INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, e-ISSN 1841-9844, 14(5), 692-710, October 2019.

A Robust Adaptive Control using Fuzzy Neural Network for Robot Manipulators with Dead-Zone

D.H. Vu, S. Huang, T.D. Tran, T.Y. Vu, V.C. Pham

D. Ha Vu*
College of Electrical and Information Engineering
Hunan University, Changsha, China
*Corresponding author: vuhadhsd@hnu.edu.cn

Shoudao Huang
College of Electrical and Information Engineering
Hunan University, Changsha, China
hsd1962@hnu.edu.cn

T. Diep Tran
College of Electrical and Information Engineering
Hunan University, Changsha, China
phuongdiep222@hnu.edu.cn

T. Yen Vu
Faculty of Electrical Engineering
Saodo University, Chilinh, Vietnam
havi2203@hnu.edu.cn

V. Cuong Pham
Faculty of Electrical Engineering
Hanoi University of Industry, Hanoi, Vietnam
cuongpv0610@haui.edu.vn

Abstract: Abstract: In this paper, a robust-adaptive-fuzzy-neural-network controller (RAFNNs) bases on dead zone compensator for industrial robot manipulators (RM) is proposed to dead the unknown model and external disturbance. Here, the unknown dynamics of the robot system is deal by using fuzzy neural network to approximate the unknown dynamics. The online training laws and estimation of the dead-zone are determined by Lyapunov stability theory and the approximation theory. In this proposal, the robust sliding-mode-control (SMC) is constructed to optimize parameter vectors, solve the approximation error and higher order terms. Therefore, the stability, robustness, and desired tracking performance of RAFNNs for RM are guaranteed. The simulations and experiments performed on three-link RM are provided in comparison with neural-network (NNs) and proportional-integral-derivative (PID) to demonstrate the robustness and effectiveness of the RAFNNs.

Keywords: adaptive control, fuzzy neural networks, robot manipulators, unknown dead-zone.

1 Introduction

In fact, Robot manipulators are multi-input multi-output (MIMO) non-linear systems. In working process, robot system always bear the external disturbance, nonlinear fiction, pay load, etc. to overcome this face. So many controller propose such as adaptive controller, robust adaptive controller, backstepping controller, and intelligent controller, etc [2,8,9,17,26,29]. Backstepping technique has been widely applied to design adaptive controller for nonlinear system [13]. Investigations base on Backstepping control method were provided a systematic framework for the design of tracking and regulation strategies, suitable for a large class of state feedback linearizable nonlinear systems [4,10,25,28]. However, there are some problems in the Backstepping design method. A major constraint is that certain functions must be ”linear in the unknown parameters", which may not be satisfied in practice. Furthermore, some tedious analysis is needed to determine "regression matrices", and the problem of determining and computing the regression matrices become even more acute. To deal these problems, intelligent controllers based on fuzzy control for RM have been proposed. The Fuzzy logic technique is a successful implementation for the approximation of non-linear systems [1,5,11,12,15,16,22,27]. In [11], the intelligent controller based on fuzzy logic was proposed for robotic manipulators under uncertain environments. In this controller, the approximation capability of fuzzy logic was used to approximate the unknown dynamic of robot system. The parameters of the intelligent controller were adjusted online bases on the Lyapunov algorithm. In [15], the authors suggested an adaptive fuzzy sliding mode control with nonlinear observer (AFSMCO) for the robot manipulators with unknown external force. Here, by combining the advantages of fuzzy logic, sliding mode control and nonlinear observer, the performance of control system was improved. However, most proposed adaptive fuzzy controllers were difficult in building suitable fuzzy control rules, membership function, and how to guarantee the system stability was a challenge problem to be solved. To deal this difficult problem, the adaptive fuzzy neural networks were proposed [3, 18, 23, 24]. In [23], Rong-Jong Wai, and Rajkumar Muthusamy suggested an intelligent controller bases on fuzzy neural network control for robot manipulator in order to improve the control performance of position tracking. In this controller, sliding mode control was developed for position tracking of robot manipulator system. Moreover, the fuzzy neural network was employed to approximation the unknown dynamic of control system. The fuzzy neural network was designed bases on the SMC rules. The adaptive laws were determined by using the Lyapunov theorem. In [24], a fuzzy neural network combined with backstepping control was proposed for robot manipulator to achieve the robustness and stability. Here, the fuzzy neural network was inherited backstepping control to improve the robustness of backstepping control. The parameters of fuzzy neural network control were determined and adjusted online by the Lyapunov theorem. The robustness and stability of proposed controller were improved.

Recently years, one of the important subjects for robotic manipulator that has attracted many researchers is the compensation of dead-zone. In fact, deal-zone is a natural and nonlinear item. To deal with compensation of non-smooth nonlinearities, many researches were proposed [6, 7, 14, 19–21]. In [21], an adaptive neural network is proposed to compensate the dead-zone of the hydraulic system. In this proposed controller, the RBF neural network is applied to identify the dead-zone parameters and a cost function is proposed to provide the best approximation of dead-zone. The parameters of the control system and the dead-zone are easier to calculate.

In this paper, to deal with the problem of compensation dead-zone with the unknown dynamic and external disturbance, an adaptive robust fuzzy neural network control based on backstepping technique has been proposed. This proposed proposal is combined the advantage of FNN, sliding mode control, adaptive control and backstepping technique. The unknown robot dynamics are approximated by the FNN and the tracking errors are compensated by using the robust term. In addition, all the parameters of the proposed controller are adjusted by the stability Lyapunov theory. Thus, the robustness and effectiveness of RAFNNs control system are guarantee.

The paper is organized as follows. The preliminaries are described in section 2. Section 3 presented control design and stability analysis. The simulation and experimental results of three-link robot manipulators are provided in section 4. Finally, section 5 gives conclusion.

2 Preliminaries

2.1 Model of Robotic Manipulators

Consider the dynamics of an n-link robot manipulator with external disturbance:

$$M_{RM}\left(\Theta\right)\ddot{\Theta}+C_{RM}\left(\Theta,\dot{\Theta}\right)\dot{\Theta}+G_{RM}\left(\Theta\right)+F_{RM}\left(\dot{\Theta}\right)=\tau-\tau_0 \tag{1}$$

With $\Theta=[\Theta_1\ \Theta_2\ldots\Theta_n]\in R^{n\times1}$ is the joint position vector, $\dot{\Theta}=\left[\dot{\Theta}_1\ \dot{\Theta}_2\ldots\dot{\Theta}_n\right]\in R^{n\times1}$ is the velocity vector and $\ddot{\Theta}=\left[\ddot{\Theta}_1\ \ddot{\Theta}_2\ldots\ddot{\Theta}_n\right]\in R^{n\times1}$ is the acceleration vector. $M_{RM}\left(\Theta\right)$, $C_{RM}\left(\Theta,\dot{\Theta}\right)$ and $G_{RM}\left(\Theta\right)$ are $[n\times n]$ expressing the symmetric inertial matrix, Coriolis and Centripetal terms, and Gravity terms, respectively. $F_{RM}\left(\dot{\Theta}\right)$ represents the $n\times1$ vector of the frictions. τ_0 represents the $n\times1$ vector of the input unknown disturbances. And τ is the $n\times1$ control input vector of joints torque. For designing controller, several properties of the robot dynamics (1) have been assumed as follows.

Property 1: M_{RM} is the symmetric inertial Matrix and bounded as:

$$\vartheta_1\|x\|^2\le x^TM_{RM}x\le\vartheta_2\|x\|^2,\forall x\in R^n \tag{2}$$

With ϑ_1 and ϑ_2 are known positive constants.

Property 2: $\dot{M}_{RM}-2C_{RM}$ is skew symmetry matrix, in which

$$x^T\left[\dot{M}_{RM}-2C_{RM}\right]x=0 \tag{3}$$

Property 3: C_{RM}, G_{RM} and F_{RM} are satisfied:

$$\|C_{RM}\dot{\Theta}\|\le C_{kRM}\|\dot{\Theta}\|^2,\|G_{RM}\|\le G_{kRM},F_{RM}\le F_{kRM}\|\dot{\Theta}\|+F_0 \tag{4}$$

With $C_{kRM},G_{kRM},F_{kRM},F_0$ are positive constants.

Property 4: $\tau_0\in R^n$ is the unknown disturbance and τ_0 is bounded as follows:

$$\|\tau_0\|\le\tau_k,\tau_k>0 \tag{5}$$

According to assumptions given in [19], the dead zone function shows Fig. 1, and which is expressed as follows:

$$\tau=D(u),\quad D(u)=\begin{cases}h_r(u-d_r) & \text{for}\quad u>d_r\\ 0 & \text{for}\quad d_l\le u\le d_r\\ h_l(u+d_l) & \text{for}\quad u<d_l\end{cases} \tag{6}$$

Here, $d_r>0,\ d_l<0$ are unknown constant parameters of dead zone. $h_l\left(u\right),\ h_r\left(u\right)$ are the unknown smooth functions.

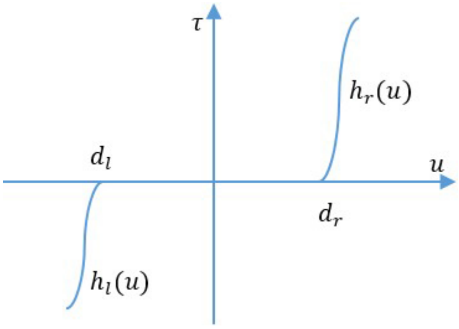

Figure 1: Dead zone model.

Where u is control input before entering the dead zone. τ is control input after entering the dead zone. Therefore (6) can be rewritten as:

$$\tau = D\left(u\right) = u - sat_D\left(u\right) \tag{7}$$

where the asymmetric saturation function is defined as:

$$sat_D(u) = \begin{cases} d_r & \text{for} \quad u > d_r \\ u & \text{for} \quad d_l \leq u \leq d_r \\ d_l & \text{for} \quad u < d_l \end{cases} \tag{8}$$

2.2 Backstepping controller

The conventional Backstepping controller for the dynamic of the RM is described as follows: **Step 1**: the tracking error vector $Z_{\Theta 1}\left(t\right)$ and derivative of $Z_{\Theta 1}\left(t\right)$ are define as the follows:

$$Z_{\Theta 1}\left(t\right) = \Theta_d - \Theta \quad \text{and} \quad \dot{Z}_{\Theta 1}\left(t\right) = \dot{\Theta}_d - \dot{\Theta} \tag{9}$$

By using $\dot{\Theta}$ as the first virtual control input. Define an intermediate function as:

$$\alpha_{\Theta 1}\left(t\right) = \dot{\Theta}_d + \lambda_{\Theta 1} Z_{\Theta 1} \quad \text{and} \quad \dot{\alpha}_{\Theta 1}\left(t\right) = \ddot{\Theta}_d + \lambda_{\Theta 1}\dot{Z}_{\Theta 1}, \lambda_{\Theta 1} > 0 \tag{10}$$

Consider the first following Lyapunov function candidate $L_{\Theta 1}$ as:

$$L_{\Theta 1}(Z_{\Theta 1}(t)) = \frac{1}{2} Z_{\Theta 1}^T Z_{\Theta 1} \tag{11}$$

The tracking error vector $Z_{\Theta 2}\left(t\right)$ is define as the follows:

$$Z_{\Theta 2}\left(t\right) = \alpha_{\Theta 1}\left(t\right) - \dot{\Theta} = \dot{Z}_{\Theta 1} + \lambda_{\Theta 1} Z_{\Theta 1} \tag{12}$$

The derivative of $L_{\Theta 1}(Z_{\Theta 1}(t))$ is:

$$\dot{L}_{\Theta 1}(Z_{\Theta 1}(t)) = Z_{\Theta 1}^T \dot{Z}_{\Theta 1} = Z_{\Theta 1}^T (Z_{\Theta 2}(t) - \lambda_{\Theta 1} Z_{\Theta 1}) \tag{13}$$

Step 2: the derivative of $Z_{\Theta 2}\left(t\right)$ along to time, we have

$$\dot{Z}_{\Theta 2}\left(t\right) = \dot{\alpha}_{\Theta 1}\left(t\right) - \ddot{\Theta} \tag{14}$$

Where $\ddot{\Theta}$ used as the second virtual control input. Substituting (9, 10, 12, 14) into (1), we have:

$$M_{RM}\dot{Z}_{\Theta 2} = M_{RM}\dot{\alpha}_{\Theta 1} + C_{RM}\alpha_{\Theta 1} - C_{RM}Z_{\Theta 2} + G_{RM} + F_{RM} + \tau_0 - \tau \tag{15}$$

Consider the second Lyapunov function $L_{\Theta 2}$ as follows:

$$L_{\Theta 2}(Z_{\Theta 1}(t), Z_{\Theta 2}(t)) = L_{\Theta 1}(Z_{\Theta 1}(t)) + \frac{1}{2} Z_{\Theta 2}^T M_{RM} Z_{\Theta 2} \tag{16}$$

The derivative of $L_{\Theta 2} = (Z_{\Theta 1}(t), Z_{\Theta 2}(t))$ is:

$$\dot{L}_{\Theta 2} = Z_{\Theta 1}^T (Z_{\Theta 2}(t) - \lambda_{\Theta 1} Z_{\Theta 1}) + \frac{1}{2} Z_{\Theta 2}^T \dot{M}_{RM} Z_{\Theta 2} + Z_{\Theta 2}^T M_{RM} \dot{Z}_{\Theta 2} \tag{17}$$

Substituting (15) into (17) and use Property 2, we have:

$$\begin{aligned} \dot{L}_{\Theta 2} &= Z_{\Theta 1}^T (Z_{\Theta 2}(t) - \lambda_{\Theta 1} Z_{\Theta 1}) + \frac{1}{2} Z_{\Theta 2}^T \dot{M}_{RM} Z_{\Theta 2} \\ &\quad + Z_{\Theta 2}^T (M_{RM} \dot{\alpha}_{\Theta 1} + C_{RM} \alpha_{\Theta 1} - C_{RM} Z_{\Theta 2} + G_{RM} + F_{RM} + \tau_0 - \tau) \\ &= Z_{\Theta 1}^T Z_{\Theta 2}(t) - Z_{\Theta 1}^T \lambda_{\Theta 1} Z_{\Theta 1} + \frac{1}{2} Z_{\Theta 2}^T \left(\dot{M}_{RM} - 2C_{RM} \right) Z_{\Theta 2} + Z_{\Theta 2}^T (y + \tau_0 - \tau) \\ &= Z_{\Theta 1}^T Z_{\Theta 2}(t) - Z_{\Theta 1}^T \lambda_{\Theta 1} Z_{\Theta 1} + Z_{\Theta 2}^T (y + \tau_0 - \tau) \end{aligned} \tag{18}$$

With

$$y = M_{RM} \dot{\alpha}_{\Theta 1} + C_{RM} \alpha_{\Theta 1} + G_{RM} + F_{RM} \tag{19}$$

To continue our design, the adaptive control law is presented as:

$$\tau = y + \lambda_{\Theta 2} Z_{\Theta 2} + Z_{\Theta 1} + \tau_0, \ \lambda_{\Theta 2} > 0 \tag{20}$$

Substituting (20) into (18), we have:

$$\dot{L}_{\Theta 2} = -Z_{\Theta 1}^T \lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^T \lambda_{\Theta 2} Z_{\Theta 2} \le 0 \tag{21}$$

Since (21), $\dot{L}_{\Theta 2} < 0$, so $\dot{L}_{\Theta 2}(Z_{\Theta 1}(t), Z_{\Theta 2}(t)) < \dot{L}_{\Theta 2}(Z_{\Theta 1}(0), Z_{\Theta 2}(0))$. If $Z_{\Theta 1}(t), Z_{\Theta 2}(t)$ are bounded with $t > 0$. By defining $\Omega(t) = Z_{\Theta 1}^T \lambda_{\Theta 1} Z_{\Theta 1} + Z_{\Theta 2}^T \lambda_{\Theta 2} Z_{\Theta 2}$ so $\Omega(t) \le \dot{L}_{\Theta 2}(Z_{\Theta 1}(t), Z_{\Theta 2}(t))$ and integrate the $\Omega(t)$ with respect to time as follows:

$$\int_0^t \Omega(\xi)\, d\xi \le L_{\Theta 2}(Z_{\Theta 1}(t), Z_{\Theta 2}(t)) - L_{\Theta 2}(Z_{\Theta 1}(0), Z_{\Theta 2}(0)) \tag{22}$$

Because $L_{\Theta 2}(Z_{\Theta 1}(0), Z_{\Theta 2}(0))$ is a bounded function and $L_{\Theta 2}(Z_{\Theta 1}(t), Z_{\Theta 2}(t))$ is nonincreasing and bounded, we have:

$$\lim_{t \to \infty} \int_0^t \Omega(\xi)\, d\xi < \infty \tag{23}$$

According to Barbalat's Lemma [20], when $\dot{\Omega}(t)$ is bounded function. It can be shown that $\lim\limits_{t \to \infty} \int\limits_0^t \Omega(t)\, dt = 0$. From this result, we see that, $Z_{\mathfrak{L}1}(t), Z_{\mathfrak{L}2}(t)$ will converge to zero when $t \to \infty$ and the global stability of the control system for RM is guaranteed.

2.3 Structure of adaptive Fuzzy Neural Networks

A fuzzy logic system includes four parts: the knowledge base, the fuzzifier, the fuzzy inference engine working on fuzzy rules, and the defuzzifier. The knowledge base of the fuzzy logic system is a collection of fuzzy IF-THEN rules of the following form:

R^1: IF s_1 is R_1^l and s_2 is R_2^l and ... and s_2 is R_n^l, THEN y is G^l, $l = 1, 2, ..., N$

Where $s = (s_1, ..., s_n)^T$ and y are the fuzzy logic system input and output, respectively. F_i^l, G^l are associated with the fuzzy membership functions $\mu_{F_i^l}(s_i)$ and $\mu_{G^l}(y)$, respectively. N is the number of rules.

The output of the fuzzy system can be expressed as:

$$y(s) = \frac{\sum_{l=1}^{N} \delta_l \prod_{i=1}^{n} \mu_{F_i^l}(s_i)}{\sum_{l=1}^{N} \left[\prod_{i=1}^{n} \mu_{F_i^l}(s_i)\right]}$$

Where $\delta_l = \max\limits_{y \in R} \mu_{G^l}(y)$ and $\delta = [\delta_1, \delta_2, ..., \delta_N]^T$.

The FNNs structure includes four - layer as shown in Fig. 2, which comprises the input, membership, rule and output layers.

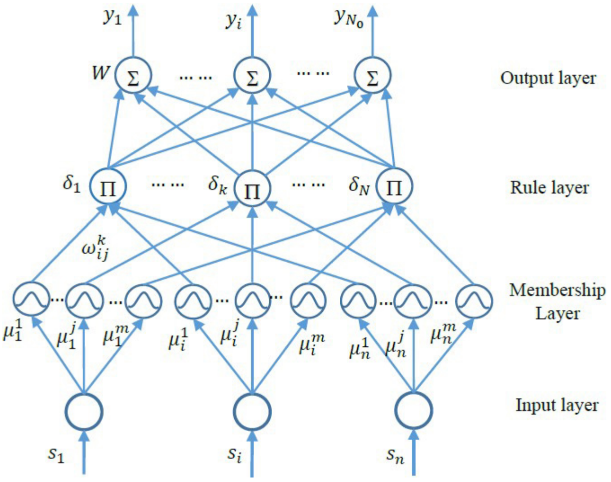

Figure 2: The FNNs structure.

Layer 1 (Input layer): Each node in this layer corresponds to one input linguistic variables $s_i\,(i = 1, 2, \ldots, n)$, and only transmits directly input values to the next layer.

Layer 2 (Membership layer): In this layer, each node represents the input values with the following Gaussian membership functions:

$$\mu_i^j\,(s_i) = exp\left[-\left(s_i - m_i^j\right)^2 / \left(b_i^j\right)^2\right] \tag{24}$$

Where m_i^j and b_i^j $(i = 1, 2, \ldots, n; j = 1, 2, \ldots, m)$ are, respectively, the center and standard deviation of the Gaussian membership function of the i^{th} input variable x_i to the node of this layer, and m denotes the total number of membership functions.

Layer 3 (Rule layer): Each node in this layer, which is described as a fuzzy rule, multiplies the inputs signal and the outputs result of the product. The output value of this layer is calculated:

$$\delta_k = \prod_{i=1}^{n} \omega_{ij}^k \mu_i^j(s_i), k = 1, 2, \ldots, N \tag{25}$$

Where δ_k is the k^{th} output of the rule layer, ω_{ji}^k is the weight between the membership layer and the rule layer, and N is the total number of rules.

Layer 4 (Output layer): In this layer, each node represents the output linguistic variables, and acts as a defuzzifier. The output can be represented as follows:

$$y_f = \sum_{1}^{N} w_k^f \delta_k \tag{26}$$

Moreover, (26) can be rewritten:

$$y = [y_1 y_2 \dots y_{N_0}]^T = W\delta = y_{FNNs}(s, W, m, b) \tag{27}$$

With

$$W = [w_1 w_2 \dots w_{N_0}]^T \tag{28}$$

$$\delta = [\delta_1 \delta_2 \dots \delta_N]^T \tag{29}$$

Where $w = \left[w_1^i \; w_2^i \dots w_N^i\right]^T$.

Next, we employ this FNNs as an approximator in our control design. Base on approximation error analysis, there exists an optimal FNNs with its optimal parameters:

$$y(s(t)) = {W^*}^T \delta^* (s(t), m^*, b^*) + \Delta(s(t)) \tag{30}$$

Where W^*, m^*, b^* are the optimal parameters of W, m, b, respectively, $\Delta(s(t))$ is the approximation error vector.

Assumption 1: the bound of optimal FNNs parameters:

$$\|W^*\| \leq \mathfrak{L}_w, \|m^*\| \leq \mathfrak{L}_m, \|b^*\| \leq \mathfrak{L}_b \tag{31}$$

Where $\mathfrak{L}_w, \mathfrak{L}_m, \mathfrak{L}_b$ are the positive real values.

Assumption 2: Error of approximation process is bounded:

$$\|\Delta^*\| \leq \mathfrak{L}_\Delta \tag{32}$$

Where $\mathfrak{L}_\Delta$ is the positive real value. The output of the FNNs is the approximate value and is represented as the following:

$$\hat{y} = \hat{W}^T \hat{\delta}\left(s(t), \hat{m}, \hat{b}\right) \tag{33}$$

Where $\hat{y}, \hat{W}, \hat{m}, \hat{b}$ are the approximate values of y, W^*, m^*, b^*, respectively.

3 Control design and stability analysis

3.1 Control design

We recommend the RAFNNs to find an adaptive law of the suitable adaptive RAFNNs model that makes control system able to achieve the required approximation errors accuracy.

Architecture of the dead zone compensator is shown in Fig. 3.

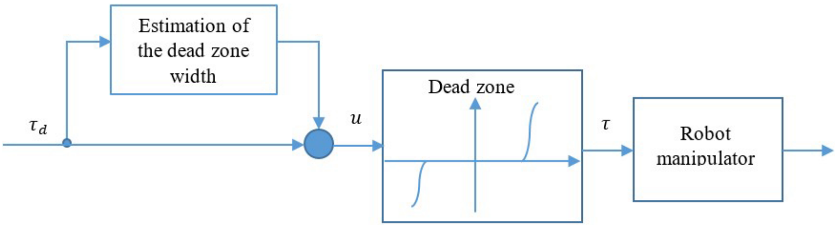

Figure 3: Adaptive dead zone compensation.

To compensate the effects of dead zone, the control input after passing the dead zone can be described in the following form [19]:

$$u = \tau_d + \eta \hat{d}_r + (I - \eta)\hat{d}_l \tag{34}$$

Where $\eta = I$ if $\tau_d \geq 0$, $\eta = 0$ if $\tau_d < 0$. The direct control input for robot manipulator can be expressed as follows:

$$\tau = \tau_d + \eta \hat{d}_r + (I - \eta)\,\hat{d}_l - E_D\left(\tau_d + \eta \hat{d}_r + (I - \eta)\,\hat{d}_l\right) = \tau_d - \tilde{D}^T \Xi + \tilde{D}^T \odot \tag{35}$$

Where $\tilde{D} = D - \hat{D}$, $\tilde{D} = diag\left\{\tilde{d}_1, \tilde{d}_2, \ldots, \tilde{d}_n\right\}$ and $\Xi = [\eta, \;\; I - \eta]^T$ and the modelling mismatch $\odot$ satisfies the bound [7].

$$\|\odot\| \leq \sqrt{n} \tag{36}$$

Here, we proposed an intelligent controller which combines adaptive fuzzy neural networks control and Backstepping technique to suppress the effects of the uncertainties and approximation errors. Thus, the unknown functions of robot manipulator control system are estimated, and the stability can be guaranteed. The block diagram of RAFNNs is described in Fig. 4.

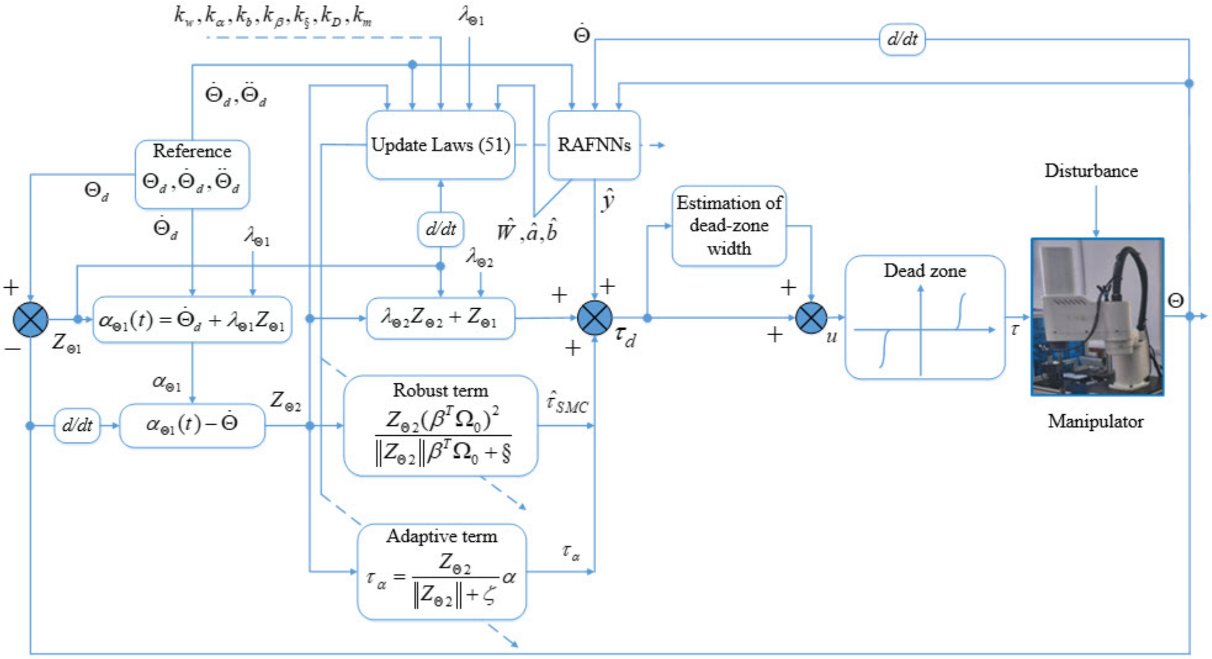
$k_w, k_\alpha, k_b, k_\beta, k_§, k_D, k_m$
$\lambda_{\Theta 1}$
$\dot{\Theta}$
d/dt
$\dot{\Theta}_d, \ddot{\Theta}_d$
Update Laws (51)
RAFNNs
Reference
$\Theta_d, \dot{\Theta}_d, \ddot{\Theta}_d$
Θ_d
Disturbance
$\hat{y}$
Estimation of dead-zone width
$\dot{\Theta}_d$
$\lambda_{\Theta 1}$
d/dt
$\hat{W}, \hat{a}, \hat{b}$
$\lambda_{\Theta 2}$
Dead zone
+
−
$Z_{\Theta 1}$
$\alpha_{\Theta 1}(t) = \dot{\Theta}_d + \lambda_{\Theta 1} Z_{\Theta 1}$
$\lambda_{\Theta 2} Z_{\Theta 2} + Z_{\Theta 1}$
+
+
+
τ_d
+
+
u
τ
Θ
$\alpha_{\Theta 1}$
$Z_{\Theta 2}$
d/dt
$\alpha_{\Theta 1}(t) - \dot{\Theta}$
Robust term
$\frac{Z_{\Theta 2}(\beta^T \Omega_0)^2}{\|Z_{\Theta 2}\| \beta^T \Omega_0 + §}$
$\hat{\tau}_{SMC}$
Manipulator
Adaptive term
$\tau_\alpha = \frac{Z_{\Theta 2}}{\|Z_{\Theta 2}\| + \zeta} \alpha$
τ_α

Figure 4: The block diagram of the adaptive control system.

The RAFNNs control law is presented as:

$$\tau_d = \hat{y} + \lambda_{\Theta 2} Z_{\Theta 2} + Z_{\Theta 1} + \tau_{SMC} + \tau_\alpha - \tilde{D}^T \Xi + \tilde{D}^T \odot \tag{37}$$

Where $\hat{y}$ is the approximation of y function, τ_{SMC} is a sliding control term, and τ_α is the adaptive control.

By using the RAFNNs control law (37) into (12), we can be rewritten as:

$$M_{RM} \dot{Z}_{\Theta 2} = \tilde{y} - (C_{RM} + \lambda_{\Theta 2})\, Z_{\Theta 2} - Z_{\Theta 1} + \tau_0 - \tau_{SMC} - \tau_\alpha - \tilde{D}^T \Xi + \tilde{D}^T \odot \tag{38}$$

With

$$\tilde{y} = y - \hat{y} = W^{*T} \delta^* - \hat{W}^T \hat{\delta} + \Delta \tag{39}$$

The parameter errors are defined as: $\tilde{W} = W^* - \hat{W}; \tilde{\delta} = \delta^* - \hat{\delta}; \tilde{m} = m^* - \hat{m}$ and $\tilde{b} = b^* - \hat{b}$. Thus, equation (39) is possible to be rephrased as:

$$\tilde{y} = W^{*T} \tilde{\delta} + \tilde{W}^T \hat{\delta} + \Delta \tag{40}$$

The function $\tilde{\delta}$ can be expanded in a Taylor series as:

$$\tilde{\delta} = \left[\frac{\partial \delta_1}{\partial m}, \frac{\partial \delta_2}{\partial m}, \dots, \frac{\partial \delta}{\partial m}\right]_{m=\hat{m}} \tilde{m} + \left[\frac{\partial \delta_1}{\partial b}, \frac{\partial \delta_2}{\partial b}, \dots, \frac{\partial \delta_n}{\partial b}\right]_{b=\hat{b}} \tilde{b} + H\left(\tilde{m}, \tilde{b}\right) \tag{41}$$

Or

$$\tilde{\delta} = \Upsilon^T \tilde{m} + \Gamma^T \tilde{b} + H\left(\tilde{m}, \tilde{b}\right) \tag{42}$$

where $H(\tilde{m}, \tilde{b}) \in R^n$ is the higher - order term vector:

$$\Upsilon^T = \left[\frac{\partial \delta_1}{\partial m}, \frac{\partial \delta_2}{\partial m}, \dots, \frac{\partial \delta_n}{\partial m}\right]_{m=\hat{m}} \in R^{n\times(nm)};\ \Gamma^T = \left[\frac{\partial \delta_1}{\partial b}, \frac{\partial \delta_2}{\partial b}, \dots, \frac{\partial \delta_n}{\partial b}\right]_{b=\hat{b}} \in R^{n\times(nm)}$$

Substitute (42) into (40), we have:

$$\begin{aligned}\tilde{y} + \tau_0 &= \tilde{W}^T\left[\hat{\delta} + \Upsilon^T(m^* - \hat{m}) + \Gamma^T\left(b^* - \hat{b}\right)\right] + \hat{W}\left(\Upsilon^T\tilde{m} + \Gamma^T\tilde{b}\right) \\ &+ W^{*T} H\left(\tilde{m}, \tilde{b}\right) + \Delta + \tau_0 \\ &= \tilde{W}^T\left(\hat{\delta} - \Upsilon^T\hat{m} - \Gamma^T\hat{b}\right) + \hat{W}^T\left(\Upsilon^T\tilde{m} + \Gamma^T\tilde{b}\right) + \omega\left(x, N\right)\end{aligned} \tag{43}$$

Where $\omega = [\omega_1, \omega_2, \dots, \omega_n]^T \in R^n$ and using (42), we obtain

$$\begin{aligned}\omega &= \tilde{W}^T\left(\Upsilon^T m^* + \Gamma^T b^*\right) + W^{*T} H\left(\tilde{m}, \tilde{b}\right) + \Delta \\ &= \left(W^{*T} - \hat{W}^T\right)\left(\Upsilon^T m^* + \Gamma^T b^*\right) + W^{*T}\left(\tilde{\delta} - \Upsilon^T\tilde{m} - \Gamma^T\tilde{b}\right) + \Delta + \tau_0 \\ &= W^{*T}\left(\tilde{\delta} + \Upsilon^T\hat{m} + \Gamma^T\hat{b}\right) - \hat{W}^T\left(\Upsilon^T m^* + \Gamma^T b^*\right) + \Delta + \tau_0\end{aligned}$$

The bound of ω is determined as:

$$\begin{aligned}\|\omega\| &= \|W^{*T}\left(\tilde{\delta} + \Upsilon^T\hat{m} + \Gamma^T\hat{b}\right) - \hat{W}^T\left(\Upsilon^T m^* + \Gamma^T b^*\right) + \Delta + \tau_0\| \\ &= \|(W^{*T}\tilde{\delta} + \Delta + \tau_0) + W^{*T}\left(\Upsilon^T\hat{m} + \Gamma^T\hat{b}\right) - \hat{W}^T\left(\Upsilon^T m^* + \Gamma^T b^*\right)\|\end{aligned}$$

Since

$$\begin{gathered}\|W^{*T}\Upsilon^T\hat{m}\| \le \|W^{*T}\Upsilon^T\|\|\hat{m}\|; \|W^{*T}\Gamma^T\hat{b}\| \le \|W^{*T}\Gamma^T\|\|\hat{b}\|; \\ \|\hat{W}^T\left(\Upsilon^T m^* + \Gamma^T b^*\right)\| \le \|\hat{W}^T\|\|\Upsilon^T m^* + \Gamma^T b^*\|\end{gathered}$$

Hence, we can infer:

$$\begin{aligned}\|\omega\| &\le \|W^{*T}\tilde{\delta} + \Delta + \tau_0\| + \|W^{*T}\Upsilon^T\|\|\hat{m}\|\|W^{*T}\Gamma^T\|\|\hat{b}\|\|\hat{W}^T\|\|\Upsilon^T m^* + \Gamma^T b^*\| \\ &\le \left[\|W^{*T}\tilde{\delta} + \Delta + \tau_0\|, \|W^{*T}\Upsilon^T\|, \|W^{*T}\Gamma^T\|, \left(\|\Upsilon^T m^* + \Gamma^T b^*\|\right)\right]^T\left[1, \|\hat{m}\|, \|\hat{b}\|, \|\hat{W}\|\right] \\ &\le \beta^{*T}\Omega_0\end{aligned} \tag{44}$$

Where

$$\beta^* = \left[\|W^{*T}\tilde{\delta} + \Delta + \tau_0\|, \|W^{*T}\Upsilon^T\|, \|W^{*T}\Gamma^T\|, \left(\|\Upsilon^T m^* + \Gamma^T b^*\|\right)\right]^T; \Omega_0 = \left[1, \|\hat{m}\|, \|\hat{b}\|, \|\hat{W}\|\right]^T$$

Follow above analysis, a sliding mode control term τ_{SMC} is designed by:

$$\tau_{SMC} = \frac{Z_{\Theta 2}\left(\beta^T\Omega_0\right)^2}{\|Z_{\Theta 2}\|\beta^T\Omega_0 + \mathcal{S}} \tag{45}$$

Where $\mathcal{S}$ is a positive scalar control gain

$$\dot{\mathcal{S}} = -k_{\mathcal{S}}\mathcal{S}, \mathcal{S}(0) > 0 \tag{46}$$

With $\beta = [\beta_1, \beta_2, \beta_3, \beta_4]^T$ is a bound of vector β^*.
To estimate the sliding control term τ_{SMC} we present adaptive term $\hat{\tau}_{SMC}$ as:

$$\hat{\tau}_{SMC} = \frac{Z_{\Theta 2}\left(\hat{\beta}^T\Omega_0\right)^2}{\|Z_{\Theta 2}\|\hat{\beta}^T\Omega_0 + \mathcal{S}} \tag{47}$$

Where $\hat{\beta}$ is the estimate of β^*.
The adaptive control τ_α is designed by:

$$\tau_\alpha = \frac{Z_{\Theta 2}}{\|Z_{\Theta 2}\| + \zeta}\alpha \tag{48}$$

Where $\zeta > 0$ and it is chosen such that $\int\limits_0^\infty \zeta dt < \infty$.
With the adaptation law as follows:

$$\dot{\alpha} = \frac{\|Z_{\Theta 2}\|^2}{\|Z_{\Theta 2}\| + \zeta} \tag{49}$$

Applying (43) to (38), yields:

$$\begin{aligned} M_{RM}\dot{Z}_{\Theta 2} = \tilde{W}^T\left(\hat{\delta} - \Upsilon^T\hat{m} - \Gamma^T\hat{b}\right) + \hat{W}^T\left(\Upsilon^T\tilde{m} + \Gamma^T\tilde{b}\right) + \omega \\ - \left(C_{RM} + \lambda_{\Theta 2}\right)Z_{\Theta 2} - Z_{\Theta 1} - \tau_{SMC} - \tau_\alpha - \tilde{D}^T\Xi + \tilde{D}^T\odot \end{aligned} \tag{50}$$

Based on the aforementioned analysis, the online adaptive update laws of RAFNNs, adaptive control, and sliding control term parameters can be chosen as:

$$\begin{cases} \dot{\hat{W}} = k_W(\hat{\delta} - \Upsilon^T\hat{m} - \Gamma^T\hat{b})Z_{\Theta 2}^T \\ \dot{\hat{m}} = k_m\hat{W}\Upsilon^T Z_{\Theta 2} \\ \dot{\hat{b}} = k_b\hat{W}\Gamma^T Z_{\Theta 2} \\ \dot{\hat{\beta}} = k_\beta\|Z_{\Theta 2}\|\Omega_0 \\ \dot{\mathcal{S}} = -k_{\mathcal{S}}\mathcal{S} \\ \dot{\alpha} = \frac{k_\alpha\|Z_{\Theta 2}\|^2}{\|Z_{\Theta 2}\| + \zeta} \\ \dot{\hat{D}} = k_D\Xi Z_{\Theta 2}^T - k_D\alpha_D\hat{D}\|Z_{\Theta 2}\| \end{cases} \tag{51}$$

Here $k_w, k_m, k_b, k_\beta, k_{\mathcal{S}}, k_\alpha, k_D$ are positive adaptation rates.

3.2 Stability analysis

Theorem 1: Consider the RAFNNs adaptive control law of an n-link robot manipulator represented by (1) is designed in (37), and a sliding control term τ_{SMC} is given by (45), the adaptation control is defined in (48) and the parameters $\dot{\hat{W}}$, $\dot{\hat{m}}$, $\dot{\hat{b}}$, $\dot{\hat{\beta}}$, $\dot{\mathcal{S}}$, $\dot{\alpha}$, $\dot{\hat{D}}$ are adjusted by the adaptive algorithm (51). Then the position tracking error and all the system parameters converges to zero.

Proof: The Lyapunov function candidate is chosen as follows as:

$$\begin{aligned} L\left(t\right) = \frac{1}{2}\Bigg[& Z_{_{\Theta 1}}^{T} Z_{\Theta 1} + Z_{\Theta 2}^{T} M_{RM} Z_{\Theta 2} + \frac{1}{k_w}\tilde{W}^T\tilde{W} + \frac{1}{k_m}\tilde{m}^T\tilde{m} + \frac{1}{k_b}\tilde{b}^T\tilde{b} + \frac{1}{k_\beta}\tilde{\beta}^T\tilde{\beta} \\ & + \frac{2}{k_{\mathcal{S}}}\mathcal{S} + \frac{1}{k_\alpha}\alpha^2 + \frac{1}{k_D} tr\left(\tilde{D}^T\tilde{D}\right)\Bigg] \end{aligned} \tag{52}$$

The derivative of $L(t)$ along to time, we have:

$$\begin{aligned} \dot{L}\left(t\right) = & Z_{\Theta 1}^{T}\left(Z_{\Theta 2}\left(t\right) - \lambda_{\Theta 1} Z_{\Theta 1}\right) + \frac{1}{2} Z_{\theta 2}^{T}\dot{M}_{RM} Z_{\Theta 2} + Z_{\Theta 2}^{T} M_{RM}\dot{Z}_{\Theta 2} - \frac{1}{k_w}\tilde{W}^T\dot{\hat{W}} \\ & - \frac{1}{k_m}\tilde{m}^T\dot{\hat{m}} - \frac{1}{k_b}\tilde{b}^T\dot{\hat{b}} - \frac{1}{k_\beta}\tilde{\beta}^T\dot{\hat{\beta}} + \frac{1}{k_{\mathcal{S}}}\dot{\mathcal{S}} + \frac{1}{k_\alpha}\alpha\dot{\alpha} - \frac{1}{k_D} tr\left(\tilde{D}^T\dot{\hat{D}}\right) \end{aligned} \tag{53}$$

Substitute (50) into (53) and using property 2, we obtain:

$$\begin{aligned} \dot{L}\left(t\right) = & -Z_{\Theta 1}^{T}\lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\lambda_{\Theta 2} Z_{\Theta 2} + Z_{\Theta 2}^{T}\Big[\tilde{W}^T\left(\hat{\delta} - \Upsilon^T\hat{m} - \Gamma^T\hat{b}\right) \\ & + \hat{W}^T\left(\Upsilon^T\tilde{m} + \Gamma^T\tilde{b}\right) + \omega - \tau_{SMC} - \tau_\alpha + \tilde{D}^T\Xi - \tilde{D}^T\odot\Big] - \frac{1}{k_w}\tilde{W}^T\dot{\hat{W}} - \frac{1}{k_v}\tilde{m}^T\dot{\hat{m}} \\ & - \frac{1}{k_b}\tilde{b}^T\dot{\hat{b}} - \frac{1}{k_\beta}\tilde{\beta}^T\dot{\hat{\beta}} + \frac{1}{k_{\mathcal{S}}}\dot{\mathcal{S}} + \frac{1}{k_\alpha}\alpha\frac{k_\alpha\left\|Z_{\theta 2}\right\|^2}{\left\|Z_{\theta 2}\right\| + \zeta} - \frac{1}{k_D} tr\left(\tilde{D}^T\dot{\hat{D}}\right) \end{aligned} \tag{54}$$

Substituting the adaptive algorithm (51) to (54), we have:

$$\begin{aligned} \dot{L}\left(t\right) = & -Z_{\Theta 1}^{T}\lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\lambda_{\Theta 2} Z_{\Theta 2} + Z_{\Theta 2}^{T}\left(\omega - \tau_{SMC} - \tau_\alpha\right) + Z_{\Theta 2}^{T}\left(\tilde{D}^T\Xi - \tilde{D}^T\odot\right) \\ & - \tilde{\beta}^T\left\|Z_{\Theta 2}\right\|\Omega_0 - \mathcal{S} + \alpha\frac{\left\|Z_{\Theta 2}\right\|^2}{\left\|Z_{\Theta 2}\right\| + \zeta} - tr\left(\tilde{D}^T\left(\Xi Z_{_{\Theta 2}}^{T} - \alpha_D\hat{D}\left\|Z_{\Theta 2}\right\|\right)\right) \end{aligned} \tag{55}$$

By using (44), (45) and (48), it becomes:

$$\begin{aligned} \dot{L}\left(t\right) \leq & -Z_{\Theta 1}^{T}\lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\lambda_{\Theta 2} Z_{\Theta 2} - Z_{\Theta 2}^{T}\frac{Z_{\Theta 2}\left(\hat{\beta}^T\Omega_0\right)^2}{\left\|Z_{\Theta 2}\right\|\hat{\beta}^T\Omega_0 + \mathcal{S}} + Z_{\Theta 2}^{T}\beta^{*T}\Omega_0 \\ & - \tilde{\beta}^T\left\|Z_{\Theta 2}\right\|\Omega_0 - \mathcal{S} + tr\left(\tilde{D}^T Z_{\Theta 2}^{T}\left(\alpha_D\hat{D} - \odot\right)\right) \\ \leq & -Z_{\Theta 1}^{T}\lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\lambda_{\Theta 2} Z_{\Theta 2} + \frac{\mathcal{S}\left\|Z_{\Theta 2}\right\|\hat{\beta}^T\Omega_0}{\mathcal{S}\left\|Z_{\Theta 2}\right\|\hat{\beta}^T\Omega_0 + \mathcal{S}} - \mathcal{S} + tr\left(\tilde{D}^T Z_{\Theta 2}^{T}\left(\alpha_D\hat{D} - \odot\right)\right) \end{aligned} \tag{56}$$

Since the sum of the last two terms in (56) is always less than zero, we can place the new upper bound on $\dot{L}$.

$$\dot{L}\left(t\right) \leq -Z_{\Theta 1}^{T}\lambda_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\lambda_{\Theta 2} Z_{\Theta 2} + tr\left(\tilde{D}^T Z_{\Theta 2}^{T}\left(\alpha_D\left(D - \tilde{D}\right) - \odot\right)\right) \tag{57}$$

By using $tr\tilde{D}^T\left(D - \tilde{D}\right) = \left(\tilde{D}, D\right) - \left\|\tilde{D}\right\|^2 \leq \left\|\tilde{D}\right\|\left\|D\right\| - \left\|\tilde{D}\right\|^2$ and using (36) into the inequality (57) could be rewritten as follows:

$$\begin{aligned} \dot{L} & \leq -Z_{\Theta 1}^{T}\Omega_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\Omega_{\Theta 2} Z_{\Theta 2} + \sqrt{n}\left\|Z_{\Theta 2}\right\|\left\|\tilde{D}\right\| + \alpha_D D_M\left\|Z_{\Theta 2}\right\|\left\|\tilde{D}\right\| - \alpha_D\left\|Z_{\Theta 2}\right\|\left\|\tilde{D}\right\|^2 \\ & \leq -Z_{\Theta 1}^{T}\Omega_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^{T}\Omega_{\Theta 2} Z_{\Theta 2} + c_0\left\|Z_{\Theta 2}\right\|\tilde{D}\| - \alpha_D\left\|Z_{\Theta 2}\right\|\left\|\tilde{D}\right\|^2\| \end{aligned} \tag{58}$$

With $c_0 = \sqrt{n} + \alpha_D D_M$.

We see that to make sure $\dot{L} \leq 0$

$$-c_0\|Z_{\Theta 2}\|\|\tilde{D}\| + \alpha_D\|Z_{\Theta 2}\|\|\tilde{D}\|^2 > 0 \tag{59}$$

So, if we choose suitable constant vectors α_D, D_M which satisfy (59), $\dot{L}\left(Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right) \leq 0$, $\dot{L}\left(Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right)$ is a negative semidefinite function, $\dot{L}\left(Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right) \leq \dot{L}\left(Z_{\Theta 1}(0), Z_{\Theta 2}(0), \mathcal{S}(0), \tilde{W}, \tilde{b}, \tilde{\beta}\right)$, if all parameters such as $Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}$ are bounded with $t > 0$. By defining $\epsilon(t) = -Z_{\Theta 1}^T \Omega_{\Theta 1} Z_{\Theta 1} - Z_{\Theta 2}^T \Omega_{\Theta 2} Z_{\Theta 2}$ so $\epsilon(t) \leq -\dot{L}(t)$ and integrate the $\epsilon(t)$ with respect to time as follows:

$$\int_0^t (\xi)\, d\xi \leq L\left(Z_{\Theta 1}(0), Z_{\Theta 2}(0),\, \mathcal{S}(0), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right) - L\left(Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right) \tag{60}$$

Because $L\left(Z_{\Theta 1}(0), Z_{\Theta 2}(0), \mathcal{S}(0), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right)$ is a bounded function, and $L\left(Z_{\Theta 1}(t), Z_{\Theta 2}(t), \mathcal{S}(t), \tilde{W}, \tilde{a}, \tilde{b}, \tilde{\beta}\right)$ is nonincreasing and bounded, we have

$$\lim_{t \to \infty} \int_0^t \epsilon(\xi)\, d\xi < \infty \tag{61}$$

According to Barbalat's Lemma [20], when $\dot{\epsilon}(t)$ is bounded function. It can be shown that $\lim\limits_{t \to \infty} \int\limits_0^t \epsilon(t)\, dt = 0$. From this outcome, we see that, $Z_\Theta(t)$ will converge to zero when $t \to \infty$ and the global stability of the control system for RM is assured by the updated law (37).

4 Simulation and experimental results

4.1 Simulation results

Here, a three-link RM is applied to confirm the efficiency of the suggested control method based on RAFNNs for illustrative purposes. The detailed system parameters of three-link RM model (Fig. 5) are given as follows:

$$M_{RM} = \begin{bmatrix} M_{RM11} & M_{RM12} & M_{RM13} \\ M_{RM21} & M_{RM22} & M_{RM23} \\ M_{RM31} & M_{RM32} & M_{RM33} \end{bmatrix}; C_{RM} = \begin{bmatrix} C_{RM11} & C_{RM12} & C_{RM13} \\ C_{RM21} & C_{RM22} & C_{RM23} \\ C_{RM31} & C_{RM32} & C_{RM33} \end{bmatrix};$$

$$G_{RM} = \begin{bmatrix} G_{RM1} \\ G_{RM2} \\ G_{RM3} \end{bmatrix};$$

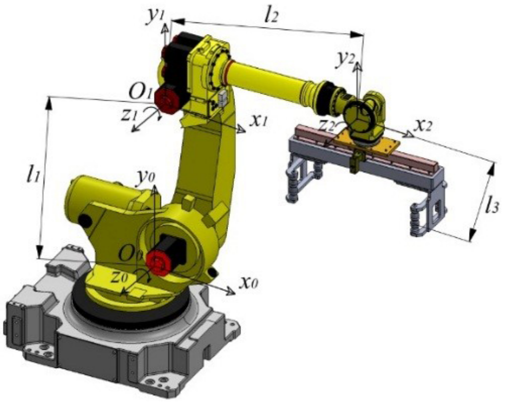

Figure 5: The model of three-joint RM.

$$
\begin{aligned}
M_{RM11} &= (m_1+m_2+m_3)\,l_1^2 + (m_2+m_3)\,l_2^2 + m_3 l_3^2 + 2\,(m_2+m_3)\,l_1 l_2 \cos(\Theta_2)\,;\\
&\quad + 2 m_3 l_1 l_3 \cos(\Theta_2+\Theta_3) + 2 m_3 l_2 l_3 \cos(\Theta_3)\,;\\
M_{RM12} &= (m_2+m_3)\,l_2^2 + m_3 l_3^2 + (m_2+m_3)\,l_1 l_2 \cos(\Theta_2)\\
&\quad + m_3 l_1 l_3 \cos(\Theta_2+\Theta_3) + 2 m_3 l_2 l_3 \cos(\Theta_3)\,;\\
M_{RM13} &= m_3 l_3^2 + m_3 l_1 l_3 \cos(\Theta_2+\Theta_3) + m_3 l_2 l_3 \cos(\Theta_3)\,;\ M_{RM21} = M_{RM12};\\
M_{RM22} &= (m_2+m_3)\,l_2^2 + m_3 l_3^2 + 2 m_3 l_2 l_3 \cos(\Theta_3)\,;\ M_{RM23} = m_3 l_3^2 + m_3 l_2 l_3 \cos(\Theta_3)\,;\\
M_{RM31} &= m_3 l_3^2 + m_3 l_1 l_3 \cos(\Theta_2+\Theta_3) + m_3 l_2 l_3 \cos(\Theta_3)\,;\\
M_{RM32} &= M_{23};\ M_{RM33} = m_3 l_3^2;\\
C_{RM11} &= -2\,(m_2+m_3)\,l_1 l_2 \sin(\Theta_2)\,\dot\Theta_2 - 2 m_3 l_1 l_3 \sin(\Theta_2+\Theta_3)\left(\dot\Theta_2+\dot\Theta_3\right)\\
&\quad - 2 m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_3;\\
C_{RM12} &= -\,(m_2+m_3)\,l_1 l_2 \sin(\Theta_2)\,\dot\Theta_2 - m_3 l_1 l_3 \sin(\Theta_2+\Theta_3)\left(\dot\Theta_2\right)\\
&\quad - 2 m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_3 - 2 m_3 l_1 l_3 \sin(\Theta_2+\Theta_3)\,\dot\Theta_3;\\
C_{RM13} &= -m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_3 - m_3 l_1 l_3 \sin(\Theta_2+\Theta_3)\,\dot\Theta_3;\ C_{RM22} = -2 m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_3;\\
C_{RM23} &= -m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_3;\ C_{RM32} = m_3 l_2 l_3 \sin(\Theta_3)\,\dot\Theta_2;\ C_{RM33} = 0;\\
G_{RM1} &= (m_1+m_2+m_3)\,g l_1 sin(\Theta_1) + (m_2+m_3)\,g l_2 sin(\Theta_1+\Theta_2)\\
&\quad + m_3 g l_3 sin(\Theta_1+\Theta_2+\Theta_3)\,;\\
G_{RM2} &= (m_2+m_3)\,g l_2 sin(\Theta_1+\Theta_2) + m_3 g l_3 sin(\Theta_1+\Theta_2+\Theta_3)\,;\\
G_{RM3} &= m_3 g l_3 sin(\Theta_1+\Theta_2+\Theta_3)
\end{aligned}
$$

Where m_1, m_2, m_3 are links masses; l_1, l_2, l_3 are links lengths; The parameters of three link RM are given as follows: $m_1 = 5.4\,(kg)$, $m_2 = 4.4\,(kg)\,, m_3 = 2\,(kg)\,, l_1 = 520\,(mm)\,, l_2 = 420\,(mm)$, $l_3 = 260\,(mm)$; $g = 9.8\,\left(m/s^2\right)$.

The desired joint trajectories of the three link robot manipulator are chosen by:

$$\Theta_d = \begin{bmatrix} \Theta_{d1} & \Theta_{d2} & \Theta_{d3} \end{bmatrix}^T = \begin{bmatrix} 0.5\sin(2\pi t) & 0.5\sin(2\pi t) & 0.5\sin(2\pi t) \end{bmatrix}^T.$$

In addition, external disturbances and friction force in this simulation are selected as following:

$$\tau_0 = \begin{bmatrix} 3\sin(\pi t) \\ 3\sin(\pi t) \\ 3\sin(\pi t) \end{bmatrix};\ F_{IRM}\left(\dot{\Theta}\right) = \begin{bmatrix} 5\dot{\Theta}_1 + 0.3sign\left(\dot{\Theta}_1\right) \\ 5\dot{\Theta}_2 + 0.3sign\left(\dot{\Theta}_2\right) \\ 5\dot{\Theta}_3 + 0.3sign\left(\dot{\Theta}_3\right) \end{bmatrix}.$$

The proposed controller parameter values are given as follows: $\lambda = diag\,(6,6,6)\,; K = diag\,(100\ ,\ 110\ ,100)\,;\ k_w = diag\,(60,\ 60,\ 60,\ 60,\ 60)\,;\ k_m = k_b = diag\,(50,40,50) \in R^{mp\times mp}$; $k_\beta = diag\,(0.001, 0.001, 0.001, 0.001, 0.001)$; $k_{\mathcal{S}} = 0.12$; $k_\alpha = 0.1$,$h_r\,(u) = u - d_r$; $h_l\,(u) = u + d_l$; $d_r = 10$; $d_l = -10$; $k_D = 10$; $\alpha_D = 0.2$.

The initial conditions are selected as follows: $\mathcal{S}\,(0) = 1$, $\Omega_0\,(0) = [1\ \ 1\ \ 1\ \ 1]$, $\zeta\,(0) = 1$.

In here, the proposed RAFNNs is applied to control the RM in comparison with the NNs [26] and PID. The simulation results of the NNs, PID and the proposed RAFNNs are shown in Fig. 6. Since the simulated results, we see that, the position tracking of three links with the RAFNNs, NNs, and PID can be guaranteed, and the tracking errors of the NNs, PID and the proposed intelligent controller are converged. However, the proposed intelligent control system converges faster than the NNs and PID systems. It means that all updated parameters in the dynamic structure RAFNNs and the amount of the rule nodes are adjusted, the approximation capability of the dynamics structure RAFNNs is also superior to the NNs and PID systems. Moreover, from Fig. 6 it can be observe that, the control force of the suggested RAFNNs is smoother and has smaller oscillation than the NNs and PID to attain the requested level of performance when the tracking errors reach the high value.

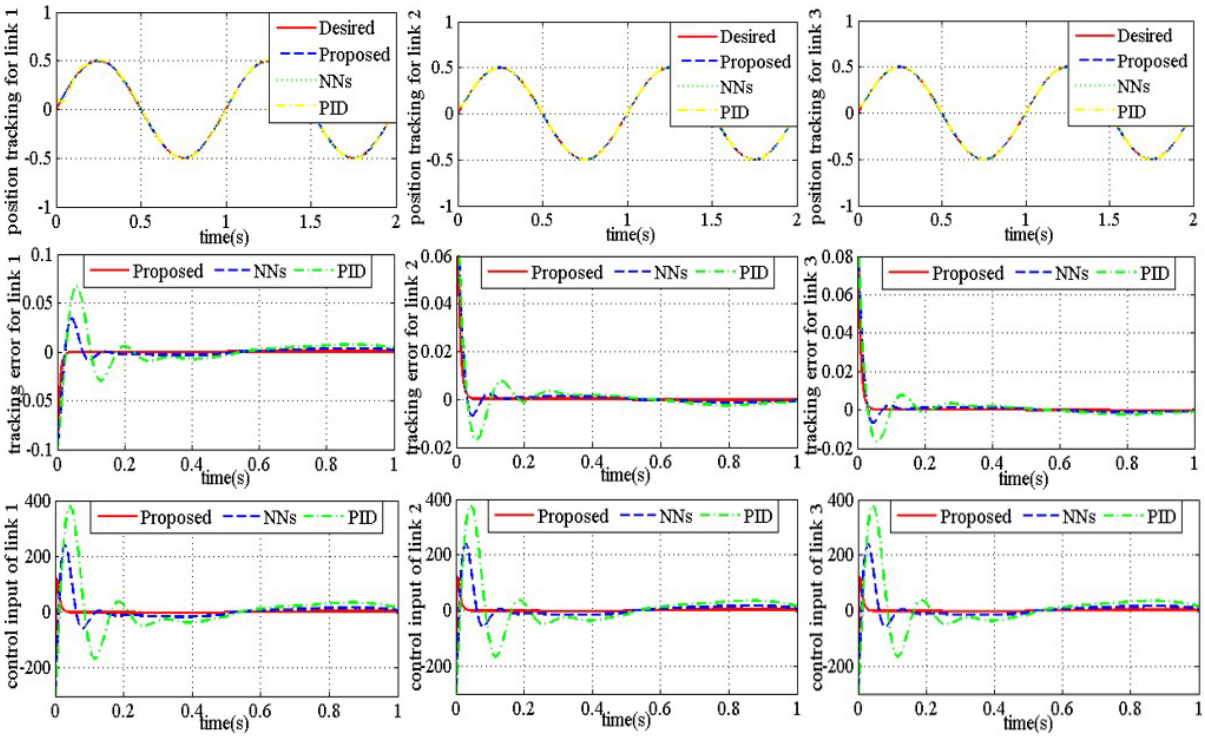

Figure 6: Simulated results of trajectory, tracking errors, and control efforts of the NNs, PID and proposed RAFNNs system.

4.2 Experimental results

Here, we implemented two experimental outcomes to prove the efficiency of the RAFNNs controller on a three-link robot manipulators. Fig. 7 illustrates the experimental control system model.

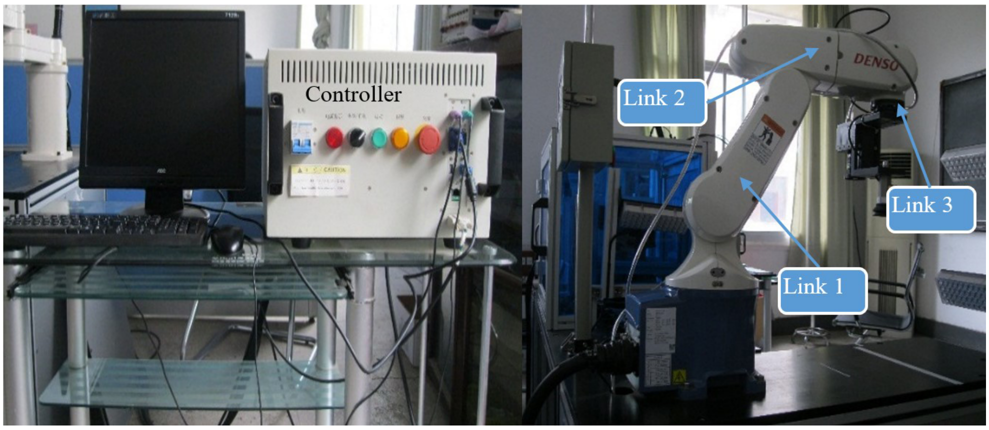

Figure 7: Experimental control system.

The first experimental example assumes that 0.5-kg payload is added in the masses of three links RM, and all parameters are the same as in the simulation model. The experimental results of joint trajectory, control torques and tracking errors are exposed in Fig. 8. From Fig. 8, we can easily see that the position tracking of RM are still obtained with PID, NNs, and RAFNNs. However the responses and the tracking error norm of the RAFNNs are quite better than the NNs and PID methods. Furthermore, from these results can see that the proposed intelligent controller torques are less and smooth than NNs in [26], and PID which still exist the chattering phenomena when a load of manipulators changed. Therefore, the position tracking performance of the recommended RAFNNs is better than the NNs and PID under parameters variation. It means that due to the dynamic structure, the proposed RAFNNs is less sensitive to the parameter variation than the NNs and PID.

The second experimental case, the external disturbance $d_e(t)$ is suddenly injected more into control system when the robot is tracking a trajectory. This occurred after the first 0.5s of the experimental period, and all other parameters are the same as in the simulation model. The external disturbance shapes are expressed as follows:

$$d_e(t) = \begin{bmatrix} 100\sin(100t) & 100\sin(100t) & 100\sin(100t) \end{bmatrix}^T$$

.

The experimental outcomes of the second case are shown in Fig. 9. According to these results, it is easy to see that, the performance of the proposed RAFNNs is just slightly affected, while the performance of PID approach is seriously affected. Therefore, the control performance and robustness of the proposed RAFNNs under external disturbance are better than the NNs [26] and PID. It is obvious that the performance of the proposed RAFNNs is better than the NNs and PID system after a period of learning.

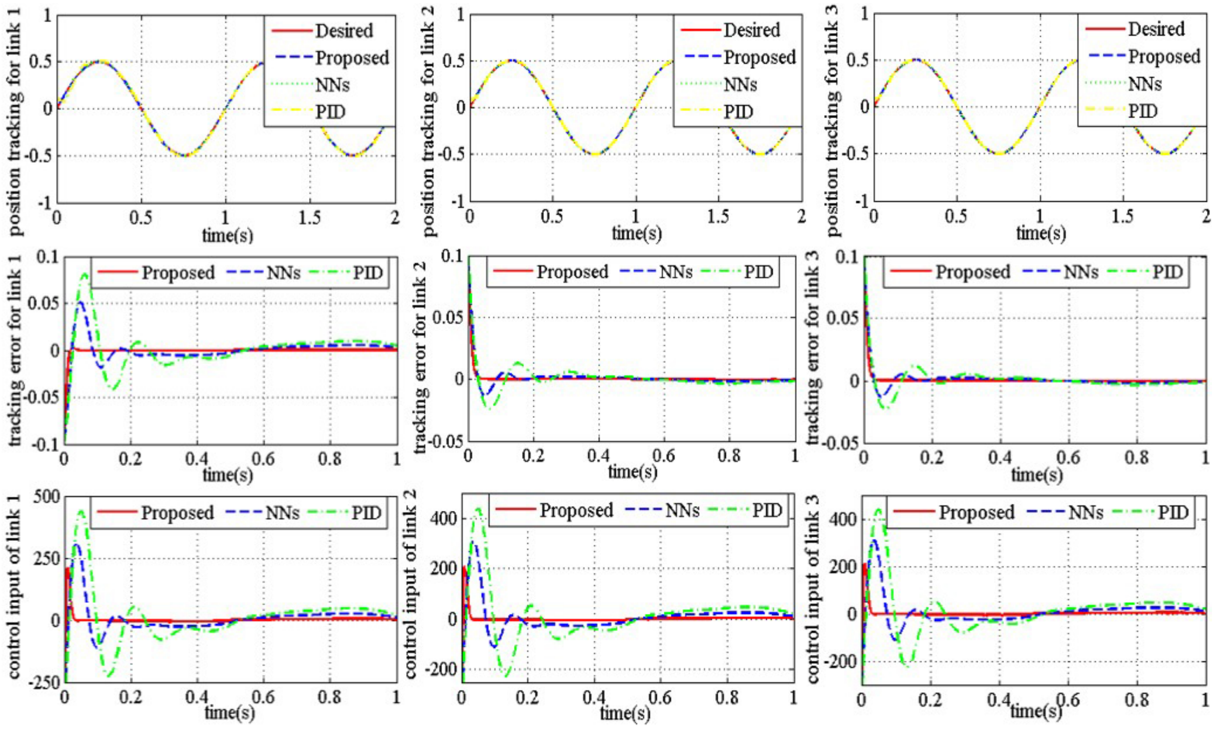

Figure 8: Trajectory, tracking errors, and control efforts of the NNs, PID and proposed RAFNNs system in the first experimental case.

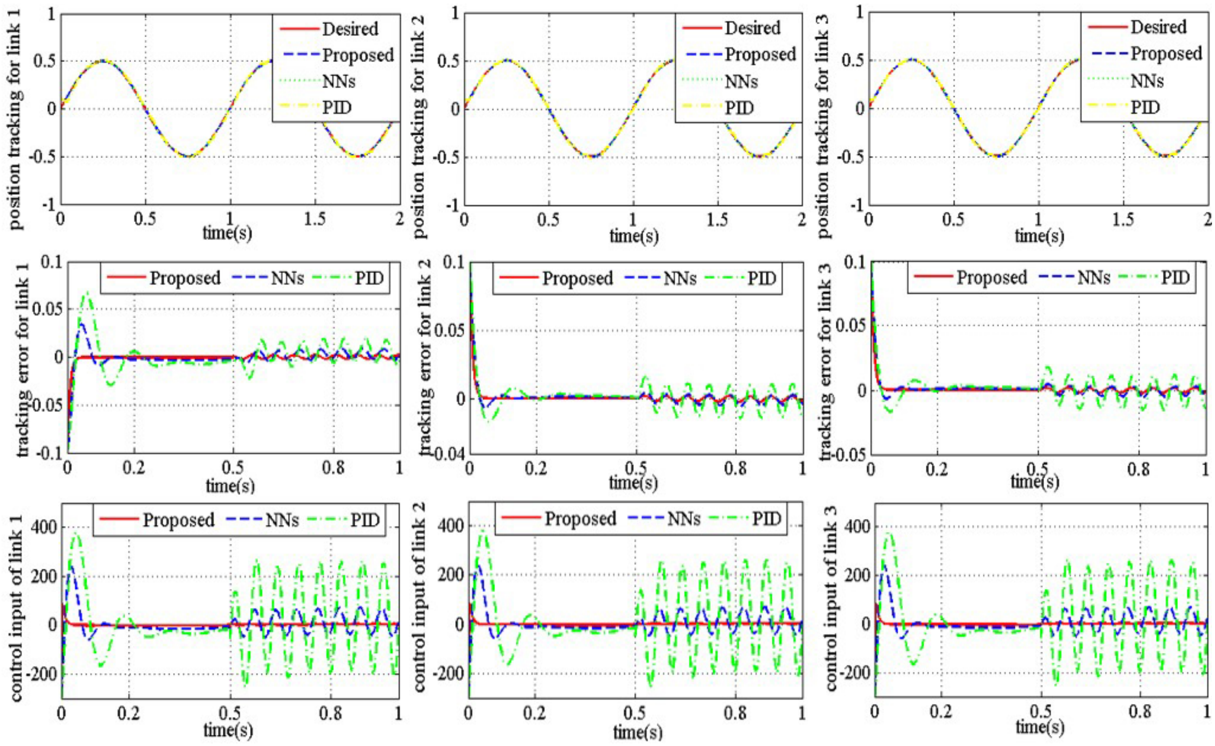

Figure 9: Trajectory, tracking errors, and control efforts of the NNs, PID and proposed RAFNNs system in the second experimental case.

5 Conclusions

In this paper, an adaptive robust control system based on structure FNN has been proposed, and the mathematical proof has been presented. It has been also successfully implemented to control the joints of three-link RM for achieving high precision position tracking and compensation dead-zone. By combining the FNN, Lyapunov stability theorem, the adaptive control laws are developed to tune all parameters of the network in order to reduce approximation error and improved control performance. In addition, the robust term is designed dealing with the approximation error, prime parameter vectors and higher order terms in Taylor series. Therefore, the proposed controller proved that this control system could achieve desired tracking performance. The stability and robustness of the closed-loop manipulators system are guaranteed. Simulation and experimental results of three-links RM via the proposed RAFNNs and NNs, PID also have provided in this study to compare and display. The proposed RAFNNs control systems can be applied to other systems, such as MMR, AC servo systems, and they can also be applied as a good alternative in the existing robot manipulator control system. This application could require further investigations.

Bibliography

[1] Baigzadehnoe, B.; Rahmani, Z.; Khosravi, A.; Rezaie, B. (2017). On position/force tracking control problem of cooperative robot manipulators using adaptive fuzzy backstepping approach, *ISA Transactions*, 70, 432–446, 2017.

[2] Bragina, A. A.; Shcherbakov, V. P.; Shiryaev, V. I. (2018). Synthesis of Adaptive Control of Robotic Manipulator by the Method of Lyapunov Functions, *IFAC-PapersOnLine*, 51, 298–303, 2018.

[3] Chen, C. (2011). Robust Self-Organizing Neural-Fuzzy Control With Uncertainty Observer for MIMO Nonlinear Systems, *IEEE Transactions on Fuzzy Systems*, 19, 694–706, 2011.

[4] Chung, C.; Chang, Y. (2013). Backstepping control of multi-input non-linear systems, *IET Control Theory & Applications*, 7, 1773–1779, 2013.

[5] He, J.; Luo, M.; Zhang, Q.; Zhao, J.; Xu, L. (2016). Adaptive Fuzzy Sliding Mode Controller with Nonlinear Observer for Redundant Manipulators Handling Varying External Force, *Journal of Bionic Engineering*, 13, 600–611, 2016.

[6] He, W.; Dong, Y.; Sun, C. (2015). Adaptive neural network control of unknown nonlinear affine systems with input deadzone and output constraint, *ISA Transactions*, 58, 96–104, 2015.

[7] Ik Han, S.; Lee, J. (2016). Finite-time sliding surface constrained control for a robot manipulator with an unknown deadzone and disturbance, *ISA Transactions*, 65, 307–318, 2016.

[8] Ishii, C.; Shen, T.; Tamura, K. (1997). Robust model-following control for a robot manipulator, *IEE ProcControl Theory Appl*, 144(1), 53–60, 1997.

[9] Jing, C.; Xu, H.; Niu, X. (2019). Adaptive sliding mode disturbance rejection control with prescribed performance for robotic manipulators, *ISA transactions*, 91, 41–51, 2019.

[10] Jing, Z.; Changyun, W.; Ying, Z. (2004). Adaptive backstepping control of a class of uncertain nonlinear systems with unknown backlash-like hysteresis, *IEEE Transactions on Automatic Control*, 49, 1751–1759, 2004.

[11] Karamali Ravandi, A.; Khanmirza, E.; Daneshjou, K. (2018). Hybrid force/position control of robotic arms manipulating in uncertain environments based on adaptive fuzzy sliding mode control, *Applied Soft Computing*, 70, 864–874, 2018.

[12] Khorashadizadeh S.; Sadeghijaleh, M. (2018). Adaptive fuzzy tracking control of robot manipulators actuated by permanent magnet synchronous motors, *Computers & Electrical Engineering*, 72, 100–111, 2018.

[13] Krstic, M.; Kanellakopoulos, I.; Kokotovic, P. V. (1995). *Nonlinear Adaptive Control Design*, New York, NY, USA: Wiley, 1995.

[14] Lewis, F. L.; Tim, K.; Wang, L. Z.; Li, Z. X. (1999). Deadzone compensation in motion control systems using adaptive fuzzy control system, *IEEE Trans. Control. Syst. Technol*, 7, 731–742, 1999.

[15] Peng, J.; Dubay, R. (2019). Adaptive fuzzy backstepping control for a class of uncertain nonlinear strict-feedback systems based on dynamic surface control approach, *Expert Systems with Applications*, 120, 239–252, 2019.

[16] Precup, R.; Tomescu, M.; Preitl, S. (2009). Fuzzy logic control system stability analysis based on Lyapunov's direct method, *International journal of computer, communication & control*, IV, 415–426, 2009.

[17] Rossomando, F. G.; Soria, C.; Carelli, R. (2014). Sliding mode control for trajectory tracking of a non- holonomic mobile robot using adaptive neural networks, *Control Engineering and Applied Informatics*, 16, 12–21, 2014.

[18] Sabahi, F. (2018). Introducing validity into self-organizing fuzzy neural network applied to impedance force control, *Fuzzy Sets and Systems*, 337, 113–127, 2018.

[19] Selmic R. R.; Lewis, F. L. (2000). Deadzone compensation in motion control systems using neural networks, *IEEE Transactions on Automatic Control*, 45, 602–613, 2000.

[20] Slotine J. J. E.; Li, W. (1991). *Applied Nonlinear Control*, Prentice-Hall, Hoboken, NJ, 1991.

[21] Tsai, C.-H.; Chuang, H.-T. (2004). Deadzone compensation based on constrained RBF neural network, *Journal of the Franklin Institute*, 341, 361–374, 2004.

[22] Vrkalovic, S.; Lunca, E.; Borlea, I. (2018). Model-free sliding mode and fuzzy controllers for reverse osmosis desalination plants, *International journal of Artificial intelligence*, 16, 208–222, 2018.

[23] Wai, R.; Muthusamy, R. (2013). Fuzzy-Neural-Network Inherited Sliding-Mode Control for Robot Manipulator Including Actuator Dynamics, *IEEE Transactions on Neural Networks and Learning Systems*, 24, 274–287, 2013.

[24] Wai, R.; Muthusamy, R. (2014). Design of Fuzzy-Neural-Network-Inherited Backstepping Control for Robot Manipulator Including Actuator Dynamics, *IEEE Transactions on Fuzzy Systems*, 22, 709–722, 2014.

[25] Wen, C.; Zhou, J.; Liu, Z.; Su, H. (2011). Robust Adaptive Control of Uncertain Nonlinear Systems in the Presence of Input Saturation and External Disturbance, *IEEE Transactions on Automatic Control*, 56, 1672–1678, 2011.

[26] Wu, Y.; Huang, R.; Li, X.; Liu, S. (2019). Adaptive neural network control of uncertain robotic manipulators with external disturbance and time-varying output constraints, *Neurocomputing*, 323, 108–116, 2019.

[27] Ying, H. (2005). Structure and stability analysis of general mamdani fuzzy dynamic Models, *International journal of intelligent systems*, 20, 103–125, 2005.

[28] Ying, Z.; Changyun, W.; Yeng Chai, S. (2000). Adaptive backstepping control design for systems with unknown high-frequency gain, *IEEE Transactions on Automatic Control*, 45, 2350–2354, 2000.

[29] Zhou, D.; Shi, M.; Chao, F.; Lin, C. M.; Yang, L.; Shang, C.; Zhou, C. (2018). Use of human gestures for controlling a mobile robot via adaptive CMAC network and fuzzy logic controller, *Neurocomputing*, 282, 218–231, 2018.