Research Article

Design of Contour Error Coupling Controller Based on Neural Network Friction Compensation

Sanxiu Wang, Guang Chen, and Yueli Cui

College of Electronic and Information Engineering, Taizhou University, Taizhou, Zhejiang 318000, China

Correspondence should be addressed to Sanxiu Wang; wsx8188@163.com

Received 22 June 2019; Revised 5 September 2019; Accepted 18 September 2019; Published 20 October 2019

Academic Editor: Mingshu Peng

Copyright © 2019 Sanxiu Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In two-axis servo contour motion control, friction and various uncertainties unavoidably exist, reducing the contour control accuracy. This paper proposes a neural network contour error coupling control method based on LuGre friction compensation, which includes a contour error calculation model, single-axis computed torque controller (CTC), and neural network friction compensation controller. The LuGre friction model can describe servo system’s complicated static and dynamic friction characteristics, and the RBF neural network has a universal approximation property to realize compensation control of friction. Simulation results indicate that the proposed contour error control method can effectively compensate for the effect of friction and improve contour control accuracy.

1. Introduction

Contour accuracy is an important factor that determines the surface quality of a workpiece and a necessary condition to guarantee high-precision machining [1, 2]. To improve contour accuracy, we must design a high-performance contour controller. Besides, in the multiaxis servo motion system kinetic model, various uncertain nonlinearities [3, 4] and various disturbances unavoidably exist, so we must design an effective control strategy to solve the impact of nonlinear uncertainty, thereby improving contour control accuracy.

Friction is a very complicated, nonlinear, and uncertain natural phenomenon [5, 6]. Friction will not only cause the wear of parts, but also reduce the mechanical strength of parts and even produce thermal deformation and thermal fatigue, thus affecting a motor’s normal running. Friction also causes the servo system to jump and crawl at low speed and causes waveform distortion when speed crosses zero, which makes the system unable to maintain smooth running [7]. For a servo system with high-precision and high-stability requirements, friction is an important factor affecting system performance. Therefore, in a mechanical motion system, if friction is not compensated for properly, it will seriously affect the system’s servo tracking performance. For a multiaxis motion system, once friction has led to the deterioration of a certain axis’s servo tracking performance, then the overall contour control performance will be affected. With the development of precision manufacturing, in order to achieve high-performance precision contour motion control, we must study the compensation of friction. Neural network has a universal approximation property and can approximate any nonlinear function with any precision [8–11]. Especially, the RBF neural network, as a local approximation network, is characterized by being fast to learn, and with this characteristic, the RBF neural network can be used for compensation control of friction that has complex nonlinear characteristics.

In view of the fact that friction in the multiaxis servo system will degrade the system’s tracking control performance and contour control performance, this paper proposes a neural network contour error coupling control algorithm based on friction compensation. This paper uses the LuGre dynamic friction model that can reflect the vast majority of friction characteristics in reality; the RBF neural network is used to approximate the LuGre friction model and serves together with the computed torque control as a single-axis tracking controller, which is combined with
a contour error calculation module to form a total contour error controller. At last, a simulation study on a two-dimensional linear servo motor is conducted to verify the effectiveness of the control scheme proposed herein.

2. System Description

2.1. System Model. With the two-dimensional linear servo motor as a controlled object, its Cartesian space dynamic model is as follows:

\[ D\ddot{q} + C\dot{q} + F = u, \]  

in which \( q, \dot{q}, \ddot{q} \) are \( 2 \times 1 \) vector, speed vector, and acceleration vector, respectively. \( D = \text{diag}[D_1, D_2] \) and \( C = \text{diag}[C_1, C_2] \) are \( 2 \times 2 \) inertia and viscous friction coefficient diagonal matrix, respectively, and \( F = [F_1, F_2]^T \) is friction torque, and \( u = [u_1, u_2]^T \) is control input vector.

The LuGre friction model assumes that two rigid bodies in relative motion microscopically contact through an elastic mane, and the material rigidity on the lower surface is greater than that on the upper surface [12, 13]. In the presence of an external force because of tangential force, the mane undergoes a deformation and thus generates friction. The establishment of the model is based on mane’s average deformation behavior, and it can describe most of the static and dynamic characteristics that can be observed in practice and accurately describe the friction phenomenon in the multi-axis numerical control system. Therefore, the friction torque in this paper considers the LuGre friction model.

Mane’s average deformation is expressed by intermediate state variable \( z \) [14]:

\[ \dot{z} = \dot{v} - \frac{\sigma_0|v|}{g(v)} z, \]

in which \( \dot{v} \) is relative motion speed of contact surface and \( g(v) \) indicates different friction effects.

The total friction torque of the LuGre model is described as

\[ g(v) = F_s + (F_s - F_d)e^{-(\omega_0v_s)} + \alpha \dot{v}, \]

\[ F = \sigma_0 z + \sigma_1 \dot{z} + \alpha \dot{v}, \]

in which \( \sigma_0 \) and \( \sigma_1 \) are dynamic friction parameters, i.e., mane’s rigidity coefficient and damping coefficient, respectively; \( F_s, F_d, \alpha, \alpha \) and \( V_s \) are static friction parameters, respectively expressed as coulomb friction coefficient, static friction coefficient, viscous friction coefficient, and Striebeck switching speed.

The LuGre friction model uses the first-order differential equation to describe many friction characteristics, including friction phenomena such as presliding, rising static friction, and friction memory, and it can also well describe the Striebeck effect, including most of the characteristics that can be observed in experiments and can well simulate real friction phenomena.

2.2. Contour Error Definition. In the actual high-speed machining process, due to the influence of various factors such as parameters mismatch of motion axes, multi-axis coordinated motion and friction, and load disturbance, contour error will inevitably emerge, and it is usually shown as the superposition of single-axis tracking errors and multi-axis-coordinated motion error. Tracking error is, in single-axis motion control, the difference between actual position response and reference position. Servo system’s parameter settings such as friction, vibration, and external disturbance in the machining process can all cause tracking error. Contour error is, in a multi-axis motion control system, the minimum distance between the actual position and contour machining trajectory [15].

Definition of contour error is shown in Figure 1.

Figure 1 shows contour errors of tracking circular trajectory and free-form trajectory. In Figure 1, \( P_d \) is any operating point’s reference position and \( P_a \) is the actual motion position at that time. In the given trajectory tracking control process, \( e \) is tracking error, \( e_x \) and \( e_y \) are tracking errors of \( x \)-axis and \( y \)-axis, respectively, and \( \theta \) is contour error. \( \theta \) is the positive angle between contour trajectory’s reference point tangent and \( x \)-axis.

Circular contour’s contour error is expressed as below:

\[ \epsilon = e_x \cdot L_x + e_y \cdot L_y = (P_{dx} - P_{ax})L_x + (P_{dy} - P_{ay})L_y, \]

in which \( (P_{dx}, P_{dy}) \) and \( (P_{ax}, P_{ay}) \) are contour trajectory’s reference point coordinate and actual coordinate position, respectively. \( L_x = -\sin \theta \) and \( L_y = \cos \theta \).

Similarly, circular contour’s contour error can also be expressed as below:

\[ \epsilon = R - \sqrt{(P_{dx} - n_x)^2 + (P_{dy} - n_y)^2}, \]

in which \( (n_x, n_y) \) are center coordinates of an inscribed circle at \( P_d \) and \( R \) is radius of circle. \( P_{dx} - n_x = R \sin \theta - e_x \) and \( P_{dy} - n_y = R \cos \theta - e_y \), and formula (5) can also be organized as follows:

\[ \epsilon = R - \sqrt{(R \sin \theta - e_x)^2 + (-R \cos \theta - e_y)^2}. \]

Second-order Maclaurin expansion of formula (6) is as follows:

\[ \epsilon = e_x \left(-\sin \theta + \frac{e_x}{2R}\right) + e_y \left(\cos \theta + \frac{e_y}{2R}\right) + \frac{1}{2R}(e_x \sin \theta - e_y \cos \theta)^2. \]

As tracking error is usually far smaller than the radius of circular contour, when second-order and higher order terms are ignored, from the above formula, it can be expressed as

\[ \epsilon = e_x \cdot L_x + e_y \cdot L_y, \]

in which \( L_x = -\sin \theta + e_x/2R \) and \( L_y = \cos \theta + e_y/2R \).

For contour trajectories of arbitrary shapes, contour errors can also be obtained by formula (8).
3. Design of Contour Error Coupling Controller

3.1. Contour Control Structure. The structure of contour error coupling control proposed in this paper is shown in Figure 2. In Figure 2, $q_{sid}$, $q_{syl}$ are the two axes’ reference positions, $q_x$, $q_y$ are the two axes’ actual output positions, and $e_x$, $e_y$ are the two axes’ contour errors.

The controller is mainly composed of a single-axis computed torque tracking controller (computed torque control, CTC), neural network friction compensation controller, contour error calculation module, and two-axis servo system with LuGre friction. To reduce the effect of contour controller, contour error calculation module, and two-axis servo control, CTC), neural network friction compensation controller to calculate control quantities, and export superpose them onto control quantities, and export the total control quantity to each axis servo actuator.

The specific work process is as follows:

1. From each axis’s reference input position and actual output obtain the two axes’ tracking errors $e_x$, $e_y$.
2. Put the two axes’ tracking errors through contour error calculation module to obtain the contour errors.
3. Put the two axes’ tracking errors, respectively, through $x$, $y$ single-axis CTC and the neural network friction compensation controller to calculate control quantities $u_x$, $u_y$.
4. Multiply the two axes’ contour errors by zoom factor, superpose them onto control quantities, and export the total control quantity to each axis servo actuator.
5. Act the total control quantity on each axis servo actuator for contour motion control.

3.2. Design of Neural Network Controller Based on LuGre Friction Compensation. For the linear servo motor control system expressed by formula (1), if friction torque is disregarded, we can use the following computed torque control:

$$u_0 = D_0(q_d - k_pe - k_pe) + C_0q_d,$$  \hspace{1cm} (9)

in which $q_d$ is desired trajectory, $q$ is actual output trajectory, $e$ and $\dot{e}$ are single-axis position tracking error and speed tracking error, respectively, and $k_p$, $k_d$ are control gain.

If friction is disregarded, then CTC is asymptotic and stable, and we can obtain a stable closed-loop error system:

$$\dot{e} + k_pe + k_de = 0.$$  \hspace{1cm} (10)

But, in an actual mechanical motion control system, friction is unavoidable. Control law (9) is introduced into the linear motor model (1) to obtain

$$\dot{e} + k_pe + k_de = -D_0^{-1}F.$$  \hspace{1cm} (11)

Take $x = (e, \dot{e})^T$ and make $f = -D_0^{-1}F$, and the above formula can be written as

$$\dot{x} = Ax + B(D_0^{-1}F) = Ax + Bf,$$  \hspace{1cm} (12)

in which $A = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

As friction parameters are hard to identify, it is hard to build an accurate friction model, and even though an accurate friction model is built, its complex expression will make the model-based friction compensation control method hard to implement. In view of neural network’s universal approximation property of approximating any nonlinear function with any precision, below we will use the RBF neural network to carry out approximation and compensation of LuGre model’s uncertainty $f = -D_0^{-1}F$.

The neural network $\tilde{f}(x)$ estimates LuGre friction model’s uncertainty $f$, expressed as below:

$$\tilde{f}(x) = \tilde{W}^T\varphi(x),$$  \hspace{1cm} (13)

in which $x \in \mathbb{R}^n$ is neural network’s input vector, $\tilde{W}$ is neural network’s weight matrix, $\varphi(x) = [\varphi_1(x), \varphi_2(x), \ldots, \varphi_n(x)]^T$ is Gaussian function, $\varphi_i(x) = \exp(-\|x - c_i\|/\sigma_i^2)$, and $c_i$ and $\sigma_i$ are the center and width of the $i$th Gaussian function.

For any given small positive number $r_0$, there must be an optimal weight vector $W^*$, which makes neural network’s approximation error $\Delta f$ satisfy the following relation:

$$\|\Delta f\| = \|W^T\varphi(x) - f(x)\| < r_0,$$  \hspace{1cm} (14)

in which $W^* = \arg \min_{W \in \mathbb{R}^n} \left\{ \sup_{x \in \mathcal{M}(\mathcal{M}_x)} \|f(x) - \tilde{f}(x, W)\| \right\}$, indicating neural network’s optimal weight vector, and $\| \cdot \|$ is vector’s $2$- norm.

Take $\eta$ as neural network’s approximation error, i.e.,

$$\eta = f(x) - \tilde{f}(x, W^*),$$  \hspace{1cm} (15)

in which $\tilde{f}(x, W^*) = W^T\varphi(x)$.

Thus,

$$f(x) - \tilde{f}(x, W) = f(x) - \tilde{f}(x, W^*) + \tilde{f}(x, W^*) - \tilde{f}(x, W) = \eta + W^T\varphi(x) - \tilde{W}^T\varphi(x) = \eta - \tilde{W}^T\varphi(x),$$  \hspace{1cm} (16)

in which $\tilde{W} = \tilde{W} - W^*$.

Formulate Lyapunov function:

$$V = \frac{1}{2}x^TPx + \frac{1}{2}\|W\|^2,$$  \hspace{1cm} (18)
in which \( \gamma > 0 \).

Matrix \( P \) is a positive definite matrix, which satisfies the Lyapunov equation,

\[
P + A^T P = -Q,
\]

in which \( Q \geq 0 \).

Make \( \|W\|^2 = \text{tr}(W^T W) \), and

\[
\dot{V} = \frac{1}{2} \left[ x^T P \dot{x} + x^T P x \right] + \frac{1}{\gamma} \text{tr} \left( \dot{W}^T \dot{W} \right)
\]

\[
= \frac{1}{2} \left[ x^T P \left( A x + B(\eta - \dot{W}^T \varphi(x)) \right) + \left( x^T A^T + \left( \eta - \dot{W}^T \varphi(x) \right)^T B^T \right) P x \right] + \frac{1}{\gamma} \text{tr} \left( \dot{W}^T \dot{W} \right)
\]

\[
= \frac{1}{2} \left[ x^T \left( PA + A^T P \right)x + \left( -x^T PB \dot{W}^T \varphi(x) + x^T PB \eta \right) \right. \\
- \left. \varphi^T(x) \dot{W} B^T P x + \eta^T B^T P x \right] + \frac{1}{\gamma} \text{tr} \left( \dot{W}^T \dot{W} \right)
\]

\[
= \frac{1}{2} x^T Q x - \varphi^T(x) \dot{W} B^T P x + \eta^T B^T P x + \frac{1}{\gamma} \text{tr} \left( \dot{W}^T \dot{W} \right).
\]

As

\[
\varphi^T(x) \dot{W} B^T P x = \text{tr} \left( B^T P x \varphi^T(x) \dot{W} \right),
\]

then

\[
\dot{V} = \frac{1}{2} x^T Q x + \frac{1}{\gamma} \text{tr} \left( -y B^T P x \varphi^T(x) \dot{W} + \dot{W}^T \dot{W} \right) + \eta^T B^T P x.
\]

Select the following neural network weight adaptive law:

\[
\dot{W} = \gamma \varphi x^T P B + k_1 \|x\| \bar{W},
\]

in which \( k_1 > 0 \).

Introduce the above neural network weight adaptive law into formula (22) to obtain

\[
\dot{V} = -\frac{1}{2} x^T Q x + \frac{1}{\gamma} \text{tr} \left( k_1 \|x\| \bar{W}^T \bar{W} \right) + \eta^T B^T P x = -\frac{1}{2} x^T Q x + k_1 \|x\| \text{tr} \left( \bar{W}^T \bar{W} \right) + \eta^T B^T P x.
\]

To make the system converge, i.e., \( \dot{V} \leq 0 \), the following conditions need to be met:

\[
\frac{1}{2} \lambda_{\min} (Q) \|x\| \geq \|\eta\| \lambda_{\max} (P) + k_1 W_{\max}.
\]

in which \( \lambda_{\min} (Q) \) and \( \lambda_{\max} (P) \) are minimum eigenvalue of matrix \( Q \) and maximum eigenvalue of matrix \( P \), respectively, and \( \eta \) is the upper bound of neural network's approximation error, which satisfies \( \|\eta\| \leq \|\eta_0\| \).

Formula (25) is organized as follows:

\[
\|x\| \geq \frac{2}{\lambda_{\min} (Q)} \left( \|\eta_0\| \lambda_{\max} (P) + k_1 W_{\max} \right),
\]

which shows \( x \)'s radius of convergence is related to \( \lambda_{\min} (Q) \), \( \lambda_{\max} (P) \), \( \eta_0 \), and \( W_{\max} \). The greater \( \lambda_{\min} (Q) \) is, the smaller \( \lambda_{\max} (P) \), \( \eta_0 \), and \( W_{\max} \) are, and the smaller \( x \)'s radius of convergence is, the better the system's trajectory tracking performance is.

In summary, the complete control input torque of the neural network friction compensation control algorithm for the uniaxial servo system is...
\[ u = u_0 + \ddot{f} = D\dot{q} (\dot{q}_d - K_F e - K_D e_\dot{e}) + C_\theta \dot{q} + \hat{W}^T \varphi (x) . \tag{27} \]

4. Simulation Analysis

To verify the effectiveness of the contour error coupling control method proposed herein, we conduct a contour motion control simulation study on the two-axis linear servo motor system with LuGre friction expressed by formula (1). Due to different load changes and environmental conditions, the two axes' model parameters do not match exactly, which causes contour error. In simulation, different x-axis and y-axis simulation parameters are selected to reflect such parameters mismatch. x-axis simulation parameters are \( D_0 = 0.1 \) and \( C_0 = 2 \), and the LuGre friction model parameters are \( \sigma_0 = 260 \), \( \sigma_1 = 2.5 \), \( \alpha = 0.02 \), \( F_c = 0.28 \), \( P_c = 0.34 \) , and \( V_s = 0.01 \). y-axis simulation parameters are \( D_0 = 0.2 \) and \( C_0 = 2 \), and the friction model parameters \( \sigma_0 = 200 \), \( \sigma_1 = 2.8 \), \( \alpha = 0.05 \), \( F_c = 0.3 \), \( F_s = 0.5 \), and \( V_s = 0.02 \).

Single-axis trajectory tracking motion control adopts computed torque control (CTC), and the neural network is used to carry out approximation and compensation of the LuGre friction model in the servo system. The parameters of the x-axis controller are \( k_p = 200 \), \( k_d = 50 \), \( \gamma = 1000 \) and \( k_i = 0.001 \); the parameters of the y-axis controller are \( k_p = 100 \), \( k_d = 80 \); \( \sigma = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix} \), \( \gamma = 800 \), \( k_i = 0.001 \); the initial values of the center and width of neural network’s Gaussian function are 0.6 and 3.0, respectively. Unit circular contour whose tracking curve radius is 1 is selected: \( L = \{(q_{dx}, q_{dy}, t) \in R^2 : q_{dx} (t) = \sin (t), q_{dy} (t) = \cos (t)\} \).

Simulink and S function are used for the design of control system, and simulation results are shown in Figure 3.

Figure 3(a) shows the two axes’ position tracking, where the solid line is single-axis desired trajectory, and the dotted line is the actual motion trajectory. Figure 3(b) is two-axis tracking error and Figures 3(c) and 3(d) are circular contour input and contour output, respectively, Figure 3(e) is two-axis LuGre friction and its neural network approximation, which shows that friction is approximated effectively by neural networks and Figure 3(f) is contour error. The simulation results can visually reflect the system’s control effect under the action of the neural network controller.

In order to better reflect the control contour performance of the control algorithm proposed in this paper and the effect of compensation of its friction, we will compare it with the following two cases.

Case 1. Using a robust controller to compensate the friction. Controller is designed as \( u_r = - (\beta / \|s\| + \varepsilon) s \), in which \( \beta \) is the upper bound of friction term and satisfies \( \| - D_0^{-1} F \| \leq \beta \). \( s = \dot{s} + \Lambda c \). The simulation parameters are chosen as \( \Lambda = 5 \) and \( \varepsilon = 0.2 \).

Case 2. Use computed torque control (CTC) alone, and there is no friction compensation controller.

Simulation effect comparison of these three cases is shown in Figure 4.

Figure 4(a) is the comparison of circular contour outputs in three situations: with the action of neural network friction compensation, with the action of friction compensation by the robust controller, and without the action of friction compensation; Figure 4(b) shows contour errors under the three kinds of controller; and Figures 4(c) and 4(d) are the comparison of the two axes’ contour tracking errors. We can clearly see that, under the action of computed torque control (CTC) alone there are obvious contour errors, and through the robust controller action, the tracking error and contour error are improved slightly, while through neural network’s compensation action on LuGre friction contour error is obviously improved. At the same time, the two axes’ tracking errors are obviously reduced. Compared with the action of these three control methods is shown in Table 1.

We can visually find from the error comparison table that the under the action of the CTC and neural network friction compensation controller the tracking errors and contour errors are both slight. Compared with the robust friction compensation controller, x-axis’s average tracking error drops from 0.0036 to 0.0014, y-axis’s average tracking error decreases from 0.0024 to 0.00064, average contour error drops from 0.0033 to 0.0013, and contour accuracy is improved 2.5 times. Compared with the CTC alone, x-axis’s average tracking error drops from 0.0184 to 0.0014 and y-axis’s average tracking error greatly decreases from 0.0107 to 0.00064. Contour error is also improved significantly, and average contour error drops from 0.0157 to 0.0013 and contour accuracy is improved 12 times.

5. Conclusions

In a mechanical motion control system, friction is unavoidable and would reduce the servo tracking performance, thereby affecting the contour control performance. Considering that the LuGre dynamic friction model can well reflect the typical actual friction characteristics, this paper proposes a neural network contour error coupling controller based on LuGre friction compensation. The controller includes a contour error calculation module, a single-axis computed torque control, and a RBF neural network friction compensation controller.

The proposed neural network friction compensation control, the robust friction compensation control, and the nonfriction compensation control are applied to the two-axis servo motor, respectively. The comparison of simulation results shows that, under the action of single-axis computed torque control, single-axis tracking error and contour error are both very obvious, and the effect is improved under the action of the robust controller, but the effect is the best under the action of neural network friction compensation. Under the action of the control method proposed in this paper, single-axis tracking control error and two-axis contour error are greatly reduced. Compared with the case without friction compensation, x-axis maximum tracking error, y-axis maximum tracking error, and y-axis maximum contour error are greatly reduced.
Figure 3: Simulation effect under the action of neural network friction compensation control. (a) Two-axis position tracking. (b) Two-axis tracking error. (c) Circular contour input. (d) Circular contour output. (e) Two-axis LuGre friction and its neural network compensation. (f) Contour error.
error, and maximum contour error dropped to 29.2%, 9.09%, and 10.31%, respectively. It proves that the influence of friction is effectively compensated and the proposed control method has a good control effect.

**Data Availability**

The data used to support the findings of this study are included within the article.
Conflicts of Interest
The authors declare that they have no conflicts of interest.

Acknowledgments
This work was supported by Joint Fund of Zhejiang Provincial Natural Science Foundation, Grant no. TY20E050003 and the Natural Science Foundation of Zhejiang Province of China under Grant no. LQ18F030002.

References


