OPTIMIZATION OF INVERSE KINEMATICS OF ROBOTIC ARM USING ANFIS

1. AMBUJA SINGH, 2. DR. MANOJ SONI

1(M.TECH STUDENT, R&A, DEPARTMENT OF MAE, IGDTUW, DELHI, INDIA)

2(ASSOCIATE PROFESSOR, DEPARTMENT OF MAE, IGDTUW, DELHI, INDIA)

ABSTRACT

In this paper, optimization of inverse kinematics of 3r robotic arm using ANFIS. This technique provides the determined joint angles for end effector's desired position. For controlling of robots get more complex when manipulator structure is complex. That time the traditional methods like geometrically, algebraic are not feasible and very much time consuming. In order to find the kinematics for the robots is very important task and computing inverse kinematics (IK) is difficult to obtained.so, there has been a solution to trained data very quickly by artificial neural network (ANN). In this paper we have been use d a hybrid solution i.e. ANFIS that learn from training data. This data will be used for a 3R robotic arm by interfacing Matlab to Arduino. This paper has been used the simulation part for the manipulator.

KEYWORDS:- *Robotic Manipulator, ANFIS, Forward Kinematics, Inverse Kinematics.*

---

1. INTRODUCTION

The robot manipulator in a modern era and mechanism of kinematics, are made by joint connection serially and using the rigid links. Kinematics is very important term in robotics that describes the relationship between motion of joint angles and resulting motion due to manipulator that combines for the robot. In any serial or parallel robotic manipulator, there are two actions one is control action and another one is robot motion. Both are executed by different parameter like control is described by joint coordinates and motion of robot is described by their specified Cartesian coordinates. Here the kinematic analysis is very important term which specified study of motion. In which there has been forward and inverse kinematics universally. In forward kinematics, there has been given the joint angles and according to that analysis correct position, it is base to end effector analysis.

In inverse kinematics (IK), finding the joint angles of manipulator with the help of position of manipulator. There is number of techniques or analysis to find the joint angles.

In forward kinematics;

$$x(t) = f\{\Theta(t)\}$$

Where, $x(t)$ -> end effector position

$\Theta(t)$ -> joint angles.

This is the unique solution and has been found using geometrically or algebraically.

In inverse kinematics;

$$\Theta(t) = f'\{x(t)\}$$

This has not been made a unique solution due to non-linear, uncertain and time variant. This is complex in nature, if there has been used algebraic method then the challenge is solving the N equation for N- unknowns. This is not appropriate for close solution for manipulator.

If we use the geometric technique then too have the challenge because this can be solved only for known geometry, in case changing the manipulator again all the calculations have to be made.

So, here the technique has been used which is ANFIS. For using neuro-fuzzy solution to make it more accurate and easy to understand. Li-Xin Wei [3] and Rasit Koker et al [4], proposed NN (neural network) based IK (inverse kinematics) solution for any serial robotic arm. In which fuzzy system automatically connect using NN (neural network) for solving IK (inverse kinematics).

There has been solved IK using NN by error back propagation, hybrid, kohonen network [5]. Here, has been used the error back propagation method to solve IK and ANFIS has been used to make more accurate over the back propagation algorithm.

This paper has been organised as overview of robotic manipulator, kinematics analysis, ANFIS structure, and simulation of robotic arm using ANFIS, their graph and results and then conclusion.

2. ROBOTIC MANIPULATOR

The robot named “3R planar robot” has been taken with 3 degree of freedom and end effector for pick and place, which has workspace of 13cm. and load capacity is 50 gm.

Here is architecture of 3R robotic arm which has been designed by ArtCAM pro 9.021 and cutting all the clamper which has been used in manipulator by Mac3 CNC cutting machine. Clampers are made by aluminium sheet which has 1mm. thickness. Servo motors are used for movement of the robotic manipulator. Arduino is used to control the robotic manipulator which is programmed by Matlab.

The work space area of 2dof is defined where the resolution is maximum. The difference in theta deduced and the data predicted with ANFIS trained for 2 dof manipulator clearly depicts that the proposed method results in an acceptable solution of the inverse kinematic solution of the inverse kinematics thereby making ANFIS as an alternate approach to map the inverse kinematic solution. Other techniques like input selection and alternate ways to model the problem may be explored for reducing the error further.

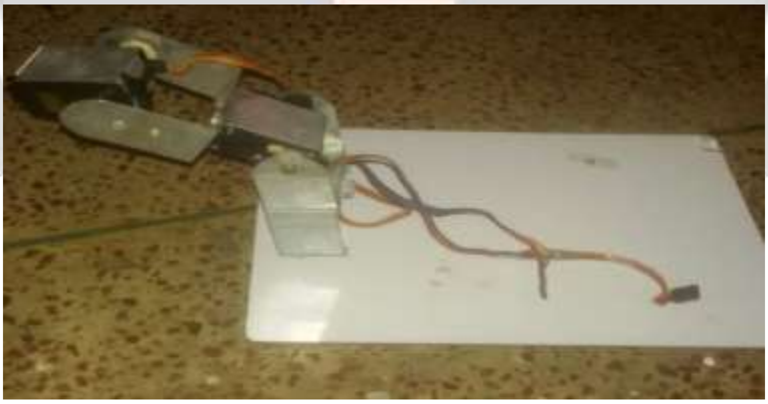

Fig 1: Model of 3R planar manipulator

3. KINEMATICS

Given a set of joint angles, the forward kinematics will always produce an unique solution giving the robot global position and orientation. On the other hand, there may be no solution to the inverse kinematics problem. The reasons include:

The given global position of the arm may be beyond the robot work space.

The given global orientation of the gripper may not be possible given that the gripper frame must be a right hand frame.

For the inverse kinematics problem, there may also exist multiple solutions, i.e. the solution may not be unique. Some solutions may not be feasible due to obstacles in the workspace.

3.1. DIRECT KINEMATICS

The value of end effector position of robot has been found in the solution of forward kinematics. There has been given the all joint angle and link length values. By using method of homogeneous transformation matrices to solve the problem and there has been found final position by using Denavit-Hartenberg's representation geometrically and also relate this method for positioning of end of each link compared with previous or reference system [11].

An information about the position and orientation of reference system has been contained in 4*4 homogeneous matrix which has been attached the *ith* link of robotic manipulator which is compared with previous *i-1* link of manipulator. Matrix ${}^{0}_{n}A$ with n degree of freedom has been represented the end position of manipulator i.e. denoted by 'T'. For 3R robotic manipulator.

For n degree of freedom;

T- Transformation matrix

T = $\prod_{i=1}^{n} {}^{i-1}_{i}A$

T = ${}^{0}_{n}A = {}^{0}_{1}A \, {}^{1}_{2}A \ldots {}^{n-1}_{n}A$

Where, ${}^{0}_{n}A$ – Homogeneous matrix

For 3R robot;

T= T = ${}^{0}_{3}A = {}^{0}_{1}A \, {}^{1}_{2}A \, {}^{2}_{3}A$

By D-H parameters i-1Ai has been defined geometrically each link and place the co-ordinate system.

Parameters are [3];

θ_i = Rotation around z_{i-1}axis

d_i = Translation along z_{i-1}axis

a_i = Translation along $x_i$axis

α_i = Rotation around $x_i$axis

Table 1: D-H Parameters

s.no.	$\theta_i(joint\ angle)$	$d_i(joint\ offset)$	$a_i(link\ length)$	$\alpha_i(twist\ angle)$
1.	θ_1	d_1	0	90°
2.	θ_2	0	a_2	0
3.	θ_3	0	a_3	0

$$ {}^{0}_{n}A = \begin{matrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ \theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{matrix} $$

From this value, solving for ${}^{0}_{3}A$ by using each matrices of ${}^{0}_{1}A$, ${}^{1}_{2}A$, ${}^{2}_{3}A$. We have

$$ T = \begin{matrix} n & o & a & P \\ 0 & 0 & 0 & 1 \end{matrix} $$

Where n, o, a is co-ordinate system for orientation and P is for position. So, it has been used only P vector to describe the position of manipulator with the help of joint angles theta1, theta2 and theta3.

$$P = \begin{matrix} P_x \\ P_y \\ P_z \end{matrix}$$

And we get;

$$P_x = a_2 C\theta_1 C\theta_2 + a_3 C\theta_1 C\theta_2 C\theta_3 - a_3 C\theta_1 S\theta_2 S\theta_3$$

$$P_y = a_2 S\theta_1 C\theta_2 + a_3 S\theta_1 C\theta_2 C\theta_3 - a_3 S\theta_1 S\theta_2 S\theta_3$$

$$P_z = a_2 S\theta_2 + a_3 C\theta_3 + a_3 C\theta_3 S\theta_2 + d_1$$

3.2. INVERSE KINEMATICS

In this analysis, the joint angles have been solved by the position a point in co-ordinate system. This means theta1, theta2 and theta3 are based on the value of Px, Py and Pz.

We have;

$$T = {}^0_nA = {}^0_1A\, {}^1_2A\, {}^2_3A$$

$$\frac{1}{{}^0_1A}\frac{1}{{}^1_2A}T = {}^2_3A$$

Solving above equation and matching 4th column and we get

For θ_1,

$$\frac{\sin\theta_1}{\cos\theta_1} = \frac{P_y}{P_x} = \tan\theta_1$$

$$\theta_1 = \tan^{-1}\frac{P_y}{P_x}$$

For θ_2, We have:-

$a = P_x C\theta_1 + P_y S\theta_1$

$b = P_z - d_1$

$c = aC\theta_2 + bS\theta_2$

$$\cos\theta_2 = \frac{1-u^2}{1+u^2}$$

$$\sin\theta_2 = \frac{2u}{1+u^2}$$

Then,

$$\tan\theta_2 = \frac{2u}{1-u^2}$$

$\theta_2 = \tan^{-1}\frac{2u}{1-u^2}$ Where, $u = \tan\frac{\theta_2}{2}$

For θ_3,

$$\tan\theta_3 = \frac{-aS\theta_2 + bC\theta_2}{a\theta_2 + bS\theta_2 - a_2}$$

$$\theta_3 = \tan^{-1}\frac{-aS\theta_2 + bC\theta_2}{a\theta_2 + bS\theta_2 - a_2}$$

4. MODEL SIMULATION SOFTWARE

4.1. ANFIS (ADAPTIVE NEURO FUZZY INFERENCE SYSTEM)

The acronym ANFIS derives its name from adaptive neuro-fuzzy inference system. Using a given input/output data set, the tool box function anfis constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a back propagation algorithm alone or in combination with a least squares type of method (i.e. steepest decent method). This adjustment allows fuzzy system to learn from the data they are modelling.

- This method is used as a teaching method for Sugeno-type fuzzy system.
- Usually this number and type of fuzzy system membership function are defined by user when applying ANFIS.
- It is hybrid method, consisting two parts:
    1. Gradient method is used to calculation of input membership function parameters.
    2. Least Square method is used to calculation of output function parameter.
- In fuzzy control tool box a useful command called ANFIS exist.
- This provides an optimization scheme to find the parameter in the fuzzy system that best fit the data.
- It is explained in the tool box manual that since most optimization algorithm require computation of the gradient, this is done with the neural network.
- In this technique there is super imposition of FIS (fuzzy inference system) over a NN (neural network).

4.2. ANFIS STRUCTURE

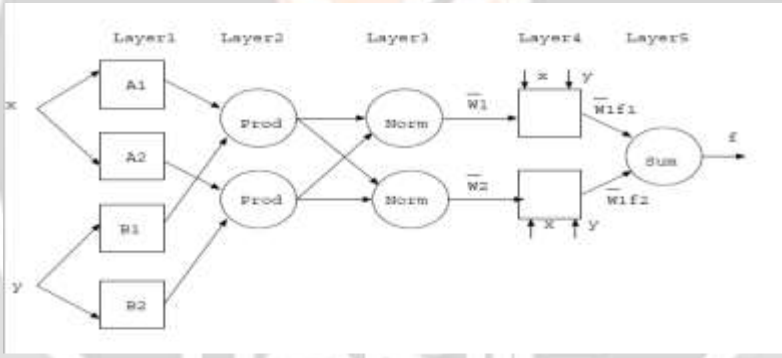

Fig 2: ANFIS structure for two input

This structure is based on Sugeno fuzzy inference system. ANFIS structure consists of five layers. In which different operations has been performed.

First layer is square adaptive node computed by a node function;

$O_i^1 = u_{Ai}$(X)

Where X is first input vector and u_{Ai} is a membership function for that input.

Second layer is fixed nodes which are shown by circle. Product of the two membership functions is the output of each node;

$O_i^2 = W_1 = u_{Ai}$(X)u_{Bi}(Y)

Third layer has normalized firing strength and this is also fixed node and shown by circle. Firing strength is in the form of;

$O_i^3 = W_{i=}$ $[W_1/(W_1 + W_2)]$

Where i= 1, 2.

Fourth layer is product of two consecutive parameter set and shown as square nodes and this the output of third layer as;

$O_i^4 = W_i(p_i X + q_i Y + r)$

And finally the fifth layer is sum of all incoming signals and shown as circle node;

$O_{5,1} = \sum Wi\, f_i$

This method has been utilized first order Sugeno fuzzy model for diff application because of the easy techniques which is transparent and more accurate [6].

5. SIMULATION AND RESULT

In this fig shows the 2 dof manipulator arm which is simulated in this work.

We have taken

Length of first arm L1=8 cm.

Length of second arm L2= 5 cm.

Along with the joint angle constraints the forward kinematic equations are,

x = l1 cos(θ1) + l2cos(θ1+θ2)

y = l1 sin(θ1) + l2sin(θ1+θ2)

And the inverse kinematics equations are;

θ2= a tan2 (sin θ2 cos θ2)

θ1 = atan (y, x)-atan2 (k2, k1)

the anfis procedure is shown, in the first step initializing the fuzzy system using genfis command, in the second step 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.

Initialize the fuzzy system, use genfis1 or genfis2 commands → give the parameters for learning, number of iterations (epochs) → start learning process, use command anfis → validate with independent data.

0 < θ1< 180 degree, 0 < θ2< 180 degree

The coordinate and the angle are used as training data to train ANFIS network with Gaussian membership function with hybrid learning algorithm. The coordinate act as input to the ANFIS and the angles act as the output. The learning algorithm “teaches” the ANFIS to map the coordinate to the angle through a process called training. In the training phase, the membership function 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 input output map and it is tested with the deduced inverse kinematics. Figure shows the difference in theta deduced analytically and the data predicted with ANFIS.

In this fig shows the 2 dof manipulator arm which is simulated in this work.

We have taken

Length of first arm L1=10

Length of second arm L2= 7

Along with the joint angle constraints the forward kinematic equations are, x = l1 cos(θ1) + l2cos(θ1+θ2)

y = l1 sin(θ1) + l2sin(θ1+θ2)

And the inverse kinematics equations are;

θ2= a tan2 (sin θ2 cos θ2)

θ1 = atan (y,x) - atan2(k2, k1)

The anfis procedure is shown, in the first step initializing the fuzzy system using genfis command, in the second step 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.

Initialize the fuzzy system, use genfis1 or genfis2 commands → give the parameters for learning , number of iterations(epochs) → start learning process, use command anfis → validate with independent data.

0 < θ1< 180 degree, 0 < θ2< 180 degree

The coordinate and the angle are used as training data to train ANFIS network with Gaussian membership function with hybrid learning algorithm. The coordinate act as input to the ANFIS and the angles act as the output. The learning algorithm "teaches" the ANFIS to map the coordinate to the angle through a process called training. In the training phase, the membership function 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 input output map and it is tested with the deduced inverse kinematics. Figure shows the difference in theta deduced analytically and the data predicted with ANFIS.

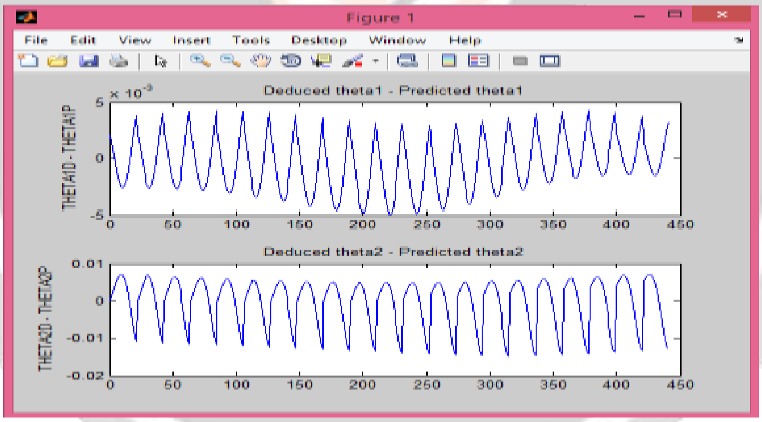

Fig 3: difference in theta(deduced) and theta (predicted data) by ANFIS trained

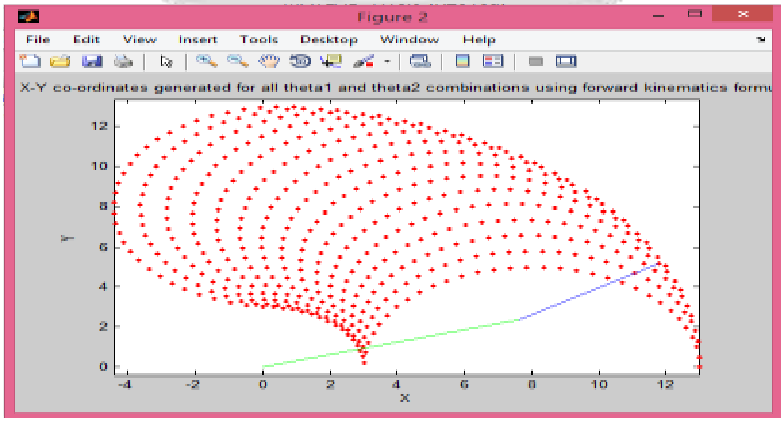

Fig 4: workspace area for robotic arm (using two joints)

6. CONCLUSION

The work space area of 2dof is defined where the resolution is maximum. The difference in theta deduced and the data predicted with ANFIS trained for 2 dof manipulator clearly depicts that the proposed method results in an acceptable solution of the inverse kinematic solution of the inverse kinematics thereby making ANFIS as an alternate approach to map the inverse kinematic solution. Other techniques like input selection and alternate ways to model the problem may be explored for reducing the error further.

REFERENCE

[1] Vargas E., R.G., Villarreal L., Reynoso J. M. y Mier-Maza R., Disenodeun Manipulator Industrial.

[2] Vargas,J.R.,G;Villarreal, L;Romero, O; Ituartre, J; Zamarripa,M;Hernandez,M; Hernandez,D; Arciniega,s; Robot manipulator.

[3] yang minglu, luguizhang, li jiangeng,"an inverse kinematics solution for manipulators".

[4] H.S.Lee, S.L.Chang,development of a CAD/CAE/CAM system for a robot manipulator.

[5] Andueza, Luis y Aguirre, Inaki, three degree of freedom robotic manipulator design for educational purpose.

[6]Anon,1992, SAE Mathematical Formulae and Statistical Tables Book.

[7] N. Abulafya, 1995, Neural networks for system identification and control.

[8] 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.

[9] Sreenivas Tejomurtula,1998, Inverse kinematics in robotics using neural networks.

[10] DavidW. Howard and Ali Zilouchian,1998 "Application of Fuzzy Logic for the Solution of Inverse Kinematics and Hierarchical Controls of Robotic Manipulators".

[11] Sreenivas Tejomurtula,1999"Inverse kinematics in robotics using neural networks Information ".

[12] SreenivasTejomurtula, Subhash Kak,1999 "Inverse kinematics in robotics using neural networks".

[13] C.M. Clark, J.K. Mills,2000 Robotic system sensitivity to neural network learning rate: theory, simulation, and experiment.

[14] Tang Y. and Velez-Diaz D,2003," Robust fuzzy control of mechanical systems".

[15] Li-Xin Wei, Hong-Rui Wang, Ying Li,2003, " A new solution for inverse kinematics of manipulator based on neural network.

[16] RasitKoker, Cemil Oz, TarkCakar, Huseyin Ekiz,2004 Robotics and Autonomous Systems.

[17] TiberiuVesselenyi, SimonaDzitac, IoanDzitac, Misu-Jan Manolescu,2007 "Fuzzy and Neural Controllers for a PneumaticActuator".

[18] SrinivasanAlavandar, 2008"Inverse Kinematics solution of 3DOF Planar Robot using ANFIS".