Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine and Application to Classical Controller

Farzin Piltan SSP.ROBOTIC@yahoo.com
Department of Electrical and Electronic
Engineering, Faculty of
Engineering,Universiti Putra Malaysia
43400 Serdang, Selangor, Malaysia

SH. Tayebi HAGHIGHI SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic
Engineering SanatkadeheSabze
Pasargad. CO (S.S.P. Co),
NO:16,PO.Code 71347-66773, Fourth
floor Dena Apr , Seven Tir Ave , Shiraz ,
Iran

N. Sulaiman nasri@eng.upm.edu.my
Department of Electrical and Electronic
Engineering, Faculty of
Engineering,Universiti Putra Malaysia
43400 Serdang, Selangor, Malaysia

Iman Nazari SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic
Engineering SanatkadeheSabze
Pasargad. CO (S.S.P. Co),
NO:16,PO.Code 71347-66773, Fourth
floor Dena Apr , Seven Tir Ave , Shiraz ,
Iran

Sobhan Siamak SSP.ROBOTIC@yahoo.com
Industrial Electrical and Electronic Engineering
SanatkadeheSabze Pasargad. CO (S.S.P. Co),
NO:16,PO.Code 71347-66773, Fourth floor
Dena Apr , Seven Tir Ave , Shiraz , Iran

Abstract

One of the most important challenges in the field of robotics is robot manipulators control with acceptable performance, because these systems are multi-input multi-output (MIMO), nonlinear and uncertainty. Presently, robot manipulators are used in different (unknown and/or unstructured) situation consequently caused to provide complicated systems, as a result strong mathematical theory are used in new control methodologies to design nonlinear robust controller with acceptable performance (e.g., minimum error, good trajectory, disturbance rejection). Classical and non-classical methods are two main categories of robot manipulators control, where the conventional (classical) control theory uses the classical method and the non-classical control theory (e.g., fuzzy logic, neural network, and neuro fuzzy) uses the artificial intelligence methods. However both of conventional and artificial intelligence theories have applied effectively in many areas, but these methods also have some limitations. This paper is focused on review of fuzzy logic controller and applied to PUMA robot manipulator.

Keywords: PUMA Robot Manipulator, Classical Controller, Artificial Intelligence Controller, Fuzzy Logic Theory.

1. INTRODUCTION

The international organizations in order to standardize the robot definition have a special description as “an automatically controlled, reprogrammable, multipurpose manipulator with three or more axes.” The institute of robotic in The United States Of America defines the robot as “a reprogrammable, multifunctional manipulator design to move material, parts, tools, or specialized devices through various programmed motions for the performance of variety of tasks”[1]. Robot manipulator is collection of links that connect to each other by joints, these joints can be revolute and prismatic that revolute joint has rotary motion around an axis and prismatic joint has linear motion around an axis. Each joint provides one or more degrees of freedom (DOF). From the mechanical point of view, robot manipulator is divided into two main groups, which called; serial robot links and parallel robot links. In serial robot manipulator, links and joints is serially connected between base and final frame (end-effector). Parallel robot manipulators have many legs with some links and joints, where in these robot manipulators base frame has connected to the final frame. Most of industrial robots are serial links, which in serial robot manipulator the axis of the first three joints has a known as major axis, these axes show the position of end-effector, the axis number four to six are the minor axes that use to calculate the orientation of end-effector, at last the axis number seven to n use to avoid the bad situation. Kinematics is an important subject to find the relationship between rigid bodies (e.g., position and orientation) and end-effector in robot manipulator. The mentioned topic is very important to describe the three areas in robot manipulator: practical application, dynamic part, and control purposed therefore kinematics play important role to design accurate controller for robot manipulators. Robot manipulator kinematics is divided into two main groups: forward kinematics and inverse kinematics where forward kinematics is used to calculate the position and orientation of end-effector with given joint parameters (e.g., joint angles and joint displacement) and the activated position and orientation of end-effector calculate the joint variables in Inverse Kinematics[6]. Dynamic modeling of robot manipulators is used to describe the behavior of robot manipulator, design of model based controller, 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 Unimation PUMA 560 serially links robot manipulator was used as a basis, because this robot manipulator widely used in industry and academic. It has a nonlinear and uncertain dynamic parameters serial link 6 degrees of freedom (DOF) robot manipulator. A non linear robust controller design is major subject in this work.

It is a well known fact, the aim of science and modern technology has making an easier life. Conversely, modern life includes complicated technical systems which these systems (e.g., robot manipulators) are nonlinear, time variant, and uncertain in measurement, they need to have controlled. Consequently it is hard to design accurate models for these physical systems because they are uncertain. From the control point of view uncertainty is divided into two main groups: uncertainty unstructured inputs (e.g., noise, disturbance) and uncertainty structure dynamics (e.g., payload, parameter variations). At present, in some applications robot manipulators are used in unknown and unstructured environment, therefore strong mathematical tools used in new control methodologies to design nonlinear robust controller with an acceptable performance (e.g., minimum error, good trajectory, disturbance rejection) [1-6]. One of the nonlinear controllers is fuzzy logic controller which it is used in nonlinear uncertain systems.

In this research we will highlight the fuzzy logic algorithm and application to classical controllers. This algorithm will be analyzed and evaluated on PUMA robotic manipulators. Section 2, serves as an introduction to the PUMA robot manipulator and it’s formulation. Part 3, introduces and review of the fuzzy logic algorithm and it’s application and the final section is describe the conclusion.

2. PUMA ROBOT MANIPULATOR: KINEMATICS AND DYNAMIC

Robot manipulators have many applications in aerospace, manufacturing, automotive, medicine and other industries. Robot manipulators consist of three main parts: mechanical, electrical, and control. In the mechanical point of view, robot manipulators are collection of serial or parallel links which have connected by revolute and/or prismatic joints between base and end-effector frame. The robot manipulators electrical parts are used to links motion, which including the following subparts: power supply to supply the electrical and control parts, power amplifier to amplify the signal and driving the actuators, DC/stepper/servo motors or hydraulic/pneumatic cylinders to motion the links, and transmission part to transfer data between robot manipulator subparts. Control part is used to adjust the timing between the subparts of robot manipulator to reach the best performance (trajectory). It provides four main abilities in robot manipulators: controlling the manipulators movement in correct workspace, sensing the information from the environment, being able to intelligent control behavior and processing the data and information between all subparts. The first person who used the word robot was Karel Capek in 1920 in his satirical play, R.U.R (Rossum's Universal Robots). The first person who used the word robotics was the famous author, Issac Asimov along with three fundamental rules. Following World War II, the first industrial robot manipulator have been installation at General Motors in 1962 for the automation. In 1978 the PUMA (Programmable Universal Machine for Assembly) and in 1979 the SCARA (Selective Compliance Assembly Robot Arm) were introduced and they were quickly used in research laboratories and industries. According to the MSN Learning & Research," 700000 robots were in the industrial world in 1995 and over 500000 were used in Japan, about 120000 in Western Europe, and 60000 in the United States." In 1940 the Ford Motor company used the word "automation" which this word is a contraction of "automatic motivation". Automation play important role in new industry which changed the slow and heavy systems to faster, lighter and smarter systems. In the recent years robot manipulators not only have been used in manufacturing but also used in vast area such as medical area and working in International Space Station. Control methodologies and the mechanical design of robot manipulators have started in the last two decades and the most of researchers work in these methodologies. In next two sections, classification of robot manipulators and their effect on design controller are presented. The following sections are focused on analysis the kinematic and dynamic equations to control of robot manipulator [1, 6]. Research about mechanical parts and control methodologies in robotic system is shown; the mechanical design, type of actuators, and type of systems drive play important roles to have the best performance controller. This section has focused on the robot manipulator mechanical classification. More over types of kinematics chain, i.e., serial Vs. parallel manipulators, and types of connection between link and join actuators, i.e., highly geared systems Vs. direct-drive systems are presented in the following sections because these topics played important roles to select and design the best acceptable performance controllers[6].

A serial link robot is a sequence of joints and links which begins with a base frame and ends with an end-effector. This type of robot manipulators, comparing with the load capacitance is more weightily because each link must be supported the weights of all next links and actuators between the present link and end-effector[6]. Serial robot manipulators have been used in automotive industry, medical application, and also in research laboratories. In contrast, parallel robot manipulators design according to close loop which base frame is connected to the end-effector frame with two or more kinematic chains[6]. In the other words, a parallel link robot has two or more branches with some joints and links, which support the load in parallel. Parallel robot have been used in many applications such as expensive flight simulator, medical robotics (I.e., high accuracy, high repeatability, high precision robot surgery), and machinery tools. With comparison between serial and parallel links robot manipulators, parallel robots are used in higher speed loads, better accuracy, with used lighter weigh robot manipulator but one of the most important handicaps is limitation the workspace compared to serial robot. From control point of view, the coupling between different kinematic chains can generate the uncertainty problems which cause difficult controller design of parallel robot manipulator[6]. One of the most important classifications in controlling the robot manipulator is how the links have connected to the actuators. This classification divides into two main groups: highly geared (e.g., 200 to 1) and direct drive (e.g., 1 to 1). High gear ratios reduce the nonlinear coupling dynamic parameters in robot manipulator. In this case, each joint is modeled the same as SISO systems. In high gear robot manipulators which generally are used in industry, the couplings are modeled as a disturbance for SISO systems. Direct drive increases the coupling of nonlinear dynamic parameters of robot manipulators. This effect should be considered in the design of control systems. As a result some control and robotic researchers' works on nonlinear robust controller design[2].

Introduction of Rigid Body Kinematics: PUMA Robot Manipulator

Study of robot manipulators is classified into two main groups: kinematics and dynamics. Calculate the relationship between rigid bodies and endeffector 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 finally in real situations and practical applications. As expected the study of manipulator kinematics is divided in two main challenges: 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 and 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 [14]:

$$\Psi(X,q) = 0 \tag{1}$$

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. 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 find the D-H convention after the frame attachment and finally find 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]. Wu has proposed PUMA 560 robot arm kinematics based on accurate analysis[9]. The inverse kinematics problem is calculation of joint variables (i.e., displacement and angles), when position and orientation of end-effector to be known. In other words, the main target in inverse kinematics is to calculate $q = h^{-1}(X)$, where q is joint variable, q =[$q_1, q_2, \dots\dots, q_n$], and X are position and orientation of endeffector, X=[X,$Y, Z, \emptyset, \theta, \Psi$]. In general analysis the inverse kinematics of robot manipulator is difficult because, all nonlinear equations solutions are not unique (e.g., redundant robot, elbow-up/elbow-down rigid body), and inverse kinematics is different for different types of robots. In serial links robot manipulators, equations of inverse kinematics are classified into two main groups: numerical solutions and closed form solutions. Most of researcher works on closed form solutions of inverse kinematics with different methods, such as inverse transform, screw algebra, dual matrix, iterative, geometric approach and decoupling of position and orientation[1, 6]. Research on the Inverse Kinematics robot manipulator PUMA 560 series, like in some applications has been working. For instance, Hong Zhang has worked on particular way of robot kinematics solution to reduce the computation[10]. Jon Kieffer has proposed a simple iterative solution to computation of inverse kinematics[11]. Ziauddin Ahmad et al., are solved the robot manipulator inverse kinematics by neural network hybrid method which this method is combining the advantages of neural network and iterative methods[12]. Singularities are one of the most important challenges in inverse kinematics which F. T. Cheng et al., have proposed a method to solve this problem[13]. Kinematics is significant topic to find the relationship between rigid bodies (e.g., position and orientation) and end-effector in robot manipulator. This topic is very important to describe three areas in robot manipulator, namely, practical application, dynamic part and control purposed. 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 Denavit-Hartenberg (D-H) parameters. Figure 1 shows the schematic of the PUMA 560 robot manipulator. The following steps show the systematic derivation of the D-H parameters.

- Locate the robot arm
- Label joints
- Determine joint rotation or translation (θ or d)

- Setup base coordinate frames.
- Setup joints coordinate frames.
- Determineα_i, thatα_i, link twist, is the angle between Z_i and Z_{i+1} about an X_i.
- 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.
- Fill up the D-H parameters table. Table 1 shows the D-H parameters for n DOF robot manipulator.

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;

$$R_i^{i-1} = U_{i(\theta_i)} V_{i(\alpha_i)} \tag{2}$$

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

$$U_{i(\theta_i)} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

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

$$V_{i(\theta_i)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \tag{4}$$

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

$$R_n^0 = (U_1 V_1)(U_2 V_2) \dots \dots \dots (U_n V_n) \tag{5}$$

Link i	θ_i(rad)	α_i(rad)	a_i(m)	d_i(m)
1	θ_1	α_1	a_1	d_1
2	θ_2	α_2	a_2	d_2
3	θ_3	α_3	a_3	d_3
....	......	....	....	....
....	....	....	....	....
n	θ_n	n	a_5	d_n

TABLE 1: The Denavit Hartenberg parameter

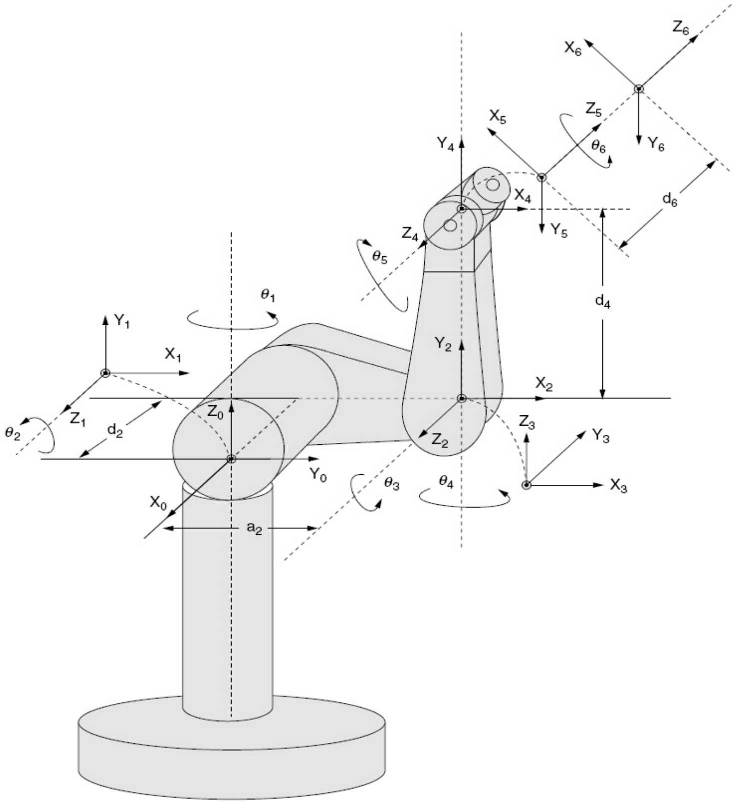

FIGURE 1: D-H notation for a six-degrees-of-freedom PUMA 560 robot manipulator[2]

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]

$$d_n^0 = (U_1 S_1) + (U_1 V_1)(U_2 S_2) + \cdots + (U_1 V_1)(U_2 V_2) \ldots .(U_{n-1} V_{n-1})(U_n S_n) \tag{6}$$

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

$${}_n^0T = {}_1^0T.{}_2^1T.{}_3^2T \ldots \ldots . {}_n^{n-1}T = \begin{bmatrix} R_n^0 & d_n^0 \\ 0 & 1 \end{bmatrix} \tag{7}$$

In PUMA robot manipulator the final transformation matrix is given by

$${}_6^0T = {}_1^0T.{}_2^1T.{}_3^2T \ldots \ldots . {}_6^5T = \begin{bmatrix} R_6^0 & d_6^0 \\ 0 & 1 \end{bmatrix} \tag{8}$$

That R_6^0 and d_6^0 is given by the following matrix

$$R_6^0 = \begin{bmatrix} N_x & B_x & T_x \\ N_y & B_y & T_y \\ N_z & B_z & T_z \end{bmatrix}; \quad d_6^0 = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \tag{9}$$

That ${}_6^0T$ can be determined by

$$ {}_6^0T = \begin{bmatrix} N_x & B_x & T_x & P_x \\ N_y & B_y & T_y & P_y \\ N_z & B_z & T_z & P_z \end{bmatrix} \tag{10} $$

Table 2 shows the PUMA 560 D-H parameters.

Link i	θ_i(rand)	α_i(rand)	α_i(m)	d_i(m)
1	θ_1	$\pi/2$	0	0
2	θ_2	0	0.4318	0
3	θ_3	$-\pi/2$	0.0203	0.15005
4	θ_4	$\pi/2$	0	0.4318
5	θ_5	$-\pi/2$	0	0
6	θ_5	0	0	0

TABLE 2: PUMA 560 robot manipulator DH parameter [4].

As equation 8 the cells of above matrix for PUMA 560 robot manipulator is calculated by following equations:

$$ \begin{aligned} Nx = {} & cos(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) + sin(\theta_1) \times sin(\\ & sin(\theta_2+\theta_3) \times cos(\theta_1)) + sin(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) - cos(\end{aligned} \tag{11} $$

$$ \begin{aligned} Ny = {} & cos(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos \\ & sin(\theta_5) \times sin(\theta_2+\theta_3) \times sin(\theta_1)) + sin(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times si \\ & cos(\theta_4) \times cos(\theta_1)) \end{aligned} \tag{12} $$

$$ \begin{aligned} Nz = {} & cos(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2+\theta_3) - sin(\theta_5) \times cos(\theta_2+\theta_3)) + \\ & sin(\theta_4) \times sin(\theta_2+\theta_3) \end{aligned} \tag{13} $$

$$ \begin{aligned} Bx = {} & -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) + sin(\theta_4) \\ & sin(\theta_5) \times sin(\theta_2+\theta_3) \times cos(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times co \\ & cos(\theta_4) \times sin(\theta_1)) \end{aligned} \tag{14} $$

$$ \begin{aligned} By = {} & -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\theta_1) - sin(\theta_4) \times co \\ & sin(\theta_5) \times sin(\theta_2+\theta_3) \times sin(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\\ & cos(\theta_1)) \end{aligned} \tag{15} $$

$$ \begin{aligned} Bz = {} & -sin(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2+\theta_3) - sin(\theta_5) \times cos(\theta_2+\theta_3)) \\ & sin(\theta_4) \times sin(\theta_2+\theta_3) \end{aligned} \tag{16} $$

(17)

$$Tx = sin(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times cos(\theta_1) + sin(\theta_4) \times sin(\theta_1)) - cos(\theta_3) \times cos(\theta_1)$$

(18)

$$Ty = sin(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos(\theta_1)) - cos(\theta_3) \times sin(\theta_1)$$

(19)

$$Tz = sin(\theta_5) \times cos(\theta_4) \times sin(\theta_2 + \theta_3) + cos(\theta_5) \times cos(\theta_2 + \theta_3) \quad \textbf{(20)}$$

$$Px = 0.4331 \times sin(\theta_2 + \theta_3) \times cos(\theta_1) + 0.0203 \times cos(\theta_2 + \theta_3) \times cos(\theta_1) - 0. sin(\theta_1) + 0.4318 \times cos(\theta_2 cos(\theta_1) \quad \textbf{(21)}$$

$$Py = 0.4331 \times sin(\theta_2 + \theta_3) \times sin(\theta_1) + 0.0203 \times cos(\theta_2 + \theta_3) \times sin(\theta_1) + 0. cos(\theta_1) + 0.4312 \times cos(\theta_2) \times sin(\theta_1) \quad \textbf{(22)}$$

$$Pz = -0.4331 \times cos(\theta_2 + \theta_3) + 0.0203 \times sin(\theta_2 + \theta_3) + 0.4318 \times sin(\theta_2)$$

Dynamic of Robot Manipulator

As mentioned before, dynamic equation is the study of motion with regard to forces. Dynamic modeling is vital for control, mechanical design, and simulation. It is used to describe dynamic parameters and also to describe the relationship between displacement, velocity and acceleration to force acting on robot manipulator. To calculate the dynamic parameters which introduced in the following lines, four algorithms are very important. **Inverse dynamics**, in this algorithm, joint actuators are computed (e.g., force/torque or voltage/current) from endeffector position, velocity, and acceleration. It is used in feed forward control. **Forward dynamics** used to compute the joint acceleration from joint actuators. This algorithm is required for simulations. **The joint-space inertia matrix**, necessary for maps the joint acceleration to the joint actuators. It is used in analysis, feedback control and in some integral part of forward dynamics formulation. **The operational-space inertia matrix**, this algorithm maps the task accelerations to task actuator in Cartesian space. It is required for control of end-effector. The field of dynamic robot manipulator has a wide literature that published in professional journals and established textbooks[1, 6].

Several different methods are available to compute robot manipulator dynamic equations. These methods include the Newton-Euler (N-E) methodology, the Lagrange-Euler (L-E) method, and Kane's methodology[1]. The Newton-Euler methodology is based on Newton's second law and several different researchers are signifying to develop this method[1]. This equation can be described the behavior of a robot manipulator link-by-link and joint-by-joint from base to endeffector, called forward recursion and transfer the essential information from end-effector to base frame, called backward recursive. The literature on Euler-Lagrange's is vast but a good starting point to learn about it is in[1]. Euler-Lagrange is a method based on calculation kinetic energy. Calculate the dynamic equation robot manipulator using E-L method is easier because this equation is derivation of nonlinear coupled and quadratic differential equations. The Kane's method was introduced in 1961 by Professor Thomas Kane[1, 6]. This method used to calculate the dynamic equation of motion without any differentiation between kinetic and potential energy functions. The equation of a multi degrees of freedom (DOF) robot manipulator is calculated by the following equation[6]:

$$M(q)\ddot{q} + N(q,\dot{q}) = \tau \quad \textbf{(23)}$$

Where τ is $n \times 1$ vector of actuation torque, M (q) is $n \times n$ symmetric and positive define inertia matrix, $N(q, \dot{q})$ is the vector of nonlinearity term, and q is $n \times 1$ position vector. In equation 23 if vector of nonlinearity term derive as Centrifugal, Coriolis and Gravity terms, as a result robot manipulator dynamic equation can also be written as [80]:

$$N(q,\dot{q}) = V(q,\dot{q}) + G(q) \tag{24}$$

$$V(q,\dot{q}) = B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 \tag{25}$$

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) \tag{26}$$

Where, $B(q)$ is matrix of coriolis torques, $C(q)$ is matrix of centrifugal torque, $[\dot{q}\,\dot{q}]$ is vector of joint velocity that it can give by: $[\dot{q}_1.\dot{q}_2, \dot{q}_1.\dot{q}_3, \ldots .., \dot{q}_1.\dot{q}_n, \dot{q}_2.\dot{q}_3, \ldots ..]^T$, and $[\dot{q}]^2$ is vector, that it can given by: $[\dot{q}_1{}^2, \dot{q}_2{}^2, \dot{q}_3{}^2, \ldots .]^T$.

In robot manipulator dynamic part the inputs are torques and the outputs are actual displacements, as a result in (26) it can be written as [1, 6, 73-74];

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \tag{27}$$

To implementation (27) the first step is implement the kinetic energy matrix (M) parameters by used of Lagrange's formulation. The second step is implementing the Coriolis and Centrifugal matrix which they can calculate by partial derivatives of kinetic energy. The last step to implement the dynamic equation of robot manipulator is to find the gravity vector by performing the summation of Lagrange's formulation. The kinetic energy equation (M) is a $n \times n$ symmetric matrix that can be calculated by the following equation;

$$M(\theta) = m_1 J_{v1}^T J_{v1} + J_{\omega 1}^{TC1} I_1 J_{\omega 1} + m_2 J_{v2}^T J_{v2} + J_{\omega 2}^{TC2} I_2 J_{\omega 2} + m_3 J_{v3}^T J_{v3} + J_{\omega 3}^{TC3} I_3 J_{\omega 3} + n \\ J_{\omega 4}^{TC4} I_4 J_{\omega 4} + m_5 J_{v5}^T J_{v5} + J_{\omega 5}^{TC5} I_5 J_{\omega 5} + m_6 J_{v6}^T J_{v6} + J_{\omega 6}^{TC6} I_6 J_{\omega 6} \tag{28}$$

As mentioned above the kinetic energy matrix in n DOF is a $n \times n$ matrix that can be calculated by the following matrix [1, 6]

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & \cdots & \cdots. & \cdots.. & M_{1n} \\ M_{21} & \cdots & \cdots & \cdots. & \cdots.. & M_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ M_{n,1} & \cdots & \cdots & \cdots. & \cdots & M_{n,n} \end{bmatrix} \tag{29}$$

The Coriolis matrix (B) is a $n \times \frac{n(n-1)}{2}$ matrix which calculated as follows;

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & \cdots & b_{11n} & b_{123} & \cdots & b_{12n} & \cdots & \cdots & b_{1,n-1,n} \\ b_{212} & \cdots & \cdots & b_{21n} & b_{223} & \cdots & \cdots & \cdots & \cdots & b_{2,n-1,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ b_{n,1,2} & \cdots & \cdots & b_{n,1,n} & \cdots & \cdots & \cdots & \cdots & \cdots & b_{n,n-1,n} \end{bmatrix} \tag{30}$$

and the Centrifugal matrix (C) is a $n \times n$ matrix;

$$C(q) = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \tag{31}$$

And last the Gravity vector (G) is a $n \times 1$ vector;

$$G(q) = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \tag{32}$$

To position control of robot manipulator, the second three axes are locked the dynamic equation of PUMA robot manipulator is given by [70-73];

$$M(\ddot{\theta})\begin{bmatrix}\ddot{\theta}1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix}+\mathbf{B}(\theta)\begin{bmatrix}\dot{\theta}_1\dot{\theta}_2\\ \dot{\theta}_1\dot{\theta}_3\\ \dot{\theta}_2\dot{\theta}_3\end{bmatrix}+C(\theta)\begin{bmatrix}\dot{\theta}_1^2\\ \dot{\theta}_2^2\\ \dot{\theta}_3^2\end{bmatrix}+G(\theta)=\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} \tag{33}$$

Where

$$M(q)=\begin{bmatrix}M_{11} & M_{12} & M_{13} & 0 & 0 & 0\\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0\\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0\\ 0 & 0 & 0 & M_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & M_{55} & 0\\ 0 & 0 & 0 & 0 & 0 & M_{66}\end{bmatrix} \tag{34}$$

$\boldsymbol{M}$ is computed as

$$\begin{aligned}&M_{11} =\\ &I_{m1}+I_1+I_3\times\cos(\theta_2)\cos(\theta_2\ )+I_7\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\ \ +\ I_{10}\sin(\theta_2+\theta_\\ &I_{11}\sin(\theta_2)\cos(\theta_2)+I_{21}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)+2+[I_5\cos(\theta_2)\sin(\theta_2+\theta_3)\\ &I_{12}\cos(\theta_2)\cos(\theta_2+\theta_3)+I_{15}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)+I_{16}\cos(\theta_2)\sin(\theta_2+\theta_3\\ &\theta_3)\cos(\theta_2+\theta_3)\end{aligned} \tag{35}$$

$$M_{12}\ =I_4\sin(\theta_2)+\ I_8\cos(\theta_2+\theta_3)+I_9cos(\theta_2)+I_{13}\sin(\theta_2+\theta_3)-I_{18}\cos(\theta_2+ \tag{36}$$

$$M_{13}\ =I_8\cos(\theta_2+\theta_3)\ +\ I_{13}\sin(\theta_2+\theta_3)-I_{18}\cos(\theta_2+\theta_3) \tag{37}$$

$$M_{22}\ =I_{m2}+I_2+I_6+2[I_5\sin(\theta_3)+\ I_{12}\cos(\theta_2)+I_{15}+I_{16}\sin(\theta_3) \tag{38}$$

$$M_{23}\ =I_5\sin(\theta_3)+\ I_6+I_{12}\cos(\theta_3)+I_{16}\sin(\theta_3)+2I_{15} \tag{39}$$

$$M_{33}\ =I_{m3}+I_6+2I_{15} \tag{40}$$

$$M_{35}\ =I_{15}+I_{17} \tag{41}$$

$$M_{44}\ =I_{m4}+I_{14} \tag{42}$$

$$M_{55}\ =I_{m5}+I_{17} \tag{43}$$

$$M_{66}\ =I_{m6}+I_{23} \tag{44}$$

$$M_{21}\ =M_{12}\ ,M_{31}\ =M_{13}\ and\ M_{32}=\ M_{23} \tag{45}$$

and Corilios ($\boldsymbol{B}$) matrix is calculated as the following

$$B(q)=\begin{bmatrix}b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 & 0\\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix} \tag{46}$$

Where,

$$\begin{aligned}&b_{112}\ =2\big[-I_3\sin(\theta_2)\cos(\theta_2)+\ I_5\cos(\theta_2+\theta_2+\theta_3)+I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta\\ &\theta_2+\theta_3)-I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3)+I_{16}\cos(\theta_2+\theta_2+\theta_3)+\ I_{21}\sin(\theta_2+\\ &\theta_3)+I_{22}(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3))\big]+I_{10}(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\\ &2\sin(\theta_2)\sin(\theta_2))\end{aligned} \tag{47}$$

$$b_{113} = 2\big[\, I_5\cos(\theta_2)\cos(\theta_2+\theta_3) + I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{12}\cos(\theta_2)\sin \\ I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{16}\cos(\theta_2)\cos(\theta_2+\theta_3) + I_{21}\sin(\theta_2+\theta_3)\cos(\\ I_{22}\big(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\big)\big] + I_{10}\big(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\big) \tag{48}$$

$$b_{115} = 2[-\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{16}\cos(\\ \theta_3) + I_{22}\cos(\theta_2+\theta_3)\cos(\theta_2+\theta_3)] \tag{49}$$

$$b_{123} = 2[-I_8\sin(\theta_2+\theta_3) + I_{13}\cos(\theta_2+\theta_3) + I_{18}\sin(\theta_2+\theta_3)] \tag{50}$$

$$b_{214} = I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta_3) + 2I_{20}\sin(\theta_2+\theta_3)(1-0.5) \tag{51}$$

$$b_{223} = 2[-I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3)] \tag{52}$$

$$b_{235} = 2[I_{16}\cos(\theta_3) + I_{22}] \tag{53}$$

$$b_{314} = 2[I_{20}\sin(\theta_2+\theta_3)(1-0.5)] + I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta_3) \tag{54}$$

$$b_{412} = b_{214} = -[I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta_3) + 2I_{20}\sin(\theta_2+\theta_3)(1-0.5) \tag{55}$$

$$b_{413} = -b_{314} = -2[I_{20}\sin(\theta_2+\theta_3)(1-0.5)] + I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta \tag{56}$$

$$b_{415} = -I_{20}\sin(\theta_2+\theta_3) - I_{17}\sin(\theta_2+\theta_3) \tag{57}$$

$$b_{514} = -b_{415} = I_{20}\sin(\theta_2+\theta_3) + I_{17}\sin(\theta_2+\theta_3) \tag{58}$$

consequently coriolis matrix is shown as bellows;

$$B(q).\dot{q}\dot{q} = \begin{bmatrix} b_{112}.\dot{q}_1\dot{q}_2 + b_{113}.\dot{q}_1\dot{q}_3 + 0 + b_{123}.\dot{q}_2\dot{q}_3 \\ 0 + b_{223}.\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}.\dot{q}_1\dot{q}_2 + b_{413}.\dot{q}_1\dot{q}_3 + 0 \\ 0 \\ 0 \end{bmatrix} \tag{59}$$

Moreover Centrifugal ($\boldsymbol{C}$) matrix is demonstrated as

$$C(q) = \begin{bmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{21} & 0 & C_{23} & 0 & 0 & 0 \\ C_{31} & C_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ C_{51} & C_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{60}$$

Where,

$$c_{12} = I_4\cos(\theta_2) - I_8\sin(\theta_2+\theta_3) - I_9\sin(\theta_2) + I_{13}\cos(\theta_2+\theta_3) + I_{18}\sin(\theta_2+ \tag{61}$$

$$c_{13} = 0.5b_{123} = -I_8\sin(\theta_2+\theta_3) + I_{13}\cos(\theta_2+\theta_3) + I_{18}\sin(\theta_2+\theta_3) \tag{62}$$

$$c_{21} = -0.5b_{112} = I_3\sin(\theta_2)\cos(\theta_2) - I_5\cos(\theta_2+\theta_2+\theta_3) - I_7\sin(\theta_2+\theta_3)\mathrm{co}$$
$$I_{12}\sin(\theta_2+\theta_2+\theta_3) + I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2+\theta_2+\theta_3) -$$
$$\theta_3)\cos(\theta_2+\theta_3) - I_{22}\left(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\right) - 0.5I_{10}\left(1-2\sin(\theta_2+\right.$$
$$\left.\theta_3)\right) - 0.5I_{11}\left(1-2\sin(\theta_2)\sin(\theta_2)\right) \tag{63}$$

$$c_{22} = 0.5b_{223} = -I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3) \tag{64}$$

$$c_{23} = -0.5b_{113} = -I_5\cos(\theta_2)\cos(\theta_2+\theta_3) - I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{12}$$
$$\theta_2) - I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2)\cos(\theta_2+\theta_3) - I_{21}\sin(\theta_2+\theta_3$$
$$I_{22}\left(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\right) - 0.5I_{10}\left(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\right) \tag{65}$$

$$c_{31} = -c_{23} = I_{12}\sin(\theta_3) - I_5\cos(\theta_3) - I_{16}\cos(\theta_3) \tag{66}$$

$$c_{32} = -0.5b_{115} =$$
$$\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2)\cos(\theta_2 +$$
$$I_{22}\cos(\theta_2+\theta_3)\cos(\theta_2+\theta_3) \tag{67}$$

$$c_{52} = -0.5b_{225} = -I_{16}\cos(\theta_3) - I_{22} \tag{68}$$

In this research $q_4 = q_5 = q_6 = 0$, as a result

$$C(q).\dot{q}^{z} = \begin{bmatrix} c_{112}.\dot{q}_2^{z} + c_{13}.\dot{q}_3^{z} \\ c_{21}.\dot{q}_1^{z} + c_{23}.\dot{q}_3^{z} \\ c_{13}.\dot{q}_1^{z} + c_{32}.\dot{q}_2^{z} \\ 0 \\ c_{51}.\dot{q}_1^{z} + c_{52}.\dot{q}_2^{z} \\ 0 \end{bmatrix} \tag{69}$$

Gravity (G) Matrix can be written as

$$G(q) = \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{70}$$

Where,

$$G_2 = g_1\cos(\theta_2) + g_2\sin(\theta_2+\theta_3) + g_3\sin(\theta_2) + g_4\cos(\theta_2+\theta_3) + g_5\sin(\theta_2 + \tag{71}$$

$$G_3 = g_2\sin(\theta_2+\theta_3) + g_4\cos(\theta_2+\theta_3) + g_5\sin(\theta_2+\theta_3) \tag{72}$$

$$G_5 = g_5\sin(\theta_2+\theta_3) \tag{73}$$

Suppose $\ddot{q}$ is written as follows

$$\ddot{q} = M^{-1}(q).\{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{74}$$

and K is introduced as

$$K = \{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \quad \textbf{(75)}$$

$\ddot{q}$ can be written as

$$\ddot{q} = M^{-1}(q).K \quad \textbf{(76)}$$

Therefore K for PUMA robot manipulator is calculated by the following equations

$$K_1 = \tau_1 - [\, b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\,\dot{q}_2\dot{q}_3] - [\, C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2] - g_1 \quad \textbf{(77)}$$

$$K_2 = \tau_2 - [\, b_{223}\dot{q}_2\dot{q}_3] - [\, C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2] - g_2 \quad \textbf{(78)}$$

$$K_3 = \tau_3 - [C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2] - g_3 \quad \textbf{(79)}$$

$$K_4 = \tau_4 - [\, b_{412}\dot{q}_1\dot{q}_2 + b_{413}\,\dot{q}_1\dot{q}_3] - g_4 \quad \textbf{(80)}$$

$$K_5 = \tau_5 - [\, C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2] - g_5 \quad \textbf{(81)}$$

$$K_6 = \tau_6 \quad \textbf{(82)}$$

An information about inertial constant and gravitational constant are shown in Tables 3 and 4 based on [73-74].

$I_1 = 1.43 \pm 0.05$	$I_2 = 1.75 \pm 0.07$
$I_3 = 1.38 \pm 0.05$	$I_4 = 0.69 \pm 0.02$
$I_5 = 0.372 \pm 0.031$	$I_6 = 0.333 \pm 0.016$
$I_7 = 0.298 \pm 0.029$	$I_8 = -0.134 \pm 0.014$
$I_9 = 0.0238 \pm 0.012$	$I_{10} = -0.0213 \pm 0.0022$
$I_{11} = -0.0142 \pm 0.0070$	$I_{12} = -0.011 \pm 0.0011$
$I_{13} = -0.00379 \pm 0.0009$	$I_{14} = 0.00164 \pm 0.000070$
$I_{15} = 0.00125 \pm 0.0003$	$I_{16} = 0.00124 \pm 0.0003$
$I_{17} = 0.000642 \pm 0.0003$	$I_{18} = 0.000431 \pm 0.00013$
$I_{19} = 0.0003 \pm 0.0014$	$I_{20} = -0.000202 \pm 0.0008$
$I_{21} = -0.0001 \pm 0.0006$	$I_{22} = -0.000058 \pm 0.00001$
$I_{23} = 0.00004 \pm 0.00002$	$I_{m1} = 1.14 \pm 0.27$
$I_{m2} = 4.71 \pm 0.54$	$I_{m3} = 0.827 \pm 0.093$
$I_{m4} = 0.2 \pm 0.016$	$I_{m5} = 0.179 \pm 0.014$
$I_{m6} = 0.193 \pm 0.016$	

TABLE 3: Inertial constant reference (*Kg.m*^{2})

$g_1 = -37.2 \pm 0.5$	$g_2 = -8.44 \pm 0.20$
$g_3 = 1.02 \pm 0.50$	$g_4 = 0.249 \pm 0.025$
$g_5 = -0.0282 \pm 0.0056$	

TABLE 4: Gravitational constant (*N.m*)

3. FUZZY LOGIC ALGORITHM AND APPLICATION TO CLASSICAL CONTROLLER

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. The first significant work in automatic control was James Watt's centrifugal governor for the speed control in motor engine in eighteenth century[2]. In recent years, artificial intelligence theory has been used in classical control systems [62-69]. Neural network, fuzzy logic, and neuro-fuzzy are synergically combined with nonlinear classical controller and used in nonlinear, time variant, and uncertainty plant (e.g., robot manipulator). Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control nonlinear, uncertain, and noisy systems. This method is free of some model-based techniques as in classical controllers. As mentioned that fuzzy logic application is not only limited to the modelling of nonlinear systems [31-36]but also this method can help engineers to design easier controller. Control robot arm manipulators using classical controllers are based on manipulator dynamic model. These controllers often have many problems for modelling [8]. Conventional controllers require accurate information of dynamic model of robot manipulator, but these models are multi-input, multi-output and non-linear and calculate accurate model can be very difficult. When the system model is unknown or when it is known but complicated, it is difficult or impossible to use classical mathematics to process this model[32]. The main reasons to use fuzzy logic technology are able to give approximate recommended solution for unclear and complicated systems to easy understanding and flexible. Fuzzy logic provides a method which is able to model a controller for nonlinear plant with a set of IF-THEN rules, or it can identify the control actions and describe them by using fuzzy rules. It should be mentioned that application of fuzzy logic is not limited to a system that's difficult for modeling, but it can be used in clear systems that have complicated mathematics models because most of the time it can be shortened in design but there is no high quality design just sometimes we can find design with high quality. Besides using fuzzy logic in the main controller of a control loop, it can be used to design adaptive control, tuning parameters, working in a parallel with the classical and non classical control method [32]. The applications of artificial intelligence such as neural networks and fuzzy logic in modelling and control are significantly growing especially in recent years. For instance, the applications of artificial intelligence, neural networks and fuzzy logic, on robot arm control have reported in [37-39]. Wai et al. [37-38]have proposed a fuzzy neural network (FNN) optimal control system to learn a nonlinear function in the optimal control law. This controller is divided into three main groups: arterial intelligence controller (fuzzy neural network) which it is used to compensate the system's nonlinearity and improves by adaptive method, robust controller to reduce the error and optimal controller which is the main part of this controller. Mohan and Bhanot [40] have addressed comparative study between some adaptive fuzzy, and a new hybrid fuzzy control algorithm for manipulator control. They found that self-organizing fuzzy logic controller and proposed hybrid integrator fuzzy give the best performance as well as simple structure. Research on combinations of fuzzy logic systems with sliding mode method is significantly growing as nonlinear control applications. For instance, the applications of fuzzy logic on sliding mode controller have reported in [7, 24-30, 41-45]. Research on applied fuzzy logic methodology in sliding mode controller (FSMC) to reduce or eliminate the high frequency oscillation (chattering), to compensate the unknown system dynamics and also to adjust the linear sliding surface slope in pure sliding mode controller considerably improves the robot manipulator control process [42-43]. H.Temeltas [46] has proposed fuzzy adaption techniques for SMC to achieve robust tracking of nonlinear systems and solves the chattering problem. Conversely system's performance is better than sliding mode controller; it is depended on nonlinear dynamic equqation. C. L. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode control based on *N* fuzzy based linear state-space to estimate the uncertainties. A multi-input multi-output FSMC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a robot manipulator [42]. Investigation on applied sliding mode methodology in fuzzy logic controller (SMFC) to reduce the fuzzy rules and refine the stability of close loop system in fuzzy logic controller has grown specially in recent years as the robot manipulator control [23]; [48-50]. Lhee et al. [48]have presented a fuzzy logic controller based on sliding mode controller to more formalize and boundary layer thickness. Emami *et al.* [51]have proposed a fuzzy logic approximate inside the boundary layer. H.K.Lee *et al.* [52] have presented self tuning SMFC to reduce the fuzzy rules, increase the stability and to adjust control parameters control automatically.

Adaptive Methodology

In various dynamic parameters systems that need to be training on-line adaptive control methodology is used [62-69]. Adaptive control methodology can be classified into two main groups, namely, traditional adaptive method and fuzzy adaptive method. Fuzzy adaptive method is used in systems which want to training parameters by expert knowledge. Traditional adaptive method is used in systems which some dynamic parameters are known. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, adaptive method is applied to artificial sliding mode controller [62-69]. F Y Hsu et al. [54]have presented adaptive fuzzy sliding mode control which can update fuzzy rules to compensate nonlinear parameters and guarantee the stability robot manipulator controller. Y.C. Hsueh et al. [43] have presented self tuning sliding mode controller which can resolve the chattering problem without to using saturation function. For nonlinear dynamic systems (e.g., robot manipulators) with various parameters, adaptive control technique can train the dynamic parameter to have an acceptable controller performance. Calculate several scale factors are common challenge in classical sliding mode controller and fuzzy logic controller, as a result it is used to adjust and tune coefficient. Research on adaptive fuzzy control is significantly growing, for instance, different adaptive fuzzy controllers have been reported in [40, 55-57]. Research on adaptive fuzzy sliding mode controller is significantly growing as many applications and it can caused to improve the tracking performance by online tuning the parameters. The adaptive sliding mode controller is used to estimate the unknown dynamic parameters and external disturbances. For instance, the applications of adaptive fuzzy sliding mode controller to control the robot manipulators have been reported in [24, 29, 45]. Generally, adaptive fuzzy sliding mode control of robot manipulator is classified into two main groups' i.e. multi-input multi-output (MIMO) and single-input single-output (SISO) fuzzy systems. Yoo and Ham [58]have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. In $\boldsymbol{n-DOF}$ robot manipulator with $\boldsymbol{k}$ membership function for each input variable, the number of fuzzy rules for each joint is equal to $\boldsymbol{3k^{2n}}$ that causes to high computation load and also this controller has chattering. This method can only tune the consequence part of the fuzzy rules. Medhafer et al. [59] have proposed an indirect adaptive fuzzy sliding mode controller to control robot manipulator. This MIMO algorithm, applies to estimate the nonlinear dynamic parameters. If each input variable have $\boldsymbol{K_2}$ membership functions, the number of fuzzy rules for each joint is $\boldsymbol{(m+1)K_2^m+K_2}$. Compared with the previous algorithm the number of fuzzy rules have reduced by introducing the sliding surface as inputs of fuzzy systems. Y. Guo and P. Y. Woo [60]have proposed a SISO fuzzy system compensate and reduce the chattering. First suppose each input variable with K_2 membership function the number of fuzzy rules for each joint is K_2 which decreases the fuzzy rules and the chattering is also removed. C. M. Lin and C. F. Hsu [61] can tune both systems by fuzzy rules. In this method the number of fuzzy rules equal to K_2 with low computational load but it has chattering. Shahnazi et al., have proposed a SISO PI direct adaptive fuzzy sliding mode controller based on Lin and Hsu algorithm to reduce or eliminate chattering with K_2 fuzzy rules numbers. The bounds of PI controller and the parameters are online adjusted by low adaption computation [44]. Table 2.2 is illustrated a comparison between computed torque controller[15-16], sliding mode controller[1, 6, 17-23, 26], fuzzy logic controller (FLC)[31-40], applied sliding mode in fuzzy logic controller (SMFC)[23, 48-50], applied fuzzy logic method in sliding mode controller (FSMC)[54-55, 60-61] and adaptive fuzzy sliding mode controller.

Foundation and Basic Definition of Fuzzy Logic Methodology

This section provides a review about foundation of fuzzy logic based on [32, 53]. Supposed that U is the universe of discourse and x is the element of U, therefore, a crisp set can be defined as a set which consists of different elements (x) will all or no membership in a set. A fuzzy set is a set that each element has a membership grade, therefore it can be written by the following definition;

$$A = \{x, \mu_A(x) | x \in X\}; A \in U \qquad \textbf{(83)}$$

Where an element of universe of discourse is x, μ_A is the membership function (MF) of fuzzy set. The membership function $(\mu_A(x))$ of fuzzy set A must have a value between zero and one. If the membership function $\mu_A(x)$ value equal to zero or one, this set change to a crisp set but if it has a value between zero and one, it is a fuzzy set. Defining membership function for fuzzy sets has divided into two main groups; namely; numerical and functional method, which in numerical method each number has different degrees of membership function and functional method used standard functions in fuzzy sets. The membership function which is often used in practical applications includes triangular form, trapezoidal form, bell-shaped form, and Gaussian form. A Trapezoidal membership function of fuzzy set is defined by the following equation

$$\mu_{F(x)} = \begin{cases} 0 \;, & x < a \\ \dfrac{x-a}{b-a}, & a \le x < b \\ \dfrac{d-x}{d-c}, & c \le x < d \\ 0 \;, & x > d \end{cases} \qquad \textbf{(84)}$$

A Triangular membership function of fuzzy set is defined by the following equation

$$\mu_{F(x)} = \begin{cases} 0 \;, & x < a \\ \dfrac{x-a}{b-a}, & a \le x < b \\ \dfrac{c-x}{c-b}, & b \le x \le c \\ 0 \;, & x > c \end{cases} \qquad \textbf{(85)}$$

A Gaussian membership function of fuzzy set is defined by

$$\mu_{F(x)} = e^{\frac{-(x-c_F)^2}{W}} \qquad \textbf{(86)}$$

and a Bell-shaped membership function of fuzzy set is defined by

$$\mu_{F(x)} = \frac{1}{1+(x-c_F)^2} \qquad \textbf{(87)}$$

Figure 2 shows the typical shapes of membership functions in a fuzzy set.

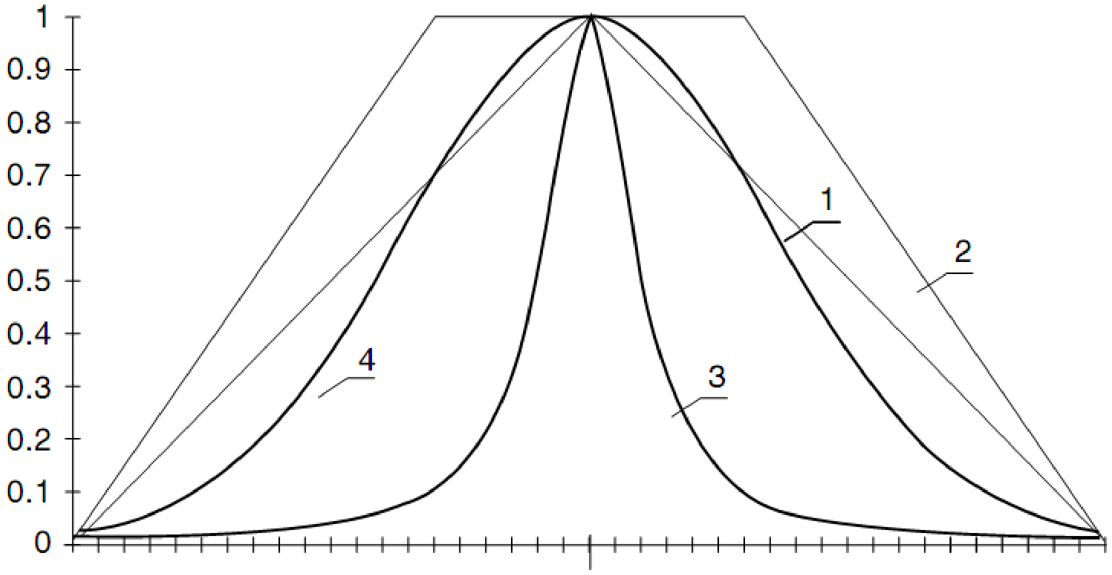

FIGURE 2: Most important membership functions in fuzzy set: 1-Trianglar, 2-Trapezoidal, 3-Gaussian, 4-Bell-shaped [53].

The union of two fuzzy set A and B ($S - norm$) is a new fuzzy set which the new membership function is given by

$$S(a,b) = \mu_{A \cup B(u)} = \max\{\mu_{A(u)}, \mu_{B(u)}\}, \qquad \forall u \in U \tag{88}$$

The intersection of two fuzzy set A and B ($T - norm$) is a new fuzzy set which the new membership function is given by

$$\begin{aligned} T(a,b) &= \mu_{A \cap B(u)} = \min\{\mu_{A(u)}, \mu_{B(u)}\} = \mu_{A(u)} \cdot \mu_{B(u)} \\ &= \max(0, \ \mu_{A(u)} + \mu_{B(u)} - 1) = \begin{cases} \mu_{A(u)} & , if\, \mu_{B(u)} = 1 \\ \mu_{B(u)} & , if\, \mu_{A(u)} = 1 \\ 0 & , if \mu_{B(u)}, \mu_{A(u)} < 1 \end{cases} \end{aligned} \tag{89}$$

In fuzzy set the min operation can resolve the statement $A\ AND\ B$ and can be shown by $min(A, B)$ operation. Using the same reason, the $A\ OR\ B$ operation can be replace by max operation in fuzzy set and at last the $NOT\ A$ operation can be replace by $1 - A$ operation in fuzzy set. The algebraic $product$ of two fuzzy set A and B is the multiplication of the membership functions which is given by the following equation

$$\mu_{A.B(u)} = \mu_{A(u)} \cdot \mu_{B(u)} \tag{90}$$

The algebraic Sum of two fuzzy sets A and B is given by the following equation

$$\mu_{A \dotplus B(u)} = \mu_{A(u)} \cdot \mu_{B(u)} - \mu_{A(u)} \cdot \mu_{B(u)} \tag{91}$$

Linguistic variable can open a wide area to use of fuzzy logic theory in many applications (e.g., control and system identification). In a natural artificial language all numbers replaced by words or sentences. In Figure 3 the linguistic variable is torque and the linguistic values are Low,$Medium$ and $High$.

$If - then$ Rule statements are used to formulate the condition statements in fuzzy logic. A single fuzzy $If - then$ rule can be written by

$$If\ x\ is\ A\ Then\ y\ is\ B \tag{92}$$

where A and B are the Linguistic values that can be defined by fuzzy set, the $If - part$ of the part of "$x\ is\ A$" is called the antecedent part and the $then - part$ of the part of "$y\ is\ B$" is called the

Consequent or Conclusion part. The antecedent of a fuzzy if-then rule can have multiple parts, which the following rules shows the multiple antecedent rules:

$$if\ e\ is\ NB\ and\ \dot{e}\ is\ ML\ then\ T\ is\ LL \tag{93}$$

where e is error, $\dot{e}$ is change of error, NB is Negative Big, ML is Medium Left, T is torque and LL is Large Left.

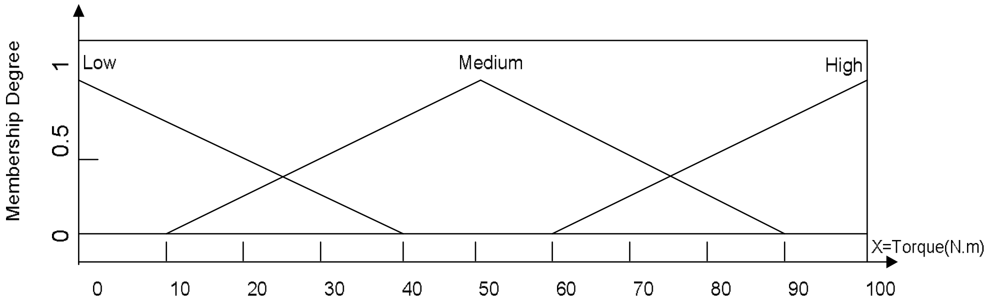

FIGURE 3: Linguistic variable and Linguistic value

$If - then$ rules have three parts, namely, fuzzify inputs, apply fuzzy operator and apply implication method which in fuzzify inputs the fuzzy statements in the antecedent replaced by the degree of membership, apply fuzzy operator used when the antecedent has multiple parts and replaced by single number between 0 to 1, this part is a degree of support for the fuzzy rule, and apply implication method used in consequent of fuzzy rule to replaced by the degree of membership. Figure 4 shows the main three parts in $If - then$ rules.

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. Design Mamdani's fuzzy inference system has 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

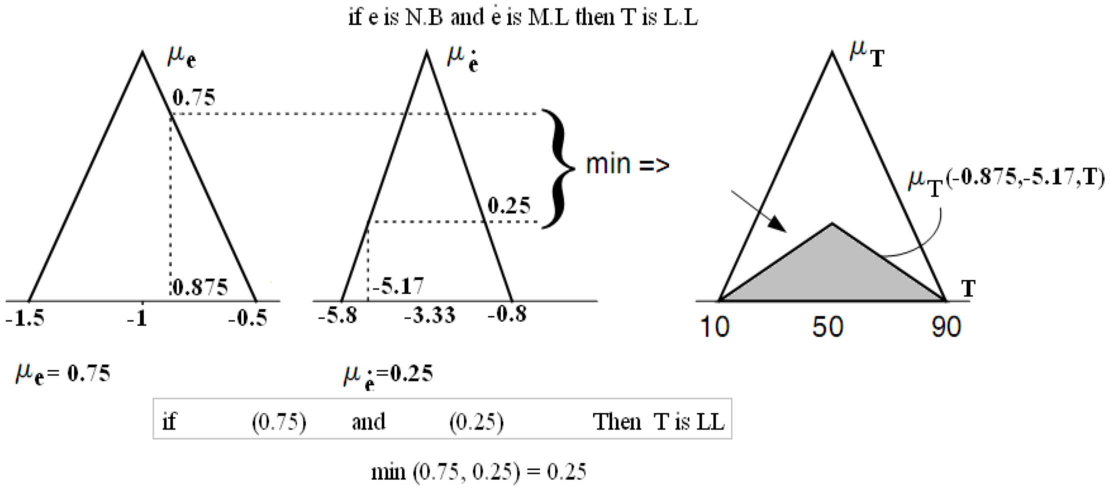

FIGURE 4: Main three parts in IF-THEN rules in fuzzy set

$$\begin{array}{llllll} Mamdani & F.R^1: if & x \text{ is } A \text{ and } y \text{ is } B & then & z \text{ is } C \\ Sugeno & F.R^1: if & x \text{ is } A \text{ and } y \text{ is } B & then & f(x,y) \text{ is } C \end{array} \quad (94)$$

When x and y have crisp values fuzzification calculates the membership degrees for antecedent part. 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

$$\mu_U(x_k,y_k,U)=\mu_{\cup_{i=1}^{r}FR^i}(x_k,y_k,U)=\max\left\{\min_{i=1}^{r}\left[\mu_{R_{pq}}(x_k,y_k),\mu_{p_m}(U)\right]\right\} \quad (95)$$

The Sum-min aggregation defined as below

$$\mu_U(x_k,y_k,U)=\mu_{\cup_{i=1}^{r}FR^i}(x_k,y_k,U)=\sum\min_{i=1}^{r}\left[\mu_{R_{pq}}(x_k,y_k),\mu_{p_m}(U)\right] \quad (96)$$

where r is the number of fuzzy rules activated by x_k and y_k and also $\mu_{\cup_{i=1}^{r}FR^i}(x_k,y_k,U)$ 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

$$COG(x_k,y_k)=\frac{\sum_i U_i\sum_{j=1}^{r}\cdot\mu_u(x_k,y_k,U_i)}{\sum_i\sum_{j=1}^{r}\cdot\mu_u(x_k,y_k,U_i)} \quad (97)$$

and COA method used the following equation to calculate the defuzzification

$$COA(x_k,y_k)=\frac{\sum_i U_i\cdot\mu_u(x_k,y_k,U_i)}{\sum_i\mu_U\cdot(x_k,y_k,U_i)} \quad (98)$$

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, $\mu_U\cdot(x_k,y_k,U_i)$ is the fuzzy set membership function, and r is the number of fuzzy rules.

EXAMPLE: If two fuzzy rules defined by

$F.R^1: if\ e\ is\ NB \quad and \quad \dot{e}\ is\ ML \quad then \quad T\ is\ LL$

$F.R^2: if\ e\ is\ NB \quad and \quad \dot{e}\ is\ FL \quad then \quad T\ is\ LL$

Where e is error, $\dot{e}$ is change of error, T is torque, NB is Negative Big, ML is Medium Left, and LL is Large Left to calculate Mamdani fuzzy inference system we must to do four steps:

Fuzzification is used to determine the membership degrees if all input fuzzy activated by crisp input values $e=-1$ and $\dot{e}=-3.92$, where fuzzy set NB, ML, and FL are defined as below

$$e_{(NB)}=\{(0,-1.5),(0.25,-1.375),(0.5,-1.25),(0.75,-1.125),(1,-1),(0.75,-0.875),(0.5,-0.75),(0.25,-0.625),(0,-0.5)$$

$$\dot{e}_{(ML)}=\{(0,-5.8),(0.25,-5.17),(0.5,-4.55),(0.75,-3.92),(1,-3.3),(0.75,-2.67),(0.5,-2.05),(0.25,-1.42),(0,-0.83)$$

$$\dot{e}_{(FL)}=\{(0,-7.5),(0.25,-6.88),(0.5,-6.25),(0.75,-5.57),(1,-5),(0.75,-4.30),(0.5,-3.92),(0.25,-3.12),(0,-2.5)$$

while

$$T_{(LL)}=\{(0,-123),(0.25,-113.5),(0.5,-104),(0.75,-94.5),(1,-85),(0.75,-75.5),(0.5,-66),(0.25,-56.5),(0,-47)$$

Rule evaluation focuses on operation in the antecedent of the fuzzy rules. This controller used AND fuzzy operation, that it can be defined by $T(a,b)=\mu_{A\cap B(u)}=\min\{\mu_{A(u)},\mu_{B(u)}\}$. The output fuzzy set can be calculated by using individual rule-base inference. So this part focuses on determine the activation degrees of antecedent of $F.R^1$ and $F.R^2$:

$$\mu_{FR_1}=\min\left[\mu_{e(N.B)}(-1),\mu_{\dot{e}(M.L)}(-3.92)\right]=\min[1,0.75]=0.75$$

$\mu_{FR_2} = \min[\mu_{e(N.B)}(-1), \mu_{e(F.L)}(-3.92)] = \min[1, 0.5] = 0.5$

The activation degrees of the consequent parts for $F.R^1$ and $F.R^2$ can be calculated by:

$\mu_{FR_1}(-1, -3.92, T) = \min[\mu_{FR_1}(-1, -3.92), \mu_{T(L.L)}] = \min[0.75, \mu_{T(L.L)}]$

$\mu_{FR_2}(-1, -3.92, T) = \min[\mu_{FR_2}(-1, -3.92), \mu_{T(L.L)}] = \min[0.5, \mu_{T(L.L)}]$

Therefore fuzzy set $T_{L.L(1)}$ and $T_{L.L(2)}$ have nine elements, that can be written by the following form

$$F.F^1(-1, -3.92, T) = \{(0, -123), (0.25, -113.5), (0.5, -104), (0.75, -94.5), (0.75, -85), (0.75, -75.5), (0.5, -66), (0.25, -56.5), (0, -47)$$

$$F.F^2(-1, -3.92, T) = \{(0, -123), (0.25, -113.5), (0.5, -104), (0.5, -94.5), (0.5, -85), (0.5, -75.5), (0.5, -66), (0.25, -56.5), (0, -47)$$

Aggregation of the rule outputs focuses on the aggregation of two fuzzy set into one output fuzzy set. Therefore by using the Max-min aggregation the output of fuzzy set can be calculated:

$\mu_{U_{12}}(-1, -3.92, T) = \mu_{\cup_{i=1}^{2} FR^i}(-1, -3.92, T) = \max\{\mu_{FR}^1(-1, -3.92, T)_{L.L}, \mu_{FR}^2(-1, -3.92, T)_{L.L}\}$

$$U_{12} = \{(0, -123), (0.25, -113.5), (0.5, -104), (0.75, -94.5), (0.75, -85), (0.75, -75.5), (0.5, -66), (0.25, -56.5), (0, -47)\}$$

and by using the Sum aggregation the output of fuzzy set can be calculated by

$$U_{12} = \{(0, -123), (0.5, -113.5), (1, -104), (1, -94.5), (1, -85), (1, -75.5), (1, -66), (0.5, -56.5), (0, -47)\}$$

Defuzzification is the last step to calculate the fuzzy inference system. In this example we use two methods, namely,COA, and COG defuzzification. For Max-min aggregation the COA defuzzification can be calculated by

$$COA = [(0.25 \times -113.5) + (0.5 \times -104) + (0.75 \times -94.5) + (0.75 \times -85) + (0.75 \times -75.5) + (0.5 \times -66) + (0.25 \times -56.5)][0.25 + 0.5 + 0.75 + 0.75 + 0.75 + 0.5 + 0.25]^{-1} = \frac{-318.75}{3.75} = -85$$

For Sum aggregation the COA defuzzification can be calculated by

$$COA = [(0.5 \times -113.5) + (1 \times -104) + (1 \times -94.5) + (1 \times -85) + (1 \times -75.5) + (1 \times -66) + (0.5 \times -56.5)][0.5 + 1 + 1 + 1 + 1 + 1 + 0.5]^{-1} = \frac{-510}{6} = -85$$

For Max-min aggregation the COG defuzzification can be calculated by

$$COG = \frac{[(-113.5 - 56.5)(0.25) + (-104 - 66)(0.5) + (-94.5 - 85 - 75.5)(0.75)}{0.25 + 0.5 + 0.75 + 0.75 + 0.75 + 0.5 + 0.25} = -85$$

For Sum aggregation the COG defuzzification can be calculated by

$$COG = \frac{[(-113.5 - 56.5)(0.5) + (-104 - 94.5 - 85 - 75.5 - 66)(1)}{0.5 + 1 + 1 + 1 + 1 + 1 + 0.5} = -85$$

Fuzzy Controller Structure

One of the most active research areas in the field of fuzzy logic is the fuzzy logic controller (FLC) design because it has a influential tool to control of highly nonlinear, uncertain or noisy systems [53]. 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
- Output defuzzification (fuzzy-to-binary[F/B]conversion).

The basic structure of a fuzzy controller is shown in Figure 5.

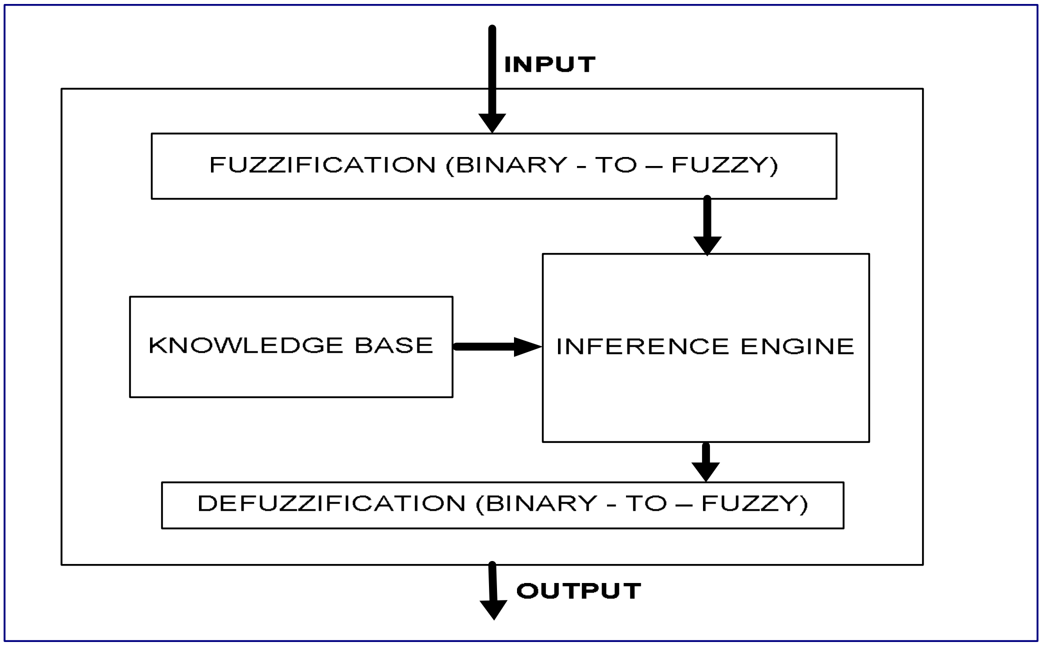
INPUT
FUZZIFICATION (BINARY - TO – FUZZY)
KNOWLEDGE BASE
INFERENCE ENGINE
DEFUZZIFICATION (BINARY - TO – FUZZY)
OUTPUT

FIGURE 5: Structure of Fuzzy Logic Controller (F.L.C)

4. CONCLUSION

In this paper, review of fuzzy logic control is discussed for PUMA robotic manipulators. In this paper, first of all, main subject of modelling PUMA robot manipulator is presented. Second part is focused on review of fuzzy logic methodology and applied to robot manipulator. Pure fuzzy logic controllers have some disadvantages, therefore, in most of design adaptive methodology is applied to main controller for reduce computation load in fuzzy logic controller.

REFERENCE

[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] K. Ogata, *Modern control engineering*: Prentice Hall, 2009.

[4] L. Cheng*, et al.*, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo*, et al.*, *Linear control system analysis and design with MATLAB*: CRC, 2003.

[6] B. Siciliano and O. Khatib, *Springer handbook of robotics*: Springer-Verlag New York Inc, 2008.

[7] I. Boiko*, et al.*, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* vol. 52, pp. 2085-2102, 2007.

[8] J. Wang*, et al.*, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* vol. 122, pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* vol. 2, pp. 171-179, 1986.

[10] H. Zhang and R. P. Paul, "A parallel solution to robot inverse kinematics," 2002, pp. 1140-1145.

[11] J. Kieffer, "A path following algorithm for manipulator inverse kinematics," 2002, pp. 475-480.

[12] Z. Ahmad and A. Guez, "On the solution to the inverse kinematic problem(of robot)," 1990, pp. 1692-1697.

[13] F. T. Cheng*, et al.*, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 27, pp. 332-343, 2002.

[14] M. W. Spong and M. Vidyasagar, *Robot dynamics and control*: Wiley-India, 2009.

[15] A. Vivas and V. Mosquera, "Predictive functional control of a PUMA robot," 2005.

[16] D. Nguyen-Tuong*, et al.*, "Computed torque control with nonparametric regression models," 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo*, et al.*, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* vol. 76, pp. 212-232, 2002.

[19] K. D. Young*, et al.*, "A control engineer's guide to sliding mode control," 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* vol. 48, pp. 2-3, 2001.

[21] J. J. Slotine and S. Sastry, "Tracking control of non-linear systems using sliding surfaces, with application to robot manipulators†," *International Journal of Control,* vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* vol. 40, pp. 421-434, 1984.

[23] R. Palm, "Sliding mode fuzzy control," 2002, pp. 519-526.

[24] C. C. Weng and W. S. Yu, "Adaptive fuzzy sliding mode control for linear time-varying uncertain systems," 2008, pp. 1483-1490.

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics,* vol. 10, pp. 239-263, 2000.

[26] P. Kachroo and M. Tomizuka, "Chattering reduction and error convergence in the sliding-mode control of a class of nonlinear systems," *Automatic Control, IEEE Transactions on,* vol. 41, pp. 1063-1068, 2002.

[27] H. Elmali and N. Olgac, "Implementation of sliding mode control with perturbation estimation (SMCPE)," *Control Systems Technology, IEEE Transactions on,* vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," 2002, pp. 996-1000.

[29] Y. Li and Q. Xu, "Adaptive Sliding Mode Control With Perturbation Estimation and PID Sliding Surface for Motion Tracking of a Piezo-Driven Micromanipulator," *Control Systems Technology, IEEE Transactions on,* vol. 18, pp. 798-810, 2010.

[30] B. Wu*, et al.*, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," 2006, pp. 5-945.

[31] L. A. Zadeh, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems,* vol. 90, pp. 111-127, 1997.

[32] L. Reznik, *Fuzzy controllers*: Butterworth-Heinemann, 1997.

[33] J. Zhou and P. Coiffet, "Fuzzy control of robots," 2002, pp. 1357-1364.

[34] S. Banerjee and P. Y. Woo, "Fuzzy logic control of robot manipulator," 2002, pp. 87-88.

[35] K. Kumbla*, et al.*, "Soft computing for autonomous robotic systems," *Computers and Electrical Engineering,* vol. 26, pp. 5-32, 2000.

[36] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Transactions on systems, man and cybernetics,* vol. 20, pp. 404-418, 1990.

[37] R. J. Wai*, et al.*, "Implementation of artificial intelligent control in single-link flexible robot arm," 2003, pp. 1270-1275.

[38] R. J. Wai and M. C. Lee, "Intelligent optimal control of single-link flexible robot arm," *Industrial Electronics, IEEE Transactions on,* vol. 51, pp. 201-220, 2004.

[39] M. B. Menhaj and M. Rouhani, "A novel neuro-based model reference adaptive control for a two link robot arm," 2002, pp. 47-52.

[40] S. Mohan and S. Bhanot, "Comparative study of some adaptive fuzzy algorithms for manipulator control," *International Journal of Computational Intelligence,* vol. 3, pp. 303–311, 2006.

[41] F. Barrero*, et al.*, "Speed control of induction motors using a novel fuzzy sliding-mode structure," *Fuzzy Systems, IEEE Transactions on,* vol. 10, pp. 375-383, 2002.

[42] Y. C. Hsu and H. A. Malki, "Fuzzy variable structure control for MIMO systems," 2002, pp. 280-285.

[43] Y. C. Hsueh*, et al.*, "Self-tuning sliding mode controller design for a class of nonlinear control systems," 2009, pp. 2337-2342.

[44] R. Shahnazi*, et al.*, "Position control of induction and DC servomotors: a novel adaptive fuzzy PI sliding mode control," *Energy Conversion, IEEE Transactions on,* vol. 23, pp. 138-147, 2008.

[45] C. C. Chiang and C. H. Wu, "Observer-Based Adaptive Fuzzy Sliding Mode Control of Uncertain Multiple-Input Multiple-Output Nonlinear Systems," 2007, pp. 1-6.

[46] H. Temeltas, "A fuzzy adaptation technique for sliding mode controllers," 2002, pp. 110-115.

[47] C. L. Hwang and S. F. Chao, "A fuzzy-model-based variable structure control for robot arms: theory and experiments," 2005, pp. 5252-5258.

[48] C. G. Lhee*, et al.*, "Sliding mode-like fuzzy logic control with self-tuning the dead zone parameters," *Fuzzy Systems, IEEE Transactions on,* vol. 9, pp. 343-348, 2002.

[49] Lhee. C. G., J. S. Park, H. S. Ahn, and D. H. Kim, "Sliding-Like Fuzzy Logic Control with Self-tuning the Dead Zone Parameters," *IEEE International fuzzy systems conference proceeding*, 1999,pp.544-549.

[50] X. Zhang*, et al.*, "Adaptive sliding mode-like fuzzy logic control for high order nonlinear systems," pp. 788-792.

[51] M. R. Emami*, et al.*, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems,* vol. 6, 1998.

[52] H.K.Lee, K.Fms, "A Study on the Design of Self-Tuning Sliding Mode Fuzzy Controller. Domestic conference," *IEEE Conference*, *1994,* vol. 4, pp. 212-218.

[53] Z. Kovacic and S. Bogdan, *Fuzzy controller design: theory and applications*: CRC/Taylor & Francis, 2006.

[54] F. Y. Hsu and L. C. Fu, "Nonlinear control of robot manipulators using adaptive fuzzy sliding mode control," 2002, pp. 156-161.

[55] R. G. Berstecher*, et al.*, "An adaptive fuzzy sliding-mode controller," *Industrial Electronics, IEEE Transactions on,* vol. 48, pp. 18-31, 2002.

[56] V. Kim, "Independent joint adaptive fuzzy control of robot manipulator," 2002, pp. 645-652.

[57] Y. Wang and T. Chai, "Robust adaptive fuzzy observer design in robot arms," 2005, pp. 857-862.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar*, et al.*, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* vol. 1, pp. 23-29, 2006.

[60] Y. Guo and P. Y. Woo, "An adaptive fuzzy sliding mode controller for robotic manipulators," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* vol. 33, pp. 149-159, 2003.

[61] C. M. Lin and C. F. Hsu, "Adaptive fuzzy sliding-mode control for induction servomotor systems," *Energy Conversion, IEEE Transactions on,* vol. 19, pp. 362-368, 2004.

[62] Piltan, F., et al., “Design sliding mode controller for robot manipulator with artificial tunable gain,” Canaidian Journal of pure and applied science, 5 (2): 1573-1579, 2011.

[63] Farzin Piltan, A. R. Salehi and Nasri B Sulaiman.," Design artificial robust control of second order system based on adaptive fuzzy gain scheduling," world applied science journal (WASJ), 13 (5): 1085-1092, 2011

[64] F. Piltan*, et al.*, "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," *Australian Journal of Basic and Applied Sciences, 5(6)*, pp. 509-522, 2011.

[65] Piltan, F., et al., "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[66] Piltan, F., et al., "Design Mathematical Tunable Gain PID-Like Sliding Mode Fuzzy Controller with Minimum Rule Base," International Journal of Robotic and Automation, 2 (3): 146-156, 2011.

[67] Piltan, F., et al., "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[68] Piltan, F., et al., "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

[69] Piltan, F., et al., "Design Adaptive Fuzzy Robust Controllers for Robot Manipulator," World Applied Science Journal, 12 (12): 2317-2329, 2011.

[70] B. S. R. Armstrong, "Dynamics for robot control: friction modeling and ensuring excitation during parameter identification," 1988.

[71] C. L. Clover, "Control system design for robots used in simulating dynamic force and moment interaction in virtual reality applications," 1996.

[72] K. R. Horspool, *Cartesian-space Adaptive Control for Dual-arm Force Control Using Industrial Robots*: University of New Mexico, 2003.

[73] B. Armstrong*, et al.*, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," 2002, pp. 510-518.

[74] P. I. Corke and B. Armstrong-Helouvry, "A search for consensus among model parameters reported for the PUMA 560 robot," 2002, pp. 1608-1613.