INVESTIGATION OF TRAJECTORY TRACKING RECURRENT NEURAL CONTROLLER FOR ROBOT MANIPULATOR EMPLOYING HIL SIMULATION TECHNIQUE

J. SENTHIL KUMAR, N. SENTHIL KUMAR

Mepco Schlenk Engineering College, Sivakasi, 626005, Tamil Nadu, India.

Email: senthilkumarj@mepcoeng.ac.in, nsk_vnr@mepcoeng.ac.in

Abstract: *In this work, recurrent neural network based controller is implemented for controlling Robot Manipulator using Hardware in the Loop (HIL) simulation technique. The gains of the PID based recurrent neural network controller scheme are initialized with the cuckoo search algorithm (CSA) optimization method rather than assuming randomly. The Least Mean Square (LMS) adaptive algorithm is then investigated for the online adaptation of the gains of the controller. The performance of the designed controller is tested against the plant parameters uncertainties and external disturbances for all the links of a three link rigid robotic manipulator with variable payload. The stability analysis of the presented control system is investigated out using Lyapunov's approach. The Simulink model of the robotic manipulator has been developed using Matlab-Simulink software and the performance of the controller implemented using the HIL technique in C2000 real time controller was analyzed. From the HIL simulation for trajectory tracking results it is evident that the dominance and effectiveness of the CSA optimized recurrent neural network PID controller (RNPID) over the optimized neural network PID (ONPID), optimized PID (OPID), and PID controllers for variable payload and disturbance rejection.*

Key words: *Robot manipulator, PID controller, Neural Network, Cuckoo Search Algorithm, Hardware-in-the-loop simulation.*

1. Introduction

In industrial automation, nowadays robotic manipulators are playing a leading role. Various domains of research were in progress to make them applicable for very high precision control of end effector and improving their performance with better stability [1]. In such crucial applications, that demands real time tuning of controller gains, which has been experimented to be a challenging task using PID controllers [14]. Since, robotic manipulators are highly nonlinear, they may not respond reasonably for high precision control applications with conventional tuning methods. Typically, the joint drive torques for each link to track the desired trajectory is adjusted iteratively based on observed trajectory errors [3]. Due to the capability of non linear mapping in robotic manipulators, implementation of fuzzy controllers it can be a better choice for the control when coupled with the PID logic [15]. Even in spite of providing a better suited option for the control of robotic manipulators, a 3 Degree of Freedom (DOF) robotic manipulator which is a coupled system that requires tuning of 3 PID controllers simultaneously. With the aid of advanced computing facilities using processors and nature inspired optimization techniques several authors have used various techniques to tune the PID controllers [7-10]. They are all confined to fixed gains which are difficult to provide optimum performance for certain nonlinear and uncertain control of robotic manipulators. While considering the requirement of high precision tracking, better performance and convergence rate for robotic manipulators, the meta-heuristic cuckoo search optimization can perform better as this its performance is independent of chosen parameters [11].

The application of artificial neural networks for dynamic control systems has recently been used by many researchers because of its excellent generalization, adaptation and learning capabilities. Due to lack of faster training algorithms, the artificial neural network for online tuning is not much explored. In the presented work all the parameters of RNPID controller are optimized using CSA. After obtaining the optimized parameters of the RNPID controller, the weights of the hidden layer are re-adjusted online using Moore Penrose generalized inverse [6]. It ensures minimizing the error and to meet the desired torque required for tracking the trajectory followed by the arm of the robotic manipulator. The online sequential training algorithm is very fast and it will not converge to local minima. So, it will be a better choice than the conventional gradient based tuning methods. A Lyapunov stability criterion is used to analyze the stability of the proposed controller. To test the robustness of the proposed controller the HIL simulation approach is employed. It comprises the actuators and dynamics of the robotic manipulator as a portion of the simulator system and the control algorithm implemented in the real hardware. This kind of HIL based simulation modeling supports fast prototyping of control algorithms, rather than investing on the actual robotic manipulator [13].

This proposed work presents the combination of PID and recurrent neural network control algorithms. The performance of algorithms is also optimized by using cuckoo search algorithm. Both the combined algorithms and its optimized versions are implemented in the hardware from Texas Instruments named as C2000 real time microcontroller. The control algorithms in the hardware act as HIL to drive the actuators in the model of robotic manipulator implemented in the Matlab Simulink. Robustness testing is performed under external disturbances and varying masses of the links and finally the performance comparison is carried out among RNPID controller, CSA tuned Neural Network PID (ONPID) controller, CSA tuned PID controller (OPID) and conventional PID controller

2. Dynamic modeling of robotic manipulator

The mathematical model of a three link planar serial robot manipulator is used to express the behavior of robot manipulators. The design of the model based controller has been derived from dynamic equations of robotic manipulator for carrying out the HIL simulation to estimate the tracking joint angle errors and position of end effector is shown in Fig.1.

The dynamic modeling of the robotic manipulator describes the relationship between joint motion, accelerations and torque [2]. Moreover, it is used to describe the particular dynamic effects such as inertia, Coriolis, centrifugal, and the other parameters of the manipulator. The dynamic mathematical model for a ‘n’ link, serially connected, direct driven robotic manipulator is given in joint space as follows:

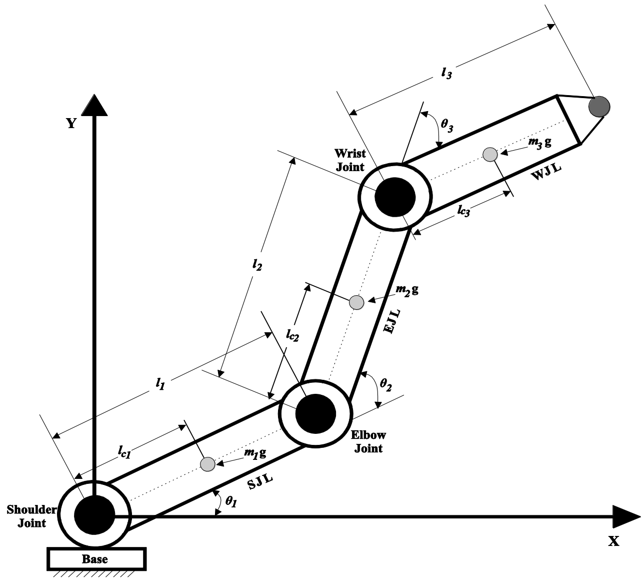


Fig.1. Model of the Robotic Manipulator

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + G(\theta) = \tau \qquad (1)$$

where, $M(\theta)$ is the $n{\times}n$ inertia matrix of the manipulator, θ is the $n{\times}1$ joint position vector , $C(\theta,\dot{\theta})$ is the $n{\times}1$ vector of Centrifugal and Coriolis terms, $G(\theta)$ is the $n{\times}1$ vector of gravity terms, τ is the $n{\times}1$ vector of joint actuator torques and forces. The manipulator chosen is modelled with three rigid links, the first one connecting the shoulder and elbow joint represented as Shoulder Joint Link (SJL), second link connecting elbow and wrist joint represented as Elbow Joint Link (EJL), and the last link connecting wrist joint and the gripper which is represented as Wrist Joint Link (WJL). The torque vector for those links can be expressed as:

$$\tau = [\tau_1\ \tau_2\ \tau_3]^T \qquad (2)$$

where τ_1 , τ_2 and τ_3 represents the torques applied to the actuators of shoulder joint, elbow joint and wrist joints respectively.

The equation (1) and equation (2) show the controller output torque and the link positions θ_1, θ_2 and θ_3 of three links. The Table 1 lists the relevant parameters used in the model of robotic manipulator, where l_1, l_2 and l_3 is the length of SJL, EJL and WJL and m_1, m_2 and m_3 are the masses of each of those links respectively.

Table 1: Parameters of the three link planar rigid robotic manipulator

Parameters	Shoulder Joint Link(SJL)	Elbow Joint Link(EJL)	Wrist Joint Link(WJL)
Mass	$m1$=0.15kg	$m2$=0.1kg	$m3$ =0.28kg
Length	l_1=0.09m	l_2=0.063m	l_3=0.115m
Distance from the joint to its center of gravity	l_{c1}=0.0492m	l_{c2}=0.0356m	l_{c3}=0.612m
Acceleration due to gravity (g)	9.81 m/s^2	9.81 m/s^2	9.81 m/s^2

3. Design of HIL based Recurrent PID controller

Designs of adaptive controllers for robotic manipulators, using the dynamic neural networks have been used because of their improved adaptation capabilities and prediction. The recurrent neural network can be easily trained and quickly because of the lesser number of parameters [16]. A small memory due to the feedback in the recurrent neural network increases the learning capabilities and approximation of the network.

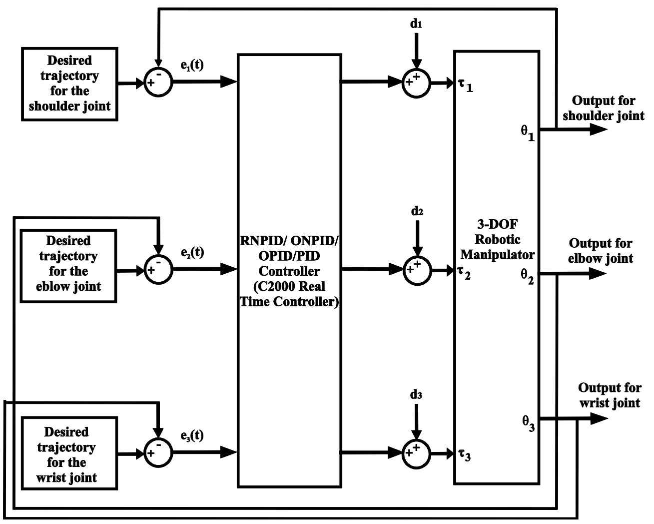


Fig.2. Block diagram of the HIL simulation strategy implementation of RNPID, ONPID,OPID and PID controllers for robotic manipulator

Modelling of PID based recurrent neural network controller using HIL simulation technique is presented in the following section. Fig.2 shows the implementation of the RNPID controller employing HIL simulation strategy using C2000 real time controller, Matlab and Code Composer Studio (CCS) environment [5].

3.1 Recurrent Neural Network PID controller

The structure of PID based recurrent controller is shown in Fig. 3. The three nodes of the structures act like proportional, integral and derivative nodes for the recurrent controller. It is implemented using HIL simulation technique is C2000 real time controller.

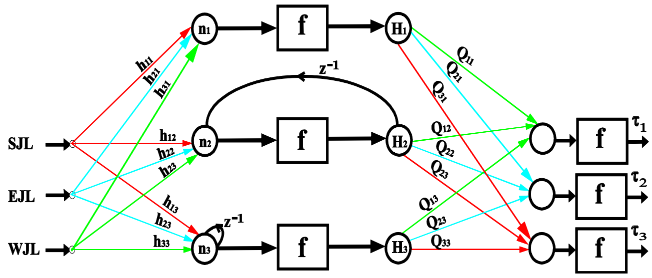


Fig. 3: Recurrent Neural Network PID controller

The three hidden layer neurons H_1, H_2 and H_3 outputs can be expressed in terms of SJL, EJL and WJL as

$$H_1(k) = \phi(h_{11}\, SJL(k) + h_{21}\, EJL(k) + h_{31}\, WJL(k)) \quad (3)$$

$$H_2(k) = \phi\big(h_{12}\, SJL(k) + h_{22}\, EJL(k) + h_{32}\, WJL(k)\big) + H_2(k-1) \quad (4)$$

$$H_3(k) = \phi\big(h_{13}\, SJL(k) + h_{23}\, EJL(k) + h_{33}\, WJL(k) + h_{13}\, SJL(k-1) + h_{23}\, EJL(k-1) + h_{33}\, WJL(k-1)\big) \quad (5)$$

The hidden layer weight (h) and output layer weight(Q) matrices can be expressed as

$$h = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}; Q = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ Q_{31} & Q_{32} & Q_{33} \end{bmatrix}$$

The output of second and third nodes with the along with unit delay z^{-1} can be expressed as

$$H_2(k) = \frac{\phi(h_{12}\, SJL(k) + h_{22}\, EJL(k) + h_{32}\, WJL(k))}{1 - z^{-1}} \quad (6)$$

$$H_3(k) = \phi(\,(1 - z^{-1})\, h_{13}\, SJL(k) + h_{23}\, EJL(k) + h_{33}\, WJL(k)\,) \quad (7)$$

The RNPID controller output torque at its final stage is

$$\tau = Q_1 H_1(k) + Q_2 H_2(k) + Q_3 H_3(k) \quad (8)$$

The equation (8) can also be represented in matrix form as in equation (9)

$$\tau = QH \quad (9)$$

3.2 Online Training Strategy

The training of the recurrent controller is done to minimize the error which is accomplished by adjusting the Q and H parameters. The recurrent controller output parameter R uses the CSA optimized output weights and the LMS error norm can be minimized by readjusting the weights in the hidden layer. The hidden layer matrix is reinitialized using equation (10) and the corresponding error is shown in equation (11)

$$E = R - HQ, \|E\|^2 = (R - HQ)(R - HQ)^T \quad (10)$$

$$\|E\|^2 = RR^T - 2HQR^T + HH^T Q^2 \quad (11)$$

The training of the RNPID controller is required to adjust Q and H so that the error $\|E\|^2$ is minimized with respect to the Q and H parameters as shown in equations (12) and (13) respectively.

$$\frac{\partial \|E\|^2}{\partial Q} = -2HR^T + 2HH^T Q \quad (12)$$

$$\frac{\partial \|E\|^2}{\partial H} = -2Q \frac{\partial H^T}{\partial w} R^T + 2Q^2 \frac{\partial Q^T}{\partial x} Q \quad (13)$$

The output weights can be calculated in a single step analytically using equation (14) if the hidden layer weights are known.

$$Q = \frac{HR^T}{HH^T} \quad (14)$$

This feature enables high speed training comparaed to the other gradient methods. The initial weights and optimized output weights were calculated using CSA is used in the RNPID controller as the weights are readjusted to obtain the minimum norm least square solution. By using $H_1(k), H_2(k)$ and $H_3(k)$, the hidden layer weights h_{ij} are readjusted. If the error $\|E\|^2$ exceeds the fixed threshold, then the sequential learning is employed, followed by applying the control torque to the joint actuators [4].

4. Optimization of RNPID controller using Cuckoo search algorithm

Cuckoo search enables faster convergence performance and it is used to tune the parameters of the RNPID controller [12]. The integral of absolute error (IAE) is chosen as objective function for all the three links of the robotic manipulator. It is used to minimize the errors $e_1(t), e_2(t)$ and $e_3(t)$ between desired and actual trajectories for SJL, EJL and WJL respectively as represented in equations (15), (16) and (17). The weighted sum of IAE of the three links represents an aggregate fitness function (AFF) as represented in equation (18).

$$Of_1 = \int |e_1(t)|\, dt \quad (15)$$

$$Of_2 = \int |e_2(t)|\, dt \quad (16)$$

$$Of_3 = \int |e_3(t)|\, dt \quad (17)$$

$$AFF = w_1 Of_1 + w_2 Of_2 + w_3 Of_3 \quad (18)$$

Where, Of_1, Of_2 and Of_3 are the objective functions of the links SJL,EJL and WJL having the weights represented by w_1, w_2 and w_3 respectively.

5. Experimental Results and Discussion

The control scheme for RNPID, ONPID, OPID, PID controllers for a three link rigid robotic manipulator trajectory tracking is shown in Fig. 2. The performance of the recurrent neural network controller is compared with CSA optimized neural network PID controller, CSA optimized conventional PID controller and PID controller. All the control algorithms were implemented in C2000 real time controller that houses TMS320F2807x digital signal processor. The fluxgate sensors integrated with the board is used to measure the current and voltage of the modeled actuators in the robot joints. Based on the control algorithm the current flow to the modeled actuators are controlled with signal conditioning using the differential 12 bit analog to digital converters. By using the on chip enhanced pulse width modulation modules the current flow to the actuators are varied based on the conditioned signal, which in turn is used to control the movements of actuators. From the HIL simulations were carried out using the Matlab Simulink and Code Composer Studio utilizing fixed step ODE 4 solver with a fixed time of 0.01s. The optimized parameters for the controller obtained with the aid of cuckoo search approach are summarized in Table2.

IAE for the proposed RNPID for the three links are tabulated in Table 5. Clearly IAE values are less for the proposed RNPID controller and high for the PID controller, when compared among the four implemented control schemes. The proposed RNPID control scheme has outperformed the ONPID, OPID and PID controllers in terms of tracking error and IAE variations. The observed IAE values for a 5% increase and 5% decrease in the mass of all the three links simultaneously are summarized in Table 3 and Table 4 respectively. Because of the adaptive nature of the RNPID controller, it is able to reject effectively the parameter variations and leads to less change in IAE values compared to ONPID, OPID and PID controllers. The test trajectory chosen is of cubic polynomial type for the three joints. The comparison of the HIL simulation results for reference trajectory tracking of SJL, EJL, WJL and the end effector implemented for RNPID, ONPID, OPID and PID controllers are shown in Fig. 4.

Table 2: Optimized Choice of tuning parameters for RNPID, ONPID, OPID and PID Controllers

Parameters	**RNPID Controller** (K_p, K_i, K_d)		**ONPID Controller** (K_p, K_i, K_d)		**OPID Controller** (K_p, K_i, K_d)	**PID Controller** (K_p, K_i, K_d)
	Hidden layer	**Output layer**	**Hidden layer**	**Output layer**		
SJL	1.337, 0.175, 2.2115	394.57, 133.9473, 39.0267	0.8192, 0.8782, 0.89822	398.9858, 135.2397, 38.5438	398.2528, 65.7475, 13.4757	399.1264, 66.694, 14.3531
EJL	0.9657, 0.0963, 4.1315	59.1318, 11.5101, 4.5675	0.9459, 0.7798, 0.042	59.9386, 11.3886, 3.2724	44.048, 18.607, 2.657	44.9195, 19.4435, 3.4585
WJL	0.95505, 0.0926, 4.09225	58.4468, 10.9501, 4.55325	0.91885, 0.7597, 0.0365	59.4336, 10.9186, 2.8374	43.1765, 17.7705, 1.8555	44.48375, 19.02525, 3.05775

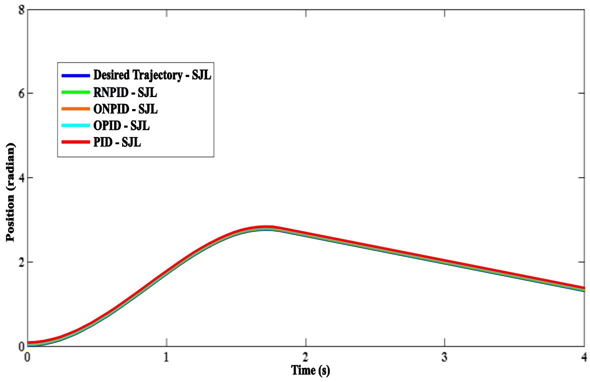
Table 3: IAE values for RNPID, ONPID, OPID and PID Controllers for 5% decrease in masses of the links

Parameter Variation (5%Decrease)	**IAE of RNPID controller**			**IAE of ONPID controller**			**IAE of OPID controller**			**IAE of PID controller**		
Mass in %	**SJL ($x10^{-4}$)**	**EJL ($x10^{-4}$)**	**WJL ($x10^{-5}$)**	**SJL ($x10^{-3}$)**	**EJL ($x10^{-3}$)**	**WJL ($x10^{-4}$)**	**SJL ($x10^{-3}$)**	**EJL ($x10^{-3}$)**	**WJL ($x10^{-4}$)**	**SJL ($x10^{-2}$)**	**EJL ($x10^{-2}$)**	**WJL ($x10^{-3}$)**
m1	0.323	0.827	1.330	0.575	1.940	3.306	0.902	2.559	4.217	1.502	4.335	7.168
m2	0.327	0.827	1.327	0.581	1.940	3.300	0.902	2.552	4.203	1.505	4.326	7.146
m3	0.337	0.839	1.341	0.590	1.953	3.315	0.914	2.568	4.222	1.528	4.354	7.181
m1,m2	0.322	0.827	1.332	0.571	1.940	3.310	0.897	2.552	4.208	1.493	4.326	7.158
m1,m2, m3	0.314	0.825	1.337	0.564	1.940	3.315	0.871	2.550	4.229	1.454	4.322	7.189

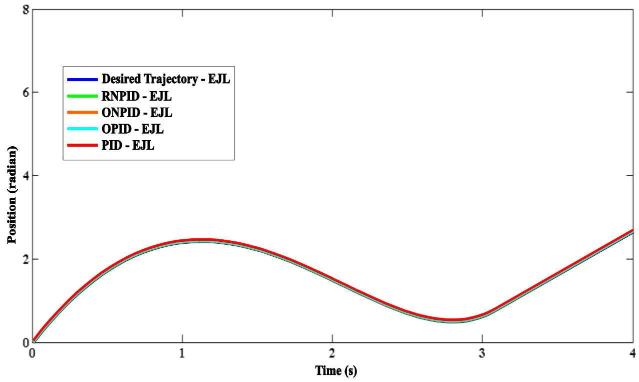
By adding a sinusoidal signal to the controllers, the external disturbance during the trajectory performance has been observed. The IAE values with the effect of disturbance on the controllers have been tabulated in Table 5. As the IAE of RNPID controller is smallest among all, it outperforms other controllers even in spite of disturbances applied to the links. With the proposed tuning procedure, the RNPID controller is easy to implement and tune which ensures minimum effects of external disturbances and parametric uncertainties. Fig.5 summarizes the joint disturbances effect of all three links on individual links. It is also evident that the base link SJL is less prone to disturbances due to the dynamics of the robotic manipulator, when compared to the upper links EJL and WJL.

Table 4: IAE values for RNPID, ONPID, OPID and PID Controllers for 5% increase in masses of the links

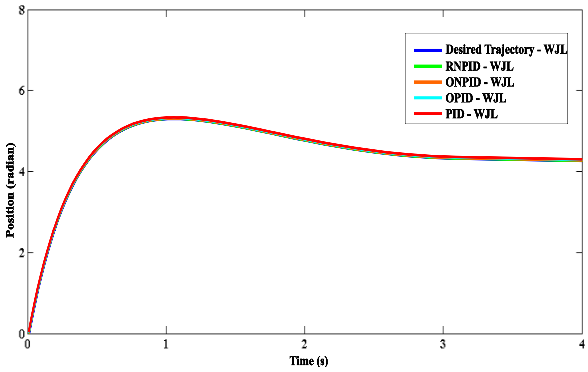
Parameter Variation (5%Increase)	IAE of RNPID controller			IAE of ONPID controller			IAE of OPID controller			IAE of PID controller		
Mass in %	**SJL ($x10^{-4}$)**	**EJL ($x10^{-4}$)**	**WJL ($x10^{-5}$)**	**SJL ($x10^{-3}$)**	**EJL ($x10^{-3}$)**	**WJL ($x10^{-4}$)**	**SJL ($x10^{-3}$)**	**EJL ($x10^{-3}$)**	**WJL ($x10^{-4}$)**	**SJL ($x10^{-2}$)**	**EJL ($x10^{-2}$)**	**WJL ($x10^{-3}$)**
m1	0.310	0.762	1.214	0.583	1.853	3.124	0.864	2.472	4.080	1.450	4.168	6.886
m2	0.307	0.762	1.217	0.577	1.853	3.129	0.864	2.472	4.080	1.447	4.168	6.888
m3	0.320	0.774	1.228	0.592	1.865	3.139	0.876	2.484	4.092	1.473	4.192	6.911
m1,m2	0.312	0.762	1.212	0.586	1.853	3.120	0.869	2.479	4.089	1.457	4.177	6.897
m1,m2, m3	0.301	0.761	1.221	0.573	1.853	3.133	0.839	2.472	4.104	1.410	4.167	6.923



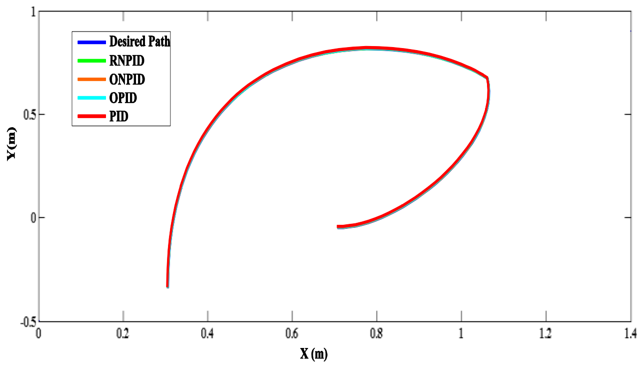
(a)



(b)



(c)



(d)

Fig. 4. Trajectory tracking of (a) SJL (b) EJL (c) WJL (d) End effector of robotic manipulator

Table 5: IAE values for RNPID, ONPID, OPID and PID Controllers for disturbances in the links

Disturbances (N-m)	IAE of RNPID controller			IAE of ONPID controller			IAE of OPID controller			IAE of PID controller		
	SJL ($x10^{-4}$)	EJL ($x10^{-4}$)	WJL ($x10^{-5}$)	SJL ($x10^{-3}$)	EJL ($x10^{-3}$)	WJL ($x10^{-4}$)	SJL ($x10^{-3}$)	EJL ($x10^{-3}$)	WJL ($x10^{-4}$)	SJL ($x10^{-2}$)	EJL ($x10^{-2}$)	WJL ($x10^{-3}$)
SJL	0.413	0.854	1.295	0.740	2.006	3.273	0.970	2.626	4.283	1.677	4.455	7.234
EJL	0.404	0.827	1.249	0.724	1.939	3.155	0.948	2.559	4.171	1.640	4.334	7.029
WJL	0.329	1.472	2.615	0.585	3.797	7.010	0.905	3.192	5.479	1.512	6.013	10.514
SJL & EJL	0.405	1.477	2.548	0.732	3.835	6.939	0.948	3.193	5.439	1.643	6.028	10.414
SJL, EJL & WJL	0.373	1.306	2.240	0.680	3.346	6.012	0.907	3.029	5.150	1.561	5.589	9.617

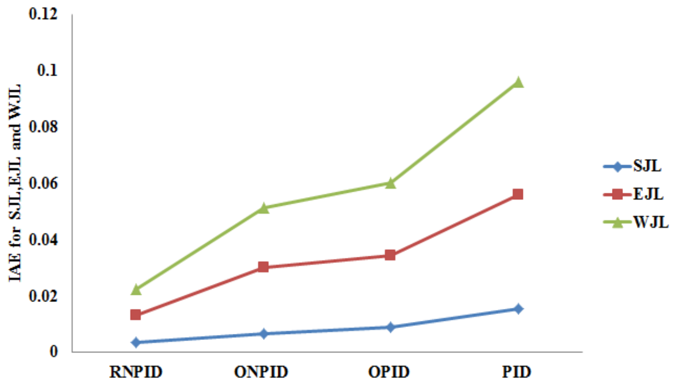


Fig. 5. IAE for RNPID, ONPID, OPID and PID Controllers for joint disturbances

6. Conclusion

A PID based recurrent neural network controller is implemented in the C2000 launch pad and the HIL simulation for the trajectory tracking control of a three link rigid robotic manipulator is inspected in this work. While the conventional PID controller uses nine parameters for tuning, the recurrent neural network controller uses eighteen, which adds more flexibility and has a simplified structure with compatible learning capabilities. To deal with nonlinear, time-varying and dynamic payload handing capabilities of robotic manipulators, updating of the parameters of the recurrent neural network controller is done online. Using a sequential cuckoo search learning algorithm, the initial gains of neural network controller are calculated. The learning algorithm deals with the robotic manipulator that possesses nonlinear dynamics and tunes the weights in a nonlinear manner in the output layer. From the analyzed results it is observed that the performance of the proposed recurrent neural network controller RNPID is less affected due to external disturbances and parametric uncertainties when compared to ONPID, OPID and PID controllers.

References

1. Iqbal, J., R. U. Islam, M. F., Khan, *Control Strategies for Robotic Manipulators*, IEEE International Conference on Robotics and Artificial Intelligence (ICRAI), Rawalpindi, October 2012, p. 26-33.
2. Ullah, M.I., Ajwad, S.A., Islam, R.U., Iqbal, U., Iqbal, J., *Modelling and computed torque control of a 6 degree of freedom robotic arm*, IEEE International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE), Islamabad, April 2014, p. 133-138.
3. Tsai, Ching-Chih, Hung, Chi-Chih, Chang, Chih-Fu, Trajectory planning and control of a 7-DOF robotic manipulator, IEEE International Conference on Advanced Robotics and Intelligent Systems (ARIS), Taipei June 2014, p. 78 - 84.
4. Y. SalimiKhaligh, M. Namvar, *Adaptive control of robot manipulators including actuator dynamics and without joint torque measurement*, IEEE International Conference on Robotics and Automation (ICRA), May 2010, p. 4639-4644.
5. Farzin Piltan, Sara Emamzadeh, Zahra Hivand, Forouzan Shahriyari& Mina Mirzaei, *PUMA-560 Robot Manipulator Position Sliding Mode Control*

Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses, International Journal of Robotic and Automation, (IJRA) June 2012, vol. 3, no. 3, p. 106-150.

6. Zhang Yan , Wang Fanzhen , Song Ying , Chen Zengqiang , Yuan Zhuzhi, *Recurrent neural networks-based multivariable system PID predictive control*, Frontiers of Electrical and Electronic Engineering in China, April 2007, vol. 2, no. 2, p. 197-201.
7. Liu Luoren, Luo Jinling, *Research of PID Control Algorithm Based on Neural Network*, International *Conference* on Energy Systems and Electric Power (ESEP), Singapore, December 2011, p. 6988-6993.
8. B. Guo, H. Liu, Z. Luo, F. Wang, *Adaptive PID Controller Based on BP Neural Network*, International Joint Conference on Artificial Intelligence (JCAI), April 2009, p. 148-150.
9. Wei Lu, Jianhua Yang, Xiao Dong Liu, *The PID Controller Based on the Artificial Neural Network and the Differential Evolution Algorithm*, Journal of Computers, Oct 2012, vol. 7, no. 10, p. 2368-2375.
10. JyotiOhri,Dhaval R. Vyas, Pretty NeelamTopno, *Comparison of Robustness of PID Control and Sliding Mode Control of Robotic Manipulator*, International Symposium on Devices MEMS, Intelligent Systems & Communication (ISDMISC) 2011, Issue:7, p. 5-10.
11. Ehsan Valian, Shahram Mohanna and Saeed Tavakoli, Improved Cuckoo Search Algorithm for Feedforward Neural Network Training, International Journal of Artificial Intelligence & Applications, July 2011, vol.2, no.3, p. 36 - 43.
12. Milan Tuba, Milos Subotic, Nadezda Stanarevic, *Performance of a Modified Cuckoo Search Algorithm for Unconstrained Optimization Problems,* World Scientific and Engineering Academy & Society Transactions on Systems, February 2012, vol. 11, no.2, p. 62-74.
13. Guo, Yu; Zhou, Feng; Marian, Nicolae; Angelov, Christo K. *Hardware-in-the-Loop Simulation of Component-Based Embedded Systems*, Proceedings of the 8th International Workshop on Research and Education in Mechatronics REM 2007, p. 268-273.
14. G. M. Khoury, M. Saad, H. Y. Kanaan, C. Asmar, *Fuzzy PID Control of a Five DOF Robot Arm*, Journal of Intelligent and Robotic Systems, July 2004, vol. 40, no. 3, p. 299-320.
15. Minh-Duc Tran, Hee-Jun Kang, *Adaptive Fuzzy PID Sliding Mode Controller of Uncertain Robotic Manipulator*, International Conference on Intelligent Computing Theories and Methodologies (ICIC), China, August 2015, p. 92-103.
16. M. U. Jamil, M. N. Noor, M. Q. Raza, S. Rizvi, *Backstepping control using adaptive neural network for industrial two link robot manipulator*, IEEE International Multi-Topic Conference (INMIC), Karachi, December 2014, p. 389-394.