International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:5, No:11, 2011

Optimization of Inverse Kinematics of a 3R Robotic Manipulator using Genetic Algorithms

J. Ramírez A., and A. Rubiano F.

***Abstract*—**In this paper the direct kinematic model of a multiple applications three degrees of freedom industrial manipulator, was developed using the homogeneous transformation matrices and the Denavit - Hartenberg parameters, likewise the inverse kinematic model was developed using the same method, verifying that in the workload border the inverse kinematic presents considerable errors, therefore a genetic algorithm was implemented to optimize the model improving greatly the efficiency of the model.

***Keywords*—**Direct Kinematic, Genetic Algorithm, Inverse Kinematic, Optimization, Robot Manipulator

I. Introduction

INDUSTRIAL robots and computer-aided systems, are the latest trend in the automation of fabrication processes, since the advances in the sensors field allow to develop more sophisticated tasks.

The use of robots in the industry is wide since they accomplish tasks that are dangerous or monotonous for humans, this is the case of a industrial robot used for cleaning in electrical substations, which works in the high voltage area[1-2].

The manipulator robots are characterized by having design limitations in terms of stability, balance and weight distribution.

Another important consideration in the design of robot manipulators is the kinematic analysis, since it involves calculating relative positions between the coordinate system attached to the moving parts causing a possible increase of the uncertainty and the accumulated error in the transformations as consequence of this it would affect the positioning accuracy and tracking of trajectories of the manipulator.

Is important to highlight that the kinematics analysis is usually treated from two points of view, direct and inverse, the last one is extremely important since it allows to calculate the joint values of the robot to approach a point in space, above the inverse kinematics solution, it is essential for robots that follow paths.

The solution of the inverse kinematics of an industrial robot can provide multiple configurations to get the manipulator positioning, studies have been based evolutionary programming using real-coded genetic algorithms for solving the multimodal problem of inverse kinematics for industrial robots, experimented with PUMA and SCARA robot, allowing to evaluate the efficiency of this approach [3], other studies suggest the optimization solution to the inverse kinematics making use of genetic algorithms [4-8]

Studies related to the solution of inverse kinematics of a robot with six degrees of freedom, using neural networks from the radial function. The most outstanding characteristics of the merger includes the accurate prediction of inverse kinematics solutions with less computation time, apart from generating training data for neural networks with the relations of the direct kinematics of the robot. [9-10]

Also have been used perceptron neural networks as "error-back-propagation algorithm", showing that the accuracy of this approach over the traditional one. [11].

Considering the previous developments, the research group "Automatización industrial" of the "Universidad Cooperativa de Colombia", designed and implemented a three degrees of freedom industrial manipulator.

The robot is raised for multiple applications in the industry by highlighting the line production, which may be used in industrial automation applications, it may add multiple end effectors for different purposes such as suction, welding, painting, manipulating parts between others.

Below, outlines the development strategies and optimization of cinematic model of the robot designed by the research group.

II. Description General del Robot

The analysis presented in this document is the result of the development of the project with title *"Design and Construction of an Anthropomorphic Robot Manipulator With Three Degrees of Freedom"*. The robot named TEACHBOT-01, was considered with three degrees of freedom and its end gripper, with a reach of 873mm and load capacity of 50 g.

It has been proposed an open architecture to allow the interaction with components of the robot, offering the capability of implement different control strategies and optimization on the robot.

III. Direct Kinematic

The solution to the direct kinematics problem, consist to find the value of the end position of the robot manipulator, this solution is a function of joint values, translational or rotary joints linked. There are several methods to resolve this problem, in this particular case was done using the homogeneous transformation matrices method and Denavit – Hartenberg's systematic representation of reference systems, because although you may find the final position geometrically, this method offers a response which could relate the position of the end of each link in the kinematic chain compared to the previous or the global reference system

J. Ramirez A. is with the Universidad Cooperativa de Colombia, Av Caracas No 37-63 (phone: 57 1 3323565 ext 160; Mobil Phone: 573208352509; e-mail: jose.ramirez@campusucc.edu.co).

A. Rubiano F. is with the Universidad Cooperativa de Colombia, Av Caracas No 37-63 (phone: 57 1 3323565 ext 160; e-mail: astrid.rubiano@campusucc.edu.co).

[12], in order to define the position of each articulation in the robot.

A homogeneous matrix $^{i-1}A_i$ is a 4x4 matrix that contains information related to the position and orientation of the reference system attached to the link i of the manipulator compared whit the reference system of the link i-1, in that way the matrix $^{0}A_1$, represents the position and orientation of the coordinate system S1 compared with the coordinate system S0, if S0 is placed in the frame of the manipulator and S1 y in the end of link one, the matrix $^{0}A_1$ represents the position of S1 compared with the fixed coordinate system of the robot. So that the representation of the end position of the manipulator is $^{0}A_n$ where n is the degrees of freedom this matrix is usually named T and is given by:

$$T = {}^{0}A_n = {}^{0}A_1\,{}^{1}A_2\,{}^{2}A_3 \ldots {}^{n-1}A_n, n \to Dof \tag{1}$$

$$T = \prod_{i=1}^{n} {}^{i-1}A_i \tag{2}$$

For the calculus of $^{i-1}A_i$ matrices, Denavit – Hartenberg's parameters must be defined, this parameters rely exclusively on the geometric characteristics of each link and allow placing the coordinate systems in each.

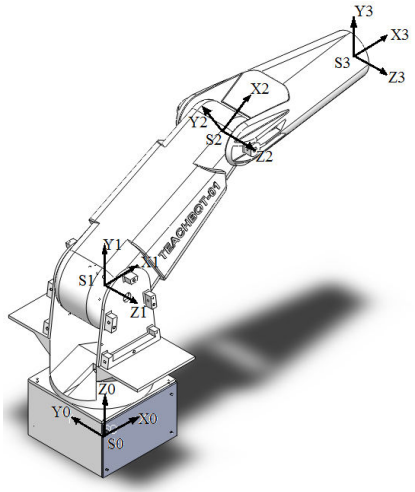

Fig. 1 Location of coordinate systems

The parameter characteristics are[13]:

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.

The coordinate systems must be placed like in figure 1, to comply with the characteristics of the parameters listed above. In table 1. Denavit – Hartenberg's parameters are shown for the TEACHBOT-01.

The general form of rotation matrices is given by:

TABLE I
DENAVIT – HARTEMBERG PARAMETERS

Articulación	θ [a]	d[m] [b]	a[m] [c]	α[rad] [d]
1	θ_1	0.28616	0	π/2
2	θ_2	0	0.30226	0
3	θ_3	0	0.285	0

[a] Rotation around Z_{i-1} axis.
[b] Translation along Z_{i-1} axis.
[c] Translation along X_i axis.
[d] Rotation around X_i axis.

$$^{i-1}A_i = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\cos(\theta_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

Replacing the values of Table 1, equation 3:

$$^{0}A_1 = \begin{bmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & 0 \\ 0 & 1 & 0 & 0.28616 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$^{1}A_2 = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & 0.30226\cos(\theta_2) \\ \sin(\theta_2) & \cos(\theta_2) & 0 & 0.30226\sin(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$^{2}A_3 = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 0.285\cos(\theta_3) \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0.285\sin(\theta_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

Finally substituting equations 4, 5 and 6 in Equation 2 gives the matrix T, this matrix consists of:

$$T = \begin{bmatrix} n & o & a & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Vectors n, o and a represent the orientation of the coordinate system and P the end position. Therefore, considering that this study is centered on the end position of the manipulator, only the P vector wil be used to describe the end position of the manipulator in terms of the angles θ_1, θ_2 y θ_3.

$$P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \tag{8}$$

The coordinates of the endpoint of the manipulator are:

$$P_x = .30226\cos(\theta_1)\cos(\theta_2) + .285\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) - .285\cos(\theta_1)\sin(\theta_2)\sin(\theta_3) \quad (9)$$

$$P_y = .30226\cos(\theta_2)\sin(\theta_1) + .285\cos(\theta_2)\cos(\theta_3)\sin(\theta_1) - .285\sin(\theta_1)\sin(\theta_2)\sin(\theta_3) \quad (10)$$

$$P_z = .30226\sin(\theta_2) + .285\cos(\theta_2)\sin(\theta_3) + .285\cos(\theta_3)\sin(\theta_2) + .28616 \quad (11)$$

IV. Inverse Kinematic

The inverse kinematics of a manipulator is a term used to denote the calculation of the manipulator joint values, necessary to position a point in space referenced to the global coordinate system of the manipulator. For this case since it is a 3R robot, we calculated the values of θ_1, θ_2 and θ_3 based on the point P_x, P_y and P_z.

There are several ways of dealing the problem of inverse kinematics, since the target of this manipulator is tracking trajectories; it is considered that the best way to solve the problem would be finding a set of closed equations through a mathematical relationship of the form:

$$\theta_k = f_k\left(P_x, P_y, P_z\right)$$
$$k = 1...n(DOF) \quad (12)$$

This model would achieve providing a real-time solution, suitable for tracking trajectories.

This kind of function can be calculated using diferent methods, for the case and since it is a manipulator with 3 degrees of freedom, the homogeneous transformation matrix method was chosen.

Equation 1 for a manipulator with three degrees of freedom is reduced to:

$$T = {}^0A_n = {}^0A_1\,{}^1A_2\,{}^2A_3 \quad (13)$$

Since T is a matrix of the form:

$$T = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

Solving the equation 13 for ${}^2A_3$, a system of three equations with three variables can be achieved, allowing calculating the joint values based on P_x, P_y and P_z.

$$\left({}^0A_1\right)^{-1}\left({}^1A_2\right)^{-1} T = {}^2A_3 \quad (15)$$

$$\begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & -0.28616 \\ \sin(\theta_1) & -\cos(\theta_1) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta_2) & \sin(\theta_2) & 0 & -0.30226 \\ -\sin(\theta_2) & \cos(\theta_2) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} *$$
$$\begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 0.285\cos(\theta_3) \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0.285\sin(\theta_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

Matching column 4 of each side of the equal are:

$$P_z\sin(\theta_2) - 286.16\sin(\theta_2) + P_x\cos(\theta_1)\cos(\theta_2) + P_y\cos(\theta_2)\sin(\theta_1) - 302.26 = 0.285\cos(\theta_3) \quad (17)$$

$$P_z\cos(\theta_2) - 286.16\cos(\theta_2) - P_x\cos(\theta_1)\sin(\theta_2) - P_y\sin(\theta_1)\sin(\theta_2) = 0.285\sin(\theta_3) \quad (18)$$

$$P_x\sin(\theta_1) - P_y\cos(\theta_1) = 0 \quad (19)$$

From equation 19, we can solve the angle θ_1:

$$\frac{\sin(\theta_1)}{\cos(\theta_1)} = \frac{P_y}{P_x} = \tan(\theta_1) \quad (20)$$

$$\theta_1 = \tan^{-1}(\theta_1) \quad (21)$$

Factorizing in equations 17 and 18:

$$P_z\sin(\theta_2) - 286.16\sin(\theta_2) + P_x\cos(\theta_2)\cos(\theta_2) + P_y\cos(\theta_2)\sin(\theta_1) - 302.26 = 285\cos(\theta_3) \quad (22)$$

$$\sin(\theta_2)[P_z - 286.16] + \cos(\theta_2)[P_x\cos(\theta_1) + P_y\sin(\theta_1)] - 302.26 = 285\cos(\theta_3) \quad (23)$$

$$P_z\cos(\theta_2) - 286.16\cos(\theta_2) - P_x\cos(\theta_2)\sin(\theta_2) - P_y\sin(\theta_2)\sin(\theta_2) = 285\sin(\theta_3) \quad (24)$$

$$-\sin(\theta_2)[P_x\cos(\theta_2) + P_y\sin(\theta_2)] + \cos(\theta_2)[P_z - 286.16] = 285\sin(\theta_3) \quad (25)$$

Replacing,

$$a = P_x\cos(\theta_1) + P_y\sin(\theta_1) \quad (26)$$

$$b = P_z - 286.16 \quad (27)$$

$$\sin(\theta_2)b + \cos(\theta_2)a - 302.26 = 285\cos(\theta_3) \quad (28)$$

$$-\sin(\theta_2)a + \cos(\theta_2)b = 285\sin(\theta_3) \quad (29)$$

Adding equations 27 and 28 squared

$$a\cos(\theta_2) + b\sin(\theta_2) = \frac{[a^2 + b^2] + 10136}{60452} \quad (30)$$

$$c = \frac{[a^2 + b^2] + 10136}{60452} \quad (31)$$

$$\cos(\theta_2) + b\sin(\theta_2) = c \quad (32)$$

Equation 32 appears to be a transcendental equation in one variable, the following relationships are used to turn the equation into a polynomial equation.

$$u = \tan\left(\frac{\theta_2}{2}\right) \quad (33)$$

$$\cos(\theta_2)=\frac{1-u^2}{1+u^2} \quad (34)$$

$$\sin(\theta_2)=\frac{2u}{1+u^2} \quad (35)$$

Replacing 34 and 35 in 32 we have:

$$a\frac{1-u^2}{1+u^2}+b\frac{2u}{1+u^2}=c \quad (36)$$

$$(1-u^2)a+2ub=c(1+u^2) \quad (37)$$

$$u^2(a+c)-2ub+c-a=0 \quad (38)$$

$$u=\frac{2ub\pm\sqrt{4b^2-4(c^2-a^2)}}{2(a+c)} \quad (39)$$

$$u=\frac{b\pm\sqrt{b^2+a^2-c^2}}{a+c}=\tan\left(\frac{\theta_2}{2}\right) \quad (40)$$

$$\theta_2=1\tan^{-1}\left(\frac{b\pm\sqrt{b^2+a^2-c^2}}{a+c}\right) \quad (41)$$

Dividing equation 29 and 28:

$$\tan(\theta_3)=\frac{-\sin(\theta_2)a+\cos(\theta_2)b}{\sin(\theta_2)b+\cos(\theta_2)a-302.26} \quad (42)$$

$$\theta_3=\tan^{-1}\left(\frac{-\sin(\theta_2)a+\cos(\theta_2)b}{\sin(\theta_2)b+\cos(\theta_2)a-302.26}\right) \quad (43)$$

Equations 21, 41 and 43 are to the solution of inverse kinematics of the proposed robot. From these equations can be evidenced that there are two possible solutions for θ_2 and therefore to θ_3.

V. Model Validation Software

A graphical interface was designed using MATLAB to validate the performance of the found models for the direct and inverse kinematics.

For the direct kinematics, the interface has a data input section in order to enter the Denavit - Hartenberg parameters, this considering the possibility of analyze different geometries of RRR robot, see Figure 2, additionally the user has the possibility of indicate the joint values change limits in θ_1, θ_2 and θ_3 and three slide bars used to modify the input angles.

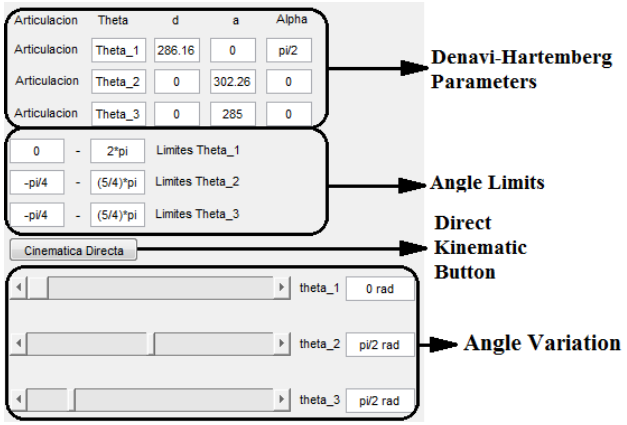

Fig. 2 Interface for direct kinematics calculation

To display the results of the direct kinematics, three graphs showing the position of the manipulator was used, see Figure3.

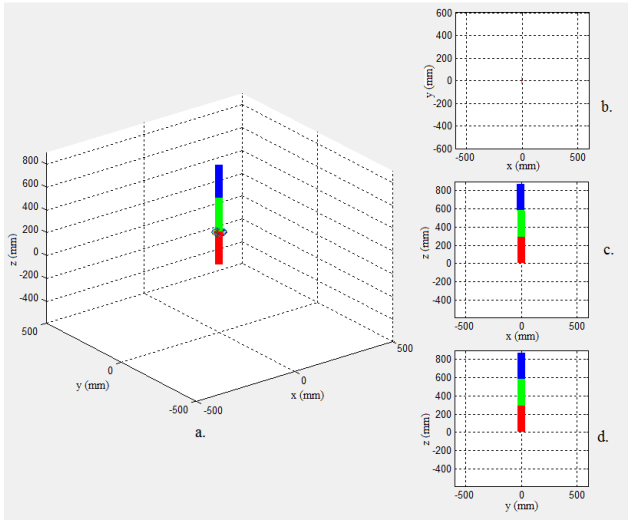

Fig. 3a View X(mm),Y(mm),Z(mm) of the robot, b. View X(mm),Y(mm), c. View X(mm),Z(mm), d. View Y(mm),Z(mm),

Once the direct Kinematics button is pressed, see Figure 2, the button activates the inverse kinematics. The interface of the inverse kinematics consists of three fields for entering The P_x, P_y and P_z coordinate, a button to start calculating and three text fields for displaying the calculated angle and the calculated position, see figure 4.

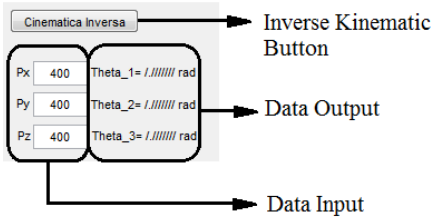

Fig. 4 Inverse Kinematic Interface

When entering the required point in the text fields P_x, P_y and P_z, the program calculates and displays the found angles and check the end point using the direct kinematics algorithm. Figure 5 shows the result of inverse kinematics.

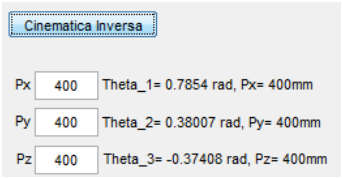

Fig. 5 Inverse Kinematic Result

VI. Inverse Kinematic Optimization

Using the tool developed for validating the inverse kinematic model, it was found that there are some points on the workload edge whose solution reaches a precision error values up to 63.72%, considering that the manipulator must have an accuracy of 1mm. and the error value on the workload edge, was decided to take an optimization strategy that would achieve much lower error values.

The chosen optimization method was genetic algorithms, due to its fast convergence, maintaining the condition of providing real-time response.

A. Genetic Algorithms Overview

Genetic algorithms are based on natural selection principle, genetics and evolution, it is assumed that the evolution involves relevant variables such as chromosomes, the crossing of individuals, mutation and new generations of individuals.

In general in the natural selection process, individuals with the best chromosomes, are who will be able to spend their genetic information to the next generation, in a similar manner in a genetic algorithm a population of individuals (solutions) who can offers the best solution for the problem will be rewarded allowing them to spend their genetic information (chromosomes) to the next generation (New population solution)[14].

The following flowchart describes generally the operation of a genetic algorithm.

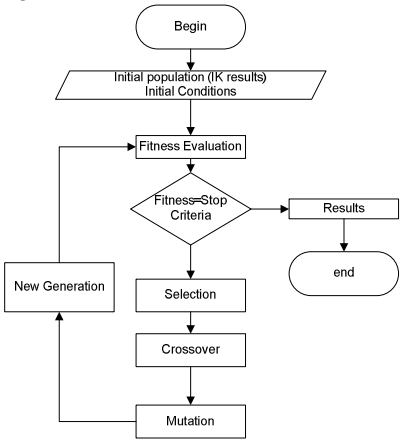

Fig. 6 Genetic Algorithm Structure.

B. Implemented Algorithm

In this case, the population is a set of individuals, each one formed by the angles θ_1, θ_2 and θ_3, the initial population will be composed of individuals close to the found values by applying inverse kinematics, the result of the inverse kinematics will be starting point of the genetic algorithm, this because this is the result which will be optimized and landed a better response time.

Each individual will be evaluated to see if its set of angles can resolve the value of P_x, P_y and P_z, for this evaluation of individuals a fitness function is used, this function describes the individual's ability to solve the problem.

The fitness function is set up in a first step by the equations 9, 10 and 11, because these equations indicate the position in terms of angles, in other words point to the position achieved by each individual, then the individual's position is compared to the sought position according to the following equation:

$$Fitness = \sqrt{(P_x - P_x(i))^2 + (P_y - P_y(i))^2 + (P_z - P_z(i))^2} \quad (44)$$

Where i represents the i-th individual.

The crossing of individuals was done heuristically by selecting as many chromosomes of the individual whose fitness function was greater.

In genetic algorithms, mutation is commonly used as a strategy to ensure convergence of the algorithm [15] to the desired solution. In this case the mutation technique implemented is based on the population success; in general, a less successful population means that probability of mutation is greater.

The following figure shows the implemented interface for the genetic algorithm to optimize the results of the inverse kinematics.

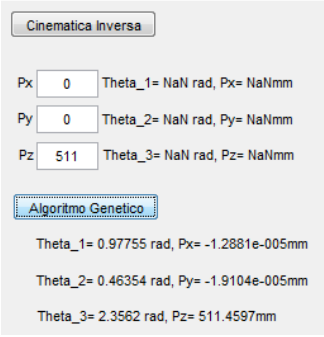

Fig. 7 Genetic Algorithm Interface

The following table shows the comparison between inverse kinematic results and the genetic algorithm optimization for 24 sought positions.

VII. Conclusion

In general, numerical analysis is an important alternative in the solution of systems whose mathematical model appears to be extremely complex, yet these have the disadvantage that they require an iterative process that in some cases fail to

reach a solution or appears to be very slow.

According to the obtained results in this study, was evidenced that an iterative process combined with the principles of artificial intelligence turns out to be an auxiliary tool to an analytical solution.

Under normal conditions, a genetic algorithm would not be appropriate to solve the inverse kinematics for a manipulator that require track trajectories in a fast and accurate way, but when the analytical solution is optimized with these algorithms high levels of accuracy are achieved and time response is acceptable.

The response times matched for the genetic algorithm applied to the solution of inverse kinematics of the manipulator is in a range of 3.1s to 4s, when it is used as a method of optimization for the inverse kinematics, the times are ranged from 1.0s 1.3s, although these times might be considered high for the solution of the problem, should be considered that will only be used when the kinematics not provide an adequate response.

TABLE II
RESULT COMPARISON

SAMPLE	INVERSE KINEMATIC FITNESS		GENETIC ALGORITHM FITNESS
	SOLUTION 1	SOLUTION 2	
1	570,20	570,15	0,01
2	569,90	570,24	0,00
3	570,47	569,91	0,02
4	0,35	0,35	0,02
5	0,27	0,27	0,02
6	0,64	0,64	0,01
7	0,38	0,38	0,02
8	570,53	569,97	0,01
9	570,10	569,92	0,00
10	0,62	0,62	0,01
11	570,12	570,20	0,01
12	0,63	0,63	0,00
13	0,13	0,13	0,02
14	0,44	0,44	0,01
15	0,63	0,63	0,02
16	0,60	0,60	0,01
17	0,35	0,35	0,02
18	0,34	0,34	0,01
19	570,44	569,92	0,02
20	570,10	569,98	0,01
21	570,15	570,52	0,02
22	0,29	0,29	0,01
23	569,89	570,41	0,01
24	570,27	570,08	0,01
GEOMETRIC MEAN	0,637	0,637	0,013
ERROR [a]	63,72%	63,72%	1,26%

[a] Accuracy 1MM

REFERENCES

[1] Vargas E., R.G., Villarreal L., Reynoso J.M. y Mier-Maza R., Diseño de un Manipulador Industrial para Aplicaciones de Limpieza en Subestaciones Eléctricas. Centro Metropolitano de Investigación en Mecatrónica, ITESM Querétaro.

[2] Vargas,J.R.,G;Villarreal,L;Romero,O;Ituarte,J; Zamarripa,M; Hernández,M; Hernández,D; Arciniega,S, Robot manipulador para aplicación de recubrimiento en líneas de alta tensión.

[3] P. Kalra, P.B. Mahapatra, D.K. Aggarwal, An evolutionary approach for solving the multimodal inverse kinematics problem of industrial robots, Mechanism and Machine Theory, Volume 41, Issue 10, October 2006, Pages 1213-1229, ISSN 0094-114X, DOI: 10.1016/j.mechmachtheory.2005.11.005.

[4] Za'er S. Abo-Hammour, Nasir M. Mirza, Sikander M. Mirza, Muhammad Arif, Cartesian path generation of robot manipulators using continuous genetic algorithms, Robotics and Autonomous Systems, Volume 41, Issue 4, 31 December 2002, Pages 179-223, ISSN 0921-8890, DOI: 10.1016/S0921-8890(02)00279-8.

[5] Maria da Graca Marcos, J.A. Tenreiro Machado, T.-P.Azevedo-Perdicoulis, Trajectory planning of redundant manipulators using genetic algorithms, Communications in Nonlinear Science and Numerical Simulation, Volume 14, Issue 7, July 2009, Pages 2858-2869, ISSN 1007-5704, DOI: 10.1016/j.cnsns.2008.10.014.

[6] Andreas C. Nearchou, Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm, Mechanism and Machine Theory, Volume 33, Issue 3, April 1998, Pages 273-292, ISSN 0094-114X, DOI: 10.1016/S0094-114X(97)00034-7.

[7] P. Th. Zacharia, N. A. Aspragathos, Optimal robot task scheduling based on genetic algorithms, Robotics and Computer-Integrated Manufacturing, Volume 21, Issue 1, February 2005, Pages 67-79, ISSN 0736-5845, DOI: 10.1016/j.rcim.2004.04.003.

[8] Wei-Min Yun, Yu-Geng Xi, Optimum motion planning in joint space for robots using genetic algorithms, Robotics and Autonomous Systems, Volume 18, Issue 4, October 1996, Pages 373-393, ISSN 0921-8890, DOI: 10.1016/0921-8890(96)00010-3.

[9] Shital S. Chiddarwar, N. Ramesh Babu, Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach, Engineering Applications of Artificial Intelligence, Volume 23, Issue 7, October 2010, Pages 1083-1092, ISSN 0952-1976, DOI: 10.1016/j.engappai.2010.01.028.

[10] RasitKoker, Cemil Oz, TarikCakar, HuseyinEkiz, A study of neural network based inverse kinematics solution for a three-joint robot, Robotics and Autonomous Systems, Volume 49, Issues 3-4, Patterns and Autonomous Control, 31 December 2004, Pages 227-234, ISSN 0921-8890, DOI: 10.1016/j.robot.2004.09.010.

[11] Evangelosa Dermatas, Andreas Nearchou, Nikos Aspragathost, Error-back-propagation solution to the inverse kinematic problem of redundant manipulators, Robotics and Computer-Integrated Manufacturing, Volume 12, Issue 4, December 1996, Pages 303-310, ISSN 0736-5845, DOI: 10.1016/S0736-5845(96)00008-7.

[12] H.S. Lee, S.L. Chang, Development of a CAD/CAE/CAM system for a robot manipulator, Journal of Materials Processing Technology, 140 (2003) 100–104, *ISSN*: 0924-0136

[13] Andueza, Luís y Aguirre, Iñaki, Three degree of freedom robotic manipulator design for educational purpose, Revista Ciencia e Ingeniería. Vol. 30, No. 1, pp. 3-14, diciembre-marzo, 2009. ISSN 1316-7081

[14] Luis Felipe Giraldo, Edilson Delgado y Germán Castellanos, Cinemática Inversa de un Brazo Robot Utilizando Algoritmos Genéticos, Av. Sist Inf., Vol. 3 No. 1 pp. 29–34, Medellín, Junio 2006, ISSN 1657–7663

[15] Maria da Graça Marcos, J.A. Tenreiro Machado, T.-P. Azevedo-Perdicoúlis Trajectory planning of redundant manipulators using genetic algorithms, Commun Nonlinear Sci Numer Simulat, 14 (2009) ISSN 2858–2869