

## Article

# Direct Fuzzy CMAC Sliding Mode Trajectory Tracking for Biaxial Position System

Wei-Lung Mao <sup>1,\*</sup>, Yu-Ying Chiu <sup>1</sup>, Bing-Hong Lin <sup>1</sup>, Wei-Cheng Sun <sup>1</sup> and Jian-Fu Tang <sup>2,\*</sup> 

<sup>1</sup> Department of Electrical Engineering, Graduate School of Engineering Science and Technology, National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou 64002, Taiwan; D10910201@yuntech.edu.tw (Y.-Y.C.); D10910203@yuntech.edu.tw (B.-H.L.); M10612060@yuntech.edu.tw (W.-C.S.)

<sup>2</sup> Department of Electro-Optical Engineering, National Taipei University of Technology, Taipei 10608, Taiwan

\* Correspondence: wlmao@yuntech.edu (W.-L.M.); jftang@ntut.edu.tw (J.-F.T.); Tel.: +886-5-534-2601 (ext. 4222) (W.-L.M.); +886-2-2771-2171 (ext. 4693) (J.-F.T.)

**Abstract:** High-precision trajectory control is considered as an important factor in the performance of industrial two-axis contour motion systems. This research presents an adaptive direct fuzzy cerebellar model articulation controller (CMAC) sliding mode control (DFCMACSMC) for the precise control of the industrial XY-axis motion system. The FCMAC was utilized to approximate an ideal controller, and the weights of FCMAC were on-line tuned by the derived adaptive law based on the Lyapunov criterion. With this derivation in mind, the asymptotic stability of the developed motion system could be guaranteed. The two-axis stage system was experimentally investigated using four contours, namely, circle, bowknot, heart, and star reference contours. The experimental results indicate that the proposed DFCMACSMC method achieved the improved tracking capability, and so reveal that the DFCMACSMC scheme outperformed other schemes of the model uncertainties and cross-coupling interference.

**Keywords:** direct fuzzy cmac sliding mode control (DFCMACSMC); permanent magnet synchronous motor (PMSM); precision motion control; position feedback sensor; trajectory tracking control



**Citation:** Mao, W.-L.; Chiu, Y.-Y.; Lin, B.-H.; Sun, W.-C.; Tang, J.-F. Direct Fuzzy CMAC Sliding Mode Trajectory Tracking for Biaxial Position System. *Energies* **2021**, *14*, 7802. <https://doi.org/10.3390/en14227802>

Academic Editor: Florin Nicolae Jurca

Received: 29 September 2021

Accepted: 16 November 2021

Published: 22 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Precision motion systems are being used extensively in precision fields, such as computer numerical control (CNC) machineries, IC manufacturing, micro-mechanical assembly, ultraprecision measurement, and microsurgery. Machine systems are generally designed to reduce tracking errors with respect to each feed drive axis.

The micro-meter level system is becoming increasingly important in satisfying the requirements for mass production in industrial 4.0. The permanent magnet synchronous motor (PMSM) [1–5] is a device designed by using permanent magnets embedded in the steel rotor to create a constant magnetic field. It provides for several better properties, including low torque ripple, high efficiency, high power density, low maintenance cost, low heat generation, and so forth. However, it still has some inherent disadvantages, such as complex control systems, higher cost, and low reliability. Moreover, the servo-performance of the biaxial motion control system is affected by backlash, nonlinear friction characteristics, unmodeled dynamics, and other time-varying conditions. To better compensate these uncertainties and nonlinearities, a better control scheme should be used for motion control systems.

Various control schemes have been proposed to improve the control properties and tackle the uncertainty challenge considering the adaptive techniques [1–9]. Based on the concept of a learning-based approach for error compensation, repetitive perfect tracking control [1] with an n-times learning filter was proposed for the X-Y stage target tracking. The data-based friction model is used for rolling compensation. A FPGA-based self-tuning

PID motion control system [2] was proposed to enhance the system performance of the X-Y table controller. This system utilizes a radial basis function (RBF) neural network with a gradient learning algorithm to reduce trajectory errors. The adaptive recurrent-neural-network (ARNN) motion control system [3] for a biaxial motion mechanism was proposed in the computer numerical control (CNC) machine. The tracking performance was substantially improved, and the robustness to uncertainties, including the cross-coupled interference and friction torque, could be obtained as well. A robust fuzzy neural network controller [4] with nonlinear disturbance observer (RFNNCND0) is proposed for the precision control of a biaxial motion system. From the experimental results, the contour tracking performance of the two-axis motion control system was significantly improved, and the robustness could be obtained as well using the proposed RFNNCND0 control system. The adaptive learning algorithms considered to be able to learn the parameters of the RNN on-line were derived using the Lyapunov stability theorem. El-Sousy [6] proposed a mixed  $H_2/H_\infty$  controller, a self-organizing recurrent fuzzy-wavelet-neural-network controller (SORFWNNC) and a robust controller. The sufficient conditions were developed for the adaptive mixed  $H_2/H_\infty$  to help track problems in terms of a pair of coupled algebraic equations instead of coupled nonlinear differential equations. Sliding mode control (SMC) is a nonlinear control algorithm that employs discontinuous control to force system state trajectories to lie on some prescribed sliding surface. To resolve this difficulty, an adaptation scheme which can improve the robustness of adaptive fuzzy control is proposed in X-Y stage dynamic systems.

The cerebellar model articulation controller (CMAC) structure has been applied widely to regulate and control the complex dynamical systems owing to its fast learning properties, better generalization capabilities, and simple computation when compared with neural networks [9–14]. A trained CMAC can approximate the nonlinear functions in a generalized lookup-table manner over a domain to any desired accuracy. The robust cerebellar model articulation controller (RCMAC) [9] was developed to achieve  $H_\infty$  robust tracking performance. This design technique is generally applied to control a chaotic circuit system to verify its effectiveness. A modified adaptive Fuzzy CMAC scheme [13] was proposed to solve the chattering problem without sacrificing the performance for a class of nonlinear systems. This scheme with the saturation compensation can prevent the chattering and steered the tracking error to converge exponentially to a residual set whose size could be adjusted. A deep CMAC (DCMAC) framework, which stacks several layers of single-layered CMACs, is proposed [15] for adaptive noise cancellation. A backpropagation algorithm is derived to train the DCMAC effectively and efficiently. The control algorithm [16] is the core of the dynamic positioning system of USV. This paper adopts the sliding mode control (SMC) algorithm based on a Cerebellar Model Articulation Controller. The tracking differentiator was added to eliminate the large chattering in the initial stage of the system. This research [17] proposes a new self-organizing FCMAC (NSOFC) for uncertain nonlinear systems. The proposed method uses an integrated sliding surface and overlaps the prior and present GMFs for each layer to form a sum of the two states that is used to predict the error values. This research [18] proposes a control system consisting of a novel type of fuzzy neural network and a robust compensator controller for robot systems. The new fuzzy neural network is implemented by integrating a number of key components embedded in a Type-2 fuzzy cerebellar model articulation controller (CMAC) and a brain emotional learning controller (BELC) network, thereby mimicking an ideal sliding mode controller. Clearly then, a synopsis of the above research studies reveals that the adaptive structures with nonlinear approximation ability indeed improve the control performances.

The direct fuzzy CMAC sliding mode control (DFCMACSMC) is developed for the two-axis PMSM positioning system. The proposed control structure consists of a two-dimensional fuzzy CMAC structure with Gaussian membership functions. This control system combines the advantages of fuzzy inference system and CMAC structure. The parameter update rules for a DFCMAC with two inputs and one output are derived, and the stability of the proposed learning algorithm is proven via a Lyapunov function.

Trajectory planning mainly depends on the NURBS curve design method. The servo table is experimentally investigated with four typical contours, namely the circle, bowknot, heart, and star reference contours. The simulation results and experimental results are presented to show that the proposed DFCMAC-based sliding mode controller indeed better accomplished the tracking performances with regard to model uncertainties. The real-time approach needs to be considered, as does the limitations. We will demonstrate that the proposed adaptive DFCMAC algorithms can provide a successful industrial application for biaxial PMSM-actuated motion tables for tracking different reference trajectories. In the future, the microcontroller-based motion controllers will be designed and implemented for industrial applications.

This paper is structured as follows. Section 2 highlights the importance of the system architecture of the X-Y stage. This is followed by Section 2.1 with a description of the two-dimensional FCMAC architecture given. The proposed DFCMAC design is given in Section 2.2. Section 3 presents the results of the simulation and experiment. The conclusion is drawn in Section 4.

## 2. Materials and Methods

### 2.1. System Model

The dynamic model of the single-axis motion stage is given as

$$m\ddot{x} + b\dot{x} + F_f = w(u) + F_u, \quad (1)$$

where  $m$  is the total equivalent mass,  $x$  is the stage displacement,  $b$  is the viscous friction coefficient,  $F_f$  is the equivalent nonlinear friction force,  $u$  is the control input,  $w(u)$  denotes the input nonlinear backlash function, and  $F_u$  is the sum of the modelling error, static load, and bounded external disturbances. The friction force  $F_f$  is represented as:

$$F_f = F_c \operatorname{sgn}(\dot{x}) + (F_s - F_c) \exp\left[-(\dot{x}/v_s)^2\right] \operatorname{sgn}(\dot{x}) + K_v \dot{x}, \quad (2)$$

where  $F_c$  is the Coulomb friction,  $F_s$  is the static friction,  $v_s$  is the Stribeck velocity parameter,  $\dot{x}$  is the linear velocity of the X- (Y-) axis,  $K_v$  is the coefficient of viscous friction, and  $\operatorname{sgn}(\cdot)$  is a sign function. The nonlinear control input function  $w(u)$  can be expressed as:

$$w(u) = cu + f_b, \quad (3)$$

where  $c$  is a positive constant, and  $f_b$  denotes the unknown backlash nonlinear function. This dynamic system can be rewritten as:

$$\ddot{x} = -g(b\dot{x} + F_f) + gcu + D, \quad (4)$$

With  $g = 1/m$ ,  $D = g(f_b + f_u)$  where  $D$  is the lumped disturbance parameter bounded by  $D^U$ . It is assumed that  $|D| < D^U$ . The experimental system of the proposed XY motion stage is depicted in Figure 1a,b.

### 2.2. Two-Dimensional FCMAC Architecture

A basic CMAC architecture [9] is shown in Figure 2. The mapping relation is described as:

$$Y : y(s) = \mathbf{A}\mathbf{W}, \quad (5)$$

where  $s$  is the continuous  $n$ -dimensional input space,  $\mathbf{A}$  is the association memory with  $n_a$  elements,  $\mathbf{R}$  is the receptive field with  $n_h$  elements,  $\mathbf{W}$  is the weight memory with  $n_h$  adjustable weights, and  $Y$  is the  $m$ -dimensional output space. The mapping structure of the two-dimensional CMAC is depicted in Figure 3. In this case, two input variables,  $s_1$  and  $s_2$ , are both quantized into nine intervals and divided into three blocks. Namely,  $s_1$  is divided into three blocks named A, B, C, and  $s_2$  is divided into a, b, c in the first layer. Then,  $s_1$  is

shifted an interval to D, E, F, and so is  $s_2$  that shifted to d, e, f in the second layer, and so on. Therefore, Aa, Ab, Ac, Ba, Bb, Bc, Ca, Cb, Cc, which are called hypercubes, are obtained in the first layer and the same as Dd, De, Df, Ed, Ee, Ef, Fd, Fe, Ff that are in the second layer. Each hypercube is composed of the blocks in the same layer of two input variables; that is to say, it is impossible to obtain a hypercube, such as Ad, Ei, or Gc. In the case, there are 81 states, 36 hypercubes, and four layers, and each state activates four association cells that exists in different hypercubes. It means that  $k = 81$ ,  $m = 4$ , and  $n = 36$ . The actual output of each state can be expressed as:

$$y_{sk} = \mathbf{a}_{sk}^T \mathbf{W} = [ a_{sk,1} \quad a_{sk,2} \quad \dots \quad a_{sk,n} ] \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \sum_{j=1}^n a_{sk,j} w_j \tag{6}$$

where  $\mathbf{a}_{sk}^T$  is the association index vector of the  $k$ -th state and  $\mathbf{W}$  is the weight vector that comprises weights stored in memory. The collection of Equation (6) can be represented as:

$$\mathbf{Y} = \begin{bmatrix} y_{s1} \\ y_{s2} \\ \vdots \\ y_{sk} \end{bmatrix} = \begin{bmatrix} a_{s1,1} & a_{s1,2} & \dots & a_{s1,n} \\ a_{s2,1} & a_{s2,2} & \dots & a_{s2,n} \\ \vdots & & & \\ a_{sk,1} & a_{sk,2} & & a_{sk,n} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} a_{sk}^T \\ a_{sk}^T \\ \vdots \\ a_{sk}^T \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \mathbf{A} \mathbf{W} \tag{7}$$

and the matrix  $\mathbf{A}$  is expressed as:

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_{s1}^T \\ \mathbf{a}_{s2}^T \\ \vdots \\ \mathbf{a}_{s81}^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ \vdots & & & & & & & & & & & & \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 & 0 & 0 & \dots & 1 \end{bmatrix}_{81 \times 36} \tag{8}$$

with

$$\mathbf{a}_{s21}^T = [ 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad \dots \quad 0 \quad 1 \quad 0 \quad \dots \quad 0 \quad 1 \quad 0 \quad \dots \quad 0 \quad 1 \quad 0 \quad \dots \quad 0 ]$$

where it means that state (3,3) activates the hypercubes 5(Bb), 14(Ee), 19(Gg), and 28(Jj). The related weights are adjusted by the update law as

$$\mathbf{W}(i) = \mathbf{W}(i - 1) + \Delta \mathbf{W} = \mathbf{W}(i - 1) + \frac{\alpha}{m} \mathbf{a}_{sk} (y_{sk}^d - \mathbf{a}_{sk} \mathbf{W}(i - 1)) \tag{9}$$

where  $i$  means the  $i$ -th learning step,  $y_{sk}^d$  is the desired output, and  $\alpha$  is the learning rate. Through the above training process, FCMAC [9–12] is able to approximate the desired trajectory. The mapping structure of the two-dimensional CMAC with the Gaussian basis function is shown in Figure 4. Here, the Gaussian function is adopted as the receptive field basis function, and it is given as:

$$\varphi_{ij}(s_i) = \exp \left[ \frac{-(s_i - m_{ij})^2}{\sigma_{ij}^2} \right] \quad \text{for } j = 1, 2, \dots, 12 \tag{10}$$

where  $\varphi_{ij}(s_i)$  presents the  $j$ -th block of the  $i$ -th input  $s_i$  with the mean  $m_{ij}$  and the variance  $\sigma_{ij}$ . Areas formed by blocks, named as Aa, Bb, and Cc, are called receptive fields. Each location of  $\mathbf{A}$  is parallel to a receptive field. The two-dimensional receptive field function is defined as:

$$b_k = \prod_{i=1}^2 \varphi_{ij}(s_i) \quad \text{for } k = 1, 2, \dots, n \tag{11}$$

where  $b_k$  is associated with the  $k$ -th receptive field. As for the results, the output of CMAC with the Gaussian basis function can be expressed as

$$y = \mathbf{a}_{sk}^T \begin{bmatrix} b_1 & 0 & \dots & 0 \\ 0 & b_2 & \dots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \dots & b_n \end{bmatrix} \mathbf{w} = \boldsymbol{\xi}^T \mathbf{w} = \mathbf{w}^T \boldsymbol{\xi} \tag{12}$$

$$\text{with } \boldsymbol{\xi}^T \triangleq \mathbf{a}_{sk}^T \begin{bmatrix} b_1 & 0 & \dots & 0 \\ 0 & b_2 & \dots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \dots & b_n \end{bmatrix}.$$

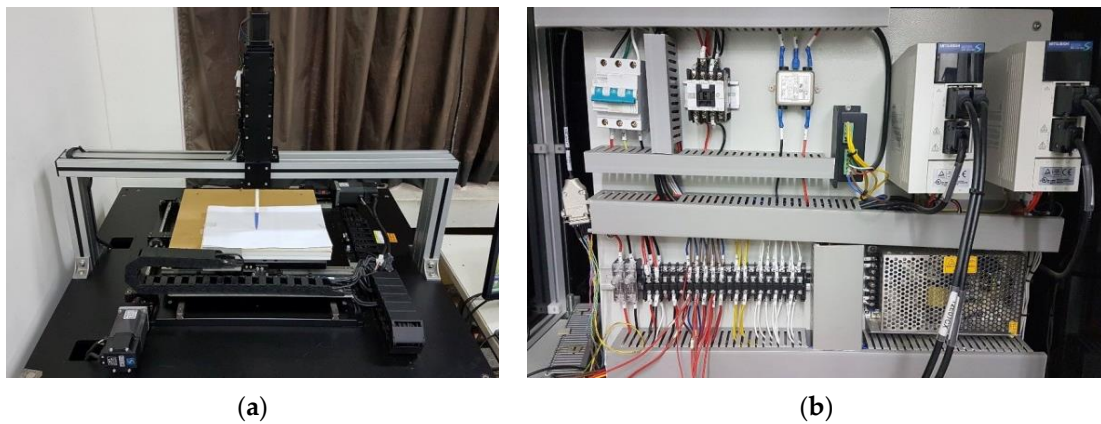


Figure 1. (a) Photographs of the two-axis motion stage, and (b) the servo drive system.

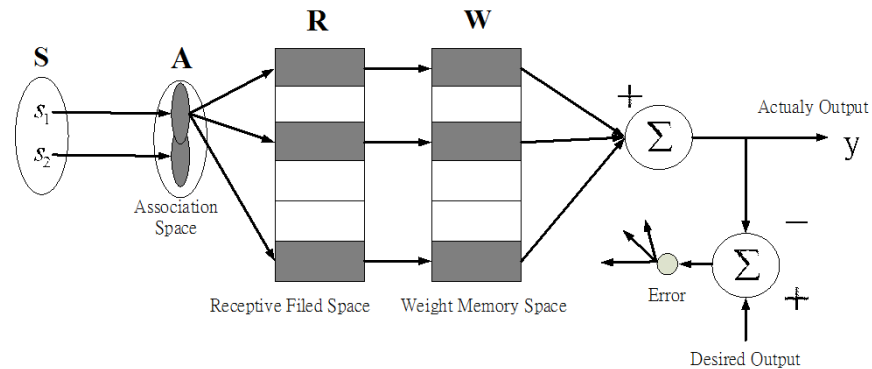


Figure 2. The basic structure of CMAC.

The mechanical system of a motion stage system with system uncertainty can be generally rewritten as a second-order nonlinear system. It is given by [4]

$$\ddot{x} = f(x) + b(x)u + d(t) \tag{13}$$

$$y = x, \tag{14}$$

where  $x$  is the stage X- or Y-axis displacement, and  $\mathbf{X} = [x_1 \ x_2]^T = [x \ \dot{x}] \in \mathbb{R}^2$  is the state vector of the system.  $f(x)$  and  $b(x)$  are the bounded real continuous nonlinear functions,  $u \in \mathbb{R}$  is the control input,  $y \in \mathbb{R}$  is the system output, and  $d$  is the bounded disturbance that cannot be measured. It is assumed that the system in Equations (13) and (14) is controllable, and it is required that  $b(x) \neq 0$  for vector  $\mathbf{X}$  in certain controllability regions. It is assumed that  $0 < b(x) < \infty$ .

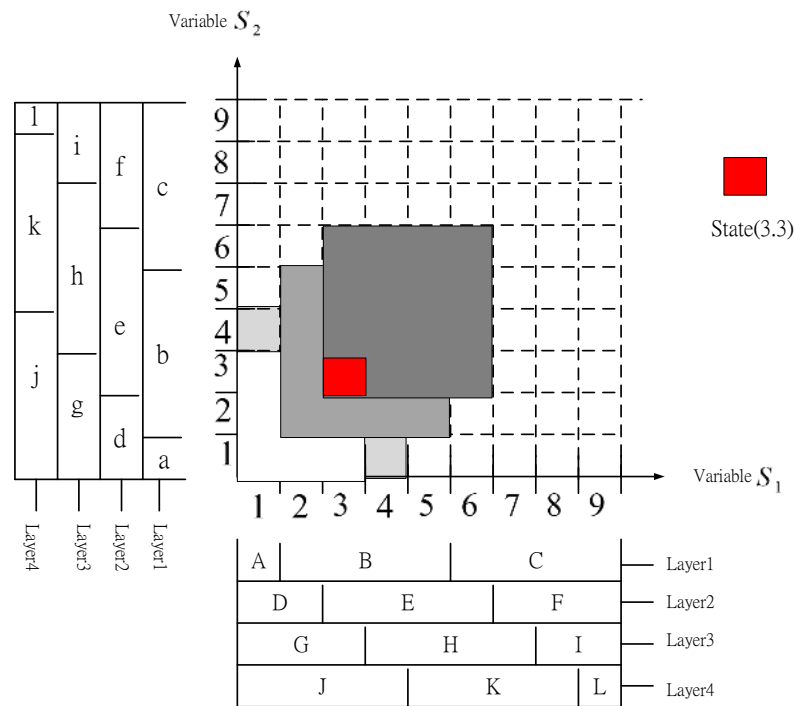


Figure 3. The mapping structure of the two-dimensional CMAC.

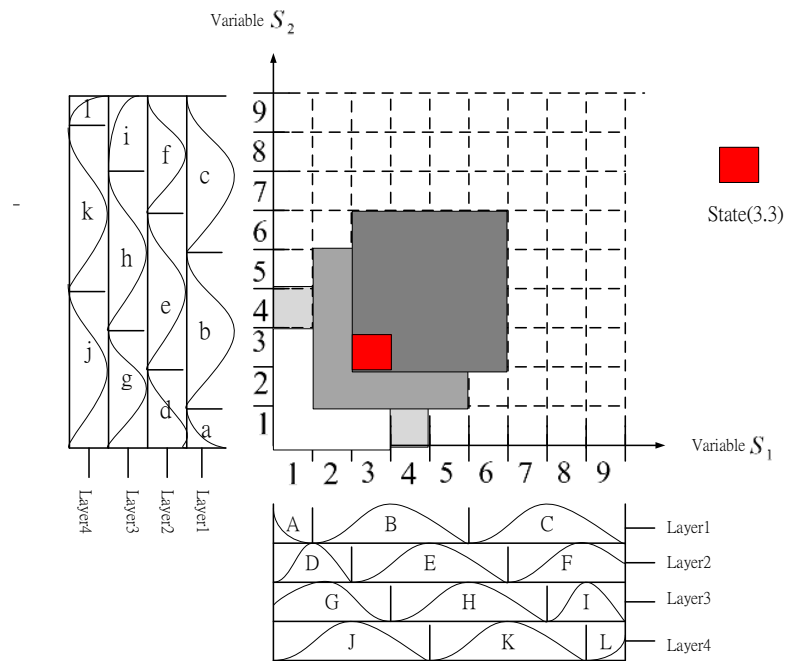


Figure 4. The mapping structure of the two-dimensional FCMAC with the Gaussian function.

Figure 5 demonstrates the architecture of the proposed DFCMACSMC method for the two-axis motion system. The output tracking error is defined as  $e = x_d - y = x_d - x$ , and the tracking error vector is  $E = [ e \quad \dot{e} ]^T = [ e_1 \quad e_2 ]^T$ .

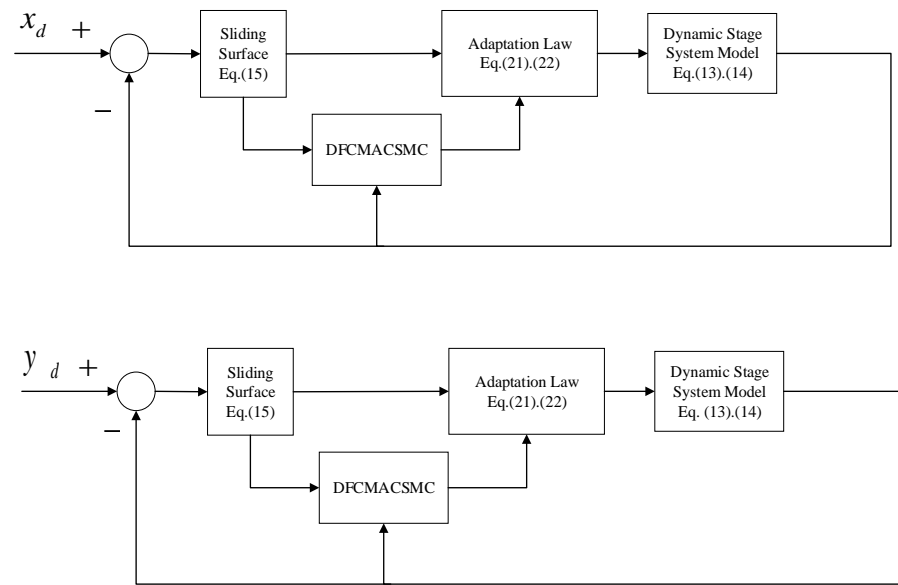


Figure 5. The overall scheme of the DFCMAC sliding mode control system.

The sliding surface parameter  $s$  is defined as follows:

$$s = -(\mathbf{cE}) = -(c_1e + \dot{e}) \tag{15}$$

with  $\mathbf{c} = [ c_1 \ 1 ]$ .

To guarantee that the trajectory of the state error vector  $\mathbf{e}$  will transfer from the approaching phase to the sliding phase, the sufficient condition is

$$s\dot{s} = -\eta|s|, \eta > 0. \tag{16}$$

Since  $f(x)$  and  $b(x)$  are known and free of external disturbances, the derivative of the sliding surface is taken with respect to time as a zero-value. It becomes

$$\dot{s} = -(c_1\dot{e} + \ddot{e}) = -(c_1\dot{e} + \ddot{x}_d - \ddot{x}) = -(c_1\dot{e} - f(\mathbf{x}) - \mathbf{b}(\mathbf{x})u_{eq}) - \ddot{x}_d = 0. \tag{17}$$

Therefore, the equivalent control can be obtained as follows:

$$u_{eq} = \frac{1}{\mathbf{b}(\mathbf{x})} (c_1\dot{e} - f(\mathbf{x}) + \ddot{x}_d) \tag{18}$$

In the approaching phase,  $s \neq 0$ , an approaching-type control  $u_{ap}$  must be added to satisfy the sufficient condition and complete sliding mode control. It can be expressed as:

$$\mathbf{u} = u_{eq} = \frac{u_{ap}}{\mathbf{b}(\mathbf{x})} \tag{19}$$

with

$$u_{ap} = \eta_{\Delta} \text{sgn}(s), \eta_{\Delta} = D + \eta, \eta_{\Delta} \geq \eta \geq 0, \text{sgn}(s) = \begin{cases} 1 & s > 0 \\ 0 & s = 0 \\ -1 & s < 0 \end{cases}$$

where  $u_{eq}$  is the equivalent control input, and  $\eta_{\Delta}$  and  $\eta$  are positive constants.

Replacing  $u_{eq}$  in Equation (18) by the FCMAC  $u(\mathbf{x}|\theta)$  gives

$$u(\mathbf{x}|\theta) = \xi^T \theta. \tag{20}$$

The control input  $u$  and the adaptive laws are chosen as:

$$u = u_D(\mathbf{x}|\boldsymbol{\theta}) - \frac{\eta \operatorname{sgn}(s)}{b(\mathbf{x})} \quad (21)$$

$$\dot{\boldsymbol{\theta}} = -2r_1 s b(\mathbf{x}) \boldsymbol{\xi}(\mathbf{x}). \quad (22)$$

The optimal parameters  $\boldsymbol{\theta}^*$  is defined as follows:

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta} \in \Omega_{\boldsymbol{\theta}}} \left[ \sup_{\mathbf{x} \in \Omega_{\mathbf{x}}} |u_D(\mathbf{x}|\boldsymbol{\theta}) - u_{eq}| \right], \quad (23)$$

where  $\Omega_{\boldsymbol{\theta}}$  and  $\Omega_{\mathbf{x}}$  are the set of suitable bounds on  $\boldsymbol{\theta}$  and  $\mathbf{x}$ , respectively.

The minimum approximation error is defined as:

$$\omega = b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta}^*) - u_{eq}) + d(t). \quad (24)$$

The time derivative of the sliding surface is

$$\begin{aligned} \dot{s} &= -(c_1 \dot{e} + \ddot{x}_d - \ddot{x}) \\ &= -(c_1 \dot{e} - f(\mathbf{x}) - b(\mathbf{x})u - d(t)) - \ddot{x}_d \\ &= -c_1 \dot{e} + f(\mathbf{x}) + b(\mathbf{x})u + d(t) - \ddot{x}_d \\ &= -c_1 \dot{e} + f(\mathbf{x}) + b(\mathbf{x})u + d(t) - \ddot{x}_d + \omega - b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta}^*) - u_{eq}) - d(t) \\ &= b(\mathbf{x}) \left( u_D(\mathbf{x}|\boldsymbol{\theta}) - \frac{\eta \operatorname{sgn}(s)}{b(\mathbf{x})} \right) + \omega - b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta}^*)) \\ &= -\eta \operatorname{sgn}(s) + \omega + b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta})) - b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta}^*)) \\ &= -\eta \operatorname{sgn}(s) + \omega + b(\mathbf{x})(u_D(\mathbf{x}|\boldsymbol{\theta}) - u_D(\mathbf{x}|\boldsymbol{\theta}^*)) = -\eta \operatorname{sgn}(s) + \omega + b(\mathbf{x}) \tilde{\boldsymbol{\theta}}^T \boldsymbol{\xi}(\mathbf{x}) \end{aligned} \quad (25)$$

The Lyapunov function is defined as:

$$V_{DFCMACSMC} = (s^2 + \frac{1}{2r_1} \tilde{\boldsymbol{\theta}}^T \tilde{\boldsymbol{\theta}}) \quad (26)$$

with  $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\theta}^*$ .

The time derivative of function  $V$  can be expressed as follows:

$$\begin{aligned} \dot{V}_{DFCMACSMC} &= s\dot{s} + \frac{1}{2r_1} \tilde{\boldsymbol{\theta}}^T \dot{\tilde{\boldsymbol{\theta}}} + \frac{1}{2r_1} \tilde{\boldsymbol{\theta}}^T \dot{\tilde{\boldsymbol{\theta}}} \\ &= s \left( -\eta \operatorname{sgn}(s) + \omega + b(\mathbf{x}) \tilde{\boldsymbol{\theta}}^T \boldsymbol{\xi}(\mathbf{x}) \right) + \frac{1}{r_1} \tilde{\boldsymbol{\theta}}^T \dot{\tilde{\boldsymbol{\theta}}} \\ &= s b(\mathbf{x}) \tilde{\boldsymbol{\theta}}^T \boldsymbol{\xi}(\mathbf{x}) + \frac{1}{r_1} \tilde{\boldsymbol{\theta}}^T \dot{\tilde{\boldsymbol{\theta}}} - s \eta \operatorname{sgn}(s) + s \omega \\ &= \frac{1}{r_1} \tilde{\boldsymbol{\theta}}^T \left( \dot{\tilde{\boldsymbol{\theta}}} + r_1 s b(\mathbf{x}) \boldsymbol{\xi}(\mathbf{x}) \right) - s \eta \operatorname{sgn}(s) + s \omega = -s \eta \operatorname{sgn}(s) + s \omega \leq 0 \end{aligned} \quad (27)$$

with  $\dot{\tilde{\boldsymbol{\theta}}} = \dot{\boldsymbol{\theta}}$ .

By choosing  $\eta_d > 0$ , Equation (31) is given as follows:

$$\dot{V}_{DFCMACSMC} \leq -|\omega||s| - \eta|s| \leq -\eta_d |s| \quad (28)$$

with  $\eta_d = (\eta + |\omega|)$ .

Integrating the above equation, it yields

$$\int_0^t |s(\tau)| d\tau \leq \frac{1}{\eta_d} [V_{DFCMACSMC}(0) - V_{DFCMACSMC}(t)] \quad (29)$$



Since  $V_{DFCMACSMC}(0)$  is bounded, and  $V_{DFCMACSMC}(t)$  is nonincreasing and bounded, the following equation can be conducted:

$$\int_0^t |s(\tau)| d\tau \leq \infty \quad (30)$$

Furthermore,  $\dot{s}(\tau)$  is also bounded. The  $s(\tau)$  is uniformly continuous. Using Barbalat's lemma [16], the following result can be obtained:

$$\lim_{t \rightarrow \infty} s(t) = 0 \quad (31)$$

It can be implied that  $\lim_{t \rightarrow \infty} |e(t)| = 0$ . The DFCMACSMC system is stable, and the error will converge to zero asymptotically [19–22]. The proposed system guarantees the asymptotical stability of the tracking error  $e(t)$ , even with the parameter variations and external disturbances.

### 3. Results

#### 3.1. Contour Planning

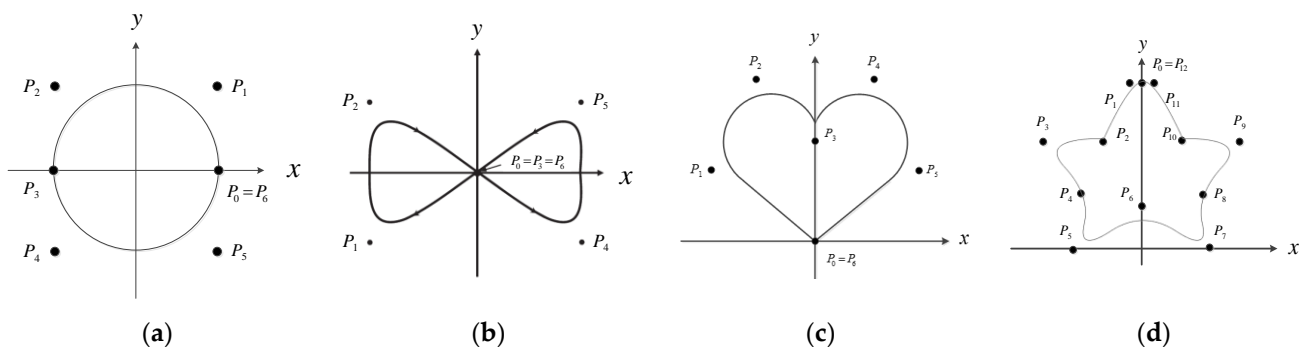
The NURBS [23] curve is applied to optimize the trajectory point to obtain the smooth trajectory. The NURBS curve expression is defined as follows:

$$F(u) = \frac{\sum_{i=0}^n w_i D_i N_{i,k}(u)}{\sum_{i=0}^n w_i N_{i,k}(u)} \quad (32)$$

$$N_{i,0} = \begin{cases} 1 & u_i \leq u \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (33)$$

$$N_{i,m} = \frac{(u - u_i)N_{i,m-1}(u)}{u_{i+m} - u_i} + \frac{(u_{i+m+1} - u)N_{i+1,m-1}(u)}{u_{i+m+1} - u_{i+1}}, \quad m \geq 1 \quad (34)$$

with  $U = [u_0, u_1, \dots, u_{n+k+1}]$ , where  $u$  is the control variable of the curve,  $k$  is the power exponent of the interpolation basis function,  $n$  is the number of interpolation points, and  $w_i$  is the weight factor, whose number is the same as the control variable.  $N_{i,k}$  is the basis function, which is selected according to the node vector. Four contour shapes were applied for this proposed adaptive DFCMAC control system, namely, the circular, bowknot, heart, and star curves. Table 1 lists the parameters of these four contours. Figure 6a–d depicts the desired contours based on the abovementioned parameter setting.



**Figure 6.** The reference contours, (a) the circle contour, (b) the bowknot contour, (c) the heart contour, and (d) the star contour (unit:  $\mu\text{m}$ ).

**Table 1.** The four types of trajectory planning.

Trajectory Type	Segment Functions
Circle contour	$k_d = 2$ $\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(2.5, 0), (2.5, 2.5), (-2.5, 2.5), (-2.5, 0), (-2.5, -2.5), (2.5, -2.5), (2.5, 0)]$ (unit: cm, cm) $\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9] = [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1]$ $\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 0.5, 0.5, 1, 0.5, 0.5, 1]$
Bowknot contour	$k_d = 2$ $\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(0, 0), (-1.5, -1.5), (-1.5, 1.5), (0, 0), (1.5, -1.5), (1.5, 1.5), (0, 0)]$ (unit: cm, cm) $\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9] = [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1]$ $\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 2.5, 2.5, 1, 2.5, 2.5, 1]$
Heart contour	$k_d = 2$ $\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(0, 0), (-3, 2), (-2, 5), (0, 3.6), (2, 5), (3, 1), (0, 0)]$ (unit: cm, cm) $\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9] = [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1]$ $\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 1, 1, 1, 1, 1, 1]$
Star contour	$k_d = 2$ $\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}, V_{12}]$ $= [(0, 6), (-0.5, 6), (-1.5, 4), (-4, 4), (-2, 2), (-2.5, 0), (0, 1.5), (2.5, 0), (2, 2), (4, 4), (1.5, 4), (0.5, 6), (0, 6)]$ (unit: cm, cm) $\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{14}] = [0, 0, 0, 0.1, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.8, 1, 1, 1]$ $\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}, w_{11}, w_{12}] = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$

### 3.2. Parameter Setting and Performance Measurement

The average tracking error (ATE,  $T_E$ ) and tracking error standard deviation (TESD,  $T_S$ ) are selected as the performance indices, which is defined as follows:

$$T_E = \sum_{k=1}^K \frac{E(k)}{n} \quad (35)$$

$$T_S = \sqrt{\frac{\sum_{k=1}^K (E(k) - T_E)^2}{K}} \quad (36)$$

with

$$E(k) = \sqrt{e_x^2(k) + e_y^2(k)},$$

where  $e_x(k)$  is the tracking error in the x-axis,  $e_y(k)$  is the tracking error in the y-axis, and  $K$  is the total number of contour points. Two types of control algorithms are considered herein: (a) the direct adaptive conventional CMAC sliding method, and (b) the proposed adaptive DFCMAC sliding method.

In our experiments, the learning parameters are selected as: (a) The conventional CMAC method:  $r_{1x} = r_{1y} = 0.001$ ,  $c_{1x} = c_{1y} = 1$ ,  $\eta = 1$ ; and (b) the proposed DFCMAC-SMC method:  $r_{1x} = r_{1y} = 0.001$ ,  $c_{1x} = c_{1y} = 1$ ,  $m = 0.5$ ,  $\sigma = 1$ ,  $\eta = 1$ . To verify the advancement of the DFCMACSMC method, four simulations are investigated.

The system parameters and Stribeck friction models of the X-Y stage are assumed as follows:

X-axis:  $K_{tx} = 0.96 \text{ N/A}$ ,  $M_x = 3 \times 10^{-3} \text{ N s}^2/\text{m}$ ,  $B_x = 0.1 \text{ N s/m}$ ,  $F_{Lx} = 0.1 \text{ N}$ ,  $F_{cx} = 0.15 \text{ N}$ ,  $F_{sx} = 0.24 \text{ N}$ ,  $V_{sx} = 10 \text{ m/s}$ .

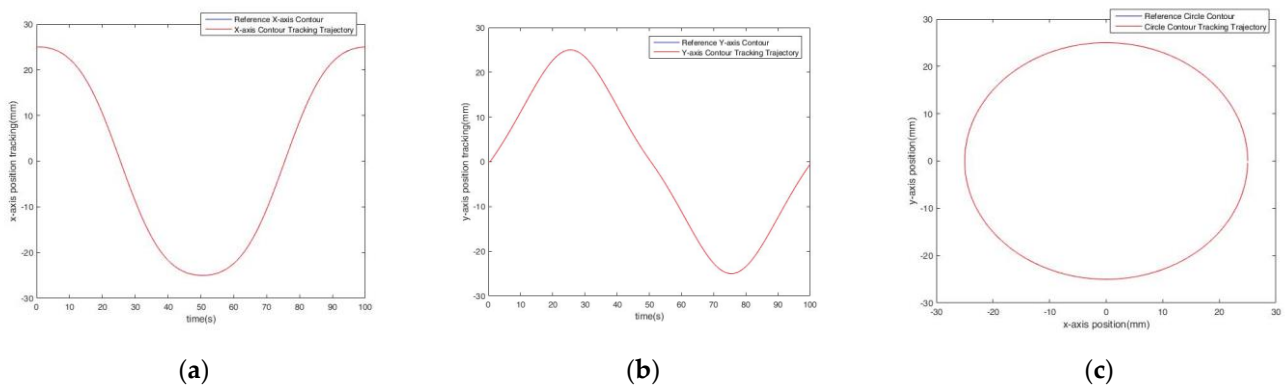
Y-axis:  $K_{ty} = 0.96 \text{ N/A}$ ,  $M_y = 2.8 \times 10^{-3} \text{ N s}^2/\text{m}$ ,  $B_y = 0.12 \text{ N s/m}$ ,  $F_{Ly} = 0.1 \text{ N}$ ,  $F_{cy} = 0.15 \text{ N}$ ,  $F_{sy} = 0.2 \text{ N}$ ,  $V_{sy} = 5 \text{ m/s}$ .

### 3.3. Simulation

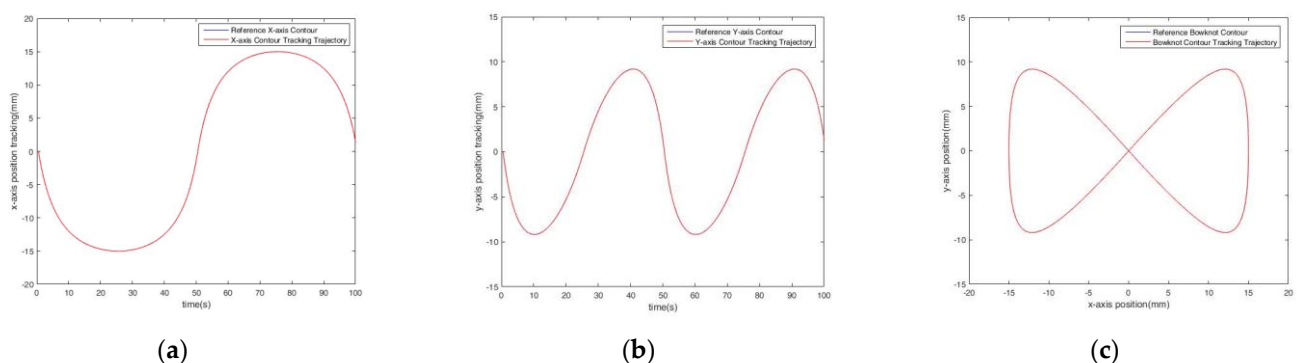
The results derived from the conventional CMAC methods and those results generated from our proposed method were compared and evaluated.

- The circle contour was evaluated to show the tracking results observed in the X and Y axes, respectively. The simulation results are illustrated in Figure 7a–c. It shows that the proposed DFCMACSMC method performed well for trajectory tracking. The average of ATE index was  $22.51 \mu\text{m}$ , and the average of TESD was  $7.92 \mu\text{m}$ .
- In the bowknot trajectory case, this method also behaves well in the X-axis and Y-axis displacement response, as shown in Figure 8a–c. The average of the ATE index was  $21.79 \mu\text{m}$ , and the average of TESD was  $7.62 \mu\text{m}$  in the bowknot case.
- The third specific application was the heart trajectory. Figure 9a–c indicates that the position errors of the X and Y axes reduced significantly during the tracking process for the two-axis system. The average of the ATE index was  $27.22 \mu\text{m}$ , and the average of TESD was  $9.52 \mu\text{m}$  in the heart case.
- The star trajectory that we designed generated the total control point, defined as 13. Figure 10a–c shows the two-dimensional trajectories. The DFCMACSMC architecture could better achieve the tracking performance, and the trajectory error decreased at a steady state. The average of the ATE index was  $21.88 \mu\text{m}$ , and the average of the TESD was  $7.66 \mu\text{m}$ .

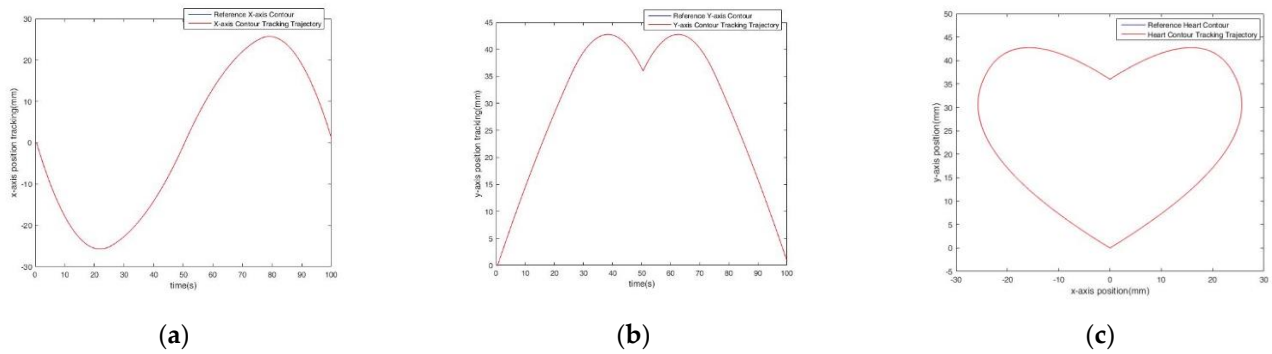
The proposed DFCMACSMC is compared to the conventional CMAC method shown in Table 2. The position errors are  $23.35 \mu\text{m}$  of the ATE index and  $8.18 \mu\text{m}$  of the associated TESD index. They indicate that the cerebellar model after adding the Gaussian function not only retains the advantages of the traditional cerebellum, but also improves the processing on uncertain factors to achieve better learning.



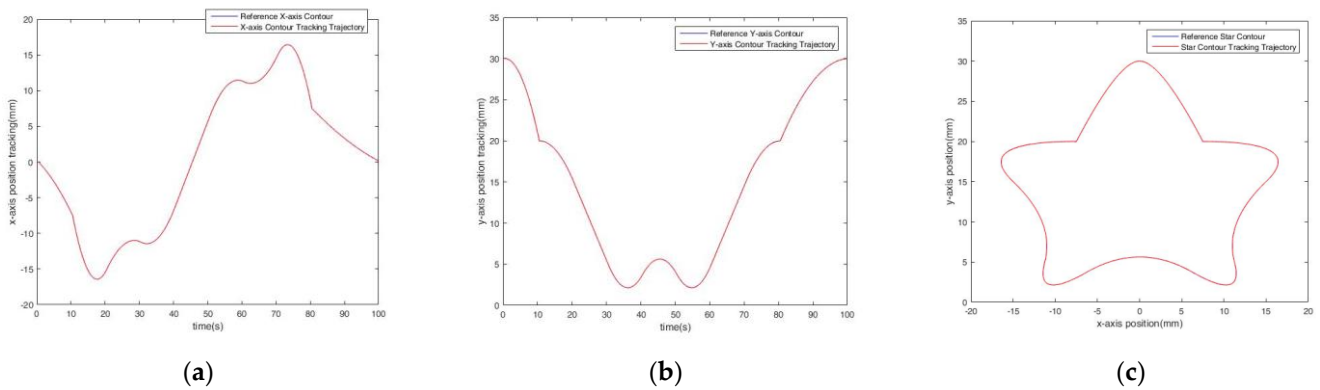
**Figure 7.** The path trajectory of the simulations. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the circle path response.



**Figure 8.** The path trajectory of the simulations. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the bowknot path response.



**Figure 9.** The path trajectory of the simulations. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the heart path response.



**Figure 10.** The path trajectory of the simulations. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the star path response.

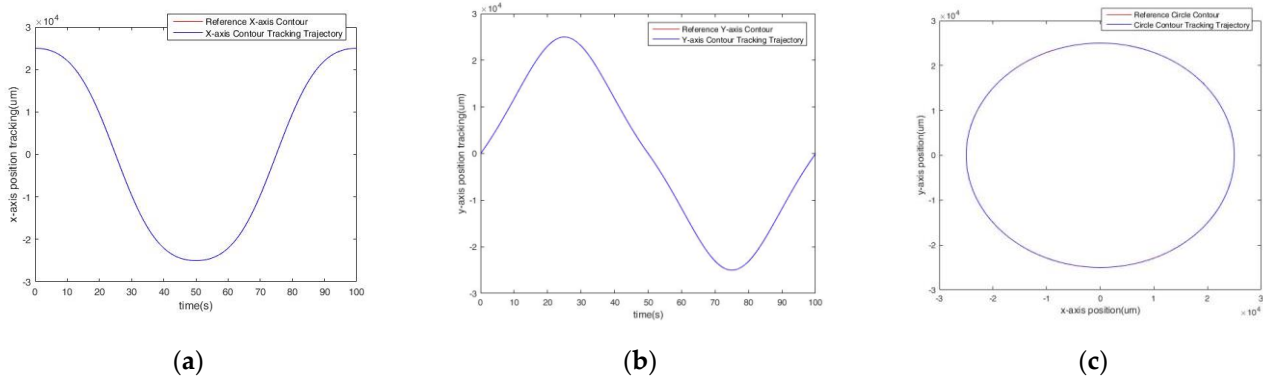
**Table 2.** The simulated results showing contour tracking errors.

Average Tracking Error (ATE) $E_M$ ( $\mu\text{m}$ )		
Trajectory Contour	The Conventional CMAC Method	The PROPOSED DFCMACSMC Method
Circle contour	50.04	22.51
Bowknot contour	32.89	21.79
Heart contour	48.67	27.22
Star contour	32.79	21.88
Average	41.09	23.35
Tracking Error Standard Deviation (TESD) $E_{STD}$ ( $\mu\text{m}$ )		
Trajectory Contour	The Conventional CMAC Method	The Proposed DFCMACSMC Method
Circle contour	17.53	7.92
Bowknot contour	11.48	7.62
Heart contour	17.04	9.52
Star contour	11.48	7.66
Average	14.38	8.18

### 3.4. Field Test Response

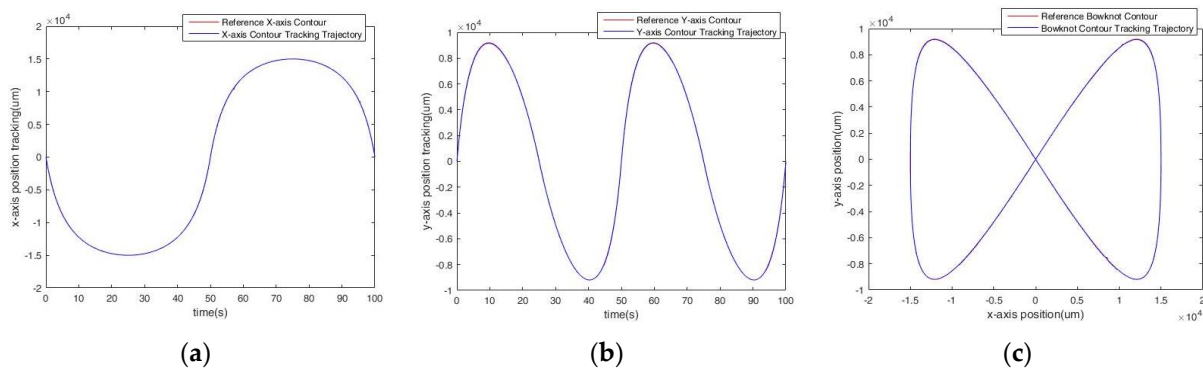
The field test was performed to verify the practicality of our proposed DFCMACSMC system. The path responses of the circle contour were measured and experimented, as shown in Figure 8. The total control points were seven. Figure 6a presents the reference

trajectory. Figure 11a,b illustrates the x- and y-axis direction responses, respectively. As shown in Figure 11c, it is obvious that the proposed DFCSMC method achieves better in dynamic tracking response. The ATE is  $41.184 \mu\text{m}$ , and the TEDS is  $12.653 \mu\text{m}$ .



**Figure 11.** The path trajectory of the experiments. (a) The X-axis tracking response, (b) the Y-axis tracking response, (c) the circle path response.

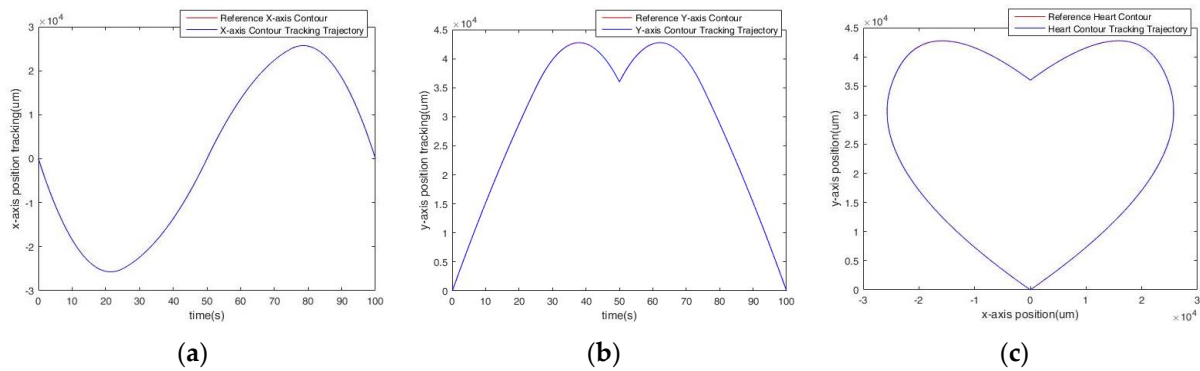
In the second test, seven control points were selected, and the corresponding trajectories are shown in Figure 6b. The responses of the bowknot outline were measured, as shown in Figure 12a,b, respectively. It is illustrated that the tracking capability could be guaranteed with small error responses. The experiment on the bowknot tracking trajectory is illustrated in Figure 12c. The tracking contours were close to the reference paths, and the steady-state response also behaved well. The position errors were  $28.888 \mu\text{m}$  for the ATE index and  $26.259 \mu\text{m}$  for the TEDS index.



**Figure 12.** The path trajectory of the experiments. (a) The X-axis tracking response, (b) the Y-axis tracking response, (c) the bowknot path response.

The heart contour is plotted in Figure 6c, and the total control point in the heart contour was seven. The experiments of the heart trajectory are illustrated in Figure 13a–c. The results illustrate that the trajectory performance was very fast and the steady-state error decreased effectively. The position errors were  $38.855 \mu\text{m}$  for the ATE index and  $17.077 \mu\text{m}$  for the TEDS index.

The star contour is plotted in Figure 6d, and the total control points in the star contour were 13. The associated star trajectories are presented in Figure 14a–c. The proposed DFCSMC method can handle the model uncertainty, and it effectively reduces the error and achieves better trajectory tracking. The position errors reached  $29.269 \mu\text{m}$  for the ATE index and  $18.349 \mu\text{m}$  for the associated TEDS index for star contour tracking.

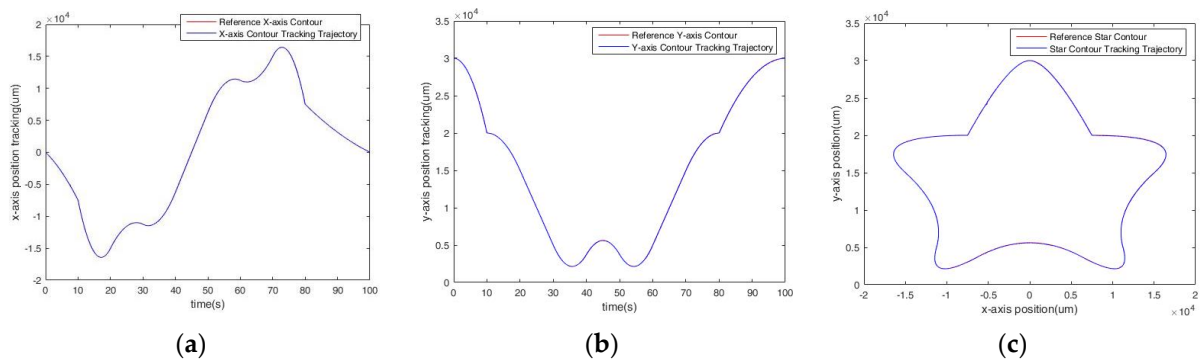


**Figure 13.** The path trajectory of the experiments. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the heart path response.

Table 3 shows the comparison of the tracking errors with the DFCMACSMC and conventional CMAC method. Four reference contours are considered. The proposed DFCMACSMC method demonstrated more accurate performances, showing a 42.93% improvement in the ATE and a 40.34% improvement in the TESD, compared with the conventional CMAC. It is shown that the DFCMACSMC method obtained the lowest indexes of ATE and TESD.

**Table 3.** The experimental results showing contour tracking errors.

Average Tracking Error (ATE) $E_M$ ( $\mu\text{m}$ )		
Trajectory Contour	The Conventional CMAC Method	The Proposed DFCMACSMC Method
Circle contour	64.520	41.184
Bowknot contour	52.836	28.888
Heart contour	69.029	38.855
Star contour	55.905	29.269
Average	60.573	34.569
$E_M$ Tracking Error Standard Deviation (TESD) $E_{STD}$ ( $\mu\text{m}$ )		
Trajectory Contour	The Conventional CMAC Method	The Proposed DFCMACSMC Method
Circle contour	18.696	12.653
Bowknot contour	39.215	26.259
Heart contour	31.581	17.077
Star contour	35.126	18.349
Average	31.155	18.585



**Figure 14.** The path trajectory of the experiments. (a) The X–axis tracking response, (b) the Y–axis tracking response, (c) the star path response.

#### 4. Conclusions

The DFCMACSMC method was presented to control the PMSM servo drive X-Y motion system. This adaptive structure could guarantee and achieve the robustness, and so this could provide superior tracking capability in the presence of parameter uncertainties, cross-coupled interference, and external disturbances. Four contour trajectories were experimented with to evaluate the proposed system. It is shown that the adaptive DFCMACSMC system could alleviate the chattering and tracking errors in the field tests. On average, it could achieve 42.9% and 40.3% improvement of ATE and TESD, respectively, compared with the traditional CMAC strategy. Our research mainly contributes to a better understanding of how successfully the DFCMACSMC method can be established to compensate the XY motion stage with four different contours. In the future, the ARM-based or DSP-based motion controllers will be designed and implemented such that our proposed DFCMACSMC architecture can be widely realized and utilized in industrial applications.

**Author Contributions:** All authors contributed to the paper. W.-L.M. and Y.-Y.C. designed the control platform and method; W.-L.M. and W.-C.S. developed the simulation and experimental software, B.-H.L. and W.-C.S. implemented and validated the algorithm; W.-L.M. and Y.-Y.C., J.-F.T. wrote and planned the original draft preparation, Y.-Y.C., J.-F.T. were project administration. All authors have read and agreed to the published version of the manuscript.

**Funding:** The authors would like to extend special thanks to the Ministry of Science and Technology of the Republic of China, Taiwan, for the financial support of this research under Contract No. MOST 109-2221-E-224-024 -, MOST 108-2731-M-224-001 -, and MOST 107-2221-E-224-040-.

**Conflicts of Interest:** The authors declare no conflict of interest.

#### References

1. Fujimoto, H.; Takemura, 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. [\[CrossRef\]](#)
2. Kung, Y.S.; Than, H.; Chung, T.Y. FPGA-realization of a self-tuning PID Controller for X-Y table with RBF neural network identification. *Microsyst. Technol.* **2016**, *24*, 243–253. [\[CrossRef\]](#)
3. Wu, J.; Xiong, Z.; Ding, H. Integral design of contour error model and control for biaxial system. *Int. J. Mach. Tools Manuf.* **2015**, *89*, 159–169. [\[CrossRef\]](#)
4. Lin, F.J.; Shieh, H.J.; Shieh, P.H.; Shen, P.H. An adaptive Recurrent-Neural-Network Motion Controller for X-Y Table in CNC Machine. *IEEE Trans. Syst. Man Cybern.* **2006**, *36*, 286–299.
5. Lin, F.J.; Chou, P.H.; Kung, Y.S. Robust fuzzy neural network controller with nonlinear disturbance observer for two-axis motion control system. *IET Control. Theory Appl.* **2008**, *2*, 151–167. [\[CrossRef\]](#)
6. El-Sousy, F.F.M. Intelligent mixed H<sub>2</sub>/H<sub>∞</sub> adaptive tracking control system design using self-organizing recurrent fuzzy-wavelet-neural-network for uncertain two-axis motion control system. *Appl. Soft Comput.* **2016**, *41*, 22–50. [\[CrossRef\]](#)
7. Xu, Q.; Li, Y. Dynamic Modeling and Sliding Mode Control of an XY Micropositioning Stage. In Proceedings of the 9th International Symposium on Robot Control (SYROCO'09), Gifu, Japan, 9–12 September 2009; pp. 781–786.
8. Lin, F.J.; Shieh, P.H.; Shen, P.H. Robust recurrent-neural-network sliding-mode control for the X-Y table of a CNC machine. *IET Control. Theory Appl.* **2006**, *153*, 111–123. [\[CrossRef\]](#)
9. Lin, C.M.; Peng, Y.F.; Hsu, C.F. Robust cerebellar model articulation controller design for unknown nonlinear systems. *IEEE Trans. Circuits Syst. II Express Briefs* **2004**, *51*, 354–358. [\[CrossRef\]](#)
10. Lin, C.S.; Kim, H. CMAC-Based Adaptive Critic Self-Learning Control. *IEEE Trans. Neural Netw.* **1991**, *2*, 530–535. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Lin, C.S.; Kim, H. Selection of Learning Parameters for CMAC-Based Adaptive Critic Learning. *IEEE Trans. Neural Netw.* **1996**, *6*, 642–647. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Wu, T.F.; Tsai, P.S.; Chang, F.R.; Wang, L.S. Adaptive fuzzy CMAC control for a class of nonlinear systems with smooth compensation. *IEE J. Proc. Control. Theory* **2006**, *153*, 647–657. [\[CrossRef\]](#)
13. Chen, J.Y.; Tsai, P.S.; Wong, C.C. Adaptive design of a fuzzy cerebellar model arithmetic controller neural network. *IEE J. Proc. Control. Theory Appl.* **2005**, *152*, 133–137. [\[CrossRef\]](#)
14. Lin, T.C. Based on interval type-2 fuzzy-neural network direct adaptive sliding mode control for SISO nonlinear systems. *Commun. Nonlinear Sci. Numer. Simul.* **2010**, *15*, 4084–4099. [\[CrossRef\]](#)
15. Tsao, Y.; Chu, H.C.; Fang, S.H.; Lee, J.; Lin, C.M. Adaptive noise cancellation using deep cerebellar model articulation controller. *IEEE Access* **2018**, *6*, 37395–37402. [\[CrossRef\]](#)
16. Piao, Z.; Guo, C.; Sun, S. Adaptive backstepping sliding mode dynamic positioning system for pod driven unmanned surface vessel based on cerebellar model articulation controller. *IEEE Access* **2020**, *8*, 48314–48324. [\[CrossRef\]](#)

17. Huynh, T.T.; Lin, C.M.; Le, T.L.; Cho, H.Y.; Pham, T.T.T.; Le, N.Q.K.; Chao, F. A new self-organizing fuzzy cerebellar model articulation controller for uncertain nonlinear systems using overlapped gaussian membership functions. *IEEE Trans. Ind. Electron.* **2020**, *67*, 9671–9682. [[CrossRef](#)]
18. Chao, F.; Zhou, D.; Lin, C.M.; Yang, L.; Zhou, C.; Shang, C. Type-2 fuzzy hybrid controller network for robotic systems. *IEEE Trans. Cybern.* **2020**, *50*, 3778–3792. [[CrossRef](#)] [[PubMed](#)]
19. Slotine, E.J.J.; Li, P.W. *Applied Nonlinear Control*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1991.
20. Wang, L.X. *Adaptive Fuzzy Systems and Control: Design and Stability*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1994.
21. Wang, J.; Rad, A.B.; Chan, P.T. Indirect adaptive fuzzy sliding mode control: Part I: Fuzzy switching. *Fuzzy Sets Syst.* **2001**, *12*, 21–30. [[CrossRef](#)]
22. Chan, T.P.; Rad, B.A.; Wang, J. Indirect adaptive fuzzy sliding mode control: Part II: Parameter projection and supervisory control. *Fuzzy Sets Syst.* **2001**, *12*, 31–43. [[CrossRef](#)]
23. Piegl, L.; Tiller, W. *The NURBS Books*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1997.