Real Time Implementation of NARMA-L2 Control of a Single Link Manipulator

[1]S.S. Mokri, [1]H. Husain, [2]W. Martono and [2]A. Shafie
[1]Faculty of Engineering,
National University of Malaysia, 43600 UKM Bangi, Selangor DE, Malaysia
[2]Kulliyah of Engineering,
International Islamic University of Malaysia, 50728, Kuala Lumpur, Malaysia

Abstract: Nonlinearities and parametric uncertainties are unavoidable problems faced in controlling robot manipulator. A single link manipulator driven by a permanent magnet brushed dc motor is a nonlinear dynamics due to effects of gravitational force, mass of the payload, posture of the manipulator and viscous friction coefficient. Furthermore, uncertainties arise because of changes of the rotor resistance with temperature and random variation of friction while operating. Due to this fact, classical PID controller can not be used effectively since it is developed based on linear system theory. In order to overcome this problem, in this research, a neural network control scheme, NARMA-L2 Control is adopted and implemented in real time for controlling a DC motor driven single link manipulator with unknown dynamics. However, the real time experimentation showed that the proposed system results in chattering of the control signal. Hence, the system also chatters within the desired trajectory. As a solution, real time Smoothed NARMA-L2 Control scheme is implemented. Physical results showed that the improved control scheme has not only reduced the chattering but has successfully controlled the single link manipulator for both point-to-point and continuous path motion control.

Key words: NARMA-L2 control, real time implementation, neural network, single link manipulator

INTRODUCTION

Industrial manipulator robots play an important role in the field of flexible automation. A single link manipulator is the most basic manipulator, which is operated to perform tasks such as moving payloads or painting objects. To obtain a high performance single link manipulator, position controllers are necessary in order that the manipulator follows a preselected positional trajectory specified either as point-to-point or continuous path tracking motion with minimal deviation.

Control of DC driven single link manipulator is a nonlinear control problem due to gravitational force, mass of the payload, posture of the manipulator and viscous friction coefficient. Besides, uncertainties arise due to changes of the DC motor (actuator) parameters such as change of rotor resistance caused by temperature change as well as random variation of friction while operating. As a consequence, classical control algorithm, which is developed based on linear system assumption such as PID controller[13] is inadequate to deal with this problem.

In addition, an overwhelming majority of the available controllers are designed based on the assumption that the actuator dynamics are negligible[12,2,10,25]. This assumption reduces the dynamic model of the robot and facilitates the design of controllers. As a result of this simplification, unmodeled disturbances exist in the robot control systems, which affect the tracking and positioning of the robot. Although there are several methods to make a controller robust, with respect to the unmodeled dynamics, the performance of the controller is not as expected, meaning that the tracking errors are bounded but do not converge to zero[23].

To deal with unknown nonlinearities, various control strategies have been proposed in the forms of variable structure controller[25], robust control[6] and adaptive controller[20]. However, the essential characteristic of these controllers is the model dependence, i.e., the requirement for explicit a priori specified model structure is still a necessity. In case of the manipulator robot, it is difficult to obtain some parameters such as the inertia matrix and mass centers at any joint with sufficient accuracy. It is then

Corresponding Author: S.S. Mokri, Faculty of Engineering, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor DE, Malaysia

considered that these controllers are pertinent in the sense that an accurate model of the manipulator is demanded prior to controller design stage.

Therefore, a viable alternative to achieve an efficient control scheme is through the appliance of intelligent control. Intelligent control approaches such as neural network and fuzzy inference system do not require mathematical model of the system under controlled and have the ability to approximate nonlinear system. The real time applications of fuzzy logic control specifically for single link manipulator were reported in[8,11].

Many researches have been attempting to use neural network intelligent controls for the trajectory control of robot manipulators. Among recent works carried out in the field of control of robot manipulators using neural network based controllers is[9]. They proposed a controller for robust backstepping control of a general nonlinear system using neural networks. One of the nonlinear systems considered is robot manipulator. Here, the inertia matrix is considered to be known. Ahmad *et al.*[1] proposed a neural network controller based on modified Kohonen's Self Organizing Map (SOM) which controls the joint of the manipulator. Basically, the proposed controller consists of two neural network schemes; the neural network controller and the robotic emulator. The neural network controller will determine the joint torque after introducing the desired end effector's coordinate. The torque value will then become the input to the robotic emulator; which output is feedback to the controller to form a closed loop control.

In addition[19], also reported a direct inverse neural network based control scheme to solve the tracking control problem for the robot arm. Attempts are made in the literature to combine sliding mode and neural network to achieve the desired position control[17]. They presented a sliding model neural network control scheme. Here, a neural network controller is developed to estimate the equivalent control in the sliding mode control.

As a matter of fact, these newly devised neural networks based controllers effectiveness is verified through simulation studies. On the other hand, actual real time applications as well as comparative experimental results study are rarely established. Illustrating one example is the work of[5], in which the author reported an indirect neural network real time control for an industrial robot arm. Also[21], described experimental results for position control of real manipulator using a kind of neural controller that operates in parallel with a conventional controller based on the feedback error learning architecture. They stated

that the advantage of this controller is that it does not require any modification of the previous conventional controller.

Therefore, in this research, from an experimental point of view, the use of different neural network based control strategy, NARMA-L2 Control for trajectory control of a DC driven single link manipulator prototype is presented. The DC driven single link manipulator system is selected since it represents the basis of robot manipulator systems and customarily used by researches as a benchmark for simulation as well as the implemented hardware substituting real industrial robot manipulator in order to show the effectiveness especially for newly formulated control methodologies in manipulator control problem. The application of the designed control strategy on a single link manipulator system could then be extended to more than one link manipulator system with less effort. Equally important, the real time control implementations are emphasized as we strongly believed that the reliability of experimental results of any devised control strategy is much more important than their abstract qualities.

As proposed by K. S Narendra and Mukhopadhyay[18], NARMA-L2 control is one of the popular neural network architectures for prediction and control. This approach has been adopted through simulations for process control such as drug dosage regimens in cancer chemotherapy[3], Scheibel contactors control[16] and magnetic levitation process[24] and no real time control application through literature studies. Theoretically, this control structure is simple and its implementation is not particularly demanding. Thus, favorably, NARMA-L2 control has the advantage to reduce the amount of memory and computation time. Faster and accurate output regulation is expected due to its mapping capability.

However, the shortcoming of this controller is that it results in chattering of the control signal as shown in this research. Thus, an improved version of NARMA-L2 Control has been designed based on[22] technique. They suggested an adjunction of a linear feedback to the NARMA-L2 controller structure in order to smooth the control action. The addition of a linear feedback to the NARMA-L2 controller is equivalent to the feedback linearization control methodology. As validated in this research, this design strategy; named as Smoothed NARMA-L2 Control has successfully alleviate the chattering in the response signals and improve the performance of NARMA-L2 controller.

Description of DC motor driven single link manipulator: The real time experiments have been

carried out on a DC driven single link manipulator prototype. The experiment is performed on a test bench consisting of 12 Volt DC motor, a single link manipulator with payload attached at the end and a rotary encoder to measure the angular position of the manipulator. The control algorithm is implemented in Matlab/Simulink software package with Real Time Workshop and xPC Target Toolbox. Using xPC Target, the host and target computers are connected directly via a serial cable using RS232 ports. The test rig is connected to the target PC using the PCI6024E DAQ card interface while the encoder connected to the same target PC using PCI QUAD04 Measurement Computing encoder card. A servoamplifier Logosol LS-DY is used to drive the DC motor. For the reference signal, voltage signal is sent to the PCI6024E DAQ card and later to the servoamplifier. The output signal is measured from an encoder connected directly to the end of the manipulator's shaft. The encoder output which represents the angular position is sent to the PCI QUAD04 which reads the feedback signal. All the components were calibrated and tested before implementation. Schematically, the whole experimental setup is concluded in Fig. 1.

NARMA-L2 Control: NARMA-L2 is one of the popular neural network architectures for prediction and control. The principle idea of this control scheme is to apply the input output linearization method[4] where the output becomes a linear function of a new control input.

Basically, there are two steps involved when using NARMA L2 control: system identification and control design. In the system identification stage design, a neural network of the plant that needs to be controlled is developed using two subnetworks for the model approximation. The network is then trained offline in batch form using data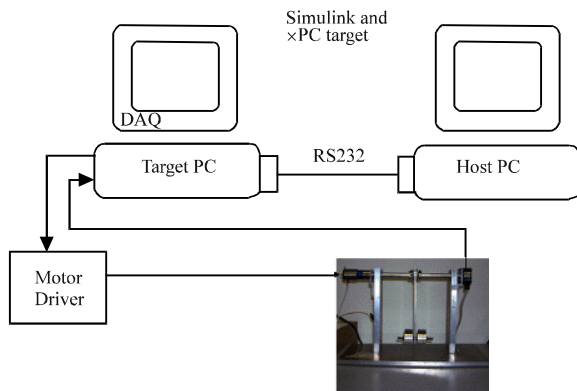 collected from the operation of the plant. Next, the controller is simply the rearrangement of two subnetworks of the plant model.Computation of the next control input to force the plant output follows a reference signal is materialized through simple mathematical equation

NARMA-L2 plant model identification: In NARMA-L2 plant identification, the model structure used is the standard NARMA[17] model adapted to the feedback linearization of affine system. A companion form system (control affine) is used as the identification model, i.e.:

$$
\begin{aligned}
y(k+1) = & \\
& f[y(k),...,y(k-n+1), \\
& u(k-1),...,u(k-m+1)] + \\
& g[y(k),...,y(k-n+1), \\
& u(k-1),...,u(k-m+1)].u(k)
\end{aligned} \tag{1}
$$

In essence, the NARMA-L2 approximate model will be parameterized by two neural networks $\hat{f}$ and $\hat{g}$ that will be used to identify the system of Eq. (1), i.e.:

$$
\begin{aligned}
\hat{y}(k+1|\theta) = & \\
& \hat{f}[y(k),...,y(k-n+1), \\
& u(k-1),...,u(k-m+1),w] + \\
& \hat{g}[y(k),...,y(k-n+1), \\
& u(k-1),...,u(k-m+1),v].u(k)
\end{aligned} \tag{2}
$$

The two subnetworks are used for the model approximation; NN1 and NN2 which are used to approximate nonlinear functions f and g respectively. The NARMA-L2 system identification structure of the single link manipulator is shown in Fig. 2.
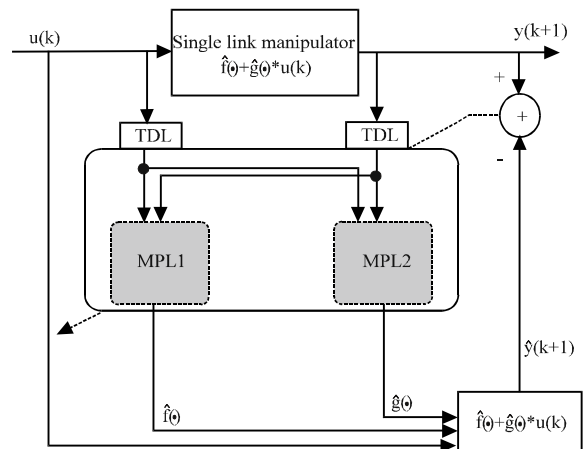

Fig. 1: Experimental Setup

Fig. 2: NARMA-L2 plant model identification

The plant model identification in NARMA-L2 Control starts off with a dataset of input output data pairs collected experimentally. If necessary, the data pairs are preprocessed. Again, the collected dataset is divided into two parts; one for the training of the neural nets and the other for cross validating the resulting neural model. Here, the NN1 subnetwork is a feedforward neural network with one hidden layer with p neurons of hyperbolic tangent (tanh) activation function and an output layer of one neuron with linear activation function. Also, the NN2 subnetwork is a feedforward neural network with q tanh hidden layer neurons and one output neuron.

For each subnetwork, the number of past output n and the past input m; which compose the input vector and the number of neurons (p and q) of the hidden layer are determined. Subsequently, the selected neural network structure is trained using the input pattern and the desired output from the dataset. Here, the parameters (weights and biases) of the two MLP subnetworks that properly approximate the nonlinear modeling representing the DC driven single link manipulator are estimated. The optimization technique that will be used to update the parameters is also importantly determined.

Finally, to measure the success at approximating the dynamical system plant model using the neural network model, the prediction error ε_k should be uncorrelated with all linear and nonlinear combination of past inputs and outputs. Thus, the validation and cross validation tests are carried out to ascertain the validity of the obtained neural network model.

NARMA-L2 controller design: The NARMA-L2 controller design is uncomplicated. The control action can be simply implemented using the obtained NARMA-L2 model based on Eq. (2) in which the functions $\hat{f}$ and $\hat{g}$ are defined. In order for a system output, y(k+1), to follow a reference trajectory y_r(k+1), we set: y(k+1) = y_r(k+1). The NARMA-L2 controller is designed through substituting y(k+1) with y_r(k+1) in Eq. 2. Then the resolving controller output would have the form of:

$$u(k) = \frac{y_r(k+1) - \hat{f}[y(k),..,y(k-n+1),u(k-1),..,u(k-m+1)]}{\hat{g}[y(k),...,y(k-n+1),u(k-1),..,u(k-m+1)]} \quad (3)$$

Figure 3 shows the block diagram of NARMA-L2 controller which clearly a rearrangement of the NARMA-L2 plant approximated model.
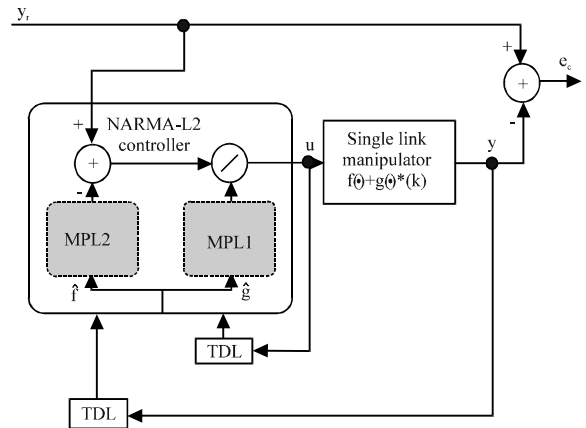

Fig. 3: NARMA-L2 controller

Smoothed narma-L2 control: The control performance of NARMA-L2 controller is improved using the technique proposed by[22] where they suggested an adjunction of a linear feedback to the NARMA-L2 controller structure in order to smooth the control action. The addition of a linear feedback to the NARMA-L2 controller is equivalent to the feedback linearization control methodology.

Pukrittayakame *et al*.[22] had proved that the effect of chattering in plant response using NARMA-L2 controller is in fact could be reduced with this technique through simulations. Accordingly, the same design strategy was carried out in this work to improve the performance of NARMA-L2 controller in real time on the DC driven single link manipulator test rig. The improved version of NARMA-L2 control is named as Smoothed NARMA-L2 control.

Pukrittayakame *et al*.[22] also states that although NARMA-L2 controller is considered analogous to the feedback linearization controller, there is still a major difference between these two. The feedback linearization controller adds a linear feedback term. Thus, if a linear feedback is included to the original NARMA-L2 control law, the control signal u(k) is in the form of:

$$u(k) = \frac{c_0 y_r(k+1) - f[y_n(k), u_m(k-1)] - d^T y_p(k)}{g[y_n(k), u_m(k-1)]} \quad (5)$$

where, $d = [d_1,...,d_p]^T$.

If f() and g() are accurately approximated, the system output will satisfy the following linear difference equation:

$$y(k+1) = c_o y_r(k+1) - d^T y_p(k)$$

or

$$y(k) + d_1 y(k-1) + ... + d_p y(k-p) = c_o y_r(k) \qquad (6)$$

Taking the z-transforms, we have:

$$Y(z) = \frac{c_0}{D(z)} Y_r(z) \qquad (7)$$

$$D(z) = 1 + d_1 z^{-1} + ... + d_p z^{-p} \qquad (8)$$

In order to obtain a response y(k) that is a smoothed version of $y_r(k)$, the roots of D(z) inside the unit circle is suitably put. Accordingly, u(k) is smoothed and the chattering found in NARMA L2 control is lessened.

RESULTS AND DISCUSSION

NARMA-L2 plant model identification results: The plant model identification is estimated based on 50000 input output data obtained from the experiment. The dataset consists of random input signals $u_k \in [-1,1]$ used to excite the plant and the angular position signal y_k is sampled at 100 Hz sampling frequency. The dataset is divided into two parts; training data intended for training the NARMA-L2 model and testing data for cross validating the resulting neural model.

In order to train the NN1 and NN2 networks, which compose the NARMA-L2 based model, the initial weights and biases are randomly selected. The error goal of mean squared error (MSE) is set to 10^{-6}. The number of past output n and past input m are set as 2 each; corresponding to the same values used in plant model identification for Neural Network Model Reference control. The best neural network structure is selected based on heuristics which gives the least mean squared error after trying different number of neurons in the hidden layer for NN1 and NN2 (p and q). The optimization routine applied Levenberg Marquardt[14] learning scheme. The number of training epochs is 1000 iterations. Ultimately, the NARMA-L2 model has the following structure, where the cost function is minimized to the order of 10^{-4}:

- The NN1 network used to model $\hat{f}(.)$ has 8 hidden hyperbolic tangent neurons with bias and 1 linear output neuron
- The NN2 network used to model $\hat{g}(.)$ has 2 hidden hyperbolic tangent neurons with bias and 1 linear output neuron
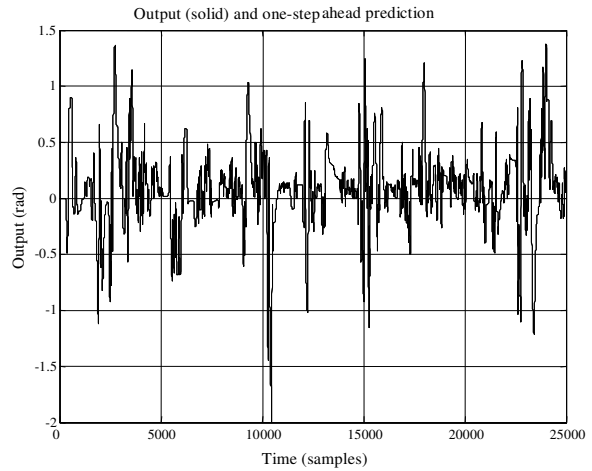

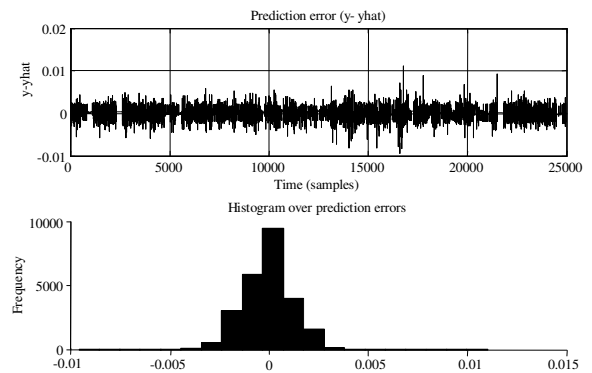Fig. 4: Validating the neural identification scheme on test data

Fig. 5: Prediction error on the test data

The validity of the obtained NARMA-L2 model was first validated on the dataset used for training and further cross validated on the testing dataset. The cross validation test is shown in Fig. 4 while Fig. 5 shows the prediction error. It is observed from the histogram that the error is close to zero for most of the samples. The correlation results are shown in Fig. 6 from which we can see that the auto correlation of the residuals lies within 95% confidence limits which gives us strong indication that the model is acceptable.

Furthermore, we can see that the cross correlation between the past inputs and the prediction error lies between the 95% confidence limits also. The cross validation results conclude that the obtained NARMA-L2 model to represent the affine form model of the DC driven single link manipulator is valid.

NARMA-L2 control results: In order to evaluate the effectiveness of the designed NARMA-L2 Control, the
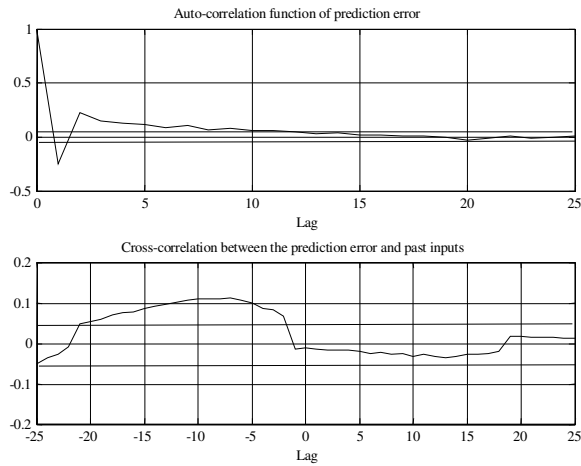
Fig. 6: Correlation analysis of the test data

controller is implemented in real time to control the DC motor driven single link manipulator. Two different control modes are tested namely point-to-point and continuous path controls.

For the PTP control, 1.5 rad and 3.0 rad step inputs are used as reference inputs. Figure 7 and 8 show the response to those reference inputs. For CP control, two different desired trajectories are used; LSPB (Linear Segment Parabolic Blend) and sinewave signals as shown in Fig. 9 and 10. Viewing the associating figures, it can be concluded that the NARMA L2 controller is theoretically convincing; yet, in our case, it performs poorly in position control of the DC driven single link manipulator system although satisfying system modeling through the defined NARMA-L2 neural network architecture is realized. This is conforming to the observed chattering of the output position. The mean squared error (MSE) for both inputs are considered large; 0.0182 and 0.0153 respectively.

Smoothed narma-L2 control results: In this research, to design the Smoothed NARMA-L2 control, the order of the linear feedback,p, is chosen to be 1. The closed loop pole (root of D(z)) is arbitrarily chosen to attain the best tracking control performance and is finally set to 0.75. Real time experimental results using the enhanced confirmed the diminution of chattering in the tracking responses.

The following Fig. 11-14 show the system responses under the Smoothed NARMA-L2 control. Those figures confirmed that the proposed Smoothed NARMA-L2 controller is considerably good in smoothing the oscillating effects (chattering) due to original NARMA-L2 control. The control performance of this controller is shown in Table 1. It indicates that

Table 1: Smoothed NARMA-L2 control performance

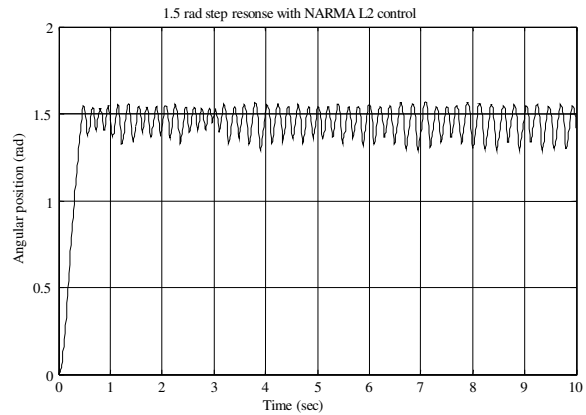| Reference	Overshoot (%)	Settling time (sec)	Rise time (sec)	Steady state error
1.5-rad	0	0.71	0.59	-0.0110
3.0-rad	0	0.80	0.67	0.0095

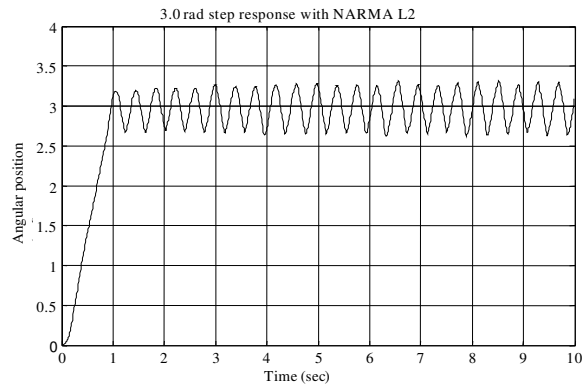Fig. 7: 1.5 rad step response under NARMA-L2 scheme

Fig. 8: 3.0 rad step response under NARMA-L2 scheme
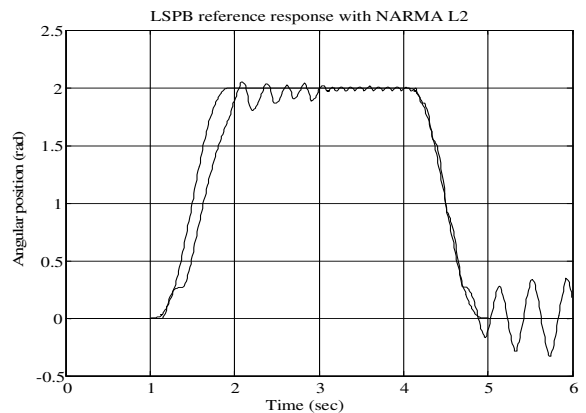

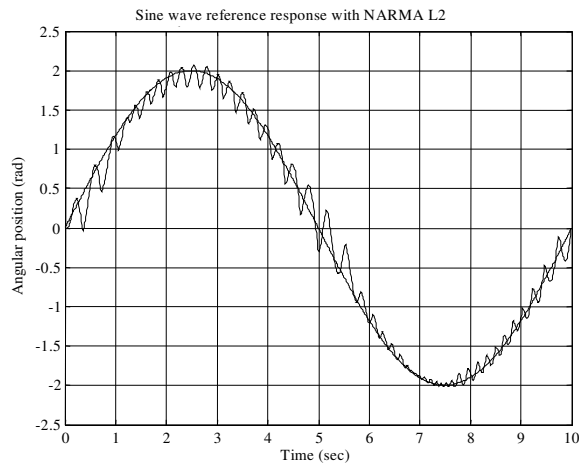Fig. 9: LSPB reference response under NARMA L2 scheme

Fig. 10: Sine wave reference response under NARMA L2 scheme

the control performance is enhanced based on elimination of oscillations (chattering), zero overshoot, faster settling time and small steady state error.

In terms of CP control, the MSE for LSPB input is 0.0027, a 50% improvement compared to NARMA-L2 control while the MSE for sinewave input is 0.0092, which is an 81% improvement.

CONCLUSION

Despite the fact that NARMA-L2 controller tenders a motivating solution to control nonlinear system, this research demonstrates that chattering as seen from the experimental single link manipulator responses to desired trajectory references is to a great extent, inevitable although we manage to control those responses to track within the vicinity of the desired trajectory. In some measure, NARMA-L2 controller for position control of lab scale single link manipulator is quite an achievement. Reason is, the neural network training in the system identification was carried out to arrive at the best solution.

As pointed earlier and repeatedly acknowledge in a number of NARMA-L2 research papers which enjoyed their control simulations success, this controller is attractive because it requires less computation either for training the neural network and the controller design. Strategically, the controller is also simply a recomposition of the neural network plant model and the neural network training is carried out offline.

It appears that the solution recommended by[22] has greatly improve the chattering observed in the signal responses as seen from the attached figures in relation with Smoothed NARMA L2 controller. Yet again,

using the linear feedback, to position the closed loop poles near the unit circle, the system response can be smoothed and control chattering is reduced.

REFERENCES

1. Ahmad, A.M., L.I. Chen and F. Mohamad, 2002. Simulation of stable adaptive control of robot arm using self organizing neural network. Student Conference on Research and Development Proceedings, ISBN 0-7803-7565-3. IEEE, Shah Alam, Selangor, Malaysia, pp: 162-164.
2. Dubowsky, S. and D.T. DesForges, 1979. The application of model reference adaptive control to robotic manipulators. ASME J. Dyn. Syst., Measurement and Control, 101 (3): 193-200.
3. Floares, A., C. Floares, M. Cucu and L. Lazar, 2003. Adaptive neural networks control of drug dosage regimens in cancer chemotheraphy. Proceedings of the International Joint Conference on Neural Networks, 2003, ISBN: 0-7803-7898-9, pp: 154-159, IEEE.
4. Garces, F., V. Becerra, C. Kambhampati and K. Warwick, 2003. Strategies for Feedback Linearisation: A dynamic neural network approach, Springer-Verlag, ISBN 1852335017, New York
5. Hitam, M.S. and K.F. Gill, 2000. Identification and control of a robot using a neural network. Proceedings of the IEEE Region 10 International Conference, (TENCON 2000). Kuala Lumpur, 2000, IEEE, pp: 308-314.
6. Jaritz, A. and M.W. Spong, 1996. An experimental comparisons of robust control algorithms in a direct drive manipulator. IEEE Trans. Control Syst. Technol., 4 (6): 627-640, ISSN: 1063-6536.
7. Jinzhu, P., W. Yaonan, S. Wei and L. Yan, 2006. Artificial neural network sliding mode controller with application to robotic manipulator. Proceedings of the 6th World Congress on Intelligent Control and Automation, ISBN: 1-4244-0332-4, pp: 2101-2105, IEEE.
8. Ketata, R. and J. Ibanez-Guzman, 1994. A robotic test rig for the experimentation with fuzzy logic. International Conference on Control, pp: 1134-1139, ISBN 0-85296-611-3, March 1994, IEEE, Coventy, UK.
9. Kwan, C. and F.L. Lewis, 2000. Robust backstepping control of nonlinear systems using neural network. IEEE Trans. Syst., Man and Cybernetics, Part A, 30 (6): 753-766, ISSN: 1083-4427.

10. Kye Lim and M. Eslami, 1987. Robust adaptive controller designs for manipulator systems. IEEE J. [legacy, pre-1988] Robotics and Automation. 3 (1): 54-66, ISSN: 0882-4967.

11. Le-Huy, H. and M. Hamdi, 1993. Control of a direct drive dc motor by fuzzy logic. Conference Record of the 1993 IEEE Industry Application Society Annual Meeting, Hong Le Huy and Maher Hamdi, pp: 732-738, Toronto, Ontario, Oct 1993, IEEE

12. Li, Y.F. and C.C. Lau, 1989. Development of fuzzy algorithms for servo system. IEEE Control Syst. Mag., 9 (3): 65-72, ISSN: 0272-1708.

13. Luh, J.Y.S., 1983. Conventional controller design for industrial robots: A tutorial. IEEE Trans. Syst., Man and Cybernetics, SMC-13 (3): 298-316, ISSN: 0018-9472.

14. Marquardt, D., 1963. An algorithm for least square estimation of nonlinear parameters. SIAM J. Applied Math., 11 (2): 164-168.

15. Miller, W.T., R.S. Suttton and P.J. Werbos, 1990. Neural networks for control. The MIT Press, ISBN: 02623161X, Cambridge

16. Mjalli Farouq, S., M. Abdel Jabbar and Nabil, 2005. Control of scheibel extraction contactors using neural network based control algorithms. Ind. Eng. Chem. Res., 44 (7): 2125-2133, ISSN: 0888-5885

17. Narendra, K.S. and Parthasaraty, K. (1990). Identification and Control of Dynamical Systems Using neural Networks. IEEE Trans. Neural Network, 18 (1): 4-27, ISSN: 1045-9227.

18. Narendra, K.S. and S. Mukhopadhyay, 1997. Adaptive Control Using Neural Networks and Approximate Models. IEEE Trans. Neural Network, 8 (3): 475-485, ISSN: 1045-9227.

19. Onishi, H. and Y. Maeda, 2002. Tracking control of robot arm using neural network with simultaneous perturbation learning rule. Proccedings of the 41st SICE Annual Conference, pp: 3188-3191, ISBN: 0-7803-7613-5, Aug 2002, IEEE.

20. Pagilla, P.R. and M. Tomizuka, 2001. An adaptive output feedback controller for robot arms: Stability and experiments, Automatica, 37 (7): 983-985, Elsevier.

21. Passolf, F. and M.R. Stemmer, 2004. Feedback error learning neural network applied to SCARA robot. Robot Motion and Control, 2004. Proceedings of the 4th International Workshop on Robot Motion and Control 2004, pp: 197-202, ISBN 83-7143-272-0, June 2004, IEEE.

22. Pukrittayakame, A., O. De Jesus and M.T. Hagan, 2002. Smoothing the control action for NARMA-L2 controllers. MWSCAS-2002. The 2002 45th Midwest Symposiom on Circuits and System, ISBN 0-7803-7523-8, pp: 37-40, IEEE.

23. Reed, J.S. and P.A. Ioannu, 1989. Instability analysis and robust adaptive control of robotic manipulators. IEEE Trans. Robotics and Automation, 5 (3): 381-386, ISSN: 1042-296X.

24. Sheirah, M.A., A. Hassan and S.A. Hammad, 2004. Identification and control of nonlinear processes using neural network. Proceedings of 2004 2nd International IEEE Conference, pp: 547-551, ISBN: 0-7803-8278-1.

25. Slotine, J.J.E. and W. Li, 1991. Applied Nonlinear Control, Prentice Hall, ISBN 0130408905, United States.

26. Slotine, J.J.E. and W. Li, 1987. On the adaptive control of robot manipulators. Int. J. Robotics Res., 6 (3): 45-59, ISSN: 0278-3649.