

## Design of cross-coupled CMAC for contour-following – a reinforcement-based ILC approach

Wei-Liang Kuo, Ming-Yang Cheng & Wei-Che Tsai

To cite this article: Wei-Liang Kuo, Ming-Yang Cheng & Wei-Che Tsai (2017) Design of cross-coupled CMAC for contour-following – a reinforcement-based ILC approach, *Automatika*, 58:3, 302-311, DOI: [10.1080/00051144.2018.1426231](https://doi.org/10.1080/00051144.2018.1426231)

To link to this article: <https://doi.org/10.1080/00051144.2018.1426231>



© 2018 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 07 Mar 2018.



Submit your article to this journal [↗](#)



Article views: 255



View related articles [↗](#)



View Crossmark data [↗](#)



# Design of cross-coupled CMAC for contour-following – a reinforcement-based ILC approach

Wei-Liang Kuo, Ming-Yang Cheng and Wei-Che Tsai

Department of Electrical Engineering, National Cheng Kung University, Tainan City, Taiwan (R.O.C.)

## ABSTRACT

One of the most popular applications of a bi-axial motion stage is precision motion control. The reduction of tracking error and contour error is one of the most coveted goals in precision motion control systems. The accuracy of a motion control system is often affected by external disturbances. In addition, system non-linearity such as friction also represents a major hurdle to motion precision. In order to deal with the aforementioned problem, this paper proposes a fuzzy logic-based Reinforcement Iterative Learning Control (RILC) and a Cross-Coupled Cerebellar Model Articulation Controller (CCCMAC). In particular, the proposed fuzzy logic-based RILC and a LuGre friction model-based compensation approach are exploited to improve motion accuracy. The fuzzy logic-based RILC aims at reducing tracking error and compensating for external disturbance, while the LuGre friction model is responsible for friction compensation. In addition, the CCCMAC consisting of a cerebellar model articulation controller and a cross-coupled controller aims at reducing contour error and dealing with the problem of dynamics mismatch between different axes. Performance comparisons between the proposed fuzzy logic-based Reinforcement Iterative Learning Cross-Coupled Cerebellar Model Articulation Controller (RIL-CCCMAC) and several existing control schemes are conducted on a bi-axial motion stage. Experimental results verify the effectiveness of the proposed RIL-CCCMAC.

## ARTICLE HISTORY

Received 25 November 2016  
Accepted 22 December 2017

## KEYWORDS

Contour error estimation; iterative learning control (ILC); cerebellar model articulation controller (CMAC); reinforcement learning; LuGre friction model

## 1. Introduction

Recently, due to the thriving development of the Information Technology (IT) industry, the Computer Numerical Control (CNC) machine tools used in manufacturing IT products have been in high demand. Most CNC machine tools are used in precision machining applications, where high precision multi-axis contour-following is essential. Consequently, the development of a sophisticated motion control scheme that is able to achieve high multi-axis contour-following accuracy is crucial [1–4]. In general, tracking error and contour error are two of the popular performance indices used to assess the contour-following accuracy of a CNC machine tool. Smaller tracking error and contour error indicate better contour-following accuracy and product quality.

In practice, due to different causes such as non-linearity, external disturbance and dynamics incompatibility between different axes, the motion accuracy of a multi-axis motion platform may deteriorate [5–7]. To deal with the problem of dynamics incompatibility between different axes, in addition to the well-known feedforward controller, the Cross-Coupled Controller (CCC) proposed by Koren is likely the most popular [8]. The idea of the CCC is to estimate contour error based on the tracking error of each axis. The estimated

contour error is used as a basis to calculate suitable compensation torque/force to each axis, yet the original version of the CCC is only suitable for the case of linear trajectory. In order to enhance the applicability of the CCC, many researchers have developed different kinds of contour error estimation algorithms that are suitable for non-linear trajectories or even free-form parametric curves [9,10]. Recently, the parameter-based contour error estimation algorithm was reported to exhibit good contour error estimation performance [11]. Therefore, this paper will incorporate the parameter-based contour error estimation algorithm into the CCC for application to different types of trajectories, as well as improve the effectiveness of the CCC. In addition to the contour error estimation algorithm, one of the possible solutions to improve the effectiveness of the CCC is the design of cross-coupled gain. In the original CCC structure, the cross-coupled gain is fixed for the case of linear trajectory. However, it is not reasonable if the cross-coupled gain is fixed for other types of trajectories.

On the other hand, due to the rapid development of artificial neural networks and other learning approaches for the past several decades, more and more learning control algorithms are exploited to improve contour-following accuracy of multi-axis motion stages [12–18]. For instance, Wen and Cheng

proposed a recurrent fuzzy Cerebellar Model Articulation Controller (CMAC) with a self-tuning learning rate for the control of a bi-axial piezoelectric-actuated micro-motion stage [18]. Shi and Shen proposed an adaptive sliding mode fuzzy CMAC structure for the control of a high-order pneumatic artificial muscles system [19]. Inspired by these previous studies, in order to cope with the problem of dynamics incompatibility between different axes, this paper incorporates the CMAC into a CCC structure (Cross-Coupled Cerebellar Model Articulation Controller, CCCMAC) to tune the cross-coupled gain based on the contour error information provided by a real-time contour error estimation algorithm.

Although the proposed CCCMAC can deal with the dynamics incompatibility problem, it may not be as effective when coping with periodic external disturbance and non-linearities such as friction resulting from repetitive contour-following motions. In order to overcome this difficulty, the fuzzy logic-based Reinforcement Iterative Learning Control (RILC) scheme proposed in this paper and the LuGre friction model [20–23] are used to suppress external disturbance and compensate for friction, respectively. Several contour-following tasks are conducted to verify the effectiveness of the proposed approach.

The rest of the paper is organized as follows. Section 2 introduces the CCCMAC with parameter-based contour error estimation algorithm. The proposed fuzzy logic-based RILC is elaborated in Section 3. Experimental set-up and contour-following results are given in Section 4. Conclusions are addressed in Section 5.

## 2. Cross-coupled CMAC with parameter-based contour error estimation

### 2.1. Real-time calculation of contour error for free-form parametric curves

As mentioned previously, contour error is an important performance indicator in contour-following applications. The calculation of contour error for special types of trajectories such as straight lines and circles is routine. However, it is not the case for free-form parametric curves. In general, real-time calculation of exact contour error is very difficult if not impossible. As a result, many real-time contour error estimation algorithms have been proposed. For instance, based on the idea of straight line approximation, Cheng and Lee proposed a real-time algorithm that is able to estimate contour error for free-form parametric curves [24]. The drawback of this method is that the reference point used to estimate the contour error is not on the parametric curve. In order to circumvent this issue, Chen et al. [11] proposed a parameter-based contour error estimation algorithm, in which the reference point used to estimate contour error is on the

parametric curve. Compared with several existing contour error estimation algorithms, the approach proposed by Chen et al. [11] was reported to exhibit better estimation accuracy. In addition, cross-coupled control-based approaches require knowledge of contour error. Better contour error estimation results will help to boost the performance of the cross-coupled control scheme so as to cope with adverse effects due to dynamics mismatch among different axes. In this paper, the parameter-based contour error estimation algorithm proposed in [11] is employed to provide accurate contour error estimation to the cross-coupled control scheme.

### 2.2. Cross-coupled CMAC controller

The original CCC structure proposed by Koren [8] was designed for straight line-following tasks. Based on contour error information, the CCC aims at reducing contour error caused by dynamics incompatibility among different axes. In order to extend the use of CCC to other types of contour-following tasks, amendments to real-time contour error estimation algorithms are essential. Figure 1 illustrates the block diagram of a bi-axial motion control scheme that contains a CCC teamed up with the parameter-based contour error estimation algorithm. In Figure 1,  $R_x$  and  $R_y$  are the position commands for  $x$ -axis and  $y$ -axis, respectively;  $P_x$  and  $P_y$  are the real current positions for  $x$ -axis and  $y$ -axis, respectively;  $K_{px}$  and  $K_{py}$  are the position loop gains for  $x$ -axis and  $y$ -axis, respectively;  $e_x$  and  $e_y$  are the tracking errors of  $x$ -axis and  $y$ -axis, respectively;  $E'_c$  is the estimated contour error,  $K_c$  is the gain of the CCC, while  $C_x$  and  $C_y$  are the cross-coupled gains for  $x$ -axis and  $y$ -axis, respectively. In the original CCC developed for straight line-following tasks, the cross-coupled gains are defined as follows:

$$C_x = -\sin\theta \quad (1)$$

$$C_y = \cos\theta \quad (2)$$

where  $\theta$  is the angle between the straight line contour and  $x$ -axis.

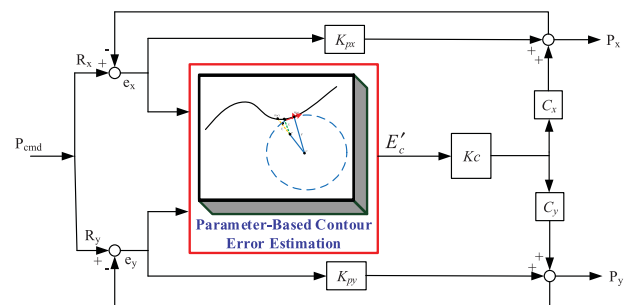


Figure 1. Block diagram of a bi-axial motion control scheme that contains a CCC in conjunction with the parameter-based contour error estimation algorithm proposed in [11].

However, in many contour-following applications, the contour to be followed is not a straight line. That is, the angle  $\theta$  required in the calculation of cross-coupled gains described by (1) and (2) is no longer available since the contour to be followed is not a straight line. One way to cope with this problem is to redefine the angle  $\theta$ . In this paper, the tangent vector at the current position command is computed.  $\theta$  is defined to be the angle between the tangent vector and  $x$ -axis.

The CMAC proposed by Albus [25,26] is an artificial neural network belonging to the category of associative memory. It uses local basis functions and has attractive characteristics such as fast learning and good approximation abilities. It has a simple structure and is suitable for online control. In general, the input space  $S$  of a CMAC is quantized into many hypercubes and each hypercube is associated with a memory cell. In addition, the content of a specific memory is called its weight. When used in a control system, Equation (3) is employed to compute the control output of a CMAC, while Equations (4) and (5) are used to update the weights of memory cells.

$$U_{\text{cmac}} = a_{Sn}^T w = \sum_{j=1}^{Nm} a_{Sn,j} w_j \quad (3)$$

$$\begin{aligned} w_j(k) &= w_j(k-1) + \Delta w_j(k-1) \\ &= w_j(k-1) + \frac{\lambda}{N_e} a_{s,j} u_e(k-1) \end{aligned} \quad (4)$$

$$u_e(k-1) = E'_c(k) - E'_c(k-1) \quad (5)$$

where  $a_{Sn}$  is the associated memory index vector of state  $s_{N_s}$ ,  $w$  represents the memory weight vector for stored data and  $Nm$  is the size of the total physical memory.

Figure 2 illustrates the block diagram of the CCCMAC proposed in this paper, in which the CCC  $K_c$  in the original CCC structure is replaced by a

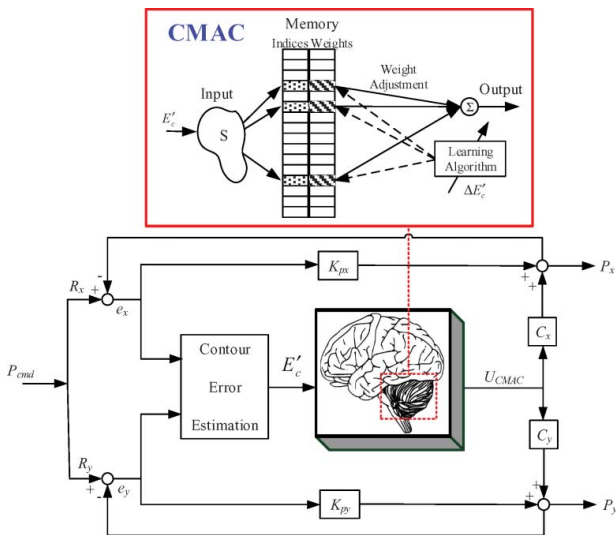


Figure 2. Block diagram of the proposed CCCMAC.

CMAC. The idea of the proposed CCCMAC is to adjust the amount of compensation to each axis using a CMAC so as to reduce contour-following errors caused by incompatibility among different axes. Note that the weights of the proposed CCCMAC are trained based on the estimated contour error obtained from the parameter-based contour error estimation algorithm.

### 3. Fuzzy logic-based reinforcement ILC with friction compensation

This section will introduce the proposed fuzzy logic-based Reinforcement Iterative Learning Cross-Coupled Cerebellar Model Articulation Controller (RIL-CCCMAC) with LuGre model-based friction compensation.

#### 3.1. LuGre model-based friction compensation

Friction is one of the most crucial non-linear phenomena that may lead to deterioration in motion accuracy for a servomechanism. Unlike the classical friction model, the LuGre friction model proposed by Wit et al. [20] is a dynamic model. According to Wit et al. [20], the LuGre friction model is described by

$$\frac{dz}{dt} = v - \frac{|v|}{g(v)} z \quad (6)$$

$$\sigma_0 g(v) = F_c + (F_s - F_c) e^{-(v/v_s)^2} \quad (7)$$

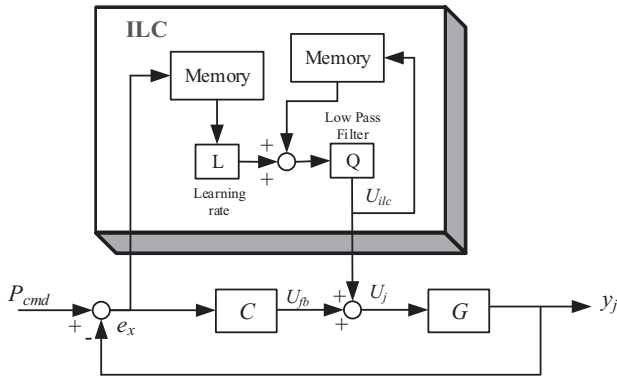
$$F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + \sigma_2 v \quad (8)$$

where  $z$  is the average deflection of the bristles;  $v$  is the relative velocity between two contact surfaces;  $v_s$  is the Stribeck velocity;  $F_c$  is the Coulomb friction;  $F_s$  is the level of stiction force;  $\sigma_0$  is the stiffness;  $\sigma_1$  is the damping coefficient;  $\sigma_2$  is the viscous coefficient. Note that the  $g(v)$  defined by Equation (7) is a positive function and is used to describe the Stribeck effect.

With the values of friction model parameters given, one can use Equations (6)–(8) to estimate the friction force.

#### 3.2. Fuzzy logic-based reinforcement ILC

Iterative Learning Control (ILC) is one of the most commonly used approaches in suppressing external periodic disturbances [13,27–32]. In many motion control applications, the servomechanism is required to perform the same contour-following tasks multiple times. In such a case, it is reasonable to exploit the idea of ILC in suppressing external periodic disturbances so as to improve contour-following accuracy. Figure 3 illustrates the block diagram of a control scheme consisting of a control-law-based ILC and a feedback controller.



**Figure 3.** Block diagram of a control scheme consisting of a control-law-based ILC and a feedback controller.

According to [Figure 3](#), the system output in the  $j$ th iteration is described by

$$y_j = (1 + GC)^{-1}Gu_j + (1 + GC)^{-1}GCP_{cmd} \quad (9)$$

In [Equation \(9\)](#),  $u_j$  is the control output in the  $j$ th iteration and is described by

$$\begin{aligned} u_{j+1}(k) &= u_{ILC(j+1)}(k) + u_{FB(j+1)}(k) \\ &= u_{ILC(j+1)}(k) + Ce_{j+1}(k) \end{aligned} \quad (10)$$

where  $e_{j+1}(k)$  is the tracking error in the  $(j+1)$ th iteration.

In [Equation \(10\)](#),  $u_{FB(j+1)}(k) = C \cdot e_{j+1}(k)$  represents the feedback term, while the feedforward term provided by the ILC is described by

$$u_{ILC(j+1)}(k) = Q[u_{ILC(j)}(k) + L_j e_j(k)] \quad (11)$$

where  $Q$  is a low-pass filter and  $L_j$  is the learning function.

In practice, the value of the learning function  $L_j$  in the  $j$ th iteration has a significant impact on the performance of ILC. If the value of  $L_j$  is too large, the tracking error may diverge eventually. In contrast, if the value of  $L_j$  is too small, the tracking error may converge in a very slow fashion so that it will take a much longer time to achieve satisfactory contour-following accuracy. In addition, if unknown disturbance is introduced in the system during the operation process, an ILC with a fixed learning rate may not be able to react quickly enough to effectively suppress the adverse effects caused by non-periodic disturbance.

In order to overcome the aforementioned difficulty, by exploiting the paradigm of reinforcement learning [33,34], this paper employs a deep learning strategy [35] to adjust the value of the learning function in the  $j$ th iteration. The proposed control scheme is called “fuzzy logic-based reinforcement-ILC” in this paper. In general, the main idea of reinforcement learning is to exploit interactions between “state measurement”, “agent” and “environment” to learn and find solutions.

In this paper, “position feedback” is regarded as “state sensing”, “learning function  $L$ ” is the “agent” and “motion accuracy” is the “environment”. Most importantly, the idea of reinforcement is exploited in this paper to adjust the value of the learning function  $L$  so as to improve contour-following accuracy.

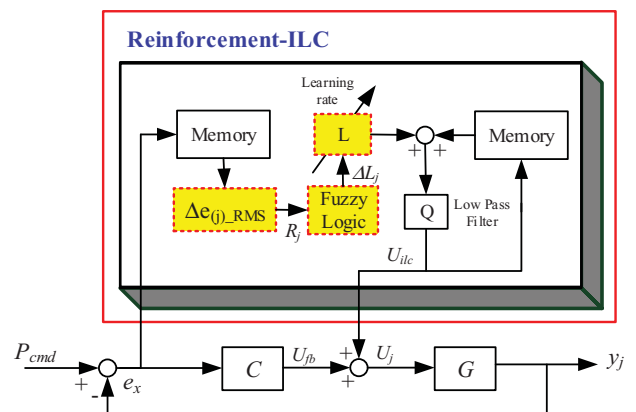
[Figure 4](#) illustrates the block diagram of the control scheme consisting of the proposed RILC and a feedback controller. As indicated in [Figure 4](#), the ILC is aimed at reducing the tracking error of each axis so as to improve contour-following accuracy. In each iteration, the root mean square (RMS) value of the tracking error of the entire contour-following task is calculated. The RMS value of the tracking error of the  $j$ th iteration is compared with that of the  $(j-1)$ th iteration as described by

$$\Delta e_{(j)\_RMS} = e_{(j-1)\_RMS} - e_{(j-2)\_RMS} \quad (12)$$

where  $e_{(j-1)\_RMS}$  (described by [Equation \(13\)](#)) is the RMS value of the tracking error of the entire contour-following task in the  $(j-1)$ th iteration;  $e_{(j-2)\_RMS}$  denotes the RMS value of the tracking error of the entire contour-following task in the  $(j-2)$ th iteration;  $\Delta e_{(j)\_RMS}$  is the difference between  $e_{(j-1)\_RMS}$  and  $e_{(j-2)\_RMS}$ .

$$e_{(j)\_RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_{(j)\_cmd}(i) - y_{(j)}(i))^2} \quad (13)$$

The negative value of  $\Delta e_{(j)\_RMS}$  indicates that the ILC is effective in reducing tracking error. Therefore, there is no need to change the value of the learning function  $L_j$ . In contrast, if the value of  $\Delta e_{(j)\_RMS}$  is positive or zero, it indicates that the ILC is ineffective in reducing tracking error and the performance of ILC deteriorates. As a result, a reinforcement signal  $R_{(j+1)}$  is generated to invoke a change in the value of the learning function  $L_j$ . In this paper, fuzzy logic [36] is



**Figure 4.** Block diagram of a control scheme consisting of the proposed RILC and a feedback controller.

exploited to adjust the value of the learning function  $L_j$  upon receiving a reinforcement signal. In particular, the reinforcement signal  $R_{(j)}$  is described by

$$R_{(j)} = \begin{cases} 0 & \text{if } \Delta e_{(j)\text{-RMS}} \geq e_{\text{limit}} \\ R_{(j)} - K_{et}\Delta e_{(j)\text{-RMS}} & \text{if } \Delta e_{(j)\text{-RMS}} < e_{\text{limit}} \end{cases} \text{ for } j = 1, 2, \dots \quad (14)$$

where  $R_{(j)}$  is the reward in the  $j$ th iteration;  $e_{\text{limit}}$  is the pre-defined threshold;  $K_{et}$  is the tuning gain.

In particular,  $\Delta e_{(j)\text{-RMS}}$  and  $R_{(j)}$  are employed as the two inputs to the fuzzy logic-based RILC learning rate regulator. The membership functions of the fuzzy term sets of  $\Delta e_{(j)\text{-RMS}}$ ,  $R_{(j)}$  and  $\Delta L_{(j)}$  are shown in Figure 5(a-c), respectively.

In addition to the periodic disturbance, the bi-axial motion platform also endures non-periodic disturbance. In general, the over-learning phenomenon of ILC is due to the high-frequency non-periodic disturbance. In the first several iterations of learning, the adverse effect caused by the high-frequency non-periodic disturbance is not significant since the

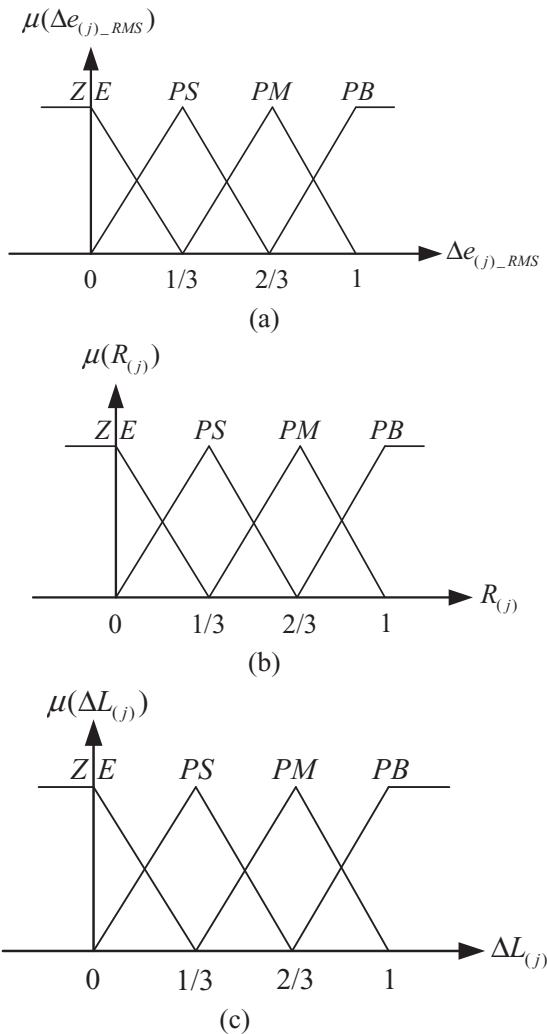


Figure 5. Membership functions of fuzzy term sets (a)  $\Delta e_{(j)\text{-RMS}}$ , (b)  $R_{(j)}$ , (c)  $\Delta L_{(j)}$ .

Table 1. Rule table for fuzzy logic-based RILC learning rate regulator.

$\Delta L_{(j)}$		$\Delta e_{(j)\text{-RMS}}$			
		ZE	PS	PM	PB
$R_{(j)}$	PB	ZE	PM	PB	PB
	PM	ZE	PS	PM	PB
	PS	ZE	PS	PS	PM
	ZE	ZE	ZE	ZE	ZE

periodic disturbance is yet to be effectively suppressed by the ILC. After several iterations, the periodic disturbance can be suppressed by the ILC, while the adverse effect due to the high-frequency non-periodic disturbance will not be effectively suppressed if a large learning rate is used. As a result, if both the values of  $e_{(j)\text{-RMS}}$  and  $R_{(j)}$  are “positive big”, one can conclude that the learning process transpires well enough that a larger learning rate can be adopted. In contrast, if both the values of  $e_{(j)\text{-RMS}}$  and  $R_{(j)}$  are very small and close to “zero”, it suggests that an “over-learning phenomenon” may occur. In this case, a smaller learning rate should be used. Based on the above observations and discussions, a rule table for the proposed fuzzy logic-based RILC regulator is constructed as shown in Table 1.

In this paper, the output of the fuzzy logic-based RILC learning rate regulator  $\Delta L_{(j)}$  in the  $j$ th iteration is calculated using the weighted average defuzzification method [37].

$$\Delta L_{(j)} = \frac{\sum_{p=1}^n \mu(C_p) C_p}{\sum_{p=1}^n \mu(C_p)} \quad (15)$$

After obtaining the value of  $\Delta L_{(j)}$ , one can update the learning function of RILC for the  $(j + 1)$ th iteration using

$$L_{(j+1)} = \Delta L_{(j)} L_{\text{initial}} \quad (16)$$

In general, to speed up the convergence process, a larger value of the learning function for the first iteration is chosen. In addition, the value of  $L_j$  in all iterations is limited to between zero and one.

#### 4. Proposed control scheme for contour-following of an X-Y motion stage

Figure 6 illustrates the block diagram of the proposed fuzzy logic-based RIL-CCCMAC with a LuGre friction model-based compensation term. In addition to a commonly used feedback controller and a command feedforward compensation term, the proposed fuzzy logic-based RIL-CCCMAC also exploits the advantages of ILC, cross-coupled control and CMAC. In particular, a control law-based ILC is employed in the outer position loop to reduce the tracking error of each axis caused by external periodic disturbance resulting from repetitive contour-following motions.

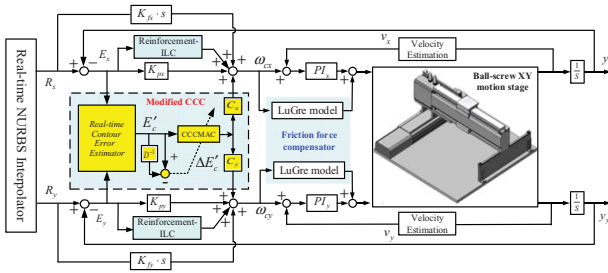


Figure 6. Block diagram of the proposed RIL-CCMAC.

In addition, based on the contour error information provided by the parameter-based contour error estimation algorithm, a CCC is exploited in the position loop to reduce contour-following errors caused by incompatibility among different axes. Moreover, a CMAC is employed to replace the CCC  $K_c$  in the original CCC structure to adjust the amount of compensation to each axis. As for the inner velocity loop, in addition to the commonly used Proportional-Integral (PI)-type feedback controller, a LuGre friction model-based compensation term is adopted to compensate for the adverse effects due to friction.

Suppose that the modified CCC (i.e. a CCC used in conjunction with the parameter-based contour error estimation algorithm as shown in Figure 1) indicated by the dotted-line rectangle in Figure 6 is inactive at first. According to Figure 6, one will have

$$\omega_{cx} = sK_{fx}R_x + Q[u_{ILC_{jx}}(k) + LE_{jx}(k)] + K_{px}E_{(j+1)x}(k) \quad (17)$$

$$\omega_{cy} = sK_{fy}R_y + Q[u_{ILC_{jy}}(k) + LE_{jy}(k)] + K_{py}E_{(j+1)y}(k) \quad (18)$$

where  $\omega_{cx}$  and  $\omega_{cy}$  are the velocity commands provided by the outer position loop controller for  $x$ -axis and  $y$ -axis, respectively;  $E_x$  and  $E_y$  represent the tracking errors of  $x$ -axis and  $y$ -axis, respectively.

Now, consider the modified CCC indicated by the dotted-line rectangle in Figure 6. Based on the contour error  $E'_c$  estimated using the parameter-based contour error estimation algorithm, Equation (19) is used to update the weights of CMAC [18].

$$w_j^{ccc}(k) = w_j^{ccc}(k-1) + \lambda_w^{ccc} \frac{b_{s,j}^{ccc}(\Delta E'_c(k-1))}{\sum_{j=1}^{N_m} b_{s,j}^{ccc}} \quad (19)$$

where

$$\Delta E'_c(k) = E'_c(k) - E'_c(k-1) \quad (20)$$

In addition, the amount of compensation force provided by the CCCMAC to the motions in  $x$ -axis

and  $y$ -axis is calculated using Equations (21) and (22), respectively.

$$U_{cmac\_ccc_x} = U_{cmac\_ccc} \times (-\sin(\theta)) \quad (21)$$

$$U_{cmac\_ccc_y} = U_{cmac\_ccc} \times \cos(\theta) \quad (22)$$

In Equations (21) and (22),  $\theta$  is the angle between the tangent vector at the current position command and  $x$ -axis.

Suppose that the modified CCC in Figure 6 is activated. Then,  $\omega_{cx}$  and  $\omega_{cy}$  in Figure 6 can be rewritten as

$$\omega_{cx} = V_{ffx} + U_{fbx} + U_{ilcx} + U_{ccc\_cmacx} \quad (23)$$

where the position loop control laws for the  $x$ -axis are

$$V_{ffx} = sK_{fx}R_x \quad (24)$$

$$U_{fbx} = K_{px}E_{(j+1)x}(k) \quad (25)$$

$$U_{ilcx} = Q[u_{ILC_{jx}}(k) + LE_{jx}(k)] \quad (26)$$

In addition

$$\omega_{cy} = V_{ffy} + U_{fby} + U_{ilcy} + U_{ccc\_cmacy} \quad (27)$$

where the position loop control laws for the  $y$ -axis are

$$V_{ffy} = sK_{fy}R_y \quad (28)$$

$$U_{fby} = K_{py}E_{(j+1)y}(k) \quad (29)$$

$$U_{ilcy} = Q[u_{ILC_{jy}}(k) + LE_{jy}(k)] \quad (30)$$

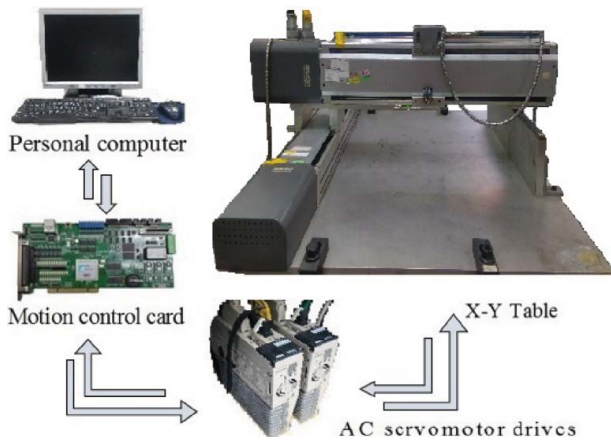
## 5. Experimental results

### 5.1. Experimental set-up

In order to verify the effectiveness of the proposed approach, several contour-following tasks have been conducted on a bi-axial motion platform as shown in Figure 7. This bi-axial motion platform consists of two ball screws actuated by two Panasonic 400 W AC servomotors (MSDA041A1A) with built-in  $2500 \times 4$  pulses/rev encoders. In addition, the pitch of the ball screw is 1 cm/rev. The servo drive of each AC servomotor is set to torque mode throughout all the contour-following experiments and the sampling time is set to 1 ms.

### 5.2. Contour-following experiment of bi-axial motion stage

The desired circle-shaped contour used in each contour-following experiment is represented in Non-



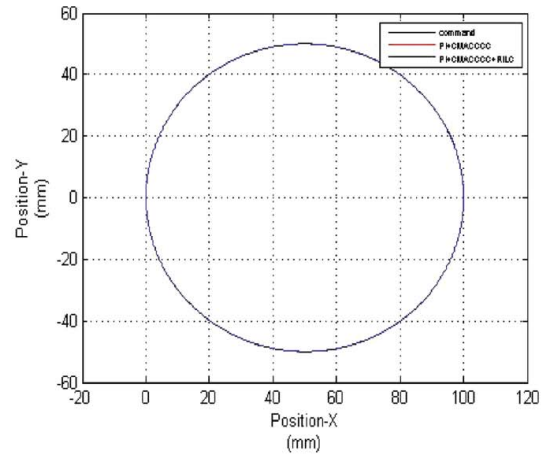
**Figure 7.** Bi-axial motion platform used in the contour-following experiments.

Uniform Rational B-Splines (NURBS) form as shown in Figure 8(a). A NURBS interpolator with the consideration of s-curve acceleration/deceleration motion planning is used to generate position commands for the  $x$ -axis and  $y$ -axis as shown in Figure 8(b,c) [38]. A total of three different control schemes for the velocity loop are tested in each contour-following experiment. They are: control scheme #1: PI-type velocity loop feedback controller with CCCMAC controller; control scheme #2: CCCMAC combined with an ILC and a PI-type velocity loop feedback controller; control scheme #3: the proposed fuzzy logic-based RIL-CCCMAC combined with a PI-type velocity loop feedback controller.

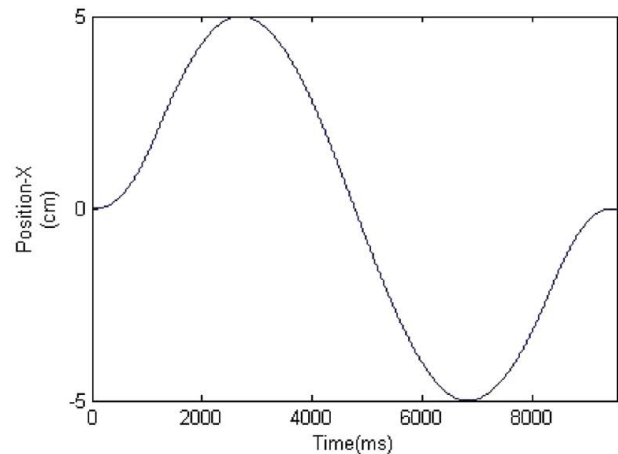
Experimental results of tracking errors for these three different control schemes are shown in Figures 9 and 10. Performance indices of these contour-following experiments are listed in Tables 2 and 3.

In particular, from Figure 9(a), one can find that the maximum tracking error on the  $x$ -axis for the case of control scheme #1 is around  $\pm 0.325$  mm, while Figure 9(b) shows that the maximum tracking error on the  $x$ -axis for the case of control scheme #2 is around  $\pm 0.13$  mm. Experimental results shown in Figure 9(a,b) indicates that control scheme #2 outperforms control scheme #1. These findings are not surprising, since in this paper the input to the CMAC is the velocity command; namely, CMAC possesses the ability of friction compensation as well as command feedforward. However, control scheme #2 suffers from the aforementioned over-learning phenomenon since the tracking errors start rising after several iterations.

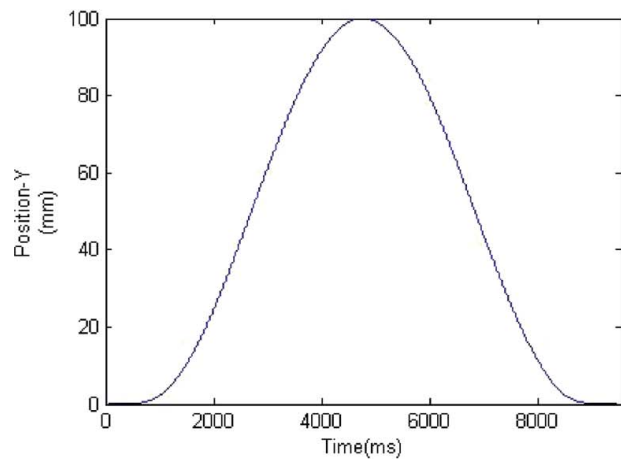
Figure 9(c,d) shows the tracking error and their RMS values on the  $x$ -axis for control scheme #3, respectively. In particular, as shown in Figure 9(d,e), the RMS value of tracking error and the value of learning function keep decreasing through iterations. In addition, Tables 2 and 3 summarize the performance comparison of contour-following among the three



(a)



(b)

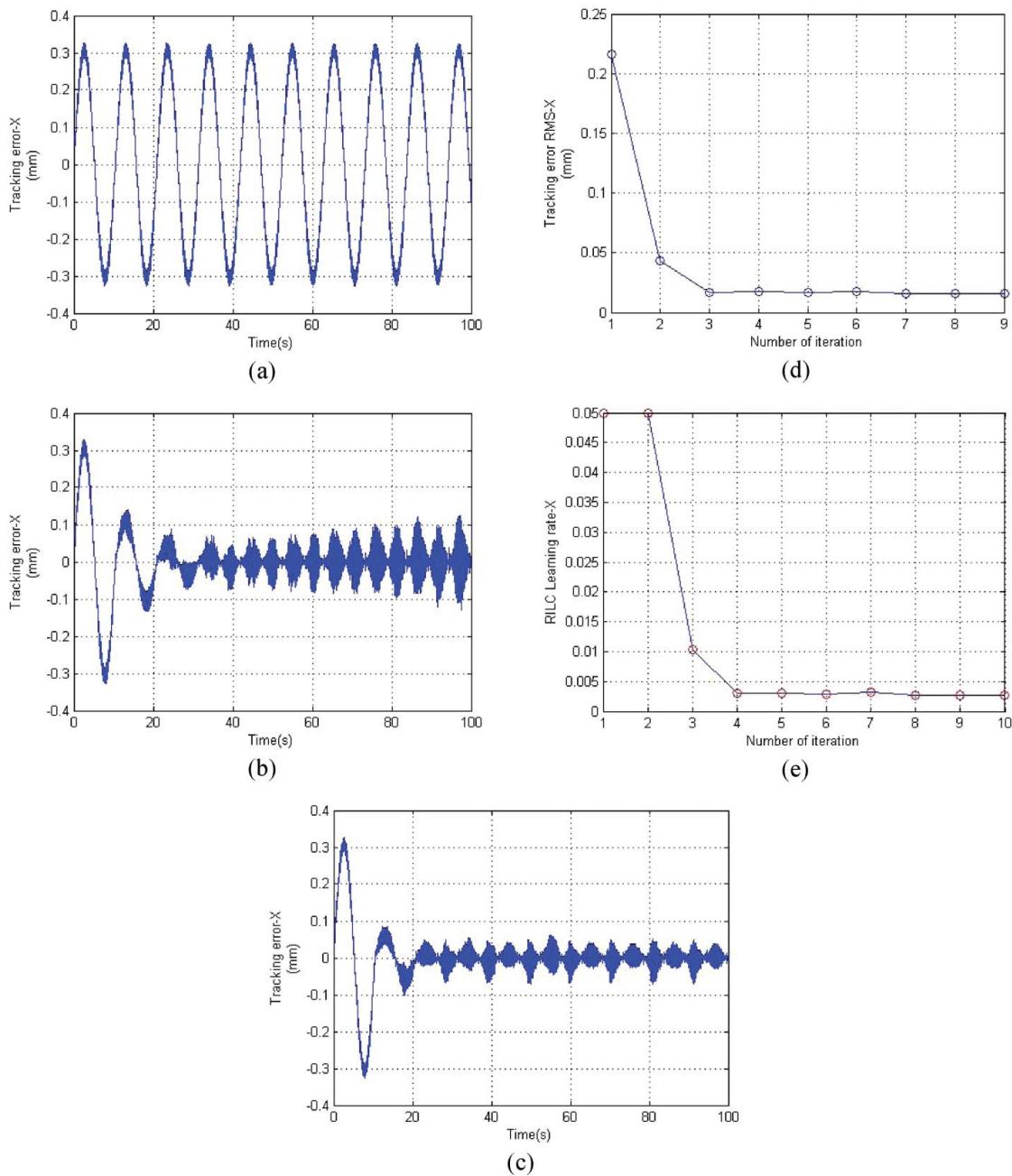


(c)

**Figure 8.** (a) Circle-shaped contour represented in NURBS form, (b) Position commands for  $x$ -axis, (c) Position commands for  $y$ -axis.

tested control schemes. Clearly, Tables 2 and 3 and the tracking errors shown in Figure 9 all indicate that the proposed fuzzy logic-based RIL-CCCMAC combined with a PI-type velocity loop feedback controller outperforms the other two control schemes also tested in the experiments. In order to verify the effectiveness of the proposed fuzzy logic-based RIL-CCCMAC scheme





**Figure 9.** (a) Tracking error of control scheme #1, (b) Tracking error of control scheme #2, (c) Tracking error of control scheme #3, (d) RMS of the tracking error of control scheme #3, (e) The value of learning function of control scheme #3.

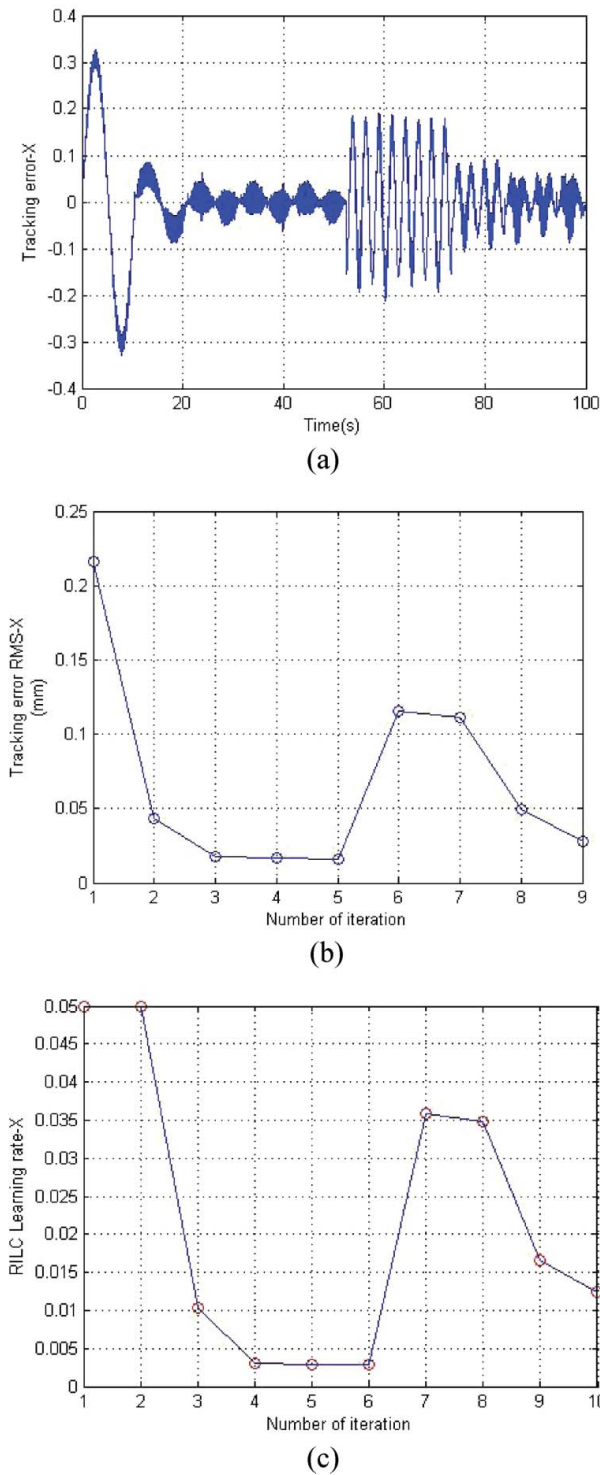
in coping with non-periodic disturbance, a signal that emulates the external disturbance is intentionally added to the bi-axial motion platform in the sixth iteration.

Experimental results shown in Figure 10(a,b) reveal that the RMS value of tracking error in the seventh iteration is larger than that of the sixth iteration so that a reinforcement signal is generated to invoke a change in the value of the learning function. In this case, the proposed fuzzy logic-based RILC learning rate regulator is exploited to adjust the value of the learning function as shown in Figure 10(c). Experimental results shown in Figure 10(b,c) indicate that the RMS value of tracking error starts decreasing after the value of the

**Table 2.** Tracking error comparison among the three tested control schemes.

	Tracking error of X-axis		
	PI + CCCMAC	PI + CCCMAC + ILC	Proposed
RMS ( $\mu\text{m}$ )	215.874	34.517	15.221
AIAE ( $\mu\text{m}$ )	194.298	25.115	11.314
Max ( $\mu\text{m}$ )	325.204	130.351	68.847
	Tracking error of Y-axis		
	PI + CCCMAC	PI + CCCMAC + ILC	Proposed
RMS ( $\mu\text{m}$ )	217.527	31.021	22.126
AIAE ( $\mu\text{m}$ )	196.408	23.088	16.692
Max ( $\mu\text{m}$ )	320.457	217.185	122.049

Note: AIAE, average of the integral of the absolute error.



**Figure 10.** Experimental results of the case that the bi-axial motion platform experiences an external disturbance in the sixth iteration (a) Tracking error of control scheme #3, (b) RMS of the tracking error of control scheme #3, (c) The value of learning function of control scheme #3.

learning function is adjusted. The above observations suggest that the proposed fuzzy logic-based RIL-CCCMAC scheme is indeed capable of suppressing external disturbance.

Note that the formulas used to compute the tracking error and performance indices listed in

**Table 3.** Contour error comparison among the three tested control schemes.

	Contour error		
	PI + CCCMAC	PI + CCCMAC + ILC	Proposed
RMS ( $\mu\text{m}$ )	16.455	13.058	8.524
AIAE ( $\mu\text{m}$ )	17.310	9.828	6.753
Max ( $\mu\text{m}$ )	99.743	51.905	41.393

Tables 2 and 3 are described by Equations (13), (31) and (32).

$$e_{\text{AIAE}} = \frac{1}{n} \sum_{j=1}^n |e(j)| \quad (31)$$

$$e_{\text{MAX}} = \max_{1 \leq j \leq n} |e(j)| \quad (32)$$

## 6. Conclusion

The contour-following control scheme proposed in this paper consists of a position outer loop controller and a velocity inner loop controller. In particular, the position loop is composed of a P-type feedback loop controller, a velocity feedforward term, a fuzzy logic-based RILC and a CCCMAC. The fuzzy logic-based RILC is mainly used to compensate for external periodic and non-periodic disturbance, while the CCCMAC is proposed to deal with the problem of dynamics incompatibility between different axes by exploiting an accurate contour error estimation algorithm. The velocity loop consists of a PI-type feedback loop controller and a LuGre model-based friction compensation scheme. Experimental results of several contour-following tasks indicate that the proposed approach outperforms several existing motion control schemes also tested in the contour-following experiments. In particular, by adjusting the learning rate online, the proposed RILC is able to effectively suppress external periodic and non-periodic disturbance so as to improve contour-following accuracy.

## Acknowledgments

The authors would like to thank the MOST of Taiwan and Delta Electronics for their support of this research under Grant MOST 103-2221-E-006-185-MY2 and B104-K79.

## Disclosure statement

No potential conflict of interest was reported by the authors.

## Funding

This work was supported by the MOST of Taiwan [grant number MOST 103-2221-E-006-185-MY2] and Delta Electronics [grant number B104-K79].

## References

- [1] Hiroshi F, Tadashi T. High-precision control of ball-screw-driven stage based on repetitive control using n-times learning filter. *IEEE Trans Ind Electron.* **2014**;61:3694–3703.
- [2] Cheng G, Peng K, Chen B-M, et al. Improving transient performance in tracking general references using composite nonlinear feedback control and its application to high-speed XY-table positioning mechanism. *IEEE Trans Ind Electron.* **2007**;54:1039–1051.
- [3] Maebashi W, Ito K, Iwasaki M. Practical feedback controller design for micro-displacement positioning in table drive systems. *Automatika.* **2013**;54:9–18.
- [4] Li Y, Xu Q. Design and robust repetitive control of a new parallel-kinematic XY piezostage for micro/nanomanipulation. *IEEE Trans Mech.* **2012**;17:1120–1132.
- [5] Hu C, Yao B, Chen Z, et al. Adaptive robust repetitive control of an industrial biaxial precision gantry for contouring tasks. *IEEE Trans Control Syst Technol.* **2011**;19:1559–1568.
- [6] Yang K-C, Hsieh C. Nanometer positioning of a dual-drive gantry table with precise yaw motion control. *J CSME.* **2015**;36(2):107–117.
- [7] Tomizuka M. Zero phase error tracking algorithm for digital control. *ASME J Dyn Syst Meas Control.* **1987**;109:65–68.
- [8] Koren Y. Cross-coupled biaxial computer control for manufacturing systems. *ASME J Dyn Syst Meas Control.* **1980**;102:265–272.
- [9] Su K-H, Cheng M-Y. Contouring accuracy improvement using cross-coupled control and position error compensator. *Int J Mach Tools Manu.* **2008**;48:1444–1453.
- [10] Golubovic E, Baran EA, Sabanovic A. Contouring controller for precise motion control systems. *Automatika.* **2013**;54:19–27.
- [11] Chen H-R, Cheng M-Y, Wu C-H, et al. Real time parameter based contour error estimation algorithms for free form contour following. *Int J Mach Tools Manu.* **2016**;102:1–8.
- [12] Wen C-M, Cheng M-Y. Contouring accuracy improvement of a piezo-actuated micro motion stage based on fuzzy cerebellar model articulation controller. *Control Eng Practice.* **2012**;20:1195–1205.
- [13] Chen C-L, Li K-S. Observer-based robust AILC for robotic system tracking problem. *J CSME.* **2009**;30(6):483–491.
- [14] Lin C-M, Li H-Y. Adaptive dynamic sliding-mode fuzzy CMAC for voice coil motor using asymmetric Gaussian membership function. *IEEE Trans Ind Electron.* **2014**;61:5662–5671.
- [15] Lin C-M, Li H-Y. Intelligent control using the wavelet fuzzy CMAC backstepping control system for two-axis linear piezoelectric ceramic motor drive systems. *IEEE Trans Fuzzy Syst.* **2014**;22(4):791–802.
- [16] Chen C-K, Lin C-J, Hwang J, et al. The iterative learning control of a Stewart platform system. *J CSME.* **2013**;34(1):21–30.
- [17] Lin C-S, Chiang C-T. Learning convergence of CMAC technique. *IEEE Trans Neural Netw.* **1997**;8(6):1281–1292.
- [18] Wen C-M, Cheng M-Y. Development of a recurrent fuzzy CMAC with adjustable input space quantization and self-tuning learning rate for control of a dual-axis piezoelectric actuated micromotion stage. *IEEE Trans Ind Electron.* **2013**;60:5105–5115.
- [19] Shi G-L, Shen W. Hybrid control of a parallel platform based on pneumatic artificial muscles combining sliding mode controller and adaptive fuzzy CMAC. *Control Eng Practice.* **2013**;21:76–86.
- [20] de Wit CC, Olsson H, Astrom KJ, et al. A new model for control of systems with friction. *IEEE Trans on Autom Control.* **1995**;40(3):419–425.
- [21] Olsson H, Åström KJ, Canudas de Wit C, et al. Friction models and friction compensation. *Eur J Control.* **1998**;4:176–195.
- [22] Åström KJ, Canudas de Wit C. Revisiting the LuGre friction model. *IEEE Control Syst Mag.* **2008**;28(6):101–114.
- [23] Chen C-Y, Cheng M-Y. Adaptive disturbance compensation and load torque estimation for speed control of a servomechanism. *Int J Mach Tools Manu.* **2012**;59:6–15.
- [24] Cheng M-Y, Lee C-C. Motion controller design for contour-following tasks based on real-time contour error estimation. *IEEE Trans Ind Electron.* **2007**;54:1686–1695.
- [25] Albus JS. A new approach to manipulator control: the cerebellar model articulation controller (CMAC). *ASME J Dyn Syst Meas Control.* **1975**;97:220–227.
- [26] Albus JS. Data storage in the cerebellar model articulation controller (CMAC). *ASME J Dyn Syst Meas Control.* **1975**;97:228–233.
- [27] Bristow DA, Tharayil M, Alleyne AG. A survey of iterative learning control. *IEEE Control Syst Mag.* **2006**;26:96–114.
- [28] Barton KL, Alleyne AG. A cross-coupled iterative learning control design for precision motion control. *IEEE Trans Control Syst Technol.* **2008**;16:1218–1231.
- [29] Shen D, Xu Y. Iterative learning control for discrete-time stochastic systems with quantized information. *IEEE/CAA J Autom Sinica.* **2016**;3(1):59–67.
- [30] Chu B, Owens DH, Freeman CT. Iterative learning control with predictive trial information: convergence, robustness, and experimental verification. *IEEE Trans Control Syst Technol.* **2016**;24(3):101–1108.
- [31] Meng D, Moore KL. Robust iterative learning control for nonrepetitive uncertain systems. *IEEE Trans Autom Control.* **2017**;62(2):907–913.
- [32] Shen D, Han J, Wang Y. Stochastic point-to-point iterative learning tracking without prior information on system matrices. *IEEE Trans Autom Sci Eng.* **2017**;14(1):376–382.
- [33] Sutton RS, Barto AG. Reinforcement learning: an introduction. 2nd ed. Cambridge (MA): MIT Press (Bradford Book); 1998. (Adaptive computation and machine learning series).
- [34] Wiering M, van Otterlo M. Reinforcement learning: state-of-the-art. Berlin Heidelberg (Germany): Springer-Verlag; **2012**.
- [35] Silver D, Huang A, Maddison CJ, et al. Mastering the game of go with deep neural networks and tree search. *Nature.* **2016**;529:484–489.
- [36] Lee S-C, Shih C-L. Optimal single input PID-type fuzzy logic controller. *J Chinese Inst Eng.* **2012**;35(4):413–420.
- [37] Lee C-C. Fuzzy logic in control system: fuzzy logic controller – part II. *IEEE T Syst Man Cyb.* **1990**;20:419–435.
- [38] Cheng M-Y, Tsai M-C, Kuo J-C. Real-time NURBS command generators for CNC servo controllers. *Int J Mach Tools Manu.* **2002**;42:801–813.