Journal of Modern Technology and Engineering
Vol.3, No.3, 2018, pp.258-264

MODELING AND CONTROLLING OF A 2 DOF ROBOT MANIPULATOR WITH ARTIFICIAL NEURAL NETWORKS

Bekir Cirak*

Karamanoglu Mehmetbey University, Engineering Faculty, Mechanical Engineering Department, Yunus Emre Campuses, Karaman, Turkey

Abstract. In this study, 2 DOF (Degree Of Free) robot manipulator were inspected. kinematic and dynamic calculations were made for robot joints and Artificial Neural Network (ANN) method was applied for control. the coordinate of the end point of the robot is assumed to be input. Robot kinematic and dynamic calculations and simulations were done with Matlab 14 a version.

Keywords: Robot manipulator, Artificial Neural Networks, kinematic, dynamic.

Corresponding author: Bekir Cirak, Karamanoglu Mehmetbey University, Engineering Faculty, Mechanical Engineering Department,Yunus Emre Campuses, Karaman, Turkey, e-mail: *bcirak@kmu.edu.tr*

Received: 17 July 2018; Accepted: 15 October 2018; Published: 07 December 2018.

1 Introduction

Robots have the potential to play a large role in our world. They are currently widely used in industrial applications for labor-intensive operations that require a high level of precision and repetition. In addition, robots can be found in the entertainment industry in the form of toys and animatronics. The function of robots in society is constantly evolving and current research endeavors to bring them further into the realm of domestic assistance, medicine, military, search and rescue, and exploration. In many of these applications, the robot must perform only one specific task and thus can be designed to handle a single operation. However, as the potential use for robots grows, so does their need to interact with objects in their environment (Aly, 2010). Robots are defined as systems that perform location and direction change operations by programmed transport. Moving is a movement at a distance far from the body dimensions. Manipulator consists of multiple joints connected to each other. Movements of the joints are provided by the engine. The robot consists of mechanical parts, actuators and control units. The mechanical parts of the robot are classified as structural parts, power transmitting parts, bearings and coupling parts. The motors can also act as pneumatic or hydraulic as well as electrically. In modern robots, the control units are PC based and have advanced structures (Dhaval et al., 2013).

Today, the usage areas of robots are increased. Especially in industrial environments, a large number of robots are used for production assembly and similar works. Because the separation media properties change rapidly, in the real world robots are required to reach the desired target without hitting the obstacles they encounter. The robots, the control technique used and the joint they contain according to their species. In terms of control technique it is possible to classify as adaptive robots, non-adaptive and intelligent robots. According to the joint types in the robots are grouped as rotary, prismatic, cylindrical, spherical and planar. In recent years, it is possible to find intelligent system applications on different stages of production. These intelligent systems are an important part of industry, health care and automation sector. In this study, the dynamic behavior of the 2-degree of free industrial robot manipulator and the change

of the robot arm configuration between the joints with time are examined and the articulated neural network model of the joints given mathematical kinematic models has been inspected (Hala & Adel, 2002).

2 Robot kinematics

Two linked or two degree of freedom manipulators are shown in Fig.1. It is assumed that the shaped O_2 point is fixed. There are servo motors at O and O_1 points. Manipulator consists of two rotating joints and two links. Basic and local coordinate systems are placed on the manipulator and kinematic calculation is started. The basic coordinate system at point O_1 is considered fixed. Local coordinate systems are located at O_1 and O_2 points. The rotation of the joints is around the z-axis, perpendicular to the paper plane. By means of the transformation matrices expressing the relation between the neighboring links,the position and orientation in the robot arm are determined according to the basic coordinate system. Similarly,using the inverse of the transformation matrices, the values of θ_1 and θ_2 can be calculated when the robot hand is in any position. This process is called inverse kinematics. Kinematic parameters and transformation matrices are determined by the Denavit Hartenberg method.

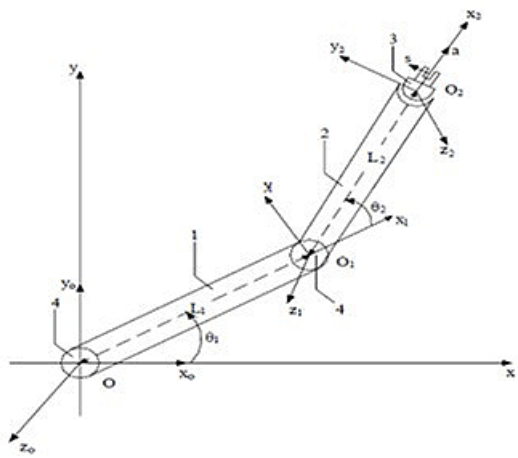

Figure 1: Physical structure of the 2 DOF manipulator

Basic parameters of the manipulator; M_1 and $M_2$are the mass of the first and second link. L_1 and L_2 are the lengths of the links. $M_1 = M_2 = 6,5kg$, $L_1 = L_2 = 0,75m$, gravity acceleration, $g = 9,81m/s^2$

3 Manipulator dynamics

The dynamic equation of a manipulator with n degrees of freedom is shown in 1).

$$T = D(q)\ddot{q} + h(q,\dot{q})\dot{q} + c(q) \tag{1}$$

where, the matrix T is $n \times 1$ in size. Describes the generalized rotational moments that effect joints. The matrix D with dimension $n \times n$ shows the effect of accelerating inertia masses. The h matrix of size $n \times 1$ shows the centrifugal and Coriolis effect. The matrix c in $n \times 1$ dimension shows the rotational moment acting on the joints due to gravity. q, which is defined as a generalized coordinate and is of dimension $n \times 1$, shows the angular displacement of the joints. Therefore, $\ddot{q}$ represents angular acceleration ($\ddot{\theta}$), and $\dot{q}$ represents angular velocity ($\dot{\theta}$). The equation (2) expresses the dynamic model of the manipulator.

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \times \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{2}$$

$$D_{11} = \frac{1}{3}L^2\left(M_1 + 4M_2 + 3M_2\cos\theta_2\right) \tag{3}$$

$$D_{12} = \frac{1}{3}M_2L^2 + \frac{1}{2}M_2L^2\cos\theta_2 \tag{4}$$

$$D_{21} = \frac{1}{3}M_2L^2 + \frac{1}{2}M_2L^2\cos\theta_2 \tag{5}$$

$$D_{22} = \frac{1}{3}M_2L^2 \tag{6}$$

$$h_1 = \frac{1}{2}L^2\sin\theta_2 \times M_2\dot{\theta}_2\left(\dot{\theta}_2 + \dot{\theta}_1\right) \tag{7}$$

$$h_2 = \frac{1}{2}L^2\sin\theta_2 \times M_2\dot{\theta}_1\dot{\theta}_1 \tag{8}$$

$$c_1 = \frac{1}{2}gL\left(M_1\cos\theta_1 + M_2\cos\left(\theta_1 + \theta_2\right) + 2M_2\cos\theta_1\right) \tag{9}$$

$$\mathrm{c}_2 = \frac{1}{2}\mathrm{gLM}_2\cos(\theta_1 + \theta_2) \tag{10}$$

Considering the problem of bringing the robot manipulator from a starting position to a target position within a fixed time interval. System kinematics can be used to calculate joint angles corresponding to the starting and target positions. What is desired here is to obtain a time-dependent $q(t)$ function for each joint between the starting position and the target position at a constant time interval.

4 Artificial neural networks (ANN)

Artificial neural networks are a method in which the learning of the human brain is tried to be applied. An artificial neural network is made up of information processing elements called neurons. A unique point of contact neurons. A unique point of contact there is weight. These calculations, which are computable, convey information directly. It is not possible to determine in advance the points of connection since the information is often disseminated to the network. For this reason, a learning algorithm in which weights are changed is needed. The purpose of the study is to calculate the point weights for untrained and untrained learning. In many cases, the network is trained using input / output pairs. The performance of this learning process is measured by using the training set to achieve the desired result and by generalizing the trained network. Educational learning is a two-layered forward feeder with the simplest network structure and an output layer. Each neuron in the output layer is signaled by all input neurons with recalculable weights. Fig. 2 shows the schematic structure of a multi-layer feed-forward network. In network input layer hidden three layers, the layer and the output layer located (Meza et al., 2012).

The BPNN (Backpropagation Neural Network) is a method for categorization and prediction 0-0. The structure mainly contains input layer, output layer, and hidden layer, as shown in Fig. 2(a). To achieve the target value, the Gradient Steepest Descent Method is applied to renew the weights, w, and the biases, b, by conveying the error gradient repeatedly. The procedure of BPNN can be divided into two phases, the feed-forward phase and the propagation phase, as shown in Fig. 2(b). The procedures are explicitly explained as below.

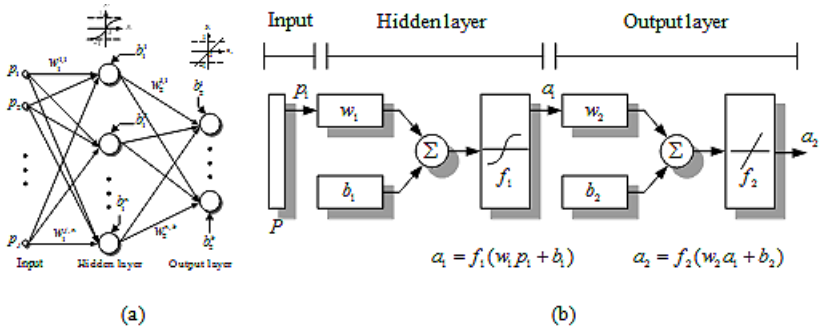

Figure 2: (a) Structure of BPNN. (b) Flowchart of BPNN.

Once the input vector is identified in artificial neural networks, the weights are adjusted according to the learning process. BPNN algorithm is used in this study. BPNN algorithm is the most used algorithm in artificial neural networks. During the back propagation learning, the network passes each input pattern through the neurons in the hidden layers to produce the output neurons. It then compares the result obtained with the expected result to find the errors in the output layer. So, the output errors are passed back to the hidden layers directly from the derivative output layer. Once the error values are found, the neurons adjust their weights to reduce their faults. The weight change equations are arranged in the smallest shape of the average error margin in the network. The learning algorithm is specified by the notation given below (Rasit & Abdullah, 2004). In n. iteration, j. Error mark at exit of neuron, j. On the neuron output layer,

$$ej(n) = dj(n) - yj(n) \tag{11}$$

In this work, sigmoid function is used as an activation function in the calculation of local gradients. The YSA algorithm was implemented using Matlab 14.a Neural Network Toolbox. The input and output data are very important in terms of convergence and learning process. Due to the nature of the sigmoid function, the output of the aging is between 0 and 1. For this reason, it is necessary to normalize the input / output data without any network training. In this study, input and output data were normalized to remain between 0-1 (Soltanpour & Jafaar, 2012), (Zhu & Zhang, 2011)

5 Simulation

The success of the designed control system is due to the fact that the two degree of freedom manipulator answers were evaluated and evaluated. A software that updates the linearization coefficients and supervisor gains along the trajectory tracked by the manipulator is written in Matlab 14a. The parameters are updated 100 times over 10 seconds intervals. New control techniques have been developed for the inspection of robot manipulators. The most important element in robot manipulator control is to follow the desired trajectory. The angular speed deviations are mostly at the beginning of control and in the case of load drop. Simulation results show that the algorithms adapt themselves very quickly to the situation. These deviations are small enough to be neglected.

The robot control with ANN in simulations is given in Fig.2. As a ANN entry used in the control algorithm, the robot manipulator has been given the normalized opening ratios of the joints. ANN output is the control voltages applied to motors mounted for robot manipulator. A $3 \times 5 \times 1$ triple layer was used for the back propagation algorithm. The robot control outputs are shown in Fig.3 and 8. Figures 3-8 show the actual acceleration-time graphs of the first and second joints.

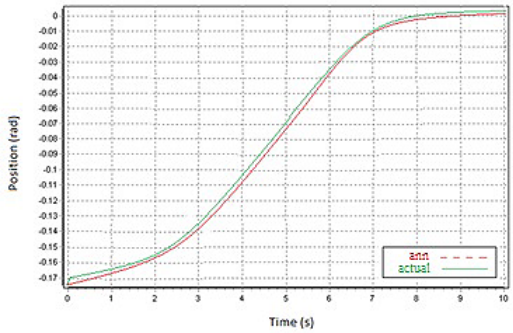

Figure 3: Position-time diagram for 1. joint

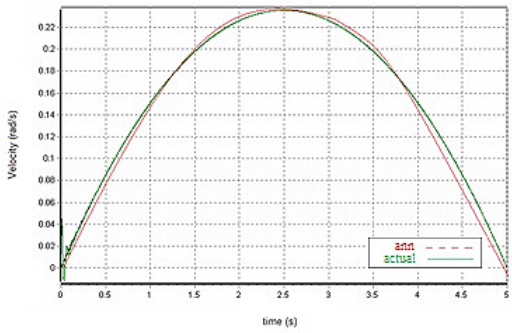

Figure 4: Velocity-time diagram for 1. Joint

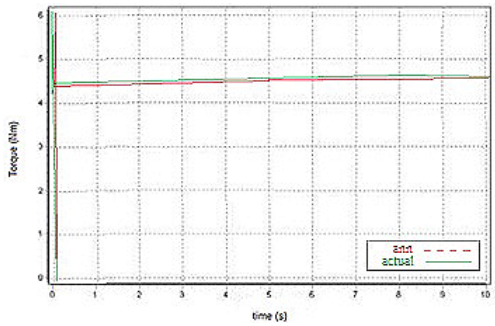

Figure 5: Torque-time diagram for 1. Joint

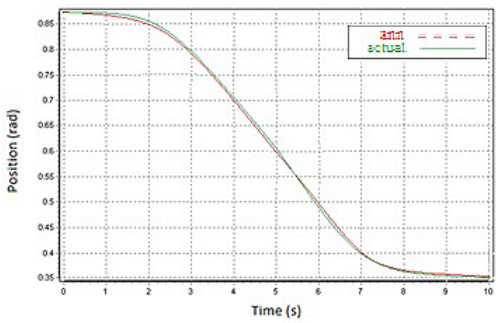

Figure 6: Position-time diagram for 2. joint

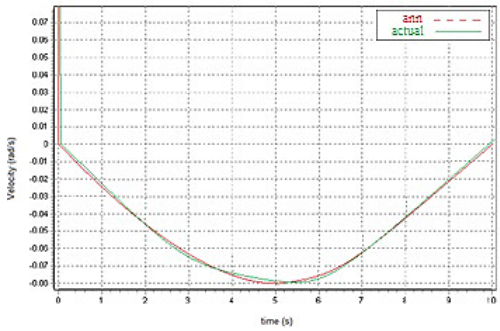

Figure 7: Velocity-time diagram for 2. joint

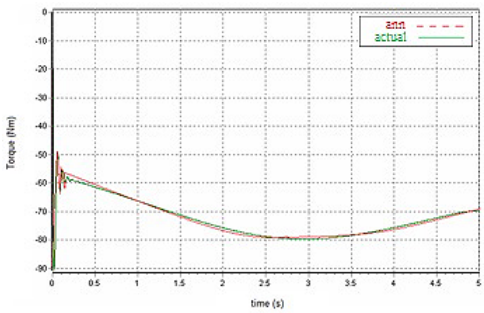

Figure 8: Torque-time diagram for 2. Joint

6 Conclusions

Two joint robot manipulators given with robot dynamics and straight kinematic equations were performed with the supervised ANN. Robot manipulator inspections are done in several different ways. However, when the robot manipulator control was performed with ANN, the control element was given an adaptation to the system by using a sufficient accuracy learning algorithm, and the position and location information of the robot joints could be determined without using the sensor. Robot manipulator control, which is important for manipulator control, can move from any starting position of a robot manipulator end to a desired end position with minimum error. The fewer the squares of angular velocity errors of a joint, the less the robot manipulator shakes. Again, the smaller the angular misalignment of the joints, the farther the distal end of the robot manipulator to the target point is, the less the error is. In the study, the independent joint control has been successfully carried out as a result of the reduction of the interaction between the joints.

References

Aly A.A. (2010). Intelligent Fuzzy Control for Antilock Brake System with Road-Surfaces Identifier, IEEE International Conference on Mechatronics and Automation (ICMA 2010) China, August 4-7.

Dhaval, V., Ohri, J. & Ankit, P. (2013). Comparison of Conventional & Fuzzy based Sliding Mode PIDController for Robot Manipulator. In: IEEE International Conference on Individual and Collective Behaviors in Robotics, 115-119.

Hala, B.N.D. & Adel, M.A. (2002). Fuzzy control of robot manipulators: some issues on design and rule base size reduction. *Eng. Appl. Artif. Intell.* 15, 401-416.

Kirby, R., Forlizzi, J. & Simmons, R. (2010). Affective social robots. *Robot. Auton. Syst.,* 58, 322-332.

Kosmatopoulos, E.B., Chassiakos, A. & Christodoulo, M.A. (2005). Robot Identification using dynamical neural networks. *Engineering Systems with Intelligence*, Kluwer Academic Publishers,187-195.

Liu, J. & Wang, X. (2011). Sliding Mode Control for Robot. Springer, Berlin.

Meza, J.L., Santibanez, V., Soto, R. & Llama, M.A. (2012). Fuzzy self-tuning PID semiglobal regulator for robot manipulators. *IEEE Trans. Ind.Electron*, *59*(6), 2709-2717.

Rasit, K. & Abdullah, F. (2004). Model based intelligent control of a 3-joint robotic manipulator: a simulation study using artificial neural networks. *ISCIS, Springer* , Berlin, 31-40.

Soltanpour, M.R., Jafaar, K. & Soltani, M. (2012). Robust nonlinear control of robot manipulator with uncertainties in kinematics, dynamics and actuator models. *Int. J. Innovative Comput. Inf.Control*, 8, 5487-5498.