Design of Adaptive Sliding-Mode Controller for Nonlinear System with Modeling Uncertainties

Zheng Wenda
Measurement & Control Technology Research Office
Hefei New Star Institute
HeFei, China
wdsubmit@qq.com

Wu Gang
Science Research Office
HeFei, China
ansarn@tom.com

Xu Gang
Measurement & Control Technology Research Office
Hefei New Star Institute
HeFei, China
1926591573@qq.com

Ding Junxiang
Measurement & Control Technology Research Office
Hefei New Star Institute
HeFei, China
djxgirl@mail.ustc.edu.cn

Abstract—Objective: The robot manipulators is a complicated uncertain nonlinear system, mainly due to the nonlinear and coupled character of system equations and the lack of a precise model for the dynamics and parameters, as well as the appearance of internal and external perturbations. A new control algorithm was proposed for the above-mentioned problem. Methods: According to the general model for the robotic system and the sliding-mode control theory, the corresponding control law was designed. For many reasons, there always exists discrepancy between nominal and actual mode. As the FBFN(Fuzzy Basis Function Networks) can approximate any function in the continuous course with arbitrary accuracy, it was applied to approximate the uncertainty of the system, and the parameters were updated by the adaptive law. Based on the theory of Lyapunov stability, the stability of the adaptive controller was given with a sufficient condition. Results: The tracking error was convergent to the switching surface in finite time. Conclusion: The simulation results demonstrate the good performance of the proposed controller.

Keywords- adaptive control; robotic dynamic systems; fuzzy basis function networks(FBFN); sliding-mode control; uncertain nonliear system

I. INTRODUCTION

The purpose of robot arm control is to maintain the dynamic response of the manipulator in accordance with some pre-specified performance [1, 2]. Although the control problem can be stated in such a simple manner, its solution is complicated for the robot's highly nonlinear dynamics. In general, the control problem consists of obtaining dynamic models of the robotic system and using these models to determine control laws or strategies to achieve the desired system response and performance. Although researchers have proposed many methods [1-3], such as the feedback linearization of nonlinear systems, which cancels the nonlinearities of robot manipulators and imposes a desired linear model so that linear control techniques can be applied. However, the method is based on the exact knowledge of robot dynamics. Actually, it is very difficult to obtain the exact knowledge and it is required to approximate an unmodled dynamics with a nonlinear component [2, 4, 5]. Neural networks and fuzzy systems provide good solutions to this challenging task. In this paper, we design an adaptive fuzzy controller for robot manipulators.

It has been proved that fuzzy basis function (FBF) networks can be universal approximators with arbitrarily small errors[2, 4-8]. Therefore, a fuzzy basis function network is used to approximate and cancel the unknown dynamics of robot manipulators. As in [9], the control structure and learning rules are derived from a Lyapunov theory extension that guarantee both tracking errors and parameter estimate errors in the closed-loop system are bounded. By taking the uncertainties including approximation errors and external disturbances into consideration, such a technique can reject the effects.

II. FUZZY BASIS FUNCTION NETWORKS AND ROBOTIC DYNAMIC SYSTEM

A. Fuzzy Basis Function Networks

Considering a fuzzy system with the following form of rules:

$$R^j: \text{IF } x_1 \text{ is } A_1^j \text{ and } x_2 \text{ is } A_2^j \text{ and } \mathrm{L} \text{ and } x_n \text{ is } A_n^j, \text{THEN } y \text{ is } B^j \quad (1)$$

where R^j is the j th rule,. j=1,2,3,L ,M .is the number of fuzzy rules,. $\mathbf{x}=(x_1,x_2,\mathrm{L},x_n)\in U\in R^n$ .denotes the output variable of fuzzy system and $i=1,2,3,\mathrm{L}$,n is the number of input variable, $y\in W\subset R$ represents the output variable, A_i^j and . B^j .are linguistic terms in the discourse U and W .

The fuzzy system can be described as follows with singleton fuzzification, product inference, and defining the defuzzifier as a weighted sum of each rule's output:

$$y(\mathbf{x})=\sum_{j=1}^{M}\xi_j(\mathbf{x})\theta_j=\boldsymbol{\xi}^{\mathrm{T}}(\mathbf{x})\boldsymbol{\theta}=\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\xi}(\mathbf{x}) \quad (2)$$

where $\boldsymbol{\theta}=(\theta_1,\theta_2,\mathrm{L},\theta_M)^{\mathrm{T}}$ is the tunable-parameter vector, θ_j is the maximum value according to the B^j, $\boldsymbol{\xi}=(\xi_1,\xi_2,\mathrm{L},\xi_M)^{\mathrm{T}}$ are the fuzzy basis functions (FBF).

The fuzzy basis function is defined as:

$$\xi_j(\mathbf{x}) = \frac{\prod_{i=1}^{n} \mu_{A_i^j}(x_i)}{\sum_{j=1}^{M}\prod_{i=1}^{n} \mu_{A_i^j}(x_i)} \quad (j = 1,2,\mathrm{L}\ ,M) \tag{3}$$

where $\mu_{A_i^j}(x_i)$ is the Gaussian membership function.

From (3), it is easy to find out that $\sum_{j=1}^{M}\xi_j(\mathbf{x}) = 1$ and $0 \le \xi_j(\mathbf{x}) \le 1$.

According to the document [4], the FBFN can approximate any function in the continuous course with arbitrary accuracy, i.e.:

$$\sup_{x\in U}\left| f(x) - g(x)\right| \le \varepsilon \tag{4}$$

B. Robotic Dynamic System

We will consider, in this section, the case of modeling robotic manipulators. The general model for this kind of robotic system is the following [2]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_{\mathrm{d}} = \boldsymbol{\tau} \tag{5}$$

where $\mathbf{q} \in \mathbf{R}^n$ denotes the link position, $\mathbf{M}(\mathbf{q}) \in \mathbf{R}^{n\times n}$ is the inner matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbf{R}^{n\times n}$ is the centripetal-Coriolis matrix, $\mathbf{G}(\mathbf{q}) \in \mathbf{R}^n$ represents the gravity vector, $\mathbf{F}(\dot{\mathbf{q}}) \in \mathbf{R}^n$ represents the friction force matrix, $\boldsymbol{\tau}_{\mathrm{d}} \in \mathbf{R}^n$ is the unmodeled disturbances vector. $\boldsymbol{\tau} \in \mathbf{R}^n$ is the vector of control input torques.

III. Controller Design

A. Nominated Plant Control Law

Considering the robotic dynamic system (5), if the mode is accurate and the disturbance is zero ($\boldsymbol{\tau}_{\mathrm{d}} = 0$), then we can design an ideal control law:

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}} - \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} - \mathbf{k}_{\mathrm{p}}\mathbf{e}) + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) \tag{6}$$

where $\mathbf{k}_{\mathrm{p}} = \begin{bmatrix} \alpha^2 & 0 \\ 0 & \alpha^2 \end{bmatrix}$, $\mathbf{k}_{\mathrm{d}} = \begin{bmatrix} 2\alpha & 0 \\ 0 & 2\alpha \end{bmatrix}$, $\alpha > 0$, $\mathbf{e} = \mathbf{q} - \mathbf{q}_{\mathrm{d}}$, $\dot{\mathbf{e}} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_{\mathrm{d}}$, $\ddot{\mathbf{e}} = \ddot{\mathbf{q}} - \ddot{\mathbf{q}}_{\mathrm{d}}$, $\mathbf{q}_{\mathrm{d}}$ is the desired robot manipulator trajectory vector.

If we apply (6) into the system (5), and we get a stable close-loop system:

$$\ddot{\mathbf{e}} + \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} + \mathbf{k}_{\mathrm{p}}\mathbf{e} = 0 \tag{7}$$

However, in the actual environment, there always exist many disturbances, and we can just get the nominated system not the real model.

The nominated mode can be represented as $\mathbf{M}_0(\mathbf{q})$, $\mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})$, $\mathbf{G}_0(\mathbf{q})$, $\mathbf{F}_0(\dot{\mathbf{q}})$, and the corresponding control law can be depicted as:

$$\boldsymbol{\tau} = \mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}} - \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} - \mathbf{k}_{\mathrm{p}}\mathbf{e}) + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}_0(\dot{\mathbf{q}}) \tag{8}$$

Substitute (8) into the mode (5), then we will get that:

$$\begin{aligned} &\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_{\mathrm{d}} \\ &= \mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}} - \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} - \mathbf{k}_{\mathrm{p}}\mathbf{e}) + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}_0(\dot{\mathbf{q}}) \end{aligned} \tag{9}$$

The above equality can be subtracted by $\mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}_0(\dot{\mathbf{q}})$, and define $\Delta\mathbf{M}(\mathbf{q}) = \mathbf{M}_0(\mathbf{q}) - \mathbf{M}(\mathbf{q})$, $\Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, $\Delta\mathbf{G}(\mathbf{q}) = \mathbf{G}_0(\mathbf{q}) - \mathbf{G}(\mathbf{q})$, $\Delta\mathbf{F}(\dot{\mathbf{q}}) = \mathbf{F}_0(\dot{\mathbf{q}}) - \mathbf{F}(\dot{\mathbf{q}})$, then:

$$\begin{aligned} &\ddot{\mathbf{e}} + \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} + \mathbf{k}_{\mathrm{p}}\mathbf{e} \\ &= \mathbf{M}_0^{-1}(\mathbf{q})\left(\Delta\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \Delta\mathbf{G}(\mathbf{q}) + \Delta\mathbf{F}(\dot{\mathbf{q}}) - \boldsymbol{\tau}_{\mathrm{d}}\right) \end{aligned} \tag{10}$$

From (10), it is clear that the system performance is falling off due to the modeling's inaccuracy. It's necessary to cancel the inaccuracy by other methods, and function approximation is a good one.

Define the inaccuracy as $\mathbf{f}(\mathbf{x})$:

$$\begin{aligned} &\mathbf{f}(\mathbf{x}) \\ &= \mathbf{M}_0^{-1}(\mathbf{q})\left(\Delta\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \Delta\mathbf{G}(\mathbf{q}) + \Delta\mathbf{F}(\dot{\mathbf{q}}) - \boldsymbol{\tau}_{\mathrm{d}}\right) \end{aligned} \tag{11}$$

If $\mathbf{f}(\mathbf{x})$ is known, the control law can be modified:

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}} - \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} - \mathbf{k}_{\mathrm{p}}\mathbf{e} - \mathbf{f}(\mathbf{x})) \\ &\quad + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}_0(\dot{\mathbf{q}}) \end{aligned} \tag{12}$$

Substitute (12) into the system (5), and we can get (7).

B. Approximation

Actually, $\mathbf{f}(\mathbf{x})$ is unknown and if we wanted to use the control law (12), we can approximate it, then the control law can be compensated [10].

It has been proved that fuzzy basis function network (FBFN) can be universal approximators with arbitrarily small errors. So, we use the FBFN to approximate $\mathbf{f}(\mathbf{x})$.

$$\mathbf{y} = \boldsymbol{\theta}^{\mathrm{T}}\varphi(\mathbf{x}) \tag{13}$$

The output vector of FBFN $\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*)$ estimates $\mathbf{f}(\mathbf{x})$ where the approximation error $\boldsymbol{\eta}$ ($\boldsymbol{\eta}$ has the superior bound $\boldsymbol{\eta}_0$).

$$\boldsymbol{\eta}_0 = \sup\left\|\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*) - \mathbf{f}(\mathbf{x})\right\| \tag{14}$$

and

$$\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*) = \boldsymbol{\theta}^{*\mathrm{T}}\varphi(\mathbf{x}) \tag{15}$$

where $\boldsymbol{\theta}^*$ represents the optimal FBFN weight value of approximation to $\mathbf{f}(\mathbf{x})$.

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\beta(M_{\boldsymbol{\theta}}),\mathbf{x}\in\varphi(M_{\mathbf{x}})} \sup\left\{\left\|\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*) - \mathbf{f}(\mathbf{x})\right\|\right\} \tag{16}$$

C. Controller Design

According to the above analysis, combing with the nominated plant (8), the control law (12) and the approximation (15), the controller can be modified as follow:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_1 + \boldsymbol{\tau}_2 \tag{17}$$

where:

$$\boldsymbol{\tau}_1 = \mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}} - \mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}} - \mathbf{k}_{\mathrm{p}}\mathbf{e}) + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}_0(\dot{\mathbf{q}}) \tag{18}$$

$$\boldsymbol{\tau}_2 = -\mathbf{M}_0(\mathbf{q})\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}) \tag{19}$$

where $\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}) = \hat{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})$ and $\hat{\boldsymbol{\theta}}$ are estimated values of the FBFN output and $\boldsymbol{\theta}^*$, respectively.

Apply (17) into the system (5), we can get that:

$$\begin{aligned}&\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}(\mathbf{q})+\mathbf{F}(\dot{\mathbf{q}})+\boldsymbol{\tau}_{\mathrm{d}}\\&=\mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{q}}_{\mathrm{d}}-\mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}}-\mathbf{k}_{\mathrm{p}}\mathbf{e})+\mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}_0(\mathbf{q})\\&\quad+\mathbf{F}_0(\dot{\mathbf{q}})-\mathbf{M}_0(\mathbf{q})\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta})\end{aligned}\tag{20}$$

The above equality can be subtracted by $\mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}}+\mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\mathbf{G}_0(\mathbf{q})+\mathbf{F}_0(\dot{\mathbf{q}})$:

$$\begin{aligned}&\Delta\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}+\Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\Delta\mathbf{G}(\mathbf{q})+\Delta\mathbf{F}(\dot{\mathbf{q}})-\boldsymbol{\tau}_{\mathrm{d}}\\&=\mathbf{M}_0(\mathbf{q})(\ddot{\mathbf{e}}+\mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}}+\mathbf{k}_{\mathrm{p}}\mathbf{e}+\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}))\end{aligned}\tag{21}$$

and

$$\begin{aligned}&\ddot{\mathbf{e}}+\mathbf{k}_{\mathrm{d}}\dot{\mathbf{e}}+\mathbf{k}_{\mathrm{p}}\mathbf{e}+\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta})=\mathbf{f}(\mathbf{x})\\&=\mathbf{M}_0^{-1}(\mathbf{q})\left(\Delta\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}+\Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}+\Delta\mathbf{G}(\mathbf{q})+\Delta\mathbf{F}(\dot{\mathbf{q}})-\boldsymbol{\tau}_{\mathrm{d}}\right)\end{aligned}\tag{22}$$

We define $\mathbf{x}=(\mathbf{e},\dot{\mathbf{e}})^{\mathrm{T}}$, and the above equation can be denoted as:

$$\dot{\mathbf{x}}=\mathbf{A}\mathbf{x}+\mathbf{B}\left\{\mathbf{f}(\mathbf{x})-\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta})\right\}\tag{23}$$

where $\mathbf{A}=\begin{pmatrix}0 & \mathbf{I}\\ -\mathbf{k}_{\mathrm{p}} & -\mathbf{k}_{\mathrm{d}}\end{pmatrix},\mathbf{B}=\begin{pmatrix}0\\ \mathbf{I}\end{pmatrix}$.

Define $\tilde{\boldsymbol{\theta}}=\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}^*$, then:

$$\begin{aligned}\mathbf{f}(\mathbf{x})-\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta})&=\mathbf{f}(\mathbf{x})-\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*)+\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta}^*)-\hat{\mathbf{f}}(\mathbf{x},\boldsymbol{\theta})\\&=\eta+\boldsymbol{\theta}^{*\mathrm{T}}\varphi(\mathbf{x})-\hat{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})=\eta-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})\end{aligned}\tag{24}$$

Equation (23) can be changed into:

$$\dot{\mathbf{x}}=\mathbf{A}\mathbf{x}+\mathbf{B}\left(\eta-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})\right)\tag{25}$$

Define the Lyapunov function as:

$$V=\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{x}+\frac{1}{2\gamma}\|\tilde{\boldsymbol{\theta}}\|^2\tag{26}$$

where $\gamma>0$, and the matrix $\mathbf{P}$ is a symmetry positive matrix which satisfies the following Lyapunov equation:

$$\mathbf{P}\mathbf{A}+\mathbf{A}^{\mathrm{T}}\mathbf{P}=-\mathbf{Q}\tag{27}$$

where $\mathbf{Q}>0$.

It can be easy to get that:

$$\|\tilde{\boldsymbol{\theta}}\|^2=tr(\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})\tag{28}$$

The time derivative of function (27) is:

$$\begin{aligned}\dot{V}&=\frac{1}{2}\left(\mathbf{x}^{\mathrm{T}}\mathbf{P}\dot{\mathbf{x}}+\dot{\mathbf{x}}^{\mathrm{T}}\mathbf{P}\mathbf{x}\right)+\frac{1}{\gamma}tr\left(\dot{\tilde{\boldsymbol{\theta}}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)\\&=\frac{1}{2}\left[\mathbf{x}^{\mathrm{T}}\mathbf{P}\left(\mathbf{A}\mathbf{x}+\mathbf{B}\left(\eta-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})\right)\right)+\left(\mathbf{x}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}+\left(\eta-\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})\right)^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\right)\mathbf{P}\mathbf{x}\right]\\&\quad+\frac{1}{\gamma}tr\left(\dot{\tilde{\boldsymbol{\theta}}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)\\&=\frac{1}{2}\left[\mathbf{x}^{\mathrm{T}}\left(\mathbf{P}\mathbf{A}+\mathbf{A}^{\mathrm{T}}\mathbf{P}\right)\mathbf{x}-\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})\right]\\&\quad+\frac{1}{2}\left[\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}\eta-\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}\right]+\frac{1}{\gamma}tr\left(\dot{\tilde{\boldsymbol{\theta}}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)\\&=-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}-\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+\frac{1}{\gamma}tr(\dot{\tilde{\boldsymbol{\theta}}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})\end{aligned}\tag{29}$$

where $\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\varphi(\mathbf{x})=\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}$ and $\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}\eta=\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}$.

Because of $\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}=\mathrm{tr}[\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}}]$, then (29) can be:

$$\dot{V}=-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}+\frac{1}{\gamma}tr(\dot{\tilde{\boldsymbol{\theta}}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}-\gamma\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}\varphi^{\mathrm{T}}(\mathbf{x})\tilde{\boldsymbol{\theta}})+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}\tag{30}$$

If we define the adaptive law as follow:

$$\dot{\hat{\boldsymbol{\theta}}}^{\mathrm{T}}=\gamma\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}\varphi^{\mathrm{T}}(\mathbf{x})+k_1\gamma\|\mathbf{x}\|\hat{\boldsymbol{\theta}}^{\mathrm{T}}\tag{31}$$

and

$$\dot{\hat{\boldsymbol{\theta}}}=\gamma\varphi(\mathbf{x})\mathbf{x}^{\mathrm{T}}\mathbf{P}\mathbf{B}+k_1\gamma\|\mathbf{x}\|\hat{\boldsymbol{\theta}}\tag{32}$$

where $k_1>0$, $\gamma>0$.

D. Stability Analysis

Applying (32) into (29), and:

$$\begin{aligned}\dot{V}&=-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+\frac{1}{\gamma}tr(k_1\gamma\|\mathbf{x}\|\hat{\boldsymbol{\theta}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})\\&=-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+k_1\|\mathbf{x}\|tr(\hat{\boldsymbol{\theta}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})\end{aligned}\tag{33}$$

According to the property of the norm of the matrix, there are:

$$tr[\hat{\boldsymbol{\theta}}^{\mathrm{T}}(\mathbf{x}-\hat{\boldsymbol{\theta}})]\le\|\hat{\boldsymbol{\theta}}\|_{\mathrm{F}}\|\mathbf{x}\|_{\mathrm{F}}-\|\hat{\boldsymbol{\theta}}\|_{\mathrm{F}}^2\tag{34}$$

and

$$tr(\hat{\boldsymbol{\theta}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})=tr(\tilde{\boldsymbol{\theta}}^{\mathrm{T}}\hat{\boldsymbol{\theta}})=tr[\tilde{\boldsymbol{\theta}}^{\mathrm{T}}(\boldsymbol{\theta}^*+\tilde{\boldsymbol{\theta}})]\le\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\|\boldsymbol{\theta}^*\|_{\mathrm{F}}-\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}^2\tag{35}$$

Substituting (35) into (33), and:

$$\begin{aligned}\dot{V}&\le-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+k_1\|\mathbf{x}\|tr(\hat{\boldsymbol{\theta}}^{\mathrm{T}}\tilde{\boldsymbol{\theta}})\\&\le-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x}+\eta^{\mathrm{T}}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{x}+k_1\|\mathbf{x}\|\left(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\|\boldsymbol{\theta}^*\|_{\mathrm{F}}-\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}^2\right)\\&\le-\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{x}\|^2+\|\eta_0\|\lambda_{\max}(\mathbf{P})\|\mathbf{x}\|+k_1\|\mathbf{x}\|\left(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\|\boldsymbol{\theta}^*\|_{\mathrm{F}}-\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}^2\right)\\&\le-\|\mathbf{x}\|\left(\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{x}\|-\|\eta_0\|\lambda_{\max}(\mathbf{P})-k_1\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\boldsymbol{\theta}_{\max}+k_1\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}^2\right)\\&=-\|\mathbf{x}\|\left(\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{x}\|-\|\eta_0\|\lambda_{\max}(\mathbf{P})+k_1(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}-\frac{1}{2}\boldsymbol{\theta}_{\max})^2-\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\right)\end{aligned}\tag{36}$$

where:

$$-k_1\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\boldsymbol{\theta}_{\max}+k_1\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}^2=k_1(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}-\frac{1}{2}\boldsymbol{\theta}_{\max})^2-\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\tag{37}$$

According to the Lyapunov theory, to assure the system is stable, $\dot{V}\le 0$, and:

$$\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{x}\|-\|\eta_0\|\lambda_{\max}(\mathbf{P})+k_1(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}-\frac{1}{2}\boldsymbol{\theta}_{\max})^2-\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\ge 0\tag{38}$$

It must meet the following conditions:

$$\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{x}\|\ge\|\eta_0\|\lambda_{\max}(\mathbf{P})+\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\tag{39}$$

or

$$k_1(\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}-\frac{1}{2}\boldsymbol{\theta}_{\max})^2\ge\|\eta_0\|\lambda_{\max}(\mathbf{P})+\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\tag{40}$$

We can get the convergence conditions:

$$\|\mathbf{x}\|\ge\frac{2}{\lambda_{\min}(\mathbf{Q})}\left(\|\eta_0\|\lambda_{\max}(\mathbf{P})+\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\right)\tag{41}$$

or

$$\|\tilde{\boldsymbol{\theta}}\|_{\mathrm{F}}\ge\sqrt{\frac{1}{k_1}\left(\|\eta_0\|\lambda_{\max}(\mathbf{P})+\frac{k_1}{4}\boldsymbol{\theta}_{\max}^2\right)}+\frac{1}{2}\boldsymbol{\theta}_{\max}\tag{42}$$

From the above analysis, if using the control law (17) and adapt law (32) and meeting the conditions (41) and (42), the system would be stable. Another advantage of this method is that the weight value of FBFN can be UUB.

From (41), when the eigenvalues of $\mathbf{Q}$ are bigger, the eigenvalues of $\mathbf{P}$ are smaller, η_0 is smaller, $\boldsymbol{\theta}_{\max}$ is smaller, and the convergence radius of $\mathbf{x}$ is smaller, the tracking result is more better.

IV. SIMULATION RESULT

The robotic manipulators' dynamics can be rewritten as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) + \boldsymbol{\tau}_{\mathrm{d}} = \boldsymbol{\tau} \qquad (43)$$

The parameters of a two-link robot are:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} p_1 + p_2 + 2p_3\cos q_2 & p_2 + p_3\cos q_2 \\ p_2 + p_3\cos q_2 & p_2 \end{bmatrix}$$

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} -p_3\dot{q}_2\sin q_2 & -p_3(\dot{q}_1+\dot{q}_2)\sin q_2 \\ p_3\dot{q}_1\sin q_2 & 0 \end{bmatrix}$$

$$\mathbf{G}(\mathbf{q}) = \begin{bmatrix} p_4 g\cos q_1 + p_5 g\cos(q_1+q_2) \\ p_5 g\cos(q_1+q_2) \end{bmatrix}$$

$$\mathbf{F}(\dot{\mathbf{q}}) = 0.02\,\mathrm{sgn}(\dot{\mathbf{q}})$$

$$\boldsymbol{\tau}_{\mathrm{d}} = d_1 + d_2\|\mathbf{e}\| + d_3\|\dot{\mathbf{e}}\|$$

The simulation values are as follows:

$$\mathbf{p} = [2.90 \quad 0.76 \quad 0.87 \quad 3.04 \quad 0.87]^{\mathrm{T}}.$$

$$\mathbf{d} = [0.1 \quad 0.2 \quad 0.6]^{\mathrm{T}}$$

$$g = 9.8$$

$$\Delta\mathbf{M}(\mathbf{q}) = 0.2 * \mathbf{M}_0(\mathbf{q})$$

$$\Delta\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = 0.2 * \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})$$

$$\Delta\mathbf{G}(\mathbf{q}) = 0.2 * \mathbf{G}_0(\mathbf{q})$$

$$\mathbf{q}_0 = [1.0 \quad 1.0]^{\mathrm{T}}$$

$$\dot{\mathbf{q}}_0 = [0.4 \quad 0.4]^{\mathrm{T}}$$

$$\mathbf{q}_{\mathrm{d}} = [1 + 0.2\sin(\pi t) \quad 1 + 0.2 * \cos(\pi t)]^{\mathrm{T}}$$

$$\alpha = 5$$

$$\mathbf{k}_{\mathrm{p}} = \begin{bmatrix} \alpha^2 & 0 \\ 0 & \alpha^2 \end{bmatrix} = \begin{bmatrix} 25 & 0 \\ 0 & 25 \end{bmatrix}$$

$$\mathbf{k}_{\mathrm{d}} = \begin{bmatrix} 2\alpha & 0 \\ 0 & 2\alpha \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$$

$$\mathbf{A} = \begin{pmatrix} 0 & \mathbf{I} \\ -\mathbf{k}_{\mathrm{p}} & -\mathbf{k}_{\mathrm{d}} \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 \\ \mathbf{I} \end{pmatrix}$$

$$\mathbf{P} = \mathrm{diag}(100,100,100,100)$$

$$k_1 = 0.001$$

$$\gamma = 25$$

For our FBFN, there are 10 rules in the rule base and the parameters of the FBFN are tuning by (32). Each rule has four inputs:

$$(\mathbf{x} = (\mathbf{e},\dot{\mathbf{e}})^{\mathrm{T}} = (\mathbf{q}_1 - \mathbf{q}_{1\mathrm{d}}, \mathbf{q}_2 - \mathbf{q}_{2\mathrm{d}}, \dot{\mathbf{q}}_1 - \dot{\mathbf{q}}_{1\mathrm{d}}, \dot{\mathbf{q}}_2 - \dot{\mathbf{q}}_{2\mathrm{d}})^{\mathrm{T}})$$

The initial values of membership functions are small random numbers, i.e., the centers and variance of Gaussian membership functions are random. From the result showed in the Fig .1, we can find that the proposed method is effective and achieves good results.

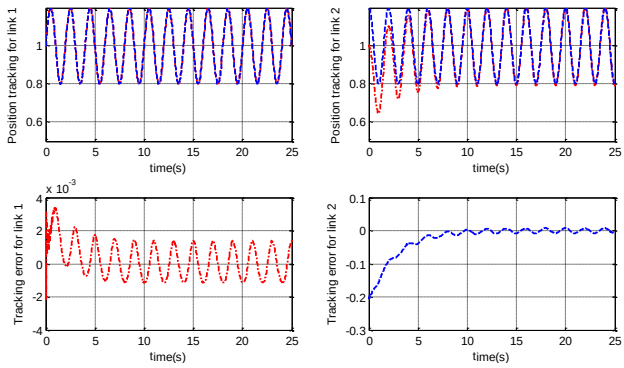

Figure 1. Simulation result figure

V. CONCLUSION

This paper presents a novel adaptive fuzzy sliding mode controller, successfully employed to control the robotic manipulator. The main advantage of this methodology is that it relaxes the required knowledge of system model. The experimental results demonstrate the good performance of the proposed controller, within the constraints of the sensorial system and the uncertainty of the theoretical models.

REFERENCES

[1] K. Lochan, and B. Roy, "Control of two-link 2-DOF robot manipulator using fuzzy logic techniques: a review." pp. 499-511.

[2] F. Piltan, A. Nabaee, M. Ebrahimi, and M. Bazregar, "Design robust fuzzy sliding mode control technique for robot manipulator systems with modeling uncertainties," *International Journal of Information Technology and Computer Science* vol. 5, no. 8, pp. 123, 2013.

[3] R.-J. Lian, "Adaptive self-organizing fuzzy sliding-mode radial basis-function neural-network controller for robotic systems," *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 3, pp. 1493-1503, 2014.

[4] J. Jantzen, *Foundations of fuzzy control: a practical approach*: John Wiley & Sons, 2013.

[5] F. Piltan, A. Hosainpour, S. Emamzadeh, I. Nazari, and M. Mirzaie, "Design sliding mode controller of with parallel fuzzy inference system compensator to control of robot manipulator," *IAES International Journal of Robotics and Automation* vol. 2, no. 4, pp. 149-162, 2013.

[6] Y. Li, S. Tong, Y. Liu, and T. Li, "Adaptive fuzzy robust output feedback control of nonlinear systems with unknown dead zones based on a small-gain approach," *Fuzzy Systems, IEEE Transactions on*, vol. 22, no. 1, pp. 164-176, 2014.

[7] Y. Li, S. Tong, T. Li, and X. Jing, "Adaptive fuzzy control of uncertain stochastic nonlinear systems with unknown dead zone using small-gain approach," *Fuzzy Sets and Systems*, vol. 235, pp. 1-24, 2014.

[8] J. Wang, A. B. Rad, and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy sets and Systems*, vol. 122, no. 1, pp. 21-30, 2001.

[9] Y.-K. Yang, T.-Y. Sun, C.-L. Huo, Y.-H. Yu, C.-C. Liu, and C.-H. Tsai, "A novel self-constructing radial basis function neural-fuzzy system," *Applied Soft Computing*, vol. 13, no. 5, pp. 2390-2404, 2013.

[10] A. Jalali, F. Piltan, A. Gavahian, and M. Jalali, "Model-free adaptive fuzzy sliding mode controller optimized by particle swarm for robot manipulator," *International Journal of Information Engineering and Electronic Business*, vol. 5, no. 1, pp. 68, 2013.