The implementation of an inverse kinematics solution of a 3-joint robotic manipulator using neural network and genetic algorithm

Raşit KÖKER[1] Tarık ÇAKAR[2]

[1]Electrical and Electronics Engineering Department
Faculty of Technology Sakarya University
Sakarya, Turkey
rkoker@sakarya.edu.tr

[2]Industrial Engineering Department
Faculty of Engineering Sakarya University
Sakarya, Turkey
tcakar@sakarya.edu.tr

ABSTRACT

The inverse kinematics problem of a 3-joint robotic manipulator has been implemented by using neuro-genetic technique in this paper. Firstly, a neural network has been designed for the inverse kinematics solution of 3-joint robotic manipulator. Then, to minimize the error at the end effector a genetic algorithm has been applied to the neural based solution method. The genetic algorithm has been used after the implementation of the neural network to improve the obtained results. The genetic algorithm has been following the neural network based solution to improve. The error at the end effector has been significantly reduced.

Keywords: Robotics, neural networks, genetic algorithms, inverse kinematics solution, machine learning**.**

INTRODUCTION

One of the most important areas in robotics is dedicated to study kinematics of robotic manipulators. Robot kinematics deal with determination the relationship between the joint and Cartesian coordinates. Denavit and Hartenberg have successfully solved the direct kinematics problem in 1955 [2]. Automatic procedures are existing currently to assign different parameters to every robot's joint and finding out the end-effector position and orientation based on a base frame expressed in the joint coordinate frame [10].

There are many algorithms suggested for the inverse kinematics of redundant manipulators including neural networks [18], genetic algorithms [5, 12], algebraic [14] and geometric methods [17]. Mao and Hsia [8] proposed an approach for the solution of the inverse kinematics problem of redundant manipulators based on training the neural network in the inverse modelling manner in an environment with obstacles using forward kinematics functions. One of the analytical tools are ellipsoids that are available to access the capabilities of redundant manipulators. For the analysis of the kinematic model of redundant manipulators various types of ellipsoids have been formulated and examined [3]. Marcos and Machado [9] have proposed a technique that is a combination of a closed loop pseudo inverse method with a genetic algorithm. Their method can also be used to avoid the joint angle drift problem for repeatable control of redundant manipulators. They reported that when the end-effector traces a closed path the robot returns to its initial posture according to simulation results. The inverse kinematics problem solution of a 7 Degrees of freedom (DOF) redundant manipulator has been done by using a closed-loop inverse kinematics algorithm [16]. The implementation of the algorithm has also been extended for finding redundancy resolutions at the velocity and acceleration level [15]. A study about using genetic algorithms to solve inverse kinematics problem has been presented by Nearchou [12] based on optimal point-to-point motion for the end-effector of complicated robots in complex environment cluttered with obstacles. Horacio and Simon [4] introduced a study about using simulated annealing algorithm to obtain a collision-free optimal trajectory among fixed polygonal obstacles in mobile robot path planning and tracking. Köker [6] presented a study about the inverse kinematics solution of a 6-joint robotic manipulator by using three neural networks and a genetic algorithm together. The obtained neural network solutions from these three neural networks have been included in the initial population of the genetic algorithm.

In this paper, the inverse kinematics solution of a 3-Joint robotic manipulator has been solved by using a genetic algorithm and a neural network. Firstly, a neural network has been designed for the inverse kinematics solution and then the obtained results have been improved by using a genetic algorithm.

INVERSE KINEMATICS PROBLEM IN ROBOTICS

Robot kinematics deal with applying geometry to the study of the movement of multi-degree of freedom kinematic chains which form the structure of robotic manipulator [11, 13]. The kinematics equations of the kinematic chains that form the robot is known a fundamental tool in robot kinematics analysis. By using these non-linear equations the joint parameters of the robot based on the configuration of the robot system can be mapped. Robot kinematics can be analysed in two group which are forward kinematics and inverse kinematics. Forward kinematics is computing the position of the end-effector from given values for the robot joint parameters [1]. On the other hand, inverse kinematics deals computation of the joint angles for any specified end-effector location. The kinematic architecture of the robot model has been given in figure 2. A symbolic visual view has also been presented in figure 1.

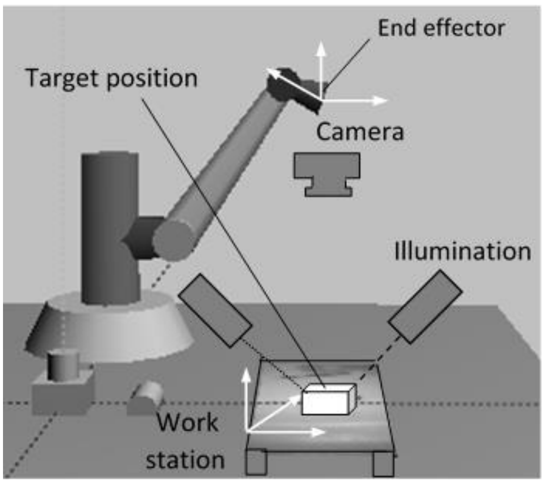

Figure 1. A symbolic view from the used robot model

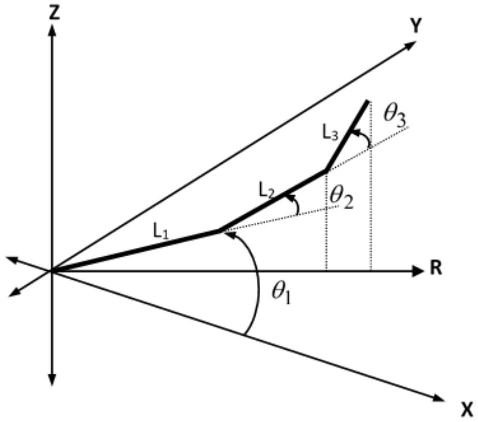

Figure 2. The kinematic structure

of the robot model

In figure 2, L_1, L_2 and L_3 refer to the link lengths for first, second and third links respectively. On the other hand, θ_1, θ_2 and θ_3 refer to joint angles for the first, second and third joints respectively.

NEURO-GENETICBASED INVERSE KINEMATICS SOLUTION

In this paper, the neural networks and genetic algorithms have been used together in the inverse kinematics problem solution of a three-joint robot for the minimization of the error at the end effector. First, a neural network has been designed for the inverse kinematic solution of the robot model. A genetic algorithm has been designed to follow the obtained solution from neural network. Here the role of the genetic algorithm is to improve the obtained result from the neural network based on searching the best fitting decimal part of the neural network solution. The pseudo-code of the solution system has been given below:

Start

Get the solution from the neural network

Repeat: *Select the solutions for the reproduction*

Apply crossover operation

Apply mutation operation

Compute the fitness function for the new generation

If the stopping criteria is not reached go to repeat

Stop.

The neural network has been trained using prepared datasets. A backpropagation neural network with sigmoidal activation function is used to train the neural network for the solution of the inverse kinematics problem. The neural designed neural network topology has been given in figure 3. The Cartesian coordinate information has been used as inputs and joint angles have been used as outputs [7].

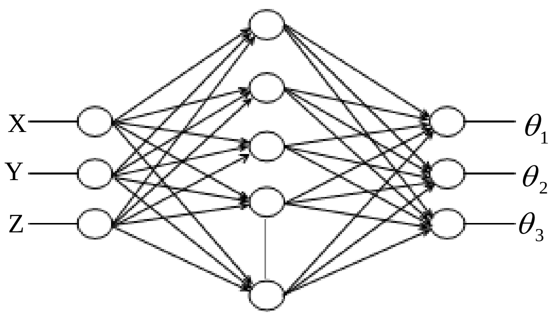

Figure 3. Used neural

network topology

This section explains the modeling and application of genetic algorithms. A genetic algorithm requires that the problem must first be coded to suit the algorithm. After the coding process, the genetic-algorithm operators are applied to the chromosomes. However, there is no guarantee that the new offspring obtained will be good solutions because of the activity of the crossover and mutation operators. Crossover, mutation, and reproduction processes continue until an optimal solution is found. The modeling of the defined problem using a genetic algorithm is presented below.

Coding operation: The obtained results from the neural network have been taken and their decimal parts have been converted to binary form. Ten digits of the decimal parts have been used in the calculations. These ten digits are represented in the binary form taking care of the maximum number possible to be in the solution cases. A sample chromosome pairs for the three joints have been shown below:

$\theta_1 = 42.\boxed{8765958515} \longrightarrow$ These decimal parts will be improved by using genetic algorithm.

$\theta_2 = 35.\boxed{1987895465}$

$\theta_3 = 22.\boxed{0987581471}$

These values can be coded in the binary form to implement genetic algorithm given as chromosomes below. And in these binary number sequences, each bit is defined as a gene.

1000001010011111011110100101110011

0001110110011111001110000010101001

0000111010110111010100110000011111

Initial population: The initial population has not been generated randomly but the obtained solution from the neural network has also been included.

Crossover operation: The crossover operation is done based on crossing obliquely from randomly selected cut point inside the chromosomes. After the crossover operation, two new ones are gotten.

Mutation: According to the defined mutation rate, a gene is randomly selected among the chromosomes in the population to be changed. It means that according to the defined mutation rate, some of genes will be converted to 1 if it is 0 or reversely.

Selection and reproduction: In this section, the determination of the list of candidate genes is done. The elitism selection method that is based on selection of the best ones has been used in this study. Since the population size has been defined as one hundred, after applying the processes given above the best hundred chromosomes are selected to survive among the old and new solutions [6]. Others will be killed and this situation will be keep going like this until the stopping criteria has been achieved. Error change according to the number of generations has been given in figure 4.

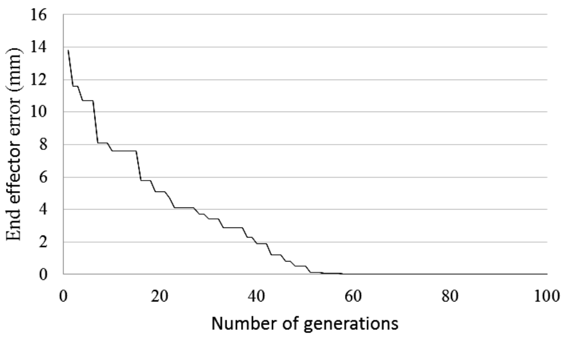

Figure 4. A sample error curve versus number of generations

Obtained sample solutions from the neural networks and genetic algorithm have been presented below:

Target Cartesian position information

X	Y	Z
61	20	56

Obtained angular positions from the neural network

θ_1	θ_2	θ_3
18.1941081	18.2136753	-70.8196293

Improved result after the genetic algorithm

θ_1	θ_2	θ_3
18.1527059851	18.2758749359	-70.9771857114

CONCLUSION

In this paper, a neural network and a genetic algorithm have been used together for the inverse kinematics problem solution of a three-joint robotic manipulator. The algorithm has been designed based on improving the inverse kinematics solution obtained from a neural network. Since the principal of a neural network is based on working with an acceptable error, the error at the end of learning can be needed to be minimized for some critical applications. The genetic algorithm has been used in this study as a search algorithm. The decimal parts of the solutions have been searched in the genetic algorithm to find better results than the current ones. The genetic algorithm takes the decimal part of the solution obtained from the neural network and then it tries to find the best solution by searching among the numbers of decimal part. The genetic algorithm uses the end effector error as a fitness function to minimize the error. The error at the end effector has been reduced significantly. This solution method can be used in case of more sensitive results are needed depending on the application. Based on using a genetic algorithm, the error can be reduced to micrometer levels.

REFERENCES

[1] Craig J.J. Introduction to Robotics: Mechanics and Control. (3rd Edition), Prentice-Hall; 2004.

[2] Denavit J., R. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. ASME Journal of Applied Mechanics 1955; 215–221.

[3] Gravagne I.A., Walker I.D., Manipulability, force, and compliance analysis for planar continuum manipulators. Robotics and Automation 18 2002; 3: 263–273.

[4] Horacio M.A., Simon G.G. Mobile robot path planning and tracking using simulated annealing and fuzzy logic control. Expert System Apply 1998; 15: 421–429.

[5] Khoogar A.R., Parker J.K. Obstacle avoidance of redundant manipulators using genetic algorithms. IEEE Proceedings of Southeastcon'91, 07–10 April 1991, Williamsburg, Virginia, 1991; Vol. 1: pp. 317–320.

[6] Köker R., A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization. Information Sciences, 10 February 2013, Volume 222:528-543.

[7] Köker R., Öz C. , Çakar T. , Ekiz H.

A study of neural network based inverse kinematics solution of a three-joint robot. Robotics and Autonomous Systems 2004; Vol. 49: pp. 227-234.

[8] Mao Z., Hsia T.C., Obstacle avoidance inverse kinematics solution of redundant robots by neural networks. Robotica 1997; 15 (1): 3–10.

[9] Marcos M.G., Machado J.A.T., Perdicoulis T.P.A., An evolutionary approach for the motion planning of redundant and hyper-redundant manipulator. Nonlinear Dynamics 2010; 60: 115–119.

[10] Martin J.A., Lope J.D. A method to learn the inverse kinematics of multi-link robots by evolving neuro-controllers. Neurocomputing. Vol. 72 (13-15): 2806-2814.

[11] McCarthy J.M. Introduction to Theoretical Kinematics. Cambridge MA: MIT Press; 1990.

[12] Nearchou A.C. Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm. Mechanism and Machine Theory (3) 1998; 33:273–292.

[13] Paul R. Robot manipulators: mathematics, programming and control: the computer control of robot manipulators. ISBN 978-0-262-16082-7.Cambridge, MA: MIT Press; 1981.

[14] Perez A., McCarthy J.M. Clifford Algebra exponentials and planar linkage synthesis equations. Journal of Mechanical Design 2005; 127:931–940.

[15] Sardana L. ,Sutar M.K. , Pathak P.M. A geometric approach for inverse kinematics of a 4-link redundant In-Vivo robot for biopsy. Robotics and Autonomous Systems 2013; 61:1306–1313.

[16] Wang J., Li Y., Zhao X. Inverse kinematics control of a 7-DOF redundant manipulator based on the closed-loop algorithm. Advanced Robotic Systems 2010; 7 (4):1–10.

[17] Yahya S., Moghavvemi M., Mohamed H.A.F. Geometrical approach of planar hyper-redundant manipulators: inverse kinematics, path planning and workspace. Simulation Modelling Practice and Theory 2011; 19:406–422.

[18] Youshen X., Wang J. A dual neural network for kinematic control of redundant robot manipulators. IEEE Transactions on Systems, Man and Cybernetics 2001; 31 (1):147–154.