Decentralized RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles

Regular Paper

Rezoug Amar [1,2,*], Hamerlain Mustapha[1] and Tadjine Mohamed[2]

1 Robotic and Manufactory Department, Center for Development of Advanced Technologies, Algiers, Algeria
2 Process Control Laboratory, Automatic Department, Polytechnic National School of Algiers, Algiers, Algeria
* Corresponding author E-mail: arezoug@cdta.dz

Abstract In the few last years, investigations in neural networks, fuzzy systems and their combinations become attractive research areas for modeling and controlling of uncertain systems. In this paper, we propose a new robust controller based on the integration of a Radial Base Function Neural Network (RBFNN) and an Interval Type-2 Fuzzy Logic (IT2FLC) for robot manipulator actuated by pneumatic artificial muscles (PAM). The proposed approach was synthesized for each joint using Sliding Mode Control (SMC) and named Radial Base Function Neural Network Type-2 Fuzzy Sliding Mode Control (RBFT2FSMC). Several objectives can be accomplished using this control scheme such as: avoiding difficult modeling, attenuating the chattering effect of the SMC, reducing the rules number of the fuzzy control, guaranteeing the stability and the robustness of the system, and finally handling the uncertainties of the system. The proposed control approach is synthesized and the stability of the robot using this controller was analyzed using Lyapunov theory. In order to demonstrate the efficiency of the RBFT2FSMC compared to other control technique, simulations experiments were performed using linear model with parameters uncertainties obtained after identification stage. Results show the superiority of the proposed approach compared to RBFNN Type-1 Fuzzy SMC. Finally, an experimental study of the proposed approach was presented using 2-DOF robot.

Keywords Artificial Muscle, Radial Basis Function Neural Network, Type-2 Fuzzy Logic, Sliding Mode Control, Lyapunov Stability

1. Introduction

It is well known that Sliding Mode Control (SMC) presents higher robustness for systems which have modeling uncertainty, parameters variations and/or external disturbances. Basically, SMC consists of two steps: a driving step that forces the system trajectory to reach a stable manifold, and sliding mode that assures the desired dynamics of the controlled system [1]. However, SMC has the chattering drawback [1]. In the other hand,

Fuzzy Logic (FL) has been found to be an effective approach to deal with complicated, ill-defined and dynamic processes, which are poorly mathematically modeled [2]. In the few last years, a new approach of FL called Type-2 Fuzzy Logic (T2FL) has been appeared. First introduced by L. Zadeh in 1975 [3-4], T2FL presents a generalization of the classical fuzzy logic called type-1. Type-2 fuzzy sets are characterized by membership grades that are themselves fuzzy [5]. The membership functions (MFs) of a type-2 fuzzy set have a footprint of uncertainty (FOU) which represents the uncertainties in both shape and position of the type-1 fuzzy set [3-6]. An Interval type-2 fuzzy set can represent and handle uncertain information effectively. That is, interval type-2 fuzzy sets let us model and minimize the effects of uncertainties in rule-base interval type-2 fuzzy logic systems (IT2FLS) [30]. Type-2 control is applied to many systems: mobile robot [34] manipulator robot [5], etc; in addition, type-2 was manifested its superiority compared to type-1 in several work [3][32]. However, the major drawback of the FL system is lack of adequate analysis and design techniques [28]; precisely type-2 fuzzy logic is used generally without any guarantee of stability.

In order to overcome the SMC and the FL drawbacks, a new field of control which consists of their combination is introduced. This control mode is called Fuzzy sliding mode control (FSMC). Consequently, FSMC can achieve various objectives such as: (1) avoiding the precise modeling of the studied system, (2) guaranteeing the stability and the robustness of the studied systems, (3) decreasing the chattering effect caused by the discontinuous part of SMC, (4) reducing significantly the rules number of FL part, and in addition (5) handling the uncertainty of the studied system in the case of IT2FC [7]. Nevertheless, this technique needs sufficient knowledge about system dynamics, which is not always available. Neural Networks (NNs) are universal approximators that can be considered as the best solution to model the unknown parts in systems.

On the other hand, the artificial muscle is a pneumatic actuator that was originally developed through the 1930s–1940s in expansive mode form and between 1950s–1960s in contraction mode form [8]. In its last form, the Pneumatic Artificial Muscle (PAM) has been created by Dr. J.L. McKibben [9] for artificial limb of handicapped peoples. After that and over several years, some precision robotic applications based on the PAM have been realized [10-13]. Basically, PAM is tubular pull-actuators with a special fibber arrangement. The fibbers form a diamond pattern in a three-dimensional mesh structure, which allows the actuator to contract when the internal pressure of the hose is increased [12]. PAM based robots have several desirable characteristics, such as high power-to-weight ratio, high power-to-volume ratio, inherent compliance [9, 14], use soft energy sources and can absorb shocks during the impacts, etc. However, because of the time varying inertia, hysteresis and joints friction [18], PAM robot's belongs to the class of highly nonlinear systems, where perfect knowledge of their exact models are very delicate; consequently it become extremely hard to control. For these challenges currently, the main objectives in pneumatic muscle applications are to have robust and best control. Through several years, researchers have been attempted various techniques to control robots manipulator with artificial muscles, we can mention: Sliding Mode Control [10-11, 15-16] in these approaches, the chattering phenomena has been attenuated usually using a saturation function. This last reduces the robustness of the system compared to the external disturbance. Higher Order Sliding Mode Control (HOMS) in [17], although the HOSM is able to eliminate the chattering, but it is very sensitive to noise and needs the information about the sliding variable and its derivatives, which are not always available. Neural Network in [18] and the combination of PID and Neural Network [19], these techniques have problem of slow learning and local minimal convergence. Self-Organizing Fuzzy Sliding Mode Controller [2] controller has a relatively complicated architecture, which requires sophisticated material for real applications.

In this paper, we propose a new adaptive and robust control approach consists on the combination of the Radial Base Function Neural Network (RBFNN), which is used to approximate the equivalent control of the SMC and an interval type-2 fuzzy system to attenuate the chattering effect caused by the hitting control part of the SMC for robot arm actuated by PAM. Stability of the robot in closed loop is guaranteed using the Lyapunov theorem. In order to proof the effectiveness of the proposed approach, simulation and real experimental studies of the control scheme are realized using 2-DOF robot manipulator actuated by PAM. Our control scheme allows us to avoid the nonlinear modeling problems, to guarantee the stability of the robot in closed loop, to provide the robustness and to obtain best performances.

For the best presentation of this work, the paper is organized as following: in section two, we present the approximation of the equivalent control part of sliding mode by RBFNN and the stability of the new controller for n-link robot manipulator. Section three, presents the use of type-2 fuzzy approach to replace the hitting control part of sliding mode to reduce the chattering and to handle the uncertainty of the system. In section four, we present the robot joints identification and the simulations of the proposed approach and its type-1 extension under linear model with parameters variation. Section five presents the experimentation of the proposed controller and finally the paper is finished by conclusion.

2. RBFT2FSMC Design for *n*-link Manipulator Robot

The general standard dynamic model of *n*-link robot manipulator is given as [20]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + T_d = \tau \tag{1}$$

Where $M(q) \in R^{nXn}$ is the inertial symmetric positive matrix, e.g. non-singular and bounded by $m_{min}\|x\|^2 \leq x^T M x \leq m_{max}\|x\|^2 \quad \forall x \in R^n$, where m_{max} and m_{mim} are minimum and maximum eigenvalues. $C(q,\dot{q}) \in R^{n\times n}$ is the vector of centrifugal; coriolis and friction terms. $G(q) \in R^{n\times 1}$ is the vector of gravity terms, $q \in R^{n\times 1}$, $\dot{q} \in R^{n\times 1}$ and $\ddot{q} \in R^{n\times 1}$ are the position, velocity and acceleration vectors, respectively. $\tau \in R^n$ is the torque input vector, and $T_d \in R^n$ are the unknown bounded disturbances i.e. $\|T_d\| < \beta_d$, with β_d is a positive vector.

The general equation (1) of *n*-link robot manipulator is not always available because of its difficult determination and imprecise knowledge of its parameters. In addition, performances realized using SMC depend on the precision of the modeling. Because of time varying inertia, hysteresis and joint friction, this problem is strongly hard in the case of robot with artificial pneumatic muscles. SMC has been studied in [29] and simulations results were obtained. However, because of its complication, this control law is extremely delicate in real application to robots. For this reason and in order to cure the cited problems, we will solicit the Soft Computing techniques such as: Radial Basis Function Neural Network, Type-2 Fuzzy Logic, which are combined with SMC to synthesize a robust controller for robot actuated by pneumatic artificial muscles.

2.1. Sliding Mode Control

In order to simplify the establishment of the proposed controller, the interactions between all links of the robot can be assumed as permanent perturbations, which will be compensated by the application of SMC, thus as in [21], [28] and others the robot will be controlled by decentralized scheme. Using this assumption, eq. (1) can be rewritten in the error state space by *n*-SISO non-linear subsystems with uncertainties as:

$$\begin{cases} \dot{e}_{1i} = e_{2i} \\ \dot{e}_{2i} = \tilde{f}_i(e,t) + \tilde{b}_i(e,t)\cdot u_i + \xi_i \end{cases} \tag{2}$$

Where: $i=1...n$ is the number of joints, $e_{1i} = q_{1i} - q_{1di}$, $\tilde{f}_i(e,t)$ is a nonlinear continuous function with unknown uncertainty whose upper bound is known as

$f_i(e,t)_{min} < \tilde{f}_i(e,t) < f_i(e,t)_{max}$. $\tilde{b}_i(e,t)$ is a nonlinear function with lower bound as: $0 < b_i(e,t)_{min} < \tilde{b}_i(e,t)$. ξ_i is the bounded disturbance, $|\xi_i| < \xi_{max\,i}$ and u_i is the control law.

In general, the design of SMC requires two steps: the choice of sliding surface and the synthesis of the control law. In classical SMC, the sliding surface for system (2) is usually chosen as a linear function:

$$S_i = e_{1i} + \lambda_i \cdot e_{2i} = \Lambda_i e_i \tag{3}$$

Where: $\Lambda_i = [1, \lambda_{1i}]$ are constant positive vectors and $e_i = [e_{1i}, e_{2i}]^T$ are the errors in stat space domain.

SMC have two terms, which are the equivalent control and the switching control while:

$$u_i = u_{eqi} + u_{hi} \tag{4}$$

For systems with uncertainties, the equivalent control can be written as [22]:

$$u_{eqi} = u_{eqi}^* + \Delta u_{eqi} \tag{5}$$

With: $u_{eqi}^* = -[\nabla S_i b_i]^{-1}[f_i]$, $\nabla u_{eqi} = -[\nabla S_i b_i]^{-1}[\nabla S_i \xi_i]$ and $\nabla S_i = \dfrac{\partial S_i}{\partial q_i}$

The equivalent control of SMC is calculated using system (2) and it is synthesized when $\dot{S}_i = 0$, thus in our case it is given as:

$$u_{eqi} = \tilde{b}_i^{-1}(\Lambda_{mi}^T e_i - \tilde{f}_i(e,t) - \xi_i) \tag{6}$$

With $\dot{S}_i$ can be written as:

$$\dot{S}_i = \Lambda_i \dot{e}_i = \Lambda_{mi}^T e_i - \tilde{f}_i(e,t) - \tilde{b}_i(e,t)\cdot u_i - \xi_i \tag{7}$$

With $\Lambda_i = [1, \lambda_{1i}]$ and $\Lambda_{mi} = [0, \lambda_{1i}]$

u_{hi} in (4) describes the switching control part also called hitting control which is given by:

$$u_{hi} = \tilde{b}_i^{-1} k_i \, sign(S_i) \tag{8}$$

With: $k_i > 0$ and $\tilde{b}_i = \sqrt{b_{max\,i} b_{min\,i}}$ = constant>0 [8].

Because of strong nonlinearity of robots with PAM, the synthesis of the equivalent control part is very delicate and the obtained performances using that are depending to the exactitude of the robot model. In addition, in our case the equivalent control Eq. (6) contains uncalculated uncertainty. In order to solve these problems, our choice

3

www.manaraa.com

is fell on the use of the Radial Basis Function Neural Network (RBFNN) to approximate that. So, this approach is able to give the equivalent control part, without any knowledge about the robot model.

2.2 Estimation of Equivalent Control Using the RBFNN

2.2.1 RBFNN architecture

Traditional back propagation NN algorithms have the disadvantages of slow learning and local minimal convergence. The RBFNN is a special type of feed-forward artificial neural network, which contain single hidden layer of neurons usually doted by Gaussian activation functions. The key property of this scheme is that the weights of the neural network are estimated adaptively [24]. RBFNN can be considered as the best candidate to solve the classical neural network problems, which have been successfully used for function approximation [23], dynamic modeling [23] and for control of systems [23-25], etc. Figure 1 shows the architecture of the used RBFNN, which is composed by three layers. The input layer contains the error in angular position and its derivative for each joint. The middle layer called hidden layer is composed by m neurons, which are doted by Gaussian function for each neuron. The output layer gives the estimation of the equivalent control. It is very interesting to note that it is possible to use a MIMO RBFNN for the control of manipulator robot as reported in [20]. However, the use of this architecture has some disadvantages. First, the complexity and the misunderstanding of the MIMO RBFNN dynamics increases more and more with the increasing of the number of links to be controlled. Second, the time of computing will be very important, with the increasing of the number of joints. In addition, the adjustment of the MIMO parameters is delicate, because it is required to find the best combination for several joints at same time.
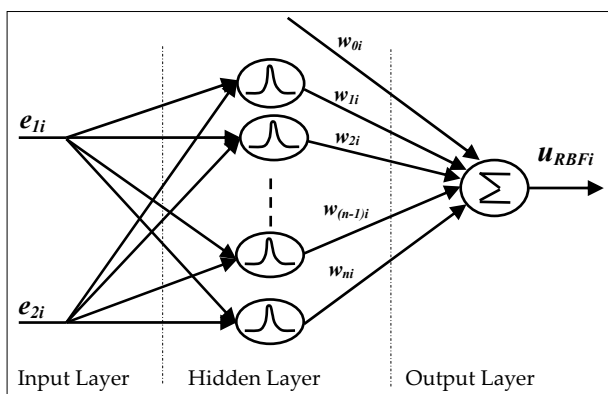

Figure 1. Structure of a RBF network.

The inputs of the used RBFNN are the error in angular position and its derivative for each joint: $e_i = [e_{1i}, e_{2i}]^T$. Each neuron of the hidden layer is characterized by the Gaussian Activation Function (GAF) which is given by:

$$h_{ji}(x) = e^{-(\frac{\|e_i - c_{ji}\|}{2.\sigma_{ji}^2})} \qquad (9)$$

Where c_{ji} is the centre of the GAF $j=1...m$, σ_{ji} is the variances of the GAF, in addition $\sigma_{ji} = [\sigma_{ji1}, \sigma_{ji2}]$ and $c_{ji} = [c_{ji1}, c_{ji2}]$ are vectors of scaling parameters present the width of the GAF, i is the number of joint. The output of the neuron peaks to one when the given sample has a zero distance from the centre of the neuron and decreases toward zero as the given sample strays further from the neuron, thus acting as a similarity measure or membership function [20]. The output control law of a network is computed as:

$$u_{RBFi} = \sum_{j=1}^{n} w_{ji} \cdot h_{ji} = W_i^T \cdot H_i \qquad (10)$$

Where: $W_i = [w_{0i}, w_{1i}, w_{2i} \cdots w_{ni}]$ with w_{ji} is the weight associated with the output of neuron j for the joint i. $H_i = [h_{0i}, h_{1i}, h_{2i} \cdots h_{ni}]$ with h_{ji} is the output of the neuron for a given data, which is a radial function of the distance between the neuron centre c_{ji} and the sample e_{ji} [23]. While:

$$\|e_i - c_{ji}\| = (e_{1i} - c_{ji2})^2 + (e_{2i} - c_{ji2})^2 \qquad (11)$$

2.2.2 the equivalent control estimation using RBFNN

Several training techniques have been successfully used for adapting of RBFNN parameters such as: gradient descends algorithm [20], k-mean algorithm for centre adjustment [23], synthesized from stability condition [24], etc. In this work, we choose to tune only the wield parameters w_{ij} for objective to minimize the computing time in the real experimentation. Using a specific scheme deduced during the stability analysis this parameters adaptation will presented in the following:

The RBFNN possesses great mapping ability and has a similar feature to the type-1 fuzzy system [20]. In this effect, we are able to use the demonstration given in [16, 23] as: If the input $x = [e, \dot{e}]^T$ is a compact set $T \in R^2$ [23], there exists a group of consequent parameter vector $W = W^* = [w_0, w_1, w_2 \cdots w_n]$ which satisfies the following condition [16]:

$$\underset{T \in R^2}{Sup} |u_{eq} - u_{BRF}^*| \le \varepsilon_0 \qquad (12)$$

With $\varepsilon_0 \ge 0$, as in [8], in our case we have chosen that:

$$u_{eqi} - u_{BRFi}^* = \rho_i \quad . \qquad (13)$$

Where: $u_{BRFi}^* = W_i^{*T} H_i$ with u_{BRFi}^* is the optimal approximation of $u_{eqi} \cdot \rho_i \in R$ and $|\rho_i| \ge \varepsilon_{0i}$.

Taken into account the above assumptions after estimation, the control law (4) becomes:

$$u_i = u_{RBFi} + u_{hi} \tag{14}$$

In the other hand, the equation of perturbation can be obtained from eq. (6) as following:

$$\xi_i = \Lambda_{mi}^T e_i - \tilde{f}_i(e,t) - b_i \cdot u_{eqi} \tag{15}$$

Substituting equations (14) and (15) in equation (2) we obtain:

$$\begin{cases} \dot{e}_{1i} = e_{2i} \\ \dot{e}_{2i} = \tilde{b}_i \cdot (u_{eqi} - u_{RBFi} - u_{hi}) - \Lambda_{mi}^T e_i \end{cases} \tag{16}$$

Equation (16) can be rewritten in the following form:

$$\dot{e}_i = A_i \cdot e_i + B_i . u_i \tag{17}$$

With: $\dot{e}_i = [\dot{e}_{1i}, \dot{e}_{2i}]^T$, $A_i = \begin{bmatrix} 0 & 1 \\ 0 & -\lambda_i \end{bmatrix}$, $B_i = \begin{bmatrix} 0 \\ b_i \end{bmatrix}$ with u_i become:

$$u_i = u_{eqi} - u_{RBFi} - u_{hi} \tag{18}$$

u_{eqi} has a best approximation where $u_{eqi} - u_{BRFi}^* = \rho_i$ thus [12]:

$$u_{RBFi} - u_{RBFi}^* = (W_i - W_i^*)^T . H_i = \tilde{W}_i^T \cdot H_i \tag{19}$$

With: $\tilde{W}_i^T = (W_i - W_i^*)^T$

2.2.3 Stability analysis and RBFNN parameters tuning

Define a Lyapunov function candidate for each joint as:

$$V_i = \frac{1}{2} S_i^2 + \frac{b_i}{\delta_i} \tilde{W}_i^T \tilde{W}_i \tag{20}$$

Where: $\tilde{W}_i = (W_i - W_i^*)$ and δ_i is the positive constant.

The derivative of the Lyapunov function V_i is given by:

$$\begin{aligned} \dot{V}_i &= S_i \dot{S}_i + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\ &= S_i \Lambda_i^T \dot{e}_i + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\ &= S_i \Lambda_i^T (A_i \cdot e_i + B_i . u_{eqi} - u_{RBFi} - u_{hi}) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\ &= S_i \Lambda_i^T A_i \cdot e_i + S_i \Lambda_i^T B_i . (u_{eqi} - u_{RBFi} - u_{hi}) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\ &= S_i \Lambda_i^T A_i \cdot e_i + S_i \Lambda_i^T B_i . (u_{RBFi}^* - u_{RBFi} - u_{hi} - \rho_i) + \frac{b_i}{\delta_i} \tilde{W}_i^T \dot{\tilde{W}}_i \\ &= -\frac{b_i}{\delta_i} \tilde{W}_i^T (\delta_i S_i H_i - \dot{\tilde{W}}_i) - b_i S_i \rho_i - b_i S_i u_{hi} \end{aligned} \tag{21}$$

Choosing the adaptive law:

$$\dot{\tilde{W}}_i = \delta_i S_i H_i \tag{22}$$

From (22) the last equation of (21) becomes:

$$\dot{V}_i = -b_i S_i \varepsilon_i - b_i S_i u_{hi} \tag{23}$$

In order to obtain the stability in closed loop, the hitting control u_{hi} is chosen with a compensator part as:

$$u_{hi} = k_{0i} S_i + k_{1i} sign(S_i) \tag{24}$$

With: $k_i = b_i k_{oi} |S_i| + b_i k_{1i}$

Using (24) in (23) we obtain:

$$\begin{aligned} \dot{V}_i &= -b_i S_i \rho_i - b_i k_{0i} S_i^2 - b_i k_{1i} S_i sign(S_i) \\ &= -b_i S_i \rho_i - b_i k_{0i} S_i^2 - b_i k_{1i} |S_i| \leq 0 \end{aligned} \tag{25}$$

With: $k_{1i} > |\rho_i|$

Finally, the control law becomes:

$$u_i = u_{RBFi} + k_{0i} S_i + k_{1i} sign(S_i) \tag{26}$$

From eq. (22) and (25) the RBFNN synaptic weights are tuned and the stability of the system has been guaranteed using the control law (26). However, the hitting control engenders a chattering effect, well known as a principal drawback of the SMC. In order to maintain the stability and to attenuate the chattering affects, we will opt to use a type-2 fuzzy logic to replace the hitting control part of the equation (26). This technique is also used for handling the uncertainty of the robot.

3. Interval Type-2 Fuzzy Control for Chattering Attenuation and uncertainty handling

3.1. type-2 fuzzy logic   .

Type-1 and type-2 fuzzy logic are mainly similar. However, there exist two essential differences between them which are: the membership functions forms and the output processor. Then, the interval type-2 fuzzy controller consists by: fuzzifier, inference engine, rules base, type reduction and a defuzzifier, where:

3.2.1 Fuzzyfier

The fuzzifier maps a crisp input into a type-2 fuzzy set.

3.2.2 Rules base

The structure of rules in type-1 and type-2 FLS is the same, but in the last case the antecedents and/or the consequences should be represented by type-2 fuzzy sets

www.intechopen.com

Rezoug Amar, Hamerlain Mustapha and Tadjine Mohamed: Decentralized RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles

5

www.manaraa.com

[5-6]. The general IF-THEN rules in the type-2 FLS are given as:

$$R^l : If\ e_1\ is\ F_1^l\ and\cdots and\ e_P\ is\ F_P^l\ Then\ y\ is\ \tilde{G}_1^l \qquad (27)$$

With: $l = 1...M$

3.2.3 Fuzzy Inference System

type-2 fuzzy inference systems used in the circumstance is too uncertain to determine exact membership grades [33]. From the IT2FS point of view, the fuzzy rule in (27) can be written as [5]:

$$F(e) = [\underline{f}^l(e), \overline{f}^l(e)] \qquad (28)$$

With :

$$\overline{f}^l = \overline{\mu}_{F_1^l}(e_1) * \cdots * \overline{\mu}_{F_p^l}(e_p)$$

$$\underline{f}^l(e) = \underline{\mu}_{F_1^l}(e_1) * \cdots * \underline{\mu}_{F_p^l}(e_p) \qquad (29)$$

Where * denotes the product operation.

3.2.4 Type-reducer

This step is the crucial part of the type-2 fuzzy logic systems. The output of the type-reducer generates a type-1 fuzzy set, which is then converted in a crisp output through the defuzzifier. For majority of type-2 fuzzy controllers they use the Center of Sets type reduction technique which is given by $U_{\cos}$ and expressed as [32][33] :

$$U_{\cos} = [y_l, y_r] = \int_{y^1 \in [y_l^M, y_r^M]} \cdots \int_{y^M \in [y_l^M, y_r^M]} \int_{f^1 \in [\underline{f}^1, \overline{f}^1]} \cdots$$
$$\cdots \int_{f^M \in [\underline{f}^M, \overline{f}^M]} 1 / \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i} \qquad (30)$$

The interval output set is determined by its left-most point y_l and its right-most point y_r. To compute y_l and y_r, the Karnik–Mendel iterative procedure is needed [5-6, 30-34]. The general equation of the center of sets technique given (30), which corresponds to the Centroid of the type-2 interval consequent set $G_{\tilde{G}^i}$ [5]:

$$G_{\tilde{G}^i} = \int_{\theta_1 \in J_{y1}} \cdots \int_{\theta_N \in J_{yN}} 1 / \frac{\sum_{i=1}^N y_i \theta_i}{\sum_{i=1}^N \theta_i} = [y_l^i, y_r^i] \qquad (31)$$

With: $y_l = \frac{\sum_{i=1}^M f_l^i y^i}{\sum_{i=1}^M f_l^i}$ and $y_r = \frac{\sum_{i=1}^M f_r^i y^i}{\sum_{i=1}^M f_r^i}$

3.2.5 Defuzzifier

consists on the average of y_r and y_l, so the defuzzified output of an interval singleton type-2 FLS is [5] [6]:

$$y = \frac{y_r + y_l}{2} \qquad (32)$$

3.3 Interval Type-2 Fuzzy Controller Based on Sliding Mode Control Design

In order to attenuate the chattering effect and handle the uncertainty of the robot with PAM, we choose to use a type-2 fuzzy controller with single input and single output for each joint. Then, the input of the controller is the sliding surface ($S_i = e_{1i} + \lambda_i.e_{2i}$) and the output replace ($k_i$sign(S_i)) term which is substituted in the global control law eq. (26). All the membership functions of the fuzzy input variable are chosen to be fully overlapped, triangular, trapezoidal and symmetric for all upper and lower MFs. The used labels of the fuzzy variable (surface) are: {negative surface (*Ns*), zero surfaces (*Zs*) and positive surface (*Ps*)}. There are three Type-1 MFs (u_N Negative, u_z Zero, and u_p Positive) for the output. We choose Type-1 (MFs) for objective to minimize the computing time of type-2 fuzzy controller. Figure 2 presents the input/output MFs distributed on discourse universes and table 1 presents the rules base:

Figure 2. Input and output membership functions.

The inference engine is the core of the fuzzy system which handles the way in which rules are combined. We used the general eq. (29) to calculate the inference step. In this paper, we have used Centroid type-reducer method given by the eq. (30). The output of type-reducer step must be defuzzified using eq. (32). The crisp value from defuzzifier should be included in the global control eq. (26).

S	Ns	Zs	Ps
$k$sign(S)	U_n	U_z	U_p

Table 1. Rules base

4. RBFT2FSMC Application

4.1 Robot description

In order to make in evidence the proposed approach, simulations and experimentation studies have been realized using 2-degrees of freedom robot depicted in

www.manaraa.com

figure 3.a. This engine is actuated by pneumatic FESTO muscles. This system is a Robosoft product dedicated to research and development actions [26]. We chose this robot for the reason that is suitable for domestic applications, each joint of the robot has 0.819 m long, and 4 Kg weight. As in skeletal muscles, two actuators are needed to be coupled in order to generate a bidirectional motion, one for each direction figure 3.b. First, each muscle was alimented by an initial pressure P_0=2 Bar. Then, the mechanical motion can be obtained by modifying the pressure ΔP in each muscle.
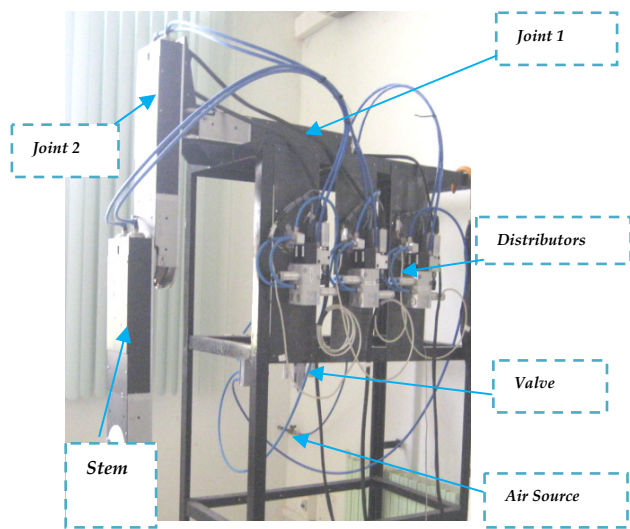
(a)
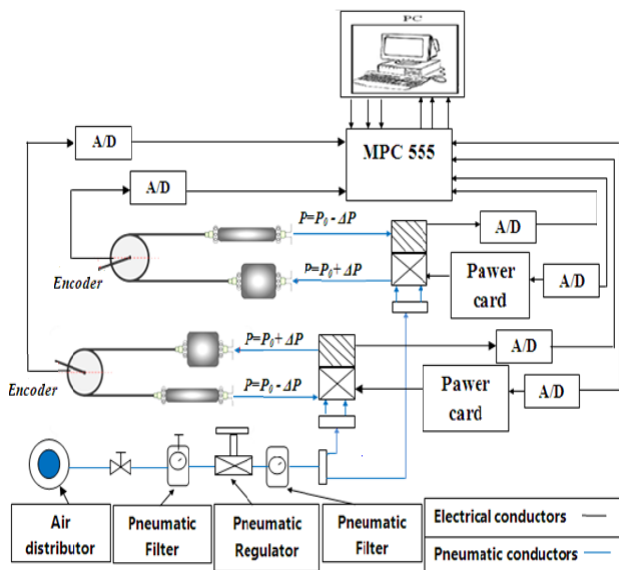
(b)

Figure 3. (a) 2-DOF robot actuated by the muscles (b) Detail architecture of the 2-DOF robot.

4.2 Control challenges

Particularly, in robots with artificial muscles, problems of time variance, hysteresis and nonlinearity of pneumatic systems have made it difficult to realize precise position control with high speed [19]. Furthermore, the relationship between length, pressure and force of the pneumatic muscles is very complex. The contraction of the muscles is limited consequently the work space of the robot is limited. For the robot with 2-DOF there exist many other difficulties in the control of our robot arm. For example: the pressure in each muscle is not identical. In all robot parts there exist a minor quantity of escaping air. The antagonistic system is affected by the temperature and volume variations in each muscle. The flow of each distributor is supposed to be identical, but really they can have a little difference from one to the other. The delay induced by the length of air conductor between the distributor and the muscle. The characteristics of a muscle change slightly when the number of operating cycle increases, these phenomenon's will change the characteristics of the system behavior, i.e. the parameters of the system are not exactly known and the modeling errors and parameters uncertainties of the model may be appeared. Because of the environment conditions are continuously changing, the pressure control signal is infected and really variable. To show some difficulties of the used robot, we inject many identical step inputs signal with deferent values to the joints of the robot. The responses on angular position for each joint are given in figure 4.
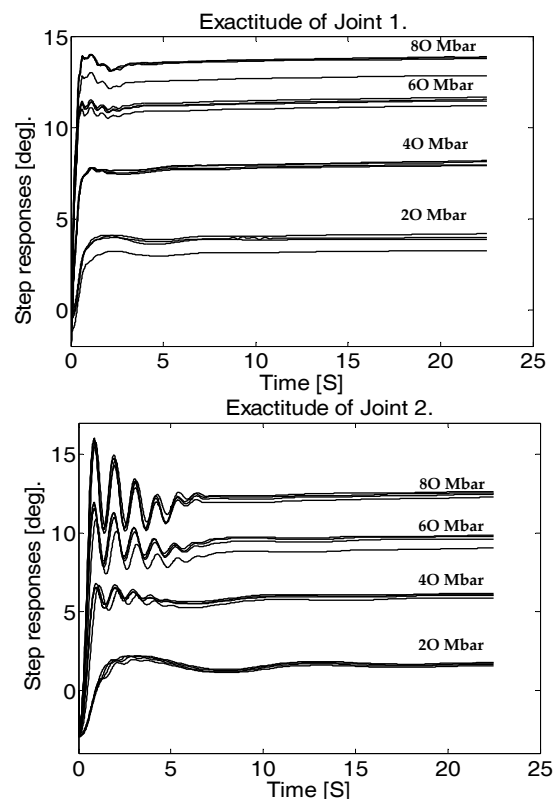
Figure 4. Responses of joints for deferent steps.

4.3 Robot joints identification

From results shown in figure 4, we can observe that the robot does not give similar curvatures for successive identical inputs. We can deduce from these results the existence of uncertainties in this engine. In addition, the system manifests the delay time phenomena and an important gravitational effect, which is appeared in the beginning of the robot responses.

As in [11], [17], [27], and others, in order to make a simulation step, we have used a linear model around a desirable point to obtain a semi exact model for each joint, the decoupled model can be also justified by the capability of these actuators to absorb the little stokes caused by the joints coupling. The obtained model after identification will be used in intermediate step which help us to examine our controller compared to other technique. In order to determine the best linear model, the dynamic behavior of the robot joints are characterized by Pseudo Random Binary Sequence (PRBS) inputs [27]. Where PRBS is a popular signal used for the identification of systems which is characterized by its large spectrum of frequencies. The adequate PRBS is injected and resulting joint angular displacement values output responses are saved. The identification has been realized around desirable point of 14 degree for the joint 1 and 10 degree for joint 2. The used signals and the identification results are presented in figure 5. The system (2) after linearization can be written as:

$$\begin{cases} e_{1i} = e_{2i} \\ e_{2i} = -\tilde{a}_{0i}e_{1i} - \tilde{a}_{1i}e_{2i} - \tilde{b}_i u_i \end{cases} \quad (33)$$

The corresponding polynomial parameters for each joint after identification are given by:

$$\tilde{A}_0 = diag\,(\tilde{a}_{01}, \tilde{a}_{02}) = A_0 + \Delta A_0$$
$$\tilde{A}_1 = diag\,(\tilde{a}_{11}, \tilde{a}_{12}) = A_1 + \Delta A_1$$
$$\tilde{B} = diag\,(\tilde{b}_1, \tilde{b}_2) = B + \Delta B \quad (34)$$

With:

$$A_0 = diag\,(a_{01}, a_{02}) = diag\,(7.8573, 16.2075)$$
$$A_1 = diag\,(a_{11}, a_{12}) = diag\,(5.6654, 2.9595)$$
$$B = diag\,(b_1, b_2) = diag\,(15.3217, 2.0194) \quad (35)$$

With the uncertainties have been chosen as:

$$\Delta A_0 \le |0.2 \cdot A_0 \cdot randn|$$
$$\Delta A_1 \le |0.2 \cdot A_1 \cdot randn|$$
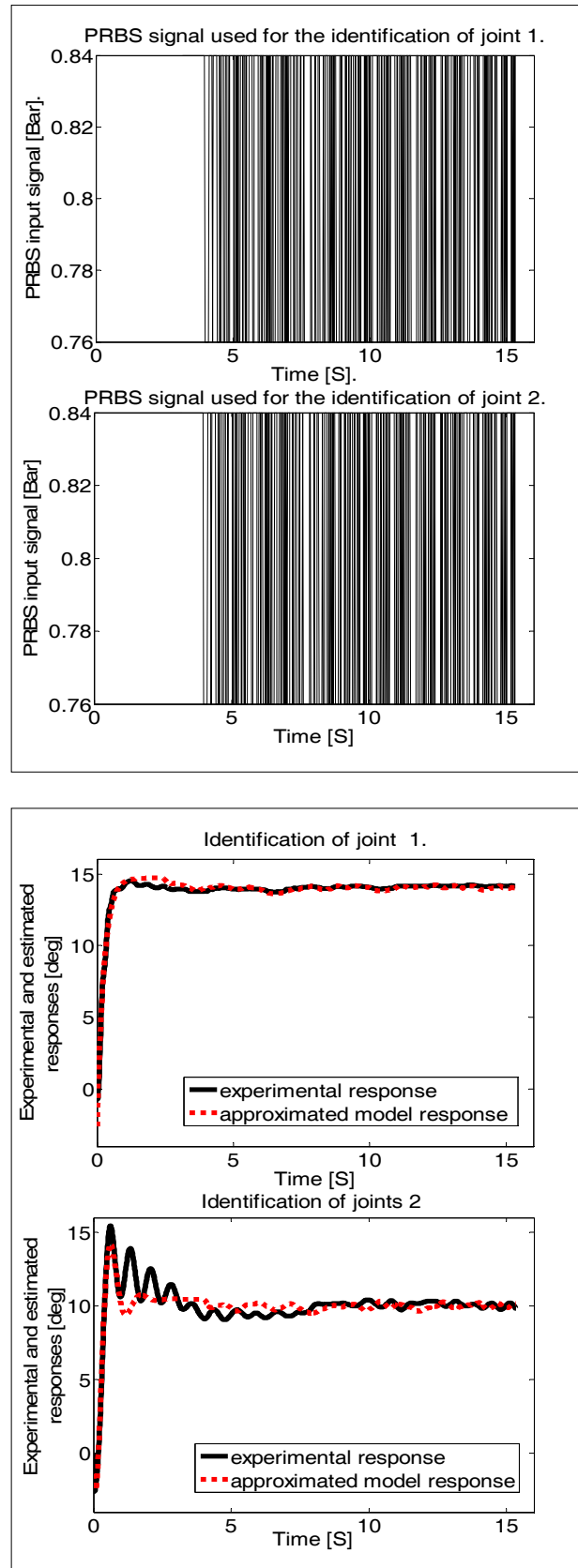$$\Delta B \le |0.2 \cdot B \cdot randn| \quad (36)$$
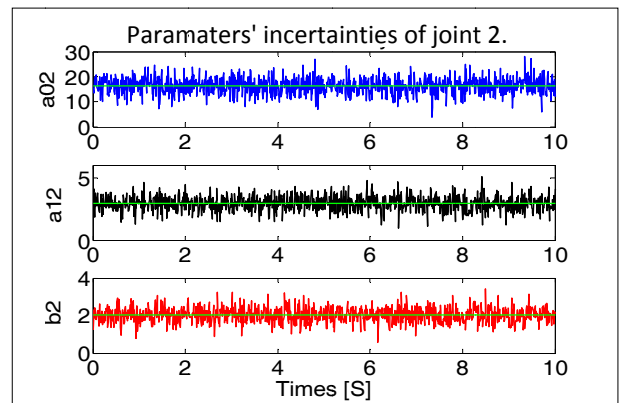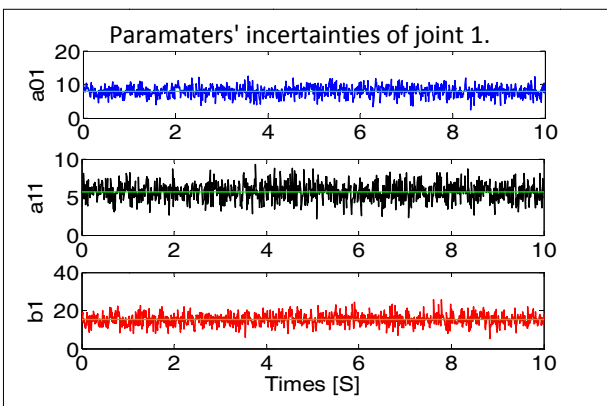

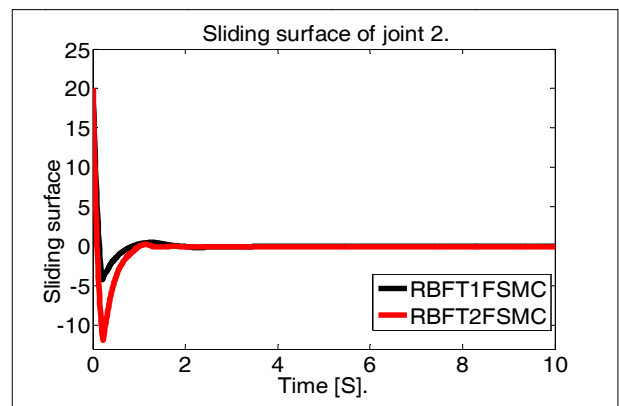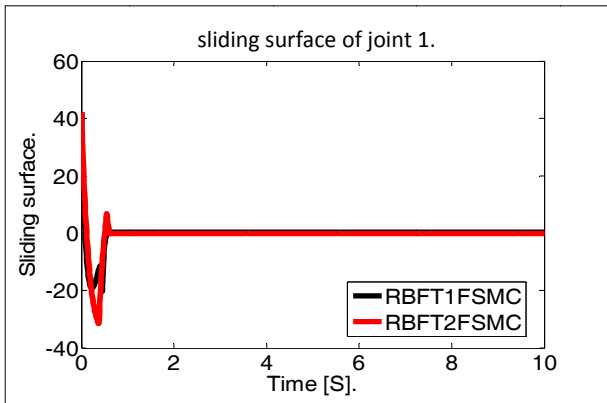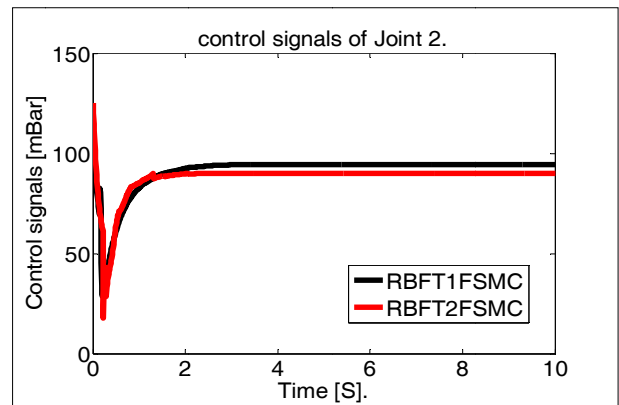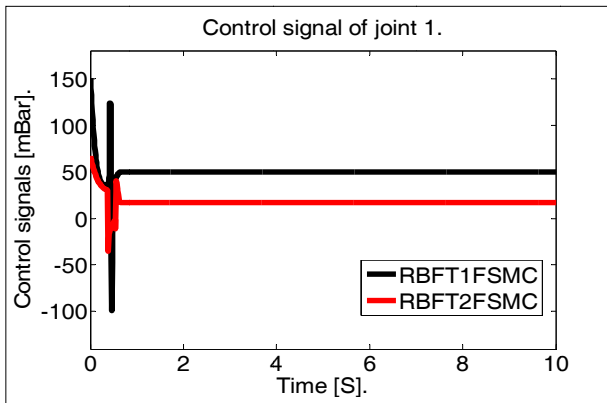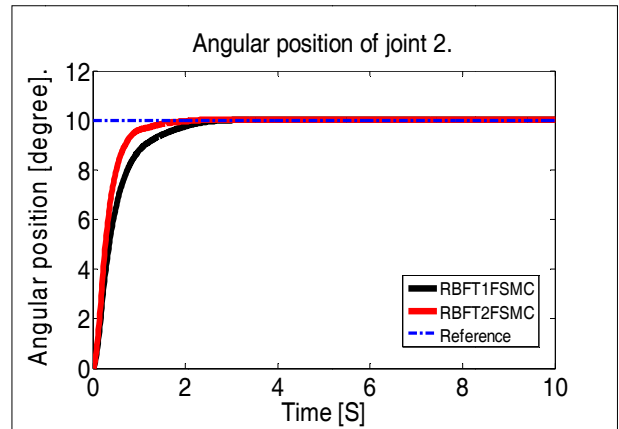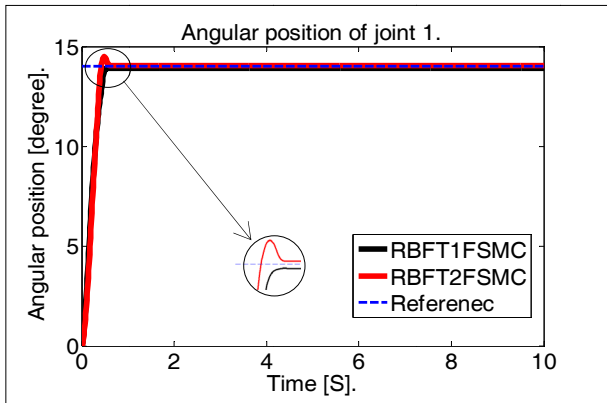**Figure 5.** Estimation of the joints linear models.

Figure 6. Simulation results for the application of RBFT2FSMC and RBFT1FSMC to joint 1.

Figure 7. Simulation results for the application of RBFT1FSMC and RBFT1FSMC to Joint 2.

www.intechopen.com

Rezoug Amar, Hamerlain Mustapha and Tadjine Mohamed: Decentralized RBFNN Type-2   9
Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles

www.manaraa.com

4.4 Simulation results

In this section, we will evaluate the proposed controllers by using MATLAB/SIMULINK. Simulations have been performed using linear model with 20% of the parameters uncertainties obtained after identification (equations 34 and 35). Figure 6 and 7 present the simulation results of RBFT1FSMC and RBFT2FSMC for joint 1 and 2 respectively. We choose to present as results: the angular position, the control signals, the evolution of the sliding surfaces and the parameters uncertainties for each joint. RBFT2FSMC is presented by red lines and RBFT1FSMC is presented by the black lines. Joint one initially is on zero degree and controlled to move one step to 14°. Similarly, for joint 2, we made the same simulations; where we moved it by 10° for initial position 0° in one step. All joints are simulated for a time of 10 second with simple time of 10ms in all cases. The simulation parameters are summarized in table 2. It is interesting to mention, in this case, that K_{Fuz_in} and K_{Fuz_out} in table 2 are the input and the output of the type-1 and type-2 fuzzy controller, respectively.

Method	RBFT1FSMC				
Simulation Parameters	λ	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	M
Joint1	3	0.7	0.1	40	6
Joint2	2.1	2	0.8	30	6
Method	RBFT2FSMC				
Simulation Parameters	λ	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	M
Joint1	3	0.5	0.1	25	6
Joint2	2	2	0.5	30	6

Table 2. Simulation Parameters.

For all results (joint 1 and 2) RBFT2FSMC presents better results compared to RBFT1FSMC. This is shown in the position responses, where we obtained an improvement of the rise time in the case of RBFT2FSMC compared to RBFT1FSMC. RBFT1FSMC presents a static error for the joint 1 (around 0.2 Degree). The control signal is important in the case of RBFT1FSMC compared to RBFT2FSMC. We have smooth control signals, which are explained by the effect of the fuzzy logic parts whether it is of type-1 or type-2. Seeing the sliding surface curvatures the chattering effect has been eliminated. The RBFT2FSMC presents better robustness and takes into account parameters variation compared to RBFT1FSMC.

4.5 Experimental results

In this section, experimental study is carried out to examine the feasibility and the validity of the proposed approach. Because RBFT2FSMC has presented better results compared to RBFT1FSMC in simulations, we have chosen to present only the experimental results of the RBFT2FSMC. RBFT2FSMC has been applied to 2-DOF robot arm presented in section 4.1. The software experiments have accomplished by C program on Pentium 4 PC. In order to handle efficiently such a distributed architecture, the software system is working under Linux with an RTAI module and SynDex interface. More details are given in [26]. The regulation mode was adopted to test the capability of the proposed controller to maintain the imposed performances and the robustness. As in simulation study, we aim that the 2-DOF robot attend the desired angular position of 14° for the joint one and 10° for the joint 2, with the initial position of 0° for all joints. We choose to present the joints angular positions, the control signals and the sliding surfaces to show the attenuation of the chattering effect. The used sampling time is 10 ms. For security reasons, the control pressure is saturated at +/- 200 mBar. The controller parameters must be adjusted for each joint are summarized in table 3, in addition $c_{ji} = \begin{bmatrix} \frac{\pi}{8} & \frac{\pi}{8} & \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{32} & \frac{\pi}{32} \\ \frac{\pi}{8} & \frac{\pi}{8} & \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{32} & \frac{\pi}{32} \end{bmatrix}$ and 0.2 for all w_{ij} of the for each joint, K_{Fuz_in} and K_{Fuz_out} present the input and the output gains of the type-2 fuzzy controller, respectively. The obtained results after application of RBFT2FSMC to the robot depicted in figure 3, are given in figure 8 (red color for joint 1 and blue color for joint 2).

Method	RBFT2FSMC				
Parameters	λi	K_{0i}	K_{Fuz_in}	K_{Fuz_out}	Mi
Joint1	7	0.5	0.1	28	6
Joint2	8.5	1.5	0.5	30	6

Table 3. Experimental parameters.

We observe from the position response curvature that joint one tracked adequately the imposed reference angle, with the existence of good static error (around 0.1 degree) and satisfactory dynamic error. We see also from the position responses are stable for all joints. We see from figures of the control signal like the one of the sliding surface the smooth signals and the chattering effect is attenuated. Because of the presence gravitational effect, the joints coupling and PAM characteristics, the control of the joint two is very delicate compared to the joint one. We observe from the position response of the RBFT2FSMC that the joint two tracked adequately the imposed angle, with the existence of a satisfactory static error (around 0.3 degree) and an acceptable dynamic error. The response of joint 2 has a little time delay and small oscillation around the desired positions which are caused by the permanent gravitational effect. On the other hand, we can see from the control curvature a smooth signal. Sliding surface presents a very considerable attenuation of the chattering effect.
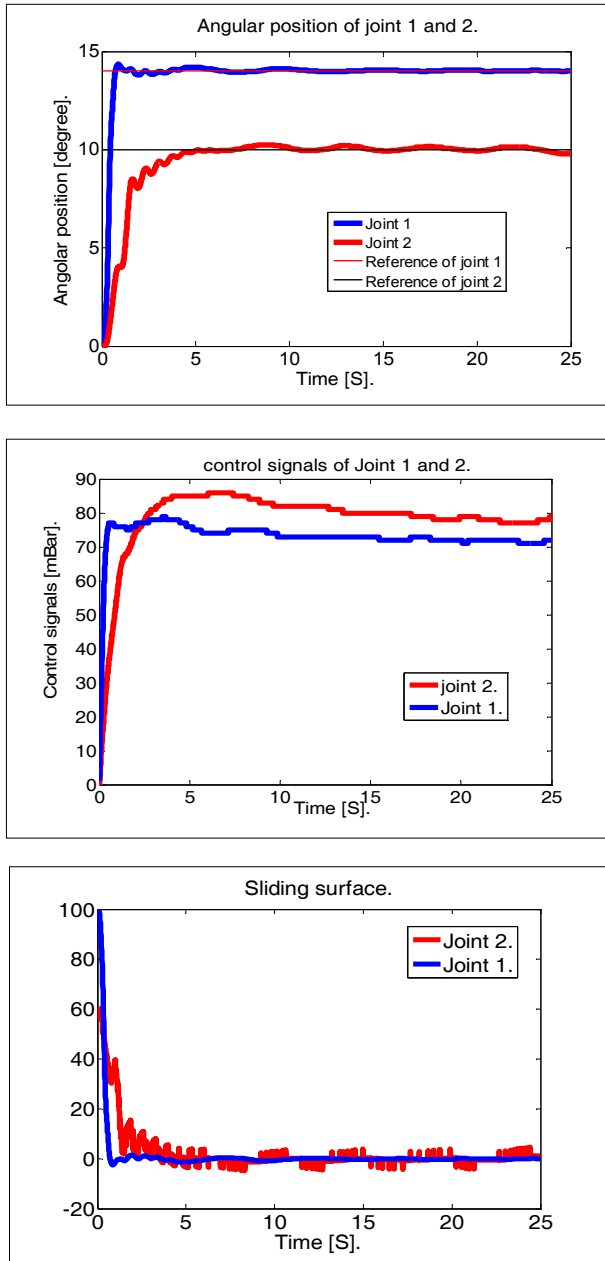
Figure 8. Experimental results of the application of RBFT2FSMC to the joints, 1 &2 of the robot.

We remark from tables 2 and 3 the deference between simulations and experimentation parameters which is justified by the real circumstance of the experimentation. The used energy to control the joint 2 is important compared to the one of joint 1 this is seen in the control signals, which is justified by the difficult control of joint 2 compared to joint 1. The experimental results suggest that the RBFT2FSMC strategy is working effectively and satisfactorily in real-time and is sufficiently robust, successful and stable despite of the presence of uncertain, nonlinearity and coupling of the real robot joints.

5. Conclusion

In this study, a new robust approach called Radial Base Function Network Type-2 Fuzzy Sliding Mode Control (RBFT2FSMC) has been proposed for robot manipulator actuated by artificial muscles. The sliding mode control is used to assure the stability of the robot. RBF network is used for the estimation of the unknown equivalent control of the SMC. IT2FC replaces the hitting control to attenuate the chattering and to handle the uncertainties of the robot with a minimal number of rules. The proposed controller does not require the robot dynamics, thus the control can be achieved without the robot model. In simulation, RBFT2FSMC presents better performances and more robustness compared to RBFT1FSMC. RBFT2FSMC was implemented in real time on the 2- DOF arm robot. The experimental results show that not only a good tracking performance has been obtained by applying the RBFT2FSMC, but also stability and robustness have been guaranteed with the attenuation of the chattering effect.

6. Reference

[1] Tzuu-Hseng S.L., Yun-Cheng H.: MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators, Information Sciences 180, pp. 4641–466 (2010).

[2] Chang, M.K.: An adaptive self-organizing fuzzy sliding mode controller for a 2-DOF rehabilitation robot actuated by pneumatic muscle actuators, Journal of Control Engineering Practice 18, pp. 13–22 (2010).

[3] Castillo, O., Melin, P.: Type-2 Fuzzy Logic: Theory and Applications, Edit Springer-Verlag Berlin Heidelberg, (2008).

[4] Biglarbegian, M., Melek, W. W., Mendel, J. M.: Design of Novel Interval Type-2 Fuzzy Controllers for Modular and Reconfigurable Robots: Theory and Experiments, IEEE Trans. on Industrial Electronics, 58(4), pp.1371–1384 (2011).

[5] Castillo, O., P. Melin, A review on the design and optimization of interval type-2 fuzzy controllers, Applied Soft Computing 12 (2012) 1267–1278

[6] Liang, Q., and Mendel, J. M.: Interval Type-2 Fuzzy Logic Systems: Theory and Design, IEEE Trans. on Fuzzy Systems, 8(5), pp. 535–550 (2000).

[7] Hsiao, M. Y.: Design of interval type-2 fuzzy sliding-mode controller, Information Sciences, 178(6), pp. 1696–1716 (2008).

[8] Lenz A., Anderson R S., Pipe A. G., Melhuish C., Dean P., and Porrill J., Cerebellar-Inspired Adaptive Control of a Robot Eye Actuated by Pneumatic Artificial Muscles, IEEE Trans. On Systems, Man, and Cybernetics—Part B: Cybernetics, 39(6), pp. 1420–1433 (2009).

www.intechopen.com

Rezoug Amar, Hamerlain Mustapha and Tadjine Mohamed: Decentralized RBFNN Type-2 Fuzzy Sliding Mode Controller for Robot Manipulator Driven by Artificial Muscles

11

[9] Daerden, F., Lefeber, D.: Pneumatic Artificial Muscles: actuators for robotics and automation, European journal of mechanical and environmental engineering. 47(1), pp. 11–21 (2002).

[10] Lilly, J. H., Yang, L.: Sliding Mode Tracking for Pneumatic Muscle Actuators in Opposing Pair Configuration, IEEE trans. on control systems technology, 13(4), pp. 550–558 (2005).

[11] Lopez, P., Nouri, A. S. (français(France)) : Théorie élémentaire et pratique de la commande par les régimes glissants, Springer-Verlag Berlin Heidelberg. (2006).

[12] Schmitt, J., Grabert, F., Raatz, A.: Design of a Hyper-Flexible Assembly Robot Using Artificial Muscles. In: Proceedings – IEEE Int. Conf. on Robotics and biomimrtics, pp. 897–902, Tianjin, (2010).

[13] Schröder, J., Kawamura, K., Gockel, T., Dillmann, R.: Improved control of a humanoid arm driven by pneumatic actuators, in Proc. of Humanoids, pp. 122-–131 (2003).

[14] VoMinha, T., Tjahjowidodob, T., Ramon, H., Brussel, H. V.: Cascade position control of a single pneumatic artificial muscle–mass system with hysteresis compensation, Mechatronics 20, pp. 402–414 (2010).

[15] Tondu, B.: Artificial muscles for humanoid robots, Humanoid Robots: Human-like Machines, Book edited by: Matthias Hackel, Itech, Vienna, Austria, pp. 89-122, (2007).

[16] Chen, J.Y.: Expert SMC-based fuzzy control with genetic algorithms, Journal of the Franklin Institute. 336, pp. 589–610 (1999).

[17] Braikia, K., Tondu, B., Chettouh, M., Acco, P.: Improved control strategy of 2-sliding controls applied to a flexible robot arm, Advanced Robotics. 25(11), pp. 1515–1538 (2011).

[18] Ozkan, M., Inoue, K., Negishi, K., Yamanak, T.: Defining a neural network controller structure for a rubbertuator robot, Neural Networks. 13, pp. 533–544 (2000).

[19] Thanh, T.D.C., Ahn, K.K.: Nonlinear PID control to improve the control performance of 2 axes pneumatic artificial muscle manipulator using neural network, Mechatronics. 16, pp.577–587 (2006).

[20] Lu, H.C., Tsai, C.H., Chang, M.H.: Radial basis function neural network with sliding mode control for robotic manipulators. In: Proc. IEEE Int. Conf. on Systems Man and Cybernetics, pp. 1209 – 1215. Turkey (2010).

[21] Jin, Y.: Decentralized adaptive fuzzy control of robot manipulators, IEEE Trans. On Sys, Man, and Cyb—Part B: Cybernetics, 28(1), pp. 47–57 (1998).

[22] Harashima F., H. Hashimoto, K. Maruyama, practical robust control of robot arm using variable structure system, IEEE 1986.

[23] Tai, N. T., Ahn, K. K.: A RBF Neural Network Sliding Mode Controller for SMA Actuator, Int. Journal of Control, Automation, and Systems 8(6), pp. 1296-1305, 2010.

[24] Fei, J., H. Ding, S. Hou, S. Wang and M. Xin, Robust Adaptive Neural Sliding Mode Approach for Tracking Control of a MEMS Triaxial Gyroscope,Int J Adv Robotic Sy, 2012, Vol. 9.

[25] Lian R.J.: Intelligent Controller for Robotic Motion Control, IEEE Tran. on Industrielles Eelectronics, 58(11), pp. 5220–5230 (2011).

[26] Pomiers, P.: Pneumatic muscles robot arm user's guide, Robosoft S.A, (2003).

[27] Aun K. K., Anh H. P. H., System Modeling Identification and Control of the Two-Link Pneumatic Artificial Muscle Manipulator Optimized with Genetic Algorithms, IEEE Int. Conf. on Control and Automation Guangzhou, China, 2007.

[28] Lin, C. M. and Y. J. Mon, Decoupling control by hierarchical fuzzy sliding mode controller, IEEE Tran. On Control Systems Technology, 13(4), pp. 593-598, 2005.

[29] Lilly, J. H. and L. Yang, Sliding Mode Tracking for Pneumatic Muscle Actuators in Opposing Pair Configuration, IEEE Tran. On control systems technology, 13(4), pp.550-558, july 2005.

[30] Juan R. C., O. Castillo, P. Melin, A. Rodríguez-Díaz, A hybrid learning algorithm for a class of interval type-2 fuzzy neural networks, Information Sciences 179 (2009) 2175–2193.

[31] Martínez R., O. Castillo, L. T. Aguilar, Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms, Information Sciences 179 (2009) pp. 2158–2174.

[32] Nohe R. C. C., L. T. Aguilar, O. Castillo, Fuzzy logic control with genetic membership function parameters optimization for the output regulation of a servomechanism with nonlinear backlash, Expert Systems with Applications 37 (2010) 4368–4378.

[33] Wu D., Tan, W. W.: A simplified type-2 fuzzy logic controller for real-time control, ISA Transactions, 45(4), pp.503–51 (2006).

[34] Castillo, O., R. M. Marroquín, P. Melin, F. Valdez, J. Soria, Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, Information Sciences 192 (2012) 19–38.

www.intechopen.com

www.manaraa.com

www.manaraa.com