

A Control Scheme for Industrial Robots Using Artificial Neural Networks

M. Dinary, Abou-Hashema M. El-Sayed, Abdel Badie Sharkawy, and G. Abouelmagd

Abstract—This paper develops a new model-free control scheme based on artificial neural networks (ANN) for trajectory tracking applied on industrial manipulators. This scheme is developed to control arm robot manipulator without calculate the model parameters or dynamics, and use the online identification instead. The scheme consists of three parts. These parts are inverse identification part, ANN controller and linear controller. Inverse dynamics of the manipulator is identified by recurrent ANN that gives the identified torque. The ANN controller works on controlling the arm robot depends on the identifying torque. The linear controller designed for trajectory tracking error regulation. The identification and control ANN work together to improve the response of the linear controller. A simulated two-link arm robot is used to apply the control scheme on it. The scheme verified by mass variation. A comparison between the response of the manipulator with linear controller only and with the fully scheme has been carried out. The results show that adding the identification and control ANN improve the results of the linear controller.

Index Terms—Industrial robots, ANN, online identification, neural control, parametric and payload uncertainty.

I. INTRODUCTION

Theoretically speaking, for joint trajectory tracking control of an industrial robot, dynamic model based control system methods can be used. System implementation, however, is difficult to perform because of the existence of the uncertainties in the parameters in the dynamics and in the formulation of the dynamics itself. On the other hand, PID controllers are usually built-in in almost all industrial robot manipulators. As the significant drawback, however, it is well known that PID control cannot guarantee precise tracking results for given dynamic trajectories since such the control system is essentially driven by trajectory errors themselves. Some approaches for approximating part of dynamics of a robot by using neural network technology have been proposed instead of the dynamic model based control [1].

The feasibility of using an ANN for controlling an

unknown dynamical plant is investigated. A layered neural network is employed to learn the inverse dynamics of the unknown dynamical plant and acts as a feed forward controller to control the plant. This inverse dynamics is represented by the connection weights between the layers; these weights are adjusted based on the difference between the actual control input to the plant and the estimated input for achieving an actual plant output according to the inverse-dynamics model [2], [3].

Using ANN direct to control arm robot need calculation of the robot matrices in the learning law of the ANN and this make the controller depend on the dynamic model of the Arm robot [1]. To solve this problem we use another ANN to make identification of the inverse dynamics of the robot and use its output to learn the ANN of control without using the arm robot model. The identification ANN in our model works online identification. Then it doesn't need any previous data about the system. So many schemes used to make a robot control using ANN with identification [4], [5].

In this study we use ANN to learn the inverse dynamics of the robot arm and then use the output to train the ANN to control it is a solution to make controller independent on the model of the arm robot with its uncertainty and noisy. This controller is simulated on two-link arm robot and the results shown in the paper.

The paper outlined as follows: in Section II, the robot model and nominal value of its parameter are introduced. Section III explains the idea of the control scheme. Section IV explains in details the using of ANN as identification and as a controller with their equations. Section V shows the simulated results with a comparison between using linear controller only and after adding the ANN. Finally some concluding remarks are given in Section VI.

II. ROBOT MODELING

Without the loss of generality, we take the two-link rigid robot shown in Fig. 1, as an example to demonstrate the proposed control scheme. The inverse dynamic model is expressed as [6]-[8]:

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

where $\theta \in R^n$ is the joint angular position vector of the robot; $u \in R^n$ is the vector of applied joint torques (or forces); $M(\theta) \in R^{n \times n}$ is the inertia matrix, positive definite; $C(\theta, \dot{\theta})\dot{\theta} \in R^n$ is the effect of Coriolis and centrifugal torques; and $G(\theta) \in R^n$ is the gravitational torques. The physical properties of the above model can be found in [9].

Manuscript received February 19, 2014; revised May 5, 2014. This work was supported in part by the Mechatronics and Industrial robotics program, Faculty of Engineering, Minia University.

M. Dinary is with the Mechatronics and Industrial Robotics Program, Faculty of Engineering, Minia University, El-Minia, Egypt (e-mail: MohamedDinary@mu.edu.eg).

Abou-Hashema M. El-Sayed is with the Electrical Engineering Department, Faculty of Engineering, Minia University, El-Minia, Egypt (e-mail: abouhashema@mu.edu.eg).

Abdel Badie Sharkawy is with the Mechanical Engineering Department, Faculty of Engineering, Assiut University, Assiut, Egypt (e-mail: Ab.shark@aum.edu.eg).

G. Abouelmagd is with the Production Engineering and Design Department, Faculty of Engineering, Minia University, El-Minia, Egypt (e-mail: G_magd@yahoo.com).

The dynamic equation for the robot shown in Fig. 1 can be rewritten as:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -h\dot{\theta}_2 & -h(\dot{\theta}_1 + \dot{\theta}_2) \\ h\dot{\theta}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$$

where

$$\begin{aligned} M_{11} &= a_1 + 2a_3 \cos(\theta_2) + 2a_4 \sin(\theta_2), \\ M_{22} &= a_2, \\ M_{21} &= M_{12} = a_2 + a_3 \cos(\theta_2) + a_4 \sin(\theta_2), \\ h &= a_3 \sin(\theta_2) - a_4 \cos(\theta_2), \\ G_1 &= b_1 \cos(\theta_1) + b_2 \cos(\theta_1 + \theta_2), \\ G_2 &= b_2 \cos(\theta_1 + \theta_2), \end{aligned}$$

with

$$\begin{aligned} a_1 &= I_1 + m_1 l_{c1}^2 + I_e + m_e l_{ce}^2 + m_e l_1^2, \\ a_2 &= I_e + m_e l_{ce}^2, \quad a_3 = m_e l_1 l_{ce} \cos(\delta_e), \\ a_4 &= m_e l_1 l_{ce} \sin(\delta_e), \quad b_1 = m_1 g l_{c1} + m_e g l_1, \\ b_2 &= m_e g l_{ce}. \end{aligned}$$

The nominal parameters of the two-link manipulator are chosen as follows:

$$\begin{aligned} m_1 &= 5 \text{ kg}, \quad m_e = 2.5 \text{ kg}, \quad l_1 = 1.0 \text{ m}, \quad l_{c1} = 0.5 \text{ m}, \quad l_{ce} = 0.5 \text{ m}, \\ \delta_e &= 30^\circ, \quad I_1 = 0.36 \text{ kgm}^2, \quad I_e = 0.24 \text{ kgm}^2. \end{aligned}$$

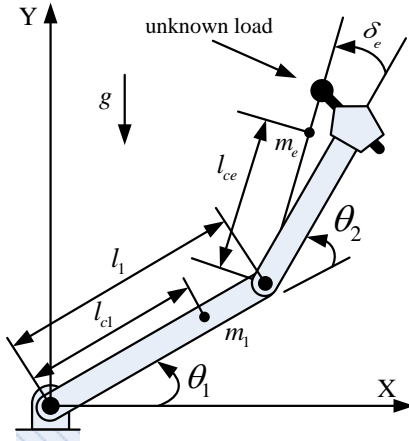


Fig. 1. An articulated two-link manipulator.

Position control, or also the so-called regulation problem is one of the most relevant issues in the operation of robot manipulators. This is a particular case of the motion control or trajectory control. The primary goal of motion control in joint space is to make the robot joints track a given time-varying desired joint position, $\theta^d = [\theta_1^d, \theta_2^d]^T$. Several control architectures related to robot control can be found in literature ranging from the simple PD, learning based, adaptive, and adaptive/learning hybrid controllers [9], [10]. The main advantage of the PD controller is that it can easily be implemented on simple microcontroller

architectures. On the other hand, the performance obtained from PD controllers is not satisfying for most of the sensitive applications [11]. Most of the other aforementioned types of controllers suffer from the complexities and the huge number of calculations needed to be carried out online.

III. IDEA AND OBJECTIVES

Artificial neural network (ANN) is used to get a controller of arm robot manipulator independent in the model of the robot. That can be made by using ANN to identify the robot inverse dynamics and other ANN control the system in parallel with a linear controller as shown in Fig. 2. The identification ANN gets the estimated input torque and edit by the actual input torque from the controllers. The controller ANN gets the required torque and edited by the output of the inverse dynamics from the ANN of identification. The linear control with manifold is used to get higher accurate and ensure in the stability of the system.

Trajectory tracking simulation test has been made for verifying the control scheme. The results are compared with the results of using the linear controller only. In order to observe how the controller behaves in the presence of various uncertainties and noisy parameters variation is used [6].

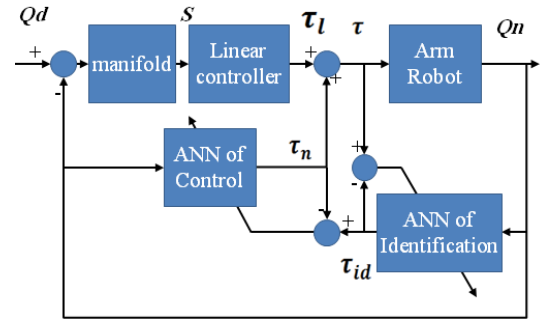


Fig. 2. Block diagram of the control scheme.

IV. ANN CONTROL SYSTEM DESIGN

A. ANN of the Inverse Dynamics Identification

An online identification of the inverse dynamics of the arm robot is made by multilayer ANN shown in Fig. 3. The identified torque, τ_{id} , get from the ANN by the equation:

$$\tau_{id} = V * f(W * Qn) \quad (2)$$

where the input is:

$$Qn = [\theta_1; \theta_2; \dots; \theta_n; \dot{\theta}_1; \dot{\theta}_2; \dots; \dot{\theta}_n] \in R^{2n} \quad (3)$$

where n is the number of joints in the arm robot. W is the weight matrix for the input layer and V is the weight matrix for the output layer. The activation function which used in the hidden layers is:

$$f(h) = \tanh(h) \quad (4)$$

where

$$h = W * Qn \quad (5)$$

The back propagation algorithm used for training the ANN and the error denote by:

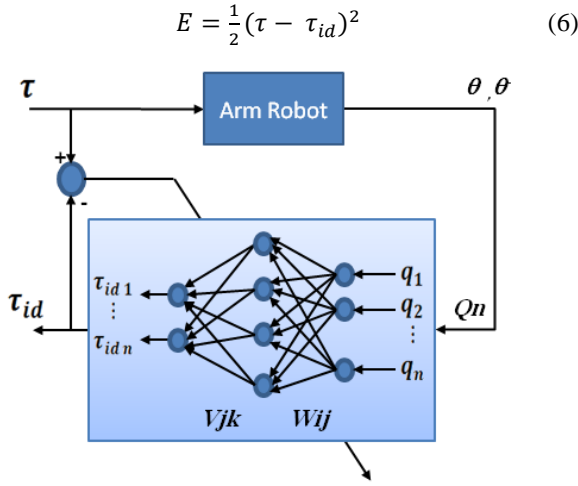


Fig. 3. Block diagram of the identification ANN.

B. ANN of the Control

This ANN, that shown in Fig. 4, takes online training on the estimated torque required to the robot arm. This estimated torque comes from the identification ANN and the error now is:

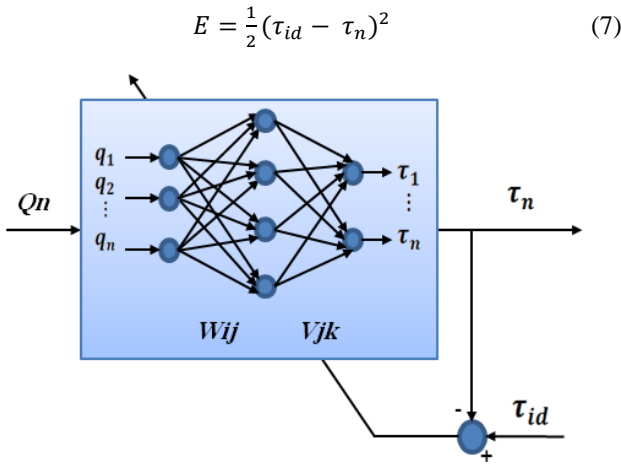


Fig. 4. Block diagram of control ANN.

And the other details of this ANN are the same as the previous one.

C. Linear Controller

For tracking planned joint trajectories, we design a manifold to describe the desired tracking performance of the robot as

$$S = 0 \quad (8)$$

where $S = \dot{e} + \lambda e$, $e = \theta - \theta_d$, and $\dot{e} = \dot{\theta} - \dot{\theta}_d$

θ_d and $\dot{\theta}_d$ are the planned joint trajectories, $\lambda \in \mathbb{R}^{n \times n}$ selected positive constant matrix.

τ_l is control input of the linear controller, and can be simply described as :-

$$\tau_l = -kS \quad (9)$$

where $k \in \mathbb{R}^{n \times n}$ is a positive-definite gain matrix [1].

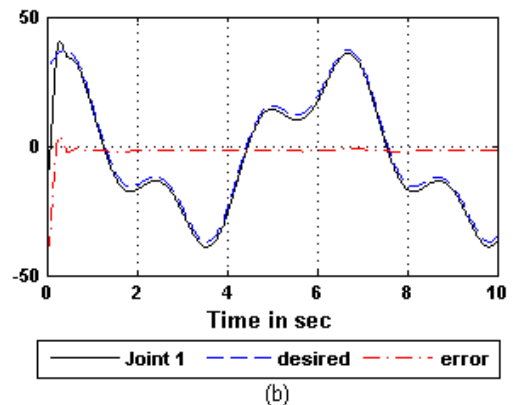
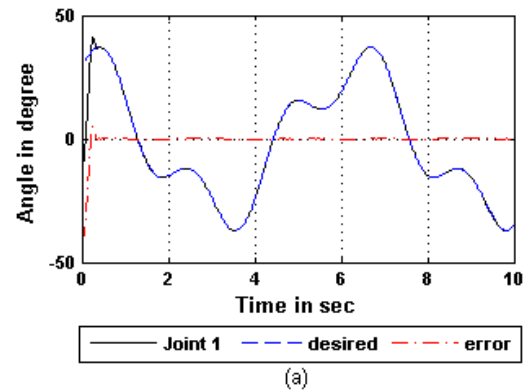
Then the total input torque from the control scheme is:

$$\tau = \tau_n + \tau_l \quad (10)$$

V. RESULTS AND DISCUSSIONS

Simulation is used to show the effect of the ANN scheme in control two link arm robot shown in section II. A comparison between the results of the linear control only and ANN scheme is shown in the figures. Fig. 5 shows the desired trajectory for θ_1, θ_2 respectively and the response of them using the linear controller only in Fig. 5 (b, d) and using ANN control scheme in Fig. 5 (a, c). Fig. 6 shows more clearly a comparison between the trajectory tracking error of the two joints e_1, e_2 using the linear controller only Fig. 6(b) and using ANN control scheme Fig. 6(a). The error tracking in Fig. 6 shows that the linear controller has an oscillatory error and after using ANN control scheme the error near to zero. Fig. 7 (a) shows the effect of adding ANN controller on the torque input compared with the torque input using linear controller only showed in Fig. 7(b). These figures show that adding the ANN control scheme improve the response of the two joint by decreasing the trajectory tracking error appears in Fig. 6.

Another simulation is used to show the effect of the mass variation on the results of the system. Fig. 8 shows the mass variation of the two masses m_1 and m_e . It is assumed that they vary randomly with time every 0.3 s. The mass of the base link m_1 varies in the range of $5 \rightarrow 7$ kg (the nominal mass is 5 kg) and the mass of the elbow link m_e varies in the range of $2 \rightarrow 5$ kg (the nominal mass is 2.5 kg). Fig. 9 shows the trajectory tracking error in the two joints according to the desired trajectories and the response of the joints shown in Fig. 10. Fig. 11 (a) shows the effect of adding ANN controller on the torque input compared with the torque input using linear controller only showed in Fig. 11 (b). These figures show that using the ANN controller in our scheme improved the response of the linear controller and it can work within mass variation.



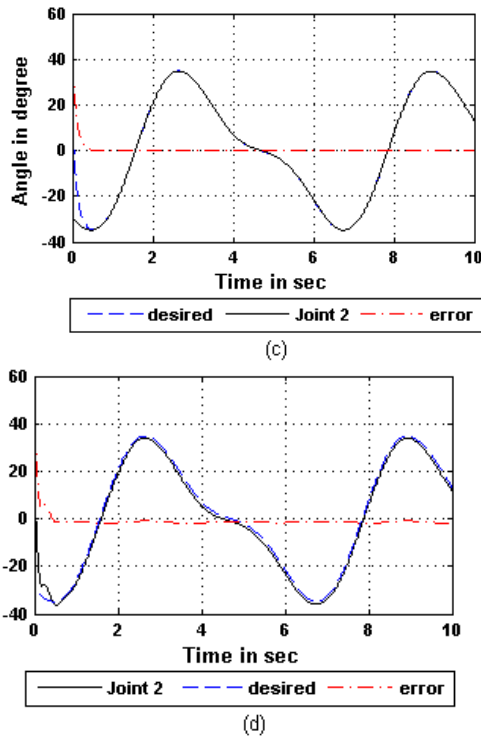


Fig. 5. The first joint response and error (a) using ANN controller; (b) using linear controller only and the second response and error; (c) using ANN controller; (d) using linear controller only.

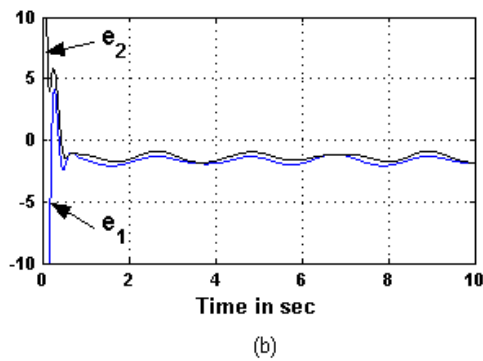
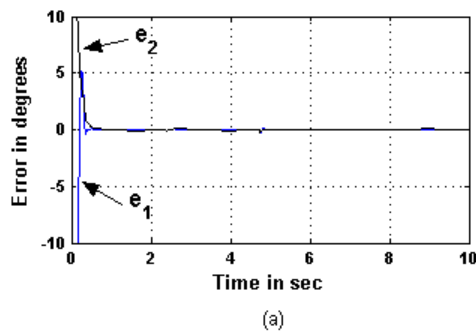


Fig. 6. Tracking error of the two joints (a) using ANN controller; (b) using linear controller only.

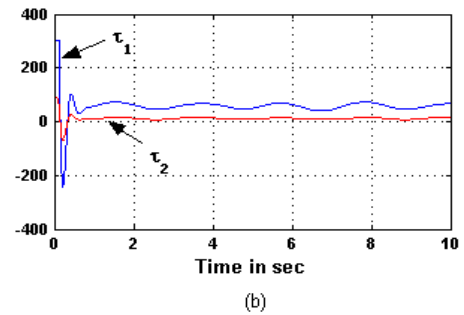
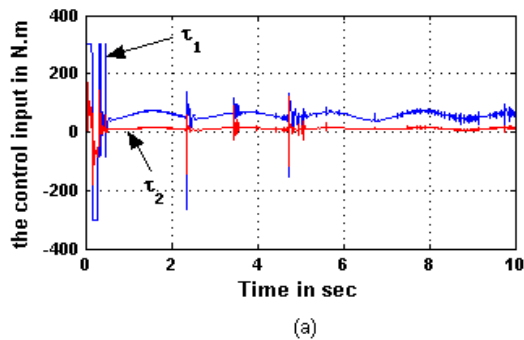


Fig. 7. (a) The total input torque of the linear and ANN controllers; (b) The input torque when using the linear controller only.

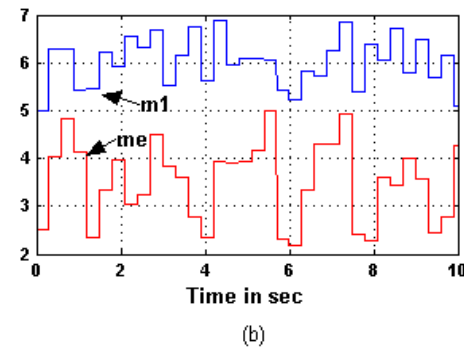
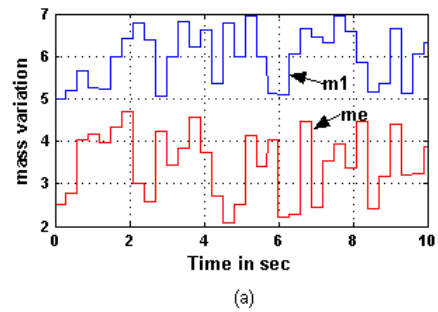


Fig. 8. (a) Mass variation of the two links (m1, me) in the ANN controller test; (b) mass variation of the two links (m1, me) in the linear controller test.

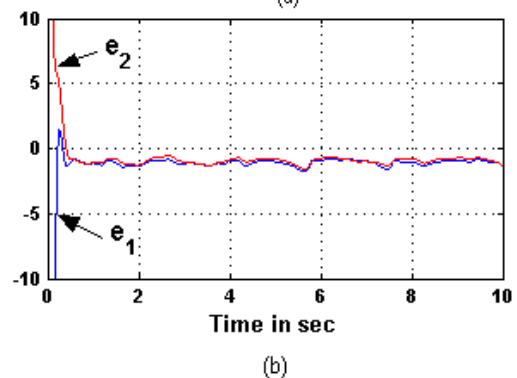
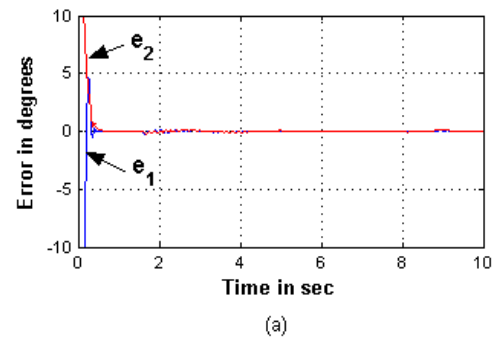


Fig. 9. Tracking error of the two joints in mass variation test (a) using ANN controller; (b) using linear controller only.

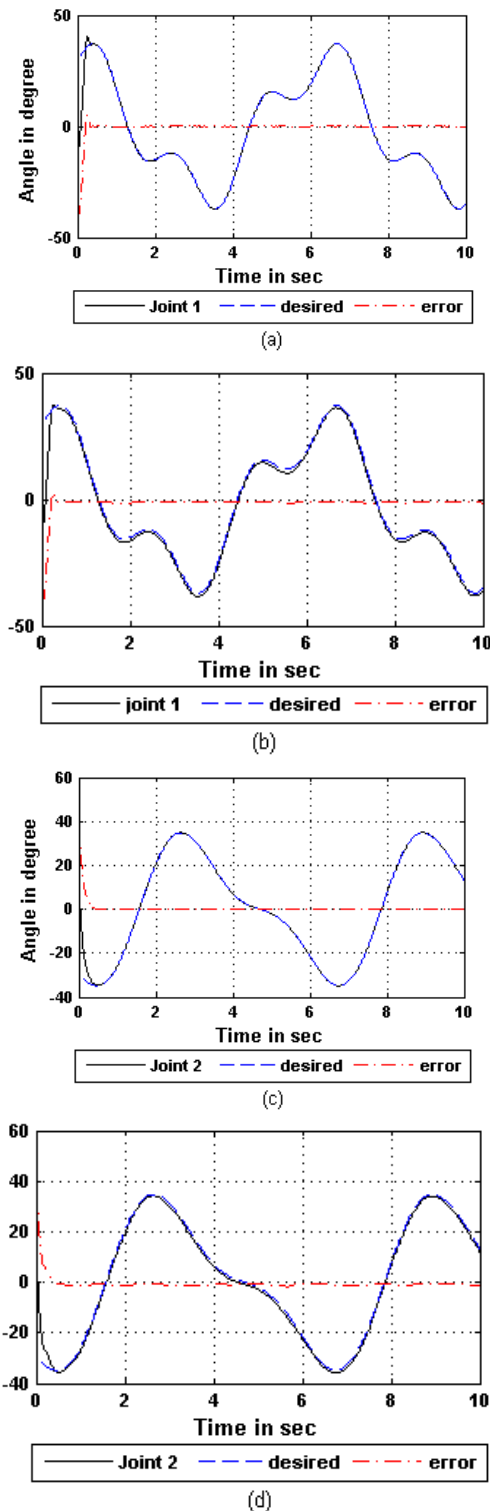


Fig. 10. Joint 1 response and error in the mass variation test (a) using ANN controller; (b) using linear controller only and Joint 2 response and error; (c) using ANN controller; (d) using linear controller only.

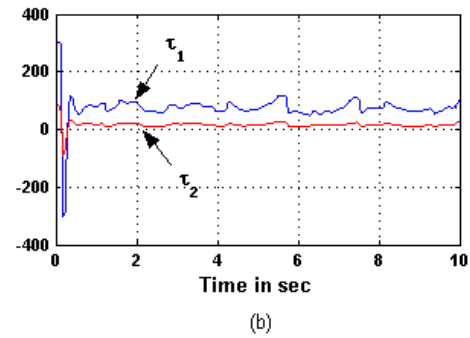
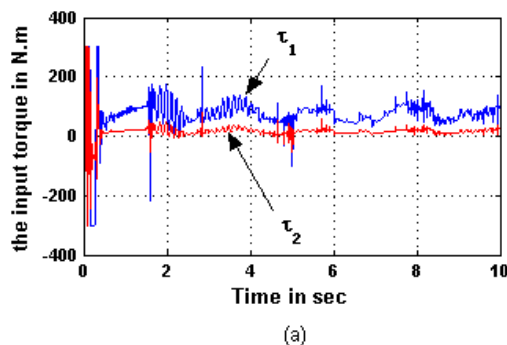


Fig. 11. The total input torque in the mass variation test using (a) linear and ANN controllers; (b) The input torque when using the linear controller only.

VI. CONCLUSION

An ANN identification and control scheme is developed for a model-free arm robot control. The scheme depends on two ANN and linear control to control the arm robot without using the model dynamics or parameters. The first ANN work as an inverse dynamic identification of the arm robot and the second ANN work as a controller for the arm in parallel with the linear controller. The identification ANN is trained by the total input torque and gives the estimated torque required to control it. The control ANN is trained by the estimated torque required that given by the identification ANN and feed a control signal that add to the linear control signal. The linear controller is required to save the stability of the arm robot.

The scheme is applied on a simulated complex model of two link arm robot with a payload and the results are compared with the results of using linear controller only. The results show that adding ANN controller has a clear effect on the tracking error. Using the ANN scheme improves the linear controller response. Searching in this trend will lead to generate a general controller that can control any system independent to its model.

REFERENCES

- [1] Z. H. Jiang and T. Ishita, "A neural network controller for trajectory control of industrial robot manipulators," *Journal of Computers*, vol. 3, no. 8, August 2008.
- [2] M.-S. Lan, "Adaptive control of unknown dynamical systems via neural network approach," in *Proc. American Control Conference*, June 1989, pp. 910-915.
- [3] Y. I. Al-Mashhadany, "Recurrent neural networks (RNNS) controller for dynamical system," *International Journal of Information Sciences and Computer Engineering*, vol. 2, no. 1, pp. 7-12, 2011.
- [4] P. Joel Perez, J. P. Perez, R. Soto, A. Flores, F. Rodriguez, and J. L. Meza, "Trajectory tracking error using pid control law for two-link robot manipulator via adaptive neural networks," *Procedia Technology Journal*, pp. 139-146, 2012.
- [5] W. Zeng and C. Wang, "Learning from NN output feedback control of robot manipulators," *Neurocomputing Journal*, vol. 125, pp. 172-182, 2014.
- [6] A. B. Sharkawy, "A computationally efficient fuzzy control scheme for a class of MIMO systems," *Alexandria Engineering Journal*, vol. 52, pp. 583-594, December 2013.
- [7] M. Sun, S. S. Ge, and I. M. Y. Mareels, "Adaptive repetitive learning control of robotic manipulators without the requirement for initial repositioning," *IEEE Transactions on Robotics*, pp. 563-568, 2006.
- [8] T.-H. S. Li and Y.-C. Huang, "MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators," *Information Sciences Journal*, pp. 4641-4660, 2010.
- [9] S. Liuzzo and P. Tomei, "A global adaptive learning control for robotic manipulators," *Automatica Journal*, pp. 1379-1384, 2008.

- [10] S. Yamacli and H. Canbolat, "Simulation of a SCARA robot with PD and learning controllers," *Simulation Modelling Practice and Theory Journal*, pp. 1477-1487, 2008.
- [11] T. Das and C. Dulger, "Mathematical modeling, simulation and experimental verification of a SCARA robot," *Simulation Modelling Practice and Theory Journal*, vol. 13, pp. 257-271, 2005.



M. Dinary received his B.Sc. degree in mechatronics and industrial robotics program from Minia University, Minia, Egypt, in 2012. He is working now as a teaching assistant at the same faculty. His research interests include control of robot manipulators, arm robots simulation and model-free control.



Abou-Hashema M. El-Sayed received his B.Sc. and M.Sc. degrees in electrical engineering from Minia University, Minia, Egypt, in 1994 and 1998, respectively. He was a Ph.D. student in the Institute of Electrical Power Systems and Protection, Faculty of Electrical Engineering, Dresden University of Technology, Dresden, Germany from 2000 to 2002. He received his Ph.D. degree in electrical power from the Faculty of Engineering, Minia University, Egypt in 2002, according to a channel system program, which means a Scientific Co-operation between the Dresden University of Technology, Germany and Minia University, Egypt. Since 1994, he has been with the Department of Electrical Engineering, Faculty of Engineering, Minia University, as a teaching assistant, a lecturer assistant, and since 2002, as an assistant professor. He was a visiting researcher at Kyushu University, Japan, from 2008 to 2009. He is the head of Mechtronics and Industrial Robotics Department, Faculty of Engineering, Minia University from 2011 till now.

His research interests include protection systems, renewable energy, power systems, Mechtronics and Robotics.



Abdel Badie Sharkawy received the B.Sc. degree in mechanical engineering, the M.Sc. degree in production engineering from Assiut University, Egypt, in 1981 and 1990 respectively. He received the Ph.D. degree in control engineering from the Slovak Technical University (STU) in Bratislava in 1999. He was a senior lecturer at the Department of Mechatronics Engineering, the Hashemite University, Jordan for three years from 2001 to 2004 and the Electrical Engineering Department, Al-Tahady University, Sirte, Libya during the fall semester, 2005. He is currently a professor at the Mechanical Engineering Department, Assiut University, Egypt since April 2012. His research interests include adaptive fuzzy identification and control, automotive control systems, robotics (modeling and control), and the use of neural networks in the control of mechanical systems.



G. Abouelmagd graduated from Faculty of Eng., Minia University in 1981. He held an M.Sc degree from Minia University in 1987 and Ph.D degree in mechanical engineering from Minia University-Technical Aachen University (Germany-Egypt Channel System) in 1992. He is working as a professor from August 29, 2006 up to February 7, 2008 in Yanbu Industrial College, Yanbu, KSA. And a professor in the Production Engineering and Design Department, Minia University, Minia, Egypt since Septebmer 24, 2008. His research interest is in material characterization.

