REGULAR PAPER

Precision Trajectory Tracking on XY Motion Stage Using Robust Interval Type-2 Fuzzy PI Sliding Mode Control Method

Wei-Lung Mao[1] · Ding-Yu Shiu[1]

Abstract

Precision contour tracking is one of the most important factors used to determine product quality in a machining tool. An interval type-2 fuzzy proportional–integral (PI) sliding mode control (IT2FPISMC) system is proposed herein to control the mover position of the two-axis motion stage with optical encoder sensors for trajectory feedback. A type-2 fuzzy method that can handle rule uncertainties is developed to approach the unknown nonlinear systems. The PI term is used to approximate the discontinuous control signal and mitigate the chattering phenomenon in the presence of unmodeled system dynamics and external disturbances. The adaptive control laws are derived based on the Lyapunov theorem, such that the closed-loop stability is guaranteed, and the output tracking errors of the system asymptotically converge to zero. A non-uniform rational B-spline interpolator with high accuracy is adopted in the biaxial linear stage. Moreover, typical circular, bowknot, heart, and star reference contours are tested. The results on the average tracking error and the tracking error standard deviation are experimented and compared to illustrate the performance of our proposed method. The tracking performance obtained from the IT2FPISMC method is better than that of the conventional method. Furthermore, the proposed method can achieve robustness for tracking different reference contours in industrial applications.

Keywords Adaptive fuzzy proportional–integral sliding mode control (FPISMC) · Interval type-2 fuzzy logic system (IT2FLS) · Permanent magnet synchronous motor (PMSM) · Optical encoder sensor · Precision motion control · Trajectory control

1 Introduction

The high-accuracy performance of the XY stage in a machine tool, which is the main requirement to achieve high-quality products, has attracted increasing attention for wide applications, including in semiconductors, and in scientific, electromechanical, and biomedical equipment manufacturing [1–5]. The ball screw system, which is popularly used in the two-axis motion stage, applies a ball knot on the screw, and is much efficient at converting rotary motion to linear motion. A precision ground ball screw can improve

accuracy, but is costlier than a lead screw and, will still wear over time, resulting in reduced accuracy and repeatability. Therefore, the effects of nonlinear friction, coupled interference, and unmodeled dynamics always exist in motion control, and can significantly affect the tracking performance of the controlled system. Reducing the effect of nonlinear uncertainty and disturbance is always a challenging problem in controller design. Accordingly, many advanced control schemes were developed to compensate for these difficulties and obtain a precise positioning.

In recent years, several methods were applied for motion control to obtain high-speed and high-accuracy machining quality. These methods include the cross-coupled [2], neural network [3], fuzzy [4], fuzzy wavelet neural network [5], and sliding mode control methods [1, 6, 7]. Wu et al. [2] proposed an integral design control method for a biaxial system using cross-coupled control (CCC). Their method used the contour algebraic equation and its partial derivatives to represent the contour error model. The contour control effort distribution was decided at a desired

✉ Wei-Lung Mao
wlmao@yuntech.edu.tw; wlmao123@yahoo.com.tw

Ding-Yu Shiu
M10512055@yuntech.edu.tw

1 Graduate School of Engineering Science and Technology, Department of Electrical Engineering, National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C.

point. The CCC stability condition was derived based on the contour error transfer function that could improve the contour following accuracy. A recurrent neural network controller [3] was developed for the two-axis computer numerical machine driven by two permanent magnet synchronous motors (PMSMs). The adaptive learning method was derived, and the system tracked various contours with robust control performance. A self-organizing recurrent fuzzy wavelet neural network controller [5] incorporating a mixed H_2/H_∞ approach was applied in the XY table system. The control algorithm was developed in dSPACE DS1104 board to control an XY table actuated by two permanent magnet linear synchronous motors. This control architecture had good performance, and was more suitable for controlling an XY table when uncertainties occurred. Thus, the effects of the nonlinear friction and disturbance uncertainty occurring in the position system are the main obstacles in achieving a high-accuracy performance. In addition, using the interpolator algorithm [8–11] is important in generating the parametric curves [8] and obtaining a better trajectory planning in motion control. Yau [10] developed the non-uniform rational B-spline (NURBS) parameters and interpolation algorithm using the FPGA architecture in a motion controller. The Cox–de Boor algorithm was implemented, and the experimental results using the XY stage verified the good computation performance and tracking capability. The integration of the NURBS interpolator and controller design is a very critical issue to be utilized for contour planning in the motion control design.

Intelligent fuzzy algorithms [12–19] combined with sliding mode control (SMC)-based methods [1, 17–23] achieve an excellent performance by designing suitable adaptation rules rather than requiring expert experience. Two types of fuzzy system were first introduced by Zadeh [8], namely type-1 fuzzy logic system (T1FLS) and type-2 fuzzy logic system (T2FLS). T2FLS [12–19] was constructed using three-dimensional membership functions and footprint of uncertainty (FOU). The main advantage of T2FLS is that it can handle a system that involves uncertainties and nonlinear dynamical characteristics [16]. Wu [17] and Navarro [18] made a comparison between a T1PI fuzzy logic controller (FLC) and T2PI FLC, showing that the T2 PI FLC can cope with uncertainty and achieve a higher accuracy.

SMC [1, 20–23] had been widely used to provide eminent features, such as ease of implementation, and is the most promising robust control method that asymptotically converges to the sliding surface. However, the chattering problem is the main shortcoming of the basic SMC. The adaptive fuzzy sliding mode controller [21] was proposed to alleviate the chattering in the nonlinear systems. The adaptive proportional–integral (PI) controller was used to obtain the smooth control input. Type-2 fuzzy neural network SMC methods [22, 23] were presented for SISO nonlinear systems. They

can have a smoother response and a better performance compared with T1 FLC structure when slow and fast disturbances are experienced. However, only numerical examples [20–23] were realized and provided to validate the proposed methods and results. The genetic algorithm [24] was developed to optimize the membership functions of type-2 fuzzy controller, including standard deviation, mean and amplitude of lower MF. The integral squared error (ISE) is utilized to obtain the optimized MFs. A nonlinear quasi-PID scheme [25] was presented for transportation control of double-pendulum cranes. The proposed controller could work well in the presence of unodeled dynamics, actuating constraints, and large swing angles. A neural network-based adaptive control method [26] is developed to provide effective control of both actuated and unactuated state variables base on the original nonlinear ship-mounted crane dynamics without any linearizing operations. The experimental results are conducted to verify the practicability and robustness of the proposed approach.

This research proposes an interval type-2 fuzzy PI sliding mode controller (IT2FPISMC) for precision trajectory tracking using optical encoder sensors in the XY stage. The scheme aims to apply the type-2 fuzzy method to approach the unknown nonlinear functions and use the PI control to alleviate the chattering phenomenon. The PI control scheme is presented to effectively attenuate the chattering and approximate the discontinuous control signal. The projection algorithm is also employed for the adaptive law to ensure the boundaries of the parameters. The Lyapunov and Barbalat lemmas are applied to guarantee the convergence of the controlled system. With the use of the NURBS interpolator algorithm, the biaxial motion stage is experimented using four standard contours to enhance the robustness of the controlled system. The four standard contours are (1) circle, (2) bowknot, (3) heart, and (4) star. The experimental results are conducted and compared to show the trajectory tracking capabilities.

This paper is divided into five sections. Section 2 elaborates the architecture of the controlled system and the dynamic model of the XY stage. Section 3 describes the proposed IT2FPISMC method. Section 4 exhibits the experimental results. Section 5 summarizes the conclusions of this study.

2 Control System Modeling

Figure 1a shows the experimental XY-type platform for our motion control application. This research implemented the adaptive interval type-2 fuzzy-based control system using a Pentium computer. We install the digital/analog converter card and an encoder card in the computer. The controlled plant is actuated by two servomotor drives, and the rated output power, torque, speed, and current of the PMSM motor are 400 W, 1.3 Nm, 3000 rpm, and 2.3 A, respectively. The

(a)
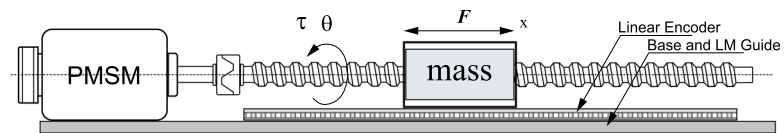
(b)

XY motion stage has a 2 mm screw pitch, 260 mm of *x*-axis travel, and 350 mm of *y*-axis travel. Mercury II 5000 series optical encoder sensors were installed to provide the high precision requirement in the position feedback loop. The system was implemented using the AC servo drive Mels-ervo-J2S-Super series manufactured by Mitsubishi. The drive system can provide three operation modes: position control, speed control, and torque control. Given that it considered system parameter uncertainty and external disturbances, including friction force, cross-coupling, and load effect, the XY stage equipment was a second-order nonlinear system in practical applications. The single-axis stage system is illustrated in Fig. 1b. The associated mathematical model can be represented as follows:

$$\ddot{x} = f(\mathbf{x}) + g(\mathbf{x})u(t) + d(t) \tag{1}$$

$$y = x \tag{2}$$

where x is the stage displacement of the *x*- or *y*-axes; $\mathbf{x}$ is the state vector; f($\mathbf{x}$) is the nonlinear function of state variables; g($\mathbf{x}$) is the control gain; u(t) is the input control voltage; d(t) is the lumped system uncertainty and coupled interference, which is bounded; $|d(t)| < D^U$; and D^U is assumed to be a given positive constant. The system in Eq. (1) is assumed to be controllable, and g($\mathbf{x}$) $\neq$ 0 is required for vector $\mathbf{x}$ in certain controllability regions. Without loss of generality, $0 < g(\mathbf{x}) < \infty$ is assumed. The control objective is to design a control law u(t), such that state x can track a desired reference trajectory x_d in the presence of the model uncertainty and external disturbance. The state vector of the system $\mathbf{x} \in R^2$ can be written as follows:

$$\mathbf{x} = [x, \dot{x}]^T = \left[x_1, x_2\right]^T \tag{3}$$

The ideal position signal is assumed to be x_d, and the tracking error is defined as e = x − x_d. The error vector is $e = [e, \dot{e}]^T$, and the sliding surface can be defined as:

$$s(\mathbf{e}) = k_1 e + \dot{e} = \mathbf{k}^T \mathbf{e} \tag{4}$$

with $\mathbf{k} = \left[k_1, 1\right]^T$, where the vector $\mathbf{k}$ is the coefficient vector of the Hurwitz polynomial h(λ) = λ + k_1. The control strategy is selected to obtain the error vector on the sliding surface s($\mathbf{e}$) = 0 for all t ≥ 0 and achieve this condition. The continuous differentiable Lyapunov function V is defined as follows:

$$V = \frac{1}{2}s^2(\mathbf{e}) \tag{5}$$

The differential operator is used, and can be expressed as follows:

$$
\begin{aligned}
\dot{V} = s \cdot \dot{s} &= s \cdot \left(k_1 \dot{e} + \ddot{e}\right) = s \cdot \left(k_1 \dot{e} - \ddot{x}_d + \ddot{x}\right) \\
&= s \cdot \left(k_1 \dot{e} - \ddot{x}_d + f(\mathbf{x}) + g(\mathbf{x})u + d\right) \\
&\leq -\eta|s|
\end{aligned} \tag{6}
$$

where η is a positive constant. If $f(\mathbf{x})$ and $g(\mathbf{x})$ are known in advance, then the ideal sliding control law u is defined as follows:

$$u = u_{eq} - u_{sw} = \frac{1}{g(\mathbf{x})}\left[-f(\mathbf{x}) - k_1 e + \ddot{x}_d - \eta_\Delta sgn(s)\right]$$
$$= \frac{1}{g(\mathbf{x})}\left[-f(\mathbf{x}) - k_1 e + \ddot{x}_d - u_p\right] \tag{7}$$

$$\text{w i t h} \quad u_{eq} = \frac{1}{g(\mathbf{x})}\left[-f(\mathbf{x}) - k_1 e + \ddot{x}_d\right], \quad u_{sw} = \frac{1}{g(\mathbf{x})}u_p,$$

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

is the equivalent control input, and u_{sw} is the switching control input. Functions $f(\mathbf{x})$ and $g(\mathbf{x})$ are usually unknown in

the actual system. Therefore, applying the control law in Eq. (7) to the XY motion system is difficult. Thus, the switching control u_{sw} can lead to the chattering phenomenon, which is undesired in conventional SMC control. Accordingly, the interval type-2 FLS approximation controller and the adaptive PI controller are constructed to solve these difficulties by introducing the fuzzy switching controller.
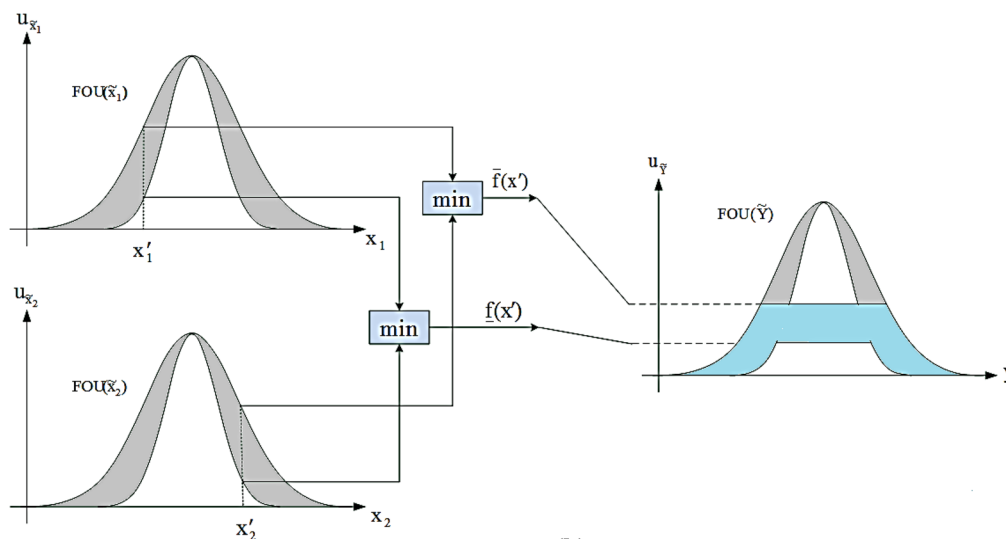
3 Proposed Adaptive IT2FPISMC Method

3.1 Interval Type-2 Fuzzy Logic System

Figure 2a illustrates a block diagram of the type-2 fuzzy system, including a fuzzifier, a rule base, an inference engine,
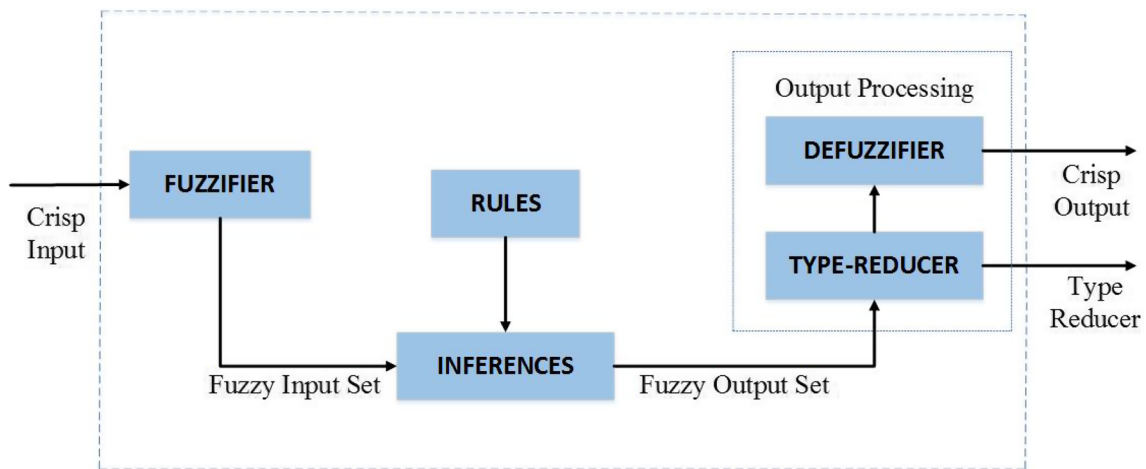
(a)

(b)

Fig. 2 Structure of fuzzy logic system **a** type 2 fuzzy system structure, **b** the singleton fuzzification with a minimum t-norm operation

a type reducer, and a defuzzifier. These functions can be designed and described as follows:

(a) *Fuzzification* The IT2FLS inference method was utilized to approximate $f(\mathbf{x})$ and $g(\mathbf{x})$. Both state variable sets (x_1, x_2) from Eq. (3) were used as the input parameters in the fuzzy system for trajectory control. These sets were used to transform the input vector $\mathbf{x} = [x_1, x_2]^T \in X_1 \times X_2 \equiv \mathbf{X}$ into the interval type-2 fuzzy set $\tilde{X}_i^n$ defined in $\mathbf{X}$. The number of inputs is two. The Gaussian primary membership functions are expressed herein as follows:

$$\bar{u}_{\tilde{X}_i^n}(x_i) = exp\left[-\left(\frac{x_i - m}{\beta}\right)^2\right] \quad i = 1, 2 \tag{8}$$

$$\underline{u}_{\tilde{X}_i^n}(x_i) = a \cdot exp\left[-\left(\frac{x_i - m}{\beta}\right)^2\right] \quad i = 1, 2 \tag{9}$$

where $\bar{u}(\bullet)$ and $\underline{u}(\bullet)$ denote the grade of the upper and lower membership functions, respectively. β is the standard deviation of the lower and upper membership functions; m is the mean value; and a is the FOU width coefficient defined between the upper and lower membership functions.

(b) *Rule base* The fuzzy rules represent the relationship between the input as the antecedent and the output as the consequent type-2 fuzzy set. N rules are assumed to exist in the T2FLS. The *n*th type-2 fuzzy rule can be expressed as follows:

Rule *n*: IF x_1 is $\tilde{X}_1^n$ AND x_2 is X_2^n,

THEN y is $\tilde{Y}^n$ $\quad n = 1, 2, \ldots N$ (10)

where $\tilde{X}_1^n$ and $\tilde{X}_2^n$ are the type-2 fuzzy sets of the IF part, and $\tilde{Y}^n$ is the consequent T2 fuzzy set replaced by $\tilde{Y}^n = [\bar{y}^n, \underline{y}^n]$. $\bar{y}^n$ and $\underline{y}^n$ are the upper and lower values of the type-1 fuzzy system, respectively.

(c) *Inference engine* The type-2 fuzzy inference engine can combine rules and establish a mapping from the input type-2 fuzzy sets to the output type-2 fuzzy sets. The firing strength of the *n*th rule can be represented as follows:

$$F^n(\mathbf{x}) = \left[\bar{f}^n(\mathbf{x}) \ \underline{f}^n(\mathbf{x})\right] = \left[\bar{f}^n \ \underline{f}^n\right] \quad n = 1, 2, \ldots N \tag{11}$$

$$\bar{f}^n(\mathbf{x}) = \bar{u}_{\tilde{X}_1^n}(x_1) \times \bar{u}_{\tilde{X}_2^n}(x_2) \tag{12}$$

$$\underline{f}^n(\mathbf{x}) = \underline{u}_{\tilde{X}_1^n}(x_1) \times \underline{u}_{\tilde{X}_2^n}(x_2) \tag{13}$$

where $\bar{f}^n(\mathbf{x})$ and $\underline{f}^n(\mathbf{x})$ are the *n*th upper and lower membership functions, respectively. The fuzzification with a minimum *t*-norm was applied here (Fig. 2b).

(d) *Output processing* Various type of reductions were proposed in the literature [12–14], including centroid, center of set, height, and modified height. The center of sets (COS)-type reduction was used herein to compute the fuzzy output. The output can be expressed as follows:

$$Y_{cos}(\mathbf{x}) = \int_{y^1} \cdots \int_{y^N} \int_{f^1} \cdots \int_{f^N} 1 \Big/ \frac{\sum_{n=1}^N f^n y^n}{\sum_{n=1}^N f^n} = [y_l, y_r] \tag{14}$$

where y_r is the right endpoint, and y_l is the left endpoint, which is derived from the consequent centroid set $\left[y_l^n \ y_r^n\right]$ and the firing strength $f^n \in F^n(\mathbf{x}) = \left[\bar{f}^n \ \underline{f}^n\right]$, respectively. y_r and y_l are given as follows using the singleton fuzzier, product inference engine, and COS type reducer:

$$y_r = \frac{\sum_{n=1}^N f_r^n y_r^n}{\sum_{n=1}^N f_r^n} = \sum_{n=1}^N y_r^n \xi_r^n = \mathbf{y}_r^T \boldsymbol{\xi}_r \tag{15}$$

$$y_l = \frac{\sum_{n=1}^N f_l^n y_l^n}{\sum_{n=1}^N f_l^n} = \sum_{n=1}^N y_l^n \xi_l^n = \mathbf{y}_l^T \boldsymbol{\xi}_l \tag{16}$$

with $\xi_r^n = f_r^n / \left(\sum_{i=1}^N f_r^n\right)$, $\xi_l^n = \underline{f}_l^n / \left(\sum_{n=1}^N f_l^n\right)$, where $\mathbf{y}_r = [y_r^1, \ldots, y_r^N]^T$ and $\mathbf{y}_l = [y_l^1, \ldots, y_l^N]^T$ are the right and left endpoint vectors, respectively; and $\boldsymbol{\xi}_r$ and $\boldsymbol{\xi}_l$ are the regressive vectors. We briefly describe the computation procedures for the rightmost point y_r. Without losing generality, we assume that y_r^n is arranged in an ascending order (i.e., $y_r^1 \leq y_r^2 \leq \cdots \leq y_r^N$).

Step 1. Calculate y_r in Eq. (15) by initially setting $f_r^n = (\bar{f}^n + \underline{f}^n)/2$ for $n = 1, 2, \ldots, N$. Here $\bar{f}^n$ and $\underline{f}^n$ are precomputed using Eqs. (12) and (13), respectively. Let $y_r' = y_r$.
Step 2. Find R $(1 \leq R \leq N - 1)$, such that $y_r^R \leq y_r' \leq y_r^{R+1}$.
Step 3. Compute y_r in Eq. (15) with $f_r^n = \underline{f}^n$ for $n \leq R$ and $f_r^n = \bar{f}^n$ for $n > R$, then let $y_r'' = y_r$.
Step 4. If $y_r' \neq y_r''$, then go to step 5. If $y_r' \neq y_r''$, then set $y_r = y_r''$ and go to step 6.
Step 5. Set $y_r' = y_r''$, and return to step 2.
Step 6. End.

The number R can be obtained from two separate sides from the abovementioned algorithm, such that one side is the lower firing strengths $\underline{f}^n$'s, and another

side is the upper firing strengths $\bar{f}^n$'s. y_r can then be rewritten as follows:

$$
\begin{aligned}
y_r &= \frac{\sum_{n=1}^{R} \underline{f}^n y_r^n + \sum_{R+1}^{N} \bar{f}^n y_r^n}{\sum_{n=1}^{R} \underline{f}^n + \sum_{n=R+1}^{N} \bar{f}^n} = \sum_{n=1}^{R} \underline{q}_r^n y_r^n \\
&+ \sum_{n=R+1}^{N} \bar{q}_r^n y_r^n = \left[\underline{\mathbf{y}}_r^T \ \bar{\mathbf{y}}_r^T \right] \left[\frac{\mathbf{Q}}{\bar{\mathbf{Q}}_r} \right] = \boldsymbol{\theta}_r^T \boldsymbol{\xi}_r
\end{aligned}
\tag{17}
$$

with $q^n = \underline{f}^n/D_r$, $\bar{q}_r^n = \bar{f}^n/D_r$, $D_r = \sum_{n=1}^{R} \underline{f}^n + \sum_{n=R+1}^{N} \bar{f}^n$, $\boldsymbol{\theta}_r = \left[\frac{\underline{\mathbf{y}}_r}{\bar{\mathbf{y}}_r} \right]$, $\boldsymbol{\xi}_r = \left[\frac{\mathbf{Q}_r}{\bar{\mathbf{Q}}_r} \right]$.

The process to compute y_l is similar to that used to compute y_r. In step 2, we must determine L ($1 \le L \le M - 1$), such that $y_l^L \le y_l' \le y_l^{L+1}$. In step 3, let $f_l^n = \bar{f}^n$ for $n \le L$, and $f_l^n = \underline{f}^n$ for $n > L$. The output variable y_l can then be expressed as follows:

$$
\begin{aligned}
y_l &= \frac{\sum_{n=1}^{L} \bar{f}^n y_l^n + \sum_{L+1}^{N} \underline{f}^n y_l^n}{\sum_{n=1}^{L} \bar{f}^n + \sum_{n=L+1}^{N} \underline{f}^n} = \sum_{n=1}^{L} \bar{q}_l^n y_l^n \\
&+ \sum_{n=L+1}^{N} \underline{q}_l^n y_l^n = \left[\bar{\mathbf{y}}_l^T \ \underline{\mathbf{y}}_l^T \right] \left[\frac{\mathbf{Q}_l}{\underline{\mathbf{Q}}_l} \right] = \boldsymbol{\theta}_l^T \boldsymbol{\xi}_l
\end{aligned}
\tag{18}
$$

with $q^n = \underline{f}^n/D_l$, $\bar{q}_l^n = \bar{f}^n/D_r$, $D_l = \sum_{n=1}^{L} \bar{f}^n + \sum_{n=L+1}^{N} \underline{f}^n$, $\boldsymbol{\theta}_l = \left[\frac{\bar{\mathbf{y}}_l}{\underline{\mathbf{y}}_l} \right]$, $\boldsymbol{\xi}_r = \left[\frac{\mathbf{Q}_l}{\bar{\mathbf{Q}}_r} \right]$.

(e) *Defuzzifier* The type reduction output is fed into the fuzzy defuzzification. The crisp output is calculated using the average value of the two endpoints, and can be written as follows:

$$
y(\mathbf{x}) = \frac{y_l + y_r}{2} = \frac{1}{2}(\boldsymbol{\theta}_l^T \boldsymbol{\xi}_l + \boldsymbol{\theta}_r^T \boldsymbol{\xi}_r)
\tag{19}
$$

3.2 Proposed IT2FPISMC Method

The sliding mode controller in Eq. (7) can be realized when functions f(**x**) and g(**x**) are known in advance. The functions f(**x**) and g(**x**) cannot be exactly obtained in practical systems. Based on the universal approximation, the fuzzy systems $\hat{f}(\mathbf{x}|\theta_f)$ and $\hat{g}(\mathbf{x}|\theta_g)$ are designed to approach functions f(**x**) and g(**x**), respectively. The adaptive PI controller combined with IT2FSMC was investigated herein to alleviate the chattering phenomenon and enhance the steady-state response. Figure 3 shows the architecture of the proposed IT2FPISMC for a single-axis motion system. The error state s(**e**) in Eq. (4) lies in the pre-specific boundary layer ($|s| < \varphi$, φ is the thickness of the boundary layer); hence, the discontinuous switch term in Eq. (7) can be substituted by the continuous PI scheme. The PI controller is presented as follows:

$$
\hat{u}_p = \theta_P \cdot s(\mathbf{e}) + \theta_I \cdot \int s(\mathbf{e}) dt
\tag{20}
$$

where $\hat{u}_p$ is the PI controller output; and θ_P and θ_I are the proportional (P) and integral (I) gains, respectively.

This can be rewritten as the following vector form:

$$
\hat{u}_p(s|\boldsymbol{\theta}_p) = \boldsymbol{\theta}_p^T \boldsymbol{\psi}(s)
\tag{21}
$$

where $\boldsymbol{\theta}_p = [\theta_p, \theta_I]^T \in \mathbf{R}^2$ is an adjustable parameter vector, and $\boldsymbol{\psi}(s) = [s(\mathbf{e}), \int s(\mathbf{e}) dt]^T \in \mathbf{R}^2$ is the regression vector. The control input $\hat{u}_p$ is kept at a saturation value because the error state is bigger than the boundary layer ($|s| \ge \varphi$). Accordingly, the PI control input $\hat{u}_p$ can be expressed as follows:

$$
\hat{u}_p(s|\boldsymbol{\theta}_p) = \begin{cases} \boldsymbol{\theta}_p^T \boldsymbol{\psi}(s) & |s| < \varphi \\ \eta_\Delta \operatorname{sgn}(s) & |s| \ge \varphi \end{cases}
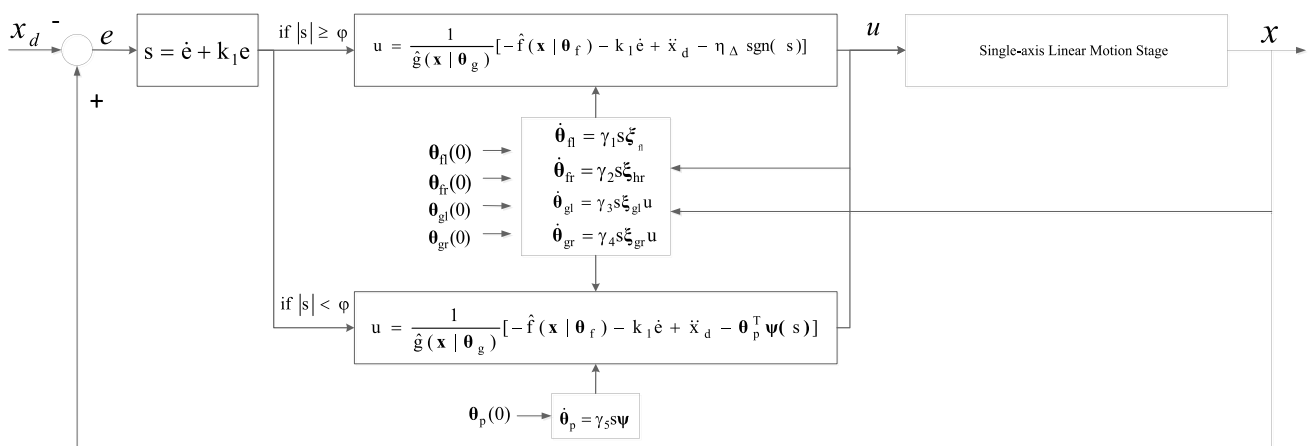\tag{22}
$$

Fig. 3 The structure of the proposed IT2FPISMC motion control stage system

The adaptive control law is expressed as follows:

$$u = \frac{1}{\hat{g}(\mathbf{x}|\theta_g)}\left[-\hat{f}\left(\mathbf{x}|\theta_f\right) - k_1 e + \ddot{x}_d - \hat{u}_p(s|\theta_p)\right] \quad (23)$$

w i t h $\quad \hat{f}(\mathbf{x}|\theta_f) = \frac{\hat{f}_l + \hat{f}_r}{2} = \frac{\theta_{fl}^T \xi_l + \theta_{fr}^T \xi_r}{2} = \theta_f^T \xi_f$,

$\hat{g}(\mathbf{x}|\theta_g) = \frac{\hat{g}_l + \hat{g}_r}{2} = \frac{\theta_{gl}^T \xi_l + \theta_{gr}^T \xi_r}{2} = \theta_g^T \xi_g, \hat{u}_p(s|\theta_p) = \theta_p^T \psi(s).$

The chattering of the sliding surface can be eliminated using a PI controller. This PI controller can be applied to substitute for the switching control when $|s| < \varphi$; otherwise, the control input was maintained at the saturation value when $|s| \geq \varphi$. Thus, $\hat{u}_p(\mathbf{z}|\theta_p) = \eta_\Delta \text{sgn}(s)$ if $|s| \geq \varphi$ happens. The adaptation control law can be designed and selected as follows:

$$\dot{\theta}_{fl} = \gamma_1 s \xi_{fl}(\mathbf{x}) \quad (24)$$

$$\dot{\theta}_{fr} = \gamma_2 s \xi_{fr}(\mathbf{x}) \quad (25)$$

$$\dot{\theta}_{gl} = \gamma_3 s \xi_{gl}(\mathbf{x}) u(t) \quad (26)$$

$$\dot{\theta}_{gr} = \gamma_4 s \xi_{gr}(\mathbf{x}) u(t) \quad (27)$$

$$\dot{\theta}_p = \gamma_5 s \psi(s) \quad (28)$$

where $\gamma_1 > 0$, $\gamma_2 > 0$, $\gamma_3 > 0$, $\gamma_4 > 0$, and $\gamma_5 > 0$ are the adaptation rates. The abovementioned control law can be modified by the projection algorithm to ensure that the adaptive parameters are bounded [21, 23]. Parameters, M_{fl}, M_{fr}, M_{gl}, M_{gr}, and M_p are the prespecified boundaries of the estimated vectors of θ_{fl}, θ_{fr}, θ_{gl}, θ_{gr}, and θ_p, respectively. Equations (24) and (25) can be modified as follows:

$$\dot{\theta}_{fl} = \begin{cases} \gamma_1 s \xi_{fl} & \left(\left|\theta_{fl}\right| < M_{fl}\right) \text{ or } \left(\left|\theta_{fl}\right| = M_{fl} \text{ and } s\theta_{fl}^T \xi_{fl} \leq 0\right) \\ P_{fl}\left(\gamma_1 s \xi_{fl}\right) & \left(\left|\theta_{fl}\right| \geq M_{fl} \text{ and } s\theta_{fl}^T \xi_{fl} > 0\right) \end{cases}$$
$$(29)$$

$$\dot{\theta}_{fr} = \begin{cases} \gamma_2 s \xi_{fr} & \left(\left|\theta_{fr}\right| < M_{fr}\right) \text{ or } \left(\left|\theta_{fr}\right| = M_{fr} \text{ and } s\theta_{fr}^T \xi_{fr} \leq 0\right) \\ P_{fr}\left(\gamma_2 s \xi_{fr}\right) & \left(\left|\theta_{fr}\right| \geq M_{fr} \text{ and } s\theta_{fr}^T \xi_{fr} > 0\right) \end{cases}$$
$$(30)$$

w i t h $\quad P_{fl}(\gamma_1 s \xi_{fl}) = \gamma_1 s \xi_{fl} - \gamma_1 s \frac{\theta_{fl} \theta_{fl}^T \xi_{fl}}{|\theta_{fl}|^2}$,

$P_{fr}(\gamma_2 s \xi_{fr}) = \gamma_2 s \xi_{fr} - \gamma_2 s \frac{\theta_{fr} \theta_{fr}^T \xi_{fr}}{|\theta_{fr}|^2}$, where M_{fl} and M_{fr} are the parameter boundaries of the estimated parameters θ_{fl} and θ_{fr},

respectively. Equations (26) and (27) can then be modified as follows:

$$\dot{\theta}_{gl} = \begin{cases} \gamma_3 s \xi_{gl} u & \left(\left|\theta_{gl}\right| < M_{gl}\right) \text{ or } \left(\left|\theta_{gl}\right| = M_{gl} \text{ and } s\theta_{gl}^T \xi_{gl} u \leq 0\right) \\ P_{gl}\left(\gamma_3 s \xi_{gl} u\right) & \left(\left|\theta_{gl}\right| \geq M_{gl} \text{ and } s\theta_{gl}^T \xi_{gl} u > 0\right) \end{cases}$$
$$(31)$$

$$\dot{\theta}_{gr} = \begin{cases} \gamma_4 s \xi_{gr} u & \left(\left|\theta_{gr}\right| < M_{gr}\right) \text{ or } \left(\left|\theta_{gr}\right| = M_{gr} \text{ and } s\theta_{gr}^T \xi_{gr} u \leq 0\right) \\ P_{gr}\left(\gamma_4 s \xi_{gr} u\right) & \left(\left|\theta_{gr}\right| \geq M_{gr} \text{ and } s\theta_{gr}^T \xi_{gr} u > 0\right) \end{cases}$$
$$(32)$$

w i t h $\quad P_{gl}(\gamma_3 s \xi_{gl} u) = \gamma_3 s \xi_{gl} u - \gamma_3 s \frac{\theta_{gl} \theta_{gl}^T \xi_{gl} u}{|\theta_{gl}|^2}$,

$P_{gr}(\gamma_4 s \xi_{gr} u) = \gamma_4 s \xi_{gr} u - \gamma_4 s \frac{\theta_{gr} \theta_{gr}^T \xi_{gr} u}{|\theta_{gr}|^2}$, where M_{gl} and M_{gr} are the parameter boundaries of the estimated parameters θ_{gl} and θ_{gr}, respectively. Equation (28) can then be modified as:

$$\dot{\theta}_p = \begin{cases} \gamma_5 s \psi(s) & \left(\left|\theta_p\right| < M_p\right) \text{ or } \left(\left|\theta_p\right| = M_p \text{ and } s\theta_p^T \psi(s) \geq 0\right) \\ P_p\left(\gamma_5 s \psi(s)\right) & \left(\left|\theta_p\right| \geq M_p \text{ and } s\theta_p^T \psi(s) < 0\right) \end{cases}$$
$$(33)$$

with $P_p(\gamma_5 s \psi) = \gamma_5 s \psi - \gamma_5 s \frac{\theta_p \theta_p^T \psi}{|\theta_p|^2}$, where M_p denotes the parameter boundaries of the estimated parameters θ_p.

3.3 Stability Analysis

Theorem 1 *The nonlinear system is considered in the form of Eqs.* (1) *and* (2). *If the control law* (23) *can be used in the system, the functions* $\hat{f}(\mathbf{x})$, $\hat{g}(\mathbf{x})$, *and* $\hat{u}_p$ *are approximated by Eqs.* (19) *and* (22), *and the parameter vectors of* θ_{fl}, θ_{fr}, θ_{gl}, θ_{gr}, *and* θ_p *are adjusted by adaptive laws* (24)–(28), *respectively. The signals of the closed-loop control system are bounded, and the trajectory errors asymptotically converge to zero.*

Proof The optimal parameter vectors can be defined as follows:

$$\theta_f^* = \arg \min_{\theta_f \in \Omega_f} \left[\sup_{\mathbf{x} \in R^n} \left|\hat{f}\left(\mathbf{x}|\theta_f\right) - f(\mathbf{x})\right|\right] \quad (34)$$

$$\theta_g^* = \arg \min_{\theta_g \in \Omega_g} \left[\sup_{\mathbf{x} \in R^n} \left|\hat{g}\left(\mathbf{x}|\theta_g\right) - g(\mathbf{x})\right|\right] \quad (35)$$

$$\theta_p^* = \arg \min_{\theta_p \in \Omega_p} \left[\sup_{\mathbf{z} \in R^2} \left|\hat{u}_p(s|\theta_p) - u_{sw}\right|\right] \quad (36)$$

where Ω_f, Ω_g, and Ω_p are the constant set values for θ_f, θ_g, and θ_p, respectively. The stability of the design control must be guaranteed by the assumption of the parameter

boundaries expressed in Eq. (29)–(33). The parameters can be defined as follows:

$$\Omega_f = \{\boldsymbol{\theta}_f \in \mathbf{R}^n \,|\, |\boldsymbol{\theta_f}| \leq M_f\} \tag{37}$$

$$\Omega_g = \{\boldsymbol{\theta}_g \in \mathbf{R}^n \,|\, 0 < \varepsilon \leq \left|\boldsymbol{\theta}_g\right| \leq M_g\} \tag{38}$$

$$\Omega_p = \{\boldsymbol{\theta}_p \in \mathbf{R}^2 \,|\, |\boldsymbol{\theta}_p| \leq M_p\} \tag{39}$$

with $M_f = \frac{M_{fl} + M_{fr}}{2}$, $M_g = \frac{M_{gl} + M_{gr}}{2}$, where M_f, M_g, and M_p are positive constants. The time derivative of the sliding surface error function $s(\mathbf{e})$ is expressed as follows:

$$
\begin{aligned}
\dot{s} &= k_1 \dot{e} + f(\mathbf{x}) + g(\mathbf{x})u + d - \ddot{x}_d \\
&= f(\mathbf{x}) - \hat{f}\left(\mathbf{x}\big|\boldsymbol{\theta}_f\right) + g(\mathbf{x})u - \hat{g}\left(\mathbf{x}\big|\boldsymbol{\theta}_g\right)u + d - \hat{u}_p(s|\boldsymbol{\theta}_p) \\
&= \hat{f}\left(\mathbf{x}\big|\boldsymbol{\theta}_f^*\right) - \hat{f}\left(\mathbf{x}\big|\boldsymbol{\theta}_f\right) + \left[\hat{g}\left(\mathbf{x}\big|\boldsymbol{\theta}_g^*\right)u - \hat{g}\left(\mathbf{x}\big|\boldsymbol{\theta}_g\right)\right]u + \left[\hat{u}_p(s|\boldsymbol{\theta}_p^*) - \hat{u}_p(s|\boldsymbol{\theta}_p)\right] + \omega - \hat{u}_p(s|\boldsymbol{\theta}_p^*) \\
&= \left(\boldsymbol{\theta}_f^{*T}\boldsymbol{\xi}_f - \boldsymbol{\theta}_f^T\boldsymbol{\xi}_f\right) + \left(\boldsymbol{\theta}_g^{*T}\boldsymbol{\xi}_g - \boldsymbol{\theta}_g^T\boldsymbol{\xi}_g\right)u + \left(\boldsymbol{\theta}_p^{*T}\boldsymbol{\psi} - \boldsymbol{\theta}_p^T\boldsymbol{\psi}\right) + d - \hat{u}_p(s|\boldsymbol{\theta}_p^*) + \omega \\
&= \boldsymbol{\varphi}_f^T\boldsymbol{\xi}_f + \boldsymbol{\varphi}_g^T\boldsymbol{\xi}_g u + \boldsymbol{\varphi}_p^T\boldsymbol{\psi} + d - \hat{u}_p(s|\boldsymbol{\theta}_p^*) + \omega \\
&= \frac{1}{2}\left(\boldsymbol{\varphi}_{fl}^T\boldsymbol{\xi}_{fl} + \boldsymbol{\varphi}_{fr}^T\boldsymbol{\xi}_{fr}\right) + \frac{1}{2}\left(\boldsymbol{\varphi}_{gl}^T\boldsymbol{\xi}_{gl} + \boldsymbol{\varphi}_{gr}^T\boldsymbol{\xi}_{gr}\right)u + \boldsymbol{\varphi}_p^T\boldsymbol{\psi} + d - \hat{u}_p(s|\boldsymbol{\theta}_p^*) + \omega
\end{aligned}
\tag{40}
$$

with $\omega = [f(\mathbf{x}) - \hat{f}(\boldsymbol{x}|\boldsymbol{\theta}_f^*)] + [g(\mathbf{x}) - \hat{g}(\boldsymbol{x}|\boldsymbol{\theta}_g^*)]u$, $\boldsymbol{\varphi}_f = \boldsymbol{\theta}_f^* - \boldsymbol{\theta}_f$, $\boldsymbol{\varphi}_{fl} = \boldsymbol{\theta}_{fl}^* - \boldsymbol{\theta}_{fl}$, $\boldsymbol{\varphi}_{fr} = \boldsymbol{\theta}_{fr}^* - \boldsymbol{\theta}_{fr}$ $\boldsymbol{\varphi}_g = \boldsymbol{\theta}_g^* - \boldsymbol{\theta}_g$, $\boldsymbol{\varphi}_{gl} = \boldsymbol{\theta}_{gl}^* - \boldsymbol{\theta}_{gl}$, $\boldsymbol{\varphi}_{gr} = \boldsymbol{\theta}_{gr}^* - \boldsymbol{\theta}_{gr}$, and $\boldsymbol{\varphi}_p = \boldsymbol{\theta}_p^* - \boldsymbol{\theta}_p$, where ω is the minimum approximation error. The Lyapunov candidate function can be defined as follows:

$$
V = \frac{1}{2}\left(s^2 + \frac{1}{2\gamma_1}\boldsymbol{\varphi}_{fl}^T\boldsymbol{\varphi}_{fl} + \frac{1}{2\gamma_2}\boldsymbol{\varphi}_{fr}^T\boldsymbol{\varphi}_{fr} + \frac{1}{2\gamma_3}\boldsymbol{\varphi}_{gl}^T\boldsymbol{\varphi}_{gl} + \frac{1}{2\gamma_4}\boldsymbol{\varphi}_{gr}^T\boldsymbol{\varphi}_{gr} + \frac{1}{\gamma_5}\boldsymbol{\varphi}_p^T\boldsymbol{\varphi}_p\right)
\tag{41}
$$

where γ_n is a positive constant value (n = 1, 2, 3, 4, 5). The time derivative of function V can be expressed as follows:

$$
\begin{aligned}
\dot{V} &= s\dot{s} + \frac{1}{2\gamma_1}\boldsymbol{\varphi}_{fl}^T\dot{\boldsymbol{\varphi}}_{fl} + \frac{1}{2\gamma_2}\boldsymbol{\varphi}_{fr}^T\dot{\boldsymbol{\varphi}}_{fr} + \frac{1}{2\gamma_3}\boldsymbol{\varphi}_{gl}^T\dot{\boldsymbol{\varphi}}_{gl} + \frac{1}{2\gamma_4}\boldsymbol{\varphi}_{gr}^T\dot{\boldsymbol{\varphi}}_{gr} + \frac{1}{\gamma_5}\boldsymbol{\varphi}_p^T\dot{\boldsymbol{\varphi}}_p \\
&= s\left(\frac{1}{2}\left(\boldsymbol{\varphi}_{fl}^T\boldsymbol{\xi}_{fl} + \boldsymbol{\varphi}_{fr}^T\boldsymbol{\xi}_{fr}\right) + \frac{1}{2}\left(\boldsymbol{\varphi}_{gl}^T\boldsymbol{\xi}_{gl} + \boldsymbol{\varphi}_{gr}^T\boldsymbol{\xi}_{gr}\right)u + \boldsymbol{\varphi}_p^T\boldsymbol{\psi} + d - \hat{u}_p(s|\boldsymbol{\theta}_p^*) + \omega\right) \\
&\quad + \frac{1}{2\gamma_1}\boldsymbol{\varphi}_{fl}^T\dot{\boldsymbol{\varphi}}_{fl} + \frac{1}{2\gamma_2}\boldsymbol{\varphi}_{fr}^T\dot{\boldsymbol{\varphi}}_{fr} + \frac{1}{2\gamma_3}\boldsymbol{\varphi}_{gl}^T\dot{\boldsymbol{\varphi}}_{gl} \\
&\quad + \frac{1}{2\gamma_4}\boldsymbol{\varphi}_{gr}^T\dot{\boldsymbol{\varphi}}_{gr} + \frac{1}{\gamma_5}\boldsymbol{\varphi}_p^T\dot{\boldsymbol{\varphi}}_p = \frac{1}{2\gamma_1}\boldsymbol{\varphi}_{fl}^T\left(\dot{\boldsymbol{\varphi}}_{fl} + \gamma_1 s\boldsymbol{\xi}_{fl}\right) \\
&\quad + \frac{1}{2\gamma_2}\boldsymbol{\varphi}_{fr}^T\left(\dot{\boldsymbol{\varphi}}_{fr} + \gamma_2 s\boldsymbol{\xi}_{fr}\right) + \frac{1}{2\gamma_3}\boldsymbol{\varphi}_{gl}^T\left(\dot{\boldsymbol{\varphi}}_{gl}\right. \\
&\quad \left. + \gamma_3 s\boldsymbol{\xi}_{gl}u\right) + \frac{1}{2\gamma_4}\boldsymbol{\varphi}_{gr}^T\left(\dot{\boldsymbol{\varphi}}_{gr} + \gamma_4 s\boldsymbol{\xi}_{gr}u\right) \\
&\quad + \frac{1}{\gamma_5}\boldsymbol{\varphi}_p^T\left(\dot{\boldsymbol{\varphi}}_p + \gamma_5 s\boldsymbol{\psi}\right) + sd - s\hat{u}_p(s|\boldsymbol{\theta}_p^*) + s\omega \\
&\leq \frac{1}{2\gamma_1}\boldsymbol{\varphi}_{fl}^T\left(\dot{\boldsymbol{\varphi}}_{fl} + \gamma_1 s\boldsymbol{\xi}_{fl}\right) + \frac{1}{2\gamma_2}\boldsymbol{\varphi}_{fr}^T\left(\dot{\boldsymbol{\varphi}}_{fr} + \gamma_2 s\boldsymbol{\xi}_{fr}\right) \\
&\quad + \frac{1}{2\gamma_3}\boldsymbol{\varphi}_{gl}^T\left(\dot{\boldsymbol{\varphi}}_{gl} + \gamma_3 s\boldsymbol{\xi}_{gl}u\right) + \frac{1}{2\gamma_4}\boldsymbol{\varphi}_{gr}^T\left(\dot{\boldsymbol{\varphi}}_{gr} + \gamma_4 s\boldsymbol{\xi}_{gr}u\right) \\
&\quad + \frac{1}{\gamma_5}\boldsymbol{\varphi}_p^T\left(\dot{\boldsymbol{\varphi}}_p + \gamma_5 s\boldsymbol{\psi}\right) + sd - s(\eta_\Delta + D)sgn(s) + s\omega
\end{aligned}
\tag{42}
$$

with $\dot{\boldsymbol{\varphi}}_{fl} = -\dot{\boldsymbol{\theta}}_{fl}$, $\dot{\boldsymbol{\varphi}}_{fr} = -\dot{\boldsymbol{\theta}}_{fr}$, $\dot{\boldsymbol{\varphi}}_{gl} = -\dot{\boldsymbol{\theta}}_{gl}$, $\dot{\boldsymbol{\varphi}}_{gr} = -\dot{\boldsymbol{\theta}}_{gr}$, $\dot{\boldsymbol{\varphi}}_{p} = -\dot{\boldsymbol{\theta}}_{p}$.

By substituting the adaptive laws of Eqs. (24)–(28) into Eq. (42), the equation can be derived as follows:

$$
\begin{aligned}
\dot{V} &\leq sd - s(\eta_\Delta + D)sgn(s) + s\omega \\
&= sd - |s|(\eta_\Delta + D) + s\omega \\
&\leq -|s|\eta_\Delta + s\omega \leq 0
\end{aligned}
\tag{43}
$$

Based on the universal approximation theorem, the minimum approximation error ω becomes a very small value, and can result in $\dot{V} \leq 0$. One can state [21] that all the feedback system signals (s, $\boldsymbol{\theta}_{fl}$, $\boldsymbol{\theta}_{fr}$, $\boldsymbol{\theta}_{gl}$, $\boldsymbol{\theta}_{gr}$, and $\boldsymbol{\theta}_p$) were bounded. The sliding surface error $s(\mathbf{e}) = \mathbf{k}^T\mathbf{e}$, and $\mathbf{e}$ was bounded if $e(0)$ was bounded. This finding implied that the reference trajectory $x_d(t)$ was bounded, and state $x(t)$ was bounded accordingly. We need to prove $\lim_{t\to\infty}|s| = 0$ to imply that $\lim_{t\to\infty}|e(t)| = 0$ and establish an asymptotic convergence. Suppose we chose variable $\eta_d > 0$, then Eq. (43) is given as follows:

$$\dot{V} \leq -|\omega||s| - \eta_\Delta|s| \leq -\eta_d|s| \tag{44}$$

with $\eta_d = (\eta_\Delta + |\omega|)$.

Integrating the above equation with respect to time, it yields:

$$\int_0^t |s(\tau)| d\tau \leq \frac{1}{\eta_d}[V(0) - V(t)] \tag{45}$$

$V(0)$ is bounded, and $V(t)$ is non-increasing and bounded. Hence, we find that the sliding surface error s(**e**) is bounded, implying that $s \in L_\infty$ [21]. With the use of Barbalat's lemma [1], s(**e**) will converge to zero as t → ∞. This lemma also implies that $\lim_{t\to\infty} |e(t)| = 0$. Thus, the designed system is stable, and the error will asymptotically converge to zero. Therefore, the stability of our developed IT2FPISMC is guaranteed.

4 Experimental Results

4.1 Contour Planning

Contour planning plays an important role in the control of a precise XY table. The mathematical NURBS method was used herein to generate the curve trajectory. The general form of the NURBS curve interpolator can be expressed as follows:

$$C(p) = \sum_{i=0}^{I} R_{i,k_d}(p)V_i = \frac{\sum_{i=0}^{I} N_{i,k_d}(p)w_i V_i}{\sum_{i=0}^{I} N_{i,k_d}(p)w_i} = \frac{A(p)}{w(p)} \tag{46}$$

$$R_{i,k}(u) = \frac{N_{i,k_d}(p)w_i}{\sum_{i=0}^{I} N_{i,k_d}(p)w_i} \tag{47}$$

where $\{V_i\}$ denotes the control points; $\{w_i\}$ denotes the weights of $\{V_i\}$; $\{w_i V_i\}$ denotes the weighted control points; $\{A(p)\}$ is the weighted B-spline; $\{w(p)\}$ is the weighting function; k_d is the degree of the NURBS curve; and (I + 1) is the number of control point. $\{N_{i,k_d}(p)\}$ is the k_d^{th}-degree B-spline basis function, and $\{R_{i,k_d}(p)\}$ is the rational B-spline basis function. The k_d^{th}-degree B-spline basis function can be expressed as follows:

$$N_{i,0}(p) = \begin{cases} 1 & for\ p_1 \leq p \leq p_{i+1} \\ 0 & otherwise \end{cases} \tag{48}$$

$$N_{i,k_d}(p) = \frac{(p - p_i)}{(p_{i+k_d-1} - p_i)}N_{i,k_d-1}(p)$$
$$+ \frac{(p_{i+k_d} - p)}{(p_{i+k_d} - p_{i+1})}N_{i+1,k_d-1}(p)$$
$$i = 0, 1, 2, \ldots, I \tag{49}$$

where $\mathbf{P} = \{p_0, p_1, \ldots, p_{I+k_d}\}$ represents the knot vector.

Four contour shapes were applied for this proposed adaptive IT2FPISMC control system, namely circular, bowknot, heart, and star curves. To generate these contours, the degree of the NURBS curve k_d, control points **V**, knot vectors **P**, and weights of the NURBS curve **w** are expressed as follows:

1. Circular contour: The parameters of the NURBS curve are set as follows:

$$k_d = 2 \tag{50}$$

$$\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(2.5, 0), (2.5, 2.5), (-2.5, 2.5), (-2.5, 0),$$
$$(-2.5, -2.5), (2.5, -2.5), (2.5, 0)] \text{ (unit : cm, cm)} \tag{51}$$

$$\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9]$$
$$= [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1] \tag{52}$$

$$\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 0.5, 0.5, 1, 0.5, 0.5, 1] \tag{53}$$

2. Bowknot contour: The parameters of the NURBS curve are set as follows:

$$k_d = 2 \tag{54}$$

$$\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(0, 0), (-1.5, -1.5),$$
$$(-1.5, 1.5), (0, 0), (1.5, -1.5), (1.5, 1.5), (0, 0)]$$
$$\text{(unit : cm, cm)} \tag{55}$$

$$\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9]$$
$$= [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1] \tag{56}$$

$$\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 2.5, 2.5, 1, 2.5, 2.5, 1] \tag{57}$$

3. Heart contour: The parameters of the NURBS curve are set as follows:

$$k_d = 2 \tag{58}$$

$$\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6] = [(0, 0), (-3, 2), (-2, 5),$$
$$(0, 3.6), (2, 5), (3, 1), (0, 0)] \text{ (unit: cm, cm)} \tag{59}$$

$$\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9]$$
$$= [0, 0, 0, 0.25, 0.5, 0.5, 0.75, 1, 1, 1] \tag{60}$$

$$\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6] = [1, 1, 1, 1, 1, 1, 1] \tag{61}$$

4. Star contour: The parameters of the NURBS curve are set as follows:

$$k_d = 2 \qquad (62)$$

$$\mathbf{V} = [V_0, V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}, V_{12}]$$
$$= [(0,6), (-0.5,6), (-1.5,4), (-4,4), (-2,2), (-2.5,0), (0,1.5),$$
$$(2.5,0), (2,2), (4,4)(1.5,4), (0.5,6), (0,6)] \quad (\text{unit}: \text{ cm, cm})$$
$$(63)$$

$$\mathbf{P} = [P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{14}]$$
$$= [0, 0, 0, 0.1, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.8, 1, 1, 1] \qquad (64)$$

$$\mathbf{w} = [w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}, w_{11}, w_{12}]$$
$$= [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1] \qquad (65)$$

Figure 4a–d depict the desired contours of the circle, bowknot, heart, and star, respectively, based on the abovementioned parameter setting.

4.2 Parameter Setting and Performance Measurement

The experiments were conducted to fulfill the performances of the adaptive IT2FPISMC control. The contour error is the difference between the desired contour and the real contour. The average tracking error (ATE) E_m can be defined as follows:

$$E_m = \sum_{k=1}^{K} \frac{E(k)}{n} \qquad (66)$$

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. The tracking error standard deviation (TESD), E_{STD}, can be written as follows:

$$E_{\text{STD}} = \sqrt{\frac{\sum_{k=1}^{K}(E(k) - E_m)^2}{K}} \qquad (67)$$

The ATE is used to compare the tracking trajectory performance, whereas the TESD is used to measure the oscillation of the contour tracking. Three types of control algorithms are compared herein: (a) the conventional PID control method, (b) the conventional IT2FSMC method with sign function, and (c) the proposed IT2FPISMC method. The membership functions for the fuzzy controllers $\hat{f}(\mathbf{x})$ and $\hat{g}(\mathbf{x})$ are designed as follows:

Fig. 4 The designed NURBS curves, **a** circular contour, **b** bowknot contour, **c** heart contour, and **d** star contour

(a)

(b)

(c)

(d)

$$\overline{u}_{\tilde{X}_1^n}(x_i) = exp[-((x_i + \pi/12)/(\pi/18))^2],$$

$$\underline{u}_{\tilde{X}_1^n}(x_i) = a \cdot exp[-((x_i + \pi/12)/(\pi/24))^2]$$

$$\overline{u}_{\tilde{X}_2^n}(x_i) = exp[-((x_i + \pi/6)/(\pi/18))^2],$$

$$\underline{u}_{\tilde{X}_2^n}(x_i) = a \cdot exp[-((x_i + \pi/6)/(\pi/24))^2]$$

$$\overline{u}_{\tilde{X}_3^n}(x_i) = exp[-((x_i)/(\pi/18))^2],$$

$$\underline{u}_{\tilde{X}_3^n}(x_i) = a \cdot exp[-((x_i)/(\pi/24))^2],$$

$$\overline{u}_{\tilde{X}_4^n}(x_i) = exp[-((x_i - \pi/6)/(\pi/18))^2],$$

$$\underline{u}_{\tilde{X}_4^n}(x_i) = a \cdot exp[-((x_i - \pi/6)/(\pi/24))^2]$$

$$\overline{u}_{\tilde{X}_5^n}(x_i) = exp[-((x_i - \pi/12)/(\pi/18))^2],$$

$$\underline{u}_{\tilde{X}_5^n}(x_i) = a \cdot exp[-((x_i - \pi/12)/(\pi/24))^2],$$

with $a = \begin{cases} 0.8 & i = 1 \\ 0.5 & i = 2 \end{cases}$, where x_1 and x_2 are the position and the velocity variables, respectively. The initial state was selected as $\mathbf{x}(0) = [x_1(0), x_2(0)]^T = [1, 0]^T$. We set all initial values of $\boldsymbol{\theta}_{fl}(0)$, $\boldsymbol{\theta}_{fr}(0)$, $\boldsymbol{\theta}_{gl}(0)$, and $\boldsymbol{\theta}_{gr}(0)$ to be 0.1 and $\boldsymbol{\theta}_p(0)$ to be $[0.02, 0.04]^T$. The 25 adjusted variables existed in these regressive vectors. The sampling frequency of our IT2FPISMC system was 1 kHz for the position loop.

4.3 Experimentation

The proposed control algorithms in the experiments were realized and implemented. A platform with 1 kHz sample frequency was applied for the encoder interface and control algorithm execution. The system read the position information of the XY stage from the optical encoder sensors. The tracking errors and their derivatives were then computed. The control commands according to our proposed IT2FPISMC algorithm were calculated and sent to the servo drive by two DAC interfaces. Table 1 shows the parameter settings used for the *x*- and *y*-axes in the following experiments. These parameters were determined by empirical rules to achieve a better transient and steady-state response in the

experimentation condition considering the stability requirement. The tracking performances are illustrated as follows:

1. *Circular contour* It shows the experimental results of the proposed IT2FPISMC controller for the circle contour in Fig. 5. The total control point was seven. Figure 4a presents the reference trajectory. Figure 5a, b illustrate the *x*- and *y*-axis direction responses, respectively. Figure 5c, d depict the x- and y-axis tracking errors. The stage position of each axis can precisely track the desired command. Figure 5e illustrates the tracking responses of the motion trajectory of the two-dimensional circular contour. Figure 6a–e show the experimental results of the conventional IT2FSMC method with sign function. The displacement errors can be effectively reduced and lie within 30 μm as the proposed control method was being developed. The ATE of our proposed scheme was 22.346 μm, and the TESD was 4.632 μm.

2. *Bowknot contour* We selected seven control points in the bowknot contour. Figure 4b shows the corresponding tracking trajectory. Figure 7a–e present the experiment results of the XY stage system for the IT2FPISMC structure, including tracking responses, associated tracking errors, and bowknot trajectory. These results indicated that both axes exhibited small error responses, and the tracking performances can be guaranteed. The tracking contours were close to the reference paths, and the displacement errors were significantly reduced in the tracking process. Figure 8a–e show the experimental results of the conventional IT2FSMC method with sign function for bowknot trajectory. The ATE was 15.956 μm, and the associated TESD for the bowknot curve was 12.132 μm. Our proposed scheme had a favorable accuracy, and the maximum position error can be controlled within 50 μm.

3. *Heart contour* The total control point in the heart contour was seven. Figure 4c shows the associated trajectory. The tracking performances, tracking errors, and

| Table 1 Experimental parameters	Controller methods	Parameters setting
	The PID method	$K_p = 200$, $K_i = 20$, $K_d = 10$
	The conventional IT2FSMC method (sgn)	
	X-axis	$\gamma_{1x} = 0.06$, $\gamma_{2x} = 0.06$, $\gamma_{3x} = 0.08$, $\gamma_{4x} = 0.08$, $k_1 = 50$, $\eta_\Delta = 1$
	Y-axis	$\gamma_{1y} = 0.06$, $\gamma_{2y} = 0.06$, $\gamma_{3y} = 0.08$, $\gamma_{4y} = 0.08$, $k_1 = 50$, $\eta_\Delta = 1$
	The proposed IT2FPISMC method	
	X-axis	$\gamma_{1x} = 0.06$, $\gamma_{2x} = 0.06$, $\gamma_{3x} = 0.08$, $\gamma_{4x} = 0.08$, $\gamma_{5x} = 0.04$, $k_1 = 50$, $\eta_\Delta = 1$, $\varphi = 0.5$, $M_{fl} = M_{fr} = M_{gl} = M_{gr} = M_p = 10$
	Y-axis	$\gamma_{1y} = 0.06$, $\gamma_{2y} = 0.06$, $\gamma_{3y} = 0.08$, $\gamma_{4y} = 0.08$, $\gamma_{5y} = 0.04$, $k_1 = 50$, $\eta_\Delta = 1$, $\varphi = 0.5$, $M_{fl} = M_{fr} = M_{gl} = M_{gr} = M_p = 10$
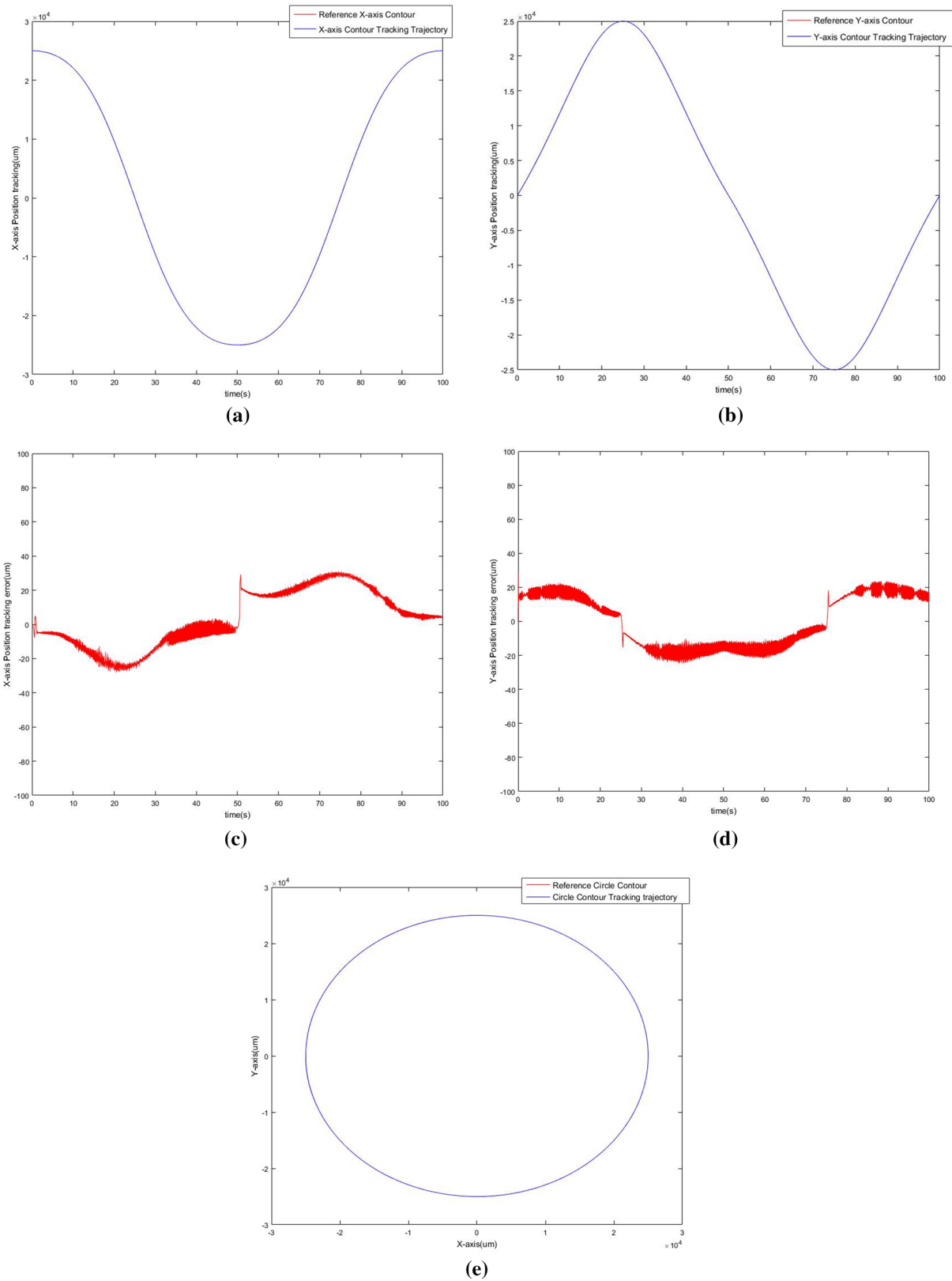
Fig. 5 Experimental results obtained using IT2FPISMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** circular trajectory
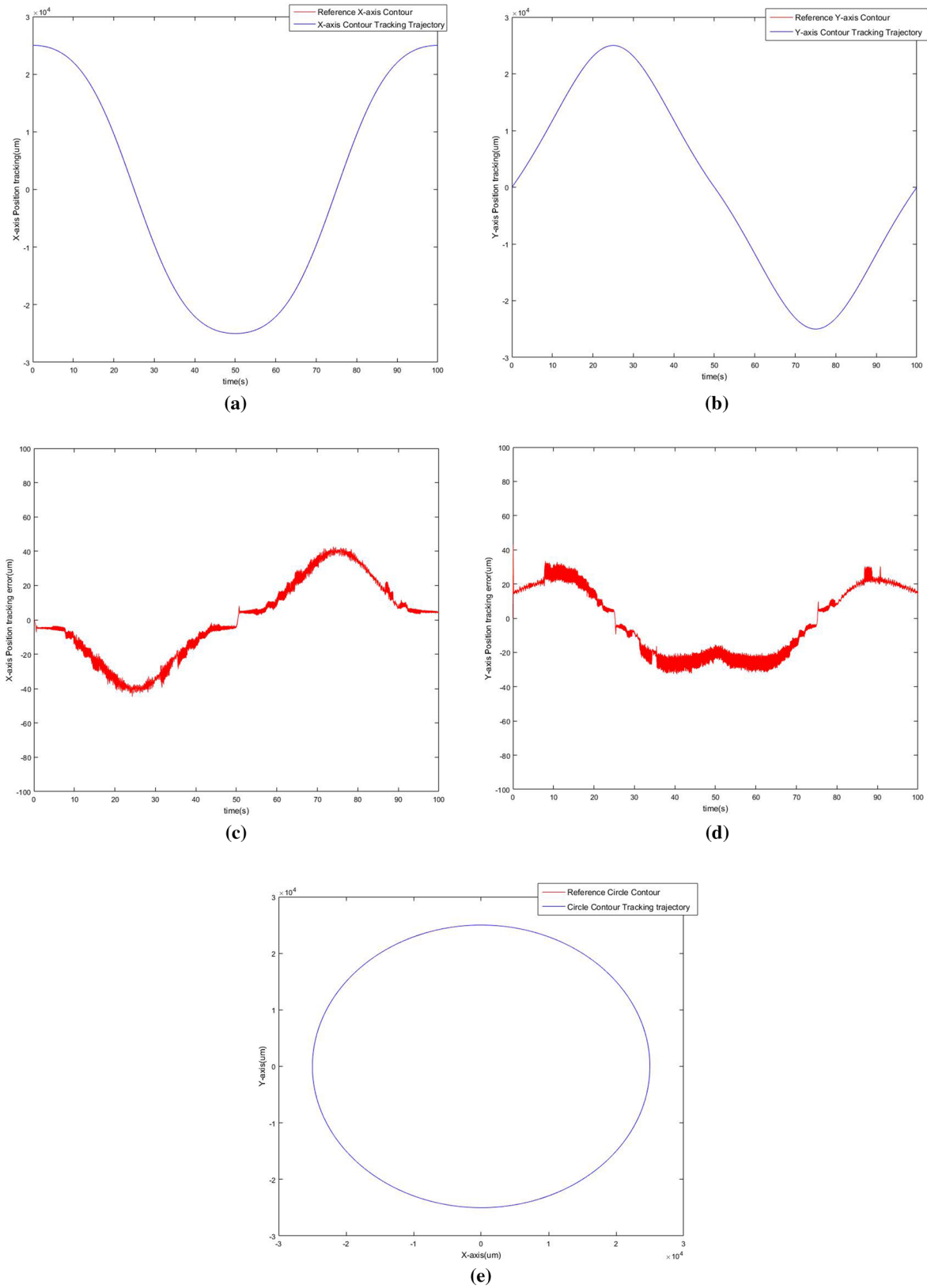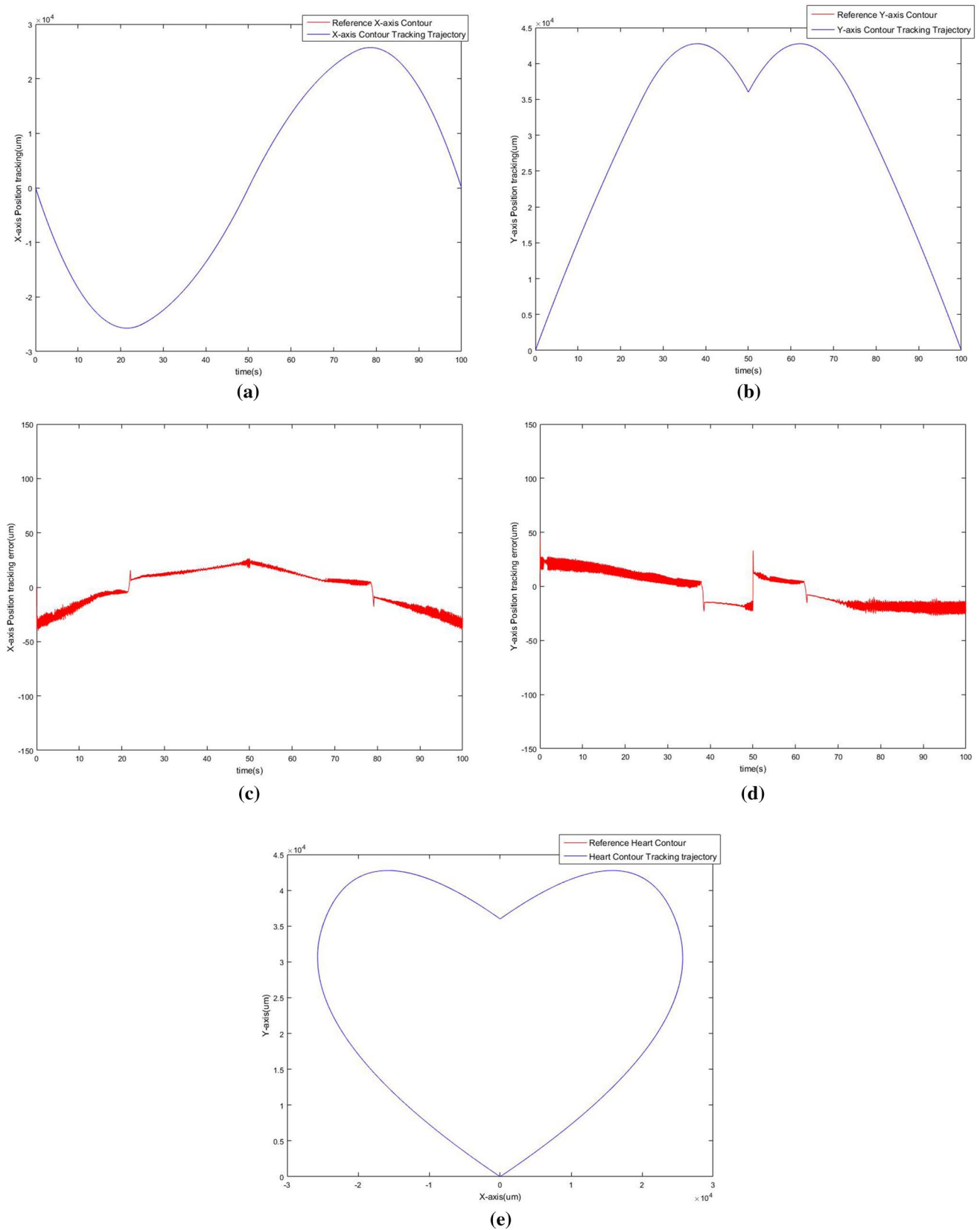
Fig. 6 Experimental results obtained using IT2FSMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** circular trajectory
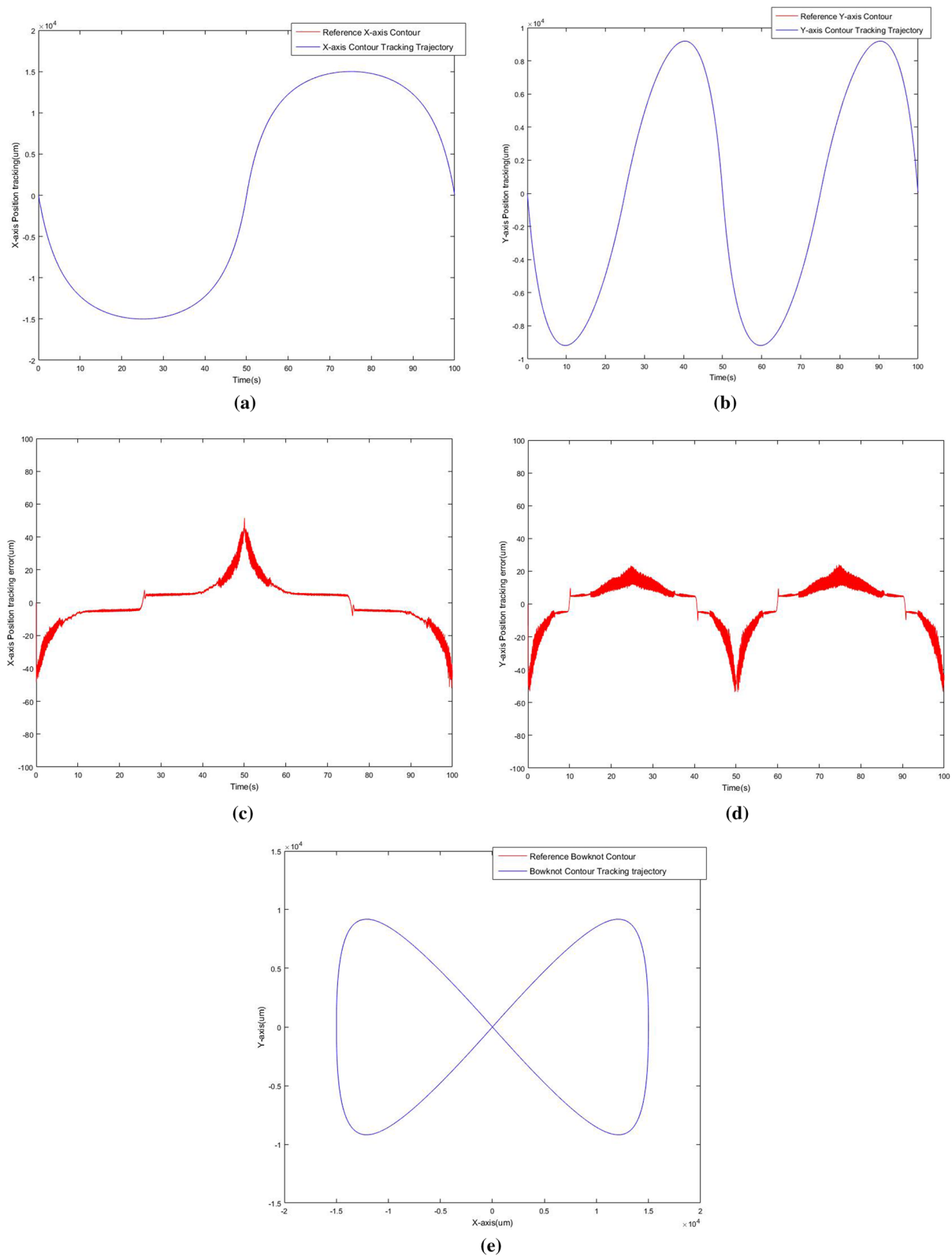
Fig. 7 Experimental results obtained using IT2FPISMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** bowknot curve trajectory
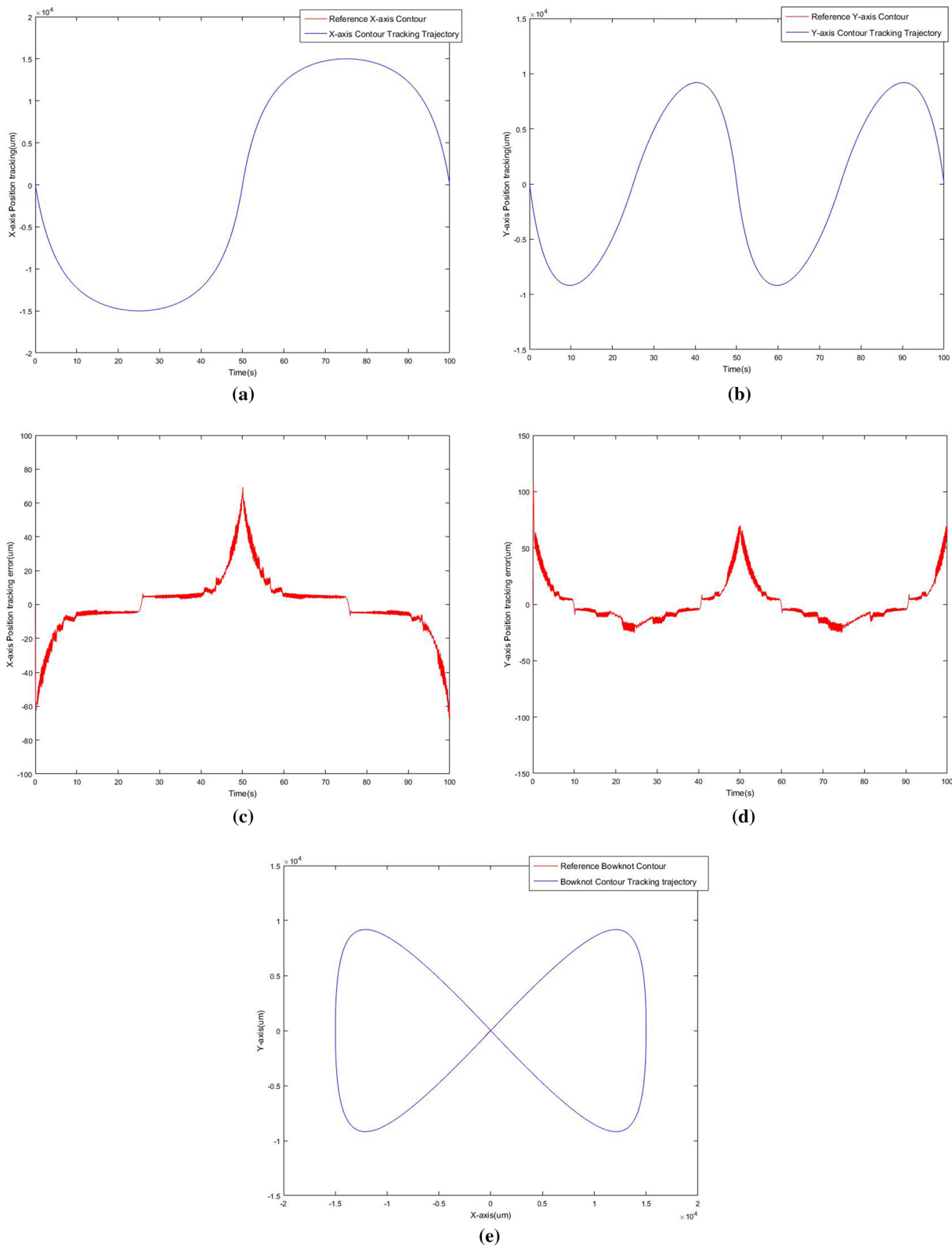
Fig. 8 Experimental results obtained using IT2FSMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** bowknot curve trajectory

Fig. 9 Experimental results obtained using IT2FPISMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** heart curve trajectory

Table 2 Experimental results of contour tracking errors

Trajectory contour	The PID method	The conventional IT2FSMC method (sgn)	The proposed IT2FPISMC method
Average tracking error E_m (μm)			
Circular curve	69.046	29.980	22.346
Bowknot curve	78.456	19.045	15.956
Heart curve	79.607	29.147	21.539
Star curve	53.704	17.407	15.350
Average	70.203	23.894	18.797
Tracking error standard deviation E_{STD} (μm)			
Circular curve	27.296	7.183	4.632
Bowknot curve	56.989	18.069	12.132
Heart curve	44.262	12.604	7.846
Star curve	46.112	9.100	6.085
Average	43.665	11.739	7.673

tracking trajectory for the *x*- and *y*-axes are demonstrated in Fig. 9a–e, respectively. The performances of the transient and steady responses illustrated the robustness of the proposed structure under parameter uncertainties and external disturbances. Figure 10a–e show the experimental results of the conventional IT2FSMC method with sign function for heart contour. The ATE of our proposed scheme was 21.539 μm, while the TESD for the heart contour was 7.846 μm.

4. *Star trajectory* We designed the total control point as 13. Figure 4d shows the reference outline. Figure 11a–e show the tracking responses of the position displacement, tracking error, and motion trajectory in the star contour, respectively. The proposed adaptive structure can handle the model uncertainty and alleviate the chattering phenomenon. Figure 12a–e show the experimental results of the conventional IT2FSMC method with sign function for star trajectory. The ATE of our proposed method was 15.350 μm, and the TESD was 6.085 μm. The proposed controller can guarantee the asymptotic stability and exhibit a better tracking capability.

Table 2 shows both the tracking errors of our IT2FPISMC method and those of the traditional PID and IT2FSMC methods. The proposed IT2FPISMC method demonstrated more accurate performances, showing a 21.33% improvement in the ATE and a 34.64% improvement in the TESD compared with the conventional IT2FSMC method. These results conclude that the proposed IT2FPISMC system achieved the lowest ATE and TESD tracking errors. Our proposed structure was robust with respect to unmodeled dynamics and external disturbances for different reference contours. In summary, the IT2FPISMC approach achieves the desired tracking performances while ensuring the robustness of the precision system (Figs. 5, 7, 9, 11).

5 Conclusion

The development of an IT2FPISMC control system for tracking the various contours was successfully demonstrated in this research. A type-2 fuzzy control structure that can handle the rules and parameter uncertainties was designed to approximate the unknown nonlinear function in the industrial XY-driven motion stage. Moreover, the fuzzy PI sliding mode control was more robust against external disturbances and alleviated the chattering condition. The adjustable parameter vectors of the adaptive type-2 fuzzy controller can be adaptively tuned by the output feedback control law, and asymptotic stability was guaranteed based on the Lyapunov synthesis approach. The real-time NURBS interpolation was realized. Accordingly, four contour trajectories, namely circular contour, bowknot contour, heart contour, and star contour, were experimented to illustrate the effectiveness of the proposed system. The experimental results were obtained, showing that the indirect control IT2FPISMC method can deal with chattering and significantly alleviate the contour errors. On average, it can achieve 21.33% and 34.64% improvement of the ATE and the TESD, respectively, compared with the conventional IT2FSMC strategy. From the performance indices of the experimental results, the proposed scheme can achieve good control performance and superior robustness for the two-axis trajectory control with regard to model uncertainties and external disturbances. Therefore, our developed strategy can be easily realized and implemented for positioning systems.
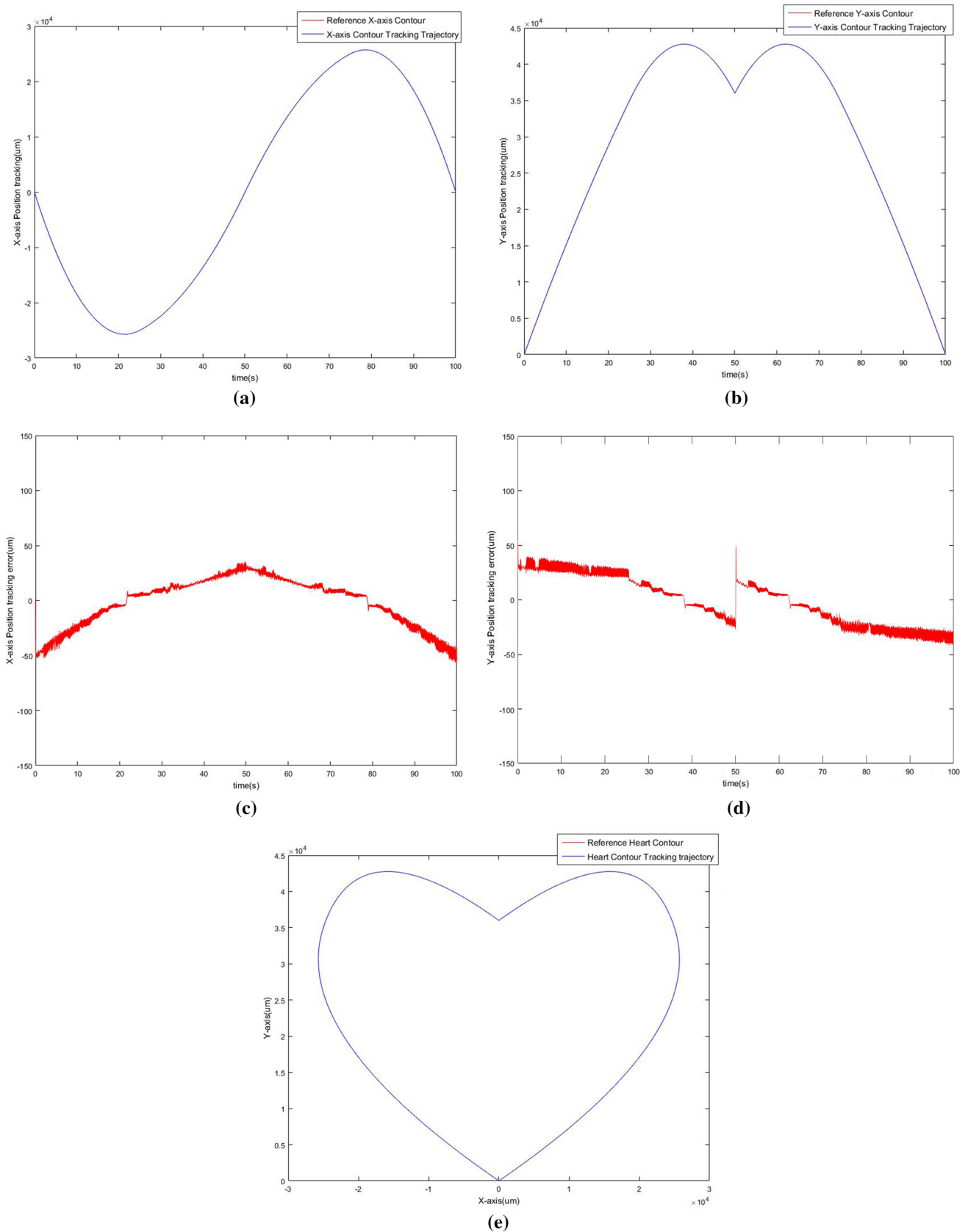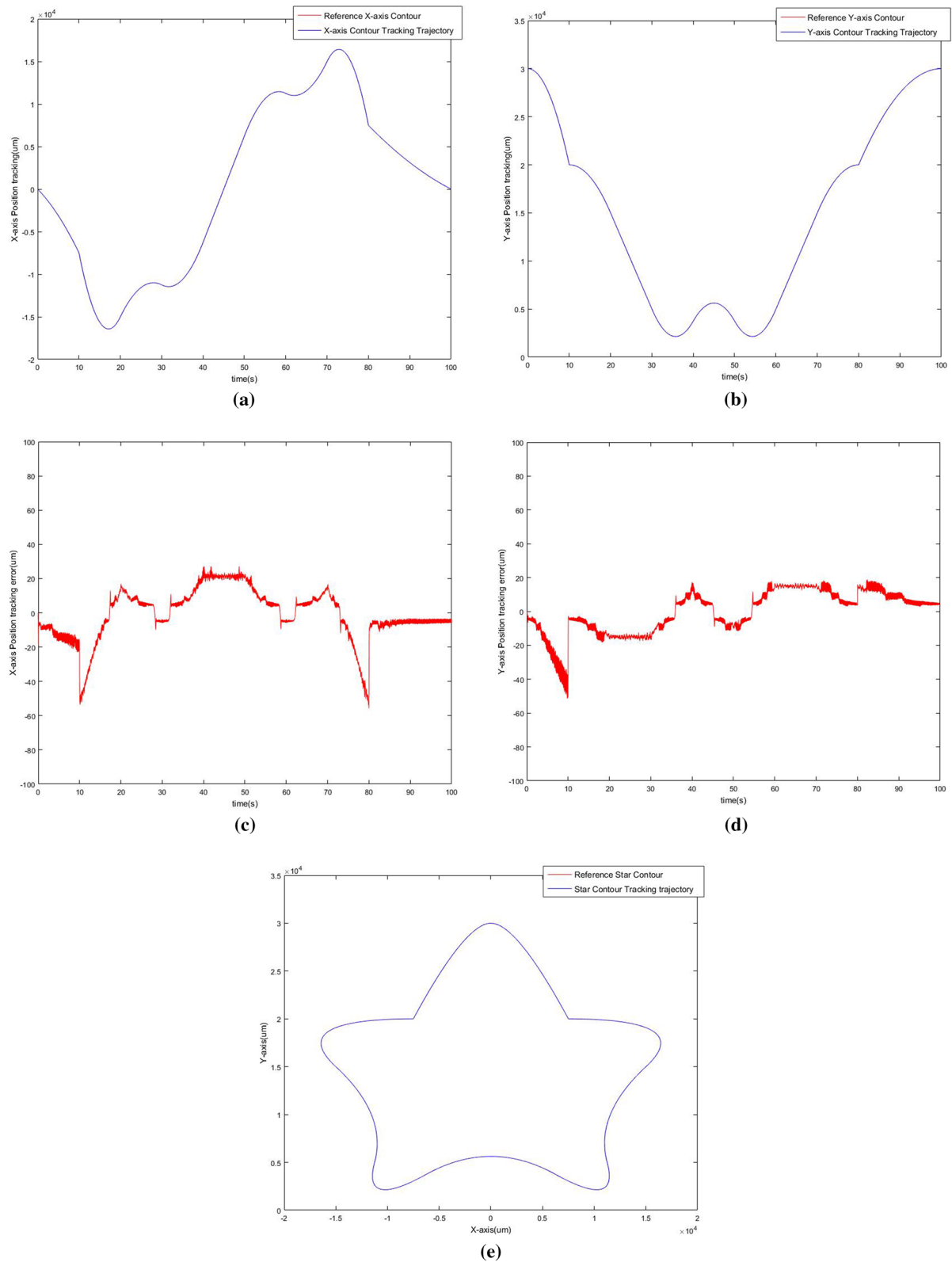
Fig. 10 Experimental results obtained using IT2FSMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** heart curve trajectory
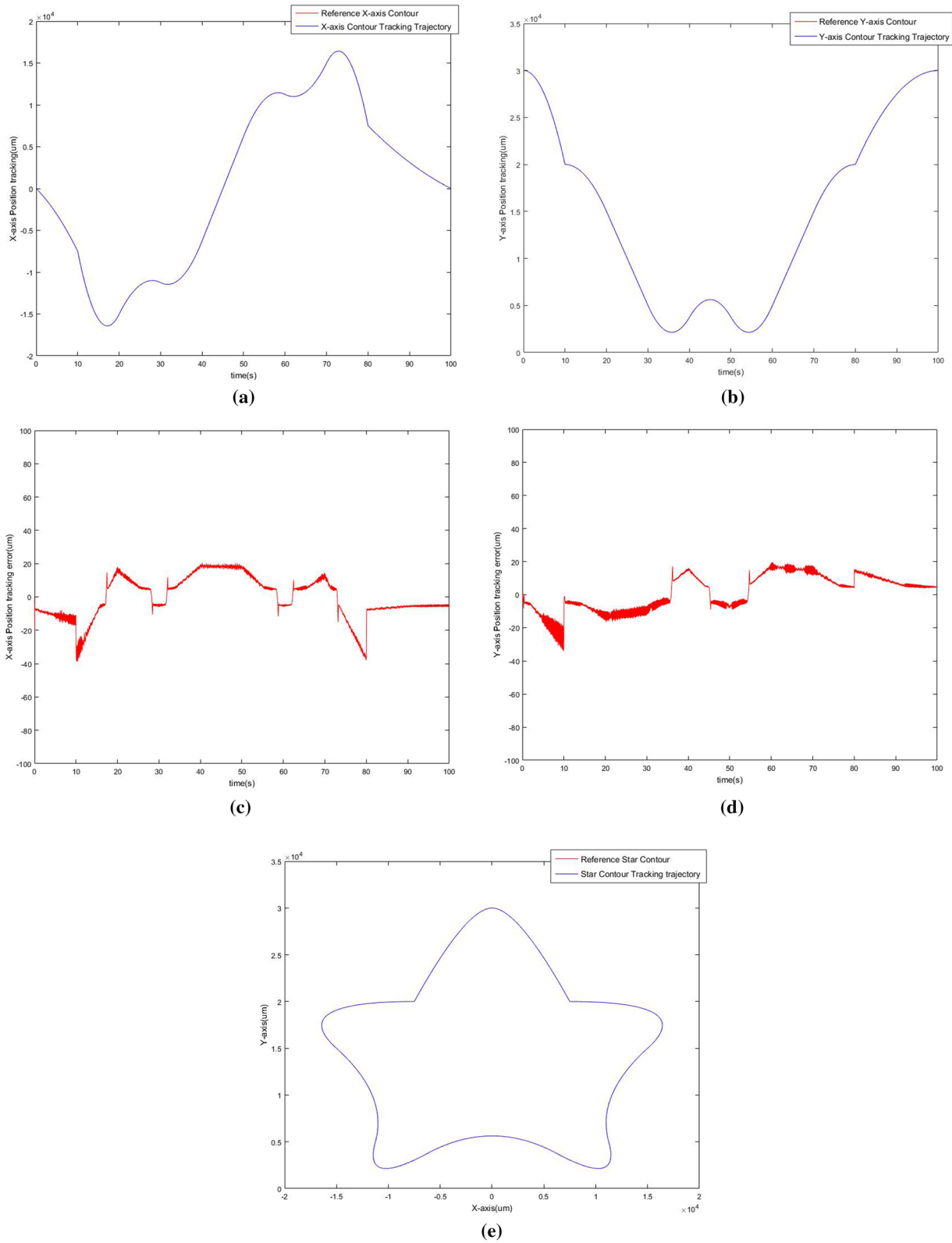
Fig. 11 Experimental results obtained using IT2FPISMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** star curve trajectory

Fig. 12 Experimental results obtained using IT2FSMC to control the two-axis stage. **a** X-axis direction response, **b** Y-axis direction response, **c** X-axis position tracking error, **d** Y-axis position tracking error, and **e** star curve trajectory

Compliance with Ethical Standards

References

1. Slotine, J. E., & Li, W. (1991). *Applied nonlinear control*. Englewood Cliffs, NJ: Prentice Hall.
2. Wu, J., Xiong, Z., & Ding, H. (2015). Integral design of contour error model and control for biaxial system. *International Journal of Machine Tools and Manufacture, 89,* 159–169.
3. Lin, F. J., Shieh, H. J., Shieh, P. H., & Shen, P. H. (2006). An adaptive recurrent-neural-network motion controller for X-Y Table in CNC machine. *IEEE Transaction on system Man and Cybernetics, 36*(2), 286–299.
4. Lin, F. J., Chou, P. H., & Kung, Y. S. (2008). Robust fuzzy neural network controller with nonlinear disturbance observer for two-axis motion control system. *IET Control Theory and Applications, 2*(2), 151–167.
5. El-Sousy, F. F. M. (2016). Intelligent mixed H2/H∞ adaptive tracking control system design using self-organizing recurrent fuzzy-wavelet-neural-network for uncertain two-axis motion control system. *Applied Soft Computing, 41,* 22–50.
6. Lin, F. J., & Shen, P. H. (2006). Robust fuzzy neural network sliding-mode control for two-axis motion control system. *IEEE Transactions on Industrial Electronics, 53*(4), 1209–1225.
7. Lin, F. J., Shieh, P. H., & Shen, P. H. (2006). Robust recurrent-neural-network sliding-mode control for the X-Y table of a CNC machine. *IEE Proceedings—Control Theory and Applications, 153*(1), 111–123.
8. Piegl, L., & Tiller, W. (1997). *The NURBS books* (2nd ed.). Berlin: Springer.
9. Lee, A. C., Lin, M. T., Pana, Y. R., & Lin, W. Y. (2011). The feedrate scheduling of NURBS interpolator for CNC machine tools. *Computer-Aided Design, 43,* 612–628.
10. Yau, H. T., Lin, M. T., & Tsai, M. S. (2006). Real-time NURBS interpolation using FPGA for high speed motion control. *Computer-Aided Design, 38,* 1123–1133.
11. Cheng, M. Y., Tsai, M. C., & Kuo, J. C. (2002). Real-time NURBS command generators for CNC servo controllers. *International Journal of Machine Tools and Manufacture, 42,* 801–881.
12. Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning-1. *Information Sciences, 8,* 199–249.
13. Mendel, J. M., John, R. I., & Feilong, L. (2006). Interval type-2 fuzzy logic systems made simple. *IEEE Transactions on Fuzzy Systems, 14*(6), 808–821.
14. Mendel, J. M. (2014). General type-2 fuzzy logic systems made simple: A tutorial. *IEEE Transactions on Fuzzy Systems, 22*(5), 1162–1182.
15. Liang, Q., & Mendel, J. M. (2000). Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems, 8*(5), 535–550.
16. Lin, P. Z., Lin, C. M., Hsu, C. F., & Lee, T. T. (2005). Type-2 fuzzy controller design using a sliding-mode approach for application to DC-DC converters. *IEE Proceedings—Electric Power Applications, 152*(6), 1482–1488.
17. Wu, D., & Tan, W. W. (2010). Interval type-2 fuzzy PI controllers: Why they are more robust. In *Proceedings of 2010 IEEE international conference on granular computing* (pp. 802–807).
18. Nie, M., & Tan, W. W. (2012). Modeling capability of type-1 fuzzy set and interval type-2 fuzzy set. In *Proceedings of 2012 IEEE international congress on fuzzy system* (pp. 1–8).
19. Navarro, G., Umberger, D. K., & Manic, M. (2017). VD-IT2, virtual disk cloning on disk arrays using a type-2 fuzzy controller. *IEEE Transactions on Fuzzy Systems, 25*(6), 1752–1764.
20. Utkin, V., Guldner, J., & Shi, J. (2009). *Sliding mode control in electro-mechanical systems* (2nd ed.). Boca Raton: CRC Press.
21. Ho, H. F., Wong, Y. K., & Rad, A. B. (2009). Adaptive fuzzy sliding mode control with chattering elimination for nonlinear SISO systems. *Simulation Modeling Practice and Theory, 17,* 1199–1210.
22. Lin, T. C. (2010). Based on interval type-2 fuzzy-neural network direct adaptive sliding mode control for SISI nonlinear systems. *Communications in Nonlinear Science and Numerical Simulation, 15,* 4084–4099.
23. Ghaemi, M., & Akbarzader-T, M. R. (2014). Indirect adaptive interval type-2 fuzzy PI sliding mode control for a class of uncertain nonlinear systems. *Iranian Journal of Fuzzy Systems, 11*(5), 1–21.
24. Ghaemi, M., Akbarzadeh M.-R. T., & Mohsen Jalaeian, F. (2012). Adaptive interval type-2 fuzzy PI sliding mode control with optimization of membership functions using genetic algorithm. In *2012 2nd international eConference on computer and knowledge engineering (ICCKE), October 18–19* (pp. 123–128).
25. Sun, N., Yang, T., Fang, Y., Wu, Y., & Chen, H. (2019). Transportation control of double-pendulum cranes with a nonlinear quasi-PID scheme: Design and experiments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49*(7), 1408–1418.
26. Yang, T., Sun, N., Chen, H., & Fang, Y. (2019). Neural network-based adaptive antiswing control of an underactuated ship-mounted crane with roll motions and input dead zones. *IEEE Transactions on Neural Networks and Learning Systems* (pp. 1–14) **(early access)**.

Wei-Lung Mao was born in Taiwan, R.O.C. in 1972. He received the B.S. degree in electrical engineering from National Taiwan University of Science and Technology in 1994, the M.S. and the Ph.D. degrees in electrical engineering from National Taiwan University in 1996 and 2004, respectively. He is now a Professor in the Department of Electrical Engineering and Graduate School of Engineering Science and Technology, National Yunlin University of Science and Technology. His research interests are satellite navigation systems, intelligent and adaptive control system, adaptive signal processing, neural networks and precision control.

Ding-Yu Shiu was born in Taiwan, R.O.C. in 1994. He received the B.S. degree in computer and intelligent robot from National Pingtung University in 2016, the M.S. degrees in electrical engineering from National Yunlin University of Science and Technology in 2018. His research interests are adaptive control system, fuzzy control system, fuzzy neural networks, and precision control.