Online Genetic-Fuzzy Forward Controller for a Robot Arm

Prof Dr. Abduladeem A. Ali
Department of Computer Engineering University of Basrsh Iraq.
(Email: abduladhem@compengbas.net)

Amal J. Kudaer
Department of computer Engineering University of Basrsh Iraq.
(Email: amaleng83@yahoo.com)

Abstract—The robot is a repeated task plant. The control of such a plant under parameter variations and load disturbances is one of the important problems. The aim of this work is to design Genetic-Fuzzy controller suitable for online applications to control single link rigid robot arm plant. The genetic-fuzzy online controller (forward controller) contains two parts, an identifier part and model reference controller part. The identification is based on forward identification technique. The proposed controller it tested in normal and load disturbance conditions.

Kywords: *Genitic fuzzy, Robot control, on line control, Model reference Control, Adaptive Fuzzy control*.

متحكم جيني مضبب مباشر وامامي على ذراع روبوت

أ.د. عبد العظيم عبد الكريم علي و أمال جبار خضير
قسم هندسة الحاسبات - كلية الهندسه – جامعة البصره – العراق

الخلاصه

يتميز الروبوت بأنه منضومه متكررة الاهداف. يعد التحكم في مثل هذه أنظمه عند تغيربرامترات المنظومه او تأثرها باظطرابات الاحمال من الدراسات الجديره بالاهتمام. يتناول هذا العمل تصميم منظومة تحكم تستعمل الخوارزميات الجينيه يمكن استخدامها بالوقت الحقيقي وتطبيقها على السيطره على ذراع روبوت ذو ارتباط احادي. يتكون هذا المسيطر والذي هو من نوع المسيطرات الاماميه من جزئين . الجزء الاول عباره عن كاشف امامي للبرامترات. أما الجزء الثاني فهو عباره عن مسيطر من نوع النموذج المتبوع. تم فحص المسيطر المقترح في حالات العمل الطبيعيه كما تم فحص تأثيرات أضطراب الاحمال.

1. Introduction

The problem of self-adjusting the parameters of the controller to compensate for the plant parameters variations and disturbances was the origin of adaptive systems [1]. Adaptive control system can be solved using fuzzy control technique by neural-fuzzy system [2] or genetic-fuzzy model [3]. Generally, in adaptive control system, identification is an essential part to determine, online, the disturbances through either explicit or implicit techniques. The Identification is the process of constructing a mathematical model of a dynamic system using experimental data from that system [3]. Identification is an important and integral part in the control of dynamic systems. There are two types of identification methods, the first is the forward identification, and the second is the inverse identification [4]. The problem of identification consists of setting up a suitably parameterized identification model and adjusting the parameters of the model to optimize a performance function. Two forms of identification are currently used in modern adaptive systems: The forward identification and inverse identification. Based on the form of identification used, there exist two techniques of control: The forward model following scheme and the inverse model following scheme. In this work the forward identification method is previewed with its simulation results. Genetic-fuzzy forward online control is structured with simulation results.

2. Dynamic Model of Robot Arm Manipulators

The type of robot arm model which is presented in this paper is a Single Link Rigid Robot Arm (SLRA) as shown in Fig. 1. Little works took the actuator dynamics and friction into consideration [5]. All these phenomena are considered in this work. The relation between the input voltage to the motor and position angle as follows:

$$\mathbf{V = C_3\theta^{\bullet\bullet\bullet} + C_2\theta^{\bullet\bullet} + C_1\theta^{\bullet} + C_0} \quad \mathbf{(1)}$$

Where:

$\mathbf{C_3 = K_1L_a(md^2 + J).}$

$\mathbf{C_2 = K_1R_a(md^2 + J) + K_1L_aB.}$

$\mathbf{C_1 = K_1L_amgd\cos(\theta) + K_1R_aB + K_e.}$

$\mathbf{C_0 = K_1R_amgd\sin(\theta).}$

V = The armature voltage.

$\mathbf{K_I}$ = Torque constant of the motor.

The parameters of this plant that we take in our work are as in Table I.

3. Genetic Algorithms

Simple Genetic Algorithm (SGA) is used here, where the entire population is generated for each generation on the basis of the previous generation [6, 7]. This kind of evolution resembles the evolution of population of fruit flies where a large part of the population is replaced simultaneously with new offspring. We will concentrate on the aspects of this approach:

∗ Representation:

In this work real-valued representation is used. This is for many reasons; first the values that deal with are real, then to prevent encoding of the floating point values with a binary encoding and this need more genes, second, the precision, and third, the processes of encoding and decoding take much time. Then, for these reasons the real-valued representation is more suitable for this problem.

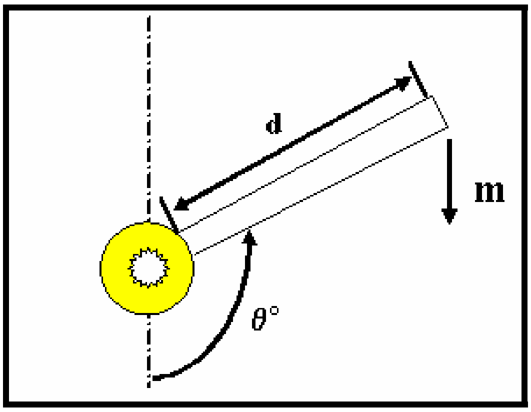

Fig.(1) Sketch of single link robot arm

∗ Initialization:

Producing the initial population of solutions is the second step in GAs, 50 individuals with real values is generated; each individual has center, width, and center of outputs for fuzzy system. This number (50 individuals) remained constant for each generation.

∗ Fitness Evaluation:

The fitness function is very important in order to develop a good GAFL approach. The fitness function that is used:

$$\mathit{Fitness}\,\text{Function}(\text{E}) = \frac{1}{2}\sum_{k=0}^{M}[y_m(k) - y_r(k)]^2 \quad (2)$$

This fitness function is called mean square error function, where $y_r(k)$ is the output of reference model and $y_p(k)$ is the plant output.

∗ Selection operators:

A Roulette Wheel Selection (RWS) is used. It is used to select those parents that have a higher fitness with a higher probability [8]. This method is used usually with SGA. There is an important point in this method that may be one individual which has a much higher fitness than all the other individuals. This highly fit individual could very quickly dominate the whole population (premature convergence) since it would be chosen for selection extremely often [9]. In our work we select 10 individuals from the initial population for next generation and to make crossover on these individuals, then from selection and crossover and mutation the population remained constant for each generation.

∗ Recombination operators:

Because the used of real encoding (i.e.), the parameters that use in the GA remained as real valued), then one of the real-valued crossovers is used which is called intermediate crossover. This method only applicable to real variables and it can be performed many ways. One of that is offspring are produced in form:

$$\textbf{Offspring}_1 = \textbf{Parent}_1 + \alpha_1(\textbf{Parent}_2 - \textbf{Parent}_1) \quad \textbf{(3)}$$
$$\textbf{Offspring}_2 = \textbf{Parent}_1 + \alpha_2(\textbf{Parent}_2 - \textbf{Parent}_1) \quad \textbf{(4)}$$

α_1 , α_2 are generated randomly for each generation between [-1.1,1.1]. The above rules used to each parameter that need optimization in fuzzy system.

∗ Replacement (Reinsertion):
Weak parent replacement is used. A new child replaces the weaker parent, 40 weaker parents is replaced with 40 child come from the recombination and mutation operators.

∗ Termination:
After several generations the best individuals of the population are then tested to determine if they satisfy the problem. The problem in this work is the tracking between the desired and the output curves or may be the tracking between the input or output of the plant with the output of identification model.

4. Genetic-Fuzzy Systems

The defuzzyfier formula:

$$f_j(\bar{x}) = \frac{\sum_{L=1}^{R} y_j^L \times \mu_{B^L}(\bar{x})}{\sum_{L=1}^{R} \mu_{B^L}(\bar{x})} \qquad (5)$$

This defuzzyfier formula is called "Center of Gravity ".
Where:

R =The number of rules.
j =The number of output.
$\bar{x}$ =The total input.
y_j^L = The center of output

And

$$\mu_{B^L}(\bar{x}) = \left[\prod_{i=1}^{n} \mu_{F_i^L}(x_i)\right]$$

F_i^L = Denote linguistic values defined by fuzzy set.
n = The number of inputs.

Because the triangle membership function is simple, it is used in this work. The mathematical eqn. for this membership function is:

$$\mu(x) = \begin{cases} \left[|x-c| * \frac{-2}{w}\right] + 1 & \text{if } c - \frac{w}{2} \le x \le c + \frac{w}{2} \\ \text{Zero} & \text{otherwise} \end{cases} \qquad (6)$$

Where:
c = The center of the membership
w = The width.

When this membership function is used and substituted in eqn. (5), the following formula is found:

$$f(\bar{x}) = \frac{\sum_{L=1}^{R} \bar{y}^L \times \left[\prod_{i=1}^{n}\left[\left[|x_i - c_i^L| \times \frac{-2}{w_i^L}\right] + 1\right]\right]}{\sum_{L=1}^{R}\left[\prod_{i=1}^{n}\left[\left[|x_i - c_i^L| \times \frac{-2}{w_i^L}\right] + 1\right]\right]} \qquad (7)$$

Where:
c_i^L = The center of triangle membership function at input i and rule.
w_i^L = The width of triangle membership function at input i and rule.
$\bar{y}^L$ = The center of the output at rule L.

The c_i^L, w_i^L , and $\bar{y}^L$ are the fuzzy parameters that have to be found using GA.

One can generalize eqn. (7) for each chromosome as:

$$f_q(\bar{x})=\frac{\sum_{L=1}^{R} y_q^L \times \left[\prod_{i=1}^{n}\left[\left|x_i-c_{iq}^L\right|\times\frac{-2}{w_{iq}^L}\right]+1\right]}{\sum_{L=1}^{R}\left[\prod_{i=1}^{n}\left[\left|x_i-c_{iq}^L\right|\times\frac{-2}{w_{iq}^L}\right]+1\right]} \quad (8)$$

Where:
q = Refer to sequence of the chromosome.

5. Series-Parallel Identification Model

The Series-Parallel model is obtained by feeding back the past values of the plant output as shown in Fig.(2). This implies that in this case the identification model has the form:

$$\hat{y}_p(k+1)=f[y_p(k),...,y_p(k-m+1), u(k),...,u(k-n+1)] \quad (9)$$

The Series-Parallel model has several advantages over that of parallel model [10]. As in the Fig.(2) the input to the plant denoted by u(k) and the plant output by $y_p(k)$.

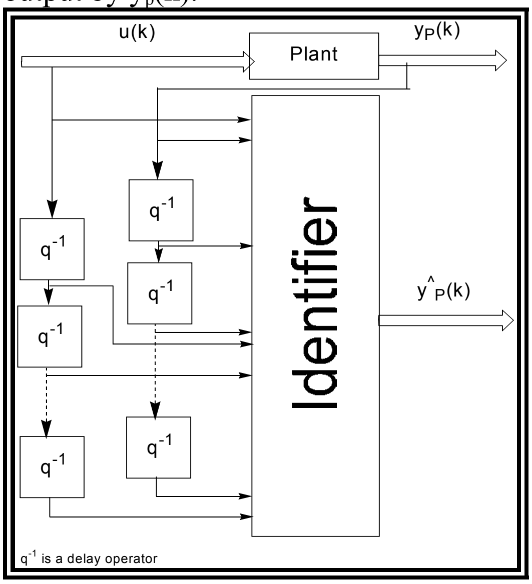

Fig.(2) Series-Parallel identification model

6. Robot Arm Identification System

It is well known that the response of a nonlinear plant like robot arm generally can not be shaped into a desired pattern using a linear controller. Consequently, a nonlinear controller is required to satisfactorily the control of such plants.

Nonlinear controller design may be viewed as a nonlinear function approximation problem [11]. Two concepts can specify the identifier structure: the type of identifier and the plant dynamics. In this paper genetic-fuzzy identifier is used as forward identifier. But after then the forward identifier will be used for control applications.

* Forward Identification:

The series parallel structure of this identifier suggest that the inputs to the identifier should be the delayed inputs and outputs of the plant, while the output of the identifier should be the same as the output of the plant at the end of the training process the training process suggests calculating the error between the output of the plant and the identifier, and the plant model supposed to be third order (as explained in a single link rigid robot arm model).

The aim of the identifier is to have the plant transfer function, and the error between plant output and identifier output must be minimum.

Rewriting Eqn.(9) to express the forward plant model:

$$\hat{y}_p(k)=f[y_p(k-1),y_p(k-2),y_p(k-3), u(k-1),u(k-2),u(k-3)] \quad (10)$$

the structure of the forward identification is illustrated in Fig. (3)

The aim of the forward identifier is to constrict the non-linearity **f** in Eq.(10) using genetic fuzzy approximation such that the identifier output $\hat{y}_p(k)$ approaches the plant output $y_p(k)$.

A population of identifier equations is generated as in Eq.(8) for the non-linearity approximation, with ($y_p(k)$, $y_p(k-1)$, …., $u(k)$, $u(k-1)$, ….) represent the inputs x_i (i=1,…..,n) of Eq.(8). The optimal chromosome is selected as the identifier output

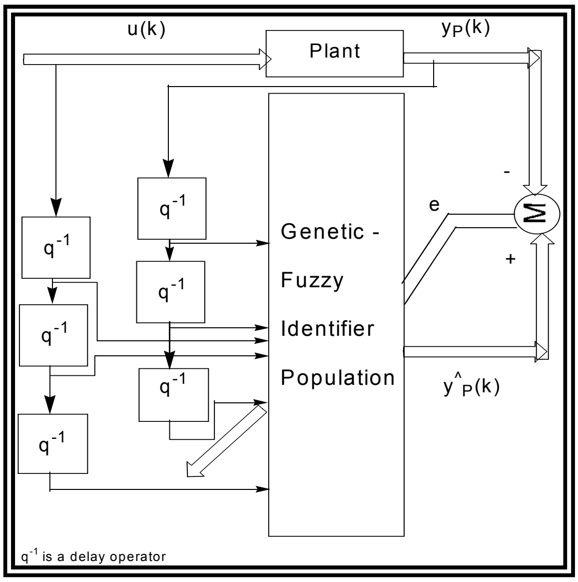

Fig.(3) Series – Parallel model forward Genetic – Fuzzy Identifier

7. Genetic-Fuzzy Forward Control Scheme

The suggested genetic – fuzzy forward control is shown in Fig. (4). This system is based on the model Reference Adaptive Control systems (MRAC). This structure may be described briefly as follow:

*It contains two parts, an identifier part and genetic model reference controller part.

*Both two parts receive the same input, which represent the desired position.

*The identified model (best chromosome for the identifier) at each time in the identification part is taken to be the plant model in the control part

*The best chromosome for the controller at each time in is taken to be the controller to the plant (robot) in this case. The developed identifier has 50 chromosomes to construct the population. The optimal chromosome is the chromosome that has the minimum fitness function.

In the identifier part, the forward identifier model can be given as:

$$\mathbf{y_{id}(k) = f[y_p(k-1), y_p(k-2), y_p(k-3), u(k-1), u(k-2), u(k-3)]} \quad \mathbf{(11)}$$

Where y_{id} is the identifier output.

In the model reference controller part, the robot model which represents forward identifier will uses the optimal parameters from the identifier in the identifier part to structure the optional identified model, as given below:

$$\mathbf{y_{mop}(\bar{x}) = \frac{\sum_{L=1}^{R} y_{op}^{L} \times \left[\prod_{i=1}^{n}\left[\left|x_i - c_{iop}^{L}\right| \times \frac{-2}{w_{iop}^{L}}\right] + 1\right]}{\sum_{L=1}^{R}\left[\prod_{i=1}^{n}\left[\left|x_i - c_{iop}^{L}\right| \times \frac{-2}{w_{iop}^{L}}\right] + 1\right]}} \quad \mathbf{(12)}$$

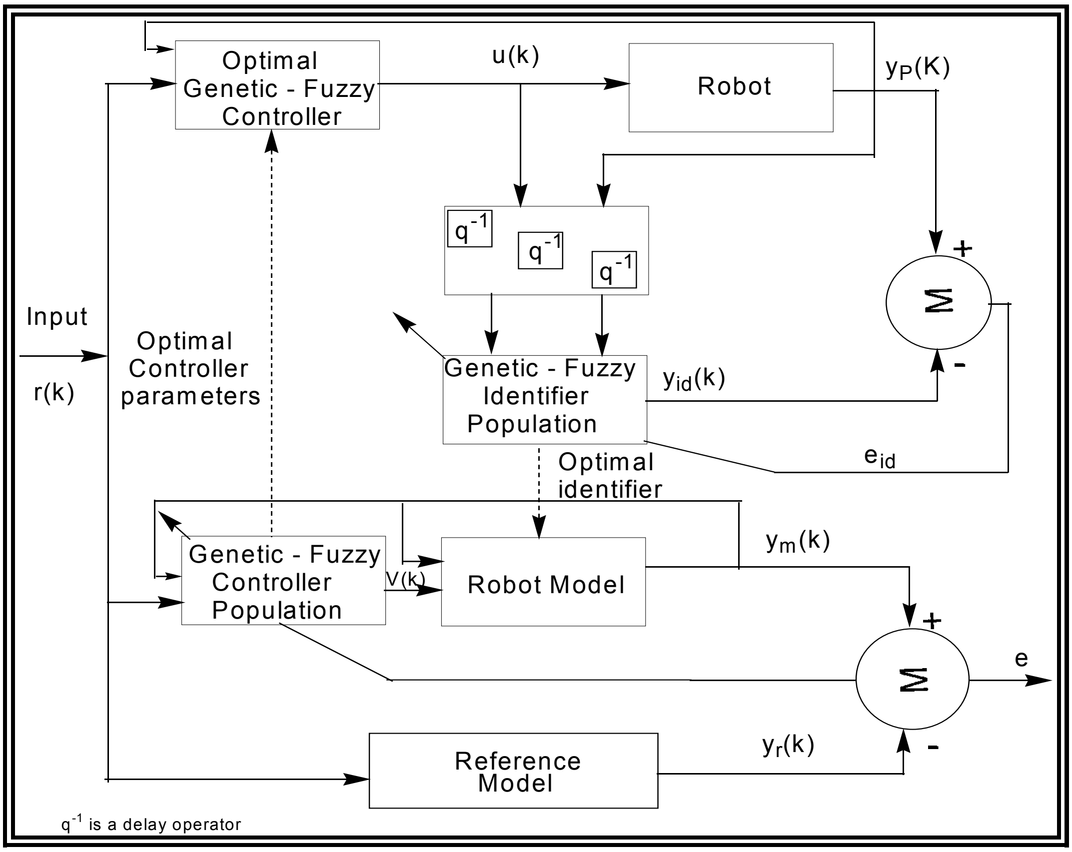

Fig. (4) Genetic – Fuzzy Forward Control Block Diagram

Where:

$\mathbf{y}_{\mathbf{mop}}(\overline{\mathbf{x}})$=The defuzzyfied output for robot model in the optimal chromosome.

R = The number of rules.

n = The number of input.

$\mathbf{c}^{\mathbf{L}}_{\mathbf{iop}}$ = The center of triangle membership function at input i and rule L for optimal chromosome.

$\mathbf{w}^{\mathbf{L}}_{\mathbf{iop}}$ = The width of triangle membership function at input i and rule L for optimal chromosome.

$\mathbf{y}^{\mathbf{L}}_{\mathbf{op}}$ =The center of output at the optimal chromosome.

The genetic – fuzzy controller in the model reference part will uses the parameter of the optimal identifier to structure a forward controller using the following control algorithm:

$$\mathbf{v(k)=f[y_m(k-1),y_m(k-2),y_m(k-3), r(k-1),r(k-2),r(k-3)]} \quad \textbf{(13)}$$

Where v(k) is the controller output.

To approximate the non-linearity of Eq.(13), a population is selected for the controller with fuzzy structure as follows:

$$\mathbf{y_{qc}(\overline{x})=\frac{\sum_{L=1}^{R} y^{L}_{qc}\times\left[\prod_{i=1}^{n}\left[\left|x_i-c^{L}_{iqc}\right|\times\frac{-2}{w^{L}_{iqc}}\right]+1\right]}{\sum_{L=1}^{R}\left[\prod_{i=1}^{n}\left[\left|x_i-c^{L}_{iqc}\right|\times\frac{-2}{w^{L}_{iqc}}\right]+1\right]}} \quad \textbf{(14)}$$

Where:
$y_{qc}(\bar{x})$=The defuzzyfied output for controller.
R = The number of rules.
n = The number of input.
c_{iqc}^{L} = The center of triangle membership function at input i and rule L.
w_{iqc}^{L} = The width of triangle membership function at input i and rule L.
y_{qc}^{L} =The center of output.

The best chromosome for the controller (optimal fuzzy controller) is selected at each sampling instant and used to generate the actual control output **u(k)** to the robot as:

$$\begin{aligned}u(k) = f[&y_p(k-1), y_p(k-2), y_p(k-3),\\ &r(k-1), r(k-2), r(k-3)]\end{aligned} \quad (15)$$

8. Simulation Results

In order to demonstrate the validity of genetic algorithm as an identification system, a single link robot arm identification scheme is simulated, and the parameters in the Eqn.(1) for this robot arm are given in Table I.
The genetic algorithm employed for the controller is similar to that employed for the identifier except for the fitness function. The fitness function used is the average of the sum error square for the error between the robot output $y_p(k)$ and the identifier output $\hat{y}_p(k)$ as in Fig.(3). The initial values for fuzzy system are selected to be 60 equally spaced membership functions as shown in Fig.(5). The identification process is done on the input to plant as shown in the initial responses in Fig.(6), after (8) generations. The final responses (after 895 generations) are shown as in Fig.(7).

Table 1. Used Robot Parameters.

Parameter	Value	Unit
d : Link length	1	m
m : Initial total mass	2	kg
Ra : Armature resistance	1	Ω
La : Armature inductance	1	H
Ke : Electric motive force constant	1	V.sec/m
K_I : Torque constant of the motor	1	A/N.m
J : Rotor inertia	0.5	Kg .m^2

Note: The parameters of the motor is refered to unity gear ratio. Coresponding parameters have to be obtained according to the used gear ratio.

From this response we observed the fast tracking for this input, and the identifier model tries to be identical to the plant model by finding the optimal fuzzy system parameters through the genetic algorithm. Also the resultant distribution of the membership functions for the optimal chromosome as in Fig.(8), from this figure we can conclude that the membership functions are reduced to less number with different width and center.

The control system is shown in Fig.(4), the reference input to the control system is a step function limited between (1 and –1). When the system is suffered from disturbances, the parameters of control block could not be able to control the output and an error appears between plant output and model reference output. In this case the identifier try to adjust itself to identify the disturbances effect. Total initial mass is selected to be (2kg), and then we make some disturbances to the system by lifting some mass from the initial mass and then adding another mass to the initial mass, in the case of initial mass (m=2) one can see a good following from the robot model output to the reference model output, as shown in Fig.(9). When applying the first disturbance (lift some mass from the initial mass (m=1)), one can observe initially a bad tracking between the robot model output and the reference model output as shown in Fig.(10), then when the identifier start to identify the disturbance, the response after convergence is shown in Fig.(11). Also when applying the second disturbance (adding another mass to the initial mass (m=3)), one can see the initial response in Fig.(12), and then after convergence the finial response is in Fig.(13). The finial distribution for the membership functions for optimal chromosome is as in Fig.(14).

9. Conclusions

In this work, genetic-fuzzy online forward controller is structured. The plant parameters of are identified by forward genetic fuzzy identification model and then used by the controller to control this plant.

The identification process is continuing in the normal conditions or disturbance conditions. The work identifier in the normal conditions to make the tracking is more accurate as well as for the small variations in some plant parameters.

The suddenly changing in the load on the robot arm at it is work make the arm diverge from a suitable path, therefore, it is need number of iterations (that depend on the value of load) that make the controller control the system and it converges to accurate path.

REFERENCES

[1] K. J. Astrom, B. Wittenmark, “Adaptive Control”, Addison Wesley, 1989.

[2] T. L. Seng, M. Khalid, R. Yusof, S. Omatu, “Adaptive Neuro-Fuzzy Control System by RBF and GRNN Neural Networks”, Intelligent and Robotic System Journal, vol. 23, PP.267-289, 1998.

[3] K. M. Passino, S. Yurkovich, “Fuzzy Control”, Addison Wesley Longman, Inc., 1998.

[4] J. G. Kuschewski, S. Hui, S. H. Zak, “Application of Feedforward Neural Networks to Dynamical System Identification and Control”, IEEE Transaction on Control System Technology, vol.1, No.1, March 1993.

[5] J. N. Abdulbaqi, “Neuro-Fuzzy Control of Robot Arm”, M.Sc. Thesis, Department of Electrical Engineering, Basrah University, 2004.

[6] J. M. Herrero, X. Blasco, M. Martinez, J. V. Salcedo, “Optimal PID Tuning with Genetic Algorithms for Nonlinear Process Models”, 15th Triennial World Congress, Barcelona, Spain, 2002.

[7] M. Mitchell, “An Introduction to Genetic Algorithms”, Aisradford Book, The MIT press, Cambridge, Massachusetts, London, England, 1998.

[8] H. A. Younis, “Attacking Stream Cipher Systems Using Genetic Algorithm”, M.Sc. Thesis, Department of Computer Science, Basrah University, 2000.

[9] M. Schmidt, T. Stidsen, “Hybrid Systems: Genetic Algorithms, Neural Networks, and Fuzzy Logic”, Aarhus University, Denmark, 1996.

[10] G. Lightbody, G. W. Irwin, “Nonlinear Control Structures Based on Embedded Neural System Models”, IEEE Transactions on Neural Networks, vol.8, No.3, May 1997.

[11] M. S. Ahmed, “Neural-Net-Based Direct Adaptive Control for a Class of Nonlinear Plants”, IEEE Transactions on Automatic Control, vol.45, No.1, January 2000.

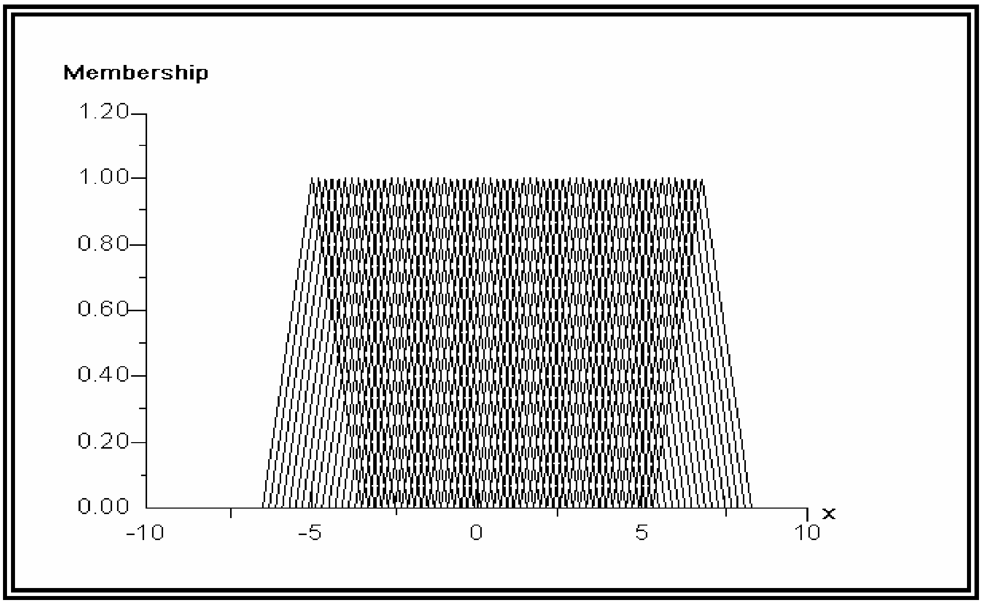

Fig. (5) Initial membership functions/ Forward Identifier

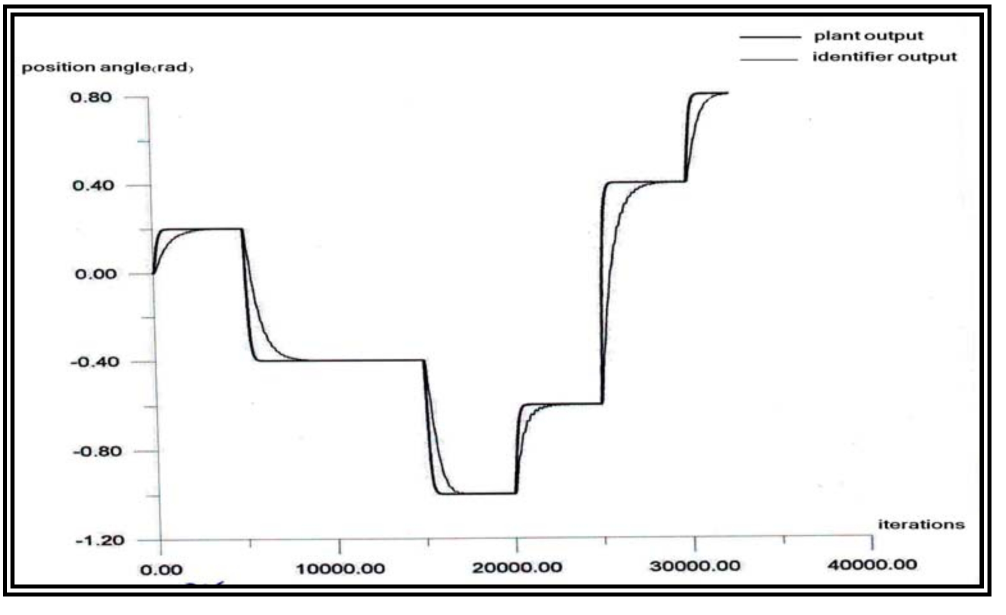

Fig. (6) Position angle for the first shape after 8 generations / Forward Identifier

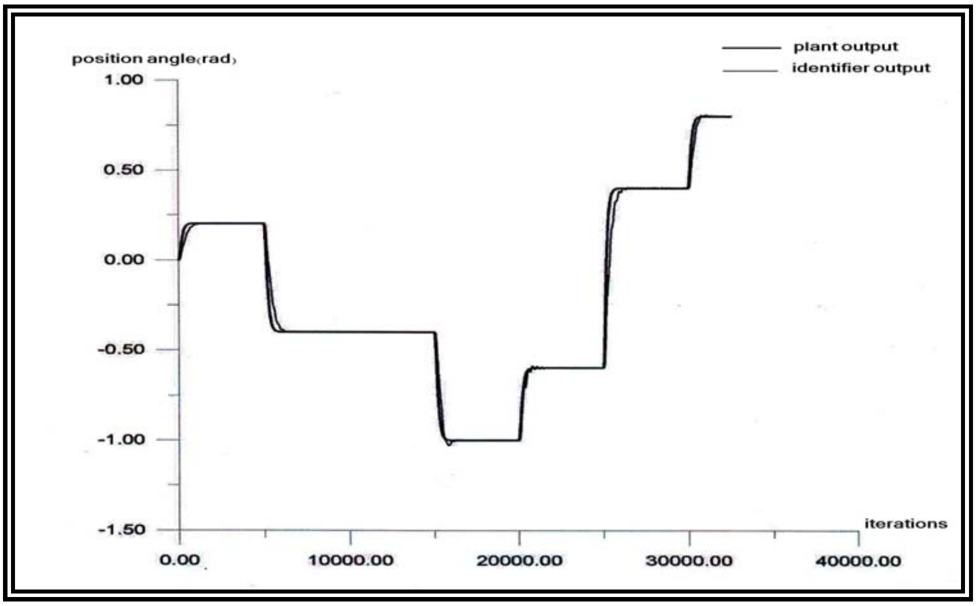

Fig. (7) position angle for the first shape after 895 generations /Forward Identifier

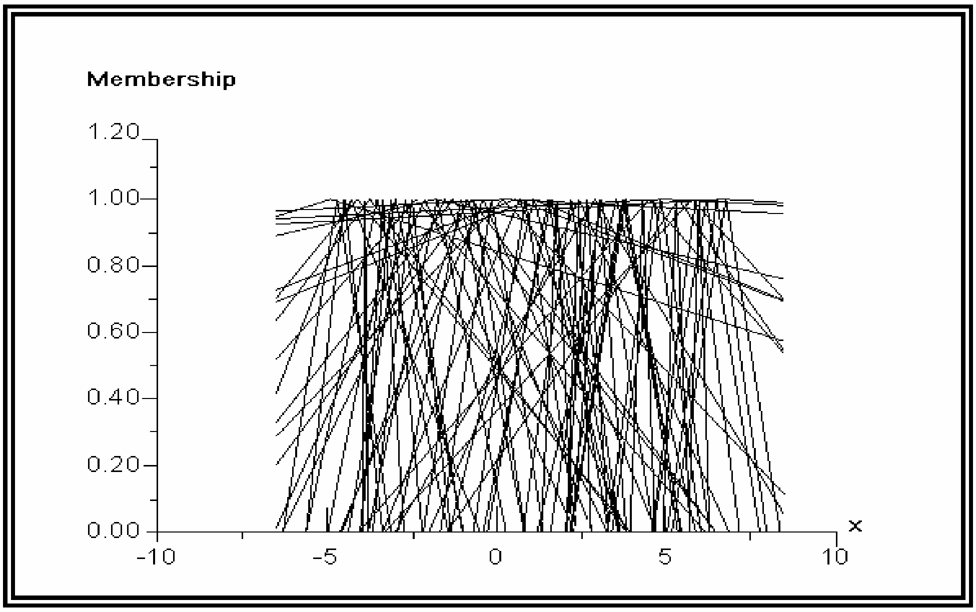

Fig. (8) Membership functions after the identification process/ Forward Identifier

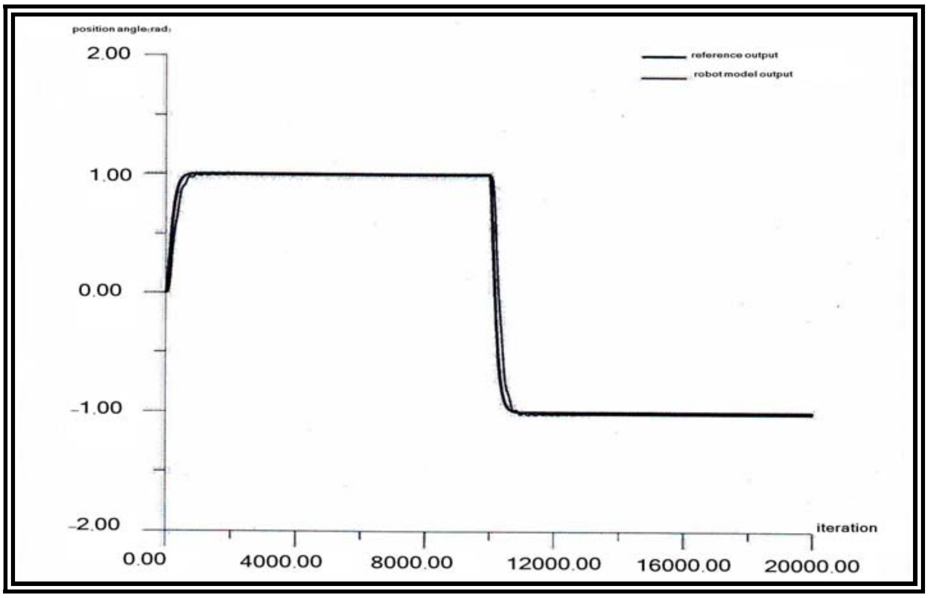

Fig. (9) Position angle at mass load = 2 / Forward Controller

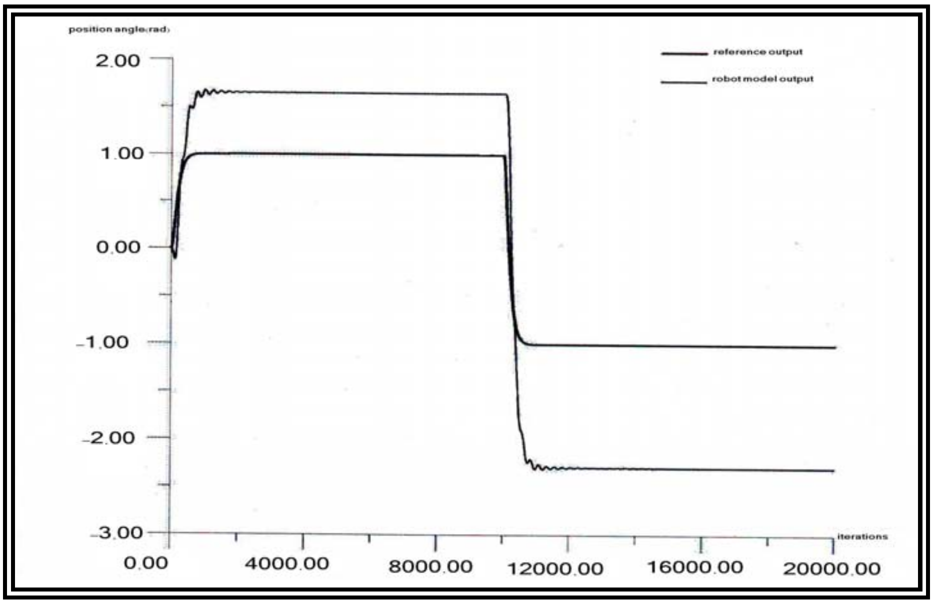

Fig. (10) Position angle at mass load = 1 (immediately after application of load) / Forward Controller

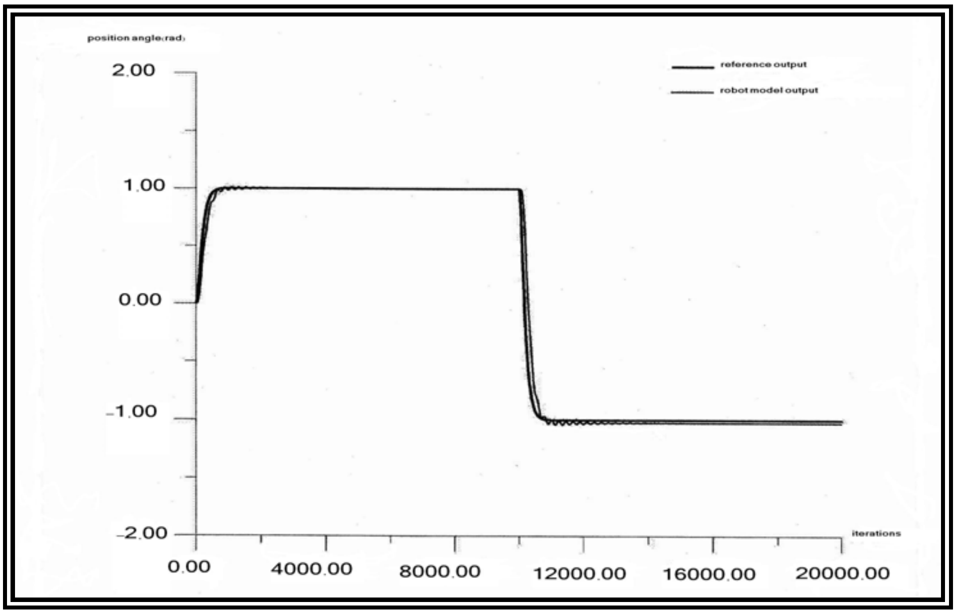

Fig. (11) Position angle at mass load = 1(after 5 generations) / Forward Controller

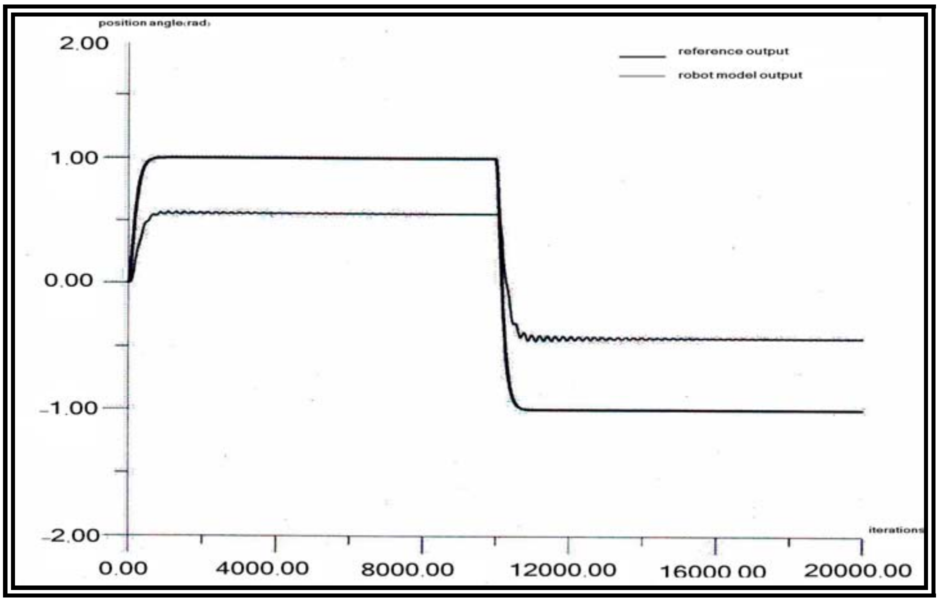

Fig. (12) Position angle at mass load = 3(immediately after application of load) / Forward Controller

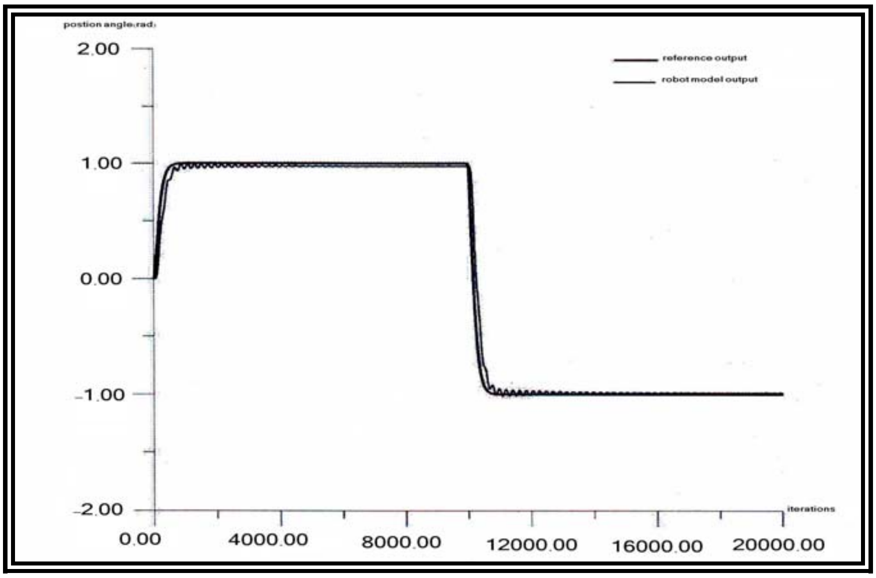

Fig. (13) Position angle at mass load = 3 (after 6 generations) / Forward Controller

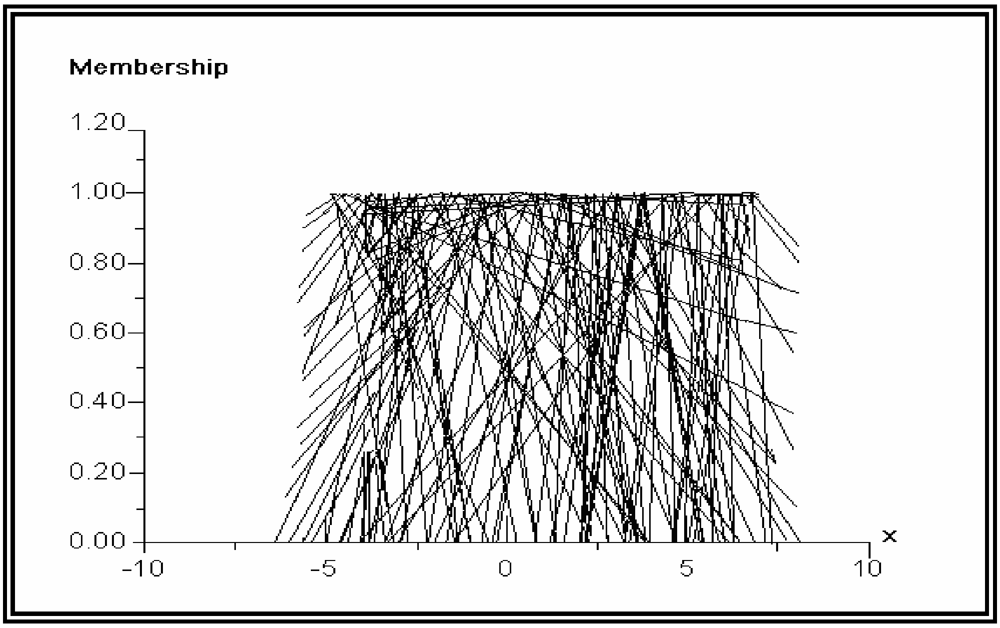

Fig. (14) Membership functions after convergence process (m=2)/
Forward Controller