

Neural Network Modeling of a Flexible Manipulator Robot

Rahma Boucetta and Mohamed Naceur Abdelkrim

University of Gabes, Engineering School of Gabes
Omar Ibn Khattab Avenue, Zrig, 6029, Gabes, Tunisia
rboucetta@yahoo.fr, naceur.abdelkrim@enig.rnu.tn

Abstract. This paper presents an artificial neural networks application for a flexible process modeling. A flexible planar single-link manipulator robot is considered. The dynamic behavior of this process is described using Lagrange equations and finite elements method. The artificial neural networks are all variations on the parallel distributed processing (PDP) idea. The architecture of each network is based on very similar building blocks which perform the processing. Therefore, two feed-forward and recurrent neural networks are developed and trained using back-propagation algorithm to identify the dynamics of the flexible process. Simulation results of the system responses are given and discussed in terms of level of error reduction. Finally, a conclusion encloses the paper.

Keywords: Flexible manipulator, dynamic model, finite elements method, Lagrange equations, neural networks.

1 Introduction

Robotic manipulators are generally built using heavy material to maximize stiffness, in an attempt to minimize system vibration and achieve good positional accuracy. As a consequence, such robots are usually heavy with respect to the operating payload. The operation speed of the robot manipulation is limited, so the actuators size is increased boosting energy consumption and increasing the overall cost. Moreover, the robot has a low payload to robot weight ratio. In order to solve these problems, robotic manipulators are designed to be lightweight [13].

Conversely, flexible manipulators exhibit many advantages over their rigid counterparts: they require less material, are lighter in weight, have higher manipulation speed, lower power consumption, require smaller actuators, are more maneuverable and transportable, are safer to operate due to reduced inertia, have enhanced back-drive ability due to elimination of gearing, have less overall cost and higher payload to robot weight ratio [13].

A first wave of interest of neural networks emerged after the introduction of simplified neurons by McCulloch and Pitts in 1943. These neurons were presented as models of biological neurons and as conceptual components for circuits that could perform computational tasks [2, 3, 8].

Most neural networks funding was redirected and researchers left the field. The interest in neural networks re-emerged only after some important theoretical results were attained in the early eighties, and new hardware developments increased the processing capacities.

Artificial neural networks can be most adequately characterized as computational models with particular properties such as the ability to adapt or learn, to generalize, or to cluster or organize data, and which operation is based on parallel processing. The intriguing question is to which extend the neural approach proves to be better suited for certain applications than existing models [1, 3, 5].

A connectionist system consists of a set of interconnected processing elements and is capable of improving its performance based on past experimental information [7]. An artificial neural network is a connectionist system that was originally proposed as a simplified model of the biological nervous system [8]. Neural networks have been shown to provide an efficient means of learning concepts from past experience, abstracting features from uncorrelated data, and generalizing solutions from unforeseen inputs. Other promising advantages of neural networks are their distributed data storage and parallel information flow, which cause them to be extremely robust with respect to malfunctions of individual devices as well as being computationally efficient.

There have been many architectures (i.e. schema consisting of various neurotic characteristics, interconnecting topologies, and learning rules) proposed for neural networks over the last ten years. Simulation experience has revealed that success is problem-dependent. Some networks are more suitable for adaptive control whereas others are more appropriate for pattern recognition, signal filtering, or associative searching. Neural networks that employ the well known back-propagation learning algorithm [9] are capable of approximating any continuous functions with an arbitrary degree of accuracy [9].

This paper is organized as follows: The flexible manipulator system is presented in the first paragraph. The dynamic robot behavior is explained in the next paragraph. The following paragraph deals with applied topologies of artificial neural networks to model the flexible manipulator robot. Simulation results are showing and discussing with each useful architecture.

2 The Flexible Manipulator System

A schematic representation of the single-link flexible manipulator system is shown in figure 1, where a control torque $\tau(t)$ is applied at the hub by an actuator motor with E , I , ρ , L and I_H represent Young's modulus, second moment of area, mass density per unit volume, length, and hub inertia moment respectively [10, 12].

The angular displacement of the link in the X_0OY_0 coordinates is denoted by $\theta(t)$. $w(x,t)$ represents the elastic deflection of the manipulator at a distance x from the hub, measured along the OX axis. X_0OY_0 and XOY represent the stationary and moving frames respectively.

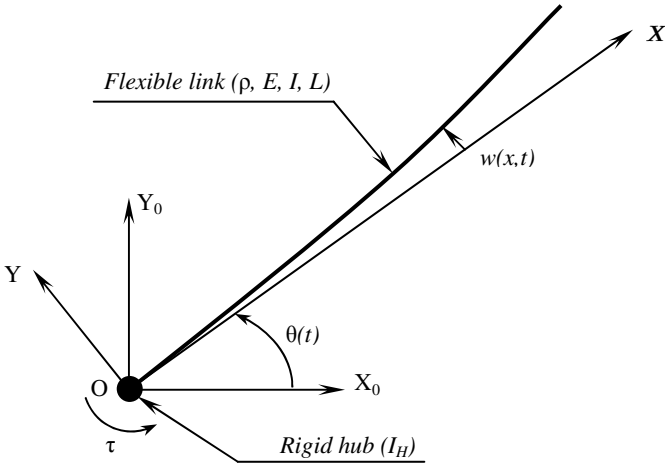


Fig. 1. Flexible manipulator scheme

The height (width) of the link is assumed to be much greater than its depth, thus allowing the manipulator to vibrate dominantly in the horizontal direction (X_0OY_0 plane). To avoid difficulties arising from time varying lengths, the length of the manipulator is assumed to be constant. Moreover, the shear deformation, the rotary inertia and the effect of axial force are ignored. For an angular displacement θ and an elastic deflection w , the total displacement $y(x,t)$ of a point along the manipulator at a distance x from the hub can be described as a function of both the rigid body motion $\theta(t)$ and the elastic deflection $w(x,t)$ [6, 12, 14], i.e.

$$y(x,t) = x\theta(t) + w(x,t) \tag{1}$$

Thus, by allowing the manipulator to be dominantly flexible in the horizontal direction, the elastic deflection of the manipulator can be assumed to be confined to the horizontal plane only.

Kinetic energy of the flexible manipulator, depending of hub rotation and modes rotation in the X_0OY_0 and XOY frames, has the following expression [6, 14]

$$T = \frac{1}{2}I_H\dot{\theta}^2 + \frac{1}{2}(\dot{q} + L\dot{\theta})^T M (\dot{q} + L\dot{\theta}) \tag{2}$$

Potential energy just depending of link flexibility has the following form

$$V = \frac{1}{2}q^T Kq \tag{3}$$

After applying Lagrange's equations, the dynamic model can be written as

$$\begin{aligned} (I_H + L^T ML)\ddot{\theta} + L^T M\ddot{q} &= \tau \\ M\ddot{q} + L^T M\ddot{\theta} + Kq &= 0 \end{aligned} \tag{4}$$

where M, K and L are the mass matrix, the stiffness matrix and the length array respectively, and q is the elastic modes vector.

3 Dynamic Behavior

The dynamic equations can be presented in a state-space form as

$$\begin{aligned} \dot{v} &= Av + Bu \\ y &= Cv + Du \end{aligned} \tag{5}$$

where the state-space matrices are

$$A = \begin{pmatrix} 0 & I \\ -M^{-1}K & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ M^{-1} \end{pmatrix}, C = (I \ 0), D = (0) \tag{6}$$

The state and control vectors are given by

$$\begin{aligned} v^T &= (\theta \ q_1 \ q_2 \ \dots \ \dot{\theta} \ \dot{q}_1 \ \dot{q}_2 \ \dots) \\ u^T &= (\tau \ 0 \ \dots) \end{aligned} \tag{7}$$

In order to simulate the flexible manipulator system, an aluminum flexible link of dimensions $L=0.61\text{m}$ and $S=3 \times 10^{-5}\text{m}^2$, with $E=200 \times 10^9\text{N/m}^2$, $I=2.5 \times 10^{-12}\text{m}^4$, $I_H=4.3 \times 10^{-3}\text{kg.m}^2$ and $\rho=7.8 \times 10^3\text{kg/m}^3$ is considered. The link is discretized into two elements.

Solving the state-space matrices gives the vector of states v, that is, the hub angle, the elastic modes and their velocities. The derived dynamic model is a nonminimum phase system, not strictly proper, and unstable. Also, the model has zeros very close to the imaginary axis; this deteriorates the time domain performance of the closed-loop system [11, 15].

Generally, linear models of flexible structures used in design of controllers are derived under restrictive assumptions which are often not valid for large motions that occur during slewing maneuvers. Hence, considerable uncertainty in the linear model exists. Another feature characteristic of lightly damped systems is the occurrence of poles ($\pm 159.67j$, $\pm 37j$, 0 , 0) and zeros ($\pm 158j$, $\pm 25j$) very close to the imaginary axis that gives rise to ill-conditioned systems. The state-space matrices arising out of such systems have largely separated singular values, posing considerable computational difficulty in controller design. In the spectral density given by figure 2, the vibration frequencies of the system are obtained as 37rad/s and 160rad/s , i.e. 5.9Hz and 25.46Hz , and the magnitude of frequency response for the two resonance modes are 122dB and 68.8dB .

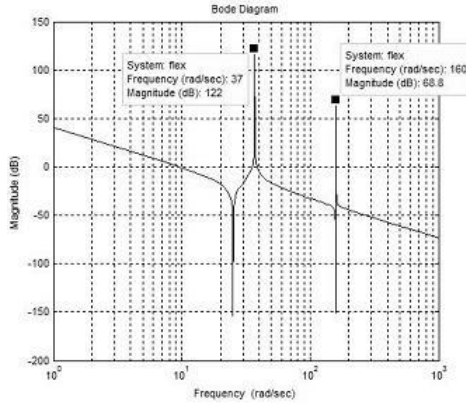


Fig. 2. Open-loop frequency response

4 Neural Modeling

4.1 Introduction

An artificial neural network consists of a pool of simple processing units which communicate by sending signals to each other over a large number of weighted connections [1, 2].

Each unit performs a relatively simple job: receive input from neighbors or external sources and use this to compute an output signal which is propagated to other units. Apart from this processing, a second task is the adjustment of the weights. The system is inherently parallel in the sense that many units can carry out their computations at the same time [4, 7, 8].

In most cases, we assume that each unit provides an additive contribution to the input of the unit with which it is connected. The total input to unit k is simply the weighted sum of the separate outputs from each of the connected units plus a bias or offset term b_k

$$s_k(t) = \sum_j w_{jk}(t)y_j(t) + b_k(t) \tag{8}$$

4.2 Network Topologies

The pattern of connections between the units and the propagation of data can be distinguished into [2, 3]

- Feed-forward networks, where the data flows from input to output units is strictly feed-forward. The data processing can extend over multiple layers of units, but no feedback connections are present, that is, connections extending from outputs of units in the same layer or previous layers.

- Recurrent networks that do contain feedback connections. Contrary to feed-forward networks, the dynamical properties of the network are important. In some cases, the activation values of the units undergo a relaxation process such that the network will evolve to a stable state in which these activations do not change anymore. In other applications, the change of the activation values of the output neurons is significant, such that the dynamical constitutes the output of the network.

4.3 Neural Dynamic Modeling

The parsimonious universal approximation property of neural networks can advantageously be exploited for dynamic modelisation of different processes. Two types of modelisation can be usually distinguished [7, 9]:

- Knowledge models, which the mathematical expression, including small number of adjustable parameters, results from (physical, chemical, etc.) analysis of the process.
- Black Box models, which are determined only from measurements made of the process, without any external knowledge

4.4 Feed-Forward Neural Model of the Flexible Process

The feed-forward neural model is defined by the input $\tau(k)$, the torque applied to the hub motor, its previous values, and the previous values of the output $\theta(k)$, the hub angle of the process. The neural model can determined, after training, the future output of the process $\hat{\theta}(k)$. The structure of this model is given by the figure 3.

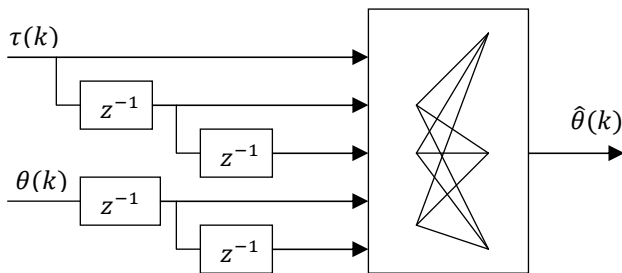


Fig. 3. Feed-forward neural model

The flexible process is excited with a single-switch bang-bang signal of amplitude 0.2Nm input torque, applied at the hub of the manipulator. Figure 4 shows the input signal. A bang-bang torque has a positive (acceleration) and negative (deceleration) period allowing the manipulator to, initially, accelerate, then decelerate and eventually stop at a target location.

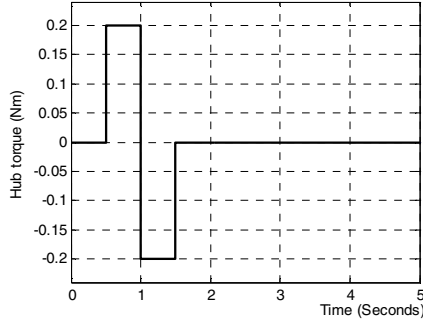


Fig. 4. A bang-bang input torque

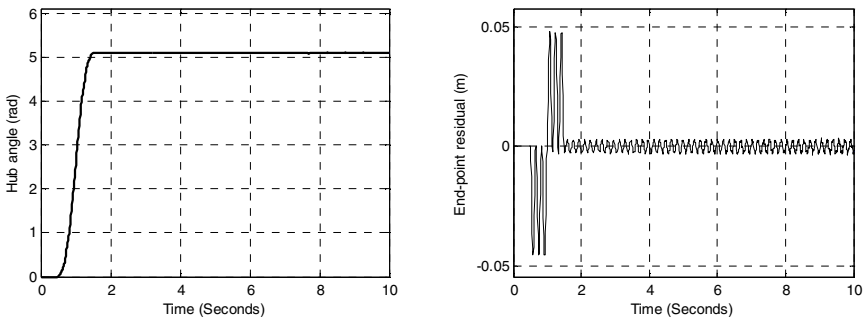


Fig. 5. Hub angle and end-point residual responses

The feed-forward neural network is composed of three layers with a nine neurons hidden layer. It is trained with the back-propagation method. The simulation results of the process responses are given in figure 6. To compare between neural and theoretical models responses, we can notice a good learning of the flexible process dynamics behavior with a hub angle error less than 10^{-3} rad and an end-point residual error less than 1mm.

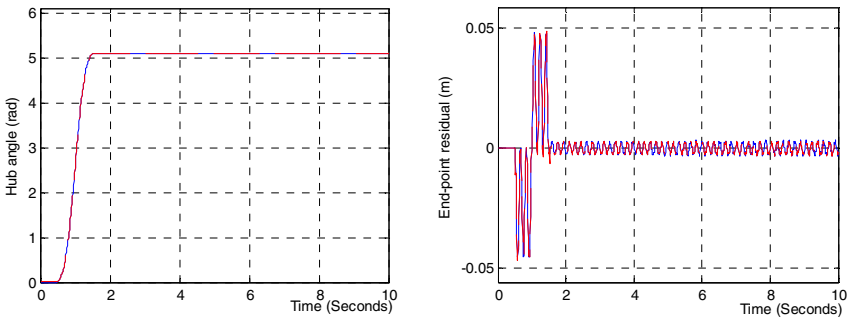


Fig. 6. Hub angle and end-point residual responses with neural and mathematical models

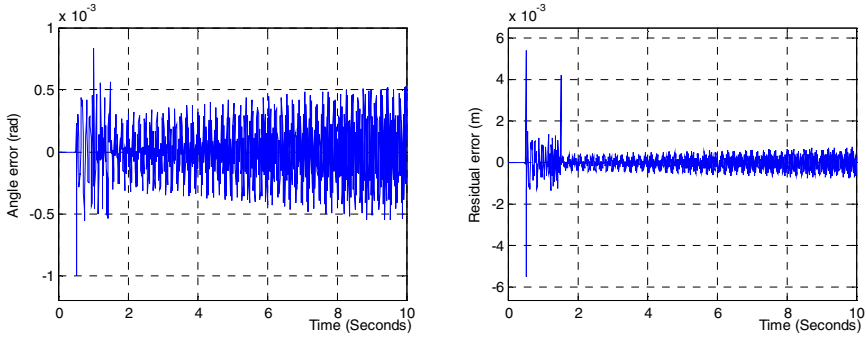


Fig. 7. Hub angle and end-point residual errors

4.5 Recurrent Neural Model for the Flexible Process

The recurrent model is developed in terms of torque input, its previous values, and the previous values of the neural model output. The figure 8 shows the simplified layout of the recurrent model.

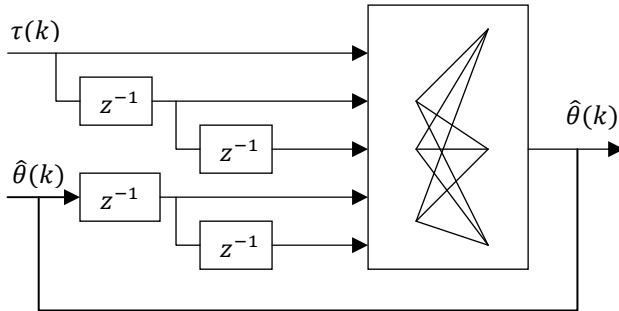


Fig. 8. Recurrent neural network model

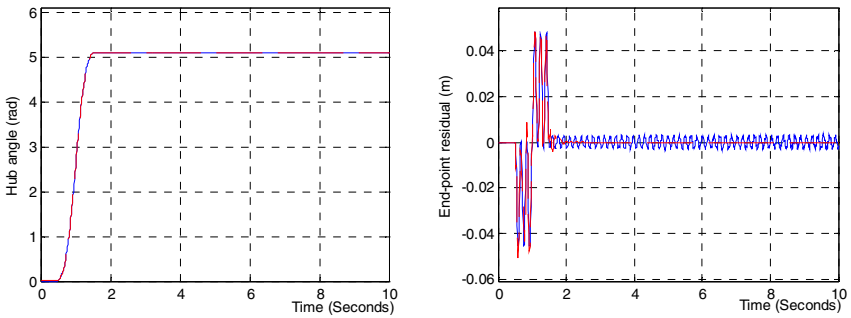


Fig. 9. Hub angle and end-point residual responses

In our case, the recurrent neural model is composed of three layers of neurons; the hidden layer comprised 13 neurons. The learning of the neural network with the back-propagation method can give us the process responses in figures 9 and 10. The error value can be noticed around 0.005rad for the hub angle and 3mm for the end-point residual response.

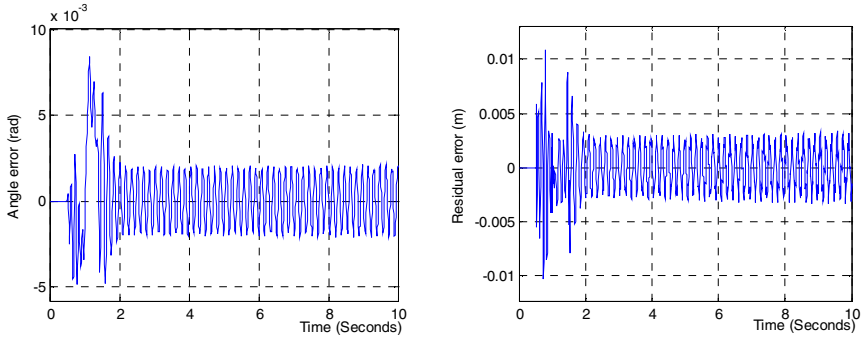


Fig. 10. Hub angle and end-point residual error responses

5 Conclusion

This paper presents a neural modeling of a flexible robot system with feed-forward and recurrent networks methodologies. Dynamic mathematical model of a flexible single-link manipulator robot is derived from Lagrange equations and finite element method. The flexible process is excited with a single-switch bang-bang input torque to consider dynamic behavior and its physical parameters. The learning ability of neural networks allows giving implicit models describing complex dynamic behavior of flexible robots. A feed-forward neural network is first trained with back-propagation algorithm to learn computed dynamic model. Next, a recurrent network is developed to form an independent black box model. A set of simulation results showing behavior of mathematical and neural models is presented. A good learning is revealed in the hub angle and end-point residual responses.

References

1. Abe, S.: Neural Networks and Fuzzy Systems, Theory and Applications. Kluwer Academic Publishers, USA (1997)
2. Hecht-Nielsen, R.: Neurocomputing. Addison-Wesley, Reading (1989)
3. Hertz, A., Krogh, A.S., Palmer, R.G.: Introduction to the Theory of Neural Computation. Addison-Wesley, Redwood City (1991)
4. Narendra, K.W., Parthasarathy, K.: Identification and Control of Dynamical Systems using Neural Networks. IEEE Transactions on Neural Networks, 4-27 (1990)
5. Omidvar, O., Elliott, D.L.: Neural Systems for Control. Academic Press (1997)

6. Usoro, P.B., Nadira, R., Mahil, S.S.: A Finite Element/Lagrange Approach to Modeling Lightweight Flexible Manipulators. *Transactions of the ASME* 108, 198–205 (1986)
7. Simon, H.: *Neural Networks*. Macmillan College Publishing (1994)
8. Zurada, J.: *Introduction to Artificial Neural System*. West Publishing Company (1992)
9. Fu, L.: *Neural Network in Computer Intelligence*. McGraw Hill Book Company (1994)
10. Bickford, W.: *Mechanics of Solids*. Richard D. Irwin, Inc. (1992)
11. Kuo, C.F., Kuo, C.Y.: Modeling and simulation of nonlinear dynamics in a flexible robot arm. *Active Control of Noise and Vibration ASM* 38, 149–156 (1992)
12. Meirovitch, L.: *Elements of Vibration Analysis*, International Student Edition, 4th printing. McGraw-Hill International Book Company (1982)
13. Wang, F.Y., Gao, Y.: *Advanced Studies of Flexible Robotic Manipulators. Modeling, Design, Control and Applications*. Series in Intelligent Control and Intelligent Automation, vol. 4. World Scientific Publishing (2003)
14. Roy Pota, H.: Finite-element/Lagrange Modeling and Control of a Flexible Robot Arm. In: 11th IFAC World Congress, Tallinn, Estonia, USSR, vol. 9, pp. 239–243 (1990)
15. Hashemi, S.M., Borneman, S.R., Alighanbari, H.: Vibration of Cracked Composite Beams: A Dynamic Finite Element. *International Review of Aerospace Engineering (IREASE)* 1(1), 110–121 (2008)
16. Mansour, T., Konno, A., Uchiyama, M.: MPID Control Tuning for a Flexible Manipulator Using a Neural Network. *Journal of Robotics and Mechatronics* 22(1), 82–90 (2010)
17. Abe, A.: Trajectory planning for flexible Cartesian robot manipulator by using artificial neural network: numerical simulation and experimental verification. *Robotica* 29(05), 797–804 (2011)
18. Mahamood, R.M., Pedro, J.O.: Hybrid PD-PID with Iterative Learning Control for Two-Link Flexible Manipulator. In: *Proceedings of the World Congress on Engineering and Computer Science*, San Francisco, USA, vol. II (October 2011)