International Journal of Innovative Research in Engineering
Volume 3, Issue 4 (July-August 2022), PP: 221-225.
www.theijire.com
ISSN No: 2582-8746

On-Line Adaptive Neuro Sliding Mode Control of Robot Manipulator

Sheheeda Manakkadu[1], Sourav Dutta[2]
[1]Department of Computer and Information Science, Gannon University, USA.
[2]Department of Computer Science, Ramapo College, USA.

How to cite this paper: Sheheeda Manakkadu[1], Sourav Dutta[2], "On-Line Adaptive Neuro Sliding Mode Control of Robot Manipulator", IJIRE-V3I04-221-225.

Abstract: *This paper proposes an online training scheme for RBFNN (Radial Basis Function Neural Network) based sliding mode controller to Control the robot manipulator. The approach is based on a sliding mode control methodology which drives the system towards a sliding surface by tuning the parameters of the controller using Gaussian radial basis function neural network. The key feature of this scheme is that prior knowledge of the system uncertainties is not required to guarantee the stability. Also the chattering phenomenon is completely eliminated. To demonstrate the effectiveness of the proposed approach, a three link Scara robot is simulated in the presence of uncertainties.*

Key Word: *Robot manipulator control, RBF Neural Network, Sliding mode control.*

I. INTRODUCTION

The increasing complexity of today's dynamical systems is frequently coupled with unknown dynamics, modeling errors, various sorts of disturbances, uncertainties, and noise. This creates a need for advanced control design techniques that are able to overcome limitations on traditional feedback control techniques. Along these lines, numerous research activities have been reported in the literature in the area of the so-called 'universal model-free controllers' that do not require a mathematical model of the controlled system but instead are able to learn on-line the systems that they are to control so that the performance can automatically be improved.

The use of artificial neural networks (NNs) is a common suggestion in this respect and NNbased controllers have nowadays entered the mainstream of control theory as a natural extension of adaptive control to systems that are nonlinear in the tunable parameters [1]. Most of the existing training methods for NNs rely on the gradient descent methodology and involve the computation of partial derivatives, or sensitivity functions. In this respect, they can be considered as extensions of the well-known back propagation algorithm for multilayer feed forward neural networks (MFNNs) [2] and hence they inherit some of its major drawbacks among which, in particular, is the difficulty to obtain analytical results concerning the convergence and stability of the learning schemes [3]. Recent research on the application of NNs for control has begun to address the closed-loop system structure and stability issues more rigorously [4]. The investigations in this area have been split over two main research directions. It has been shown in several works that the Lyapunov approach can be directly implemented to obtain robust training algorithms for continuous-time NNs [5–8]. Another proposed way to design a robust learning scheme is to utilize the variable structure systems (VSS) theory in constructing the parameter adaptation mechanism of the NNs [9,10] since the robustness of the variable structure control (VSC) scheme against unmodelled dynamics, disturbances, time delays and nonlinearities is well known Robotic manipulators are hard to control nonlinear systems that are frequently used as a test bed for evaluation of new control methods. Their coupled nonlinear equations with time-varying parameters and the ambiguities in the friction-related dynamics inevitably require the use of flexible control architectures. In the past decade, the applications of intelligent control techniques (fuzzy control or NN control) to the motion control for robot manipulators have received considerable attention [10].The motivation of this study is to design a stable neuro-adaptive control scheme for tracking control of robot manipulator with a guaranteed error convergence and without a requirement for prior knowledge of the dynamics of the controlled plant. The control scheme makes use of sliding mode control theory [7].

This paper proposes the RBF neural network based sliding mode controller for robot manipulator control. The results of simulation experiments exhibit the rapid convergence. Furthermore, it yields efficient training for constructing RBFNN. The weights of the hidden layer of the RBF neural networks are updated online to compensate for the system uncertainties and to guarantee the stability of the overall system, without prior knowledge of the system uncertainties.

II.MODEL OF ROBOTIC MANIPULATORS

The dynamic equation of an n-link rigid robotic manipulator system can be described by the following second-order nonlinear vector differential equation

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

where $q, \dot{q}, \ddot{q} \in R^n$ are joint position, velocity and acceleration vectors respectively, $M(q) \in R^{n\times n}$ denotes the inertia matrix, $C(q,\dot{q}) \in R^{n\times n}$ expresses the coriolis and centrifugal torques, $F(q,\dot{q}) \in R^{n\times n}$ is the unstructured uncertainties of the dynamics including friction and other disturbances, $G(q) \in R^n$ is the gravity vector and $u(t) \in R^{n\times 1}$ is the actuator torque vector acting on joints.

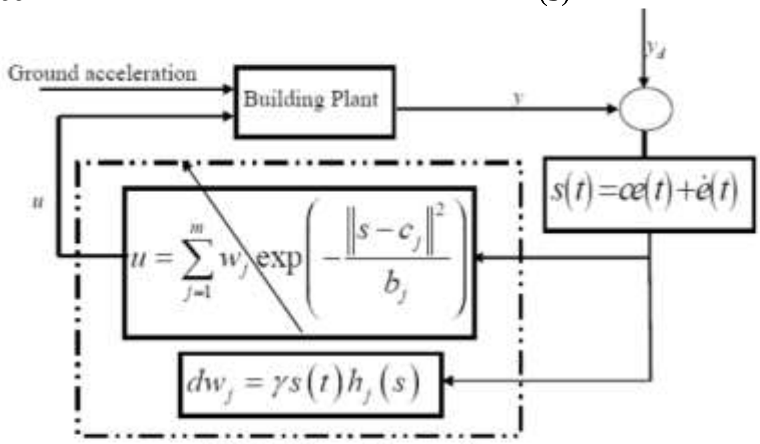
III.RBF BASED SLIDING MODEL CONTROL MODEL OF ROBOTIC MANIPULATORS

In the design of SMC for a robotic manipulator, the control objective is to drive the joint position q to the desired position q_d . So by defining the tracking error to be in the following form:

$$e = q - q_d \tag{2}$$

The siding surface can be written as

$$s = \dot{e} + ce \tag{3}$$



RBF sliding mode controller

Based on RBF neural network, the combined controller constructs the equivalent control of the sliding mode control signal (Li and Chen 1994; Onder *et al.* 2001; Yu *et al.* 1995; Huang *et al.* 2003). The sliding surface variable $s(t)$, and the control law u is achieved by RBF neural network. The excitation values of the Gaussian function in the hidden layer are distances between the input values of the sliding variables and the central positions

$$\phi_i(s) = \|s - \mu_i\| \tag{4}$$

Where, $s(t)$, is the input sliding variable and μ_i is the central position of neuron *i*. The weightings, w_i between the hidden layer and the output layer are adjusted based on an adaptive rule.Then the output of a RBFNN is also the control law which can be achieved byThe weight function and adaptive rules are used to adjust the weightings in order to search the optimal weighting values and obtain the stable convergence of the controller. With the application of the regulated weightings and adaptive rule, RBFNN can approximate the nonlinear mapping between the sliding input variable and the control law.

$$u = \sum_{i=1}^{p} w_i \exp\left(-\frac{\|s-\mu_i\|^2}{2\sigma_i^2}\right) \tag{5}$$

Based on the Lyapunov theory, the objective of the controller is to make $s(t)\dot{s}(t) \to 0$.
Hence the weight function can be adjusted by following these procedures:

$$E = s(t)\dot{s}(t) \tag{6}$$

$$\begin{aligned} dw_i &= -\eta \frac{\partial E}{\partial w_i(t)} = -\eta \frac{\partial s(t)\dot{s}(t)}{\partial w_i(t)} \\ &= -\eta \frac{\partial s(t)\dot{s}(t)}{\partial u} \frac{\partial u}{\partial w_i(t)} \end{aligned} \tag{7}$$

Since

$$\frac{\partial s(t)\dot{s}(t)}{\partial u} = s(t)\frac{\partial \dot{s}(t)}{\partial u} = -\gamma s(t)$$
$$\frac{\partial u}{\partial w_i(t)} = \exp\left(-\frac{\left\|s-\mu_i\right\|^2}{2\sigma_i^2}\right) \tag{8}$$

Hence, the learning algorithm is

$$dw_i = \gamma s(t)\exp\left(-\frac{\left\|s-\mu_i\right\|^2}{2\sigma_i^2}\right)$$
$$= \gamma s(t) h_i(s) \tag{9}$$

The weightings between the hidden and output layers neurons can be adjusted online to achievethe learning ability of RBFNN.

IV.SIMULATION

Simulation results shows that the proposed method working considerably well for different trajectories. The proposed adaptive SMC is used on a three-link scara robot, with parameter matrices given by [14]

$$M(q) = \begin{pmatrix} M_{11} & M_{12} & 0 \\ M_{21} & M_{22} & 0 \\ 0 & 0 & M_{33} \end{pmatrix}, \quad C(q,\dot{q}) = \begin{pmatrix} C_{11} & C_{12} & 0 \\ C_{21} & C_{22} & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad G(q) = \begin{bmatrix} 0 \\ 0 \\ G_3 \end{bmatrix}$$

Where

$$M_{11} = l_1^2\left(\frac{m_1}{3} + m_2 + m_3\right) + l_1 l_2 \left(m_2 + 2m_3\right)\cos(q_2) + l_2^2\left(\frac{m_2}{3} + m_3\right)$$
$$M_{13} = M_{23} = M_{31} = M_{32} = 0$$
$$M_{12} = l_1 l_2\left(\frac{m_2}{2} + m_3\right)\cos(q_2) - l_2^2\left(\frac{m_2}{3} + m_3\right) = M_{21}$$
$$M_{22} = l_2^2\left(\frac{m_2}{3} + m_3\right)$$
$$M_{33} = m_3$$
$$C_1 = l_1 l_2 \sin(q_2)$$
$$C_{11} = -\dot{q}_2 C_1\left(m_2 + 2m_3\right)$$
$$C_{12} = -\dot{q}_2 C_1\left(\frac{m_2}{2} + m_3\right) = C_{21}$$
$$C_{13} = C_{22} = C_{23} = C_{31} = C_{32} = C_{33} = 0$$
$$G_3 = -m_3 g$$

In which q_1, q_2, q_3 are the angle of joints 1,2 and 3; m_1, m_2, m_3 are the mass of the links 1,2 and 3; l_1, l_2, l_3 are the length of links 1,2 and 3; g is the gravity acceleration.The system parameters of the scara robot are selected:

$$l_1 = 1.0m; l_2 = 0.8m; l_3 = 0.6m$$
$$m_1 = 1.0kg; m_2 = 0.8kg; m_3 = 0.5kg;$$
$$g = 9.8$$

Important parameters that effect the control performance of the robotic system are the externaldisturbance $t_1(t)$, and friction term $f(\dot{q})$ .External disturbances are selected as:

$$t_1(t) = \begin{bmatrix} 5\sin(2t) \\ 5\sin(2t) \\ 5\sin(2t) \end{bmatrix}$$

Friction forces considered in these simulations as the following:

$$f(\dot{q}) = \begin{bmatrix} 12\dot{q}_1 + 0.2sign(\dot{q}_1) \\ 12\dot{q}_2 + 0.2sign(\dot{q}_2) \\ 12\dot{q}_3 + 0.2sign(\dot{q}_3) \end{bmatrix}$$

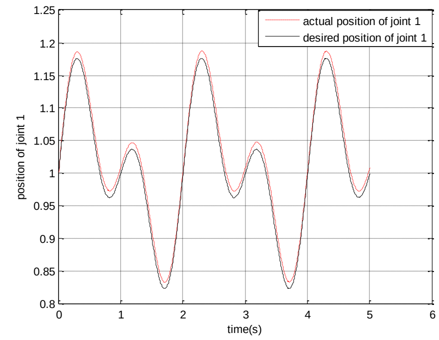
In this simulation the robot manipulator considered to carry a load of 10kg to 20kg with no prior knowledge of the weight; using 1second to 4second of the total simulation time. The desired trajectories for the three joint to be tracked are given:

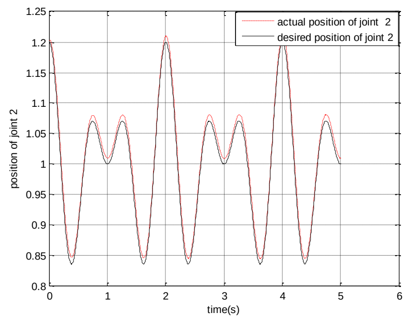
$$q_{d1}(t) = 1 + 0.1(\sin(t) + \sin(2t))$$

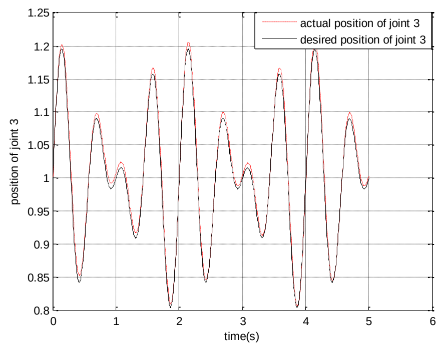
$$q_{d2}(t) = 1 + 0.1(\cos(2t) + \cos(3t))$$

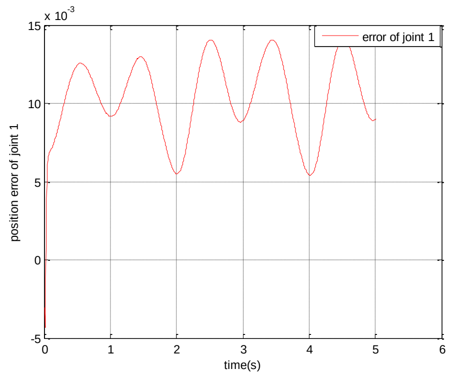
$$q_{d3}(t) = 1 + 0.1(\sin(3t) + \sin(4t))$$

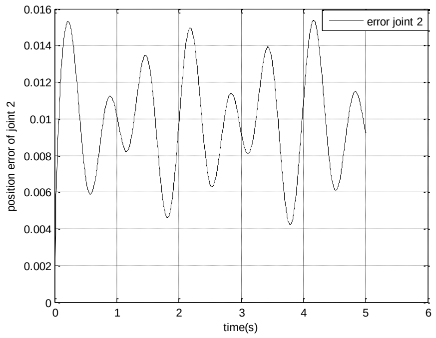
In this simulation the model is estimated by applying a random factor to the corresponding parameter matrices of the original system in each environment to count uncertainties, The control input u is chosen as in (5) $c = diag[30,30,30]$ in (3).

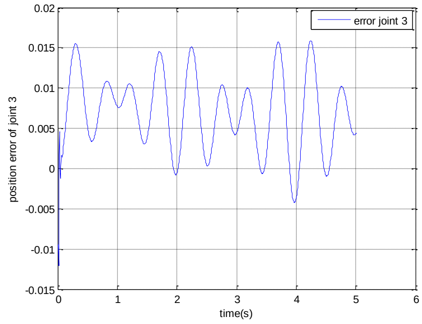












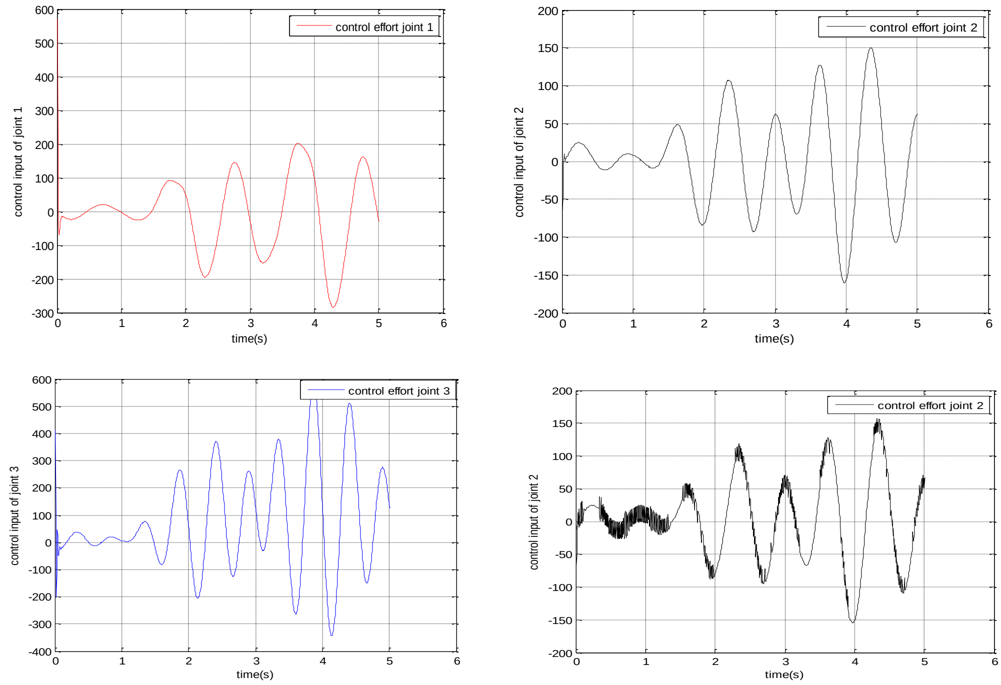


Fig 1. (a) position of joint 1 (b) position of joint 2 (c) position of joint 3 (d) position error of joint 1 (e) position error of joint 2 (f) position error of joint 3 (g) control input of joint 1 (h) control input of joint 2 (i) control input of joint 3 (j) control input of joint 2 by classical sliding mode control.

V. CONCLUSION

As shown in simulation the proposed method gives considerably much better results than classical sliding mode controller (SMC)In this paper an adaptive sliding mode controller using RBFNN neural network is proposed for robotic manipulators. The discontinuity of the classical sliding mode controller are compensated by the smooth function approximation properties of RBFNN which is nonlinear and continuous, is used to avoid the chattering. As shown in simulation the proposed RBFNN neural network can compensate the system uncertainties properly.

References

[1] J. E. Slotine, W. Li, Applied Nonlinear Control. Prentice-Hall Inc. New Jersy, 1991
[2] K. D. Yong, "Controller design for a manipulator using theory of variable structure system,".IEEE Trans.Systs, Man, Cybern., vol. SMC-8, no .2, pp. 210-218, Feb. 1978.
[3] K. S. Narendra, J. Balakrishnan, "Adaptive control using multiple models, "IEEE Trans. Automatic Control, vol. 42,no. 2, pp. 171-187, Feb. 1997.
[4] Kumpati. S. Narendra, Cheng xiang, "Adaptive control of discrete-time systems using multiple models," IEEE Trans.Automatic control, vol. 45, no. 9, pp. 1669-1686, sept. 2000.
[5] K S. Narendra, J. Balakrishnan, "Improving transient response of adaptive control systems using multiple models and switching,"IEEE Trans. Automatic Control, vol. 39, pp.1861-1866, Sept. 1994.
[6] O. Kaynak, K. Erbartur, M. Ertugrul. "The Fusion of computationally intelligent methodologies and sliding mode control-a survey,"IEEE Trans. on Industrial Electronics, vol. 48, no. 1, pp. 4-17, Feb. 2001.
[7] K. Zerenik, M. Rodic, R. Safaric, B. Curk, "Neural network sliding mode robot control, "Robotica, vol. 15, pp.23-30, 1997.
[8] M. Ertugrul, O. Kaynak, "Neuro sliding mode control of robotic manipulators,"Mechatron, vol. 10, no 1-2, pp.243-267, 2000..
[9] M. Ertugrul, O. Kaynak, "Neural network adaptive sliding mode control and its application to SCARA type robot manipulator," in Proc. IEEE ICRA'97, pp. 2932-2937,1997.
[10]R.J Wai and K.Y Hsich, "Tracking Control Design for Robot Manipulator Via Fuzzy Neural Network". Proceedings of the IEEE Int. Conference on Fuzzy Systems .Volume 2,pp. 1422-142,2002.