CHEMICAL ENGINEERING TRANSACTIONS

VOL. 46, 2015

A publication of AIDIC
The Italian Association of Chemical Engineering
Online at www.aidic.it/cet

Guest Editors: Peiyu Ren, Yancang Li, Huiping Song

ISBN 978-88-95608-37-2; **ISSN** 2283-9216

DOI: 10.3303/CET1546026

Servo System PID Control of Neural Network Algorithm Based on LuGre Model

Xiaohui Xia*[a], Rui Yu[b]

[a] Changzhou College of Information Technology, Changzhou,Jiangsu, 213164, China,
[b] Changzhou Vocational Institute of Engineering, Changzhou, Jiangsu, 213164, China.
xiaohui0617@126.com

Considering the parameters of the traditional PID control algorithm are not capable of online adjustment and do not have adaptive capability once they are determined, which can not satisfied the precision control requirement of servo system with complex nonlinear friction, a PID controller based on radial basis function neural network (RBFNN) is designed, the advantages of RBFNN such as infinity approaching nonlinear system, little operation quantity and speedy constringency are combined with PID control technology organically to obtain high position tracking accuracy. The simulate results show that compared with common PID control, the RBFNN PID controller can adaptively adjust the controller parameters, greatly improving the control precision of the system.

1. Introduction

In the servo system of precision machinery, due to the existence of nonlinear friction (Xiang et al (2010), Erkorkmaz and Kamalzadeh (2006)), the static and dynamic performance of the system is greatly affected, mainly manifesting as the crawling of the mechanical servo system at low speed, large static errors or the occurrence of limit-cycle oscillation at steady state and other adverse effects. With industrial development, there are increasing requirements for the performance of mechanical servo system, and the compensation for friction moment has become a hot research topic at present (Kong et al (2010), Han and Lee (2010)). Currently, there are a lot of friction models used in the high-precision servo control system, among which, the LuGre friction model (Canudas(1995), Felix et al (2009)) is a more perfect dynamic friction model, and can accurately describe the complex static and dynamic characteristics in the process of friction, such as crawling, limit-cycle oscillation, pre-sliding deformation, friction memory, static friction and static Stribeck characteristics etc. (Leonid (2010), Zhe Cheng) (2012)).

Among the control methods of servo system, the most used is the PID control based on the linear control theory (Dong et al (2014), Ahmed A. et al (2013)). Although PID controller is characterized by simple control algorithm and strong robustness, its proportion, integral and differential parameters are preset and fixed. In the servo control system, the traditional PID control is difficult to meet the accuracy requirements due to the existence of nonlinear friction. For this reason, people combine the neural network (NN) with PID control for the implementation of controlling complex system (Tichun et al (2015), Yan (2015), Isaac (2013)). Radial basis function (RBF) neural network (Yen et al (2012), Yoo et al (2006)) is a kind of nonlinear network that can approach any complex process with any precision, which has little operation quantity and speedy constringency and other advantages (Dang and Zhao (2004), Wang and Huang (2005)). In this paper, based on the RBF neural network and PID controller, the RBF neural network is organically combined with the PID control technology to play their own strengths. The Neural network PID, through online learning, can adjust the link weights according to the impact of the changes in working condition of servo system on the system output performance, and change the proportion, integral and differential coefficient of the PID controller, so that the system can have good dynamic and static performance.

2. Servo system modeling

The dynamic equation of servo system can be expressed as:

Please cite this article as: Xia X.H., Yu R., 2015, Servo system pid control of neural network algorithm based on lugre model, Chemical Engineering Transactions, 46, 151-156 DOI:10.3303/CET1546026

$$J\ddot{\theta} = u - F \tag{1}$$

Where, J is the rotational inertia, θ is the angle, u is the control moment and F is the friction moment. F can be described by the LuGre model:

$$F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + \sigma_2 \dot{\theta} \tag{2}$$

$$\frac{dz}{dt} = v - \frac{|v|}{g(v)} z \tag{3}$$

$$\sigma_0 g(\dot{\theta}) = F_c + (F_s - F_c) e^{-(\dot{\theta}/v_s)^2} \tag{4}$$

Where, z presents the average amount of deformation of bristle on the contact surface, σ_0 is the bristle stiffness, σ_1 is the micro damping coefficient, σ_2 is viscous friction coefficient, F_c is the coulomb friction, F_s is the static friction, and v_s is the Stribeck speed.

3. RBF Neural Network PID Parameter Tuning Principle

After attaching RBF neural network to the traditional PID controller, the PID parameter can be determined with the self-learning function of neural network, and its system structure block-diagram is shown in Figure 1. According to the input and output information of the system, RBF network can get three optimized adjustable parameters k_P , k_I , k_D corresponding to PID controller under the steady state of the system through repeated self-learning and adjustment weight coefficient of the system.

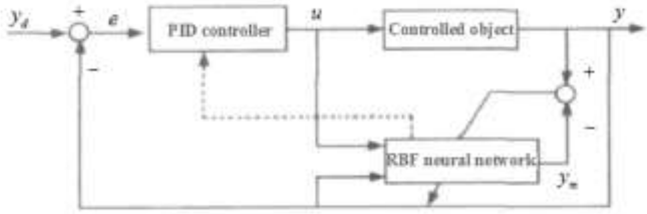

Figure 1: Block-diagram of the PID control system based on RBF neural network

3.1 PID parameter tuning principle

With the use of incremental PID controller, the control error is:

$$e(k) = y_d(k) - y(k) \tag{5}$$

Three inputs of PID are:

$$x(1) = e(k) - e(k-1) \tag{6}$$

$$x(2) = e(k) \tag{7}$$

$$x(3) = e(k) - 2e(k-1) + e(k-2) \tag{8}$$

The control algorithm is:

$$u(k) = u(k-1) + \Delta u(k) \tag{9}$$

$$\begin{aligned} \Delta u(k) = {} & k_P \left(e(k) - e(k-1) \right) + k_I e(k) \\ & + k_D \left(e(k) - 2e(k-1) + e(k-2) \right) \end{aligned} \tag{10}$$

The tuning indicator of neural network is:

$$E(k) = \frac{1}{2} e(k)^2 \tag{11}$$

The adjustment of k_P, k_I, k_D uses the gradient descent:

$$\Delta k_P = -\eta \frac{\partial E}{\partial k_P} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial \Delta u}\frac{\partial \Delta u}{\partial k_P} = \eta e(k)\frac{\partial y}{\partial \Delta u}x(1) \tag{12}$$

$$\Delta k_I = -\eta \frac{\partial E}{\partial k_I} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial \Delta u}\frac{\partial \Delta u}{\partial k_I} = \eta e(k)\frac{\partial y}{\partial \Delta u}x(2) \tag{13}$$

$$\Delta k_D = -\eta \frac{\partial E}{\partial k_D} = -\eta \frac{\partial E}{\partial y}\frac{\partial y}{\partial \Delta u}\frac{\partial \Delta u}{\partial k_D} = \eta e(k)\frac{\partial y}{\partial \Delta u}x(3) \tag{14}$$

Where, $\frac{\partial y}{\partial \Delta u}$ is the Jacobian information of the controlled object, and can be obtained through the identification by neural network.

3.2 RBF neural network

The radial basis function (RBF) neural network is a three-layer feed-forward neural network with good performance, which can map any complex nonlinear relationship and has the characteristics of simple learning rules as well as strong robustness, memory ability and self-learning capability. The structure of RBF network is shown in Figure 2.

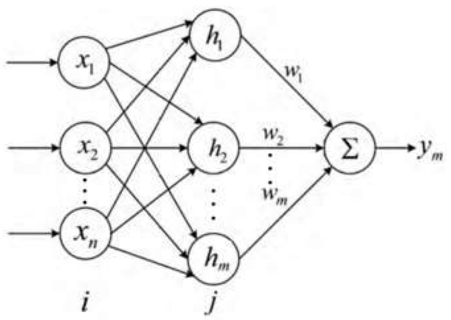

Figure 2: Structure of RBF neural network

In RBF network structure, $x = [x_1, x_2, \cdots, x_n]^T$ is the input vector of network; $h = [h_1, h_2, \cdots, h_m]^T$ is the output vector of m hidden units, where h_j is a Gaussian function:

$$h_j = \exp\left(-\frac{\|x - c_j\|^2}{2b_j^2}\right) \tag{15}$$

Where, $j = 1, 2, \cdots, m$; $W = [w_1, w_2, \cdots, w_m]^T$ is the link weight vector between m hidden units and output units; y_m is the network output, whose target is to approach the actual output of object y.

Set the data center matrix of hidden units as:

$C = [c_1, c_2, \cdots, c_m]^T$

Then the data center vector of the j –th hidden unit is:

$C_j = [c_{1j}, c_{2j}, \cdots, c_{nj}]^T$

The spread constant vector of the hidden unit is:

$B = [b_1, b_2, \cdots, b_m]^T$

b_j is the spread constant of the j –th hidden unit, and is greater than zero.

The linear combination that RBF network output is the output of m hidden units is:

$$y_m = \sum_{j=1}^{m} w_j h_j \tag{16}$$

The RBF network performance indicator function is:

$$J = \frac{1}{2}\left(y(k) - y_m(k)\right)^2 \tag{17}$$

In the k-th sampling period, after adjusting the network output weight coefficient, data center, spread constant according to the gradient descent to lead to the maximum value of performance indicator function of the identifier, we can get:

$$\begin{aligned} w_j(k) &= w_j(k-1) + \eta\left(y(k) - y_m(k)\right)h_j \\ &\quad + \alpha\left(w(k-1) - w(k-2)\right) \end{aligned} \tag{18}$$

$$\Delta b_j = \left(y(k) - y_m(k)\right) w_j h_j \frac{\left\|X - C_j\right\|^2}{b_j^3} \tag{19}$$

$$\begin{aligned} b_j(k) &= b_j(k-1) + \eta \Delta b_j \\ &\quad + \alpha\left(b(k-1) - b(k-2)\right) \end{aligned} \tag{20}$$

$$\Delta c_{ji} = \left(y(k) - y_m(k)\right) w_j \frac{x_i - c_{ji}}{b_j^2} \tag{21}$$

$$\begin{aligned} c_{ji} &= c_{ji}(k-1) + \eta \Delta c_{ji} \\ &\quad + \alpha\left(c_{ji}(k-1) - c_{ji}(k-2)\right) \end{aligned} \tag{22}$$

Where, η is the learning rate, and α is the momentum factor.

At the moment of k, due to $y_m(k) \approx y(k)$, if Δu is regarded as the first input node of RBF network, i.e., $\Delta u = x_1$, the sensitivity of the output of the object against the control of input is:

$$\begin{aligned} \frac{\partial y(k)}{\partial \Delta u(k)} &\approx \frac{\partial y_m(k)}{\partial \Delta u(k)} = \frac{\partial y_m(k)}{\partial \Delta h_j} \frac{\partial \Delta h_j}{\partial \Delta u(k)} \\ &= \frac{w_j h_j}{b_j^2}(c_{ij} - x_1) \end{aligned} \tag{23}$$

The equation (23) is the information necessary for the PID parameter setting obtained through the identification object of RBF network.

4. Simulation Analysis

The PID control algorithm and traditional PID control algorithm based on the RBF network tuning is programmed in the matlab software, and the simulation analysis is implemented to verify the validity of RBF neural network PID control method proposed in this paper.

In the servo system (1) and friction model (2-4), the $J = 0.01$, $\sigma_0 = 260$, $\sigma_1 = 2.5$ $\sigma_2 = 0.02$, $F_c = 0.28$, $F_s = 0.34$ and $v_s = 0.01$ is taken. There are 6 neurons in the hidden layer of RBF network. 10 is taken as the initial value of data center matrix, 20 as the initial value of spread constant vector, 1 as the initial value of output weight vector, the learning rate of the network is $\eta = 0.25$, and the momentum factor $\alpha = 0.5$. The simulation results are shown in Figure 3-5.

Figure 3 shows three parameters of PID controller tuned by the RBF neural network. From Figure 3, the neural network will automatically adjust the parameter of PID controller according to the operating condition of the system, in order to optimize the control performance of the system and improve the self adaptability of the system.

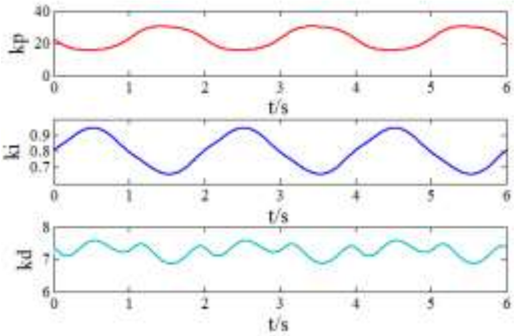

Figure 3: Adaptively tuning curves of PID parameters

Figure 4 and 5 are the position tracking curve and position tracking error curve respectively. Figure 4 shows that the position path errors are different with different algorithms, but the trend is basically the same. From Figure 5, the tracking error of traditional PID control leads to larger error (approximately 0.22rad), and the error of RBFPID is much smaller (about 0.005rad). Simulation results indicate that the RBFPID has better control effects than general PID control, which proves the validity of PID control algorithm based on RBF neural network.

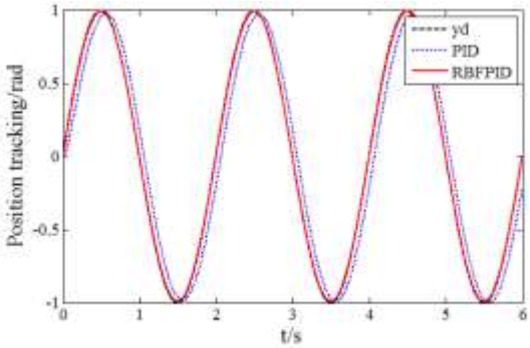

Figure 4: Position tracking curves

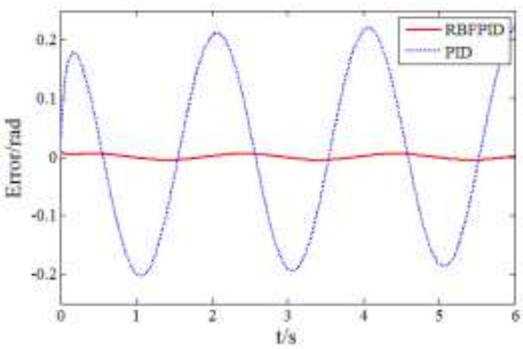

Figure 5: Position tracking error curves

5. Conclusions

In this paper, the LuGre model is used to describe complex nonlinear friction characteristics of the servo system, a RBFPID combined the RBF neural network with conventional PID is designed, and the specific control algorithm is derived in details. The controller achieves the adaptive adjustment of PID parameters with the self-learning ability and the ability to approach any function of the neural network. Simulation results show that compared with the general PID control, the RBF neural network control algorithm greatly improves the control precision of the system, so as to verify the validity of this control algorithm.

References

Canudas C., 1995, A new model for control of systems with friction [J]. IEEE Tran. on Automatic Control, 40(3): 419-425. DOI: 10.1109/9.376053

Chairez I., 2013, Differential Neuro-Fuzzy Controller for Uncertain Nonlinear Systems [J]. IEEE Transactions on Fuzzy Systems, 21(2): 369-384. DOI: 10.1109/TFUZZ.2012.2215875

Cheng Z., Hun N.G., Zhang X.F., 2012, Crack Level Estimation Approach for Planetary Gearbox Based on Simulation Signal and GRA [J]. Journal of Sound and Vibration, 5853–5863. DOI: 10.1016/j.jsv.2012.07.035

Dang K.F., Zhao H., 2004, Supervised control using fuzzy radial basis function networks for an electrohydraulic servo system [J]. Journal of System Simulation, 16(3): 593-595. (in Chinese) DOI: 10.16182/j.cnki.joss.2004.03.059

Elshafey A.A., Dawood N., 2013, Crack Width in Concrete Using Artificial Neural Networks [J]. Engineering Structures, 52: 676–686. DOI: 10.1016/j.engstruct.2013.03.020

Erkorkmaz K., Kamalzadeh A., 2006, High bandwidth control of ball screw drives [J]. CIRP Annals Manufacturing Technology, 55(1): 393-398. DOI: 10.1016/S0007-8506(07)60443-0

Felix J., Balthazar J., Brasil R., et al., 2009, On LuGre friction model to mitigate nonideal vibrations [J]. Journal of Computational and Nonlinear Dynamics, 4(3): 1-5. DOI: 10.1115/1.3124783

Freidovich L., Robertsson A., Shiriaev A., Johansson R., et al., 2010, LuGre-Model-Based Friction Compensation [J]. IEEE Transactions on Control Systems Technology, 18(1): 194-200. DOI: 10.1109/TCST.2008.2010501

Han S.L., Lee K.S., 2010, Robust friction state observer and recurrent fuzzy neural network design for dynamic friction compensation with backstepping control [J]. Mechatronics, 20(3): 381-401. DOI: 10.1016/j.mechatronics.2010.02.005

Kong X.Z., Wang Y., Jiang S.Y., 2010, Friction chatter-compensation based on stribeck model [J]. Journal of Mechanical Engineering, 46(5): 68-73. (in Chinese) DOI: 10.3901/JME.2010.05.068

Mian D., Gang S.Y., Qiang H.Y., et al., 2014, Modeling and Simulation Research of Active Heave Compensation System [J]. Review of Computer Engineering Studies, 1(2): 15-18. DOI: 10.18280/rces.010204

Wang D., Huang J., 2005, Neural network-based adaptive dynamic surface control for a class of uncertain nonlinear systems in strict-feedback form [J]. IEEE Transactions on Neural Networks, 16(1): 195-202. DOI: 10.1109/TNN.2004.839354

Wang T.C., Yan H., Zhong S.S., et al., 2015, Research of Fire Alarm System Based on Extension Neural Network [J]. Review of Computer Engineering Studies, 2(1): 9-16. DOI: 10.18280/rces.020102

Xiang H.B., Qiu Z.R., Li X.F., et al., 2010, Nonlinear friction modeling and compensation of high-precision experimental platforms [J]. Optics and Precisio n Engineering, 18(5): 1119-1127. (in Chinese) DOI: 10.3788/OPE.20101805.1119

Yan Y.Q., 2015, A Practice Guide of Predicting Consumption in a Web Server [J]. Review of Computer Engineering Studies, 2(6): 1-8. DOI: 10.18280/rces.020301

Yen H.M., Li T.H., Chang Y.C., 2012, Adaptive neural network based tracking control for electrically driven flexible-joint robots without velocity measurements [J]. Computers and Mathematics with Applications, 64(5): 1022-1032. DOI: 10.1016/j.camwa.2012.03.020

Yoo S.J., Park J.B., Choi Y.H., 2006, Adaptive dynamic surface control of flexible-joint robots using self-recurrent wavelet neural networks [J]. IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics, 36(6): 1342-1355. DOI: 10.1109/TSMCB.2006.875869