 1. The parameter \( \lambda \) is used to limit the change of the control input \( \Delta u(k) \). It is often applied in control system design, as it guarantees the smoothness of the control input signal.

B. ESTIMATION ALGORITHM

The criterion function of traditional parameter estimation is a two-norm that minimizes the difference between the system model desired output and actual output. However, when applying the parameter estimation algorithm derived from such a criterion function, its parameter estimation may be too sensitive to certain inaccurate sample data caused by interference or sensor failure. In response to this problem, the following PPD estimation criterion function is proposed:

\[ J(\hat{\phi}_c(k)) = |y(k) - y(k - 1) - \phi_c(k) \Delta u(k - 1)|^2 \]  

\[ + \mu \left| \hat{\phi}_c(k) - \hat{\phi}_c(k - 1) \right|^2 \]  

(5)

where \( \mu \) is the weighting factor that limits the variance of the control input \( u(k) \). Calculate the extreme value of \( \phi_c(k) \) in (5), the estimation algorithm for PPD can be obtained as follows:

\[ \hat{\phi}_c(k) = \hat{\phi}_c(k - 1) + \frac{\eta \Delta u(k - 1)}{\mu + \Delta u(k - 1)^2} \times \Delta y(k) \]  

(6)

where \( \eta \in (0, 1] \) is a step size constant added to make (6) general and is used in analytical stability proofs. In addition, \( \hat{\phi}_c(k) \) is the PPD estimated value.

C. CONTROL SCHEME

Combining the above PPD estimation algorithm (6) and the control algorithm (4), the SISO-CFMFAC control method is proposed as follows:

\[ u(k) = u(k - 1) + \frac{\rho \hat{\phi}_c(k)}{\lambda + |\hat{\phi}_c(k)|^2} \left( y^*(k + 1) - y(k) \right) \]  

(7)
\[ \phi_c(k) = \hat{\phi}_c(k - 1) + \frac{\eta \Delta u(k - 1)}{\mu + \Delta u(k - 1)^2} \times (\Delta y(k) - \hat{\phi}_c(k)\Delta u(k - 1)) \]  
(8)

where \( \hat{\phi}_c(k) = \hat{\phi}_c(1), \) if \( |\hat{\phi}_c(k)| \leq \epsilon \) or \( |\Delta u(k - 1)| \leq \epsilon \) or

\[ \text{sign}(\hat{\phi}_c(k)) \neq \text{sign}(\hat{\phi}_c(1)) \]  
(9)

where \( \epsilon \) is a sufficiently small integer, \( \hat{\phi}_c(1) \) is the initial value of \( \hat{\phi}_c(k) \). The algorithm reset mechanism (9) is introduced to enable the PPD algorithm (8) to have a better tracking ability for time-varying parameters. Parameter \( \rho \) and \( \lambda \) in (7) are of great significance in control systems and are the targets of parameter self-tuning described in this study.

The above-mentioned control scheme uses only online I/O measurement data of control object for controller design instead of information about the dynamic model. Since PPD is not sufficiently sensitive to time-varying parameters, time-varying structures, time-varying phases, or even lags [34], SISO-CFMAC has strong adaptability and robustness, which is difficult to achieve for a model-based control method.

III. SISO-CFMAC-LSTM METHODOLOGY

Section II mainly introduces the SISO-CFMAC control scheme for the parameter self-tuning of SISO-CFMAC. Selecting suitable parameters for the model operation requires the experience and knowledge of the relevant industry. Therefore, advanced methods are needed to tune parameters online. In this section, an LSTM neural network is introduced to address this problem with its powerful feature learning ability for time-series data.

A. RECURRENT NEURAL NETWORK

An RNN [35] is a type of neural network proposed by Jordan in 1986. It takes sequence data as input, performs recursively in the direction of the sequence evolution, and all nodes are connected in a chain. Fig. 1 illustrates the RNN network structure.

FIGURE 1. Structure diagram of recurrent neural network.

In Fig. 1, \( x_t \) represents the network input at time \( t \), \( s_t \) is the value of the hidden layer, \( o_t \) is the network output, \( U \) is the weight value of the hidden layer to the output layer, \( V \) is the weight value of the hidden layer to the output layer, and \( W \) is the hidden value of the previous moment. The forward propagation formula for the RNN is as follows:

\[ s_t = f (Ux_t + Ws_{t-1}) \]  
(10)

\[ o_t^{i} = g (Vs_t) \]  
(11)

where \( f (x) \) is the \( \text{tanh} \) activation function, and \( g (x) \) is the \( \text{softmax} \) activation function. The training method of the RNN is the back-propagation through time (BPTT) [36]. However, the gradient vanishing and exploding problems [37] appear in the back-propagation process; thus, an RNN is not suitable for model training with long time-series input.

B. LONG SHORT-TERM MEMORY NEURAL NETWORK

Aiming to solve the gradient vanishing and exploding problems in an RNN, LSTM is a special RNN model proposed by Hochreiter and Schmidhuber in 1997 [38] that improves the structure of an RNN. LSTM solves the gradient problem of an RNN by introducing a gate mechanism. Fig. 2 presents the hidden layer structure of an LSTM neural network.

FIGURE 2. Long short-term memory hidden layer structure diagram.

In Fig. 2, \( u^{(i)} \) is the input of the hidden layer at time \( t \), and \( h^{(i)} \) is the hidden state at time \( t \). The gate mechanism is the key component of LSTM and is a tool for selectively passing information. The LSTM gate mechanism includes three main types: the forget gate, input gate, and output gate.

The forget gate [38] is proposed to determine whether to drop the information of the previous layer with a certain probability. The formula is as follows:

\[ f^{(i)} = \sigma \left( w_{1f} u^{(i)} + w_{2f} h^{(i-1)} + b_f \right) \]  
(12)

where \( w_{1f}, w_{2f} \) are weight coefficients; \( b_f \) is the bias parameter; and \( \sigma \) is the following activation function:

\[ \sigma (z) = \frac{1}{1 + e^{-z}} \]  
(13)

The input gate [38] consists of two parts. The first part introduces the \( \text{sigmoid} \) activation function, where the output is \( i^{(i)} \), determining the values that must be updated. The second part uses the \( \text{tanh} \) activation function, where output is \( a^{(i)} \), constituting a new candidate value vector. The mathematical equations are as follows:

\[ i^{(i)} = \sigma \left( w_{1i} u^{(i)} + w_{2i} h^{(i-1)} + b_i \right) \]  
(14)

\[ a^{(i)} = \text{tanh} \left( w_{1a} u^{(i)} + w_{2a} h^{(i-1)} + b_a \right) \]  
(15)
where \( w_{1i}, w_{2i}, w_{1a}, w_{2a} \) are weight coefficients; \( b_i, b_o \) are bias parameters; and \( \text{tanh} \) is the following activation function:

\[
\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}
\]

The cell state at time \( t \) can be updated by (12), (14), and (15). The mathematical expression is as follows:

\[
e^{(t)} = e^{(t-1)} \odot f^{(t)} + i^{(t)} \odot a^{(t)}
\]

The cell state (17) consists of two parts. The first part represents unnecessary information that should be forgotten, while the second part is the new candidate value that should transmit the next cell state.

The output formula of the output gate is as follows:

\[
\sigma^{(t)} = \sigma(w_{1o}u^{(t)} + w_{2o}h^{(t-1)} + b_o)
\]

\[
h^{(t)} = \sigma^{(t)} \odot \text{tanh}(e^{(t)})
\]

where \( w_{1o}, w_{2o} \) are weight coefficients and \( b_o \) represents the bias parameter.

### C. PARAMETER SELF-TUNING METHOD BASED ON SISO-CFMFAC-LSTM

As mentioned in Section I, parameter self-tuning for a SISO discrete nonlinear system is a difficult and time-consuming process that requires certain expertise [39], thus imposing restrictions on the industrial application of MFAC. As discussed in Section II, the choice of parameters \( \lambda, \rho \) can greatly impact the control performance. In addition, these two parameters change continuously and are subject to external interference in a real-world setting. Therefore, sensitive parameter self-tuning is of great research significance. For parameters \( \lambda, \rho \) in SISO-CFMFAC, an LSTM neural network is proposed to help the SISO-CFMFAC control scheme perform accurate and fast parameter self-tuning.

1) MODEL OVERVIEW

As illustrated in the structure diagram in Fig. 3, this study proposes a new parameter self-tuning method based on the combination of the classical SISO-CFMFAC method and an LSTM neural network. The proposed method takes the systematic error set as input, minimizes the square of the systematic error as the optimization goal, quickly trains through the LSTM neural network, and outputs the final tuned parameters. It then passes these parameters to SISO-CFMFAC to calculate the new system error set. Then, the next neural network training begins. After numerous rounds of LSTM neural network training, online parameter self-tuning of SISO-CFMFAC can be achieved, and the two key parameters \( \lambda, \rho \) can be determined.

2) ANALYSIS OF PROPOSED ALGORITHM

The proposed methodology mainly introduces the LSTM neural network to achieve the self-tuning of SISO-CFMFAC. The implementation steps of the proposed algorithm are as follows:

**Step 1:** Determine the parameters to be tuned in SISO-CFMFAC, initialize the SISO-CFMFAC model, and set the number of nodes in each layer of the LSTM neural network.

**Step 2:** Initialize the weight coefficient to be trained for the input gates, forget gates, cell states, and output gates in
the LSTM hidden layer, and initialize the weight coefficient to be trained in the LSTM output layer.

**Step 3:** Referring to the current time as $k$, the actual system output value $y(k)$ and desired output value $y^d(k)$ are obtained by sampling. Then, calculate the systematic error $e(k)$ at the current time, which is $e(k) = y(k) - y^d(k)$, and determine the input of the LSTM neural network:

$$
x_1 = e(k),
$$
$$
x_2 = e(k) - e(k - 1),
$$
$$
x_3 = \sum_{i=0}^{k} e(k),
$$
$$
X_k = [x_1, x_2, x_3]
$$

(20)

**Step 4:** The outputs of all gates of the LSTM hidden layer are as follows:

$$
net_{hi}(k) = w_{hi}[X_k, h_{k-1}] + b_{hi}
$$
$$
f_i(k) = \sigma(net_{hi}(k))
$$
$$
net_{hi}(k) = w_{hi}[X_k, h_{k-1}] + b_{hi}
$$
$$
I_i(k) = \sigma(net_{hi}(k))
$$
$$
net_{ci}(k) = w_{ci}[X_k, h_{k-1}] + b_{ci}
$$
$$
c_i(k) = c_i(k - 1) \odot f_i(k) + I_i(k) \odot \tilde{c}_i(k)
$$
$$
net_{oi}(k) = w_{oi}[X_k, h_{k-1}] + b_{oi}
$$
$$
o_i(k) = \sigma(net_{oi}(k))
$$
$$
h_i(k) = o_i(k) \odot \tanh(c_i(k)), \quad i = 1, 2, \ldots \text{Istmnum}
$$

(21)\quad (22)\quad (23)\quad (24)\quad (25)\quad (26)

where $w_{hi}, w_{ci}, w_{oi}$ are the weight coefficients of each gate state; $b_{hi}, b_{ci}, b_{oi}$ are the bias factors of each gate state; and \text{Istmnum} is the number of hidden layers.

The output of the LSTM output layer is as follows:

$$
onet_1(k) = w_{yh}h_i(k) + b_{yh}
$$
$$
onet_1(k) = \sigma(onet_1(k))
$$
$$\lambda = out_1(k)
$$
$$\rho = out_2(k)
$$

(27)\quad (28)\quad (29)

where $w_{yh}$ is the weighting factor of the LSTM output and $b_{yh}$ is the bias factor of the LSTM output. Parameters $\lambda$ and $\rho$ are tuned parameters of SISO-CMFAC at time $k$.

**Step 5:** Based on the systematic error $e(k)$ in Step 3 and tuned parameters in Step 4, the control input $u(k)$ can be calculated by (4). Calculate the partial derivative of $u(k)$ with respect to $\lambda$ and $\rho$:

$$
\frac{\partial u(k)}{\partial \lambda} = -\frac{\rho \phi(k) e(k)}{(\lambda + |\phi(k)|)^2}
$$

(30)

$$
\frac{\partial u(k)}{\partial \rho} = \frac{\phi(k) e(k)}{\lambda + |\phi(k)|^2}
$$

(31)

where $\phi(k)$ is the PG estimate.

**Step 6:** Calculate the performance index function as follows:

$$
J = \frac{1}{2} [y^d(k) - y(k)]^2 = \frac{1}{2} \varepsilon^2(k)
$$

(32)

The weight coefficients to be learned in LSTM are updated using a chain-based back propagation algorithm (BPTT) [31]. For the sake of brevity, only the updated formula of the weight matrix is presented below. The updated principle of the bias matrix is the same as the update of the weight matrix.

$$
\frac{\partial J}{\partial w_{hi}} = \frac{\partial J}{\partial y(k+1)} \frac{\partial y(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda} \frac{\partial \lambda}{\partial net_1(k)} \frac{\partial net_1(k)}{\partial h_i(k)} \frac{\partial h_i(k)}{\partial net_{hi}(k)} \frac{\partial net_{hi}(k)}{\partial w_{hi}}
$$

(33)

$$
\frac{\partial J}{\partial w_{ci}} = \frac{\partial J}{\partial y(k+1)} \frac{\partial y(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda} \frac{\partial \lambda}{\partial net_1(k)} \frac{\partial net_1(k)}{\partial h_i(k)} \frac{\partial h_i(k)}{\partial net_{ci}(k)} \frac{\partial net_{ci}(k)}{\partial w_{ci}}
$$

(34)

$$
\frac{\partial J}{\partial w_{oi}} = \frac{\partial J}{\partial y(k+1)} \frac{\partial y(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda} \frac{\partial \lambda}{\partial net_1(k)} \frac{\partial net_1(k)}{\partial h_i(k)} \frac{\partial h_i(k)}{\partial net_{oi}(k)} \frac{\partial net_{oi}(k)}{\partial w_{oi}}
$$

(35)

$$
\frac{\partial J}{\partial w_{yh}} = \frac{\partial J}{\partial y(k+1)} \frac{\partial y(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial \lambda} \frac{\partial \lambda}{\partial net_1(k)} \frac{\partial net_1(k)}{\partial h_i(k)} \frac{\partial h_i(k)}{\partial net_{oi}(k)} \frac{\partial net_{oi}(k)}{\partial w_{oi}}
$$

(36)

where $\eta$ is the learning rate parameter. The most important component of the above weight update is $\frac{\partial u(k)}{\partial \lambda} \frac{\partial \lambda}{\partial net_1(k)}$, which is the partial derivative of $u(k)$ with respect to $\lambda$ and $\rho$, and the formulas are presented in (30) and (31).

**Step 7:** After the control input $u(k)$ acts on the controlled object, the actual value of the system output of the controlled
object at time \( k + 1 \) is obtained. Then, return to Step 3 and repeat Step 3 to Step 7. The parameter self-tuning of SISO-CFMFAC-LSTM at time \( k \) is completed.

IV. SIMULATIONS

A. SISO DISCRETE NONLINEAR SYSTEM SIMULATION

A classical SISO discrete nonlinear system can be described as [40]:

\[
y(k + 1) = \frac{2.5y(k) (y(k) - 1)}{1 + y^2(k) + y^2(k - 1)} + 1.2u(k) \\
+ 0.09u(k) u(k - 1) + 1.6u(k - 2) \\
+ 0.7 \sin(0.5(y(k) + y(k - 1))) \\
\times \cos(0.5(y(k) + y(k - 1)))
\]  

(38)

The desired value of the system output is as follows:

\[
y^*(k + 1) = 5 \sin(k \pi/50) + 2 \cos(k \pi/20)
\]  

(39)

According to the literature, initial values of the system are set to be: \( u(1) = u(2) = u(3) = 0, y(1) = y(2) = 0, \phi(1) = 1 \), the initial value of the controller is set to \( \eta = 0.5, \mu = 1, \varepsilon = 10^{-5} \), the structure of the LSTM neural network is "3-9-2" which means the input layer number is 3, hidden layer number is 9, output layer number is 2, output layer number indicates that the number of parameters to be tuned is two, the learning rate is set to 0.02. The comparison methodology selected in this experiment are SISO-PID [12], SISO-CFMFAC [18], SISO-CFMFAC-BP [26]. All tracking curves of all algorithms are shown in Fig. 4. It should be noted that the curves of SISO-CFMFAC in Fig. 4 and Fig. 5 are almost consistent with the corresponding experimental curve in the cited reference, so it can be used as a benchmark for the comparison of the proposed algorithm.

![FIGURE 4. Tracking performance of each algorithm curve.](image)

According to Fig. 4, it can be found that the tracking effect of the desired output curve optimized by the SISO-CFMFAC-LSTM methodology is the best among all approaches, especially in period 0s to 60s, compared with other curves with obvious oscillation, the algorithm proposed in this paper has better tracking performance of speed and stability. After the 60s, the tracking curve of each algorithm becomes stable except PID algorithm, it can still be seen that the proposed algorithm performs best in tracking by enlarging the picture. Fig. 5 shows the system control input signals optimized by each algorithm.

![FIGURE 5. Each algorithm controller input \( u(k) \).](image)

![FIGURE 6. Parameter self-tuning results of \( \lambda \) (a) and \( \rho \) (b).](image)
values of the proposed method quickly stabilized and are relatively small, so $\phi_e (k)$ can be approximated as a constant. That is, to obtain the smoothness of input, $\rho$ is in the numerator, and $\lambda$ is in the denominator, they will reach a compromise and inhibit each other, so as to obtain a more robust control performance. Therefore, $\lambda$ and $\rho$ have a certain degree of similarity in this method.

From all figures mentioned above, the proposed algorithm can adjust parameters quickly and reach perfect tracking performance in this SISO discrete nonlinear system simulation. Fig. 7 demonstrates the PPD value curves of SISO-CFMFAC, SISO-CMFAC-BP and SISO-CMFAC-LSTM. In the first 60s, the dynamics of SISO-CFMFAC’s PPD are more complicated, which results in the projection estimation algorithm not being able to track its true value well, so it shows the large fluctuation in Fig. 4. Compared to SISO-CFMFAC, the PPD estimated value curves of SISO-CMFAC-BP and SISO-CMFAC-LSTM are much smoother, and SISO-CMFAC-LSTM performs best among them. After the 60s, three PPD estimated value curves all tend to be stable and gradually close to each other which can explain the similar tracking performance of the above-mentioned three algorithms in Fig. 4.

### B. THREE-TANK SYSTEM SIMULATION

The three-tank system [41] is a classic control plant, the schematic diagram of the three-tank system is shown in Fig. 8. The three-tank system belongs to the controlled object in the liquid level control system. Its output $Y$ (cm) is the level of the third tank, and the control input $U$ is the flow opening (%) into the tank. In simple terms, the working principle of this system is: when the liquid level rises to a certain high and low pressure, the outflow is increased to be equal to the inflow, to re-establish a balanced relationship, the liquid level finally stabilized at a certain height.

According to the literature, the transfer function of the liquid level and control input in the simple three-tank system is calculated as follows:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{Ke^{-\tau s}}{(T_1 s + 1)(T_2 s + 1)(T_3 s + 1)}$$  \hspace{1cm} (41)

where $K$ is system gain, $\tau$ is delay factor, $T_1$, $T_2$, $T_3$ are time constants. In this test, the following parameters are selected:

$$[K \tau T_1 T_2 T_3] = \begin{cases} 
[4.5 \ 24 \ 8 \ 8 \ 8], & k < 400 \\
[5 \ 24 \ 8 \ 8 \ 8], & 400 \leq k < 800 \\
[5 \ 40 \ 6 \ 6 \ 6], & 800 \leq k < 1000 
\end{cases}$$  \hspace{1cm} (42)

Calculate the following SISO nonlinear system based on the transfer function and parameters:

$$y(k + 1) = 2.6475 y(k) - 2.3364 y(k - 1) + 0.6873 y(k - 2) + 0.001334 u(k - 24) + 0.00486 u(k - 25) + 0.001106 u(k - 26), \quad k < 400$$

$$y(k) - 2.3364 y(k - 1) + 0.6873 y(k - 2) + 0.001482 u(k - 24) + 0.0054 u(k - 25) + 0.001229 u(k - 26), \quad 400 \leq k < 800$$

$$2.5394 y(k) - 2.1646 y(k - 1) + 0.6065 y(k - 2) + 0.003406 u(k - 40) + 0.01203 u(k - 41) + 0.0026 u(k - 42), \quad 800 \leq k < 1000$$  \hspace{1cm} (43)

The desired value of the system output is as follows:

$$y^*(k) = 10$$  \hspace{1cm} (44)

Initial values of the system are set to be: $u(1) = u(2) = 0$, $y(1) = 0$, $y(2) = 0$, $\phi(1) = 1$, the initial value of the controller is set to $\eta = \mu = 1$, $\epsilon = 10^{-5}$, the structure of the LSTM neural network is 3-7-2, the learning rate is 0.01. Similar to the SISO discrete nonlinear system simulation, the comparison algorithms selected in this experiment are SISO-PID [12], SISO-CMFAC [18], SISO-CMFAC-BP [26]. It should be noted that the curves of SISO-CMFAC in the Fig. 9 and Fig. 10 are highly consistent with the corresponding experimental curve in the cited reference which can be used as a benchmark to reflect the advantages of the proposed algorithm.

The tracking curves of all algorithms are shown in Fig. 9, the proposed methodology performs better than other algorithms, which is most obvious in the period between...
FIGURE 9. Tracking effect of each algorithm curve.

FIGURE 10. Each algorithm controller input $u(k)$.

0s to 400s. Especially starting from 250s, the tracking curve of the proposed algorithm has tended to be stabilized quickly, while other algorithms still have large oscillations and over- shoot. In the remaining periods, the proposed algorithm also performs best in all algorithms. Fig. 10 shows the system control input signals optimized by each algorithm.

Fig. 11 shows parameter self-tuning results of the proposed method, similar to SISO discrete nonlinear system simulation, the proposed algorithm can tune parameters sensitively in this experiment. The changes of these two tuned parameters curves have a certain degree of similarity, the reason for this phenomenon is similar to that in the SISO discrete nonlinear system simulation. Fig. 12 demonstrates the PPD value curves of SISO-CFMFAC, SISO-CFMFAC-BP and SISO-CFMFAC-LSTM. It can be easily found that the PPD estimated value curve of the proposed algorithm tends to be stable after the 40s while the other two curves are still fluctuating, proving that the proposed algorithm can achieve better tracking performance.

C. SIMULATION RESULTS AND ANALYSIS

To further analyze the performance of SISO-LSTMSE, six individual performance indices are used for quantitative comparison. The first three indices are the root mean square error ($RMSE$), the integral absolute error ($IAE$), and the integral time-weighted absolute error ($ITAE$). These indices are used to evaluate the tracking accuracy of each approach, while the integral absolute variation of the control signal ($IAVU$) is used to evaluate the stability of the control input. These four indices are expressed in (45) – (48) as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^{N} e(k)^2}$$  (45)
TABLE 1. Evaluation results of each algorithms in SISO discrete nonlinear system simulation.

| Algorithm          | RMSE  | IAE   | ITAE  | IAVU  | MO    | ICR(0.1) |
|--------------------|-------|-------|-------|-------|-------|----------|
| PID [12]           | 3.812 | 526.086 | 75831.938 | 69.755 | 9.602 | 0.960    |
| SISO-CFMFAC [18]   | 4.402 | 395.882 | 20877.901 | 307.600 | 23.043 | 0.895    |
| SISO-CFMFAC-BP [26]| 0.955 | 153.182 | 13314.346 | 94.974 | 3.095 | 0.920    |
| SISO-CFMFAC-LSTM   | 0.699 | 110.393 | 12187.146 | 49.748 | 2.000 | 0.890    |

TABLE 2. Evaluation results of each algorithms in three-tank system simulation.

| Algorithm          | RMSE  | IAE   | ITAE  | IAVU  | MO    | ICR(0.1) |
|--------------------|-------|-------|-------|-------|-------|----------|
| PID [12]           | 2.754 | 1398.004 | 234679.631 | 29.815 | 4.885 | 0.869    |
| SISO-CFMFAC [18]   | 2.689 | 1232.765 | 195502.149 | 20.966 | 2.616 | 0.801    |
| SISO-CFMFAC-BP [26]| 2.689 | 1204.472 | 185613.579 | 19.717 | 2.147 | 0.747    |
| SISO-CFMFAC-LSTM   | 2.570 | 1032.047 | 134529.583 | 16.296 | 1.248 | 0.643    |

\[
IAE = \int_{0}^{t} |e_i(t)|dt \quad (46)
\]

\[
ITAE = \int_{t_c + t_s}^{t_c} (t - t_c) |e_i(t)|dt \quad (47)
\]

\[
IAVU = \int_{0}^{t} \left| \frac{du(t)}{dt} \right| dt \quad (48)
\]

In addition to the four indices mentioned above, this paper proposes two other indices to evaluate the algorithm. The first is the maximum overshoot (MO), which is proposed to evaluate the tracking instability, and the second is the imprecise control ratio (ICR), which is used to calculate the time proportion of imprecise control. These indices are expressed in (49) and (50) as follows:

\[
MO = \max \left\{ (y(1) - y^*(1)), \ldots, (y(N) - y^*(N)) \right\} \quad (49)
\]

\[
ICR(\xi) = \frac{1}{N} \sum_{k=1}^{N} IC(k, \xi) \quad (50)
\]

where

\[
IC(k, \xi) = \begin{cases} 
0 & \text{when } |y(k) - y^*(k)| < \xi \\
1 & \text{when } |y(k) - y^*(k)| \geq \xi 
\end{cases} \quad (51)
\]

The performance evaluation results of each algorithm in the experiment are presented in Tables 1 and 2. It can be seen that SISO-CFMFAC-LSTM performs best in all indices in both experiments. Especially in comparison with the original SISO-CFMFAC, the proposed algorithm exhibits significant improvement in various indices.

According to the evaluation results of two simulations as listed in Table 1 and Table 2, the proposed algorithm in this paper achieves the best performance on all six indices.

For results of SISO discrete nonlinear system simulation as listed in Table 1, since parameters are difficult to set to optimal values, the tracking curve of SISO-CFMFAC fluctuates greatly at the beginning, which can be found in Fig. 4, resulting in its evaluation results not better than PID. In reference [17], SISO-CFMFAC-BP, which introduces BP neural networks to carry out the parameters self-tuning of SISO-CFMFAC, has reduced the ITAE index by 36.23% compared to SISO-CFMFAC. Furthermore, our proposed SISO-CFMFAC-LSTM reduces the ITAE index by 8.46% compared to SISO-CFMFAC-BP. The reason is that LSTM neural networks can deal with time-series data in nonlinear systems with its gate mechanism while BP neural networks are just feedforward networks that fail to capture the features of time-series data.

In the three-tank water system simulation as listed in Table 2, SISO-CFMFAC with appropriate controller parameters is better than PID. Meanwhile, SISO-CFMFAC-BP reduced the ITAE index by 5.06% compared to SISO-CFMFAC. Furthermore, the proposed algorithm reduces the ITAE index by 27.51% compared to SISO-CFMFAC-BP. These results prove the effectiveness and superiority of our proposed SISO-CFMFAC-LSTM.

V. CONCLUSION

In this paper, an optimization algorithm called SISO-CFMFAC-LSTM is proposed for SISO discrete-time nonlinear systems. An LSTM neural network takes the system error of the SISO-CFMFAC as input and begins forward calculation and backward propagation to help the SISO-CFMFAC control scheme perform parameter self-tuning to improve the control performance. This method utilizes the time-series data analysis capabilities of LSTM with its gate mechanism, which avoids the gradient vanishing and exploding problems present in an RNN. Experiments are conducted to compare the proposed algorithm with other algorithms. The experimental results indicate that SISO-CFMFAC-LSTM outperforms other algorithms, and its tracking performance is the best among all algorithms. The proposed algorithm thus has advantages in stability and accuracy.

The algorithm proposed in this paper still has room for improvement, it only validates its advantages in SISO discrete nonlinear system simulation and classical practical industrial scene like the three-tank system. However, it has not been evaluated in the actual industrial scene with some factors such as measurement noise and control saturation. Therefore, in future work, actual applications of the proposed algorithm...
will be studied, such as the chemical production process that is different from the experiments conducted in this paper. Studying these factors in the actual industrial scene is an important research area from an engineering perspective. Besides, the combination of PFDFL-MFAC or FFDFL-MFAC and LSTM neural networks is also worth studying. As mentioned in the first section, PFDFL and FFDFL method have more complex structures and more parameters, so they can better capture the nonlinear features in the control systems. Therefore, introducing LSTM neural networks into these two methods is of great research significance.

REFERENCES

[1] R. E. Kalman, “On the general theory of control systems,” IRE Trans. Autom. Control, vol. 4, no. 3, p. 110, Dec. 1959.
[2] R. Kalman and J. Bertram, “General approach to control theory based on the methods of Lyapunov,” IRE Trans. Autom. Control, vol. 4, no. 3, p. 20, Dec. 1959.
[3] B. Gough, B. Wilson, and D. Matovich, “Advanced model-based control for continuous process industries,” in Proc. IEEE Ind. Appl. Dyn. Modeling Control Appl. Ind. Workshop, Vancouver, BC, Canada, Apr./May 1998, pp. 33–39, doi: 10.1109/DMA.1998.703483.
[4] U. Desai, S. Banerjee, and S. Kiaei, “A canonical correlation analysis approach to (i) reduced-order LQR design, and (ii) characterization of strong and weak interactions in large scale interconnected systems,” in Proc. 23rd IEEE Conf. Decis. Control, Las Vegas, NV, USA, Dec. 1984, pp. 1523–1528, doi: 10.1109/CDC.1984.272337.
[5] C. C. Blackwell, “Lyapunov function gradient generated robust control in the absence of the nominal stabilizing control,” in Proc. Amer. Control Conf., Minneapolis, MN, USA, 1987, pp. 1595–1600.
[6] Q. M. Lam and R. N. Morgan, “Spacecraft control law design using Lyapunov-based adaptive controllers,” in Proc. 1st IEEE Conf. Ctrl Appl., Dayton, OH, USA, vol. 1, Sep. 1992, pp. 348–350, doi: 10.1109/CACC.1992.269851.
[7] K. Shao, J. Zheng, H. Wang, F. Xu, X. Wang, and B. Liang, “ Recursive sliding mode control with adaptive disturbance observer for a linear motor positioner,” Mech. Syst. Signal Process., vol. 146, Jan. 2021, Art. no. 107014.
[8] B. Deng, K. Shao, and H. Zhao, “Adaptive second order recursive terminal sliding mode control for a four-wheel independent steer-by-wire system,” IEEE Access, vol. 8, pp. 75936–75945, 2020, doi: 10.1109/ACCESS.2020.2989326.
[9] K. Shao, J. Zheng, K. Huang, H. Wang, Z. Man, and M. Fu, “Finite-time control of a linear motor positioner using adaptive recursive terminal sliding mode,” IEEE Trans. Ind. Electron., vol. 67, no. 8, pp. 6659–6668, Aug. 2020, doi: 10.1109/TIE.2019.2970070.
[10] M. A. Johnson and M. J. Grimble, “ Control for competitive and economic industrial process operations,” in Proc. 35th IEEE Conf. Decis. Control, Kobe, Japan, vol. 4, Dec. 1996, pp. 4508–4509, doi: 10.1109/CDC.1996.577572.
[11] Z. S. Hou and J. X. Xu, “On data-driven control theory: The state of the art and perspective,” Zidonghua Xuebao/Acta Automat. Sinica, vol. 35, no. 6, 2009, pp. 630–667, 2009.
[12] K. H. Ang, G. Chong, and Y. Li, “PID control system analysis, design, and technology,” IEEE Trans. Control Syst. Technol., vol. 13, no. 4, pp. 559–576, Jul. 2005.
[13] J. Shangtai, H. Zhongsheng, and W. Weihong, “Model-free adaptive control used in permanent magnet linear motor,” in Proc. Chin. Control Conf., Hunan, China, Jul. 2006, pp. 748–751.
[14] S. Abedini and H. Zarrabipour, “Tuning of an optimal PID controller with iterative feedback tuning method for DC motor,” in Proc. 2nd Int. Conf. Control, Instrum. Automat., Shiraz, Iran, Dec. 2011, pp. 611–615.
[15] C. Riegler, Q. Zhu, and T. Basar, “Agent-based cyber control strategy design for resilient control systems: Concepts, architecture and methodologies,” in Proc. 5th Int. Symp. Resilient Control Syst., Salt Lake City, UT, USA, Aug. 2012, pp. 40–47, doi: 10.1109/ISRCS.2012.6309291.
[16] S. Tong and Y. Shi, “Adaptive fuzzy output feedback control for nonlinear MIMO systems,” in Proc. IEEE Int. Conf. Fuzzy Syst., Budapest, Hungary, vol. 3, Jul. 2004, pp. 1209–1213, doi: 10.1109/FUZZY.2004.1375536.
[39] C. Liu, Y. Chu, L. Wang, and Y. Zhang, “Application and the parameter tuning of ADRC based on BFO-PSO algorithm,” in Proc. 25th Chin. Control Decis. Conf. (CCDC), Guiyang, China, May 2013, pp. 3099–3102.

[40] Z. S. Hou and S. T. Jin, Model Free Adaptive Controller: Theory and Applications. Boca Raton, FL, USA: CRC Press, 2013.

[41] F. Tahir, N. Iqbal, and G. Mustafa, “Control of a nonlinear coupled three tank system using feedback linearization,” in Proc. 3rd Int. Conf. Electr. Eng., Lahore, Pakistan, 2009, pp. 1–6.

Ye Yang received the B.S. degree from the Bell Honors School, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2018. He is currently pursuing the Ph.D. degree with the State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Zhejiang, China. His current research interests include artificial intelligence in industries, intelligent modeling, intelligent control, and industrial big data.

Chen Chen received the B.S. degree from the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2018. She is currently pursuing the Ph.D. degree with the State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Zhejiang, China.

Her current research interests include data-driven control, intelligent control, and their industrial applications.

Jiangang Lu received the B.S. and Ph.D. degrees from Zhejiang University, China, in 1989 and 1995, respectively.

He is currently a Full Professor with the Institute of Industrial Process Control. He is also with the College of Control Science and Engineering, Zhejiang University. He has published one book and more than 120 papers in journals and conferences. His research interests include artificial intelligence in industries, intelligent sense, intelligent modeling, intelligent control, intelligent optimization, and industrial big data.

He is a member of the Technical Committee on Process Control in Chinese Association of Automation. He has received the First Prize of China’s National Science and Technology Progress Award, in 2013, the Geneva International Invention Gold Medal Award, in 2013, and the Second Prizes of Zhejiang Province Science and Technology Progress Award, in 1999 and 2007. He is the Standing Director of the Zhejiang Association of Automation.