Taylor & Francis
Taylor & Francis Group

Automatika

Journal for Control, Measurement, Electronics, Computing and Communications

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/taut20

Research on parallel control of CMAC and PD based on U model

Fengxia Xu, Junhua Xu, Jiaqi Zhang, Chunda Zhang & Zifei Wang

To cite this article: Fengxia Xu, Junhua Xu, Jiaqi Zhang, Chunda Zhang & Zifei Wang (2021) Research on parallel control of CMAC and PD based on U model, Automatika, 62:3-4, 331-338, DOI: 10.1080/00051144.2021.1954782

To link to this article: https://doi.org/10.1080/00051144.2021.1954782

Published online: 23 Jul 2021.

AUTOMATIKA
2021, VOL. 62, NOS. 3–4, 331–338
https://doi.org/10.1080/00051144.2021.1954782

OPEN ACCESS

Research on parallel control of CMAC and PD based on U model

Fengxia Xu[a], Junhua Xu[b], Jiaqi Zhang[b], Chunda Zhang[b] and Zifei Wang[b]

[a]College of Mechanical and Electrical Engineering, Qiqihar University, Qiqihar, People's Republic of China; [b]College of Computer and Control Engineering, Qiqihar University, Qiqihar, People's Republic of China

ABSTRACT

In this paper, the nonlinear U model with time-varying coefficients is investigated and the transformation of the nonlinear model is accomplished by the Newton iterative algorithm. Based on the nonlinear U model, a control algorithm with cerebellar model articulation controller and proportional derivative (PD) in parallel is proposed. The algorithm learns online through a neural network while optimizing the output of the PD, which ultimately enables the actual output of the system to track up to the desired output. Considering that the nonlinear object has the characteristic of rapid change with time, the article improves the PD algorithm to nonlinear PD control algorithm to complete the design of the system. The algorithm automatically adjusts the weights according to the error magnitude to complete the controller parameter adjustment, thus reducing the error of the system. The simulation results show that the nonlinear PD algorithm is better than the PD algorithm, meanwhile, the tracking speed and control precision of the system are improved.

ARTICLE HISTORY
Received 15 December 2020
Accepted 2 July 2021

KEYWORDS
Nonlinear U model; Newton iteration; CMAC neural network; nonlinear PD

1. Introduction

Non-linear characteristics are prevalent in the actual production process and have been studied by a large number of scholars, who have proposed research methods such as point linearization, segmental linearization, inverse step feedback linearization and feedback input–output linearization. Segment linearization means dividing the input/output curve of a nonlinear system into several segments, replacing the corresponding segments with linear segments, and then approximating the corresponding linear segments with a linear system model [1]. Inverse feedback linearization is a composite design that combines controller inverse step design with feedback linearization [2]. The input–output linearization is defined based on the order of the nonlinear system, and the minimum phase nonlinear system is linearized using an output feedback and coordinate transformation algorithm to complete the feedback input–output linearization [3]. Neural networks have powerful learning ability and fault tolerance, and achieve an infinite approximation of controlled objects, and their learning algorithms are simple and easy for computer implementation, which is a common modelling method. A neural network linearization design has been proposed, using neural networks to compensate for the errors produced by linearization [4,5]. However, some existing methods for designing nonlinear systems have certain limitations, as well as modelling errors, poor generalizability, and complex algorithms. Thus, the nonlinear U model was developed in this context, which is a polynomial with time-varying parameters and represents a large class of smooth nonlinear systems with high generality, high accuracy and ease of controller design [6,7].

Based on the biological finding that the cerebellum makes reflexive responses without thinking when controlling limb movements, scholars have proposed a new type of neural network, named the cerebellar model neural network, or cerebellar model articulation controller (CMAC) network for short. CMAC networks have strong associative capabilities, and the output of the network is determined by only a small number of neurons corresponding to the network weights, the input and output of a CMAC network appears to be a linear relationship. Such characteristics make CMAC have fast learning ability, strong fault tolerance, fast convergence speed, no local minimal problems and other characteristics, so it is widely used in robot arm control, adaptive control, robot control, pattern recognition, signal processing and other fields. Adaptive fuzzy CMAC neural network controller for pneumatic artificial muscle-driven spring mass position control system [8]. In order to solve the control problem of uncertain nonlinear systems and the problem of system mixed interference upper bound that is difficult to measure in practical applications, a recursive CMAC neural network model decomposition control algorithm is proposed [9]. In order to compensate the hysteresis nonlinearity inherent in the telescopic actuator caused by the super magnetism and to

CONTACT Fengxia Xu ✉ xufengxia_hit@163.com College of Mechanical and Electrical Engineering, Qiqihar University, Qiqihar 161006, People's Republic of China

improve its accuracy, a real-time hysteresis compensation control strategy is proposed by combining CMAC neural network and proportional integral derivative (PID) control to achieve high-precision tracking control [10]. In view of the nonlinear, large inertia and time-varying characteristics of the temperature control system of central air-conditioning room, the composite control of CMAC neural network and single neuron PID was proposed [11]. A nonlinear quantization-based CMAC neural network algorithm is proposed, which adaptively designs the conceptual mapping of CMAC and improves the computational speed and accuracy of CMAC to meet the needs of nonlinear real-time control in complex dynamic environments [12]. To address the nonlinearity and various uncertainty factors in high-precision servo systems, the fast learning of PD + feed-forward control + CMAC neural network algorithm is proposed, which ensures fast real-time tracking and further improves tracking accuracy [13]. Taking electro-hydraulic servo system as the control object, the control strategy of combining CMAC network and PID controller is discussed. The CMAC neural network is used as a feedforward controller to achieve the inverse dynamic control [14]. To address the problem of poor control effect of temperature control system in metal heat treatment process, a compound control algorithm based on CMAC and PD is proposed to realize the fast tuning and self-learning function of PID parameters [15].

Since the nonlinear objects are not approximated by the U-model, model accuracy is guaranteed. At the same time, considering the polynomial expression of the model, the Newtonian iterative algorithm is used to transform the model, after which the nonlinear objects can easily be used to complete the system design using linear system design [16]. Based on these characteristics, the U model is widely used. An internal mode control technique based on the U-model is proposed for the adaptive control of nonlinear dynamic objects such as DC motors [17]. Hasan et al. designed an adaptive controller based on the U-model for a gas equipment control device and demonstrated that the U-model outperformed other models by comparing it with other models [18]. An adaptive U-model online airflow identification device is proposed to achieve adaptive online tuning of system parameters [19]. The U model is applied to an electrically excited synchronous motor, and an anti-interference controller is designed on the basis of the U model [20]. In order to improve the robot's trajectory tracking speed, a U-model-based trajectory tracking control method is proposed, which also alleviates the requirement of dynamic mathematical model and simplifies the design of the robot trajectory tracking controller [21]. A new U-model-based sliding mode augmented control method for controlling a class of single input single output (SISO) dynamic systems with internal uncertain parameters and external system control noise/disturbance is proposed [22]. Presents a weighted multiple U-model control scheme to address the control problem of some classes of discrete-time nonlinear systems with large parameter uncertainties including parameter jumps [23].

The article proposes a design scheme for the parallel control of CMAC neural networks and traditional PD to achieve basic control of nonlinear U-model objects. In order to improve the speed and accuracy of nonlinear system control, on the basis of conventional PD, a parallel control design of CMAC neural network and nonlinear PD is proposed to complete the adjustment of system performance, we can learn quickly for time-varying objects and achieve system convergence through algorithmic adjustment.

2. Nonlinear U-model

In 2002, the concept of U-model was first explicitly proposed by Prof. Zhu Quanmin in the literature [7], using a simple generic mapping to convert almost all smooth nonlinear discrete systems into mathematical expressions that can be used in linear control system design methods. The model expression is as follows:

$$y(t) = \sum_{j=0}^{M} a_j(t)u^j(t-1) + e_j(t) \tag{1}$$

Here, $y(t)$ represents the actual output of the nonlinear system, j is the order of the input system, u represents the actual control input of the nonlinear system. M represents the nonlinear controlled object order, and n represents the maximum delay of input and output, $a_j(t)$ represents the time-varying coefficient of the nonlinear system as a function between $y(t-1), y(t-2), \cdots, y(t-n), u(t-2), \cdots, u(t-n)$ and $e_j(t-1), e_j(t-2), \cdots, e_j(t-n)$. $e_j(t)$ represents the error caused by the uncertainties of the nonlinear system, such as modelling, external interference, etc. Further transform the model of the nonlinear system, assuming

$$U(t) = y(t) \tag{2}$$

$$U(t) = \sum_{j=0}^{M} a_j(t)u^j(t-1) + e_j(t) \tag{3}$$

The expression (3) is generally regarded as a nonlinear U model. In designing the controller, $U(t)$ in (3) can be designed using linear control, then solved for $u(t-1)$ using Newton's iterative solution equation, and finally input into the actual object to get the actual output.

In Figure 1, $y_m(t)$ represents the given signal of the system, compare it with the actual output $y(t)$, and the difference obtained is used as the input of the controller. The control variables of the system are obtained by the controller, and the approximate value of the last time is obtained by Newton iteration, which acts on the controlled object and finally outputs $y(t)$. Model transformation is the part from Newton iteration to controlled object output.

It can be seen that Newton iteration plays an important role in the nonlinear U model control system.

The Newton iteration algorithm is:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{4}$$

suppose x is the root of $f(x) = 0$, then x_{n+1} is called an approximate solution to the $n + 1$ degree of x, $f(x_n)$ is the value of the function at x_n, $f'(x_n)$ is the partial derivative at x_n.

In the article:

$$\begin{aligned} u(t-1)_k &= u(t-1)_{k-1} - \frac{y(t) - U(t)}{d[y(t)]/du(t-1)}. \\ &= u(t-1)_{k-1} \\ &\quad - \left. \frac{\sum_{j=0}^{M} a_j(t)u^j(t-1) - U(t)}{\frac{d\left[\sum_{j=0}^{M} a_j(t)u^j(t-1)\right]}{du(t-1)}} \right|_{u^j(t-1)=u_k^j(t-1)} \end{aligned} \tag{5}$$

k represents the number of iterations.

Compared with other models, nonlinear U model has many advantages: First of all, the U-model has high universality; Secondly, the application of Newton iteration can apply the design scheme of linear control to the design of nonlinear control, providing a good theoretical support for the large-scale application of U model. The transformation of the nonlinear U model does not lose any information of the original controlled object, so the high accuracy of the model is guaranteed (Figure 1).

3. CMAC neural network

CMAC neural network consists of three layers: Input layer, middle layer and output layer. Nonlinear mapping technology is used between the input layer and the middle layer, and linear mapping technology is used between the middle layer and the output layer. The nonlinear relationship between input and output of CMAC network is realized by two basic mappings: conceptual mapping and practical mapping. This paper adopts SISO CMAC network. The structural diagram of CMAC neural network is shown in Figure 2.

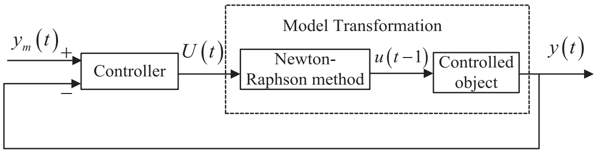

Figure 1. Standard control design drawing for nonlinear U model.

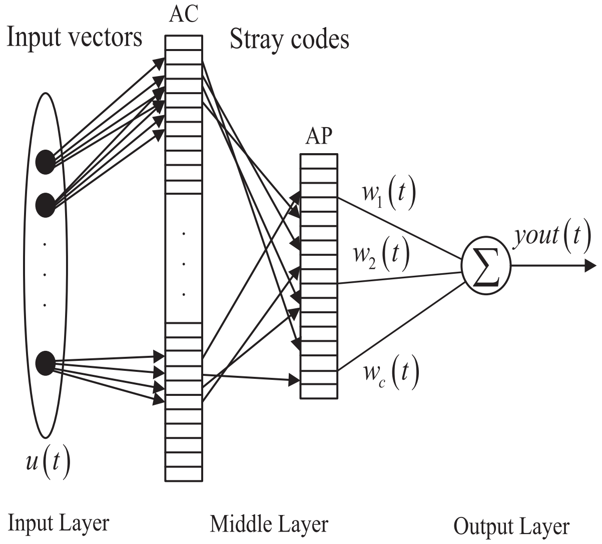

Figure 2. Structural diagram of cerebellar model neural network.

Concept mapping is a mapping from input space to concept memory AC. Assuming that the input space vector is ${u_n}^T$, n is the number of quantifications, and the input space is mapped to c storage units in AC, and $c << n$. The mapped vector $R_n = [s_1(u_n), s_2(u_n), \cdots, s_c(u_n)]^T$. The mapping principle is that in the input space, two adjacent input data have partially identical units activated at AC (same output). The greater the degree of adjacent, the more identical units; the smaller the degree of adjacent, the fewer identical units. This mapping is often called local generalization, where c is the generalization parameter. Actual mapping is the mapping from the c-cells in A to the c-cells in AP, which is the actual memory, using encoder technology. In each of these c units, corresponding weighted values are stored, and the actual output of the network is finally obtained through the sum of these c weighted values.

The algorithm of CMAC network is as follows:

$$y = \sum_{j=1}^{c} w_j(t)s_j(u_n) \tag{6}$$

$w_j(t)$ is the weighted value of the neural network.

The indicators of weight adjustment is:

$$J = \frac{1}{2c}e(t)^2 \tag{7}$$

In(7), $e(t) = rin(t) - yout(t)$, $rin(t)$ is the mentor signal, which is the ideal output value.

Adopt the gradient descent method to adjust the weights online:

$$\Delta w_j(t) = -\eta \frac{\partial J}{\partial w_j(t)}$$

$$= -\eta \frac{rin(t) - yout(t)}{c} \cdot \frac{\partial yout(t)}{\partial w_j(t)}$$

$$= \eta \frac{e(t)}{c} \tag{8}$$

$$w_j(t) = w_j(t-1) + \Delta w_j(t) + \alpha(w_j(t-1) - w_j(t-2)) \tag{9}$$

In (9), α is the inertia coefficient.

CMAC network has many advantages in practical applications: First, it is a neural network based on local learning, with a small number of weights adjusted each time and a fast learning speed, which is suitable for real-time control; second, CMAC has a strong generalization capability, that is, similar input produces similar output, different input gives different output; third, as a nonlinear approximator, it is insensitive to the order in which the learned data appear; fourth, it has the ability of continuous (analog) input and output; fifth, address programming is adopted. When using serial computer simulation, it can speed up the response speed. Based on the above advantages, CMAC has better nonlinear approximation ability than general neural networks. It is more suitable for the nonlinear real-time control in complex dynamic environment.

4. Parallel controller design

4.1. Parallel control design of CMAC neural network and PD controller

Considering the universality of the controller design, conventional PD is selected as the controller of the system, and CMAC neural network is adopted as the feedforward controller. The parallel composite control algorithm of neural network and PD is designed to improve the stability and anti-interference ability of the control system. The CMAC neural network here is SISO, and its structure is shown in Figure 3.

The controller is designed to take the given signal $r(t)$ of the system as the input value of the CMAC neural network, and adjust the weight value $w_j(t)$ by the difference between the actual output value of the neural network and the control quantity $U(t)$ of the system. In addition, PD control can optimize the neural network learning process according to the dynamic characteristics of the system and improve the stability and interference resistance of the controller. At the end of each control process, the output $u_2(t)$ of CMAC neural network is compared with the system control quantity $U(t)$ to correct the weight and enter the learning stage of neural network. The purpose of learning is to minimize the difference between the system control quantity and the output of the neural network, that is, the system control quantity is generated by the neural network controller. The output of CMAC neural network is:

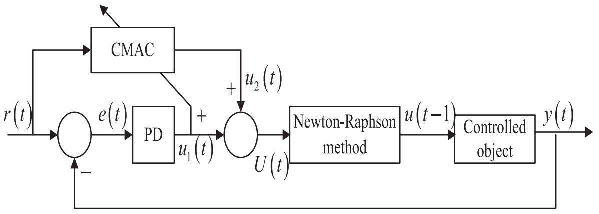

Figure 3. CMAC and PD parallel control system structure diagram.

$$u_2(\mathrm{t}) = \sum_{j=1}^{c} w_j(t) \tag{10}$$

PD output:

$$u_1(t) = k_p e(t) + k_d e(t) \tag{11}$$

System control quantity:

$$U(t) = u_1(t) + u_2(t) \tag{12}$$

The performance index function of the network is:

$$J = \frac{1}{2c}(U(t) - u_2(t))^2 = \frac{1}{2c}{u_1}^2 \tag{13}$$

According to the gradient descent method, the partial derivative of $w_j(t)$ is obtained as follows:

$$\Delta w_j(t) = -\eta \frac{\partial J}{\partial w_j(t)} = \eta \frac{U(t) - u_2(t)}{c} = \eta \frac{u_1(t)}{c} \tag{14}$$

η is the learning rate of the network, which is the normal number. The iterative algorithm of output weight is as follows:

$$w_j(t) = w_j(t-1) + \Delta w_j(t) + \alpha(w_j(t-1) - w_j(t-2)) \tag{15}$$

In (15), α is the coefficient of inertia and is the normal number.

4.2. Parallel control design of neural network and nonlinear PD controller

The expression form of the nonlinear U model established in this paper is time-varying polynomial, which needs to quickly adjust the characteristics of the controller. The coefficient learning of neural networks using conventional PD has some limitations in terms of speed and is not ideal for the control of nonlinear systems and time-varying parameter perturbations. Therefore, a parallel control design of CMAC neural network and nonlinear PD is proposed, in which the time-varying objects are quickly learned and the system converges through algorithmic adjustment. The control system adopts the control structure as shown in

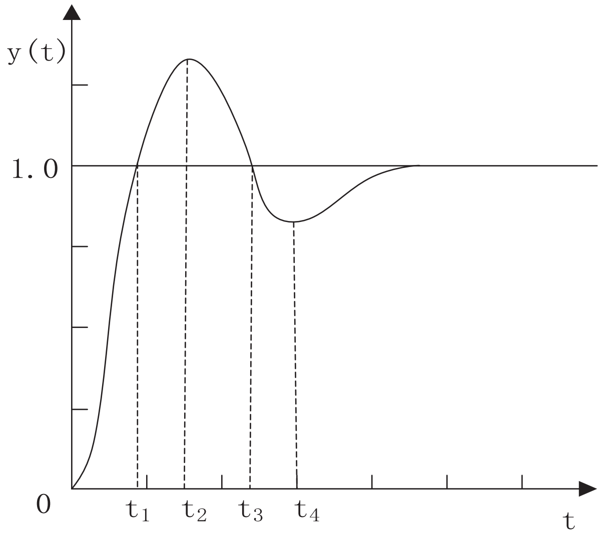

Figure 4. General system step response curve.

Figure 3. The controller PD is changed to nonlinear PD, and the output of nonlinear PD is:

$$u_1(t) = k_{pf}e(t) + k_{df}e(t) \tag{16}$$

The step response curve for a general system is shown in Figure 4. The design concept of a nonlinear PID controller can be derived from the analysis of this curve. In this paper, nonlinear PD control is adopted, so only proportional and differential parameters are introduced here.

Proportional gain parameter k_{pf}:In phase $0 \leq t \leq t_1$, in order to speed up the response speed of the system, the initial value of parameter k_{pf} should be relatively large. Meanwhile, in order to avoid overshoot as far as possible, when the error value e decreases gradually, parameter k_{pf} should also decrease with it. In phase $t_1 \leq t \leq t_2$, in order to increase the reverse control of the controller and reduce the overshoot of the system, parameter k_{pf} should be gradually reduced. In phase $t_2 \leq t \leq t_3$, in order to make the actual output of the system return to the desired point quickly and avoid large inertia, parameter k_{pf} should be gradually reduced. In phase $t_3 \leq t \leq t_4$, in order to reduce the error, parameter k_{pf} should increase, which is the same as in phase $t_1 \leq t \leq t_2$. To sum up, the variation rule of parameter k_{pf} with error e is shown in Figure 4, according to which a corresponding nonlinear function can be constructed:

$$k_{pf}(e(t)) = a_p + b_p(1 - \mathrm{sech}(\mathrm{c_p}e(t))) \tag{17}$$

In (17), a_p, b_p, c_p are positive real constants. When $e \to \pm\infty$, k_{pf} takes the maximum value $a_p + b_p$. When $e = 0$, k_{pf} takes the minimum value a_p; b_p is the interval of variation of k_{pf}, and the rate of change of k_{pf} can be adjusted by changing c_p.

Differential gain parameter k_{df}: In phase $0 \leq t \leq t_1$, in order to speed up the response and avoid overshoot as much as possible, parameter k_{df} should be gradually increased from small to large. In phase $t_1 \leq t \leq t_2$, in order to increase the reverse control and reduce the overshoot, we should continue to increase k_{df}. At t_2, reduce the parameter k_{df}, after that, it gradually increases by k_{df} again in phase $t_2 \leq t \leq t_4$, thus inhibiting the overshoot. To sum up, the variation rule of parameter k_{df} with error e is shown in Figure 4, and a nonlinear function is constructed according to the figure:

$$k_{df}(e(t)) = a_d + \frac{b_d}{1 + c_d exp(d_d e(t))} \tag{18}$$

In (18), a_d, b_d, c_d, and d_d are positive real constants, the maximum and minimum values of k_{df} are $a_d + b_d$ and a_d, respectively. When $e = 0$, $k_{df} = a_d + b_d/(1 + c_d)$, change d_d adjusts the rate of change of k_{df}. The nonlinear PD parameter change curve is shown in Figure 5.

Nonlinear PD is used to realize the control, mainly considering the change of P and D parameters with the change of control error. Especially for the nonlinear objects represented by the time-varying U-model, it can realize fast real-time control, improve the response

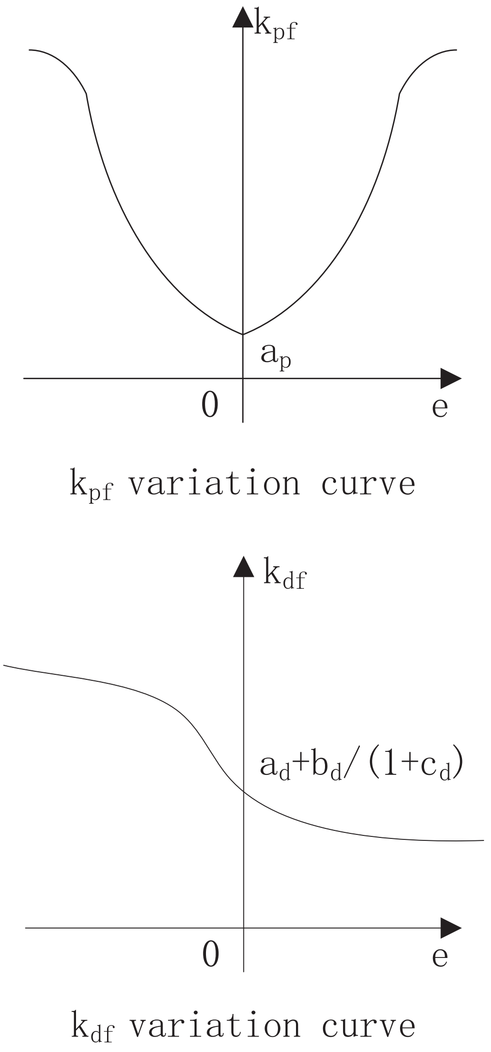

Figure 5. Nonlinear PD parameter change curve.

speed of the system, improve the control accuracy and anti-interference ability of the system. The design steps for parallel control of neural networks and nonlinear PD are the same as above.

5. System simulation

Simulation 1: Taking the continuous stirred tank reactor as an object, the general model of the system is $\dot{y} = -(1+2a)y + au - uy - ay^2$. In the formula, y represents the output of the controlled object, as a dimensionless of the concentration of a component. u represents the input of the controlled object, as a dimensionless of flow rate. The above equation is discretized and processed into the form of U model:

$$U(t) = a_0(t) + a_1(t)u(t-1) + a_2(t)u^2(t-1) + a_3(t)u^3(t-1)$$

Where,

$$a_0(t) = 0.8606y(t-1) - 0.0401y^2(t-1) + 0.0017y^3(t-1) - 0.000125y^4(t-1)$$

$$a_1(t) = 0.0464 - 0.045y(t-1) + 0.0034y^2(t-1) - 0.00025y^3(t-1)$$

$$a_2(t) = -0.0012 + 0.0013y(t-1) - 0.0001458y^2(t-1)$$

$$a_3(t) = 0.00002083 - 0.00002083y(t-1)$$

CMAC neural network parameters: The quantization number n is 100, the learning rate η is 0.9, the inertial coefficient α is 0.01, and the generalization parameter c is 8. CMAC concept mapping:

$$s(t) = \text{round}\left[(r(t) - x_{\min}) \cdot \frac{n}{x_{\max} - x_{\min}}\right]$$

where, $x_{\max}$ and $x_{\min}$ are the maximum and minimum values of the input value, where the values are 0.5 and −0.5 respectively. $rin(t)$ is the input value of the neural network, namely the ideal output value. The actual mapping of CMAC, where the code mapping technique uses the divisor remainder method: $ad(i) = (s(t) + i \;\; MOD \;\; N) + 1$, where, $i = 1, 2, \cdots, c$, and $N = c$. Nonlinear PD parameters: When the input signal is a square wave, $a_p = 0.7$, $b_p = 0.1$, $c_p = 0.1$, $a_d = 0.01$, $b_d = 0.01$, $c_d = 0.1$, $d_d = 0.1$. When the input signal is a triangular wave, $a_p = 1$, $b_p = 1$, $c_p = 1$, $a_d = 0.1$, $b_d = 0.1$, $c_d = 0.1$, $d_d = 0.1$. Fixed PD parameters: When the input signal is a square wave, $k_p = 0.35$, $k_d = 0.01$. When the input signal is a triangular wave, $k_p = 0.4$, $k_d = 0.05$.

Figures 6 and 8, respectively, show the output response of the control system in the case of two control algorithms. It can be seen from the figure that the improved control algorithm based on the combination of CMAC and nonlinear PD, compared with the control method based on the combination of CMAC and traditional PD, the system output is closer to the ideal output, and the control accuracy is high. As can be seen from Figures 7 and 9, the improved control algorithm

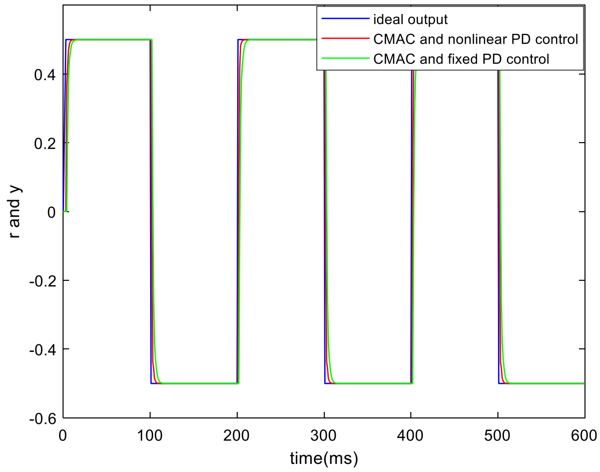

Figure 6. Control effect of the stirrer control system.

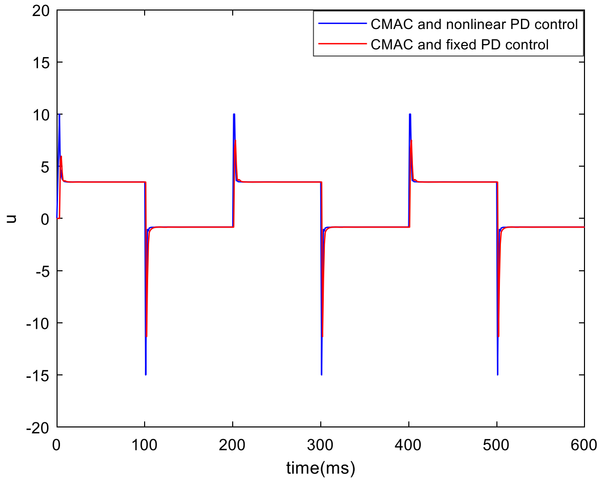

Figure 7. Stirrer control system control input signal.

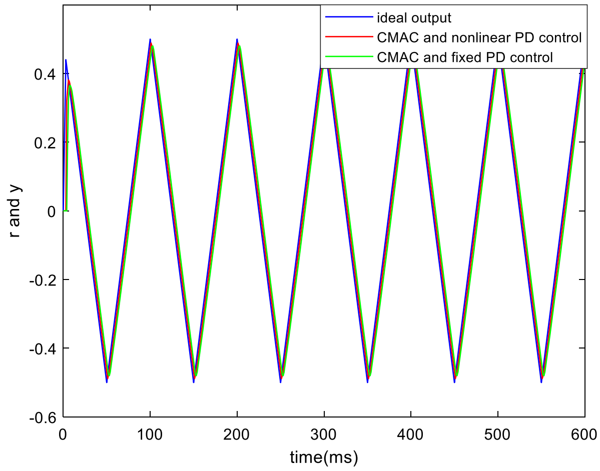

Figure 8. Control effect of the stirrer control system.

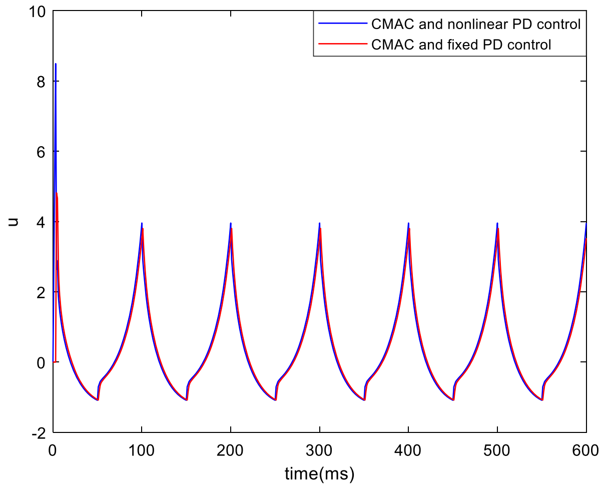

Figure 9. Stirrer control system control input signal.

has a faster system response speed and shorter adjustment time.

Simulation 2: The nonlinear model of laboratory liquid level control system is used to verify the rationality of the controller design. The U model of the system liquid level control system is:

$$y(t) = a_0 + a_1 u(t-1)$$

where,

$$\begin{aligned} a_0(t) &= 0.9722y(t-1) - 0.04288y^2(t-2) + 0.1663 \\ &\quad \times y(t-2)u(t-2) + 0.2573y(t-2)e(t-1) \\ &\quad - 0.03259y^2(t-1)y(t-2) - 0.3513y^2(t-1) \\ &\quad \times u(t-2) + 0.3084y(t-1)y(t-2)u(t-2) \\ &\quad + 0.2939y^2(t-2)e(t-1) \\ &\quad - 0.1295u(t-2) + 0.6398u^2(t-2)e(t-1) \\ a_1(t) &= 0.3578 - 0.3103y(t-1) + 0.1087y(t-2) \\ &\quad \times u(t-2) + 0.4770y(t-2)e(t-1) \end{aligned}$$

The parameter of the CMAC network when the input signal is sinusoidal: The quantization series n is 100, the generalization parameter is 8, the learning rate parameter is 0.9, the inertia coefficient is 0.01, the weight values are initialized to 0, and the initial values of k_p and k_d are 0.3 and 0.01 respectively. The parameters of the CMAC network when the input signal is a triangular: The parameters of CMAC network are: the quantization series n is 100, the generalization parameter is 8, the learning rate parameter is 0.85, the inertia coefficient is 0.01, the weight values are initialized to 0, and the initial values of k_p and k_d are 0.35 and 0.01, respectively.

It can be seen from Figures 10 and 12 that the control algorithm using the combination of CMAC and non-linear PD has a smaller tracking error, and the algorithm's anti-interference ability is enhanced. Simulation Figures 11 and 13 show that the response speed of the system is accelerated by the improved control algorithm.

The simulation results show that, because the nonlinear object is time-varying, the nonlinear PD controller can optimize the tracking speed of the time-varying nonlinear system, obtain the control signal with good smoothness, reduce the fluctuation of the system,

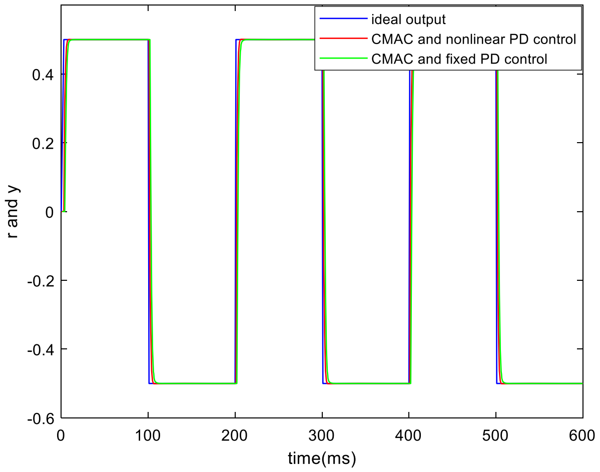

Figure 10. Control effect of liquid level control system.

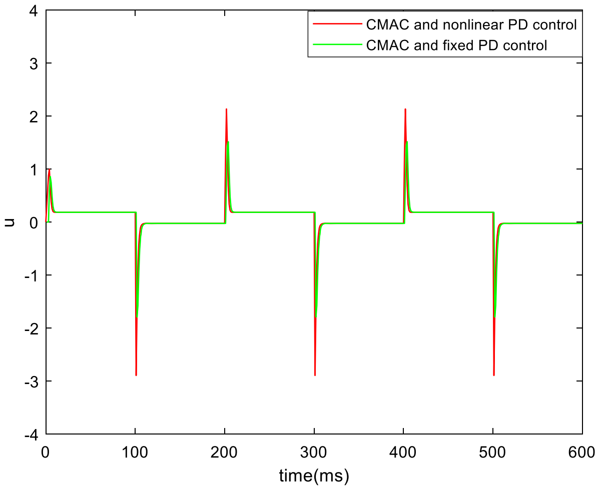

Figure 11. Control input of liquid level control system.

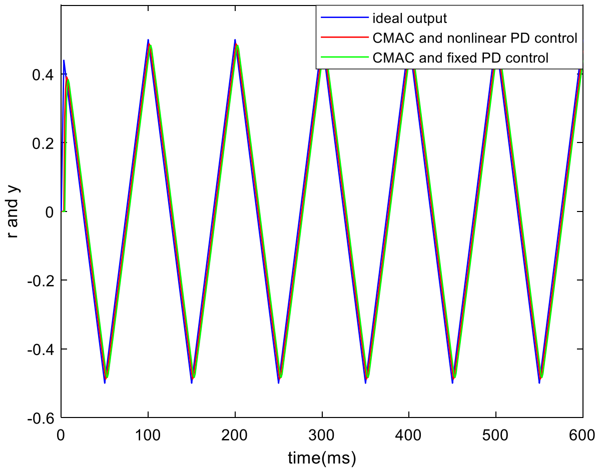

Figure 12. Control effect of liquid level control system.

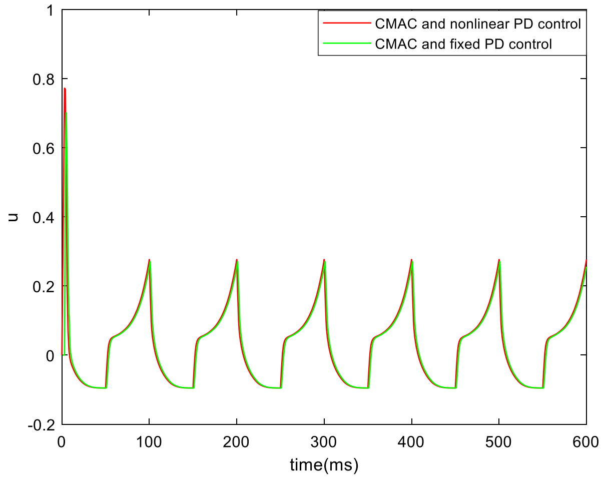

Figure 13. Control input of liquid level control system.

and improve the control accuracy and response speed of the system.

6. Conclusion

For the time-varying nonlinear U model object, this paper proposes the design of a nonlinear control system with CMAC neural network parallel to conventional PD. The control algorithm can effectively track the ideal output and obtain a smooth control input. On this basis, the influence of neural network controller learning speed on system performance is analysed, and a nonlinear PD control algorithm is proposed. The algorithm automatically adjusts the weights according to the size of the error according to the real-time changes of the object, and adjusts the controller parameters online to reduce the error of the system. This method can optimize the tracking speed of fast-varying nonlinear systems, improve the control accuracy of the system, and provide an effective solution for the control of time-varying nonlinear systems.

Acknowledgments

This work was supported in part by the Fundamental Research Funds in Heilongjiang Provincial Universities under [Grant No:135409602 No: 135409102].

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the Fundamental Research Funds in Heilongjiang Provincial Universities: [Grant No: 135409602 No:135409102].

References

[1] Leenaerts DMW, Van Bokhoven WMG. Piecewise linear modeling and analysis. Boston: Springer; 1998.

[2] Wang L, Wang QL. The feedback linearization based on backstepping technique. Proceedings of 2009 IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS) China. 2009. pp. 305–309.

[3] Khalil HK. Nonlinear systems. 3rd ed New Jersey: Prentice Hall; 2002.

[4] Hagan M, Demuth H, Beale M. Neural network design. Beijing: PWS Machinery Industry Press; 2002.

[5] Zhu Q, Ma Z, Warwick K. Neural network enhanced generalised minimum variance self-tuning control for nonlinear discrete-time systems. IEE P-Contr Theor Appl. 1999;146(4):319–326.

[6] Xu F, Zhu Q, Zhao D, et al. U-model based design methods for nonlinear control systems a survey of the development in the 1st decade. Control Decis 2013;28(7):961–977.

[7] Zhu Q, Guo L. A pole placement controller for nonlinear dynamic plants. Proc IMechE. 2002;216(16): 467–476.

[8] Shen W, Shi G. Adaptive fuzzy CMAC position tracking control based on pneumatic artificial muscle. J Shanghai Jiaotong Univ. 2012;46(4):579–583.

[9] Duan H. Wang F.Shi X. Recurrent CMAC model decomposition control for flying robot. Electr Machines Control. 2011;15(1):91–97.

[10] Hu S, Zhu S, Lou J, et al. High-precision control of giant magnetostrictive actuator based on CMAC neural network. J Vib Shock. 2009;28(03):68–72.

[11] Zhao Y, Yao J. Adaptive CMAC controller for energy saving in central air conditioning systems. Electr Technol Softw Eng. 2014;04:138–139.

[12] Wang H, Liao X. Research and application of nonlinear quantization CMAC. Comput Eng Appl. 2010;46(28): 218–221.

[13] Liu Y, Wang W. CMAC controller Research for servo system. J Aeronaut. 2006;03:515–519.

[14] Ma C, Hao L. Simulation studies of CMAC-PID combined control for electro-hydraulic position servo system. Proceedings of 2017 2nd International Conference on Computer, Mechatronics and Electronic Engineering(CMEE). 2017; 5. p. 45–48.

[15] Cao S. The design of carburization furnace potential and temperature DCS system based on CMAC and PD algorithm. Manufact Automat. 2010;32(14):209–212.

[16] Xu F, Tang D, Wang S. Research on parallel nonlinear control system of PD and RBF neural network based on U model. Automatika. 2020;61(2):284–294.

[17] Shafiq M, Butt NR. Real-time adaptive tracking of DC motor speed using U-model based IMC. Autom Control Comput Sci. 2007;41(1):31–38.

[18] Hasan E, Ibrahim RB, Ali SSA, et al. U-Model based adaptive control of gas process plant. Procedia Comput Sci. 2017;105:119–124.

[19] E. Hasan, R.B. lbrahim, S.S.A. Ali, H.S. Miya and S.F. Gilani, U-model based online identification of air flow plant. Robotics and Manufacturing Automation (ROMA), 2016 2nd IEEE International Symposium on IEEE; 2016.

[20] Zhao L, Wang J, Xiao X. A U model-based ADRC speed-sensorless control system for electrically excited synchronous motors. J Wuhan Univ Sci. 2016;39(4): 295–301.

[21] Ma X, Zhao Y, Di Y, et al. Trajectory tracking control of robot manipulators based on U-model. Math Probl Eng. 2020;08:1–10.

[22] Li W, Wang Z. U-Model-Based sliding mode augmented control method. Meas Contr Technol. 2021: 1–11.

[23] Zhang W, Zhang Z, Li Q, et al. Multiple U-model control of uncertain discrete-time nonlinear systems. Int J model Ident Contr. 2020;34(4):293–300.