ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/244995561

PID969658

DATASET · JULY 2013

READS
25

4 AUTHORS, INCLUDING:

Safanah Mudheher Raafat
University of Technology, Iraq
35 PUBLICATIONS **55** CITATIONS

SEE PROFILE

Rini Akmeliawati
International Islamic University Malaysia
133 PUBLICATIONS **337** CITATIONS

SEE PROFILE

All in-text references underlined in blue are linked to publications on ResearchGate, letting you access and read them immediately.

Available from: Safanah Mudheher Raafat
Retrieved on: 08 April 2016

Improving Trajectory Tracking of a Three Axis SCARA Robot Using Neural Networks

[1] Safanah M. Raafat , [2] Waladin K. Said,[1] Rini Akmeliawati, [2] Nagham M. Tariq

[1] Mechatronics Engineering Dept. International Islamic
University Malaysia (IIUM)
Kuala Lumpur, Malaysia.
Email: safanamr@gmail.com

[2] Control and System Dept.
University of Technology
Baghdad, Iraq

Abstract— **In this paper, a neural-network based robust adaptive controller is proposed to control an industrial robot considering non- linearities, uncertainties and external perturbations. Three-axis SCARA robots is used to test the performance of this controller. The nonlinear system is treated as a partially known system. The known dynamic is used to design a nominal feedback controller based on the well-known feedback linearization method and PD controller. A Variable Structure Controller is added to the PD loop to provide robustness to uncertainties in the model of the system in order to improve accuracy of the trajectory tracking. A Neural Network (NN) based robust adaptive tracking controller is applied to further improves the control action. The outputs of the NNs are used to compensate the effects of the system uncertainties and to improve the tracking performance. Using this scheme, strong robustness with respect to uncertain dynamics and nonlinearities can be obtained, the output tracking error between the plant output and the desired reference output can asymptotically converge to zero as well. This controller exhibited superior performance characteristics where the maximum absolute error for the three-axis SCARA robot is considerably reduced.**

***Keywords*—feedback linearization; Neural Network; robust adaptive tracking controller; SCARA robot; uncertainties.**

I. INTRODUCTION

Control architectures for nonlinear affine systems may be classified according to the degree of prior knowledge available about the system's nonlinearities. When exact knowledge of the nonlinearities is available, the techniques of feedback linearization are well understood. When system's nonlinearities are given as linear combination of known functions, adaptive control methods may be used to achieve asymptotic tracking [1].

To overcome the effects of specific types of uncertainties, robust control techniques have been proposed. The design of a robust controller calls for system identification which, besides knowledge about the process to be controlled, provides some kind of quantitative knowledge about the model mismatch as well [2]. However, when large uncertainties exist, the control effort exercised by a robust controller is typically large and exhibits large chattering which may excite high frequency un-modeled dynamics.

When the uncertainties of the system cannot be linearly parameterized, and are too large to be effectively compensated for by a robust controller, none of the above methods can be applied. In such cases; a neural network approach may be effectively implemented [1]. In recent years, the neural-network-based control technique has represented an alternative method to solve the problems in control engineering [3]. The nonlinear mapping and learning properties of NNs are key factors for their use in the control field. These types of controllers take advantage of the capability of a NN for *learning* nonlinear functions and of the massive parallel computation, required in the implementation of advanced control algorithms. [4].

In this paper, a NN-based robust adaptive tracking control scheme for a three- axis SCARA robot was developed. NNs are not directly used to learn the system uncertainties, but they are used to adaptively learn the bounds of uncertain dynamics in a compact set. The outputs of the NNs then adaptively adjust the gain of the sliding mode controller so that the effects of system uncertainties can be eliminated and the output tracking error between the plant output and the desired reference signal can asymptotically converge to zero.

Since the adaptive neural learning skill and the sliding mode control technique are combined the proposed neural control scheme behaves with strong robustness with respect to unknown dynamics and nonlinearities.

The paper is organized as follows: Section II describe the SCARA robot and its dynamic model, Section III summarizes the problem formulation for control law design, in Section IV the stability analysis of the controlled system is given, a demonstration of the simulation results is given in V and finally, some conclusions is presented in Section VI.

II. SCARA ROBOT

Selective Compliant Assembly Robot Arm or Selective Compliant Articulated Robot Arm (SCARA) robots are a blend of the articulated and cylindrical robots, providing the benefits of each. A typical SCARA robot structure is shown in Fig.1. Due to its characteristics this robot is mainly used for precision, high-speed and light assembly. Common applications are: inserting components on printed circuit board, assembling small electromechanical devices and assembling computer disk drives [5]. This type of robot has a linear vertical axis, two rotary axes that move the two arm linkages in the horizontal plane, and usually one additional axis for the wrist rotation. It is compact and the working envelopes are relatively limited (ranges<1000mm). The range of payloads that can be supported by this robot is 10-100 kg.

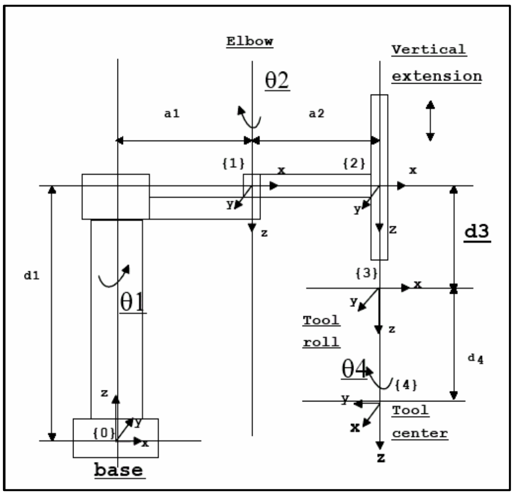


Figure.1 Configuration of SCARA robot [6].

A. Dynamic Model of SCARA Robot

The dynamic model of the three axis SCARA robot is formulated using the Lagrange-Euler [7]:

1st joint:

$$\tau_1 = [(\frac{m_1}{3} + m_2 + m_3)a_1^2 + (m_2 + 2m_3)a_1 a_2 C_2 +$$

$$(\frac{m_2}{3} + m_3)a_2^2]\ddot{q}_1 - [(\frac{m_2}{2} + m_3)a_1 a_2 C_2 + (\frac{m_2}{3} + m_3)a_2^2]\ddot{q}_2$$

$$+ b_1(\dot{q}_1) - a_1 a_2 S_2[(m_2 + 2m_3)\dot{q}_1 \dot{q}_2 - (\frac{m_2}{2} + m_3)\dot{q}_2^2] \tag{1a}$$

Since axis1 is aligned with the gravitational field, the gravity term on joint one is zero.

2nd joint:

$$\tau_2 = -[(\frac{m_2}{2} + m_3)a_1 a_2 C_2 + (\frac{m_2}{3} + m_3)a_2^2]\ddot{q}_1$$

$$+ (\frac{m_2}{2} + m_3)a_2^2 \ddot{q}_2 + (\frac{m}{2} + m_3)a_1 a_2 S_2 \dot{q}_1^2 + b_2(\dot{q}_2) \tag{1b}$$

Again, there is no gravitational loading on joint two.

3rd joint:

$$\tau_3 = m_3 \ddot{q}_3 - g_0 m_3 + b_3(\dot{q}_3) \tag{1c}$$

where τ is the control torque, m_i is mass of link, a_i is the length of link i, b_i is friction coefficient, C_i is the centrifugal and Coriolis torque, $\dot{q}_i$ is actual joint velocity and $\ddot{q}_i$ is actual joint acceleration.

Equations (1) are referred to as the inverse dynamic equations of the three-axis SCARA robot. These equations are time varying, nonlinear, and coupled differential equations. The trajectory problem of this robot manipulator revolves around finding the joint torques $\tau_i(t)$ such that the joint angles $q_i(t)$ track the desired trajectories $q_{id}(t)$.

III. PROBLEM FORMULATION FOR CONTROL LAW DESIGN

The form of the dynamic equation of an n-degree of freedom manipulator is:

$$D(q)\ddot{q} + d(q,\dot{q}) = \tau \tag{2}$$

The robot dynamic (2) is a highly nonlinear and coupled MIMO system. The control objective is to elaborate a required control law τ that forces the output q to follow their reference q_d. The perfect model of the system is difficult to be obtained, and in most practical cases, the model is not exactly known due to plant uncertainties or parameter variations, and only the nominal model of the system can be obtained. In this situation, the system parameter matrices in (2) can be expressed as follows:

$$D(q) = D_0(q) + \delta D(q) \tag{3}$$

$$d(q,\dot{q}) = d_0(q,\dot{q}) + \delta d(q)$$

where $D_0(q)$ and $d_0(q,\dot{q})$ are known parts and δD and δd are unknown. Substituting (3) in (2) results in

$$(D_0(q) + \delta D(q))\ddot{q} + (d_0(q,\dot{q}) + \delta d(q)) = \tau \tag{4}$$

A nominal feedback controller based on feedback linearization method was designed using the known dynamics.

Given the plant model in 2, a joint space inverse dynamics control is given by:

$$D(q)v + d(q,\dot{q}) = \tau \tag{5}$$

where $v \in R^n$. Since the inertia matrix $D(q)$ is invertible for all q, the closed loop system reduces to the decoupled double integrator system; which is given by

$$\ddot{q} = v \tag{6}$$

The acceleration term v is now the control input to the double integrator system.

The objective of a motion controller is to move the robot or the robot manipulator according to a desired motion trajectory q_d, while the actual motion trajectory is defined as q, the output tracking error in this case can be defined as:

$$e = q_d - q \tag{7}$$

The Brunovsky canonical form can be developed by differentiating e twice. Then the state space equation is [7]:

$$\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \hat{A}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \hat{B}\,v \tag{8}$$

where $\hat{A} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}$ and $\hat{B} = \begin{bmatrix} 0 \\ I \end{bmatrix}$

First a PD controller is designed to provide v for the nominal system for three axis SCARA robot. Then the sliding mode technique is used to obtain the control signal v.

A. PD Control

For a given joint space trajectory, q_d, the outer loop term v can be controlled by PD + feed-forward acceleration control [8]:

$$v = \ddot{q}_d + Ke \tag{9}$$

where $K = [k_p, k_d]$. The matrix K is designed such that the matrix $\hat{A}$ in 8 becomes

$$\bar{A} = \hat{A} + \hat{B}K \tag{10}$$

The gain matrix is selected as positive definite in order to keep the tracking error dynamics stable. A nominal PD feedback for v that is used to stabilize the error dynamics of the linearized system with a derivative gain matrix k_d and a proportional gain matrix k_p produce the PD computed torque controller. There will be an outer tracking loop and an inner feedback linearization loop consisting of the nonlinear component $d(q,\dot{q})$ and the nominal feedback control which are used to stabilize the error dynamics of the linearized system. The governing equation for the nominal torque is;

$$\tau_0 = d_0(q,\dot{q}) + D_0(q)(\ddot{q}_d + k_p(q_d - q) + k_d(\dot{q}_d - q)) \tag{11}$$

The tracking error dynamics is given by;

$$\ddot{e} = -k_d\dot{e} - k_p e \tag{12}$$

The PD controller should be designed to make the output tracking of the nominal system converge asymptotically to zero. It was designed using equations 9 to 12.

Considering the system expressed in equation (4) with uncertain dynamics; the control input of the closed loop system can be set as a summation of the nominal control and a compensation term as follows:

$$\tau = \tau_0 + \delta\tau \tag{13}$$

Adding equation (13) and equation (4), the global error dynamics becomes:

$$\ddot{e} + k_d\dot{e} + k_p e = D_0(q)^{-1}(\delta\tau - \delta D(q)\ddot{q} - \delta d(q)) \tag{14}$$

B. Variable Structure Control VSC

The expression for the sliding controller can be chosen as follows [9]:

$$\delta\tau = -K_v \operatorname{sgn}(s) \tag{15}$$

where K_v is the absolute value of the maximum signal control and sgn is the sign function:

$$\operatorname{sgn}(s) = +1 \qquad \text{if } s>0$$
$$\operatorname{sgn}(s) = -1 \qquad \text{if } s<0$$

By choosing K_v to be large enough, we obtain

$$\frac{1}{2}\frac{d}{dt}s^T D(q)s = -s^T K_v \operatorname{sgn}(s) \tag{16}$$

Thus, the sliding condition in equation (16) can be easily verified. Controller gains K_v can be defined as [10]:

$$K_v = \sum_{j=1}^{2}\left[\delta D_{ij}\left|\ddot{q}_{dj} - \lambda_i\dot{e}\right| + \delta C_{ij}\left|\dot{q}_j - s_j\right|\right] \tag{17}$$

where δD_{ij} and δC_{ij} are the known bounded uncertainties of the inertia matrix and the centrifugal and coriolis torques, respectively and λ is a positive diagonal constant matrix.

C. Neural Network Approximation of the Unknown Functions

The control architecture of the NN based robust adaptive tracking controller employs NNs to approximate the unknown functions $\delta D(x)$ and $\delta d(x)$. One hidden layer NNs are considered, with general structure in the form $\sum_{i=1}\theta_i\omega_i(x)$

where the output is linearly dependent on the parameters. This choice is motivated by the use of the robust adaptive control to prove the stability of the closed loop system. RBF networks are universal approximators of any continuous functions in regression and classification. Its architecture is a feed-forward multilayer type network, so it has good approximation capabilities and has one hidden layer with nonlinear inputs. Each hidden (kernel) node computes the distance between the input vector and the center of the corresponding RBF. It has an output layer of linear nodes which compute the sum of the weighted outputs from the RBF nodes [10].

The following RBF NNs are used to adaptively learn the uncertain bounds [11]:

$$\widehat{\delta D}(x) = \hat{\theta}_D^T(x)W_D(x) \tag{18a}$$

$$\widehat{\delta d}(x) = \hat{\theta}_d^T(x)W_d(x) \tag{18b}$$

where $\hat{\theta}_D^T(x)$ and $\hat{\theta}_d^T(x)$ are the weight vectors of the RBF NNs which are computed by an appropriate adaptation algorithm. The vectors $W_D(x)$ and $W_d(x)$ are Guassian type of functions defined as:

$$W_{D_i}(x) = \exp(-\frac{\left\|x - \mu_{D_i}\right\|^2}{\sigma_{D_i}^2}) \qquad i = 1,2,...,n_D \tag{19a}$$

$$W_{d_i}(x) = \exp(-\frac{\left\|x - \mu_{d_i}\right\|^2}{\sigma_{d_i}^2}) \qquad i = 1,2,...,n_d \tag{19b}$$

where μ is the center of the receptive field and σ is the width of Guassian function.

* *Assumption*: the assumption considered is that there exist two optimal weight vectors θ_D^* and θ_d^*, such that the outputs of the optimal NNs with enough nodes satisfy:

$$\delta D(x) = \theta_D^{*T}(x)W_D(x) + \varepsilon_D \tag{20a}$$

$$\delta d(x) = \theta_d^{*T}(x)W_d(x) + \varepsilon_d \tag{20b}$$

where, ε_D *and* ε_d are the functional reconstruction error. This assumption reflects the approximation capability of NNs. It should be noted that θ_D^* and θ_d^* obviously exist and explicit expression for their computation is not required since this value can be learned by using an adaptive algorithm. So, the objective is to use the RBF NNs in (20) to learn the uncertain bounds $\delta D(q)$ and $\delta d(q)$. The outputs of the NNs are then used as the parameters of the controller so that the output tracking error can asymptotically converge to zero and strong robustness with respect to uncertain dynamics can be guaranteed.

D. Design of a Neural-Network-Based Controller

For the design of the NN based controller, the adjustment of the weights, and the analysis of the error convergence, we consider the state form of (1) with assumption above. If the control input is designed such that [11]:

$$\delta\tau = \hat{\theta}_D^T(x)W_D(x)y^{(n)} + \hat{\theta}_d^T(x)W_d(x) - K_v \operatorname{sgn}(s) \quad (21)$$

where $K_v = \bar{\varepsilon}_D \mid y^{(n)} \mid + \bar{\varepsilon}_d + D_0^{-1}\varsigma$ (22)

$\bar{\varepsilon}_D$ and $\bar{\varepsilon}_d$ are respectively the upper bounds of $|\varepsilon_D|$ and $|\varepsilon_d|$, ζ is a small positive constant and the weight vectors are adjusted by using the following adaptive mechanisms [11]:

$$\dot{\hat{\theta}}_D = -\eta_D W_D(x) y^{(n)} s^T \quad (23a)$$

$$\dot{\hat{\theta}}_d = -\eta_d W_d(x) s^T \quad (23b)$$

with adaptive gains $\eta_D > 0$, $\eta_d > 0$, then the output tracking error asymptotically converges to zero.

The robustness properties of the NN based adaptive controller in (21) may be summarized as follows:

1) When the output tracking error is large due to the effects of system uncertainties, the outputs of the RBF NNs are adaptively increased according to the update laws in (23). The control gain can then be increased to eliminate the effects of uncertain dynamics, and drive the switching plane variable s to the sliding mode. In the sliding mode, the output tracking error asymptotically converges to zero.

2) The error dynamics of the closed-loop system are only determined by the sliding mode parameters and are insensitive to system uncertainties and bounded disturbances in the sliding mode.

3) It can be seen from (21) that the controller does not require any knowledge of the controlled nonlinear system. Only the outputs, tracking error and its derivative are used for the design of the controller though the system has nonlinearities and parameter uncertainties.

From (21) it can be also seen that the NNs used are essential to realize the nonlinear adaptive control law because the uncertainty bounds are unknown nonlinear functions. However, if these bounds are constants, the design of the controller and adaptive laws can be greatly simplified without using NNs. The overall structure of the controller is shown in Fig. 2.

IV. STABILITY ANALYSIS

The following development investigates the stability of the NN based controller. The Lypunov approach is used to prove the stability of the closed loop system. Define the parameters errors:

$$\tilde{\theta} = \hat{\theta} - \theta^* \text{ then } \dot{\tilde{\theta}} = \dot{\hat{\theta}}$$

After introducing the compensator given by 21, the expression of the error dynamics will be:

$$\dot{e} = Ae + BD_0^{-1}(\tilde{\theta}_D^T W_D y^{(n)} + \tilde{\theta}_d^T W_d ... - K_v \operatorname{sgn}(s) - \varepsilon) \quad (24)$$

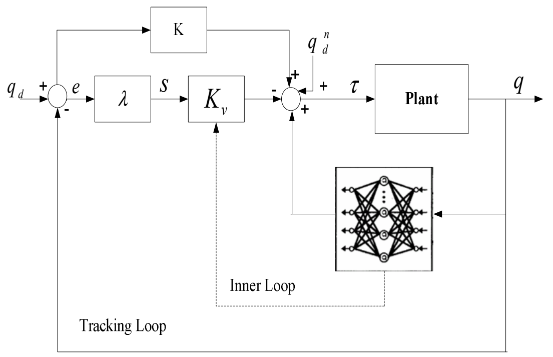


Figure 2. Block Diagram of the Control System

where $\varepsilon = \varepsilon_D y^{(n)} + \varepsilon_d$, $A = \begin{bmatrix} 0 & I \\ k_p & k_d \end{bmatrix}$ and $B = \begin{bmatrix} 0 \\ I \end{bmatrix}$

Proof: Defining a Lyapunov function

$$V = \frac{1}{2}e^T Pe + \frac{1}{2}(\tilde{\theta}_D^T \eta^{-1} \tilde{\theta}_D) + \frac{1}{2}(\tilde{\theta}_d^T \eta^{-1} \tilde{\theta}_d) \quad (25)$$

where $P \in R^{m.n \times m.n}$ is symmetric positive definite matrix solution of the Lyapunov equation

$$A^T P + PA = -Q$$
$$PB = C^T \quad (26)$$

$Q \in R^{m.n \times m.n}$ is symmetric positive definite matrix.

The time derivative of the Lyapunov function in equation (25) is expressed in the following equation:

$$\dot{V} = \frac{1}{2}e^T[A^T P + PA]e + s^T D_0^{-1}\tilde{\theta}_D^T W + s^T D_0^{-1}\tilde{\theta}_d^T W$$
$$-s^T D_0^{-1}(K_v \operatorname{sgn}(s) + \varepsilon) + (\tilde{\theta}_D^T \eta^{-1} \dot{\tilde{\theta}}_D) + (\tilde{\theta}_d^T \eta^{-1} \dot{\tilde{\theta}}_d) \quad (27)$$

Using the expression of the error dynamic in equation (24) the final form of $\dot{V}$ will be:

$$\dot{V} = \dot{V}_1 + \dot{V}_2 + \dot{V}_3 + \dot{V}_4 \quad (28)$$

where

$$\dot{V}_1 = -\frac{1}{2}e^T Qe$$

$$\dot{V}_2 = s^T D_o^{-1}\tilde{\theta}^T W + (\tilde{\theta}_D^T \eta^{-1} \dot{\tilde{\theta}}_D)$$

$$\dot{V}_3 = s^T D_o^{-1}\tilde{\theta}^T W + (\tilde{\theta}_d^T \eta^{-1} \dot{\tilde{\theta}}_d)$$

$$\dot{V}_4 = -s^T D_o^{-1}\varepsilon - s^T D_o^{-1} K_v \operatorname{sgn}(s)$$

The first component $\dot{V}_1 \le 0$ is negative since Q is by definition symmetric positive definite. The adaptation algorithm of the networks parameters in (23) makes $\dot{V}_2 = 0$ and $\dot{V}_3 = 0$.

The expression of the sliding gain of (22) can be written as:

$$K_v \ge |\bar{\varepsilon}| + D_0^{-1}\mu$$

Then

$$\dot{V}_4 \le s^T D_0^{-1}(\varepsilon - |\bar{\varepsilon}| \operatorname{sgn}(s)) - \mu s^T \operatorname{sgn}(s)$$

From this it follows that $\dot{V}_4 \leq -\mu \sum_{i=1}^{m} |s_i|$

As a results the time derivative of the Lyapunov functions will be negative ($\dot{V} \leq 0$). V is then decreasing with time along the system's trajectory. So e and all the system's signals are bounded. It can be seen that the weights of the RBF NNs are adjusted in Lyapunov sense. Therefore, it is not necessary for the weights of the NNs to converge to their optimal values, but the values of the weights are adaptively increased until the switching plane variable s converges to zero. Then the weights will become constants to guarantee that the output tracking error asymptotically converges to zero in the sliding mode.

V. SIMULATION AND RESULTS

The simulation was carried out using MATLAB with a fourth-order Runge-Kutta algorithm and sampling time equal to 0.01s. The parameters of the three-axis manipulator used to generate data in the simulation are listed in Table 1. The default values of PD controller gains are selected according to the desired closed loop poles. The selection of the closed loop poles was based on a desired settling time of the order 0.3 to 0.4 seconds approximately [11]. The control law as expressed by 22 will be used with the controller's parameters set as shown in Table 2.

Figure 3 shows the actual and the desired trajectory for the three types of controllers that were used. It is clear that Fig.3 (c) with NNs has the best tracking performance. The comparison between the three controllers' errors and control signals for the trajectory of Fig. 3 is given in Table 3. Using NNs causes a reduction in the maximum absolute values and the chattering of the control inputs where it is reduced to 19.08Nm for joint1, 16.47Nm for joint2 and 44.73N for joint3 as given in Table 3.

The results of some other set of experiments are presented in Table 4. Different initial conditions of Gaussian functions, load and nonlinearities were implemented. The effect of changing robot load to $m_L = 1.5$ Kg with friction is also shown in Table 4; the maximum absolute value of the control signal for joint3 is increased to 59.72N to counteract the effect of gravity to stop the fall of the arm. It is also obvious from Table 4 that changing the "sign" function in the control input to "saturation" function with ε=0.1, 0.1, and 0.2 for joints 1,2 and 3, respectively, will eliminate the chattering and reduce the amplitude of the control signal while increasing the tracking error .

TABLE 1.

PARAMETER VALUES OF THE THREE-AXIS SCARA ROBOT [6]

joint No.	1	2	3
mass (kg)	8	5	3
Link length (m)	0.8	0.3	0.3
coefficient of viscous friction (Nm s/rad, Nm s/rad, N s/m)	1	1	1
Coefficient of dynamic friction (Nm, Nm , N)	0	0	0
coefficient of static friction (Nm,Nm,N)	15	15	15
ε (rad/s, rad/s,m/s)	0.1	0.1	0.1

TABLE 2.

THE CONTROLLER PARMETERS FOR THE THREE-AXIS SCARA ROBOT

η_{D1}	η_{D2}	η_{D3}	η_{d1}	η_{d2}	η_{d3}
5	5	5	5	5	5
$\mu 1$	$\mu 2$	$\mu 3$	$\bar{\varepsilon}_D$	$\bar{\varepsilon}_d$	
0.15	0.15	0.15	0.005	0.005	
x = $[q_1\ q_2\ q_3\ \dot{q}_1\ \dot{q}_2\ \dot{q}_3]$					
$e_1 = q_{d1}-q_1$	$e_2 = q_{d2}-q_2$	$e_3 = q_{d3}-q_3$			
$s_1 = 20e_1 + \dot{e}_1$	$s_2 = 20e_2 + \dot{e}_2$	$s_3 = 20e_3 + \dot{e}_3$			

TABLE 3.

COMPARISON BETWEEN THE THREE CONTROLLERS' ERRORS AND CONTROL INPUT

Controller Type	Error		
	joint1 (rad)	Joint2 (rad)	joint3 (m)
PD	0.02	0.05	0.01
PD+VS	0.01	0.04	$4*10^{-3}$
PD+VS+NN	$4*10^{-3}$	$4*10^{-3}$	$3*10^{-5}$
Controller Type	Control Input		
	Joint 1 (Nm)	Joint 2 (Nm)	Joint 3 (N)
PD	25.95	15.42	52.94
PD+VS	32.29	16.82	103.10
PD+VS+NN	19.08	16.47	44.73

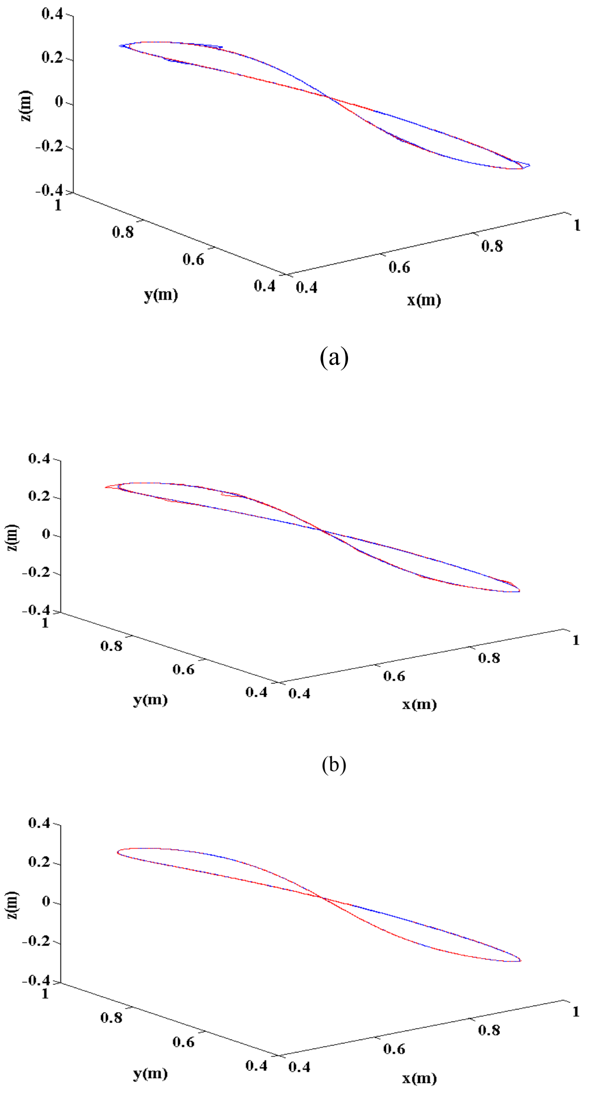
TABLE 4-a.

RESULTS OF SIMULATION FOR THREE- AXIS PLANAR ROBOT

Function	Error		
	joint1 (rad)	joint2 (rad)	joint3 (m)
	Max.	Max.	Max.
sign, m_L=0kg, σ=0.84, 0.5 (initial), 10 nodes	$3*10^{-4}$	$4*10^{-3}$	$4*10^{-6}$
sign, m_L=0kg, σ=0.25, 0.5 (initial), 20 nodes	$4*10^{-3}$	$4*10^{-3}$	$5*10^{-6}$
sign, m_L=1.5kg, σ=0.25 0.5 (initial), 20 nodes	$1*10^{-5}$	$4*10^{-3}$	$8*10^{-6}$
saturation, m_L=0kg, σ=0.25 0.5 (initial),20 nodes	$5*10^{-3}$	$4*10^{-3}$	$2*10^{-5}$

TABLE 4-b.

Function	Control Input		
	joint1 (Nm)	joint2 (Nm)	joint3 (N)
	Max.	Max.	Max.
sign, m_L=0kg, σ=0.84, 0.5 (initial), 10 nodes	143.30	17.38	44.93
sign, m_L=0kg, σ=0.25, 0.5 (initial), 20 nodes	19.08	16.47	44.73
sign, m_L=1.5kg, σ=0.25 , 0.5 (initial), 20 nodes	18.78	16.39	59.72
saturation, m_L=0kg, σ=0.25, 0.5 (initial), 20 nodes	18.87	15.68	44.75



(a)

(b)

(c)

Figure 3. The actual and the desired trajectory for the three types of controller with k_p=200, k_d=25 and m_L=0kg and "sign" function. (a) PD controller. (b) PD+VS controller, and (c) PD+VS+ NNs.

VI. CONCLUSIONS

NN-based robust adaptive tracking control scheme for a SCARA robot was developed. NNs are not directly used to learn the system uncertainties, but they are used to adaptively learn the bounds of uncertain dynamics in a compact set. The outputs of the NNs then adaptively adjust the gain of the sliding mode controller so that the effects of system uncertainties can be eliminated and the output tracking error between the plant output and the desired reference signal can asymptotically converge to zero. Since the adaptive neural learning skill and the sliding mode control technique are combined, the proposed neural control scheme behaves with strong robustness with respect to unknown dynamics and nonlinearities. The stability of the closed loop system and convergence towards zero of the tracking error is guaranteed. The effects of the width of the Guassian functions, initial values of the weight vectors, and the number of nodes on the system performance were investigated as well. The simulation results have shown good tracking performance when adding NNs to the control scheme where the tracking error was found to be less than 4x10^{-3}rad for joints1 and 2 and 0.03mm for joint3.

REFERENCES

[1] Wams, B., Botto, A., van den Boom, T. J. J. and Costa, J. Sa da "Training Neural Networks For Robust Control of Nonlinear MIMO Systems", In Proceedings of The International Conference on Control 1998, pages 141-149, Swansea, UK, September, 1998.

[2] Patiño, H. D., Caelli, R. and Kuchen, R. "Neural Networks for Advanced Control of Robot Manipulators", IEEE Transactions on Neural Networks, Vol. 13, No. 2, March 2002.

[3] Ge, S. S. and Lee, T. H. "Robust Model Reference Adaptive Control of Robots Based on Neural Network Parameterization", Proceedings of American Control Conference, Albuquerque New Mexico, June 1997.

[4] Riedmiller, M. and Janusz, B. "Using Neural Reinforcement Controllers in Robotics", Proceeding of the AI95, Conberra, Australia, 13-18th, Nov. 1995.

[5] Dr. Bob, "Advanced Robotics I Class Notes", Mechanical Engineering Ohio University.

[6] Spong, M. W, Hutchinson, S. and Vidyasagar, M. "Robot Dynamics and Control", John Wiley & Sons, Inc., New York, Second Edition, 2004.

[7] http://www.ent.ohiou.edu/~bobw/PDF/IntroRob.pdf.

[8] Keleher, P. G. and Stoniery, R. J. "Adaptive terminal sliding mode control of a rigid robotic manipulator with uncertain dynamics incorporating constraint inequalities", Anziam J. 43 (E) PP E102-E153, 2002.

[9] Spong, M. W. "Robust and Adaptive Motion Control of Manipulators", University of Illinois at Urbana-Champaign, 2000.

[10] Garcia, A., De Pieri, E.R., Guenther, R. "Experimental study applied to an industrial robot by using variable structure controllers and friction compensation", J. Braz. Soc. Mech. Sci, Vol.24, No.4, Rio de Janeiro, Nov., 2002.

[11] Tariq,N.M.,"Improving Trajectory Tracking of SCARA Robot Using Neural Networks",M.Sc. Thesis,Control and Systems Dept., University of Technology, Baghdad,Iraq,2005.