Neural Network-Based Adaptive Control of Robotic Manipulator: Application to a Three Links Cylindrical Robot

Abdul-Basset A. AL-Hussein
Electrical Engineering Department
Basrah University
Basrah, Iraq.
email: abdulbasset.alhusein@gmail.com

Abstract:
A composite PD and sliding mode neural network (NN)-based adaptive controller, for robotic manipulator trajectory tracking, is presented in this paper. The designed neural networks are exploited to approximate the robotics dynamics nonlinearities, and compensate its effect and this will enhance the performance of the filtered error based PD and sliding mode controller. Lyapunov theorem has been used to prove the stability of the system and the tracking error boundedness. The augmented Lyapunov function is used to derive the NN weights learning law. To reduce the effect of breaching the NN learning law excitation condition due to external disturbances and measurement noise; a modified learning law is suggested based on e-modification algorithm. The controller effectiveness is demonstrated through computer simulation of cylindrical robot manipulator.

***Index Terms*— Neural Network, Adaptive Control, Cylindrical Robot, e-modification, Sliding Mode Controller.**

I. INTRODUCTION

The development of efficient control algorithm for robot manipulators has been recently the object of considerable interest. Where the robotic manipulators have become increasingly important in the high-speed and high-precision trajectory tracking applications. The difficult of the manipulator control problem is mainly reflect of manipulator dynamics itself. The robotics arms are subject to structured and / or unstructured uncertainties [1]. The structured uncertainty rises due to the uncertainty in the robotic arm parameters, unknown loads and actuator torque constant. While the unstructured uncertainty is called to be the case of unmodeled dynamics, such that neglecting time delay, friction and so on [1]. So, that the performance of the standard control techniques such as computed torque or inverse dynamics methods which are very sensitive to parametric uncertainties will not provide adequate performance [2]. Moreover, even that the adaptive control schemes can effectively compensate the effect of the structured uncertainties but it is not overcome the unstructured uncertainties effect [1]. Recently, neural network (NN) used to comply with control of unknown dynamics nonlinear system, since it can approximate any continuous function over a compact set, and their inherent parallel distributed structure [3] combined with the potential ability to learn, this is make the NN a powerful real time control tool [4].

In [5] a decentralized adaptive NN sliding mode controller has been proposed where a BP neural network is trained with sampled input-output data to learn the reconfigurable manipulator nonlinear dynamics then the gradient decent algorithm is used for high precision tuning. In [6] authors proposed stable controller for robotics manipulator based on combining non-singular terminal sliding mode controller and Radial basis function neural network (RBFNN). The NN are used to approximate algorithm parameters and attenuate the chattering effect of the sliding mode controller. In [7] a two-joint rigid manipulator controlled using a combined backstepping technique and radial basis function neural network controller, the NN employed to approximate the system unknown dynamics along with the actuator dead zone effect. H. Xin and C. Qiang [8] developed a controller for two-link planar robot manipulator using full-order terminal sliding mode control with neural network employed to firstly approximate then compensate the effect of the dead zone together with the system uncertainties.

In this paper, a combined PD and sliding mode neural network-based adaptive tracking controller, for robotic manipulator trajectory tracking, is presented. The designed neural networks are employed to approximate the robotics dynamics nonlinearities, and compensate its effect which will enhance the performance of the filtered error based PD and sliding mode controller. Lyapunov theorem has been used to prove the stability of the system and the tracking error boundedness. The augmented Lyapunov function is used to derive the sliding mode gain adaptation law and the NN weights learning law. To reduce the effect of breaching the NN learning law excitation condition due to external disturbances and measurement noise; a modified learning law is suggested based on e-modification algorithm. The controller effectiveness is demonstrated through computer simulation applied to three links cylindrical robot manipulator.

In section **II** the robot system is described and the dynamic model is established. The neural network controller is designed in section **III**. The simulation results are shown in section **IV** and final conclusions are given in section **V**.

II. ROBOTICS MATHEMATICAL MODEL

In this paper we will consider the cylindrical geometry robot which linearly move in two directions and can rotate in one direction. Since this type of robotic arm moves in two separate prismatic directions and rotates in one direction it is given the designation of R2P, where R stands for rotational. This type of robot manipulator is mostly used in assembly, machine tending, material handling and palletizing operations. Robotic arms with cylindrical geometry move linearly in two directions vertical motion is called stroke, while horizontal motion is called reach and rotational motion is called swing. Cylindrical geometry robots have a cylindrical work envelope, the advantages of this type of work envelope is a robot arm reach its deep at both the top and bottom of the stroke. Additionally, the robot structure allows for quick movements with high repeatability with a small use of proper space and a larger payload capacity due to structural rigidity.
The manipulator motion is represented by a mathematical model as follows

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

where $\boldsymbol{q},\ \dot{\boldsymbol{q}},\ \ddot{\boldsymbol{q}} \in \Re^n$ are the vectors of the generalized displacements of the robotic manipulator. $M(\boldsymbol{q}) \in \Re^{n*n}$ is the inertia matrix which is symmetric and positive definite for all $\Re^n$ $\boldsymbol{q}$ vectors. $C(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \Re^n$ the Coriolis and centrifugal forces. $G(\boldsymbol{q}) \in \Re^n$ denotes the gravitational torques. $\boldsymbol{u}(t) \in \Re^n$ represents the generalized control input torques. As for the requirements of the upcoming development, the following two properties and definition are presented

Prop. **(I)**: $M(\boldsymbol{q})$ matrix is assumed to be a symmetric and positive definite such that:

$$M_m \le \|M\,(\boldsymbol{q})\| \le M_M$$

where M_m and M_M are known lower and upper bound constants.

Prop. **(II)**: $C(\boldsymbol{q},\dot{\boldsymbol{q}})$ satisfy the following
$\dot{M}\,(\boldsymbol{q}) - 2\,C\,(\boldsymbol{q},\dot{\boldsymbol{q}})$ is skew symmetric;
$\|C\,(\boldsymbol{q},\dot{\boldsymbol{q}})\| \le C_M\,\|\dot{\boldsymbol{q}}\|$ and $C\,(\boldsymbol{q},\boldsymbol{x})\boldsymbol{y} = C\,(\boldsymbol{q},\boldsymbol{y})\,\boldsymbol{x}$, $\boldsymbol{x},\boldsymbol{y} \in \Re^n, C_M > 0$ where C_M is the upper bound of C.
Defin. **(I):**

$$\begin{cases} \|\boldsymbol{x}\|_1 = \sum_i^n |x_i| \\ \|A\|_F^2 = tr(AA^T) \end{cases} \tag{2}$$

III. NN-BASED CONTROLLER DESIGN

To achieve a good trajectory tracking controller one can use the sliding mode and PD controller along with the NN great approximation feature to compensate the nonlinearities in the robotic dynamic model. By defining the tracking error e and the filtered error $\boldsymbol{e_f}$ as follows [9]:

$$\boldsymbol{e} = \boldsymbol{q} - \boldsymbol{q}_d \tag{3}$$

$$\boldsymbol{e}_f = \dot{\boldsymbol{e}} + \Lambda\, \boldsymbol{e} \tag{4}$$

where $\boldsymbol{q}_d \in \Re^n$ are the desired trajectory in the joint space. Λ is positive definite design parameter. By using (1), (3) and (4), the dynamics of the manipulator can be represented in term of the filtered tracking error as follow:

$$\dot{\boldsymbol{e}_f} = \left(-\ddot{\boldsymbol{q}}_{\boldsymbol{d}} + \Lambda\, \mathbf{e}_f\right) - M^{-1}[C\,(\boldsymbol{q},\dot{\boldsymbol{q}})\,\dot{\boldsymbol{q}} + G\,(\boldsymbol{q}) + F\,(\boldsymbol{q},\dot{\boldsymbol{q}})] + M^{-1}\,u(t) \tag{5}$$

$$\dot{\boldsymbol{e}_f} + r\,\boldsymbol{e}_f = \left(-\ddot{\boldsymbol{q}}_{\boldsymbol{d}} + \Lambda\, \mathbf{e}_f\right) - M^{-1}[C\,(\boldsymbol{q},\dot{\boldsymbol{q}})\,\dot{\boldsymbol{q}} + G(\boldsymbol{q}) + F\,(\boldsymbol{q},\dot{\boldsymbol{q}})] + r\,\boldsymbol{e}_f + M^{-1}\,u(t) \tag{6}$$

where r is related to the bandwidth of the closed-loop control system.
multiply M to both sides of (6):

$$M\tilde{\boldsymbol{e}}_f = M\left(-\ddot{\boldsymbol{q}}_{\boldsymbol{d}} + \Lambda\, \mathbf{e}_f\right) - [C\,(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + G\,(\boldsymbol{q}) + F\,(\boldsymbol{q},\dot{\boldsymbol{q}})] + \boldsymbol{M}\,r\,\boldsymbol{e}_f + u(t) \tag{7}$$

where

$$\tilde{\boldsymbol{e}}_f = \dot{\boldsymbol{e}_f} + r\boldsymbol{e}_f \tag{8}$$

Then (7) can be written as:

$$M\tilde{\boldsymbol{e}}_f = R + u(t) \tag{9}$$

where

$$R(\boldsymbol{x}) = M\left(-\ddot{\boldsymbol{q}}_{\boldsymbol{d}} + \Lambda\, \mathbf{e}_f\right) - [C\,(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + G\,(\boldsymbol{q}) + F\,(\boldsymbol{q},\dot{\boldsymbol{q}})] + \boldsymbol{M}\,r\boldsymbol{e}_f \tag{10}$$

Then, the input vector can be defined as:

$$\boldsymbol{x} = [\boldsymbol{q}_d^T\, \dot{\boldsymbol{q}}_d^T\, \ddot{\boldsymbol{q}}_d^T\, sgn(\dot{\boldsymbol{q}})] \tag{11}$$

The following NN is used to approximate (9)

$$M\tilde{\boldsymbol{e}}_f = \hat{R}(\boldsymbol{x}) + u(t) \tag{12}$$

where $\hat{R}\,(\boldsymbol{x}) \in \Re^n$, defined as the nonlinear component of the NN. By considering the following control law for the robot (1)

$$u(t) = -\hat{R}\,(\boldsymbol{x}) + u_n(t) \tag{13}$$

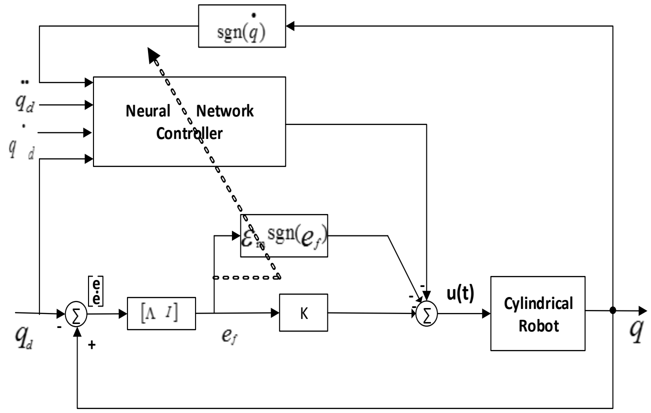

Fig. (1) The Proposed NN-Based Adaptive Controller Structure

where $u_n(t)$ represents the nonlinear control part, which is including a PD control component and sliding mode control:

$$u_n(t) = -K\mathbf{e}_f - \varepsilon_m\, sgn(\mathbf{e}_f) \tag{14}$$

where $K \in \Re^{n*n}$ positive diagonal PD controller gains matrix, and $\varepsilon_m \geq 0$ is sliding mode bound, which is usually an unknown quantity, and will be estimated through the estimating law. The overall controller structure diagram is shown in Fig. (1). By defining $\hat{\varepsilon}$ as the estimation of ε_m, then the estimation error $\tilde{\varepsilon}$:

$$\tilde{\varepsilon} = \varepsilon_m - \hat{\varepsilon} \tag{15}$$

So that (9) can be written as

$$M\tilde{\boldsymbol{e}}_f = 0 \tag{16}$$

Since, the system behavior of the NN can be determined by $\tilde{\boldsymbol{e}}_f = 0$, i.e.

$$\tilde{\boldsymbol{e}}_f = \dot{\boldsymbol{e}_f} + r\,\boldsymbol{e}_f = 0 \tag{17}$$

In this case the NN acts as a feedforward controller and in the feedback control loop, there are a PD controller $K\mathbf{e}_f$ and a sliding controller $\varepsilon_m\, sgn(\mathbf{e}_f)$. The sliding mode controller term is included to enhance system robustness against the NN approximation error. Since the approximation error could be as small as required, by selecting the NN structure carefully, the effect of the sliding mode term can be designed as small as possible i.e.

$$R_i = \theta_{ij}^* \, y_\psi + \gamma_i \tag{18}$$

where

$y_\psi = \left[y_{\psi 1}, \dots \dots, y_{\psi n_\psi}\right]^T \in \Re^{n_\psi}$ represent the basis function vector, $\theta_{ij}^* \in \Re^{n_\psi}$ are the optimal weights selected as the values of $\hat{\theta}_{ij}$ that will minimize the error in neural network approximation γ_i for all $q \in \Omega_1, \dot{q} \in \Omega_2$, where $\Omega_1 \in \Re^n$, $\Omega_2 \in \Re^n$ are appropriate compact sets. The following NN function component is defined as:

$$\hat{R}_i = \hat{\theta}_{ij} \, y_\psi \tag{19}$$

with $\hat{\theta}_{ij} \in \Re^{n_\psi}$ defined as the NN weights estimates. Then, the closed loop dynamic equation can be rewritten as:

$$M(q)\dot{\boldsymbol{e}}_f = -M(q)\, r\boldsymbol{e}_f + \widetilde{\Theta}^T \boldsymbol{Y} + \varepsilon - K\mathbf{e}_f - \varepsilon_m \, sgn(\mathbf{e}_f) \tag{20}$$

where $\widetilde{\Theta} = \Theta^* - \widehat{\Theta} = [\widetilde{\boldsymbol{\theta}}_1, \dots, \widetilde{\boldsymbol{\theta}}_n] \in \Re^{n_\psi * n}$ and $\varepsilon = [\varepsilon_1, \dots, \varepsilon_n]^T \in \Re^n$, $\varepsilon_i = \sum_{i=1}^n \gamma_i$ and $\boldsymbol{Y} \in \Re^{n_\psi}$ nonlinear activation function.

One can derive the learning law for the NN and the adaptation law of the bound estimate beside studying the convergence of the closed loop filtered error through the following theorem.

***Theorem* (I)**

For the 3-link cylindrical robot manipulator described by the dynamics model presented by (1) and the filtered error dynamic equation (20) and the control law (13) and (14):

If the NN learning algorithm is selected as follow:

$$\dot{\widehat{\Theta}} = \eta \, \boldsymbol{Y}\big(\boldsymbol{q}_d, \dot{\boldsymbol{q}}_d, \ddot{\boldsymbol{q}}_d, sgn(\dot{\boldsymbol{q}})\big) \, \boldsymbol{e}_f^T \tag{21}$$

And the bound estimate adaptation law as:

$$\dot{\hat{\varepsilon}} = \eta_\varepsilon \, [\|\mathbf{e}_f\|_1 + \sigma(\varepsilon_o - \hat{\varepsilon})] \tag{22}$$

where $\eta \in \Re^{n_\psi * n_\psi}$ is a constant positive definite matrix represent the learning rate. $\eta_\varepsilon, \sigma > 0, \varepsilon_o$ are learning design parameters. Then the closed loop system is asymptotically stable and $\lim_{t\to\infty} \tilde{q} = 0$ and $\lim_{t\to\infty} \dot{\tilde{q}} = 0$

where

$$\tilde{q} = q - q_d$$
$$\dot{\tilde{q}} = \dot{q} - \dot{q}_d \tag{23}$$

If K_m the minimum eigenvalue of K that satisfies:

$$\tfrac{1}{2}\|\dot{M}(q)\| \le K_m \tag{24}$$

Then the considered filtered error of the cylindrical robot manipulator is bounded [10].

Proof

A Lyapunov function is defined as follows:

$$L(\mathbf{e}_f, \widetilde{\Theta}, \tilde{\varepsilon}) = \tfrac{1}{2}\boldsymbol{e}_f^T M(q)\mathbf{e}_f + \tfrac{1}{2}\boldsymbol{e}_f^T \mathbf{e}_f + \tfrac{1}{2} tr(\tilde{\theta}^T \eta^{-1} \tilde{\theta}) + \tfrac{1}{2}\eta_\varepsilon^{-1}\tilde{\varepsilon}^2 \tag{25}$$

Which satisfy

$$\frac{1}{2} P_m y^T y \le L(\mathbf{e}_f, \tilde{\theta}, \tilde{\varepsilon}) \le \frac{1}{2} P_M y^T y$$

where

$$y = [\boldsymbol{e}_f^T, \boldsymbol{e}_f^T, \|\tilde{\theta}\|, \tilde{\varepsilon}]^T$$
$$P_m = \min\{M_m, 1, \eta_M^{-1}, \eta_\varepsilon^{-1}\}$$
$$P_M = \max\{M_M, 1, \eta_m^{-1}, \eta_\varepsilon^{-1}\}$$

Differentiating (25) w.r.t. time yields:

$$\dot{L}(\mathbf{e}_f, \tilde{\theta}, \tilde{\varepsilon}) = \tfrac{1}{2}\boldsymbol{e}_f^T \dot{M}(q)\mathbf{e}_f + \boldsymbol{e}_f^T[-M(q)r\mathbf{e}_f + \tilde{\theta}^T \boldsymbol{Y}(x) + \varepsilon + u_n(t)] - \boldsymbol{e}_f^T r\mathbf{e}_f - tr\left(\tilde{\theta}^T \eta^{-1} \dot{\hat{\theta}}\right) - \eta_\varepsilon^{-1}\tilde{\varepsilon}\dot{\hat{\varepsilon}} \tag{26}$$

$$\begin{aligned} &\le -\left(K_m - \frac{1}{2}\|\dot{M}(q)\|\right)\|\mathbf{e}_f\|^2 - M_m r_m \|\mathbf{e}_f\|^2 - r_m\|\mathbf{e}_f\|^2 - \sigma\tilde{\varepsilon}(\varepsilon_o - \hat{\varepsilon}) \\ &\le M_m r_m \|\mathbf{e}_f\|^2 - r_m\|\mathbf{e}_f\|^2 - \frac{1}{2}\sigma\tilde{\varepsilon}^2 + \gamma \\ &\le -\frac{\lambda L}{P_m} + \gamma \end{aligned} \tag{27}$$

where

$\lambda = \min\{2M_m r_m, 2r_m, \sigma\}$

$\gamma = \frac{1}{2}\sigma(\varepsilon_o - \hat{\varepsilon})^2$

And this is conclude the proof.

▪

To solve the well known problem, that the persistence of the excitation condition cannot be guaranteed in the presence of external disturbance and noisy output measurements [9], which can majorly affect the neural network NN learning laws. To solve this problem many methods were suggested such as the projection and dead-zone modification presented in [11], and σ-modification [12], in this paper, we will use the e-modification method presented in [13] which will introduce an additional term related to the filtered error and parameter vector in the adaptive learning law. This method has the advantage over the other methods where it assures error boundedness and improves the overall system robustness in all respects even in case the persistence excitation condition un satisfied [13].

Then (21) can be modified as follow:

$$\dot{\widehat{\Theta}} = \eta\, Y\big(\boldsymbol{q}_d, \dot{\boldsymbol{q}}_d, \ddot{\boldsymbol{q}}_d, sgn(\dot{\boldsymbol{q}})\big)\, \boldsymbol{e}_f^T - \delta\left\|\mathbf{e}_f\right\|\eta\,\widehat{\Theta} \tag{28}$$

where δ is a small positive design parameter.

IV. Simulation Results

In this section the simulation results are presented to illustrate the effectiveness of proposed algorithm, where a three links cylindrical robot is considered, with the following dynamic model:

$$m_R\ddot{\gamma} + (0.5\, m_R L - m_R\gamma)\dot{\psi}^2 = u_\gamma \tag{29}$$

$$\left(\frac{m_B\gamma_B^2}{2} + \frac{m_R L^2}{3} - m_R L\gamma + m_R\gamma^2\right)\ddot{\psi} + m_R(2\gamma - L)\,\dot{\gamma}\,\dot{\psi} = u_\psi \tag{30}$$

$$m_R\ddot{z} + g\, m_R = u_z \tag{31}$$

where m_R, m_B represent the mass of the upper and lower arms respectively, L is the length of the upper arm, γ_B is the radius of the base, g is the gravity constant. The generalized coordinates variables $q = [\gamma\ \psi\ z]$ are specified in Fig. (2), and defined as follow:

γ is the reach motion, ψ is the swing and z is the stroke, and the kinematic equations of the cylindrical robot are given by:

$$x = \gamma\cos\psi$$

$$y = \gamma\sin\psi$$

$$z = z$$

In this simulation, the actual parameters of the cylindrical robot are given in Table. 1

Parameter	Value	Unit
m_R	1	Kg
m_B	2	Kg
L	0.5	m
γ_B	0.1	m
g	9.8	m/s^2

Table. 1 The Cylindrical Robot Parameters

The desired circular path is given by the following trajectories in the Cartesian coordinates:

$$x_d = 0.2 + 0.1\sin\frac{2\pi t}{10} \tag{32}$$

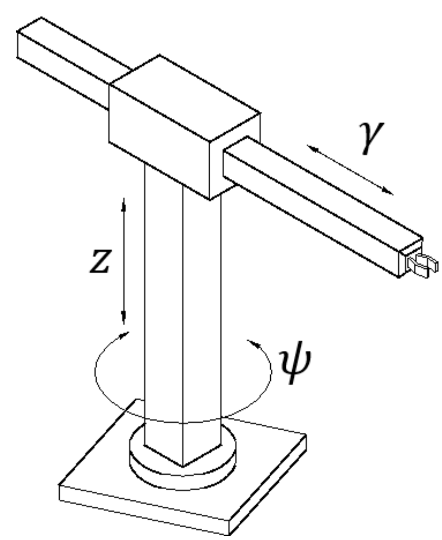

Fig. (2) Cylindrical Robot Geometry.

$$y_d = 0.1 + 0.1\cos\frac{2\pi t}{10} \tag{33}$$

$$z_d = 1.3\sin\left(\frac{2\pi t}{241}\right) \tag{34}$$

It can be noted, from the dynamic model of the cylindrical robot that the stroke dynamics is decoupled from the other reach and swing dynamics, for that reason and to reduce the number of inputs to the NN controller, we suggest building two NN controllers the first for controlling the reach and swing motion and the second for controlling the stroke motion. The decentralize methodology considered here will greatly reduce the complexity of the controller. The design parameters for the neural network controllers are chosen as follows: $\Lambda = 10\,I$, $K = 100\,I$ and $\eta = 0.109\,I$ where I is an appropriate identity diagonal matrix for each of the two controllers. $\eta = 0.109\,I$, $\eta_\varepsilon = 0.5$, $\sigma = 0.01$, $\varepsilon_o = 0.05$, $\delta = 0.01$. The first neural controller has eight inputs and the second neural controller has four inputs for both controllers each input has eleven radial basis activation functions with the following means:

[-1.35 -1.08 -0.81 -0.54 -0.27 0 0.27 0.54 0.81 1.08 1.35]

and the standard deviation have been set for all inputs equal to 3. Moreover, all the initial weights of the output layer for both networks are chosen equal to zero. The simulation results are presented in Fig. (3) where the cylindrical robot links desired and actual trajectories are shown. Then the tracking error corresponding to the links trajectories are given in Fig. (4), it can be noted that the trajectory followed within finite time and small bounded error. In Fig. (5) the tracked path is presented from top view in the xy-plane moreover a three dimensions plot of the desired and actual paths are shown. The generalized coordinate γ and ψ of the reach and swing motion are given in Fig. (6). And finally the three links torque generated by the controller are given in Fig. (7) where it is clear that the controllers output torques within the acceptable range and have a very little chattering profile.

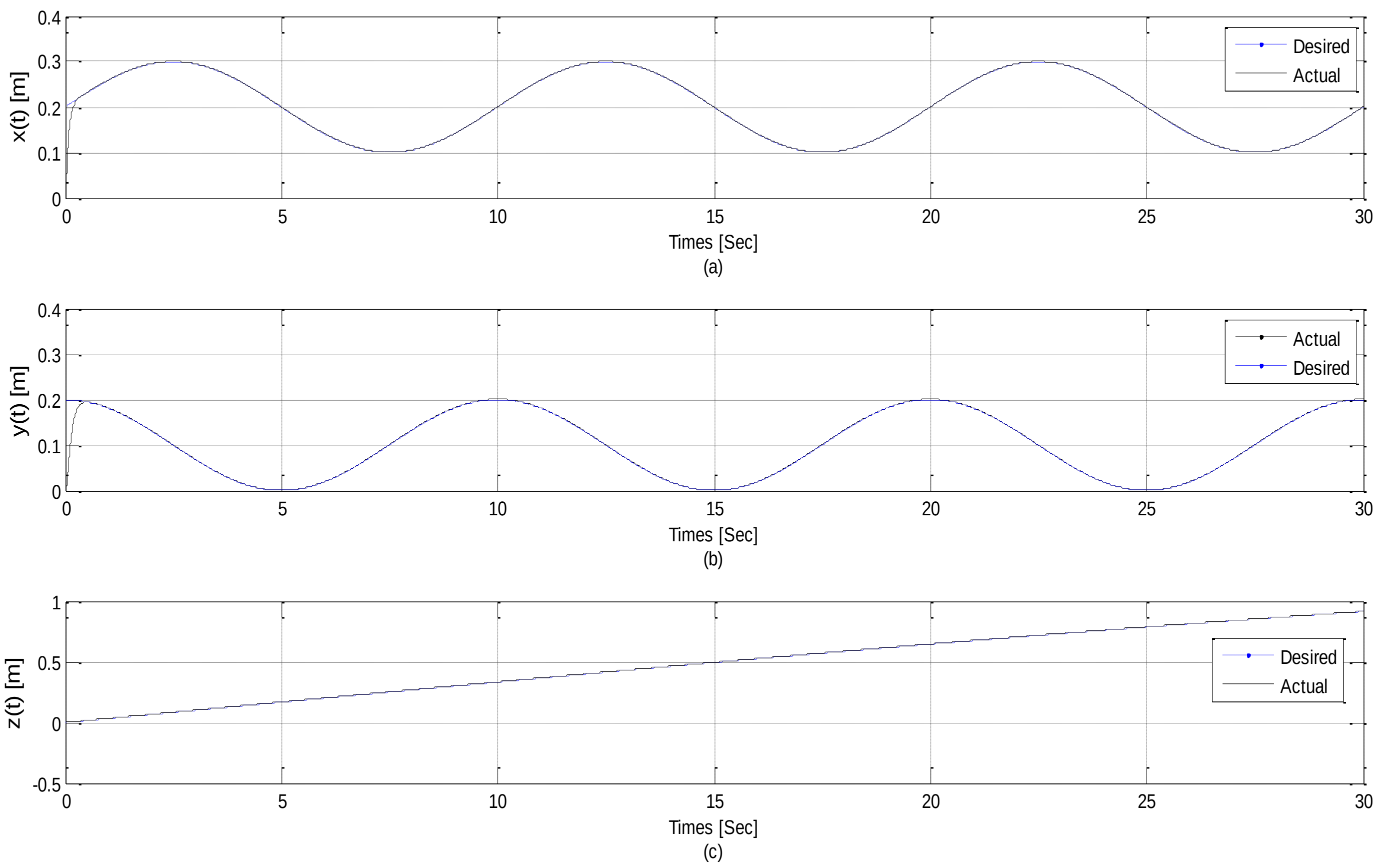

Fig. (3) The cylindrical Robot links trajectories: (a) the x(t), (b) the y(t) and (c) the z(t) trajectories.

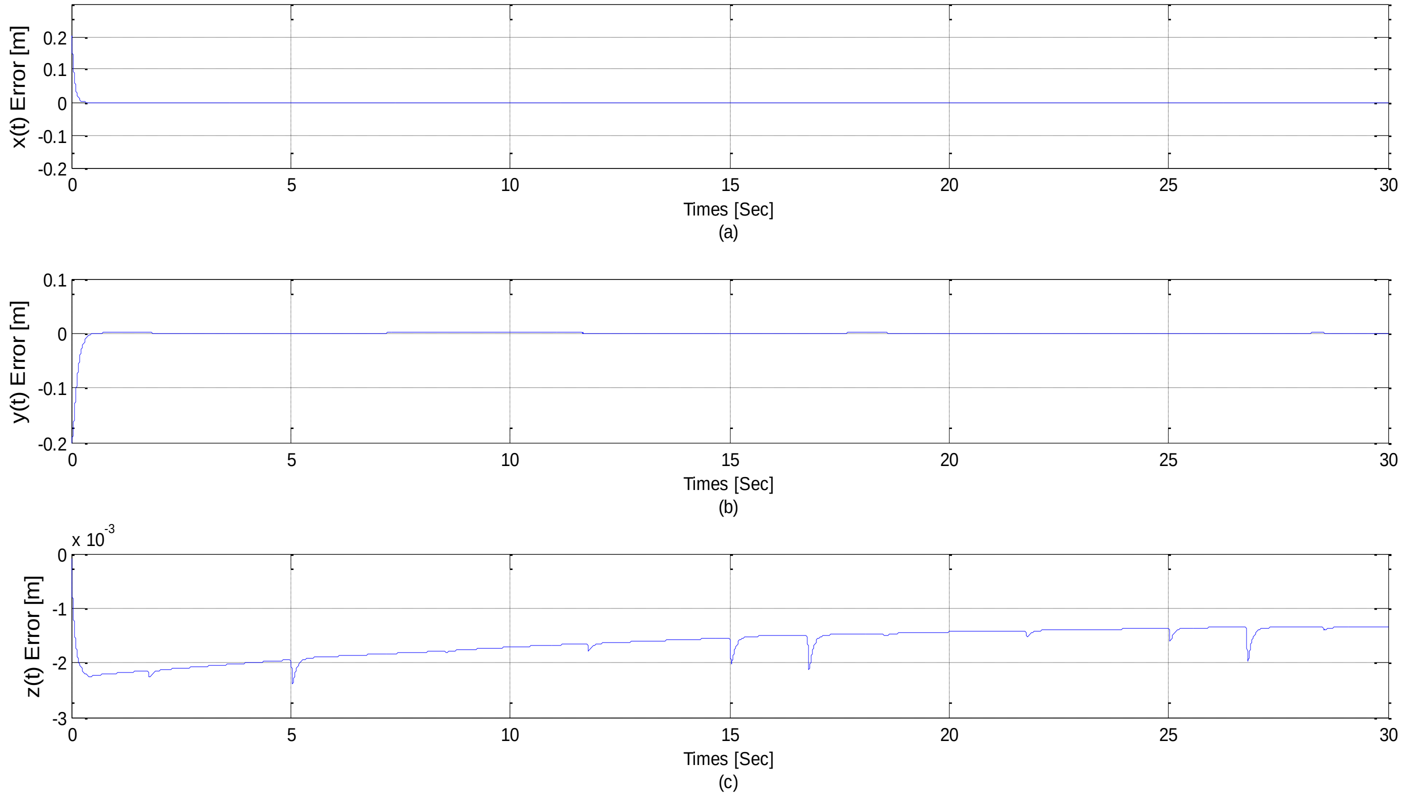

Fig. (4) Links trajectory error: (a) x(t) trajectory error, (b) y(t) trajectory error and (c) z(t) trajectory error.

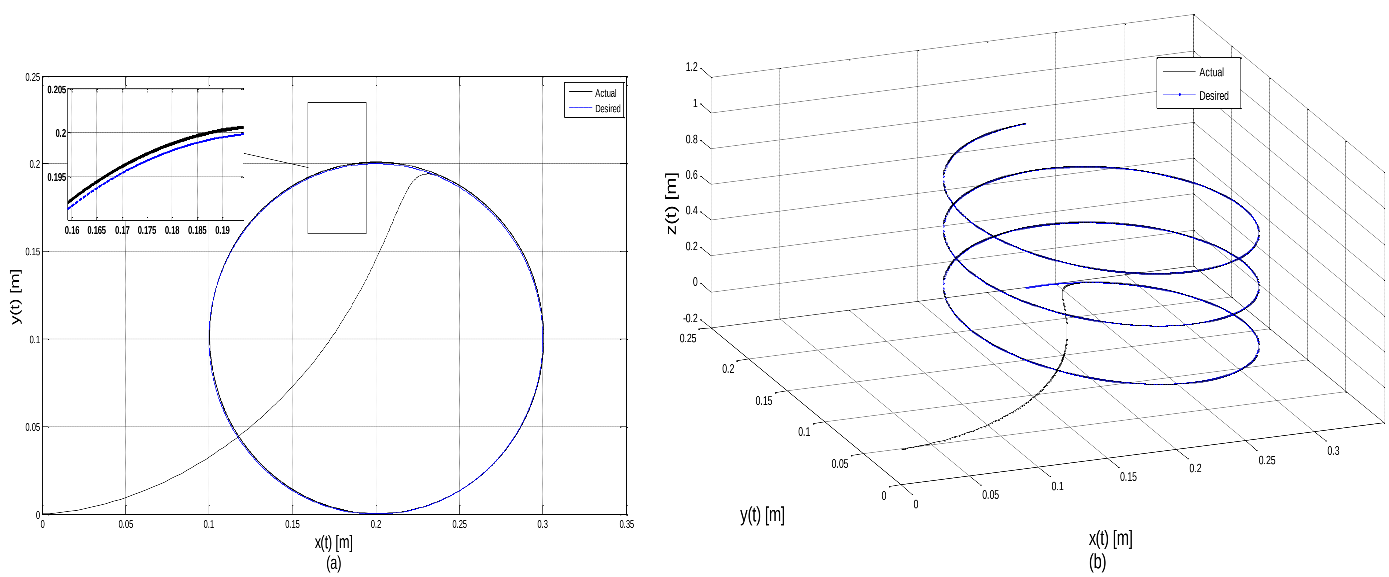

Fig. (5) The path followed by the cylindrical robot: (a) in the xy-plane (top view), (b) three dimensions view.

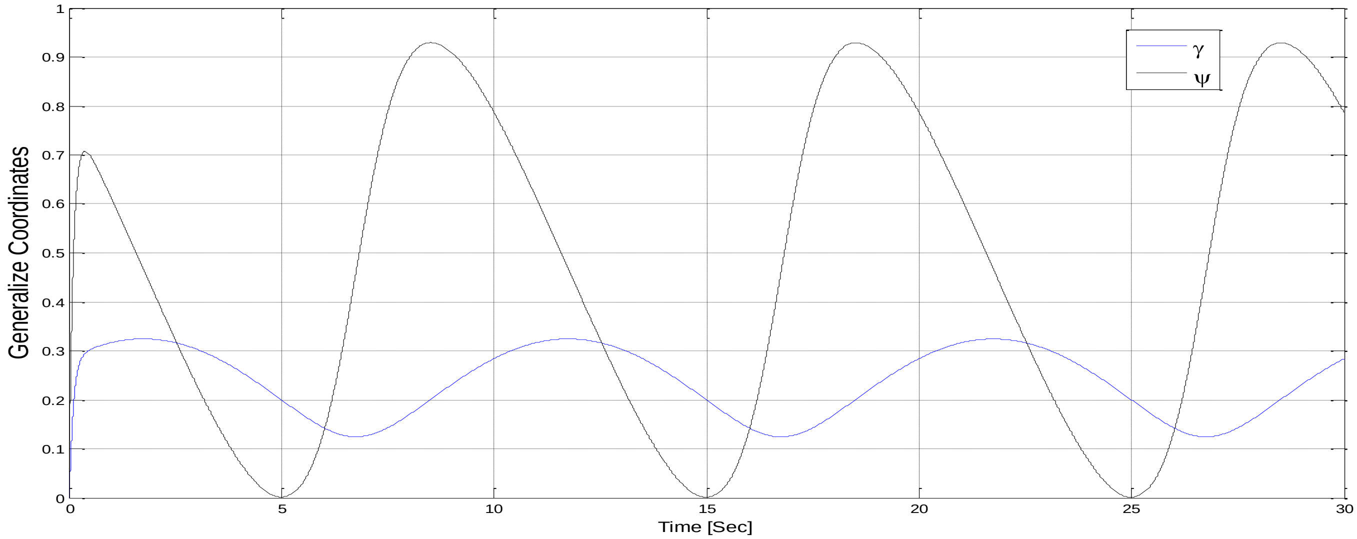

Fig. (6) The cylindrical robot generalized coordinate: reach γ and swing ψ motions

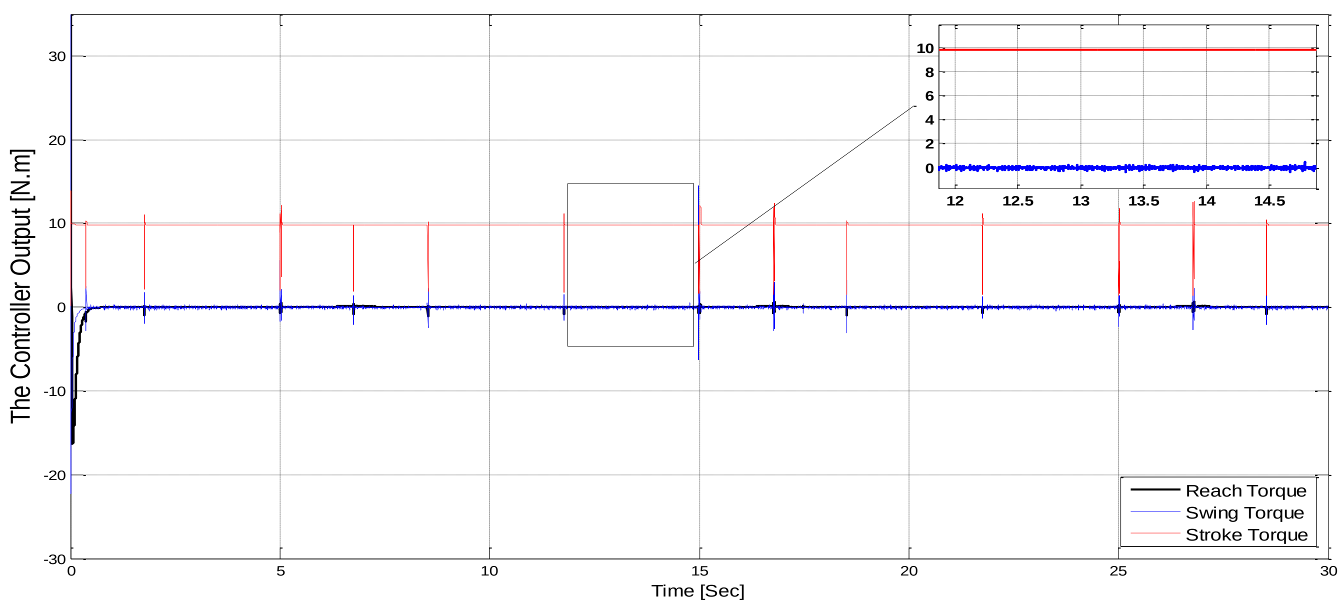

Fig. (7) The controllers output torques.

V. Conclusion

In this paper, a composite PD and sliding mode neural network (NN)-based adaptive control algorithm has been designed for the control of cylindrical robot manipulator to guarantee goals of the trajectory tacking in the presence of unknown dynamics and nonlinearities. To achieve a robust NN learning law a Lyapunov sense has been employed to derive these law and then the e-modification algorithm used to augment the leaning law to overcome the problems of breaching the ideal excitation conditions. The NN learned and compensate the nonlinearity in the system, and the PD term based on the filtered error with sliding mode controller are successfully reduced the effect of inherent approximation error in the NN. The simulation results show that the overall combined controller has been guaranteed tracking error exponentially stable in the closed loop. Future work can be oriented toward the real time implementation of the proposed algorithm on the robotic manipulators.

Reference:

[1] T. Ozaki, T. Suzuki, T. Furuhashi and S. Okuma, "Trajectory Control of Robotic Manipulators Using Neural Networks", IEEE Trans. on Industrial Electronics, Vol. 38, No. 3, June 1991.

[2] J-J. E. Slotine, "Robustness Issues in The Control of High Performance Robots", Proce. Of 23rd Conf. on Decision and Control Las Vegas, NV, December 1984.

[3] C. J. Harris, "Advances in Intelligent Control", Taylor & Francis, 1994.

[4] F. C. Sun and Z. Q. Sun, "Stable sampled-data adaptive control of robot arms using neural networks", Journal of Intelligent and Robotic Systems, vol. 20, no. 2, pp. 131-155, 1997.

[5] Y. Li, G. Ding, B. Zhao, B. Dong and G. Liu, "Decentralized Adaptive Neural Network Sliding Mode Control for Reconfigurable Manipulators with Data-based Modeling", IEEE International Joint Conf. on Neural Networks (IJCNN), 2016.

[6] A. Rezoug and M. Hamerlain, "Adaptive Neural Higher Order Terminal Sliding Mode Controller", 8th International Conf. on Modelling, Identification and Control (ICMIC) , Algiers, November 15-17, 2016.

[7] J. Ma, P. Li, L. Geng and Z. Zheng, "Adaptive Finite-time Tracking Control for a Robotic Manipulator with Unknown Deadzone", 2015 IEEE 54th Annual Conference on Decision and Control (CDC) Osaka, Japan, December 15-18, 2015.

[8] H. Xin and C. Qiang, "Full-Order Neural Sliding Model Control of Robotic Manipulator with Unknown Dead-Zone" , IEEE, 2015.

[9] B. Rahmani and M. Belkheiri, "Robust Adaptive Control of Robotic Manipulators Using Neural Networks: Application to a Two Link Planar Robot", 8th International Conf. on Modeling, Identification and Control (ICMIC), Algeria, November 15-17, 2016.

[10] F. Suny, Z. Sunz, N. Liy and L. Zhangy," Stable Adaptive Control for Robot Trajectory Tracking Using Dynamic Neural Networks", Machine Intelligence & Robotic Control, Vol. 1, No. 2, 71–78, 1999.

[11] J. Farrell and M. Polycarpou, "Adaptive Approximation Based Control: Unifing Neural, Fuzzy and Traditional Adaptive Approximation Approaches", Hoboken, NJ: Wiley-Inter Science, 2006.

[12] K. Volyanskyy, W. Haddad and A. Calise, " A New Neuroadaptive Control Architecture for Nonlinear Uncertain Dynamical Systems: Beyond σ- and e- Modifications", IEEE Tans. Neural Network, Vol. 20, no. 11, pp. 1707-1723, 2009.

[13] K. S. Narendra and A. M. Annaswamy, "A New Adaptive Law for Robust Adaptation without Persistent Excitation", IEEE Trans. Automat. Contr., Vol. 32, no. 6, pp. 134-145, 1987.