International Journal of Hybrid Information Technology
Vol. 7, No. 6 (2014), pp. 175-188
http://dx.doi.org/10.14257/ijhit.2014.7.6.15

Tuning New Fuzzy Control for Nonlinear Second Order System

Babak Ranjbar, Gholam Dashti, Ali Omidvar, Javad Mahmoodi and Hasan Karbasi

Department of Electrical and Electronic Engineering, Delvar Branch, Islamic Azad University, Delvar, Iran
Babakranjbar2014@gmail.com

Abstract

Cognitive method is used in this research to create portfilo of movement robot manipulator. Gradient descent (GD) fuzzy controller was used and robot's postures and trajectory were expected in MATLAB/SIMULINK environment. Fuzzy logic controller (FLC) is an influential nonlinear controller to certain and uncertain systems which it is based on artificial intelligent and computes the required torques using the nonlinear feedback control law. Practically a large amount of systems have uncertainties accordingly this method has a challenge. Linear proportional-integral-derivative (PID) controller is a significant combination nonlinear stable-robust controller under condition of partly uncertain dynamic parameters of system. This technique is used to control of highly nonlinear systems especially in nonlinear time varient nonlinear dynamic system. To compensate for the dependence on model-free parameters, PID methodology is used. The nonlinear model dynamic formulation problem in uncertain system can be solved by using PID theorem. Proportional-Derivative-Integral (PID) theory is used to estimate the system dynamic.

Keywords**:** *Mamdani's fuzzy inference system, gradient descent optimization, PID controller*

1. Introduction

In modern usage, the word of control has many meanings, this word is usually taken to mean regulate, direct or command. The word feedback plays a vital role in the advance engineering and science. The conceptual frame work in Feed-back theory has developed only since world war II. In the twentieth century, there was a rapid growth in the application of feedback controllers in process industries. According to Ogata, to do the first significant work in three-term or PID controllers which Nicholas Minorsky worked on it by automatic controllers in 1922. In 1934, Stefen Black was invention of the feedback amplifiers to develop the negative feedback amplifier [3]. Negative feedback invited communications engineer Harold Black in 1928 and it occurs when the output is subtracted from the input. Automatic control has played an important role in advance science and engineering and its extreme importance in many industrial applications, *i.e.*, aerospace, mechanical engineering and robotic systems. Controller is a device which can sense information from linear or nonlinear systems to improve the systems performance [1-3]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error [4-5]. Several industrial systems are controlled by linear methodologies (*e.g.*, Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when system works in various situation and have uncertainty in dynamic models nonlinear methodology is introduced. In some applications systems are used in an unknown and unstructured environment, therefore strong mathematical tools used in new control methodologies to design nonlinear robust. Nonlinear controllers are divided

ISSN: 1738-9968 IJHIT
Copyright ⓒ 2014 SERSC

into to six groups, namely, feedback linearization (computed-torque control), passivity-based control, sliding mode control (variable structure control), artificial intelligence control, Lyapunov-based control and adaptive control [1, 6- 11].

Fuzzy Logic controller (FLC) is a powerful model-free nonlinear controller which it widely used in control of nonlinear systems. It is based on rule base of system and computes the required arm torques using the nonlinear feedback model free control law. This controller works very well when the formulation of dynamic and physical parameters are difficult or sometimes are unknown but when the system has variation in dynamic parameters, the controller has no acceptable performance [11]. In practice, most of physical systems parameters are time variant, therefore, online tuning fuzzy controller used to compensate nonlinear dynamic equation of this system [1, 6]. Research on fuzzy logic controller is significantly growing on nonlinear system application which has been reported in [1-5, 6-7]. Vivas and Mosquera [6] have proposed a predictive functional fuzzy controller and compare to feedback linearization controller for tracking response in uncertain environment. However both controllers have been used in feedback linearization, but predictive fuzzy strategy gives better result as a performance. A fuzzy logic control with non parametric regression models have been presented for a robot arm [7]. This controller also has been problem in uncertain dynamic models. Based on [1-5]and [6-7] fuzzy logic controller is a significant nonlinear controller to partly certain systems Proposed methodology is used to control of highly nonlinear systems especially for robot manipulator. In this method in the first time fuzzy logic controller is design and in the second step PID controller is used to design online tuning. However fuzzy logic controller works very good especially in nonlinear systems but it has an important challenges; nonlinear system's dynamic formulation in uncertain dynamic parameter. The nonlinear dynamic formulation problem in uncertain system is solved by using conventional PID controller [8]. Pure fuzzy logic controller and proposed methodology have difficulty in tune the controller's coefficient. It is possible to solve this problem by combining proposed methodology and gradient descent optimization (GDO). This method is based on resolve the switching feedback linearization controller (φ).

This paper is organized as follows:

- In Section 2, main subject of modeling robot manipulator formulation, forward kinematics, fuzzy logic method and PID methodology are presented.
- Detail of proposed methodology is presented in Section 3.
- In Section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. Theory

Dynamic Formulation of 2-DOF Robot Manipulator: Dynamic modeling of robot manipulators is used to describe the behavior of robot manipulator such as linear or nonlinear dynamic behavior, design of model based controller such as pure sliding mode controller which design this controller is based on nonlinear dynamic equations, and for simulation. The dynamic modeling describes the relationship between joint motion, velocity, and accelerations to force/torque or current/voltage and also it can be used to describe the particular dynamic effects (*e.g.,* inertia, coriolios, centrifugal, and the other parameters) to behavior of system[1]. The equation of an *n-DOF* robot manipulator governed by the following equation [1, 4]:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{N}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{\tau} \quad \textbf{(1)}$$

 Copyright ⓒ 2014 SERSC

Where τ is actuation torque, M (q) is a symmetric and positive define inertia matrix, $N(q,\dot{q})$ is the vector of nonlinearity term. This robot manipulator dynamic equation can also be written in a following form [1-4]:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{B}(\boldsymbol{q})[\dot{\boldsymbol{q}}\,\dot{\boldsymbol{q}}] + \boldsymbol{C}(\boldsymbol{q})[\dot{\boldsymbol{q}}]^{2} + \boldsymbol{G}(\boldsymbol{q}) \quad \textbf{(2)}$$

Where B(q) is the matrix of coriolios torques, C(q) is the matrix of centrifugal torques, and G(q) is the vector of gravity force. The dynamic terms in equation (2) are only manipulator position. This is a decoupled system with simple second order linear differential dynamics. In other words, the component $\ddot{q}$ influences, with a double integrator relationship, only the joint variableq_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3]:

$$\ddot{\boldsymbol{q}} = \boldsymbol{M}^{-1}(\boldsymbol{q}).\{\boldsymbol{\tau} - \boldsymbol{N}(\boldsymbol{q},\dot{\boldsymbol{q}})\} \quad \textbf{(3)}$$

This technique is very attractive from a control point of view.

Forward Kinematics of PUMA robot: Calculate the relationship between rigid bodies and end-effector without any forces is called Robot manipulator Kinematics. Study of this part is pivotal to calculate accurate dynamic part, to design with an acceptable performance controller, and in real situations and practical applications. As expected the study of manipulator kinematics is divided into two main parts: forward and inverse kinematics. Forward kinematics has been used to find the position and orientation of task (end-effector) frame when angles and/or displacement of joints are known. Inverse kinematics has been used to find possible joints variable (displacements and angles) when all position and orientation of end-effector be active [1].
The main target in forward kinematics is calculating the following function:

$$\boldsymbol{\Psi}(\boldsymbol{X},\boldsymbol{q}) = \boldsymbol{0} \quad \textbf{(4)}$$

Where $\Psi(.) \in R^{n}$ is a nonlinear vector function, $X = [X_1, X_2, \dots \dots, X_l]^T$ is the vector of task space variables which generally endeffector has six task space variables, three position and three orientation, $q = [q_1, q_2, \dots.., q_n]^T$ is a vector of angles or displacement, and finally n is the number of actuated joints. The Denavit-Hartenberg (D-H) convention is a method of drawing robot manipulators free body diagrams. Denvit-Hartenberg (D-H) convention study is necessary to calculate forward kinematics in serial robot manipulator. The first step to calculate the serial link robot manipulator forward kinematics is link description; the second step is finding the D-H convention after the frame attachment and finally finds the forward kinematics. Forward kinematics is a 4×4 matrix which 3×3 of them shows the rotation matrix, 3×1 of them is shown the position vector and last four cells are scaling factor [1, 6]. Singularity is a location in the robot manipulator's workspace which the robot manipulator loses one or more degrees of freedom in Cartesian space. Singularities are one of the most important challenges in inverse kinematics which Cheng *et al.*, have proposed a method to solve this problem [11]. A systematic Forward Kinematics of robot manipulator solution is the main target of this part. The first step to compute Forward Kinematics (F.K) of robot manipulator is finding the standard D-H parameters. The following steps show the systematic derivation of the standard D-H parameters.

1. Locate the robot arm
2. Label joints

3. Determine joint rotation or translation ($\theta\ or\ d$)
4. Setup base coordinate frames.
5. Setup joints coordinate frames.
6. Determine α_i, that α_i, link twist, is the angle between Z_i and Z_{i+1} about an X_i.
7. Determine d_i and a_i , that a_i, link length, is the distance between Z_i and Z_{i+1} along X_i. d_i, offset, is the distance between X_{i-1} and X_i along Z_i axis.
8. Fill up the D-H parameters table.

The second step to compute Forward kinematics for robot manipulator is finding the rotation matrix (R_n^0). The rotation matrix from $\{F_i\}$ to $\{F_{i-1}\}$ is given by the following equation;

$$\boldsymbol{R_i^{i-1} = U_{i(\theta_i)} V_{i(\alpha_i)}} \quad \textbf{(5)}$$

Where $U_{i(\theta_i)}$ is given by the following equation [1];

$$\boldsymbol{U_{i(\theta_i)} = \begin{bmatrix} cos(\theta_i) & -sin(\theta_i) & 0 \\ sin(\theta_i) & cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}} \quad \textbf{(6)}$$

and $V_{i(\alpha_i)}$ is given by the following equation [1];

$$\boldsymbol{V_{i(\theta_i)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\alpha_i) & -sin(\alpha_i) \\ 0 & sin(\alpha_i) & cos(\alpha_i) \end{bmatrix}} \quad \textbf{(7)}$$

So (R_n^0) is given by [1]

$$\boldsymbol{R_n^0 = (U_1V_1)(U_2V_2) \ldots \ldots \ldots (U_nV_n)} \quad \textbf{(8)}$$

The third step to compute the forward kinematics for robot manipulator is finding the displacement vector d_n^0, that it can be calculated by the following equation [1]

$$\boldsymbol{d_n^0 = (U_1S_1) + (U_1V_1)(U_2S_2) + \cdots + (U_1V_1)(U_2V_2) \ldots . (U_{n-1}V_{n-1})(U_nS_n)} \quad \textbf{(9)}$$

The forth step to compute the forward kinematics for robot manipulator is calculate the transformation ${}_n^0T$ by the following formulation [1]

$$\boldsymbol{{}_n^0T = {}_1^0T . {}_2^1T . {}_3^2T \ldots \ldots . {}_n^{n-1}T = \begin{bmatrix} R_n^0 & d_n^0 \\ 0 & 1 \end{bmatrix}} \quad \textbf{(10)}$$

Figure 1 shows the block diagram of two degrees of freedom robot manipulator.

 Copyright ⓒ 2014 SERSC

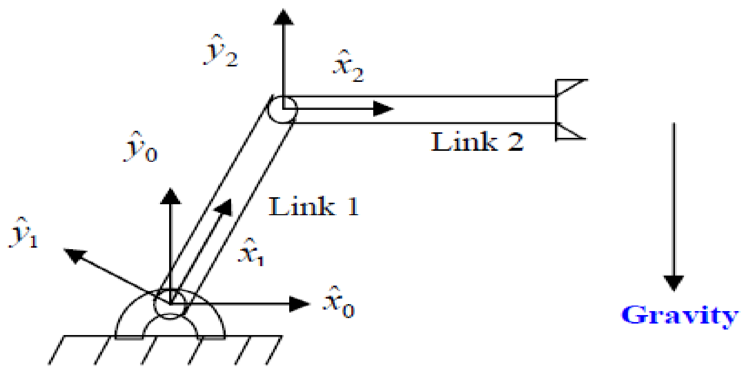

Figure 1. Manipulator Used for Simulations

Design PID Controller: Design of a linear methodology to control of continuum robot manipulator was very straight forward. Since there was an output from the torque model, this means that there would be two inputs into the PID controller. Similarly, the outputs of the controller result from the two control inputs of the torque signal. In a typical PID method, the controller corrects the error between the desired input value and the measured value. Since the actual position is the measured signal. Figure 2 is shown linear PID methodology, applied to continuum robot manipulator [1-6].

$$e(t) = \theta_a(t) - \theta_d(t) \quad (11)$$

$$U_{PID} = K_{p_a} e + K_{V_a} \dot{e} + K_I \sum e \quad (12)$$

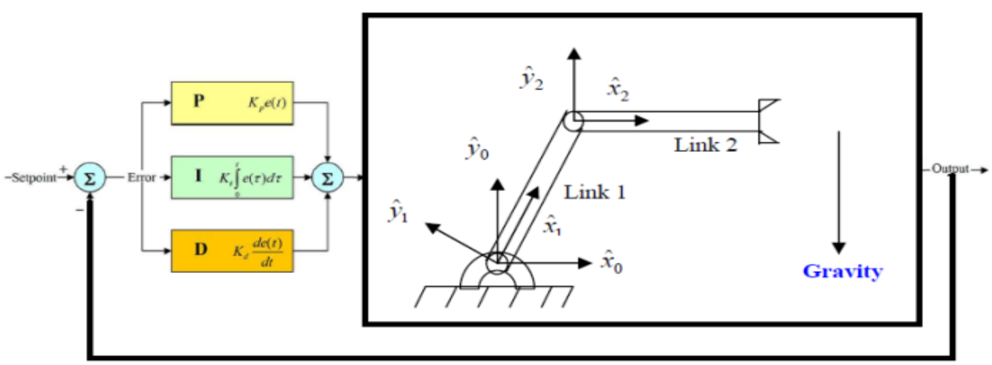

Figure 2. Block Diagram of Linear PID Method

The model-free control strategy is based on the assumption that the joints of the manipulators are all independent and the system can be decoupled into a group of single-axis control systems [11]. Therefore, the kinematic control method always results in a group of individual controllers, each for an active joint of the manipulator. With the independent joint assumption, no a priori knowledge of robot manipulator dynamics is needed in the kinematic controller design, so the complex computation of its dynamics can be avoided and the controller design can be greatly simplified. This is suitable for real-time control applications when powerful processors, which can execute complex algorithms rapidly, are not accessible.

Copyright ⓒ 2014 SERSC

However, since joints coupling is neglected, control performance degrades as operating speed increases and a manipulator controlled in this way is only appropriate for relatively slow motion [10-11]. The fast motion requirement results in even higher dynamic coupling between the various robot joints, which cannot be compensated for by a standard robot controller such as PID [10], and hence model-based control becomes the alternative.

Fuzzy Logic Methodology: Based on foundation of fuzzy logic methodology; fuzzy logic controller has played important rule to design nonlinear controller for nonlinear and uncertain systems [11-12]. However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of;
Input fuzzification (binary-to-fuzzy [B/F] conversion) Fuzzy rule base (knowledge base), Inference engine and Output defuzzification (fuzzy-to-binary [F/B] conversion). Figure 3 shows a fuzzy controller part.

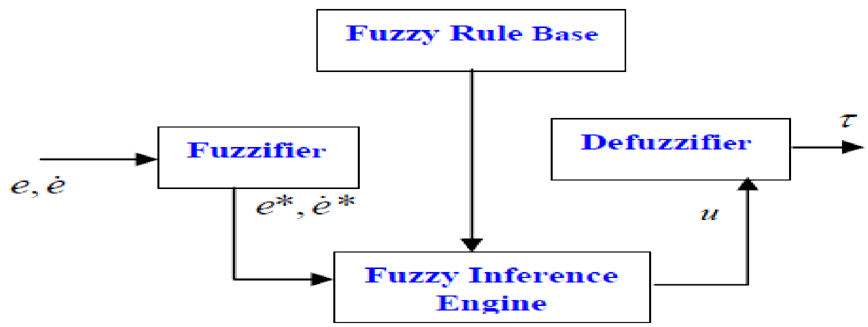

Figure 3. Fuzzy Controller Detail

Block diagram of fuzzy logic controller to control of 2 DOF robot manipulator shows in Figure 4.

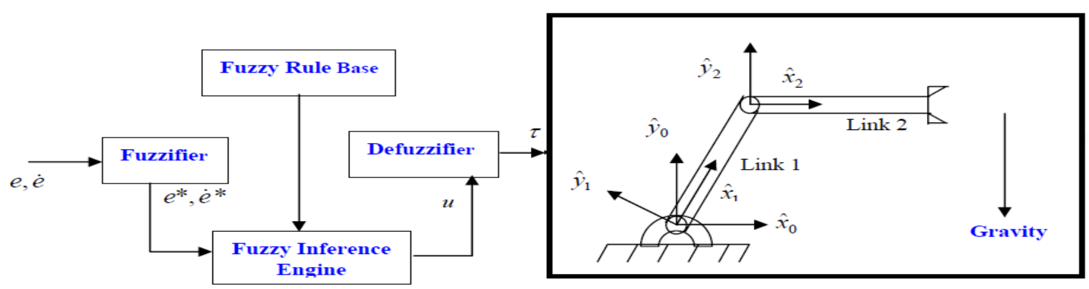

Figure 4. Block Diagram of Fuzzy Controller

The fuzzy inference engine offers a mechanism for transferring the rule base in fuzzy set which it is divided into two most important methods, namely, Mamdani method and Sugeno method. Mamdani method is one of the common fuzzy inference systems and he designed one of the first fuzzy controllers to control of system engine. Mamdani's fuzzy inference system is divided into four major steps: fuzzification, rule evaluation, aggregation of the rule outputs and defuzzification. Michio Sugeno use a singleton as a membership function of the rule consequent part. The following definition shows the Mamdani and Sugeno fuzzy rule base

 Copyright ⓒ 2014 SERSC

$$\textbf{\textit{if x is A and y is B then z is C 'mamdani'}} \tag{13}$$
$$\textbf{\textit{if x is A and y is B then z is f(x, y) 'sugeno'}}$$

When x and y have crisp values fuzzification calculates the membership degrees for antecedent part. The Figure of membership functions and linguistic variable shows in Figure 5.

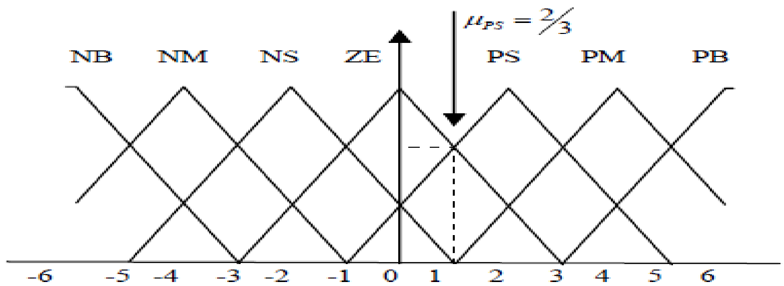

Figure 5. Membership Functions of Linguistic Variables

Rule evaluation focuses on fuzzy operation (AND/OR) in the antecedent of the fuzzy rules. The aggregation is used to calculate the output fuzzy set and several methodologies can be used in fuzzy logic controller aggregation, namely, Max-Min aggregation, Sum-Min aggregation, Max-bounded product, Max-drastic product, Max-bounded sum, Max-algebraic sum and Min-max. Two most common methods that used in fuzzy logic controllers are Max-min aggregation and Sum-min aggregation. Max-min aggregation defined as below;

$$\boldsymbol{\mu_U(x_k, y_k, U) = \mu_{\bigcup_{i=1}^r FR^i}(x_k, y_k, U) = \max\left\{\min_{i=1}^r\left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U)\right]\right\}} \tag{14}$$

The Sum-min aggregation defined as below

$$\boldsymbol{\mu_U(x_k, y_k, U) = \mu_{\bigcup_{i=1}^r FR^i}(x_k, y_k, U) = \sum \min_{i=1}^r\left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U)\right]} \tag{15}$$

where r is the number of fuzzy rules activated by x_k and y_k and also $\mu_{\bigcup_{i=1}^r FR^i}(x_k, y_k, U)$ is a fuzzy interpretation of $i-th$ rule. Defuzzification is the last step in the fuzzy inference system which it is used to transform fuzzy set to crisp set. Consequently defuzzification's input is the aggregate output and the defuzzification's output is a crisp number. Centre of gravity method (COG) and Centre of area method (COA) are two most common defuzzification methods, which COG method used the following equation to calculate the defuzzification

$$\boldsymbol{COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^r . \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^r . \mu_u(x_k, y_k, U_i)}} \tag{16}$$

and COA method used the following equation to calculate the defuzzification

$$\boldsymbol{COA(x_k, y_k) = \frac{\sum_i U_i . \mu_u(x_k, y_k, U_i)}{\sum_i \mu_U . (x_k, y_k, U_i)}} \tag{17}$$

Copyright ⓒ 2014 SERSC

Where $COG(x_k, y_k)$ and $COA(x_k, y_k)$ illustrates the crisp value of defuzzification output, $U_i \in U$ is discrete element of an output of the fuzzy set, $\mu_U.(x_k, y_k, U_i)$ is the fuzzy set membership function, and r is the number of fuzzy rules.

3. Methodology

In fuzzy error-based controller; error based Mamdani's fuzzy inference system has considered with two inputs, one output and totally 49 rules. Based on fuzzy logic methodology

$$\boldsymbol{f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x)} \tag{18}$$

where $\boldsymbol{\theta^T}$ is adjustable parameter (gain updating factor) and $\boldsymbol{\zeta(x)}$ is defined by

$$\boldsymbol{\zeta(x) = \frac{\sum_i \mu(x_i) x_i}{\sum_i \mu(x_i)}} \tag{19}$$

Where $\boldsymbol{\mu(x_i)}$ is membership function. $\boldsymbol{\tau_{fuzzy}}$ is defined as follows;

$$\boldsymbol{\tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x)} \tag{20}$$

Design of error-based fuzzy based on Mamdani's fuzzy inference method has four steps , namely, fuzzification, fuzzy rule base and rule evaluation, aggregation of the rule output (fuzzy inference system) and defuzzification.

***Fuzzification**:* the first step in fuzzification is determine inputs and outputs which, it has two inputs $(e, \dot{e})$ and one output (τ_{fuzzy}). The inputs are error (e) which measures the difference between desired and actual output position, and the change of error $(\dot{e})$ which measures the difference between desired and actual velocity and output is fuzzy equivalent torque. The second step is chosen an appropriate membership function for inputs and output which, to simplicity in implementation because it is a linear function with regard to acceptable performance triangular membership function is selected in this research. The third step is chosen the correct labels for each fuzzy set which, in this research namely as linguistic variable. Based on experience knowledge the linguistic variables for error (e) are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and experience knowledge it is quantized into thirteen levels represented by: -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6 the linguistic variables for change of error $(\dot{e})$ are; Fast Left (FL), Medium Left (ML), Slow Left (SL),Zero (Z), Slow Right (SR), Medium Right (MR), Fast Right (FR), and it is quantized in to thirteen levels represented by: -6, -5, -0.4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, and the linguistic variables to find the output are; Large Left (LL), Medium Left (ML), Small Left (SL), Zero (Z), Small Right (SR), Medium Right (MR), Large Right (LR) and it is quantized in to thirteen levels represented by: : -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6

 Copyright ⓒ 2014 SERSC

Fuzzy rule base and rule evaluation: the first step in rule base and evaluation is to provide a least structured method to derive the fuzzy rule base which, expert experience and control engineering knowledge is used because this method is the least structure of the other one and the researcher derivation the fuzzy rule base from the knowledge of system operate and/or the classical controller. Design the rule base of fuzzy inference system can play important role to design the best performance of fuzzy sliding mode controller, that to calculate the fuzzy rule base the researcher is used to heuristic method which, it is based on the behavior of the control of robot manipulator. The complete rule base for this controller is shown in Table 1. Rule evaluation focuses on operation in the antecedent of the fuzzy rules in fuzzy sliding mode controller. This part is used AND/OR fuzzy operation in antecedent part which AND operation is used.

Table 1. Modified Fuzzy Sliding Mode Rule Table

Decrease the overshoot → $\dot{e}$

$\dot{e}$ \ e	FL	ML	SL	Z	SR	MR	FR
NB	LL	LL	LL	ML	SL	SL	Z
NM	LL	ML	ML	ML	SL	Z	SR
NS	LL	ML	SL	SL	Z	SR	MR
Z	LL	ML	SL	Z	SR	MR	LR
PS	ML	SL	Z	SR	SR	MR	LR
PM	SL	Z	SR	MR	MR	MR	LR
PB	Z	SR	SR	MR	LR	LR	LR

e → Decrease the rise time

Aggregation of the rule output (Fuzzy inference): Max-Min aggregation is used in this work.

Defuzzification: The last step to design fuzzy inference in our fuzzy sliding mode controller is defuzzification. This part is used to transform fuzzy set to crisp set, therefore the input for defuzzification is the aggregate output and the output of it is a crisp number. Center of gravity method (COG) is used in this research. The lookup table for fuzzy controller shows in Table 2.

Table 2. Lookup Table for the Fuzzy Controller

→ e	MEMBERSHIP FUNCTION												
$\dot{e}$ ↓	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-5.6	-5.4	-5.0	-4.8	-4.8	-4.7	-4.7	-4.6	-4.5	-4.4	-4.3	-4.3	-4.2
-5	-4.7	-4.5	-4.4	-4.3	-4.2	-4.1	-4.0	-3.9	-3.8	-3.8	-3.7	-3.6	-3.5
-4	-3.7	-3.6	-3.5	-3.2	-3.0	-3.0	-3.0	-2.9	-2.9	-2.8	-2.8	-2.7	-2.7
-3	-2.0	-2.0	-1.9	-1.9	-1.8	-1.8	-1.7	-1.7	-1.6	-1.5	-1.4	-1.3	-1.3
-2	0.0	0.0	-0.8	-1.0	-1.2	-1.7	-2.3	-2.2	-2.2	-2.0	-2.0	-1.0	-1.0
-1	1.0	1.0	0.0	0.0	-0.5	-0.5	-0.5	-1.0	-1.2	-1.5	-1.7	-1.0	-1.0
0	1.3	1.2	1.0	0.8	0.6	0.0	-0.2	-0.4	-0.6	-0.8	-1.0	-1.0	-1.0
1	2.0	2.0	1.9	1.8	1.8	1.8	1.8	1.8	1.5	0.0	-0.3	-1.0	-0.8
2	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	1.2	0.8	0.0	0.0
3	2.0	2.1	2.3	2.5	2.5	2.5	2.6	2.7	2.8	2.8	2.9	2.9	3.0
4	2.7	2.7	2.8	3.1	3.2	3.3	3.5	3.6	3.6	3.8	3.8	3.9	3.9
5	3.6	3.3	3.7	4.0	4.1	4.3	4.3	4.4	4.4	4.5	4.5	4.6	4.7
6	4.4	4.4	4.3	4.8	5.0	5.0	5.1	5.2	5.3	5.4	5.6	5.6	5.6

Copyright ⓒ 2014 SERSC

The next step in this research is online tuning the fuzzy inputs and outputs gain updating factors. Figure 6 shows the fuzzy controller and gain updating factors. In this Figure GE, GCE and GU are gain updating factors and block F is fuzzy controller block.

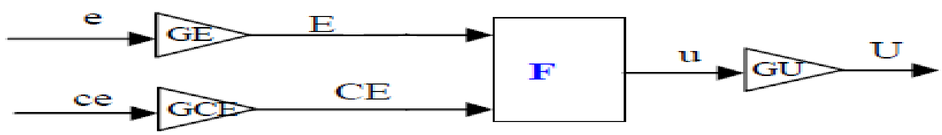

Figure 6. Gain Updating Factors in Fuzzy Controller

To tuning these coefficients conventional PID controller is used. Figure 7 shows the block diagram of online tuning fuzzy controller based on PID method.

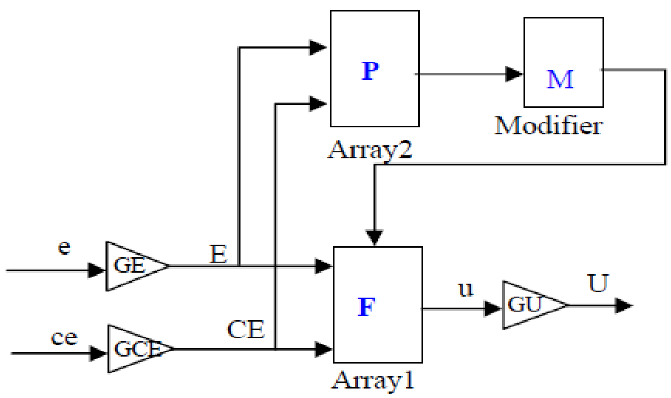

Figure 7. Block Diagram of PID Tuning Fuzzy Controller

Table 3 shows the PID tuning fuzzy controller.

Table 3. Lookup Table for the Proposed Method

$\dot{e}$ \ e	Membership Function												
	-6	**-5**	**-4**	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**	**4**	**5**	**6**
-6	-6	-6	-6	-6	-6	-6	-6	-5	-4	-3	-2	-1	0
-5	-6	-6	-6	-6	-5	-4	-4	-4	-3	-2	-1	0	0
-4	-6	-6	-6	-5	-4	-3	-3	-3	-2	-1	0	0	1
-3	-6	-6	-5	-4	-3	-2	-2-	-2	-1	0	0	1	2
-2	-6	-5	-4	-3	-2	-1	-1	-1	0	0	1	2	3
-1	-5	-4	-3	-2	-1	-1	0	0	0	1	2	3	4
0	-5	-4	-3	-2	-1	0	0	0	1	2	3	4	5
1	-3	-2	-1	0	0	0	0	1	1	2	3	4	5
2	-2	-1	0	0	0	1	1	1	2	3	4	5	6
3	-1	0	0	0	1	2	2	2	3	4	5	6	6
4	0	0	0	1	2	3	3	3	4	5	6	6	6
5	0	0	1	2	3	4	4	4	5	6	6	6	6
6	0	1	2	3	4	5	6	6	6	6	6	6	6

Copyright ⓒ 2014 SERSC

4. Results and Discussion

In this research, proposed controller is used to test the output position and orientation. To test the validity this method is compared with fuzzy logic controller in certain and uncertain environment.

Optimization the Proposed Method: in proposed methodology; controller's performance are depending on the PID coefficient K_p, K_v, K_i and gain updating factor coefficients. These coefficients are computed by Gradient Descent Algorithm optimization; Figure 8 and Figure 9.

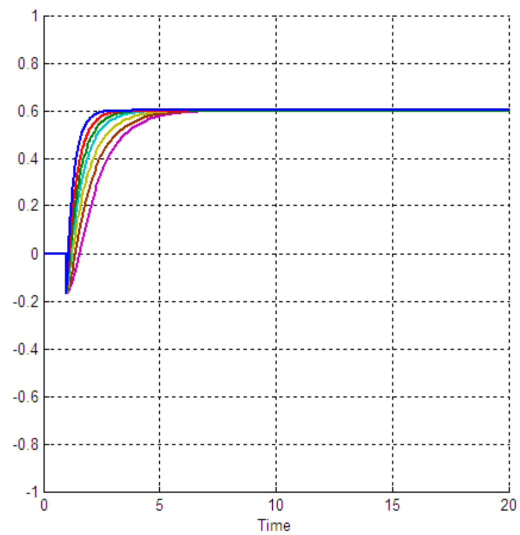

Figure 8. Trajectory Following Gradient Descent Optimization

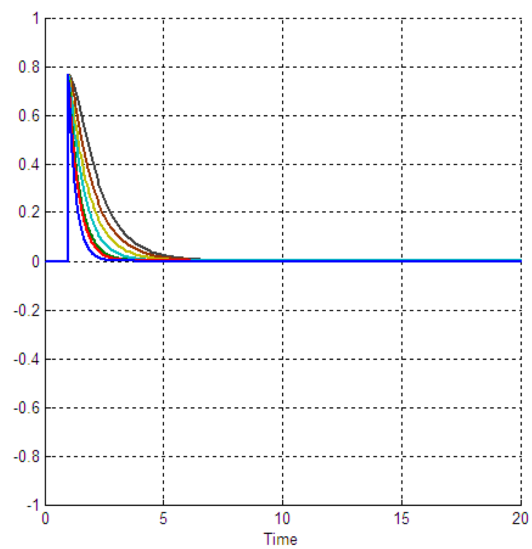

Figure 9. Error's Gradient Descent Optimization

Position and Orientation Test: Figures 10 and 11 are shown the XYZ trajectory following in pure fuzzy logic controller and proposed controller. Based on these two figures, proposed methodology has better performance to better enforcement of the order.

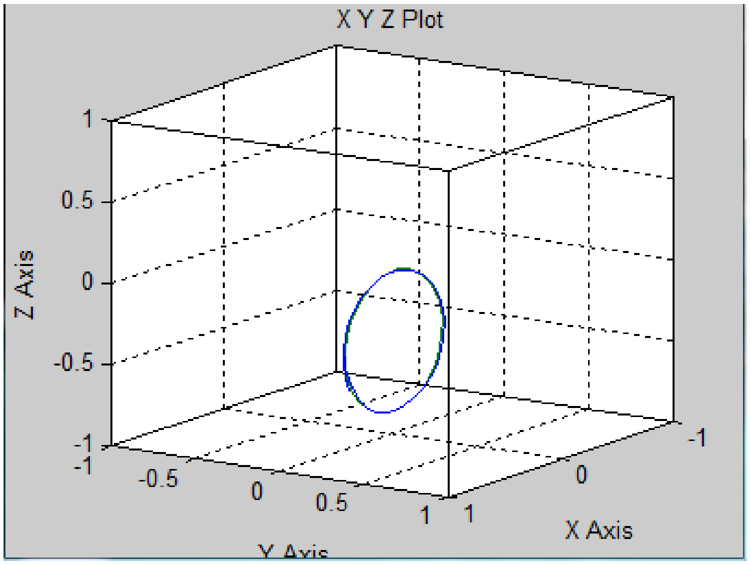

Figure 10. Proposed Position and Orientation in 3D Dimensions

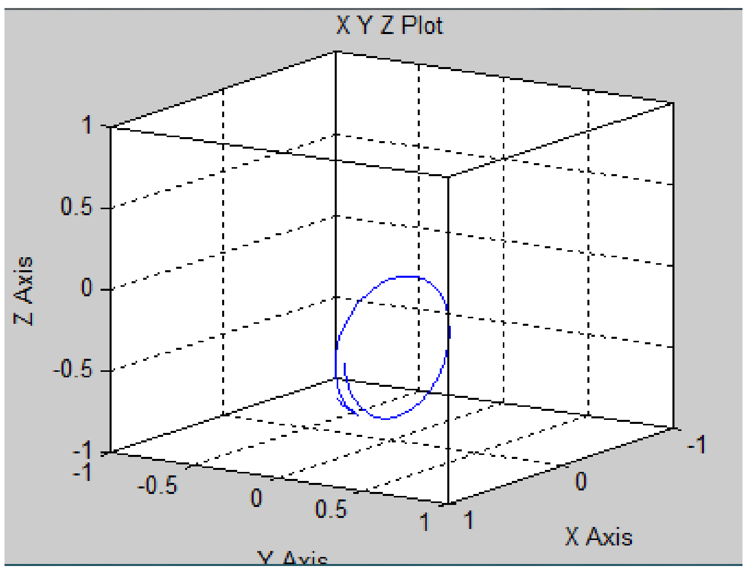

Figure 11. Fuzzy Position and Orientation in 3D Dimensions

It may be claimed that however fuzzy logic control is used in many research and industry but proposed method has a better XYZ position and orientation (POSE). The data indicates that proposed method can guarantee the stability and minimal error.

Disturbance Rejection: Figures 12 and 13 shows the power disturbance and external noise elimination in proposed method and FLC. As a matter of fact the disturbance rejection is used to test the robustness of these two methods. Surely after applied different external noise in these two methods it found fairly fluctuations in POSE responses. Actually one of the main advantages of proposed methodology is stability in presence of external disturbance. A further advantage of proposed method is increase the robustness compared to FLC.

 Copyright ⓒ 2014 SERSC

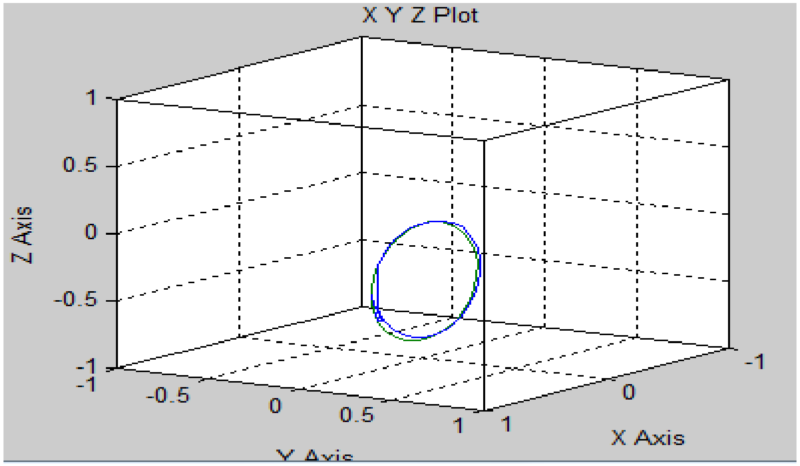

Figure 12. Disturbance Rejection in Proposed Position and Orientation in 3D Dimensions

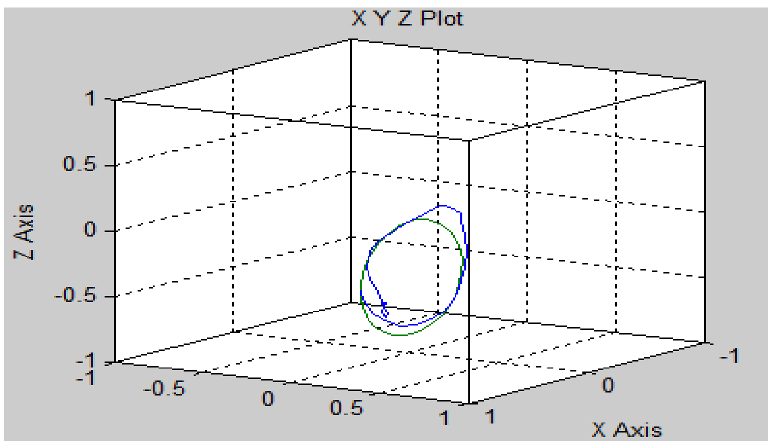

Figure 13. Disturbance Rejection in FLC Position and Orientation in 3D Dimensions

Based on Figure 13, PID tuning caused to increase the stability and robustness and it can eliminate the limited external disturbance.

5. Conclusion

Refer to this research, design a PID tuning fuzzy logic methodology is proposed to tune the POSE in two degrees of freedom robot manipulator. At first pure fuzzy logic controller is design for POSE tune in this robot. This method has two important challenge; stability and robustness because this method is work based on intelligent rule base formulation. PID controller is used to resolve uncertainty problem. To solve stability challenge in above method PID methodology is used. Eventually gradient descent optimization is applied to proposed methodology to tune all four parameters. The POSE results demonstrate that the gradient descent proposed method is a partly model base method which works well in certain and partly uncertain system.

Copyright ⓒ 2014 SERSC

References

[1] T. R. Kurfess, "Robotics and automation handbook: CRC", (**2005**).
[2] J. J. E. Slotine and W. Li, "Applied nonlinear control vol. 461: Prentice hall Englewood Cliffs", NJ, (**1991**).
[3] B. Siciliano and O. Khatib, "Springer handbook of robotics: Springer-Verlag New York Inc"., (**2008**).
[4] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 27, no. 2, (**2002**), pp. 332-343.
[5] M. W. Spong and M. Vidyasagar, "Robot dynamics and control: Wiley-India", (**2009**).
[6] A. Vivas and V. Mosquera, "Predictive functional control of a PUMA robot," Conference Proceedings, (**2005**).
[7] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," IEEE conference proceeding, (**2008**), pp. 212-217.
[8] F. Piltan, N. Sulaiman, Z. Tajpaykar, Payman Ferdosali, Mehdi Rashidi, "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, vol. 2, no. 3, (**2011**), pp. 205-220.
[9] F. Piltan, A. R. Salehi and N. B. Sulaiman," Design artificial robust control of second order system based on adaptive fuzzy gain scheduling," world applied science journal (WASJ), vol. 13, no. 5, (**2011**), pp. 1085-1092.
[10] F. Piltan, N. Sulaiman, A. Gavahian, S. Soltani and S. Roosta, "Design Mathematical Tunable Gain PID-Like Sliding Mode Fuzzy Controller with Minimum Rule Base," International Journal of Robotic and Automation, vol. 2, no. 3, (**2011**), pp. 146-156.
[11] F. Piltan, A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, vol. 12, no. 12, (**2011**), pp. 2330-2336.

Author

Babak Ranjbar, he is a lecturer at Electrical Engineering department at IAU, Bushehr, Iran.

Copyright ⓒ 2014 SERSC