International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 4, June - 2012

ANFIS Implementation for Robotic Arm Manipulator

Gurjeet Singh
Department of ECE, PTU university
ACET College, Amritsar

Dr. Vijay Kumar Banga
Professor, Department of ECE, PTU university
ACET college, Amritsar

ABSTRACT
The inverse kinematics problem in robotics requires the determination of the joint angles for a desired position of the end-effector. The most important problem in robot kinematics and control is, finding the solution of Inverse Kinematics. If the joint structure of the manipulator is more complex the traditional method such as geometric, iterative and algebraic are inadequate. As the complexity of robot increases, obtaining the inverse kinematics is difficult and computationally expensive. For this under constrained and ill-conditioned problem we propose a solution based on structured neural networks that can be trained quickly. In this paper, using the ability of ANFIS (Adaptive Neuro-Fuzzy Inference System) to learn from training data, it is possible to create ANFIS with limited mathematical representation of the system. Computer simulations conducted on 2DOF and 3DOF robot manipulator shows the effectiveness of the approach. Workspace area of 2DOF and 3DOF is also shown .

Keywords
kinematics, Fuzzy control system(FCS),Adaptive Neural networks(ANN), Fuzzy Interface System(FIS).

1. INTRODUCTION
Modern robot manipulators and kinematic mechanisms in general, are typically constructed by connecting different joints together using rigid links. A number of links are attached serially by a set of actuated joints. The kinematics of a robot manipulator describes the relationship between the motion of the joints of a manipulator and resulting motion of the rigid bodies that form the robot. Robot control actions are executed in the joint coordinates while robot motions are specified in the Cartesian coordinates. Conversion of the position and orientation of a robot manipulator end-effector from Cartesian space to joint space, called as inverse kinematics problem, which is of fundamental importance in calculating desired joint angles for robot manipulator design and control.[14] For a manipulator with n degree of freedom, at any instant of time joint variables is denoted by $\theta i = \theta(t)$, i = 1, 2, 3,..., n and position variables $x_j = x(t)$, j = 1, 2, 3,...,m. The relations between the end-effector position x(t) and joint angle θ(t) can be represented by forward kinematic equation.[12]

$$x(t) = f(\theta(t)) \qquad (1)$$

Where f(θ(t)) is a nonlinear, continuous and differentiable function. On the other hand, with the given desired end effector position, the problem of finding the values of the joint variables is inverse kinematics, which can be solved by.

$$\theta(t) = f'(x(t)) \qquad (2)$$

Solution of (2) is not unique due to nonlinear, uncertain and time varying nature of the governing equations. Different methods used for solving inverse kinematics The three main methods for solving inverse kinematics, namely, algebraic, geometric and iterative are described below.[14]

Algebraic: To solve for inverse kinematics algebraically, it is necessary to solve equations for θ_1 ,θ_2 ,,,,θ_n for N degrees of freedom. The problem can be formulated as follows: given the end-effector position. where the right-hand side describes the required position and orientation of the end-effector. The problem comes down to solving N equations for N unknowns. This method does not guarantee a closed form solution for a given manipulator. [5]

Geometric: As opposed to the algebraic method, a closed form solution using the geometry of the manipulator is derived. This involves projecting of the link coordinate frame on the X_{i-1} and Y_{i-1} frame. This method can be applied to any manipulator with known geometry. The limitation of this method is that the closed-form solution for the first three joints of the manipulator must exist geometrically. Apart from that, the closed-form solution for one class of manipulators cannot be used in other manipulators of a different geometry.[5]

Iterative: This method solves inverse kinematics by iteratively solving for the joint angles. This method converges to only one solution as opposed to the two methods presented by Korein and Balder[3] . There are three components that constitute iterative methods, namely, the Jacobian, pseudo-inverse and minimization methods. If the joints of the manipulator are more complex, the inverse kinematics

Gurjeet Singh is working as a lecturer and pursuing M.Tech (Electronics and Communication Engineering) in Department of Electronics & Communication Engineering, Amritsar College of Engineering and Technology, Amritsar, Punjab, India(mann_567@yahoo.com).

Dr. Vijay Kumar Banga is working as professor and Head of the Department of Electronics & Communication Engineering, Amritsar College of Engineering and Technology, Amritsar, Punjab, India (e-mail: vijaykumar.banga@gmail.com.

solution by using these traditional methods is a time consuming. In other words, for a more generalized m-degrees of freedom manipulator, traditional methods will become prohibitive due to the high complexity of mathematical structure of the formulation[5].

Utilization of Neural network (NN) and Fuzzy logic for solving the inverse kinematics is much reported.Li-Xin Wei[11] , and Rasit Koker et al[12], proposed neural network based inverse kinematics solution of a robotic manipulator. In this paper, neuro-fuzzy systems which provide fuzzy systems with automatic tuning using Neural network is used to solve the inverse kinematics problem**.** [2]The most common neural networks used to solve the problem of inverse kinematics are error backpropagation, hybrid,, Kohonen networks.[7] The error-backpropagation algorithm takes a very long time for forward training. We have proposed a variant of the error-backpropagation algorithm to solve this problem. This new approach has the advantage of accuracy over the error-backpropagation algorithm. The paper is organized as follows, in section 2, the structure of ANFIS used is presented. Section 3 describes results and discussion. Section 4 workspace area is discussed, Section 5 ends with conclusion..

2. ARTIFICAL NEURAL NETWORK

Artificial neural network (ANN) is a parallel-distributed information processing system. This system is composed of operators interconnected via one-way signal flow channels. ANN stores the samples with a distributed coding, thus forming a trainable nonlinear system. It includes hidden layer(s) between the inputs and outputs. This section introduces the basics of ANFIS network architecture and its hybrid learning rule. Adaptive Neuro-Fuzzy Inference System is a feedforward adaptive neural network which implies a fuzzy inference system through its structure and neurons.[13] Jang was one of the first to introduce ANFIS[11].An adaptive neuro -Fuzzy Inference System (ANFI S) is a cross between an artificial neural network and a fuzzy inference system (FIS). An artificial neural network is designed to mimic the characteristics of the human brain and consists of a collection of artificial neurons. An adaptive network is a multi-layer feed-forward network in which each node (neuron) performs a particular function on incoming signals. The form of the node functions may vary from node to node. In an adaptive network, there are two types of nodes: [10]

adaptive and fixed. The function and the grouping of the neurons are dependent on the overall function of the network. Based on the ability of an ANFIS to learn from training data, it is possible to create an ANFIS structure from an extremely limited mathematical representation of the system. In sequel, the ANFIS architecture can identify the near-optimal membership functions of FLC for achieving desired input-output mappings. The network applies a combination of the least squares method and the back propagation gradient descent method for training FIS membership function parameters to emulate a given training data set. The system converges when the training and checking errors are within an acceptable bound.The ANFIS system generated by the fuzzy toolbox available in MATLAB allows for the generation of a standard Sugeno style fuzzy inference system or a fuzzy inference system based on sub-clustering of the data [4]. Figure.1 shows a simple two-input ANFIS architecture.

.

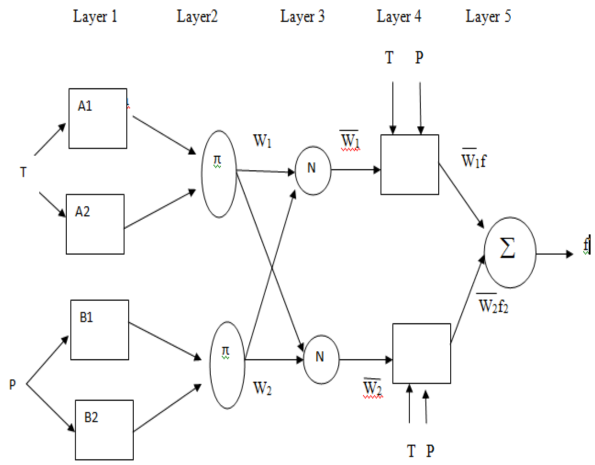


Figure1: ANFIS Architecture for a Two-Input System

The above ANFIS architecture is based on a Sugeno fuzzy inference system. The sugeno FIS is similar to Mamadani format except the output membership are singleton spikes rather than a distributed fuzzy set. Using singleton output simplifies the de-fuzzification step.[8] The ANFIS network shown in Figure 1 is composed of five layers. Each node in the first layer is a square (adaptive) node with a node function computed as follows:[16]

$$Oi^1 = u_{Ai}(T) \qquad (3)$$

Where T is the first input vector and u_{Ai} is the membership function for that particular input. Layer two only consists of circle (fixed) nodes.The output of each node is the product of the two membership functions:

$$Oi^2 = W_1 = u_{Ai}(T)\ \ u_{Bi}(P) \qquad (4)$$

Layer three only contains circle (fixed) nodes with their normalized firing strengths in the following form:

$$O_3^i = \overline{Wi} = [\ W_1 / (W_1 + W_2)] \text{ where } i = 1,2 \qquad (5)$$

The fourth layer (square nodes) is computed from the product of consequentv parameter set and the output of the third layer as:

$$O_4^i = \overline{W}i(p_i T + qP + r) \qquad (6)$$

Finally, layer five, consisting of circle nodes is the sum of all incoming signals.

$$O_{5,1} = \sum \overline{Wi}\ f_i \qquad (7)$$

The above adaptive architecture is functionally equivalent to Sugeno fuzzy model. This ANFIS structure can update its parameters according to the gradient descent procedure. Other ANFIS structures corresponding to different types of FIS and defuzzification mechanism h are also proposed by the researcher [16].However, throughout this chapter, we shall utilize the above first order Sugeno fuzzy model for the microbot application due to its transparency and efficiency.[15]

3. SIMULATION AND RESULT

Figure 2 shows the two degree of freedom (DOF) planar manipulator arm which is simulated in this work.

Two Degree of Freedom planar manipulator

Considering length of first arm l_1 = 10 and length of second arm l_2 = 7 along with joint angle constraints the forward kinematic equations are,

$$x = l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) \qquad (8)$$

$$y = l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2) \qquad (9)$$

and the inverse kinematics equations are ,

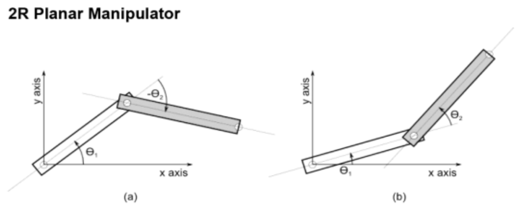


Figure 2: Two DOF planar manipulator Arm

$$\theta_2 = a\tan 2\,(\sin\theta_2 \cos\theta_2) \qquad (10)$$

$$= a\tan 2\,(\pm\sqrt{\mathbf{1-\cos^2\theta_2}}.\ \cos\theta_2$$

$$= \operatorname{atan2}\left(\pm\sqrt{\mathbf{1-\left(\frac{x^2+y^2-l_1{}^2-l_2{}^2}{2l_1\ l_2}\right)}},\ \frac{\mathbf{x^2+y^2-l_1{}^2-l_2{}^2}}{\mathbf{2l_1\ l_2}}\right)$$

$$\theta_1 = \operatorname{atan}(y,x)-\operatorname{atan2}(k_2, k_1) \qquad (11)$$

In figuren.3 the ANFIS procedure is shown, in the first step initializing the fuzzy system using genfis command, in the second stem learning process start and the number of epochs is set. in the third step learning process start by using anfis command and last in the fourth step validation occur with independent data.

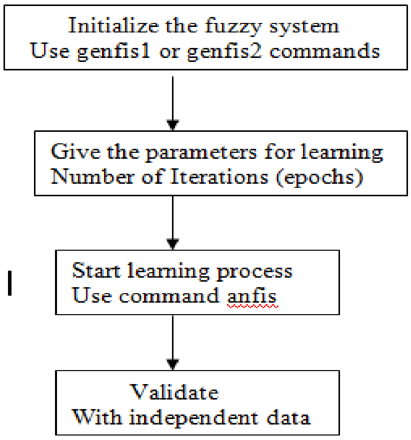
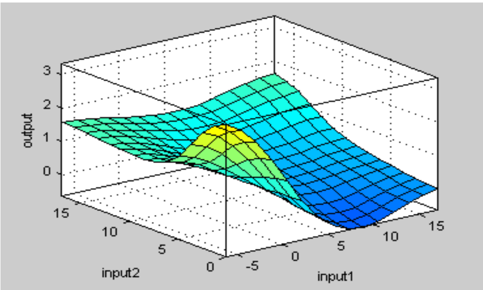


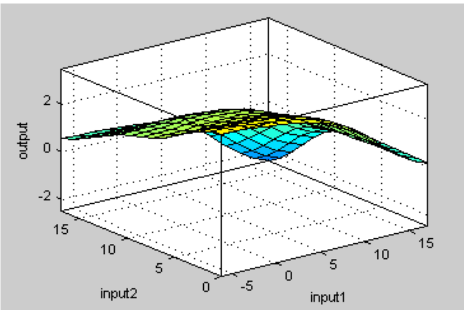
Figure 3: ANFIS Procedure

$0 < \theta_1 < \frac{\pi}{2}$, $0 < \theta_2 < \pi$

The x and y coordinates of the arm are calculated for two joints using forward kinematics.



(a)



(b)

Figure 4: Training data of (a)θ_1 and (b)θ_2.

The coordinates and the angles are used as training data to train ANFIS network with Gaussian membership function with hybrid learning algorithm Figure 4 shows the training data of two ANFIS networks for two joint angles. The coordinates act as input to the ANFIS and the angles act as the output. The learning algorithm" teaches" the ANFIS to map the co-ordinates to the angles through a process called training. In the training phase, the membership functions and the weights will be adjusted such that the required minimum error is satisfied or if the number of epochs reached. At the end of training, the trained ANFIS network would have learned the input output map and it is tested with the deduced inverse kinematics. Figure 5 shows the difference in theta deduced analytically and the data predicted with ANFIS.

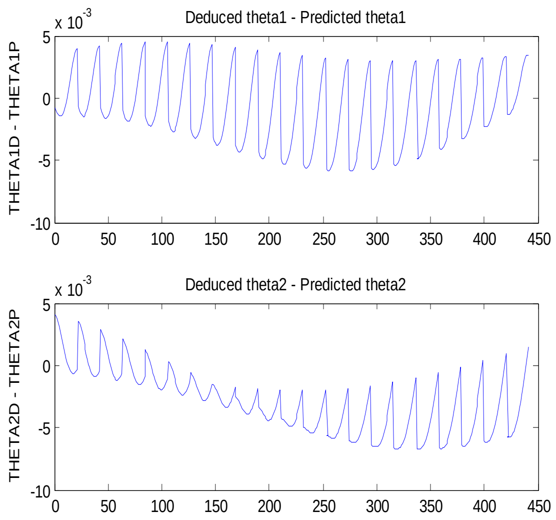


Figure 5: Difference in theta deduced and the data predicted with ANFIS trained

Three Degree of Freedom planar manipulator

For a 3 DOF planar redundant manipulator, the forward kinematic equations are,

$$x = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) \quad -(12)$$

$$y = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) \quad -(13)$$

The inverse kinematics equations are

$$\boxed{\phi = \theta_1 + \theta_2 + \theta_3} \quad -(14)$$

By solving we get

$$\cos(\alpha - \gamma) = \frac{-R}{\sqrt{P^2 + Q^2}}$$

The above equation has two solutions

$$\alpha = \gamma + \sigma \cos^{-1}\left(\frac{-R}{\sqrt{P^2+Q^2}}\right), \sigma = \pm 1$$

Where

P$= -2l_1x'$, Q$= -2l_1y'$, $\alpha = \theta_1$

R$= x'^2 + y'^2 + l_1^2 - l_2^2$

Now finding θ_1, θ_2 and θ_3 as;

$$\theta_1 = \gamma + \sigma \cos^{-1}\left(\frac{-(x'^2 + y'^2 + l_1^2 - l_2^2)}{2l_1\sqrt{x'^2 + y'^2}}\right) \quad (15)$$

$$\theta_2 = \text{atan2}\left(\frac{y' - l_1\sin\theta_1}{l_2}, \frac{x' - l_1\cos\theta_1}{l_2}\right) - \theta_1 \quad (16)$$

Where

$x' = x - l_3 \cos\phi$,

$y' = y - l_3 \sin\phi$.

And θ_3 is given by

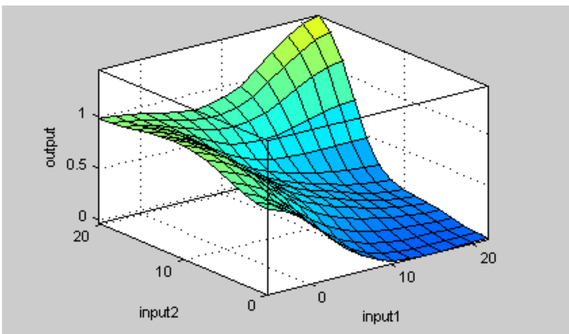
$$\theta_3 = \phi - \theta_1 - \theta_2 \quad (17)$$

For simulation, the length for three links are l1 = 10, l2 = 7 and l3 = 5 with joint angle constraints

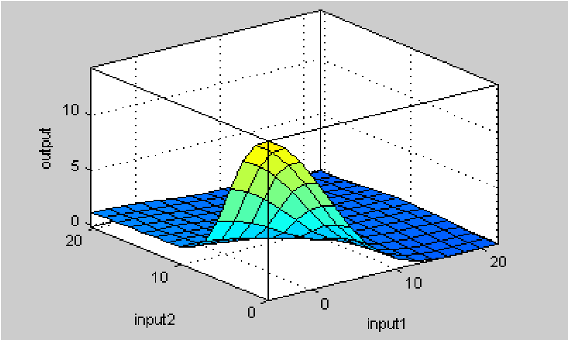
$0 < \theta_1 < \pi/3$, $0 < \theta_2 < \pi/2$, $0 < \theta_3 < \pi$

the same procedure is repeated. Figure 6 shows the training data of three

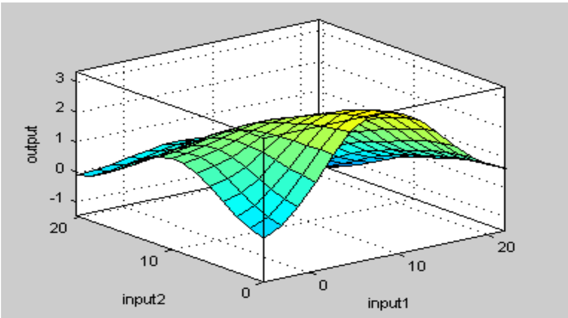
ANFIS networks for three joint angles. Figure 7 shows the difference in theta deduced analytically and the data predicted with ANFIS.



(a)



(b)



(c)

Figure 6: Training data of (a) θ_1, (b) θ_2 and (c) θ_3

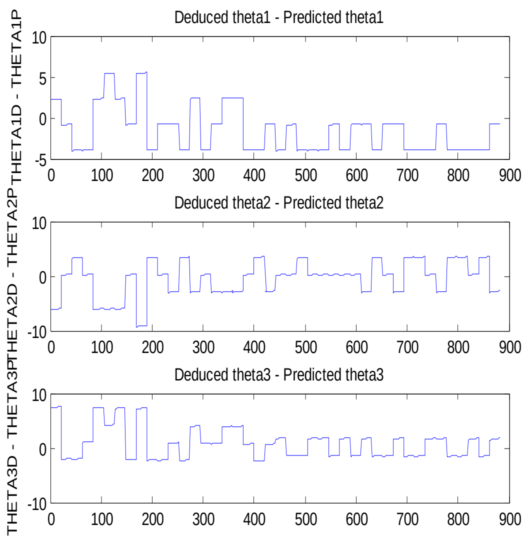
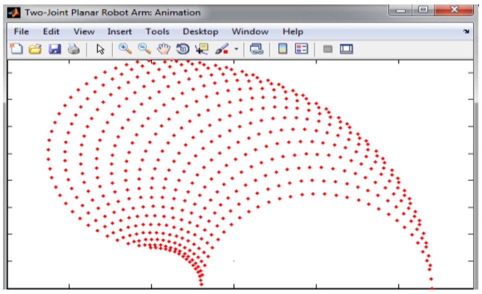


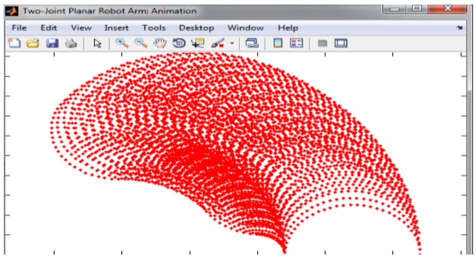
Figure 7: Difference in theta deduced and the data predicted with ANFIS trained

4. WORKSPACE AREA

The workspace area is the area where the endeffector can move easily & the resolution is maximum in this area . The work space area of 2 and 3 arm Robot with help of GUI (Graphic User Interface) is shown below .The red star show the resolution



(a)



(b)

Figure 7: Workspace area of (a) Two Arm (b) Three Arm

5. CONCLUSION

The workspace area of 2 -3 DOF is defined where the resolution is maximum .The difference in theta deduced and the data predicted with ANFIS trained for 2DOF and 3DOF planar manipulator clearly depicts that the proposed method results in an acceptable error. Trained ANFIS can be utilized to provide fast and acceptable solutions of the inverse kinematics thereby making ANFIS as an alternate approach to map the inverse kinematic solutions. Other techniques like input selection and alternate ways to model the problem may be explored for reducing the error further.

6. REFERENCES

[1] Anon,1992, SAE Mathematical Formulae and Statistical Tables Book, Secondary Education Authority.

[2] Yang Ming Lu, Lu Guizhang, Li Jiangeng,2001, “An Inverse Kinematics Solution for Manipulators,” Proceedings of IEEE,Vol.4, 400-404.

[3] [3] N. Abulafya,1995, Neural networks for system identification and control, MSc thesis, University of London.

[4] P.C.Y. Chen, J.K. Mills, K.C. Smith,1996, Performance improvement of robot continuous-path operation through

iterative learning using neural networks, Machine Learn. J. 23, 191–220.

[5] Sreenivas Tejomurtula,1998, Inverse kinematics in robotics using neural networks, M.S. Thesis,Louisiana State University, December.

[6] DavidW. Howard and Ali Zilouchian,1998 “Application of Fuzzy Logic for the Solution of Inverse Kinematics and Hierarchical Controls of Robotic Manipulators,” Journal of Intelligent and Robotic Systems, 23, 217-247.

[7] Sreenivas Tejomurtula,1999"Inverse kinematics in robotics using neural networks Information "Sciences 116 ,147-164.

[8] Sreenivas Tejomurtula, Subhash Kak,1999 “Inverse kinematics in robotics using neural networks,” Information Sciences, 116,147-164.

[9] C.M. Clark, J.K. Mills,2000 Robotic system sensitivity to neural network learning rate: theory, simulation, and experiments, Int. J. Robotic.Res. 19 (10) 955–968.

[10] Tang Y. and Velez-Diaz D,2003," Robust fuzzy control of mechanical systems". IEEE Transactions on Fuzzy Systems 11(3), 411- 418,.

[11] Li-Xin Wei, Hong-Rui Wang, Ying Li,2003, “ A new solution for inverse kinematics of manipulator based on neural network,”Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xian, 3(5), 1201-1203,November .

[12] Rasit Koker, Cemil Oz, Tark Cakar, Huseyin Ekiz,2004 “A study of neural network based inverse kinematics solution for a three-joint robot,” Robotics and Autonomous Systems, 49, 227-234.

[13] Tiberiu Vesselenyi, Simona Dzitac, Ioan Dzitac, Misu-Jan Manolescu,2007 “Fuzzy and Neural Controllers for a PneumaticActuator,” International Journal of Computers, Communications and Control, Vol. II, No. 4, pp. 375-387.

[14] Srinivasan Alavandar,2008"Inverse Kinematics solution of 3DOF Planar Robot using ANFIS", Int. J. of Computers, Communications & control, Vol.III, 150-155.

[15] S.G. Patil ,2011,"Neuro-fuzzy based approach for wave transmission prediction of horizontally interlaced multilayer moored floating pipe breakwater "Ocean Engineering 38, 186–196.

Rodrigo Pérez-Rodríguez,2012"Inverse kinematics of a 6 DoF human upper limb using ANFIS and ANN for anticipatory actuation in ADL-based physical Neurorehabilitation"Expert Systems with Applications 39 , 9612–9622.

Gurjeet Singh is working as a lecturer in Department of Electronics & Communication Engineering in Amritsar College of Engineering and Technology, Amritsar, Punjab, India. He did his B.Tech (Electronics and Communication Engineering) from Punjab Technical University, Jalandhar. India. and pursuing M.Tech (Electronics and Communication Engineering) from Amritsar College of Engineering and Technology, Amritsar, Punjab, India.

Vijay Kumar Banga is working as professor and Head of the Department of Electronics & Communication Engineering, Amritsar College of Engineering and Technology, Amritsar, Punjab, India. He obtained his B. E (Electronics and Instrumentation Engineering) from Punjabi University, Patiala, Punjab, India, M.Tech (Electronics and Instrumentation) from Panjab University , Chandigarh, India and Ph.D. in Electronics (Artificial Intelligence) form Thapar University, Patiala., India. Presently, he has 13 years of research and UG & PG teaching experience. He has 35 research papers to his credit in various international journals and conferences. He is member of Board of Studies in Electronics and Communication Engineering of Punjab Technical University, Jalandhar, Punjab, India. His areas of research and interest include Robotics, Artificial Intelligence, automated control systems and Genetic algorithms.