

Sequential Adaptive RBF-Fuzzy Variable Structure Control Applied to Robotics Systems

Mohammed Salem

University of Oran, Faculty of Exact and Applied Sciences, Oran, 31000, Algeria
Email: salem@univ-mascara.dz

Mohamed F. Khelifi

University of Oran, Faculty of Exact and Applied Sciences, RIIR Lab, Oran, 31000, Algeria
Email: mf_khelifi@yahoo.fr

Abstract— In this paper, we present a combination of sequential trained radial basis function networks and fuzzy techniques to enhance the variable structure controllers dedicated to robotics systems. In this aim, four RBFs networks were used to estimate the model based part parameters (Inertia, Centrifugal and Coriolis, Gravity and Friction matrices) of a variable structure controller so to respond to model variation and disturbances, a sequential online training algorithm based on Growing-Pruning "GAP" strategy and Kalman filter was implemented. To eliminate the chattering effect, the corrective control of the VS control was computed by a fuzzy controller. Simulations are carried out to control three degrees of freedom SCARA robot manipulator where the obtained results show good disturbance rejection and chattering elimination.

Index Terms— Radial Basis Function Networks, Sequential Training, Growing and Pruning, Fuzzy Control, Variable Structure Control, Robot Manipulator.

I. INTRODUCTION

Robust control structures are widely applied to control robot manipulators which are known to be highly coupled nonlinear systems [1].

Robots are usually exposed to load variations, uncertainties and disturbances which make them very hard to control. Variable structure controllers are robust schemes that use a high gain switching term to drive the system states toward a specific surface and maintain the system along it [2].

These controllers are robust against structured and unstructured uncertainties, but they have two important drawbacks:

The first one is high frequency oscillations in the control signal (chattering) and the other is the model dependency where the knowledge of the exact system parameters is required to compute the equivalent control. Several researches works have been developed over the use of neural networks and fuzzy techniques to fix up such disadvantages.

Authors of [3] used a fuzzy controller to compute the gain K in the discontinuous control part while [4] trained one RBF network to estimate the global robot model function. In [5], a sliding mode control with a neural network is used to predict the unknown interconnection terms of nonlinear interconnected systems. In [6], fuzzy

sliding mode control is used to control nonlinear systems with structured and unstructured uncertainties with feedback linearization.

Authors of [7] trained one layer MLP neural network to design a sliding mode of single and coupled inverted pendulum. A neural adaptive based sliding mode controller is designed to trajectory tracking for mobile robots in [8].

Radial basis function (RBF) neural networks with their approximation capabilities [9] are very emergent powerful tools for system identification, so they can estimate any complex nonlinear system parameters even in presence of disturbances [10] [11].

In this paper, and to overcome the variable structure control lacks, a hybrid control law is proposed where four RBF networks are used to estimate the model parameters for the equivalent control taking advantage from the basic knowledge of the robot dynamic model. To eliminate the chattering, the corrective control part is designed with a fuzzy controller to update the gain so to avoid the use of a constant predefined value.

To train the proposed networks, a sequential training algorithm based on Growing-Pruning (GAP) strategy and Kalman filter is implemented [9]. A similar approach in [12] was applied to control two degrees of freedom robot manipulator where we ignored the frictions. The present work represents an extension of [12] where a fuzzy controller is implemented and neuron significance is introduced in the GAP criteria. The proposed approach is implemented to control a more complex nonlinear plant, three degrees of freedom RRP SCARA more constrained system than two degrees of freedom robot system used in [12] with consideration of frictions and disturbances.

The remainder of this paper is organized as follows:

Section 2 is dedicated to the variable structure controller while in section 3; the main steps of GAP-EKF are explained. The application of the fuzzy-RBF based variable structure control is presented in section 4. Simulations results are summarized in section 5. Section 6 gives some concluding remarks.

II. VARIABLE STRUCTURE CONTROL SCHEME

A. The robot model

The dynamic equation of a manipulator of n degrees of freedom is given by (1) [2]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) = \Gamma \quad (1)$$

where q, \dot{q}, \ddot{q} are position, velocity and acceleration vectors respectively, Γ is the actuators torques vector. $M(q)$ is the symmetric and positive definite inertia matrix, $C(q, \dot{q})$ is the Coriolis/Centrifugal matrix while $G(q)$ is the gravity vector. $F(\dot{q})$ is the friction terms vector.

In the sequel, we will use the symbols M , C , G and F instead of $M(q)$, $C(q, \dot{q})$, $G(q)$ and $F(\dot{q})$.

B. Variable structure controller

A classical variable structure control is designed based on driving the robot states to slide over a surface s where all motion in neighborhood of the manifold is driven toward it [2][14].

Let's define the tracking error like:

$$e = q - q_d. \quad (2)$$

where q_d is the desired position and q is joint position. The sliding surface is defined by [14]:

$$s = \dot{e} + \lambda e. \quad (3)$$

where $\lambda = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_n]$, λ_i is a positive constant.

We have to choose λ so that the energy of the system will decay when s is not null, so it has to satisfy the condition [2]:

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s|. \quad (4)$$

The variable structure controller is given by:

$$\Gamma = \Gamma_{eq} + \Gamma_{cc}. \quad (5)$$

where the equivalent control part is given by:

$$\Gamma_{eq} = \hat{M}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q}_r + \hat{G}(q) + \hat{F}(\dot{q}). \quad (6)$$

and the corrective one is defined by:

$$\Gamma_{cc} = -K \text{sign}(s). \quad (7)$$

$\hat{M}, \hat{C}, \hat{G}, \hat{F}$ are estimations of M, C, G and F respectively [2]. $K = \text{diag}[K_1, K_2, \dots, K_n]$ is a diagonal positive definite matrix in which K_n is a positive constant and \ddot{q}_r is given by:

$$\ddot{q}_r = \ddot{q}_d - \lambda \dot{e}, \quad \dot{q}_r = \dot{q}_d - \lambda e. \quad (8)$$

The closed loop relation is given by [2]:

$$M\dot{s} + Cs = \Delta f - K \text{sign}(s). \quad (9)$$

where Δf is the estimation error:

$$\Delta f = (\hat{M} - M)\ddot{q}_r + (\hat{C} - C)\dot{q}_r + (\hat{G} - G) + (\hat{F} - F). \quad (10)$$

To prove the stability of the proposed controller, a positive definite Lyapunov function candidate V and its time derivative are given by [2]:

$$V(s) = \frac{1}{2} s^T M s. \quad (11)$$

$$\dot{V}(s) = |\Delta f - K|. \quad (12)$$

The closed loop system is stable if the gain K is chosen greater than the boundary of uncertainties [2]:

$$K > \Delta f. \quad (13)$$

In a real environment, we have an imprecise idea about the value Δf and to ensure the stability of the controlled robot we have to work with high gains which cause high frequency oscillations and discontinuities in the control signal.

The designer has to estimate the exact model of the robot to minimize the boundary of Δf , and decrease the value of K to cancel the effect of chattering, which is not always evident due to disturbances, friction and uncertainties. This estimation could be done efficiently by using Radial Basis Function "RBF" with sequential training.

In other hand, the value of the gain K is estimated by a fuzzy controller so to start with high values when the states of the system are far from the surface and decrease the gain value near the sliding surface.

III. RBF SEQUENTIAL TRAINING ALGORITHM

A. RBF Network description

Radial basis function networks (RBF) are a variety of artificial neural networks [20]; they have one hidden layer and a linear output. The network output $\hat{y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{N_y}]$ is given by (14) [10]:

$$\hat{y} = f(X) = \Phi^T(X)A. \quad (14)$$

$A = [a_1, a_2, \dots, a_N]$ is the weight's vector connecting the hidden layer to the output layer with N is the hidden layer neurons number, $\Phi^T(X)$ is the response of the hidden layer to the input vector $X = [x_1, x_2, \dots, x_{N_x}]$, N_x and N_y are the inputs and outputs dimensions respectively and Φ is the Gaussian function given by [9]:

$$\Phi(X) = \exp\left(-\frac{\|X - \mu\|^2}{\sigma^2}\right). \quad (15)$$

μ is the center's vector of the hidden neurons and σ is the width of the Gaussian function.

B. Sequential network training algorithm

To obtain the desired behavior y_i responding to the input x_i , where i is the sample index, the neural network is trained using a sequential algorithm with a Growing and Pruning strategy (GAP) [9] to optimize the network architecture and an Extended Kalman Filter to adjust the

network parameters. The growing of the RBF neural network is controlled regarding the neuron's influence in the global output [13]. The influence of the j^{th} neuron is given by :

$$Inf_j = \left\| \frac{1.8\sigma_j^{N_x} a_j}{\sum_{j=1}^N (1.8\sigma_j)^{N_x}} \right\|. \quad (16)$$

1). Growing criterium

While using and training the network, the algorithm decides to add a new neuron to the network if the criteria in (17) and (18) are satisfied [13]:

$$\|x_i - \mu_{ir}\| > \varepsilon_i. \quad (17)$$

$$\|e_i\| \frac{(1.8\kappa \|x_i - \mu_{ir}\|)^{N_x}}{\sum_{j=1}^{N+1} (1.8\sigma_j)^{N_x}} > \varepsilon_m. \quad (18)$$

where:

$$e_i = y_i(x_i) - \hat{y}_i(x_i). \quad (19)$$

and

$$\varepsilon_i = \max \left\{ \varepsilon_{\max} \gamma^i, \varepsilon_{\min} \right\}. \quad (20)$$

μ_{ir} are the centers of the closest hidden unit to current input x_i .

ε , ε_{\max} , ε_{\min} and γ are thresholds to be selected appropriately. Equation (17) checks if the current input data x_i are far from all existing hidden neurons so they can not reproduce the desired outputs while (18) checks if the influence of the new added $N+1$ node is greater than a threshold ε_m .

The parameters of the new added neuron are described in (21):

$$\begin{cases} a_{N+1} = e_i \\ \mu_{N+1} = x_i \\ \sigma_{N+1} = \kappa \|x_i - \mu_{ir}\| \end{cases} \quad (21)$$

where κ is the overlap factor.

2). Adjusting the network parameters with EKF

If the growing criteria in (17) and (18) are not met, only the network parameters of the nearest node to the current inputs are updated using the extended Kalman filter (EKF) as follows [9]:

$$w_i^r = w_{i-1}^r + K_i e_i. \quad (22)$$

The vector w_i^r contains the weights, centers and width of the nearest neuron to the current input data x_i :

$$w^r = [a_r, \mu_r, \sigma_r]. \quad (23)$$

and K_i is the Kalman gain matrix which is computed for every new input/output data by:

$$K_i = P_{i-1} B_i [R_i + B_i^T P_{i-1} B_i]^{-1}. \quad (24)$$

$B_i = \nabla_w \hat{y}$ is the gradient matrix of the function \hat{y} . R_i is the variance of the measurement noise. P_i is the error covariance matrix which is updated by:

$$P_i = [I_{z^*z} - K_i B_i^T B_i] P_{i-1} + q_0 I_{z^*z}. \quad (25)$$

q_0 is a scalar that determines the allowed random step in the direction of the gradient matrix. If the number of parameters to be adjusted is z , then P_i is a $z \times z$ positive definite symmetric matrix. When a new hidden neuron is added, the dimensions of P_i will be:

$$P_i = \begin{pmatrix} P_{i-1} & 0 \\ 0 & q_0 I_{g^*g} \end{pmatrix}. \quad (26)$$

The new rows and columns are initialized by q_0 (an initial uncertainty estimation). g is the number of new parameters introduced by the new hidden neuron.

3). Pruning criterium

For each presented data, the j^{th} neuron could be deleted if the criterium in (27) is verified:

$$\|a_j\| \frac{(1.8\sigma_j)^{N_x}}{\sum_{j=1}^N (1.8\sigma_j)^{N_x}} > e_p. \quad (27)$$

where e_p is a pruning threshold to be selected.

IV. RBF-FUZZY BASED VARIABLE STRUCTURE CONTROLLER

A. RBF Network based equivalent control

To fix the problems of model dependency of the variable structure controller in (5), four RBF networks are trained online to estimate the parameters of matrices M , C , G and F in (6).

The main idea is to use RBF networks defined in section III to estimate the parameters of the equivalent control. The networks are online trained using the GAP-EKF based algorithm. In this work we use a RBF network to estimate inertia matrix M , another one to estimate Coriolis matrix C , a third RBF neural network to estimate gravity vector G and the last one is used to compensate friction terms F . The outputs of the RBF networks are the estimated components of the model matrices as given in (28, 29, 30 31)

$$\begin{aligned} \hat{M}(q) &= f_M(X_M) \\ &= \Theta_M^T(X_M) A_M, X_M = [q^T] \end{aligned} \quad (28)$$

$$\begin{aligned} \hat{C}(q, \dot{q}) &= f_C(X_C) \\ &= \Theta_C^T(X_C)A_C, X_C = [q^T, \dot{q}^T] \end{aligned} \quad (29)$$

$$\begin{aligned} \hat{G}(q) &= f_G(X_G) \\ &= \Theta_G^T(X_G)A_G, X_G = [q^T]. \end{aligned} \quad (30)$$

$$\begin{aligned} \hat{F}(\dot{q}) &= f_F(X_F) \\ &= \Theta_F^T(X_F)A_F, X_F = [\dot{q}^T]. \end{aligned} \quad (31)$$

where f_M, f_C, f_G and f_F are four independent RBF neural networks with inputs X_M, X_C, X_G and X_F . Their weights vectors are A_M, A_C, A_G and A_F respectively

Equation (6) becomes:

$$\begin{aligned} \Gamma_{eq} &= \Theta_M^T(q^T)A_M\ddot{q}_r + \Theta_C^T(q^T, \dot{q}^T)A_C\dot{q}_r + \\ &\quad \Theta_G^T(q^T)A_G + \Theta_F^T(\dot{q}^T)A_F. \end{aligned} \quad (32)$$

The resulted RBF based equivalent control is shown in Fig.1.

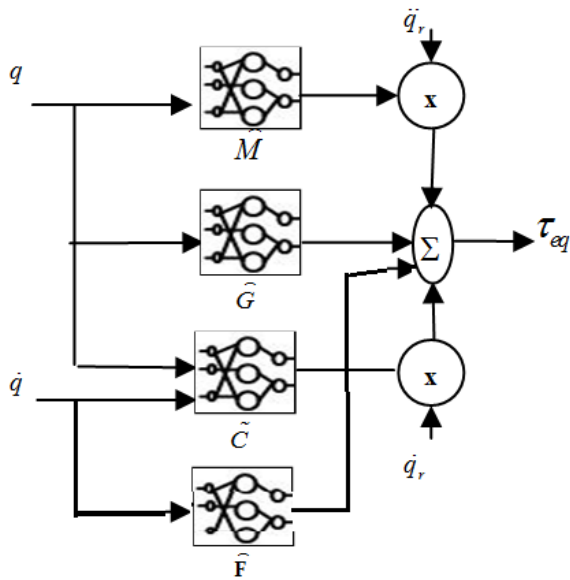


Fig 1. RBF based equivalent controller

B. Fuzzy corrective control

Usually, a fuzzy system has one or more inputs and a single output with a rule base and an inference engine as showed in Fig.2. The rules in the rule base are in the following form [15]:

IF x_i is A_i THEN y_i is B_i

where A_i and B_i are fuzzy sets

The output of the fuzzy controller is given by (33) [16]:

$$y = \frac{\sum_{m=1}^M \theta^m \prod_{i=1}^n \mu_{A_i}(x_i^*)}{\sum_{m=1}^M \prod_{i=1}^n \mu_{A_i}(x_i^*)} = \theta^T \varpi(x) \quad (33)$$

where $\theta = [\theta^1, \dots, \theta^m, \dots, \theta^M]^T$ is the y membership functions centers vector. $\varpi(x) = [\varpi^1(x), \dots, \varpi^m(x), \dots, \varpi^M(x)]^T$ is the vector of heights of the output y membership functions and $\varpi^m(x)$ is given by[21]:

$$\varpi^m(x) = \frac{\prod_{i=1}^n \mu_{A_i}(x_i^*)}{\sum_{m=1}^M \prod_{i=1}^n \mu_{A_i}(x_i^*)} \quad (34)$$

where M is the amount of rules

The corrective control in (7) is obtained by a fuzzy controller. The tuning rule for the gain K is to decrease its value near the sliding surface in order to damp the chattering amplitude and to increase it far from the surface. In this regard, a fuzzy controller rule base is proposed for tuning K as illustrated in table (1), in which inputs are the surface s and its time derivative respectively. (ZE: Zero; B: Big; M: Medium; N: Negative; P: Positive). For multiple-output system, they can be considered as a combination of several single-output systems.

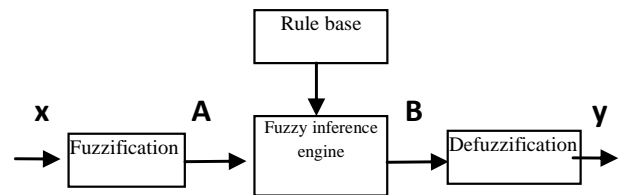


Fig. 2. Fuzzy controller

Table.1. Fuzzy rule base for tuning the gain “K”

$\dot{s} \backslash s$	NB	NS	ZE	PS	PB
N	KB	KB	KM	KS	KB
Z	KB	KM	KS	KM	KB
P	KB	KS	KM	KB	KB

To design the corrective control, membership functions for the fuzzy controller input (surface s, its derivation) and the output (gain K) are presented in Fig.3, Fig.4 and Fig. 5.

The whole fuzzy-GAP-RBF neural networks based variable structure controller is shown in Fig.6.

V. SIMULATION RESULTS

Simulations were carried out over a three dof RRP SCARA robot manipulator whose dynamic model is given by (35) [4]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) = \Gamma. \quad (35)$$

with:

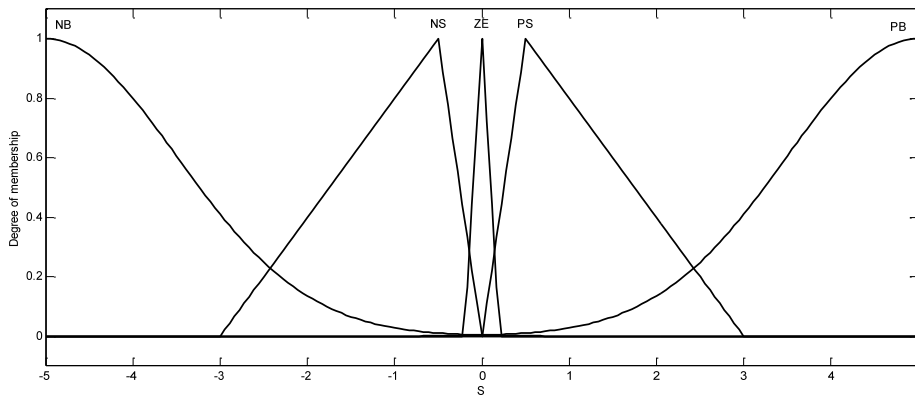


Fig. 3. Membership function of s (input 1)

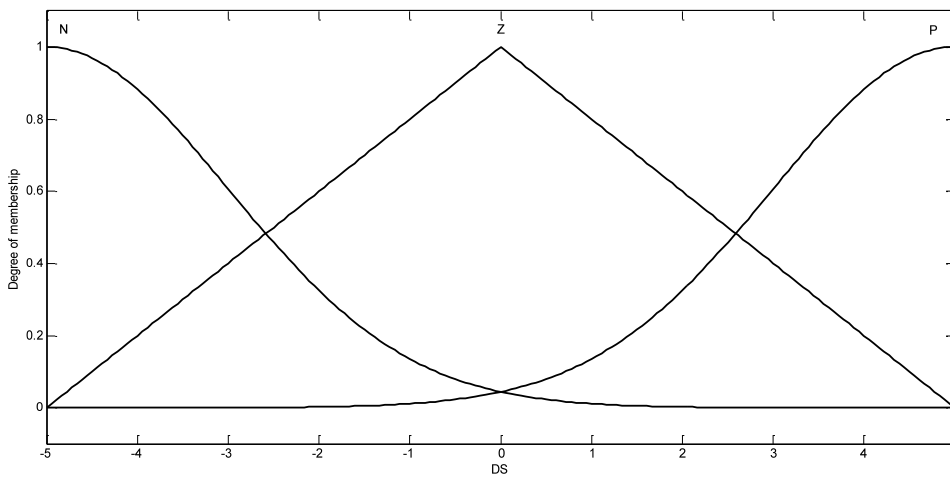


Fig. 4. Membership function of (input 2)

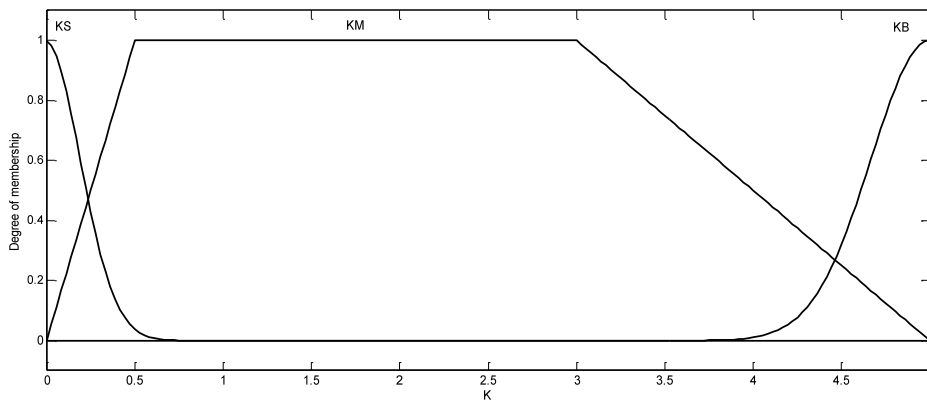


Fig.5. Membership function of K (output)

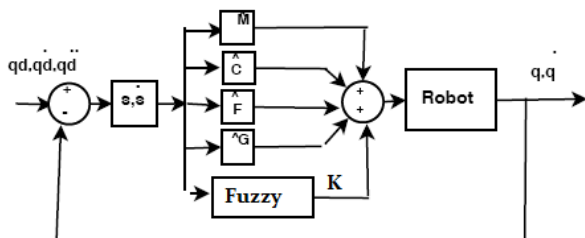


Fig. 6. Neural adaptive-Fuzzy variable structure controller

$$M(q) = \begin{cases} M_{11} = 2.1240 + 1.44\cos(q_2) \\ M_{12} = -0.4907 - 0.72\cos(q_2) \\ M_{13} = 0 \\ M_{21} = -0.4907 - 0.72\cos(q_2) \\ M_{22} = -0.4907, M_{23} = 0 \\ M_{31} = M_{32} = M_{33} = 0 \end{cases} \quad (36)$$

$$C(q, \dot{q}) = \begin{cases} C_{11} = 1.44\dot{q}_2 \sin(q_2) \\ C_{12} = -0.72\dot{q}_2 \sin(q_2) \\ C_{22} = -0.72\dot{q}_2 \sin(q_2) \\ C_{13} = C_{21} = C_{23} = 0 \\ C_{31} = C_{32} = C_{33} = 0 \end{cases} \quad (37)$$

$$\Gamma_d = \begin{bmatrix} 5\sin(2t) \\ 5\sin(2t) \\ 5\sin(2t) \end{bmatrix} \quad (40)$$

$$G(q) = \begin{cases} G_1 = 0 \\ G_2 = 0 \\ G_3 = -4.9 \end{cases} \quad (38)$$

$$F(\dot{q}) = \begin{cases} F_1 = 12\dot{q}_1 + 0.02\text{sign}(\dot{q}_1) + 3\sin(3t) \\ F_2 = 12\dot{q}_2 + 0.02\text{sign}(\dot{q}_2) + 3\sin(3t) \\ F_3 = 12\dot{q}_3 + 0.02\text{sign}(\dot{q}_3) + 3\sin(3t) \end{cases} \quad (39)$$

We used the variable structure controller in (5) to control the SCARA robot where the Inertia, Centrifugal and Coriolis and friction model matrices are estimated using the proposed sequential algorithm. In our case (38), the gravity vector $G(q)$ remains constant and does not need to be estimated.

To test the robustness of the proposed controller; we considered the disturbances in (40) and altered the parameters of the inertia matrix by adding a quantity $\Delta m_2 = 0.001\text{kg}$ to the second link mass m_2 .

The desired trajectory to be tracked is a sinusoidal signal as shown in (41):

$$q_d = 1 + \sin(\pi t) \quad (41)$$

Fig.7, Fig.8, Fig.9 show the desired and real positions of the three joints and Fig.10, Fig.11, Fig.12 show the tracking error of the robot joints while velocity errors are presented in Fig.13, Fig.14, Fig.15. We see that both position and velocity errors are acceptable and converge toward zero in a short time (after 1s). That means that the proposed RBF neural networks (with the sequential learning algorithm) needed just few data samples to learn the robot behavior.

The external disturbances are:

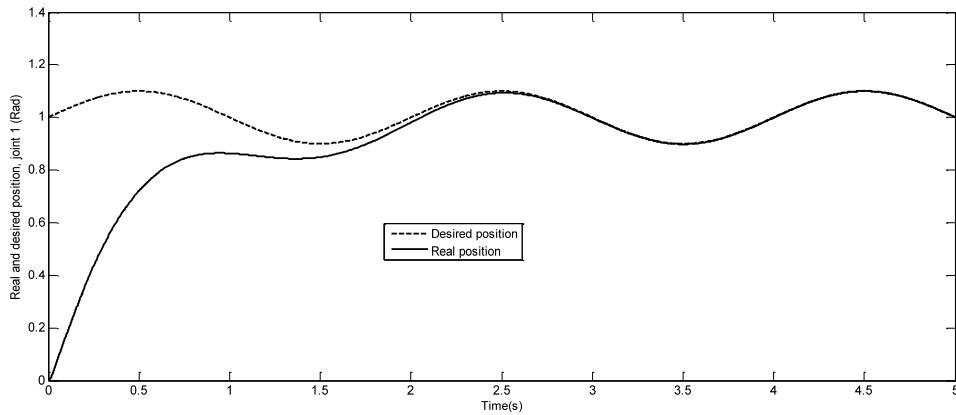


Fig. 7. Desired and real position of Joint 1

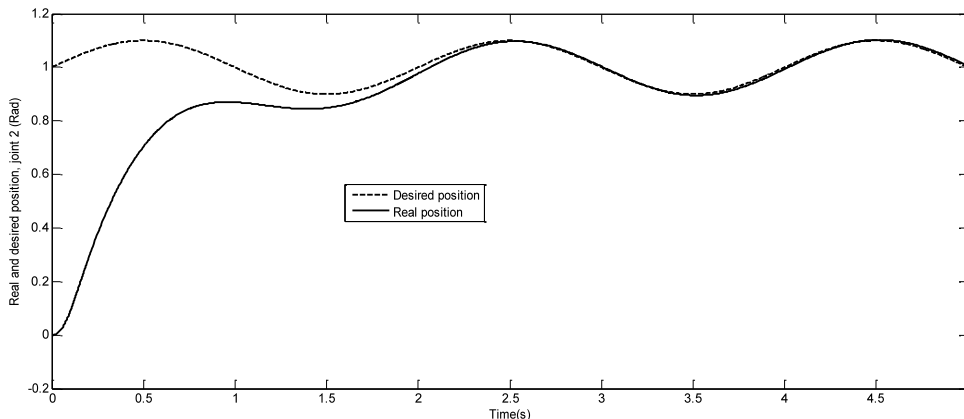


Fig. 8. Desired and real position of Joint 2

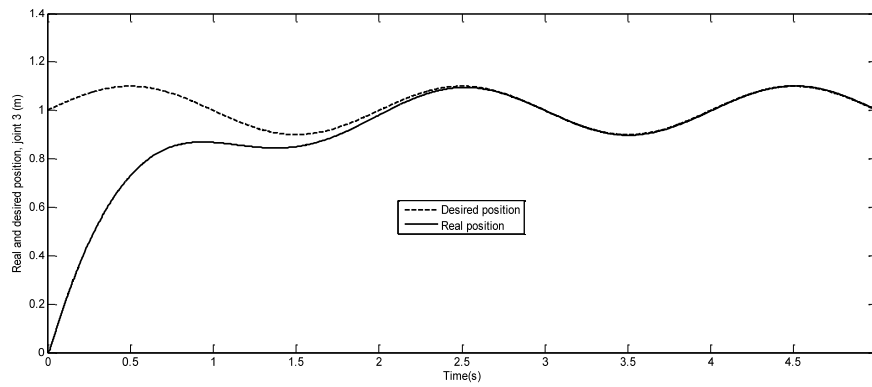


Fig. 9. Desired and real position of Joint 3

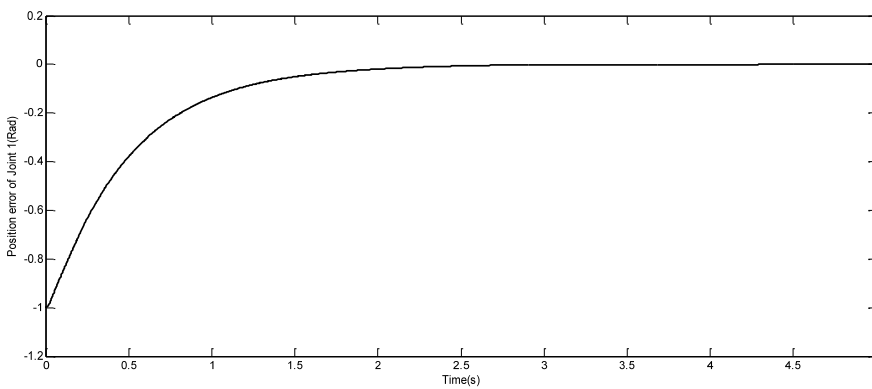


Fig.10. Position error of Joint 1

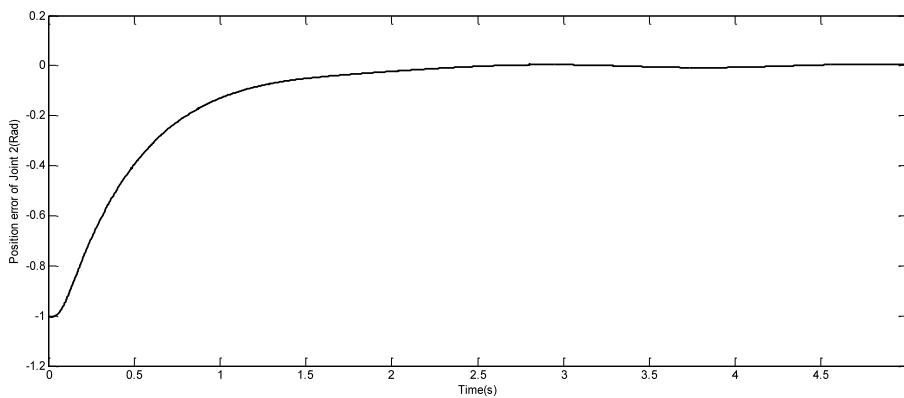


Fig.11. Position error of Joint 2

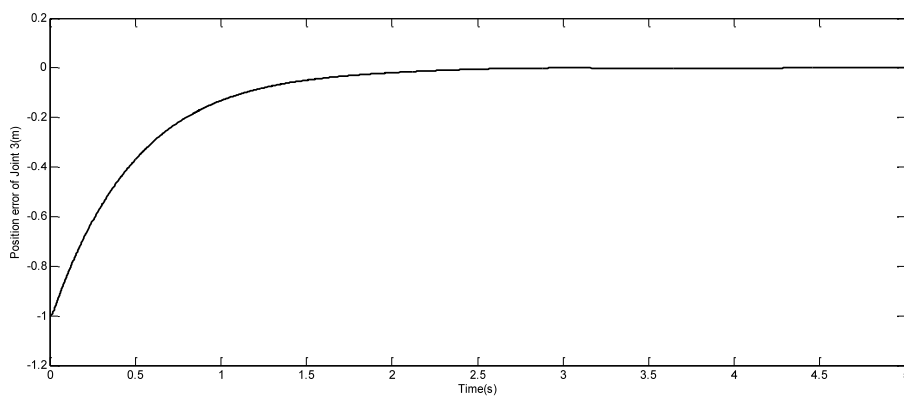


Fig. 12. Position error of Joint 3

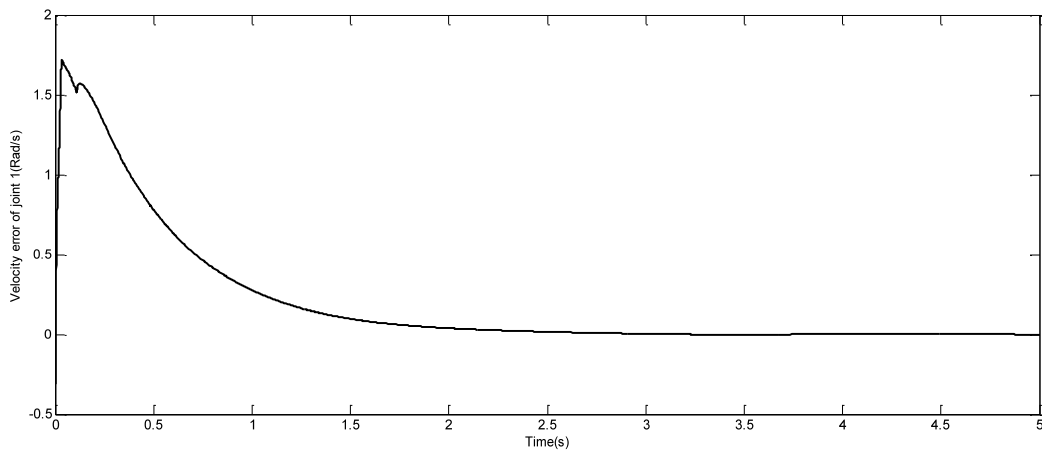


Fig. 13. Velocity error of Joint 1

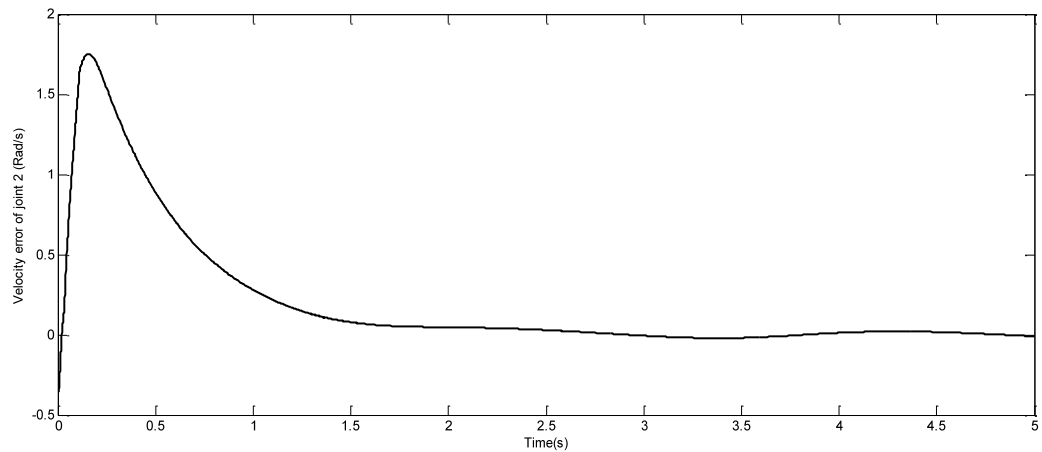


Fig. 14. Velocity error of Joint 2

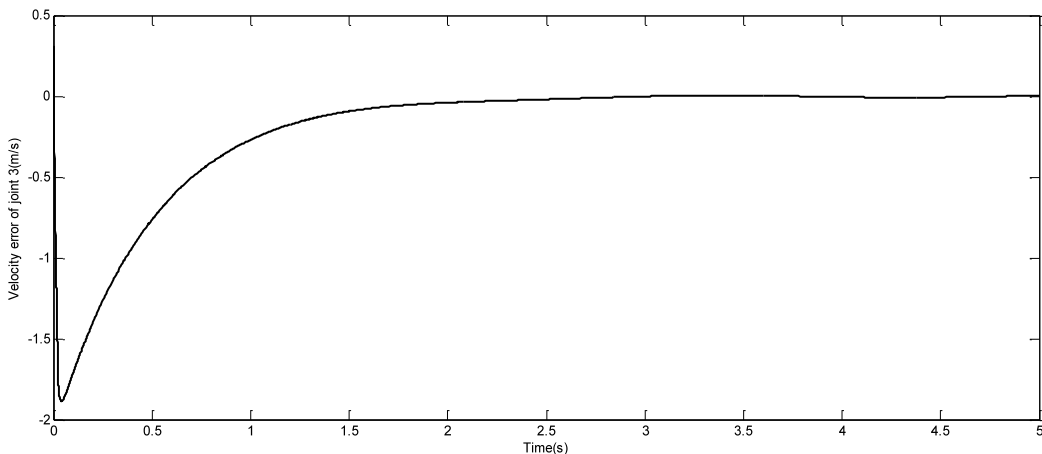


Fig. 15. Velocity error of Joint 3

Torques acting on the robots joints and sliding surface are presented in Fig.16, Fig.17, Fig.18 and Fig.19 respectively. We see clearly from the control signal figures that chattering is completely eliminated due to the good estimation of the model parameters and so we do

not need to work with high gain values to compensate the model variations, uncertainties and external disturbances.

Values of the gain K are presented in Fig. 20 and we see that K is big far from the surface and when the system states are close to the sliding surface the values of k are decreased to be null.

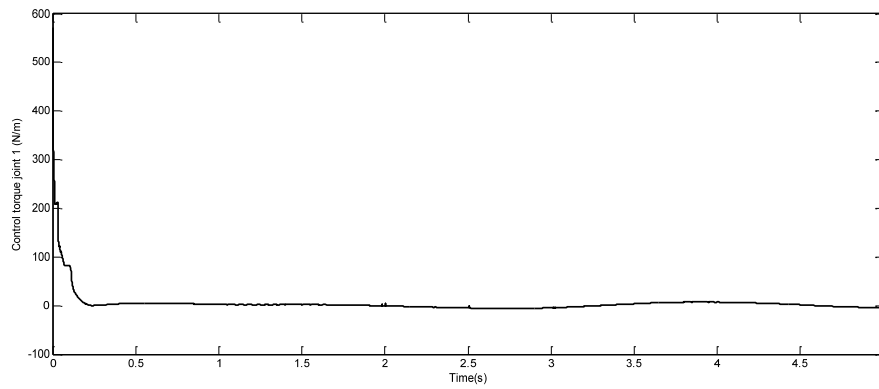


Fig. 16. Control torque for joint 1

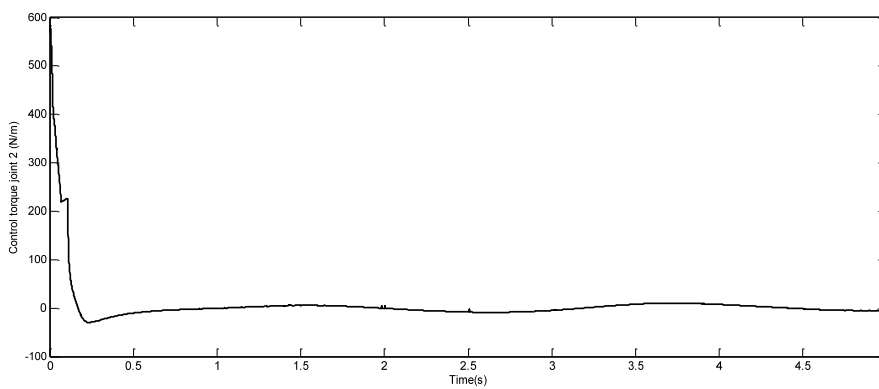


Fig. 17. Control torque for joint 2

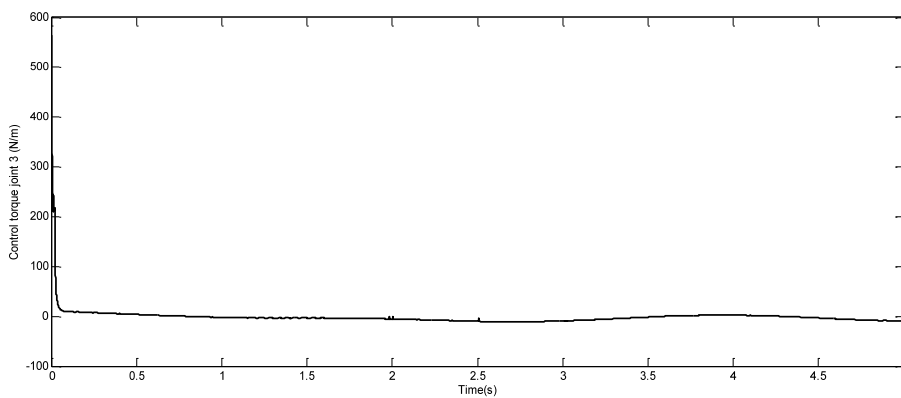


Fig. 18. Control torque for joint 3

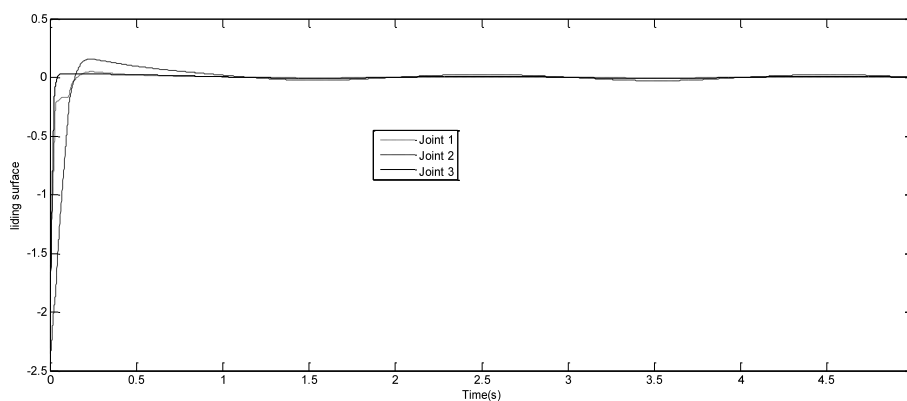


Fig. 19. Sliding surface

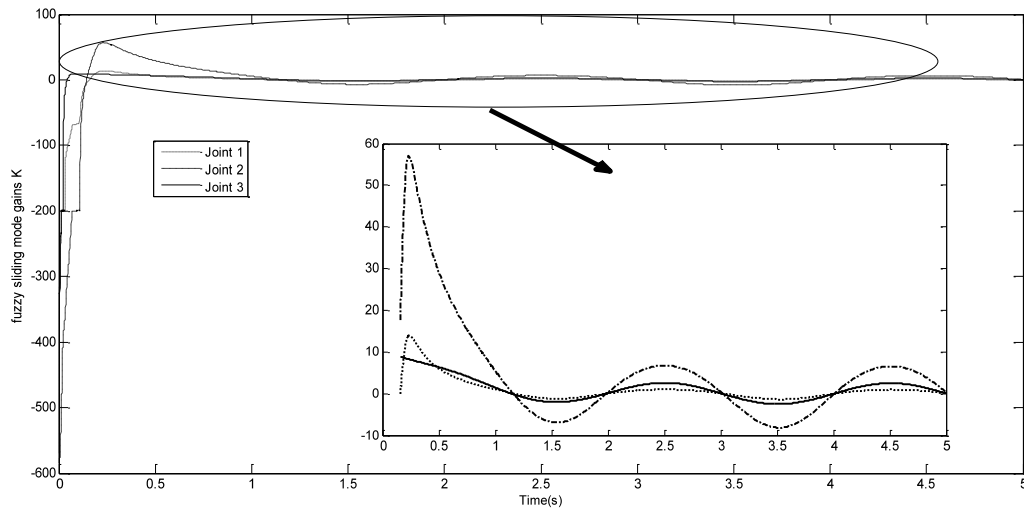


Fig. 20. fuzzy controller output (Gain K)

The training of RBF neural networks is achieved online during the control process. Table.2 summarizes mean square errors of training for the three RBF neural

networks for each second of the simulation time (5s) in presence of external disturbances and model uncertainties.

Table 2. Estimation errors of the RBF neural networks

Seco-ndes	without disturbances and uncertainties			without disturbances and uncertainties		
	M	C	F	M	C	F
1	5.41e-6	4.53e-8	1.16e-16	2.62e-5	9.21e-8	3.88e-15
2	6.27e-8	4.45e-8	2.22e-17	3.54e-7	6.55e-8	3.29e-16
3	2.49e-8	7.92e-8	8.88e-17	4.02e-7	8.22e-8	1.24e-16
4	1.56e-8	3.33e-8	3.05e-17	8.23e-7	5.36e-8	5.86e-16
5	2.85e-8	9.74e-8	8.72e-17	5.14e-7	1.94e-7	2.11e-16

The training errors are less than $1e-6$, $1e-8$, $1e-16$ for estimating inertia, Coriolis and friction matrices respectively. These small training errors are due to the strength of the sequential training algorithm. Indeed, the fact that the training is achieved online, allows the RBF to react against any disturbance or model variation.

VI. CONCLUSION

We presented in this paper a variable structure controller for robot manipulator where the equivalent control was achieved using four radial basis function neural networks to estimate the robot matrices (Inertia, Centrifugal and Coriolis, Gravity and Friction). To enhance the estimation abilities of the networks, we implemented a sequential training algorithm to identify the model parameters online. The corrective control was replaced by a fuzzy controller to compute the K gain values along the control process.

The obtained results are very acceptable even in presence of uncertainties and the chattering problem of variable structure controllers is avoided.

REFERENCES

- [1] F.L. Lewis, C. T. Abdallah, and D. M. Dawson, *Control of Robot Manipulators*, Macmillan Publishing Company, Inc, USA, 1993.
- [2] J.J.E. Slotine and W. Li, *Applied Nonlinear Control*. Prentice-Hall, 1991.
- [3] S.E. Shaffei, "Sliding mode control of robot manipulators via intelligent approaches", *Advances strategies for Robots manipulators*. Sciyo, pp. 135-172., 2010.
- [4] A.G. Ak and G. Cansever, "Sliding mode controller with RBF networks for robotic manipulator trajectory tracking", *Lecture notes in control and information sciences (Intelligent Control and Automation)*, vol.144, pp.527-532, 2006.
- [5] S. Sefriti, J. Boumhidi, M. Benyakhlef and I. Boumhidi, "adaptive decentralized sliding mode neural network Control of a class of nonlinear interconnected systems", *International Journal of Innovative Computing, Information and Control*, vol. 9-7, pp. 2941-2947, 2013.
- [6] M. R. Soltanpour, B. Zolfaghari, M. Soltani and M. H. Khooban, "Fuzzy sliding mode control design for a class of nonlinear systems with structured and unstructured uncertainties", *International Journal of Innovative Computing, Information and Control*, Vol. 9-7, pp. 2713-2726, 2013.

- [7] L. Mehri, M. Salimifar, M. Mansouri and M. Teshnelab, "Sliding Mode Trained Neural Control for Single and Coupled Inverted Pendulum System", *Researches and Applications in Mechanical Engineering*, Vol. 2-4, 2013.
- [8] F. G. Rossomando, C. Soria and R. Carelli, "Sliding Mode Neuro Adaptive Control in Trajectory Tracking for Mobile Robots", *Journal of Intelligent & Robotic Systems*, 2013.
- [9] Y. Li, N. Sundararajan and P. Saratchandran, "Analysis of minimal radial basis function network algorithm for real-time identification of nonlinear dynamic systems: Control theory and application", *IEEE Proceedings-control Theory and Applications* vol. 147-4, pp.476-484, 2000.
- [10] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, N.J.: Prentice Hall, 1999. 2nd Edition.
- [11] D. Broomhead and D. Lowe, "Multivariate function approximation and adaptive networks," *Complex Systems*, vol. 2, pp. 321-355, 1988.
- [12] M.F. Khelfi and M. Salem, M., "RBF base sliding mode control of robot manipulator", *Proceedings of CCCA11, Hammamat, Tunisia*, 2011.
- [13] H. Rong, N. Sundararajan, G. Huang and P. Saratchandran, "Sequential adaptive fuzzy inference system (SAFIS) for nonlinear system identification and prediction", *Fuzzy Sets and Systems*, vol.157-9, pp.1260-1275, 2006.
- [14] T.P. Leung, C.-Y. Su and Q.-J. Zhou, "Sliding mode control of robot manipulators: a case study", *Industrial Electronics Society, IECON '90*, 1990.
- [15] T. Yoshimura, "Adaptive fuzzy sliding mode control for uncertain multi-input multi-output discrete-time systems using a set of noisy measurements", *International Journal of Systems Science*, 2013.
- [16] Q. P. Ha Q. H. Nguyen, D. C. Rye and H. F. Durrant-Whyte, "Fuzzy Sliding-Mode Controllers with Applications", *IEEE transactions on industrial electronics*, vol. 48-1, 2001.
- [17] V.Q. Leu, F. Mwasilu, H.H Choi, J. Lee and J.W. Jung, "Robust fuzzy neural network sliding mode control scheme for IPMSM drives", *International Journal of Electronics*, 2013.
- [18] D. Xu, B. Jiang, M. Qian and J. Zhao, "Terminal Sliding Mode Control Using Adaptive Fuzzy- Neural Observer", *Mathematical Problems in Engineering*, Vol 2013, 2013.
- [19] G. Xia and H. Wu, "Network-based neural adaptive sliding mode controller for the ship steering problem", *Advances in Swarm Intelligence Lecture Notes in Computer Science*, Vol. 7928. pp 497-505. 2013.
- [20] S. Bhuvanewari and J. Sabarathinam, "Defect Analysis Using Artificial Neural Network". *IJISA*, vol.5, no.5, pp.33-38, 2013, DOI: 10.5815/ijisa.2013.05.05
- [21] S.Bangia, P.R Sharma and M. Garg, "Simulation of Fuzzy Logic Based Shunt Hybrid Active Filter for Power Quality Improvement", *IJISA*, vol.5, no.2, pp.96, 2013, DOI: 10.5815/ijisa.2013.02.12

Authors' Profiles



Mohammed Salem was born in Mascara, Algeria in 1974; he received an Engineer degree in computer science from Sidi Bel Abbes University Algeria on 1999 and Master degree in Industrial computing from Oran University on 2007.

His research fields are the evolutionary computation and the application of artificial intelligence in identification and control of nonlinear systems.

He joined the Science & technology Faculty (Computer science department) University of Mascara, Algeria; he is also a Research Member of the Laboratory of Research in Industrial Computing and Networks University of Oran.



Mohamed F. Khelfi was born in Algiers, Algeria. He received Ph.D. degree in Automatic Control from Nancy University, France, in 1995. He is currently Professor at the Computer Science Department - Faculty of Sciences Exact and Applied Sciences - University of Oran - Algeria. He is also a research member at the Laboratory of Research in Industrial

Computing and Networks. His main research interests include Automatic Control, Industrial Computing, Robotics and Networks

How to cite this paper: Mohammed Salem, Mohamed F. Khelfi, "Sequential Adaptive RBF-Fuzzy Variable Structure Control Applied to Robotics Systems", *International Journal of Intelligent Systems and Applications(IJISA)*, vol.6, no.9, pp.19-29, 2014. DOI: 10.5815/ijisa.2014.09.03