



Inverse Kinematics Solution for Redundant Robot Manipulator using Combination of GA and NN

Hind Z. Khaleel

Department of Control and Systems Engineering/ University of Technology/ Baghdad/ Iraq
Email: hhindzuhair@yahoo.com

(Received 8 August 2016; accepted 15 October 2017)

<https://doi.org/10.22153/kej.2018.10.008>

Abstract

A demonstration of the inverse kinematics is a very complex problem for redundant robot manipulator. This paper presents the solution of inverse kinematics for one of redundant robots manipulator (three link robot) by combing of two intelligent algorithms GA (Genetic Algorithm) and NN (Neural Network). The inputs are position and orientation of three link robot. These inputs are entering to Back Propagation Neural Network (BPNN). The weights of BPNN are optimized using continuous GA. The (Mean Square Error) MSE is also computed between the estimated and desired outputs of joint angles. In this paper, the fitness function in GA is proposed. The sinwave and circular for three link robot end effector and desired trajectories are simulated by MATLAB program. Joint angles and end effector positions of robot results values of circular trajectory are better than joint angles end effector positions of robot results values of NN work in another paper. Three link redundant robot workspace is also simulated. The outputs results of best three joint angles are evaluated for two trajectories sinwave and circular, with 50 generations the algorithm is fast. This paper presents the simulations results that are obtained based on MATLAB R2010b program.

Keywords: Redundant robot, Genetic Algorithm, Neural network, Inverse kinematics.

1. Introduction

One of the main problems for serial-chain manipulators robots is the inverse kinematics problem, where, it needs to find the values of the joint positions given the position and orientation of the end-effector relative to the base. For solving the inverse kinematics problem, there are many solutions [1]. Neural networks have been widely applied to solve the robot manipulator inverse kinematics problems. The proposed hybrid approach combined the electromagnetism-like method and the neural network to solve the inverse kinematics problem. Extreme Learning Machine (ELM) is used to train the neural network [2]. Another neural network solution of the inverse kinematics problem is used for a three-joint robotic manipulator where, the neural network is trained until the error is acceptable [3]. In addition, an Artificial Neural Network (ANN) model is presented to find the inverse kinematics

solution of a SCARA manipulator [4]. The genetic algorithm is determined the optimal joint angles in a given search space for three planar manipulator robot that would contribute to a productive and a quality way of material handling and processing [5]. The only genetic algorithm is used to optimization the inverse kinematics of three degrees of freedom model industrial robot manipulator and improving the efficiency of the model [6].

2. Related Work

1. The combing of GA and BPNN algorithm of the end effector positions (x, y) and joint angles of circular trajectory are evaluated as shown in Figure 7. These positions and joint angles of circular trajectory are better than the positions and joint angles of circular trajectory of the Adrian's work [7]. Adrian's work [7] is just used NN

algorithm without GA. In addition, in this paper, the end effector positions (x, y) and joint angles of the sinwave trajectory are demonstrated and simulated as shown in Figure 8, while in Adrian's work, the only circular trajectory is presented.

2. The proposed flowchart of GA and BPNN algorithm is illustrated as shown in Figure 6 and the two difficult trajectories results are simulated, one for the sinwave trajectory and the second for the circular trajectory, but the researchers Md and Tania [10] are just presented GA and NN algorithm but they did not evaluated any results of any trajectory.

3. Modeling of Redundant Robot Manipulator

The modeling of three link redundant robot is presented as the following:

3.1. Forward Kinematics for Three Link Manipulator

The forward kinematics means that using the kinematic equations of the three link robot manipulator in order to evaluate the position of the end-effector from the joint values [7][8]. The three link manipulator robot configuration with all parameters is as shown in Figure 1.

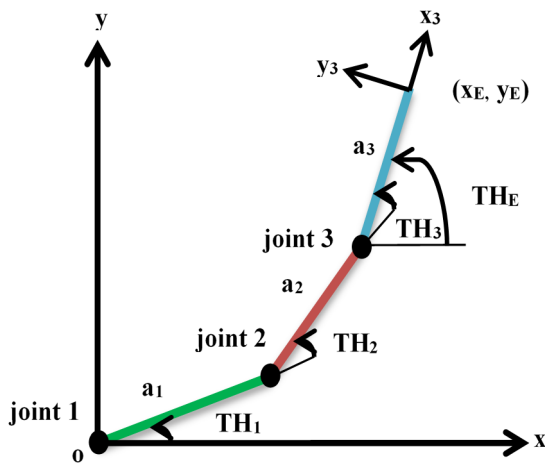


Fig. 1. Three link planner manipulator robot configuration [7].

Where a_1, a_2, a_3 are the three link of robot respectively and TH_1, TH_2, TH_3 are the joint angles. Three links are connected by joints whose all axes are perpendicular to the plane of the links.

The forward kinematics of the robot equations are illustrated below [7][8].

$$x_E = a_1 \cos(TH_1) + a_2 \cos(TH_1 + TH_2) + a_3 \cos(TH_1 + TH_2 + TH_3)$$

$$y_E = a_1 \sin(TH_1) + a_2 \sin(TH_1 + TH_2) + a_3 \sin(TH_1 + TH_2 + TH_3) \quad \dots(1)$$

Where, (x_E, y_E) are the coordinates (position) of end effector for the three link robot and the orientation is (TH_E) as in Eq. 2

$$TH_E = TH_1 + TH_2 + TH_3 \quad \dots(2)$$

The link parameter table of forward kinematics for three link robot is presented in the Table 1.

Table 1. Link parameter table of forward kinematics for three link robot

Link	a_i (m)	α_i (deg.)	d_i (m)	θ_i (deg.)
1	2	0	0	θ_1^*
2	2	0	0	θ_2^*
3	2	0	0	θ_3^*

Where, d_i is distance, θ_i is angle, a_i is length and α_i is twist angle, $\theta_1^* = TH_1, \theta_2^* = TH_2$ and $\theta_3^* = TH_3$

The ranges of three joint movements are:

$$0 < TH_1 < \pi, -\pi < TH_2 < 0, -\pi/2 < TH_3 < \pi/2$$

The three link robot is required to move along the circular trajectory in its workspace. The coordinates of the circle equation are demonstrated as below:

$$x_p = x_c + r \cos\varphi$$

$$y_p = y_c + r \sin\varphi \quad \dots(3)$$

Where, (x_p, y_p) are the desired position of the end-effector, (x_c, y_c) are the coordinates of the circle's center, r is the circle's radius. The angle (φ) ranges between 0 and 2π . The desired orientation of the end-effector is expressed by:

$$TH_d = \tan^{-1}(y_p / x_p) \quad \dots(4)$$

The three link robot is also move along the sinwave trajectory in its workspace. The coordinates and the desired orientation of the end-effector are expressed as:

$$x_p = 0: 2\pi,$$

$$y_p = \sin x_p \quad \dots(5)$$

$$\phi_i = \tan^{-1}(y_p / x_p) \quad \dots(6)$$

By using the forward kinematics Eq. 1 and Eq. 2 the workspace of redundant three link robot are simulated for 1000 random points as shown in Figure 2:

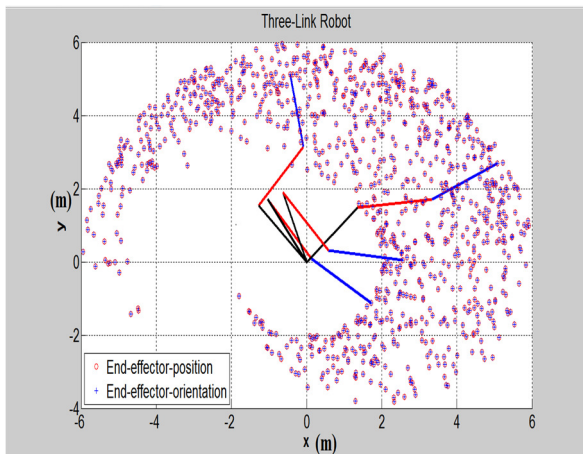


Fig. 2. Simulation results of three joints for three-link robot workspace.

Where the black line is the first link, the red line is the second link and the blue line is the third link of redundant robot.

4. Combination of GA and NN Algorithm

In this paper, the GA is used in order to optimize the weights of NN as explained in below:

- 1- The initialize inputs are end effector position and orientation (x_E, y_E, TH_E) from Eq. 1 and Eq. 2 and initialization the weights and biases values randomly.
- 2- BPNN algorithm is divided into three programs in order to evaluate each target output so that we have three programs one for TH_1 , second for TH_2 and third for TH_3 .
- 3- The multilayer feed forward neural networks consist of three layers.
- There are three programs, in each program the number of these layers is: input layer= 3, hidden=3 and output=1 for neuron=30 so that finally there are three outputs of joint angles.
- 4- The back propagation neural network (BPNN) is used and the epoch=1000 as shown in Figures (3, 4, 5).

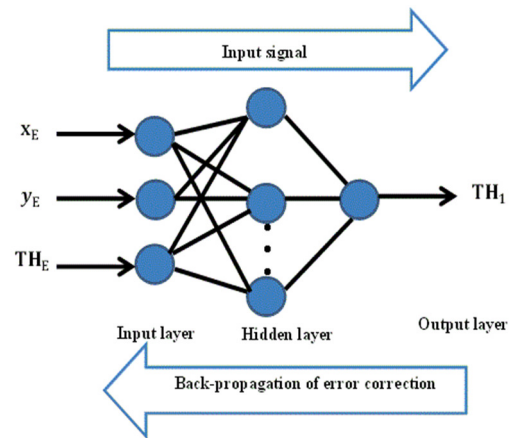


Fig. 3. BPNN structure for three-link robot of TH_1 output.

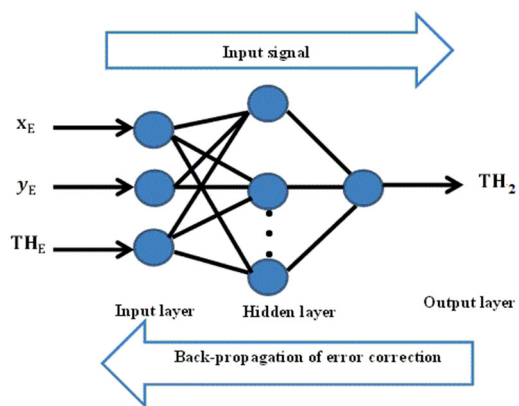


Fig. 4. BPNN structure for three-link robot of TH_2 output.

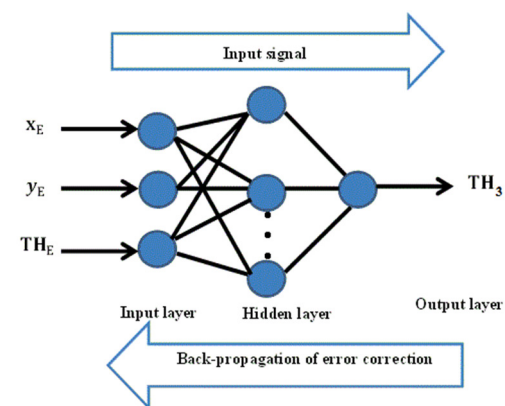


Fig. 5. BPNN structure for three-link robot of TH_3 output.

5- The performance of the neural network is evaluated by the general form of Mean Square Error (MSE) for output is:

$$MSE = \frac{1}{n} \sum_{i=1}^n (\text{desired}_i - \text{estimated}_i)^2 \quad \dots(7)$$

Where, n: total number of samples.

6- If the MSE < minimum error then go to optimise the weights and biases values by using continuous GA.

7- The continuous GA is programmed using toolbox [9] and the parameters are shown in Table 2:

Table 2.
The specifications of GA

Simulation Parameter	Value
Population	30
Number of Generations	50
Crossover Rate	80%
Mutation Rate	5%

8- The fitness function is proposed as minimization the following form:

$$\text{fitness} = \frac{1}{1+\text{MSE}} \quad \dots(8)$$

According to the step 2 there are three programs so that the Eq. 8 is applied once time in each program in order to get the optimal angles outputs (TH₁, TH₂, TH₃) by optimization the NN weight values. Figure 6 shows that the proposed flowchart of the combination GA and BPNN algorithm as compared with [10]. In this paper, the proposed equation is used as a fitness function as in Eq. 8 and also, the continues GA is used in order to optimize the weights values.

The continues GA is used because is faster than the binary GA. The two difficult trajectories are presented, one for sinwave trajectory and second for circular trajectory. The results of this paper are demonstrated the best results of joint angles for three link redundant robot. As compared with another work of the researchers Md and Tania [10] where, they did not have any results of any trajectory.

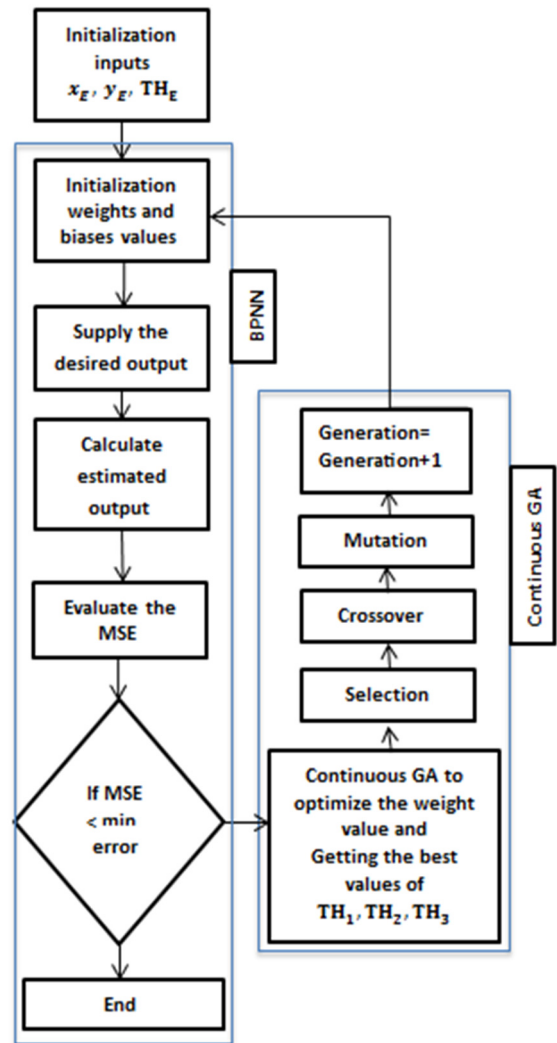


Fig. 6. Flowchart of BPNN optimization by GA.

5. Simulation and Discussion Results

MATLAB program version R2010b is used in this work in order to simulate the results of three link redundant robot by using combination of GA and NN algorithm. In this paper the outputs results is evaluated of best three joints angles (TH₁, TH₂, TH₃) in radians of the sinwave and the circular trajectory for the redundant three link robot as shown in the following Table 3 and Table 4:

Table 3.
Best three joints angles results of the sinwave trajectory for three link redundant robot.

	TH ₁ (rad.)	TH ₂ (rad.)	TH ₃ (rad.)
1	2.09439510239320	2.09439510239320	-4.18879020478639
2	2.79682251513915	2.25366107507819	-4.26842319112889
3	2.71152693748771	2.40412061902860	-4.34365296310410
4	2.62427838158158	2.54472526439636	-4.41395528578797
5	2.53526739838127	2.67462289897448	-4.47890410307703
6	2.44446364128157	2.79330131331349	-4.53824331024654
7	2.35159812137175	2.90073067337049	-4.59195799027504
8	2.25618015885843	2.99749242117126	-4.64033886417542
9	2.15756521427283	3.08487018524838	-4.68402774621398
10	-1.06323016907846	3.11831562487515	-1.55915781243758
11	-1.09481259810948	3.04305133092376	-1.52152566546188
12	-1.13254807333171	2.96945160644182	-1.48472580322091
13	-1.17282514346554	2.89434286113368	-1.44717143056684
14	-1.21168625695458	2.81483514145648	-1.40741757072824
15	-1.24526823875282	2.72868180908046	-1.36434090454023
16	-1.27025409882385	2.63451890278899	-1.31725945139450
17	-1.28420404400100	2.53192829443235	-1.26596414721617
18	-1.28568282337062	2.42132944016560	-1.21066472008280
19	-1.27418523676859	2.30375290665779	-1.15187645332890
20	-1.24992622513004	2.18056245366256	-1.09028122683128
21	-1.21357817080457	2.05317572280596	-1.02658786140298
22	-1.16601680014912	1.92280583262665	-0.961402916313326
23	-1.10810435642057	1.79022314001794	-0.895111570008968
24	-1.04051183255196	1.65552230081735	-0.827761150408674
25	-0.963563644590466	1.51786835868874	-0.758934179344371
26	-0.877069353067111	1.37517061049141	-0.687585305245703
27	-0.780065882625219	1.22355531083850	-0.611777655419249
28	-0.670264929463752	1.05625087844625	-0.528125439223125
29	-0.542489244228584	0.860473857929541	-0.430236928964771
30	-0.382265572924653	0.604664848551176	-0.302332424275588

Table 4.
Best three joints angles results of the circular trajectory for three link redundant robot.

	TH ₁ (rad.)	TH ₂ (rad.)	TH ₃ (rad.)
1	-0.359000500128900	1.74470794844022	-0.891416538376323
2	-0.349000367928000	1.74470829362557	-0.880525068783556
3	-0.319586894854553	1.73157976642175	-0.863870052585719
4	-0.259220256446807	1.72581950794822	-0.861147611399690
5	-0.225801224656687	1.73670909020314	-0.868219308432655
6	-0.202804732134982	1.76905729194382	-0.891939490523948
7	-0.190725784329956	1.83226560408357	-0.918965817896581
8	-0.186648895748110	1.90991302234134	-0.965276548758722
9	-0.207409242369409	1.99777818433564	-0.993677994456423
10	-0.211033839456256	2.06970777585943	-1.05095161771735
11	-0.222800551424235	2.16448724068362	-1.11296511384970
12	-0.298593737649912	2.31664157226902	-1.14749549766792
13	-0.323658967067925	2.40977504100077	-1.21451016137534
14	-0.386628436397731	2.50458713022614	-1.24553517872607
15	-0.469689148396207	2.61568585061130	-1.31873208478356
16	-0.553184555200184	2.68625454397890	-1.34068568646420
17	-0.637319007507458	2.77123450838716	-1.37817289235731
18	-0.696122532778854	2.80714614211468	-1.40876908296648
19	-0.809744922978563	2.83702052480132	-1.41827793872577
20	-0.891817858560614	2.82927432255935	-1.41558668669993
21	-0.918072453635902	2.75427010517449	-1.40346938555446
22	-0.990258741674132	2.74176278949666	-1.38096679821293
23	-0.995110369446019	2.65892617286818	-1.33715324615576
24	-0.999999999999810	2.61648188831068	-1.28374208696240
25	-0.958696617519971	2.51550410086786	-1.24420863108735
26	-0.849059015280863	2.31437465818569	-1.18259768460465
27	-0.819666197772313	2.24490448157502	-1.12570211327658
28	-0.757983669931919	2.14688005730974	-1.08253153714546
29	-0.694423514277570	2.07019207007971	-1.03606206660692
30	-0.628635577131678	2.00167217994412	-0.995038006886263

Figure 7 below shows that the desired circular trajectory is simulated depending on Eq. 3 and Eq. 4. Where, the combining of GA and BPNN algorithm of the end effector circular trajectory is evaluated and it is better than the circular trajectory of the Adrian's work [7] where, the Adrian's work is just used NN algorithm.

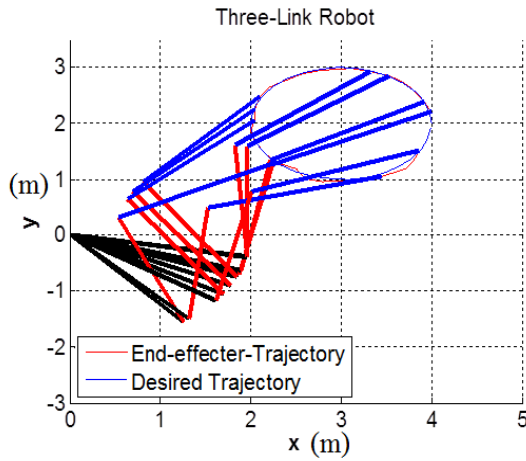


Fig. 7. The three link robot end effector and the desired circular trajectories.

In Figure 8 the three link robot sinwave end effector trajectory is simulated according to combine of GA and BPNN algorithm, while, the desired sinwave trajectory is simulated according to the Eq. 5 and Eq. 6.

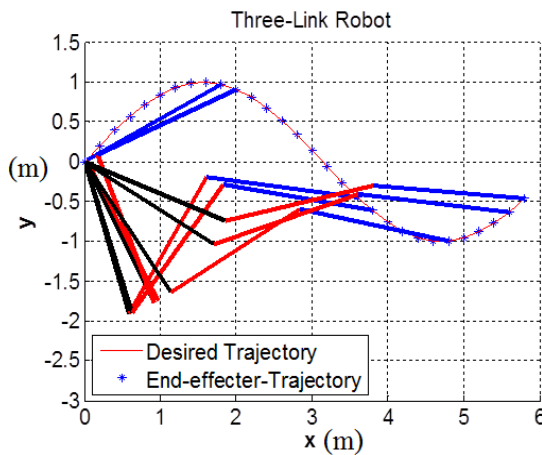


Fig. 8. The three link robot end effector and the desired sinwave trajectories.

In each trajectory there are three fitness functions are evaluated according to the Eq. 8 as shown in Figures (9, 10, 11, 12, 13, 14), where the black color is the best fitness values and the blue color is mean fitness:

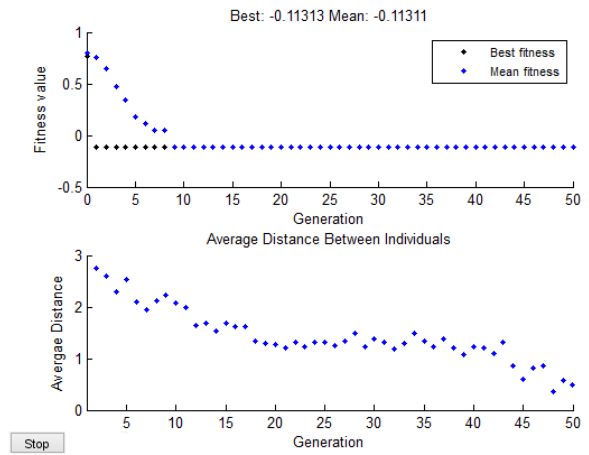


Fig. 9. Best fitness value for TH₁ output of sinwave trajectory and average distance between individuals.

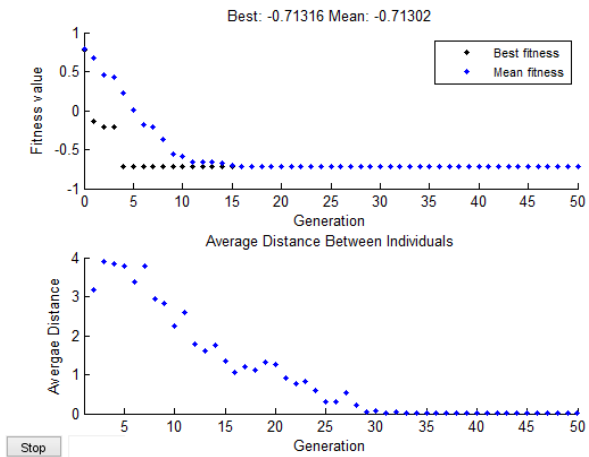


Fig. 10. Best fitness value for TH₂ output of sinwave trajectory and average distance between individuals.

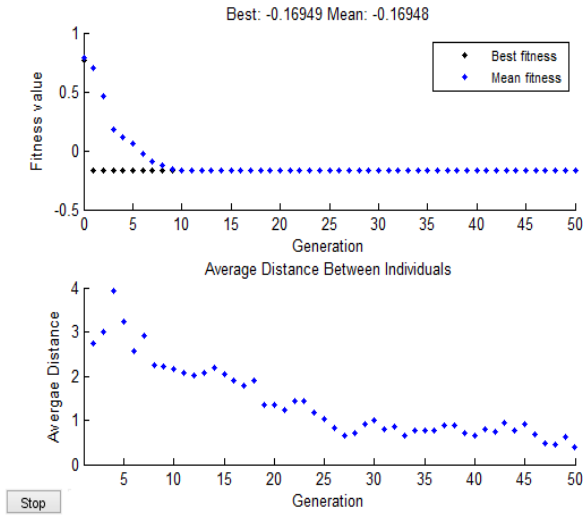


Fig. 11. Best fitness value for TH₃ output of sinwave trajectory and average distance between individuals.

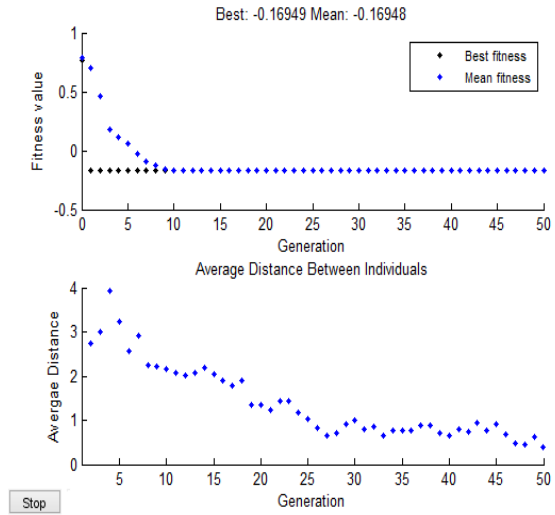


Fig. 14. Best fitness value for TH₃ output of circular trajectory and average distance between individuals.

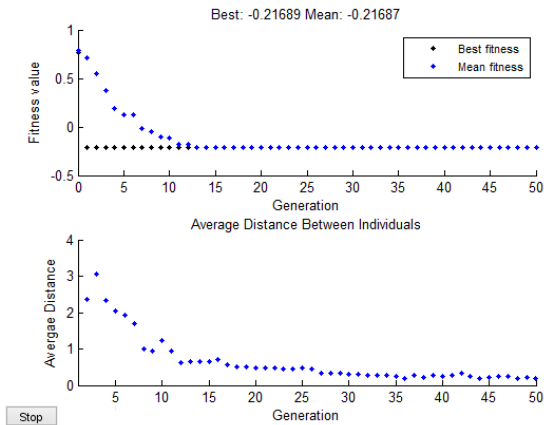


Fig. 12. Best fitness value for TH₁ output of circular trajectory and average distance between individuals.

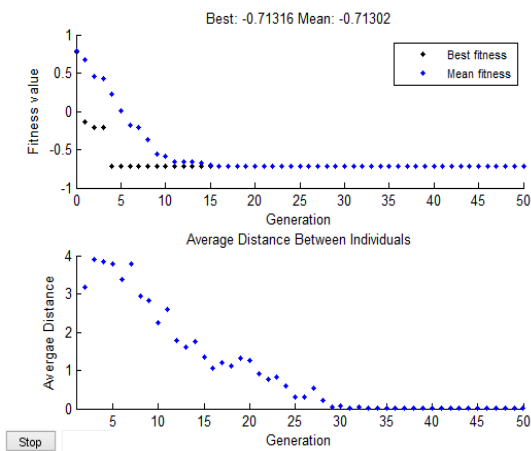


Fig. 13. Best fitness value for TH₂ output of circular trajectory and average distance between individuals.

6. Conclusions

In this paper the combining of GA and NN algorithm is presented in order to solve the problem of inverse kinematics for three link redundant robot. The inputs (x_E, y_E, TH_E) are end effector position and orientation of the three link robot. The Back Propagation Neural Network (BPNN) is used for training the data. The outputs of BPNN are computed then the MSE is evaluated between the estimated and desired outputs. The proposed fitness function is computed with GA. The continues GA is used in order to optimize the weights values with training the data in BPNN by combing GA and NN algorithm. This algorithm is presented in order to solve the inverse kinematics problem and to compute the best values of three joint angles. The continues GA is so fast with 50 generations. The best joint angles outputs results (TH₁, TH₂, TH₃) of sinwave and circular trajectories are evaluated from the combing GA and NN algorithm. In addition, the simulation of three link redundant robot workspace is presented.

Notation

- x_E, y_E position of end effector three link robot
- TH₁, TH₂, TH₃ three joint angles of three link robot
- TH_E orientation of three link robot
- x_p, y_p desired position of the end-effector of the and sinwave circle trajectories

x_c, y_c	coordinates circle's center
r	circle's radius
TH_d	desired orientation of the end-effector of the circle trajectory
ph_i	desired orientation of the end-effector of the sinwave trajectory
GA	Genetic Algorithm
NN	Neural Network
BPNN	Back Propagation Neural Network
MSE	Mean Square Error
m	meter
d_i	distance
θ_i	angle
a_i	length
α_i	twist angle
deg.	degree
rad.	radian

Greek Letters

φ	circle's angle
θ_i	joint angle

7. References

- [1] Bruno Siciliano, Oussama Khatib, *Springer Handbook of Roboti*c, Springer-Verlag Berlin Heidelberg, 2008.
- [2] Y. Feng, W. Yao-nan and Y. Yi-min, Inverse Kinematics Solution for Robot Manipulator based on Neural Network under Joint Subspace, INT J COMPUT COMMUN, ISSN 1841-9836, Vol.7, No. 3 September, pp. 459-472, 2012.
- [3] Raşit Köker, Cemil Oz, Tarık C, akarb and Hüseyin Ekiz, A study of neural network based inverse kinematics solution for a three-joint robot, ELSEVIER, Robotics and Autonomous Systems 49 -227–234, 2004.
- [4] Panchanand Jha, BB Biswal, A Neural Network Approach for Inverse Kinematic of a SCARA Manipulator, International Journal of Robotics and Automation (IJRA), Vol. 3, No. 1, ISSN: 2089-4856, March, 2014.
- [5] F.Y.C. Albert, S.P. Koh, C.P. Chen, S.K.Tiong and S.Y.S. Edwin, Optimizing Joint Angles of Robotic Manipulator using genetic Algorithm, IACSIT Press, Singapore, 8IPCSIT vol.2, 2011.
- [6] J. Ramirez A., and A. Rubiano F., Optimization of Inverse Kinematics of a 3R Robotic Manipulator using Genetic Algorithms, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering Vol:5, No:11, 2011.
- [7] Adrian-Vasile Duka, Neural network based inverse kinematics solution for trajectory tracking of a robotic arm, ELSEVIER, The 7th International Conference Interdisciplinarity in Engineering, 2014.
- [8] Adri`a Colom´, Smooth Inverse Kinematics Algorithms for Serial Redundant Robots, Master Thesis, Institut de Rob`otica i Inform`atica Industrial, 2011.
- [9] http://www.mathworks.com/help/pdf_doc/gads/gads_tb.pdf. Accessed on 1/08/2016.
- [10] Md. Mijanur Rahman, Tania Akter Setu, An Implementation for Combining Neural Networks and Genetic Algorithms, IJCST Vol. 6, Iss ue 3, July – Sept., 2015.

حل الكينماتيكا العكسية للروبوت المتعدد المناور باستخدام دمج GA و NN

هند زهير خليل

قسم هندسة السيطرة والنظم / الجامعة التكنولوجية / العراق / بغداد
البريد الإلكتروني: hhindzuhair@yahoo.com

الخلاصة

حساب الكينماتيكا العكسية هي مشكلة معقدة جداً للروبوت المتعدد المناور. هذه المقالة تقدم حلاً للكينماتيكا العكسية لأحد الروبوتات المناورة المتعددة (الروبوت الثلاثي) بواسطة دمج اثنين من الخوارزميات الذكية (الخوارزمية الجينية GA) و (الشبكة العصبية NN). المدخلات هي موقع واتجاه الروبوت الثلاثي الروابط. تدخل هذه المدخلات إلى الشبكات العصبية العكسية المنشورة (BPNN). الأوزان BPNN تعد الأفضل باستخدام الخوارزمية الجينية المستمرة GA. يتم احتساب (متوسط مربع الخطأ) MSE أيضاً بين المخرجات المقدرة والمطلوبة للزوايا المشتركة. في هذه المقالة، دالة الفتنس في GA تم اقتراحها. مسارات الموجة و الدائرية لنهاية طرف الروبوت ثلاثي الروابط المطلوبة وتم محاكاتها ببرنامج MATLAB. ان نتائج الزوايا المشتركة و مواقع نهاية الطرف للروبوت للمسار الدائري افضل من نتائج الزوايا المشتركة و مواقع نهاية الطرف للروبوت للمسار الدائري لعمل اخر حيث فقط في العمل الاخر تم استخدام NN. كما تم محاكاة مساحة الروبوت الثلاثي متعدد الروابط. ان نتائج المخرجات لافضل ثلاث زوايا تم حسابها للمسارين الموجي والدائري. ب ٥٠ جيل للخوارزمية فانها سريعة. تقدم هذه المقالة نتائج المحاكاة التي يتم الحصول عليها بالاعتماد على برنامج MATLAB R2010b.