

# Robot Arm Control Using Hybrid intelligent

TWFIQ H. ELMENFY  
Department of Electrical  
Engineering,  
Faculty of Engineering  
University of Benghazi, LIBYA

MONA MOHAMMED MOSA  
Aljeagdif  
Department of Medical Engineering  
Collage of Medical Technology  
Benghazi, LIBYA

SAMAH ABDELSALAM  
Higher Institute for Technology  
and Science, Regdalleen,  
LIBYA

**Abstract:** The design of a fuzzy logic control suffers from select parameters of the membership functions, scaling factors, defuzzification action, Inference engine and base rules. speaking generally, such prosecutions are executed by traditionally techniques which do not assure an robust fuzzy control system design. There are various techniques introduced in literatures that used Genetic Algorithms to optimize a fuzzy logic control component. . In this paper, the suggested control law consists of Fuzzy Logic Control (FLC) tuning via Genetic Algorithm (GA). The FLC used because it is efficient tools for control of nonlinear and uncertain parameters systems. GA is mainly presented to find a simultaneous near optimum design of the membership functions, scaling factors, defuzzification Method, Inference engine and control base rules. GA with different fitness functions in a form of the cumulative response error which are widely used as an efficient optimization technique. This paper also introduce a new methodology with new multi-objective function to improve fuzzy control parameters based on Genetic Algorithm techniques. The dynamic model of the robot manipulator its done by differential equations, these equations are hardly nonlinear, parameters uncertainty and time varying with multiple input and multiple output (MIMO).The manipulator robot and the fuzzy genetic control are modeled in MATLAB SIMULINK; the manipulator robot model driven nonlinear controller to draw a circle in the space with and without parameters uncertainties. The proposed techniques showed that the proposed fuzzy controller gives superior response in the output performance. When the parameter uncertainties including in the system, given satisfactory response.

**Keywords:** Genetic optimization, PUMA 560 Robot Manipulator, MATLAB SIMULINK fuzzy logic

Received: June 21, 2022. Revised: September 2, 2023. Accepted: October 4, 2023. Published: November 2, 2023.

## 1. Introduction

Intelligent control technologies provide solutions to traditional approaches by leveraging insights from intelligent biological systems. Such ideas can either come from experts, to solve an automated control problem or by looking at how a biological system works and translating it into techniques suitable in solving control problems [1].

In [2] a process system is intelligent if it is able to improve its response in case of uncertainty. The main characteristics of intelligence are learning or adaptation and past experiences in general, improvement and error development and self-searching for the most effective values

Which achieves the machine's ability to check and modify its behavior in a limited sense using the following methods:

### 1.1 Artificial neural networks (ANN)

ANN are computational models that attempt to make mathematical representations of the structure of the brain, consisting of a set of elements to perform simple manipulations that communicate through weighted connections [3].

The ANN are characterized via its patterns between the neurons which working as processing units, its method of determining the weights on the connections (called its training) by back propagations, and its activation function .

### 1.2 Fuzzy logic system (FLS)

In 1965, Zadeh introduced his research on the fuzzy logic system, and Zadeh was the first designer to use fuzzy group theory (FS) and fuzzy logic (FL), and he previously presented the concept of the language variable in 1973. Zadeh also proposed fuzzy control system by contrast, Conventional control may take inputs/outputs between false and true as a percentage, in traditional set theory the function definition

does not matter but the number belongs to the set or not, yes or no and also zero or one takes on the value, this approach is inconvenient in many applications practical and life like lifetime group or temperature group, but element in fuzzy group has variable values between minimum(0) and maximum (1), that declare that elements of these sets not only represent (1) or (0) values but also represent the percentage of each input between these two values.

Among the many applications of fuzzy systems has been the area of FLCs. FLC is a rule base system that acts as a closed-loop control with its brain and is useful in executing complex, poorly defined operations, particularly those that can be controlled by an experienced human operator without knowledge of underlying system dynamics. Fuzzy control has recently been applied to many fields of industrial applications

FLCs have been used in nonlinear control systems with unpredictable range of operating points. The block diagram of a elements of fuzzy control is showed in Figure 1.

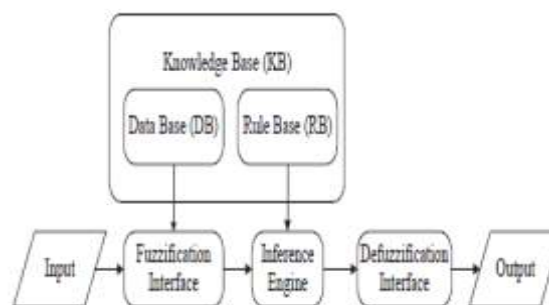


Fig. 1. Basic concept of fuzzy logic control.

The fuzzy logic control is consisting of the following four elements:

a) *A fuzzification* :a fuzzification term that converts real-world crisp values into fuzzy sets through membership functions.

b) *Rule base (RB)*: The rule base is composed of IF-THEN rules, from which an inference engine is created.

A standard expression of RB with rule bases are represented as:

Rule m: IF  $x_1 = A_{m1}$  and  $x_2 = A_{m2}$   $x_n = A_{mn}$  THEN  $y = B_m$

where  $x_1, \dots, x_n$  are input variables and  $y$  is control action.  $A_{i1}, \dots, A_{mn}$  and  $B_i (i = 1, \dots, m)$  are, respectively, the linguistic variables for  $x_1, \dots, x_n$  and  $y$  in the universal of discourse of  $U_1, \dots, U_n$ .

c) *Data base (DB)*: The data base is created via the appropriate membership functions of linguistic variables  $A_{i1}, \dots, A_{mn}$  and  $B_i$  that convert crisp inputs into fuzzy sets.

Bell-shaped, triangle and trapezoidal membership functions are used.

d) *Inference engine*: Operators within rule base form the inference mechanism. In general, the rules of the AND rules (with the lowest value) or OR (with the maximum value) are used as connecting operators between the case variables.

e) Defuzzification for making a output of FLC action, defuzzification is the calculating the output from inference results for all activated logic rules into an explicit output. Selecting of defuzzification method are some effect on control action of FLC.

### 1.3 Genetic algorithms (GAs)

Referencing Darwin's evolutionary theory, computer scientist John Holland proposed genetic algorithms in his paper on a general method for creating programmed solutions through evolutionary adaptive processes (Park, 1995). A GA is a global optimization method used to solve problems for optimal performance. The GA process in Fig. 2 consists of developing an initial set of points, that is, an initial set of randomly generated individuals (chromosomes), and coding solutions to an optimization technique to going through appropriate final solutions. This process is carried out by repeating a large ensemble of fixed size N. The evolution of successive ensembles from a  $k^{th}$  generation to a  $[(k + 1)]^{th}$ , the generation depends on the processes of selection, crossover and mutation, inspired by evolutionary biology:

*Selection*: A population is selected from each successive generation to raise a new generation. There are two methods of selection that are most commonly used and the most common is to choose the roulette wheel and the random rest without choosing an alternative.

- Crossover*: The replication crossover operator produces good generations of parents selected by a computer probabilistic method from among the remaining surviving individuals.
- Mutation*: This process denotes changes in the genetic sequence (mutation) with a probability of  $P_m$  in some selected chromosomes to break out of the local optima and confirm the accessibility of the overall solution space..

These processes working on all members of the  $k^{th}$  generation. For each generation, GA selects the best individuals according to adaptability or suitability which is an index function

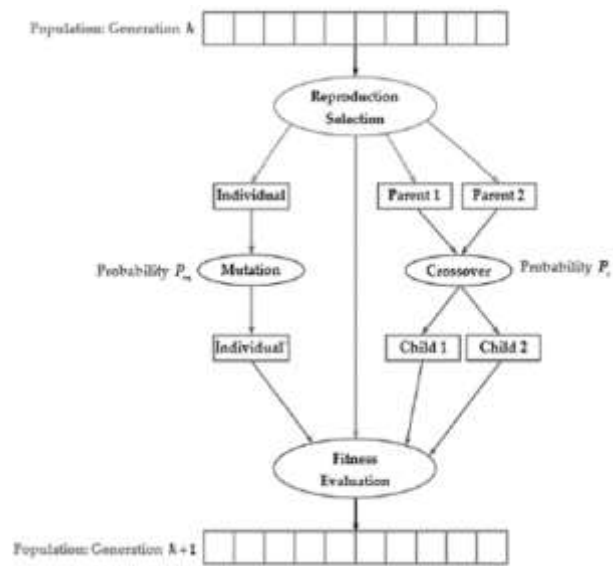


Fig. 2. Principle of a genetic algorithm.

- Intelligent hybrid systems are a combination of two or more computing paradigms, constructed in such a way that the advantages of one method are used to compensate for the disadvantages of another. some techniques or methods
- They are integrated as follows: Neural networks that used to designing fuzzy logic.
- Fuzzy systems which used to designing variuos neural networks.
- Genetic algorithms which used to designing of fuzzy logic systems and selected his parameters.

Genetic algorithms in automatically training and generating neural networks.

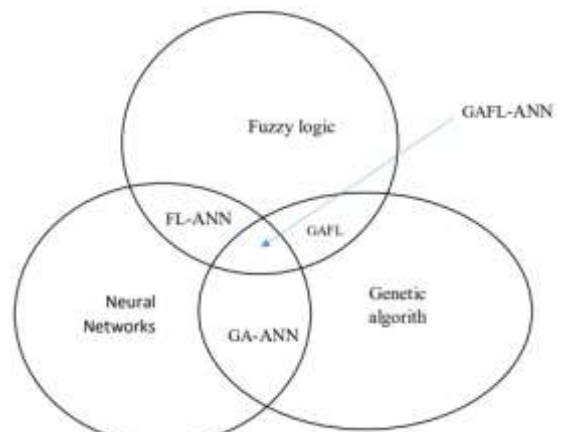


Fig.3. Intelligent hybrid systems.

GFLC are one of the most successful hybrid algorithms. But GFLC recently. It is popular and has gained a lot of interest in recent studies. Some of them have been mentioned previously, and it is worth noting that there is no agreed-upon method between the two areas, and we have incorporated a

special method for Park, Foran, Byrne and Loudini in order for us to finally arrive at the complete GFS model. A method will be explained in details in the next Section.

In our research, we present the application of GFLC method to design control algorithm using Simulink non-linear robot 3 DOF PUMA 560 which is short for (Three Degree of Freedom Programmable Universal Manipulator for Assembly) which was released in the past as the first modern application of robot and became very common, it is an industrial robot My application, this robot has 6 degrees of freedom with 6 rotational joints, Puma 560 was used for research in the past and it was a very popular laboratory robot, we used a lot in research because they are well studied and they are well known parameters that were described as "white mice" in research that studied bots [4].

## 2. Mathematical Modelling of the Puma Robot

All the parameters and the hardly nonlinear model used Langrange Euler formulation to derive this model of the 6 DOF PUMA 560 used in the research can be found in [5] The 3DOF PUMA robot has the same configuration space equation general form as its 6DOF convenient. In this sort, the last three joints are blocked so they keep their initial states and parameters while the robot is moving.

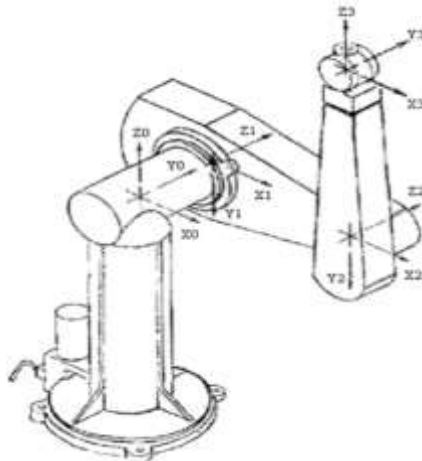


Fig.4. Debavit and Hartenberg Coordinate frames of the 3DOF PUMA Robot .

Using the formation equation of the robot, and by appointed the last joints as zero always, we can define a general equations that allows us to use PUMA robot as a 3 DOF robot.

To find the kinematics of the 3-DOF robot, a new Debavit and Hartenberg DH coordinate system is established, and a homogenous transformation matrix relating the coordinate frame to the first coordinate frame is developed. However, the 3 DOF PUMA will have the same kinematics of its 6-DOF convenient with  $q_4$ ,  $q_5$  and  $q_6$  to zero.

1. For the formation space equation of the robot:

$$\Gamma = M(q).\ddot{q} + B(q).[\dot{q} \dot{q}] C(q).[\dot{q}^2] + G(q) \quad (1)$$

Where,

$\Gamma$  : nx1 torques vector .

$q$  : nx1 position vector ,

$M(q)$  : nxn inertia matrix of the manipulator,

$C(q)$  is coriolis torques matrix

$G(q)$  is centrifugal torque matrix

$[\dot{q} \dot{q}]$  is joint velocity vector that it can give by:  $[\dot{q}_1 \dot{q}_2, \dot{q}_1 \dot{q}_3, \dots, \dot{q}_1 \dot{q}_n, \dot{q}_2 \dot{q}_3, \dots, \dots]^T$ .

$[\dot{q}^2]$  is vector, that it can given by:  $[\dot{q}_1^2, \dot{q}_2^2, \dot{q}_3^2, \dots]^T$ .

We set  $q_4 = q_5 = q_6 = 0$  this yields

$$[\ddot{q}] = [\ddot{q}_1 \dots \ddot{q}_2 \dots \ddot{q}_3 \dots 0 \dots 0 \dots 0]^T$$

$$[\dot{q} \dot{q}] =$$

$$[\dot{q}_1 \dot{q}_2 \dots \dot{q}_1 \dot{q}_3 \dots 0 \dots 0 \dots 0 \dots \dot{q}_2 \dot{q}_3 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0]^T$$

$$[\dot{q}^2] = [\dot{q}_1^2 \dots \dot{q}_2^2 \dots 0 \dots 0 \dots 0]^T$$

$$B(q).[\dot{q} \dot{q}] = [b_{112} \cdot \dot{q}_1 \dot{q}_2 + b_{113} \cdot \dot{q}_1 \dot{q}_3 + b_{123} \cdot \dot{q}_2 \dot{q}_3 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0 \dots 0]^T$$

And

The angular acceleration is found as to be

$$\text{Now let } I = \{ \Gamma - [B(q).[\dot{q} \dot{q}] C(q).[\dot{q}^2] + G(q)] \}$$

$$\ddot{q} = M^{-1}(q).I$$

$$(2.15)$$

$$I_1 = \Gamma_1 - [b_{112} \cdot \dot{q}_1 \dot{q}_2 + b_{113} \cdot \dot{q}_1 \dot{q}_3 + b_{123} \cdot \dot{q}_2 \dot{q}_3] - [c_{12} \dot{q}_2^2 + c_{13} \dot{q}_3^2]$$

$$I_2 = \Gamma_2 - [b_{223} \cdot \dot{q}_2 \dot{q}_3] - [c_{21} \dot{q}_1^2 + c_{23} \cdot \dot{q}_3^2] - g_2$$

$$I_3 = \Gamma_3 - [c_{31} \dot{q}_1^2 + c_{32} \cdot \dot{q}_2^2] - g_3$$

$$I_4 = \Gamma_4 - [b_{412} \cdot \dot{q}_1 \dot{q}_2 + b_{413}]$$

$$I_5 = \Gamma_5 - [c_{51} \cdot \dot{q}_1^2 + c_{52} \dot{q}_2^2] - g_5$$

$$I_6 = \Gamma_6$$

These equations tell us that in instruction to ensure that  $\ddot{q}_4$ ,  $\ddot{q}_5$  and  $\ddot{q}_6$  keep their zero values, it is better to set  $I_4 = I_5 = I_6 = 0$  ; so by holding the control torques of the last three joints as

$$I_4 = \Gamma_4 = [b_{412} \cdot \dot{q}_1 \dot{q}_2 + b_{413}]$$

$$I_5 = [c_{51} \cdot \dot{q}_1^2 + c_{52} \dot{q}_2^2] + g_5$$

And  $I_6 = 0$  the last three joints are blocked at their initial states.

## 3. Genetic Fuzzy Systems

The primary drawbacks of GA when using the Pittsburgh approach are that it is a more computationally convenient process due to the larger chromosome size required, thus increasing the search base for GA, with each chromosome indicating an entire FLC. The goal is to reduce the required chromosome length. A number of assumptions have been made regarding the FLC whose performance is desired

The following sections will describe the most commonly used methods and assumptions for encoding chromosome member parameters and rules with the fewest number of bits inspired by [6] [7][8][9][10].

### 3.1 Decoding Rule Base

Trying to code the base of the FLC connected to the FLC inputs are the error  $e(t)$ , the error derivative  $\Delta e(t)$ , and the output is the control action,  $u(t)$ . Each input from these three controlled variables is evaluated on a universe of natural discourse (UOD) level from [-1 1] starting assumptions that are set as follows:

- The Number of Fuzzy Sets (NFS ) for the controller is fixed at five (NB, NS, Z, PS PB.)
- The universe of discourse is symmetrical about the central.
- The magnitude of the control signal

Coordinated with the magnitude of the input signal.

These generalizations can be used decreasing the number of bits required for the base rule. This can be summarized to build a rule base by knowing the following parameters [7]:

- The grid spacing parameters [PSG<sub>(e)</sub>, PSG<sub>e</sub>] are the meaning of the distribution spaces of the fuzzy groups. on the [x, y] axis of each of the inputs to generate the base grid of the rule table and the [x, y] axis of the intersection produces network nodes that populate the outputs corresponding to each rule rule
- The following angle [0:90] specifies the slope of the line through the origin on which the initial points are placed.
- Consequent line order, (CLO) is defines order of consequent on the italic line .The CLO can be in two sequences ( NB-NS-Z-PS-PB or PB-PS-Z-NS-NB) and in effect doubles the range of possible consequent line angles to 0-360) .
- The Grid Spacing parameters of output (PSG<sub>Δu</sub>) is represent distribution of spaces for the fuzzy sets on the italic line .

After calculating of the all coordinates of the points (network nodes and output points), it is possible to proceed to the assignment by specifying the minimum distance between all the distances separating each node in the network from all other output points located on the straight line. Then we select the nearest output point for each node in the network. Determining the locations of the initial points and the grid spacing parameters of the inputs we will find by the way presented by Loudini [10]. Which ensures that the PS values are between [-1 1] as shown below.

The Ci principles are then defined in terms of the divergence coefficient PSG as follows

$$C_i = \text{sign} (CE_{q_i} ) * |CE_{q_i}|^{PCG} \quad (2)$$

where

$$\text{sign} (x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$$

$$CE_{q_i} = 2 \left[ \frac{i-1}{NEF-1} \right] - 1 \quad i=1, 2, 3, \dots, NFS \quad (3)$$

Descriptive examples of Ci computation are given in Table I (five FSs) for different values of the spacing parameter.

TABLE I. Ci for PSG for five FSs [10].

PSG	C <sub>i</sub>				
	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	C <sub>4</sub>	C <sub>5</sub>
0.25	-1	-0.84	0	0.84	1
0.5	-1	-0.70	0	0.70	1
1	-1	-0.50	0	0.50	1
2	-1	-0.25	0	0.25	1
4	-1	-0.06	0	0.06	1

To understand the decision table deriving procedure, detailed example is given below.

The PSG<sub>(e)</sub>= 0.5.

The PSG<sub>Δe</sub> = 1.

The PSG<sub>Δu</sub> = 2.

Angle = 30 . The construction parameters are given for the previous ones. The networks and their consistent decision tables are shown in Fig. 5. Note that the nodes are indicated by red stars and the output points by blue circles. Purple arrows are an example of minimum distances between output points and network nodes related to FS mapping to the decision table

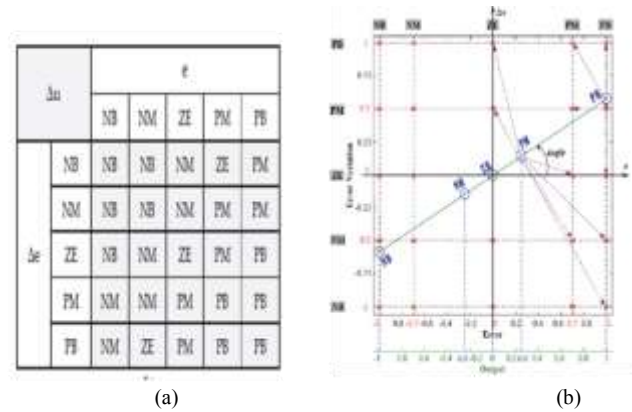


Fig.5. Example: (a) Grid constitution for decision table construction. (b) Derived decision table[10].

### 3.2 Decoding Membership Functions

In the studying to design the FLC membership functions with the two inputs and one output, a various of assumptions are made in admiration of the distribution of fuzzy sets across the UOD for each fuzzy variable. These assumptions are:

The internal and central UOD range of membership functions could adopt either triangular ,two sided Gaussian, sigmoidal, Gaussian curve, Generalized bell, product of two sigmoidally, Trapezoidal .shaped membership function shapes .

- Outer UOD range MFs is unbounded s-shaped or z-shaped.
- A half overlapping is maintained between adjacent membership functions.
- The apex for (trimf), or plateau for (trapmf), are coincident with zero segments of other adjacent and non-adjacent MF within the UOD.
- the number of fuzzy sets for the MFs fixed at five sets (NB, NS, Z, PS PB).
- Member of centers are distributed uniformly in the UOD.
- Using the presumption that have been invoked above we are able to find the MFs for inputs/outputs by finding the three parameters to minimizing the chromosomes size.

for inputs/output by finding the three parameters to minimizing the chromosomes size.

The membership function shape is specified by  $T_{MF}$  and  $S_{MF}$  as follows:

First, we have to choose 'Trapezoidal' Membership Function and find the four parameters using two variables, The MF shape is specified by  $T_{MF}$  and  $S_{MF}$  as follows:

a)  $T_{MF}$  determines the type of MF shape as follows:

- for  $T_{MF}$  1: 'dsigmf';
- for  $T_{MF}$  2 : gauss2mf.
- For  $T_{MF}$  3: gaussmf.
- for  $T_{MF}$  4: Gbellmf.
- for  $T_{MF}$  5: Psigmf.
- for  $T_{MF}$  6: trimf.

b)  $S_{MF}$  determines the symmetric space with respect to the centre of the MF as shown in figure 6 .

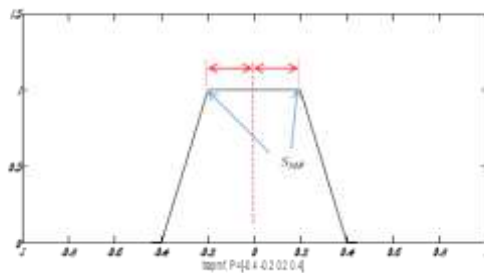


Fig. 6. Trapezoidal shape MF .

The author can replace the find PSG which used in creating rules base by degree of MF-center shift to effect MF to reduced or expand it,by using MF Compounding Field (CF) and is fired to update the MF position parameters of each MF by raising them to the power of CF [7]:

$$P_{New} = (P_{old})^{CF} \quad (4)$$

When

$P_{New}$  is new MF position parameters of each MF

$P_{old}$  is old MF position parameters of each MF .

And the value of CF is effect MF compression or expansion:

- a) for  $CF < 1$  : Z-MF is compressed, NB and NS expand.
- b) for  $CF > 1$  : Z-MF expands, NB and NS compress.
- c) for  $CF = 1$  : uniform MF distribution.

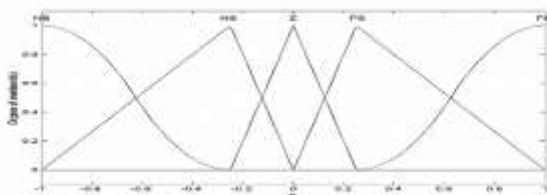


Fig. 7. MF GA-chromosome segment (triangular MFs with non-uniform distribution)[7].

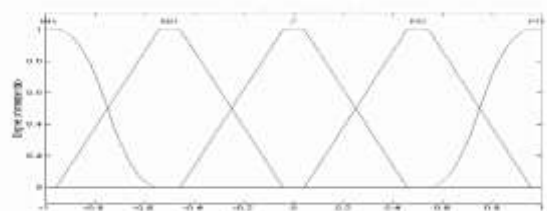


Fig.8. MF GA-chromosome segment (trapezoidal MFs with uniform distribution)[7].

### 3.3 Decoding defuzz Method

The Genetic Algorithm attempts to optimize the scaling factors (defuzzMethod type) of the  $e$  and  $\dot{e}$  inputs of the fuzzy controller. They are included in the GA-chromosome each consisting of 3-bits, which are decoded to obtain the values DFM , determines the type of defuzzMethod as follows:

- a) if DFM =1 then defuzzification is 'centroid'.
- b) if DFM =2 then defuzzification is 'bisector'.
- c) if DFM = 3 then defuzzification is 'mom' .
- d) if DFM = 4 then defuzzification is 'som'
- e) if DFM = 5 then defuzzification is 'lom' .

### 3.4 Decoding andMethod

There are two options for specifying and method (ANDM) which are specified by one bit as follows

- a) if ANDM= 1 then and method is min.
- b) if ANDM=2 then and method is pord.

### 3.5 Decoding or Method

There are three choices for specifying or method (ORM) which are specified by one bit as follows:

- a) If ORM=1 then method is max.
- b) If ORM=2 then method is pord.

### 3.6 Decoding aggMethod

There are three choices for specifying aggMethod (AGGM) which are specified by two bit as follows:

- a) If AGGM =1 then agg Method is max.
- b) if AGGM =2 then agg Method is sum .
- c) if AGGM =3 then agg Method is probor .

### 3.7 Decoding FLC Scaling Factors

The genetic algorithm also attempts to optimize the gains of scaling factors of the  $G_e$  and  $G_{\Delta e}$  inputs and  $G_{\Delta u}$  gain of output of the fuzzy logic control. The  $G_e$ ,  $G_{\Delta e}$ ,  $G_{\Delta u}$  are included in the GA chromosome, which are decoded to yield the appropriate gain blocks of the Simulink model used to evaluate each controller.

### 3.8 Measure of FIS performance GA Objective Function

After the fuzzy controller initial solution and begins the iterative evaluation of the generated new solutions by an objective (Index) function (OF ). The mathematical equation of (Of) in control applications found that using the following OF:

$$OF = \sum 0.5 * \text{sum}((e).^2 .* \text{time}) \quad (5)$$

Where  $e$  is the error.

## 4. Simulations and Results

The case study is to track a trajectory with the shape of circle. The figure 9 shows the GFL controlled robot system which includes the GFL controlling the robot block, The dynamic parameters and equation of the 3 DOF PUMA-560 robots are shown in the second chapter , the dynamic equations are simulations in 3 link PUMA robot blocks, and receiving the desired path from the path generator block. The output of the robot block is connected to a 2-D scope showing the desired and the actual motion of the joints, and to a 3-D scope showing the real and the desired path followed.

The 3D scope block is used to draw the trajectory followed by the robot in space. The inputs of this graphical block are

the *actual* and the desired coordinates of the end-effector of the robot, while its output is a 3-D graph showing the actual path followed in blue and the desired path in green. The 3-D scope block is based on the “Multitrack 3D Simulink Scope”.

The joint signals as extended as the desired path of the end-effector drawing a circle in the space are shown in figure 9 ; these signals enter the GFL controller that generates the corresponding torques needed to control the robot.

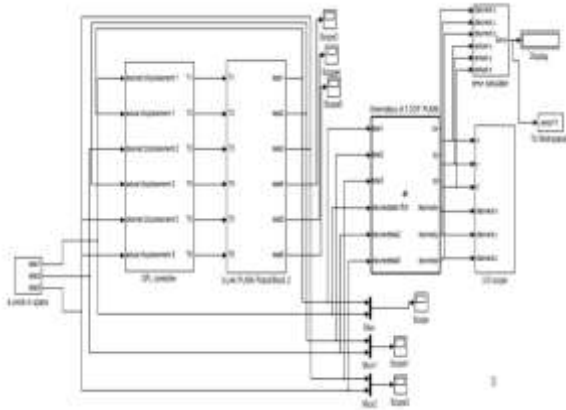


Fig. 9. GFL controlled robot system.

### 4.1 GA optimization of the FLC at 3 DOF PUMA robot

The genetic algorithm which used in this research is used to search of the fuzzy logic controller parameters based on the method described in third section . The tuning method is created MATLAB M-files .

Checking the output performance and redesign the fuzzy system for many times to reach to an satisfactory results. The OF at instant  $t$  is

$$\sqrt{(x_{desired} - x_{actual})^2 + (y_{desired} - y_{actual})^2 + (z_{desired} - z_{actual})^2} \quad (4.2)$$

Used as a multi-objective function of the system performance to give a better performance indicator of a control system response , the GA looks for the tuned of the FLC parameters to reach the designer specifications. Solutions with low OF are considered as the fittest. After many iteration, we have adopted the parameter encoding shown in Table II .

TABLE II. Parameters adopted for encoding at 3DOFPUMA robot.

Parameter	Interval	Number of encoding bits
$G_e, G_{\Delta e}$	[0,50]	7
$G_{\Delta u}$	[0,100]	7
$PSG_e, PSG_{\Delta e}, PSG_{\Delta u}$	[0.25,4]	4
ca	$[0,((15*\pi)/16)]$	4
cs	[0.5,2]	4
clo	[0,1]	1
$S_{MF}$	[0,0.25]	3

cf	[0.5,2]	4
$T_{MF}$	[1,6]	3
DFM	[1;5]	3
ANDM	[0,1]	1
ORM	[0,1]	1
AGGM	[0,3]	2

Several execution threads were executed with all GA parameters being different. The best result was obtained from the sum of errors =0.896. The corresponding GA characteristics are illustrated in Table IV .

TABLE IV GA adopted parameters at 3DOFPUMA robot.

Population size	16
Number of generations	1000
Selection method	Roulette wheel selection
Crossover method	1-point slicing Crossover
Crossover probability	0.5
Mutation probability	0.05

The GFLC design parameters and the main characteristics (MFs and decision table) are shown in Table V , Figure 10 and Table VI .

TABLE V GFLC design parameters at 3DOFPUMA robot.

Parameter	value		
	Joint 1	Joint 2	Joint 3
$G_e$	27.55905	41.338582	38.18897
$G_{\Delta e}$	29.92125	31.496062992125985	6.692913
$G_{\Delta u}$	6.45669	84.409448818897640	17.37007
$PSG_e$	0.25	1.5	0.75
$PSG_{\Delta e}$	0.5	2	1
$PSG_{\Delta u}$	2	3.25	1.5
ca	0.7854	0.5890	0.7854
cs	1.4	1.4	0.6
clo	0	0	0
$S_{MF}$ input 1	0.25	0.2143	0.1071
$S_{MF}$ input 2	0.0714	0.1071	0.25
$S_{MF}$ output	0.1071	0.2143	0.2143

<b>Cf input 1</b>	1.1	1.9	1.4
<b>Cf input 2</b>	1.6	1.1	2
<b>Cf output</b>	0.7	1.2	1.4
<b>DFM</b>	mom	bisector	bisector
<b>ANDM</b>	min	min	min
<b>ORM</b>	max	max	MAX
<b>AGGM</b>	sum	sum	SUM

TABLE VI .a The Rule table of the GFLC for first joint

$\Delta e1$ $e1$	NB	NS	Z	PS	PB
<b>NB</b>	NB	NB	NB	NS	NS
<b>NS</b>	NB	NB	NB	NS	Z
<b>Z</b>	NS	NS	Z	PS	PS
<b>PS</b>	Z	PS	PS	PB	PB
<b>PB</b>	PS	PS	PB	PB	PB

TABLE VI .b The Rule table of the GFLC for second joint .

$e2$ $\Delta e2$	NB	NS	Z	PS	PB
<b>NB</b>	NB	NB	NB	NS	NS
<b>NS</b>	NB	NB	NS	NS	Z
<b>Z</b>	NS	NS	Z	PS	PS
<b>PS</b>	Z	PS	PS	PB	PB
<b>PB</b>	PS	PS	PB	PB	PB

TABLE VI.c The Rule table of the GFLC for third joint .

$\Delta e3$ $e3$	NB	NS	Z	PS	PB
<b>NB</b>	NB	NB	NB	NS	NS
<b>NS</b>	NB	NS	NS	Z	Z
<b>Z</b>	NS	NS	Z	PS	PS
<b>PS</b>	Z	Z	PS	PS	PB
<b>PB</b>	PS	PS	PB	PB	PB

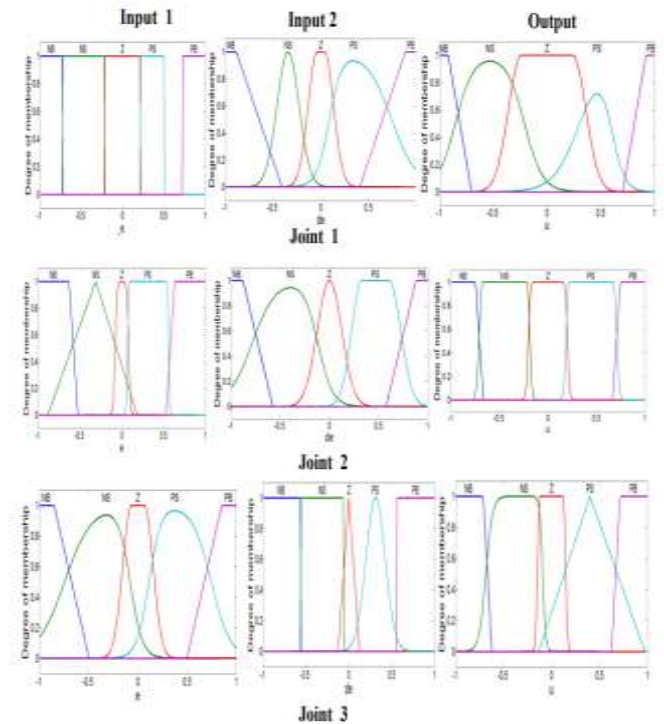


Fig.10. Optimal membership functions of the GFLC for 3 DOF PUMA robot

The figure 11 shown the following Trajectories in the simulation combined together with the desired spatial circle trajectory, and Actual versus Desired joint angles of the system.

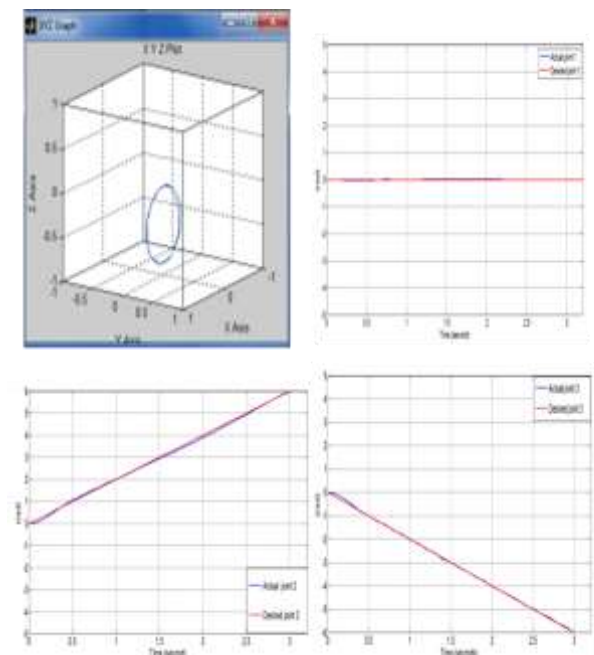


Fig.11. Actual versus desired joint angles and trajectory.

## 4.2 Adding uncertainty at 3DOF PUMA robot

The 3 DOF Puma 560 with parameters uncertainty It will be interpreted as changes in inertia values . The Inertial Constants with parameters uncertainty are thus introduced as ( $J_{m2} = 4.71 + 0.01, J_{m5} = 0.179 + 0.004, J_{m6} = 0.193 + 0.01$ ) kg.m<sup>2</sup>. The figure 4.14 shown the Actual versus Desired joint angles of the system with uncertainty added, it was a result OF total errors = 0.9305 , The difference is 0.0365 for not adding the uncertainty to the system. .The tracking error in task space is shown in Figure 12 .

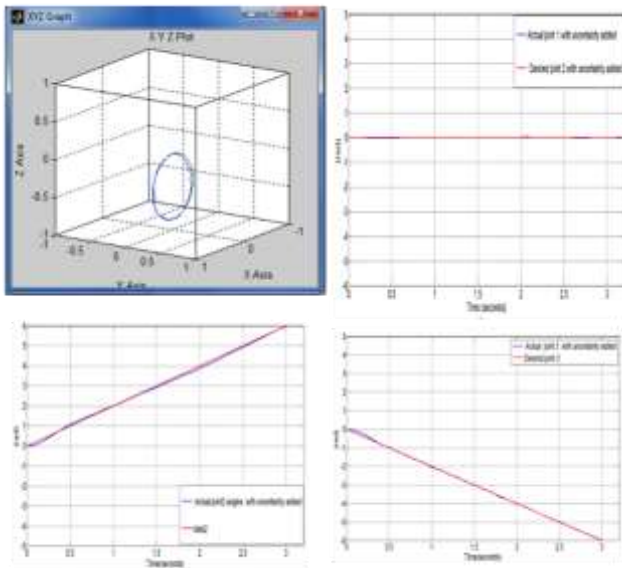


Fig. 12. Real Actual versus Desired joint angles and trajectory of the system with uncertainty added

## 5. Conclusion

This work has been introduced to study of advanced control systems based on fuzzy logic controls and genetic algorithm optimization technique which applied to the end - point 3 DOF PUMA 560 robot positioning of a planar three .

Due to the nonlinear characteristics and parameter changes in real environments, strong model uncertainty, and very complex, MIMO tracking control of a robot arm system is challenging. FLC was used in this research because it is a large scale system for controlling the nonlinear and uncertain system of parameters.

The performance of proposed GFLC has shown quite satisfactory performances I terms of settling time, rise time, steady state error, and settling in the endpoint position of the manipulator arm. The composite intelligent controller has also demonstrated robustness capabilities against additional system uncertainty. In other words, the retuning design approach by genetic the algorithm offers a complete and fast way to design a

robust controller. But there are many cons of using GLFC which can be summarized in the following points

- a) It takes several attempts to select parameters for GA.
- b) Determine the appropriate field of research for each variable that represents FLC.
- c) Writing the code for GFLC has the difficulties of representing and dealing with FLC variables because they are type structure.
- d) It takes a long time to run the GFLC program , as it sometime needs more than 48 continuous hours at MATLAB SIMULINK to make a computer and it will not be able to bear this load , which makes it stop before the search process is completed .running the algorithm for too long is inefficient while stopping the algorithm too early can result in an infeasible or sub-optimal solution. In the context of a multimodal FLC solution space, the use a restart strategy for GA in this study provided a successful solution to the problem of algorithm termination .The work stage is divided into several stages, and each research phase takes the last population and uses it in the initial population for the next stage.

## References

[1] Passino, K. M., Yurkovich, S., & Reinfrank, M. (1998). *Fuzzy control* (Vol. 42, pp. 15-21). Reading, MA: Addison-wesley.

[2] Medsker, L. R. (2012). *Hybrid intelligent systems*. Springer Science & Business Media.

[3] Muhammad R., Razali, Ahmad A. (2022). A hybrid controller method with genetic algorithm optimization to measure position and angular for mobile robot motion control *Front Robot AI*.

[4] Corke, P. I., & Khatib, O. (2011). *Robotics, vision and control: fundamental algorithms in MATLAB* (Vol. 73, p. 2). Berlin: Springer.

[5] Armstrong, B., Khatib, O., & Burdick, J. (1986, April). The explicit dynamic model and inertial parameters of the PUMA 560 arm. In *Proceedings. 1986 IEEE international conference on robotics and automation* (Vol. 3, pp. 510-518). IEEE.

[6] Mazhari, S. A., & Kumar, S. (2008). PUMA 560 optimal trajectory control using genetic algorithm, simulated annealing and generalized pattern search techniques. *International Journal of Electrical and Computer Engineering*, 2(5), 830-839.

[7] Byrne, J. P. (2003). GA-optimization of a fuzzy logic controller. *School of Electronic Engineering*.

[8] Mohammad V. and Ahmed F. (2018). Energy efficiency in the robot arm using genetic algorithm. Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN).

[9] Mazhari, S. A., & Kumar, S. (2008). PUMA 560 optimal trajectory control using genetic algorithm, simulated annealing and generalized pattern search techniques. *International Journal of Electrical and Computer Engineering*, 2(5), 830-839.

]32\_!Nqwf kpk 'O 0\*4235+00 qf gnkpi 'cpf 'kpwgnki gpv'eqpvtqri' qh' cp' grxwle' nkpm' tqdqv' o cplk wrvqt0kpwgtpcvkwpcni' lqwtpcni'qh'cf xcpegf 'Tdqdvle'U{ ugo u.'32\*3+': 30



**Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)**

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

**Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself**

No funding was received for conducting this study.

**Conflict of Interest**

The authors have no conflicts of interest to declare that are relevant to the content of this article.

**Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)**

This article is published under the terms of the Creative Commons Attribution License 4.0

[https://creativecommons.org/licenses/by/4.0/deed.en\\_US](https://creativecommons.org/licenses/by/4.0/deed.en_US)