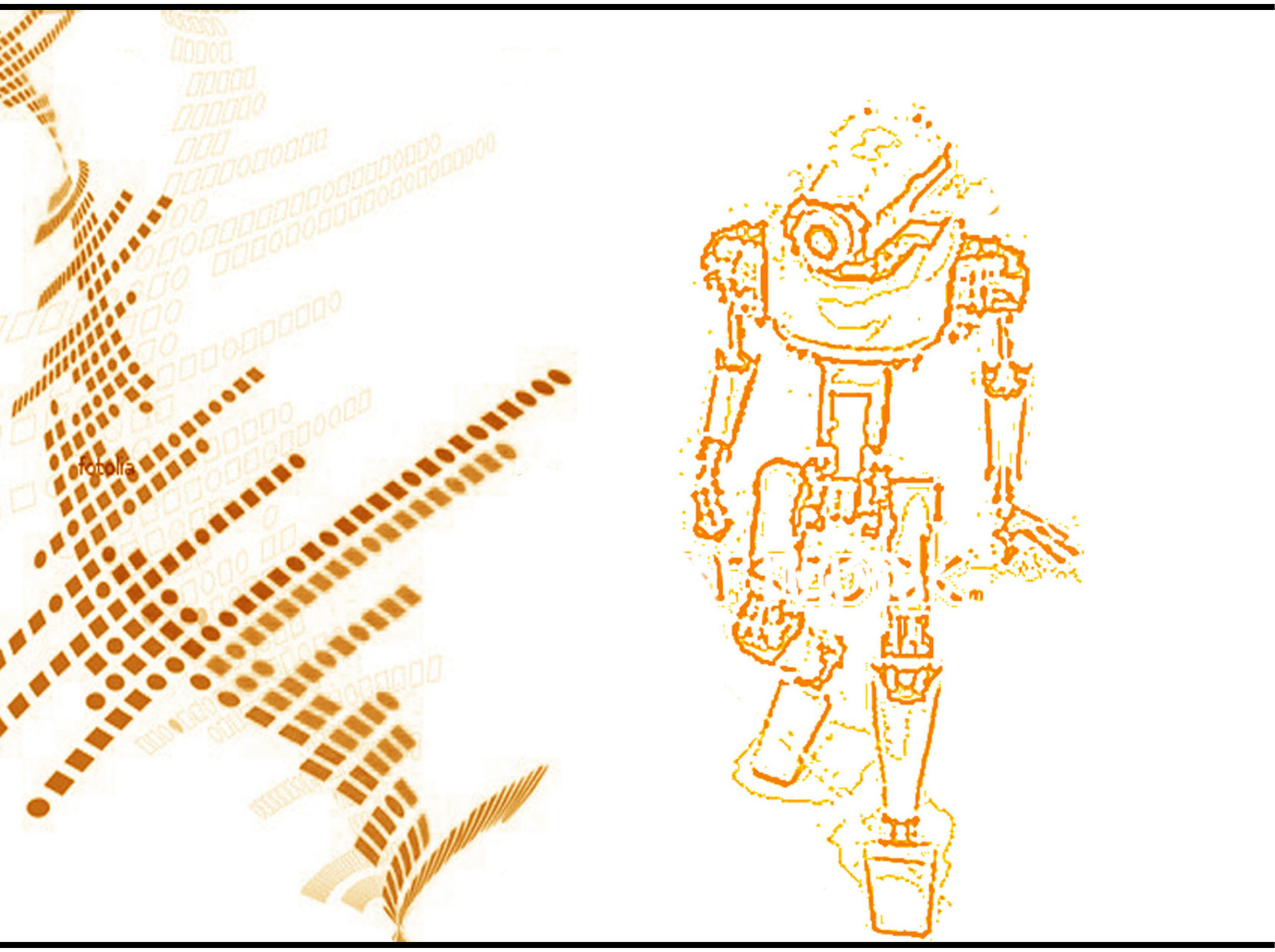
Volume 2 ▪ Issue 5 ▪ November / December 2011

INTERNATIONAL JOURNAL OF

ROBOTICS AND AUTOMATION (IJRA)

ISSN : 2180-1312
Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org



INTERNATIONAL JOURNAL OF ROBOTICS AND AUTOMATION (IJRA)

VOLUME 2, ISSUE 5, 2011

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 2180-1312

I International Journal of Robotics and Automation (IJRA) is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJRA Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ROBOTICS AND AUTOMATION (IJRA)

Book: Volume 2, Issue 5, November / December 2011
Publishing Date: 15-12- 2011
ISSN (Online): 2180-1312

IJRA Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2011

EDITORIAL PREFACE

Robots are becoming part of people's everyday social lives - and will increasingly become so. In future years, robots may become caretaking assistants for the elderly or academic tutors for our children, or medical assistants, day care assistants, or psychological counselors. Robots may become our co-workers in factories and offices, or maids in our homes. It is the *Fifth* Issue of Volume *Two* of International Journal of Robotics and Automation (IJRA). IJRA published six times in a year and it is being peer reviewed to very high International standards.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with Volume 3, 2012, IJRA appear with more focused issues. Besides normal publications, IJRA intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

IJRA looks to the different aspects like sensors in robot, control systems, manipulators, power supplies and software. IJRA is aiming to push the frontier of robotics into a new dimension, in which motion and intelligence play equally important roles. IJRA scope includes systems, dynamics, control, simulation, automation engineering, robotics programming, software and hardware designing for robots, artificial intelligence in robotics and automation, industrial robots, automation, manufacturing, and social implications etc. IJRA cover the all aspect relating to the robots and automation.

The IJRA is a refereed journal aims in providing a platform to researchers, scientists, engineers and practitioners throughout the world to publish the latest achievement, future challenges and exciting applications of intelligent and autonomous robots. IJRA open access publications have greatly speeded the pace of development in the robotics and automation field. IJRA objective is to publish articles that are not only technically proficient but also contains state of the art ideas and problems for international readership.

In order to position IJRA as one of the top International journal in robotics, a group of highly valuable and senior International scholars are serving its Editorial Board who ensures that each issue must publish qualitative research articles from International research communities relevant to signal processing fields.

IJRA editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJRA. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJRA provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Robotics and Automation (IJRA)

Editorial Board

TABLE OF CONTENTS

Volume 2, Issue 5, November / December 2011

Pages

Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review

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

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

Mehdi Rashidi 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

Zahra Tajpaikar 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

Payman Ferdosali 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

Refer to the research, review of sliding mode controller is introduced and application to robot manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Regarding to the positive points in sliding mode controller, fuzzy logic controller and adaptive method, the output in most of research have improved. Each method by adding to the previous algorithm has covered negative points. Obviously robot manipulator is nonlinear, and a number of parameters are uncertain, this research focuses on comparison between sliding mode algorithm which analyzed by many researcher. Sliding mode controller (SMC) is one of the nonlinear robust controllers which it can be used in uncertainty nonlinear dynamic systems. This nonlinear controller has two challenges namely nonlinear dynamic equivalent part and chattering phenomenon. A review of sliding mode controller for robot manipulator will be investigated in this research.

Keywords: Robotic System, Nonlinear System, Robust Controller, Sliding Mode Controller.

1. INTRODUCTION

There are a lot of control methodologies that can be used for control of robot manipulators. These range of various controllers applied from linear to nonlinear, to lots of non-classical non-linear and adaptive non-classical non-linear. In this paper an attempted has been made to do a review of sliding mode control (SMC) for robotics manipulator.

Non linear control methodologies are more general because they can be used in linear and non linear systems. These controllers can solve different problems such as, invariance to system uncertainties, and resistance to the external disturbance. The most common non linear methodologies that have been proposed to solve the control problem consist of the following methodologies: feedback linearization control methodology, passivity-based control methodology, sliding mode control methodology, robust Lyapunov-based control methodology, adaptive control methodology, and artificial intelligence- based methodology [2].

Sliding mode controller (SMC) is a powerful nonlinear controller which has been analysed by many researchers especially in recent years. This theory was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices [19, 21]. The main reason to select this controller in wide range area is have acceptable control performance and solve two most important challenging topics in control which names, stability and robustness [2; 17; 62]. However, this controller used in wide range but, pure sliding mode controller has following disadvantages. Firstly, chattering problem; which can caused the high frequency oscillation of the controllers output. Secondly, sensitivity; this controller is very sensitive to the noise when the input signals very close to the zero. Last but not the least, nonlinear equivalent dynamic formulation; which this problem is very important to have a good performance and it is difficult to calculation because it is depending on the nonlinear dynamic equation [63, 25].

Chattering phenomenon can cause some problems such as saturation and heat for mechanical parts of robot manipulators or drivers. To reduce or eliminate the chattering, various papers have been reported by many researchers and classified in two most important methods, namely, boundary layer saturation method and estimated uncertainties method [2, 19, 21-23, 27, 51, 53, 56].

In recent years, artificial intelligence theory has been used in sliding mode control systems. Neural network, fuzzy logic, and neuro-fuzzy are synergically combined with sliding mode controller and used in nonlinear, time variant, and uncertainty plant (e.g., robot manipulator). The strategies for robotics are classified in two main groups: classical and non-classical methods, where the classical methods use the mathematical models to control systems and non-classical methods use the artificial intelligence theory such as fuzzy logic, neural networks and/or neuro-fuzzy.

After the invention of fuzzy logic theory in 1965 by Zadeh, this theory was used in wide range applications. Fuzzy logic controller (FLC) is one of the most important applications in fuzzy logic theory. This controller can be used to control of nonlinear, uncertain systems and transfer expert knowledge to mathematical formulation. However pure FLC works in many areas but, it cannot guarantee the basic requirement of stability and acceptable performance [53]. Some researchers applied fuzzy logic methodology in sliding mode controllers (FSMC) to reduce the chattering and equivalent problems in pure sliding mode controller so called fuzzy sliding mode controller [41, 46, 61] and the other researchers applied sliding mode methodology in fuzzy logic controller (SMFC) to improve the stability of system that is most important challenge in pure FLC [4, 23, 48-50].

Fuzzy sliding mode controller (FSMC) is a sliding mode controller which combined to fuzzy logic system (FLS) to reduce or eliminate the high frequency oscillation (chattering), to compensate the unknown system dynamics, and also to adjustment of the linear sliding surface slope. H.Temeltas [46] has proposed FSMC to achieving robust tracking of nonlinear systems. C. L. Hwang et al. [8] have proposed a fuzzy model based sliding mode control based on N fuzzy based linear state-space. A multi-input multi-output FSMC to reduce the chattering and constructed to approximate the unknown system has been presented for a robot manipulator [42].

Sliding mode fuzzy controller (SMFC) is a fuzzy logic controller based on sliding mode methodology to reduce the fuzzy rules and to refine the stability of close loop system. Research on SMFC is significantly growing as their applications, for instance, in control robot manipulator which, have been reported in [4, 23, 48-50]. H.K.Lee et al. [52] have presented self tuning SMFC to reduce the fuzzy rules, increase the stability and to automatically adjusted control parameters. Palm R [23] has proposed SMFC to increase the robustness and trajectory disturbance.

Another method to intelligent control of robot manipulator is Artificial Neural Networks (ANNS) or Neural networks (NNs). It is a parametric nonlinear function and the parameters are the weights of the NNs. It can be used in two areas in robotics, namely, control robot manipulator and identification. Neural networks control is very effective tool to control robot manipulator when robot manipulators have uncertainty in dynamic part. With this method, researcher can design approximate for an unknown dynamical system only by knowing the input-output data of systems (i.e., training data) [24-25].

Adaptive control used in systems whose dynamic parameters are varying and need to be training on line. In general states adaptive control classified in two main groups: traditional adaptive method and fuzzy adaptive method, that traditional adaptive method need to have some information about dynamic plant and some dynamic parameters must be known but fuzzy adaptive method can training the variation of parameters by expert knowledge. Combined adaptive method to artificial sliding mode controllers can help to controllers to have better performance by online tuning the nonlinear and time variant parameters. 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 use of saturation function.

This paper is organized as follows. In section 2, main subject of sliding mode controller and formulation are presented. This section covered the following details, classical sliding mode control, classical sliding for robotic manipulators, equivalent control and chatter free sliding control. A review of the fuzzy logic system and application to sliding mode controller is presented and introduces the description and analysis of adaptive artificial sliding mode controller is presented in section 3. In section 4, the conclusion is presented.

2 SLIDING MODE CONTROL (VARIABLE STRUCTURE CONTROL) AND THE ROBOT MANIPULATOR APPLICATIONS

Sliding mode controller (SMC) is a powerful nonlinear controller which has been analyzed by many researchers especially in recent years. This theory was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices [2]. The main reason to opt for this controller is its acceptable control performance in wide range and solves two most important challenging topics in control which names, stability and robustness [7, 17-20]. Sliding mode control theory for control of robot manipulator was first proposed in 1978 by Young to solve the set point problem ($\dot{q}_d = 0$) by discontinuous method in the following form [19, 3];

$$\tau_{(q,t)} = \begin{cases} \tau_i^+(q,t) & if\ S_i > 0 \\ \tau_i^-(q,t) & if\ S_i < 0 \end{cases} \tag{1}$$

where S_i is sliding surface (switching surface), $i = 1, 2, \dots \dots, n$ for *n*-DOF robot manipulator, $\tau_i(q,t)$ is the i^{th} torque of joint. Sliding mode controller is divided into two main sub controllers: discontinues controller(τ_{dis}) and equivalent controller(τ_{eq}).

Discontinues controller causes an acceptable tracking performance at the expense of very fast switching. In the theory of infinity fast switching can provide a good tracking performance but it also can provide some problems (e.g., system instability and chattering phenomenon). After going toward the sliding surface by discontinues term, equivalent term help to the system dynamics match to the sliding surface[1, 6]. However, this controller used in many applications but, pure sliding mode controller has following challenges: chattering phenomenon, and nonlinear equivalent dynamic formulation [20].

Chattering phenomenon can causes some problems such as saturation and heat the mechanical parts of robot manipulators or drivers that has shown in Figure 2.6. To reduce or eliminate the chattering, various papers have been reported by many researchers which classified into two most important methods: boundary layer saturation method and estimated uncertainties method [1].

In boundary layer saturation method, the basic idea is the discontinuous method replacement by saturation (linear) method with small neighborhood of the switching surface. This replacement caused to increase the error performance against with the considerable chattering reduction. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode with boundary layer to improve the industry application [22]. R. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. As mentioned [24]sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most exceptional stability and robustness. Conversely above method has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. To eliminate the above problems control researchers have applied artificial intelligence method (e.g., fuzzy logic) in nonlinear robust controller (e.g., sliding mode controller) besides this technique is very useful in order to implement easily.

Estimated uncertainty method used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [25-28]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Their design had three main parts i.e. equivalent, discontinuous and fuzzy estimator tuning part which has reduced the chattering very well. Elmali et al. [27]and Li and Xu [29]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering. This method was tested for the tracking control of the first two links of a SCARA type HITACHI robot. In this technique, digital controller is used to increase the system's response quality. Conversely this method has the following advantages; increasing the controller's response speed and reducing dependence on dynamic system model by on-line control, the main disadvantage are chattering phenomenon and need to improve the performance.

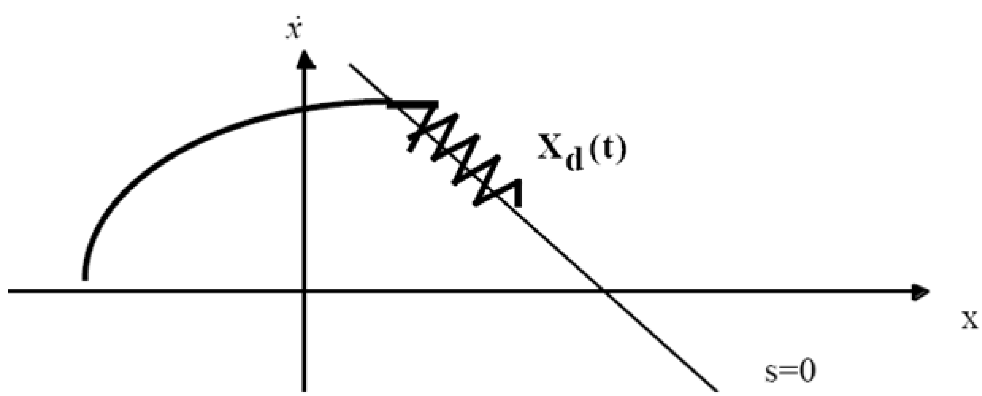


FIGURE 1: Chattering as a result of imperfect control switching [1]

Robot manipulators are one of the highly nonlinear and uncertain systems which caused to needed to robust controller. This section provides introducing the formulation of sliding mode controller to robot manipulator based on [1, 6]Consider a nonlinear single input dynamic system of the form [6]:

$$x^{(n)} = f(\vec{x}) + b(\vec{x})u \quad \textbf{(2)}$$

Where u is the vector of control input, $x^{(n)}$ is the n^{th} derivation of x, $x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T$ is the state vector, $f(x)$ is unknown or uncertainty, and $b(x)$ is of known *sign* function. The control problem is truck to the desired state; $x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d^{(n-1)}]^T$, and have an acceptable error which is given by:

$$\tilde{x} = x - x_d = [\tilde{x}, \dots, \tilde{x}^{(n-1)}]^T \quad (3)$$

A time-varying sliding surface $s(x,t)$ is given by the following equation:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\tilde{x} = 0 \quad (4)$$

where λ is the positive constant. To further penalize tracking error integral part can be used in sliding surface part as follows:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\left(\int_0^t \tilde{x}\,dt\right) = 0 \quad (5)$$

The main target in this methodology is kept the sliding surface slope $s(x,t)$ near to the zero. Therefore, one of the common strategies is to find input U outside of $s(x,t)$.

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)| \quad (6)$$

where ζ is positive constant and in equation (6) forces tracking trajectories is towards sliding condition as shown in Figure 2.

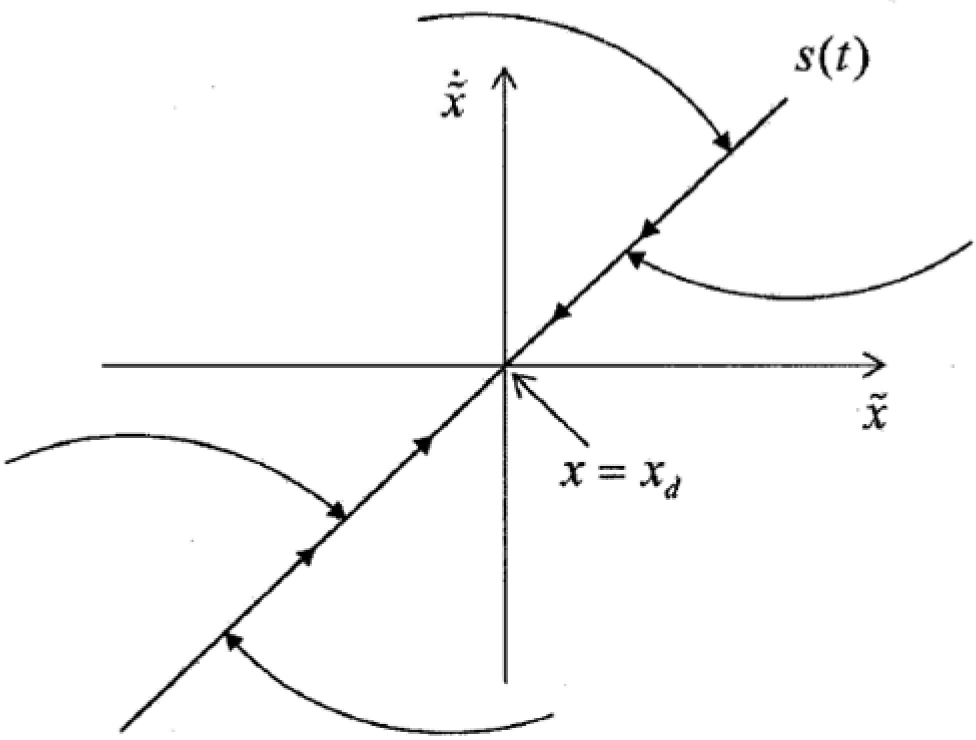


FIGURE 2: Sliding surface [2]

$$\text{If } S(0)>0 \rightarrow \frac{d}{dt}S(t) \le -\zeta \quad (7)$$

To eliminate the derivative term, it is used an integral term from t=0 to t=t_{reach}

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt}S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \quad (8)$$

Where t_{reach} is the time that trajectories reach to the sliding surface so, suppose $S(t_{reach}=0)$ defined as

$$0-S(0)\le -\eta(t_{reach}) \to t_{reach} \le \frac{S(0)}{\zeta} \tag{9}$$

and

$$if\ S(0)<0 \to 0-S(0)\le -\eta(t_{reach}) \to S(0)\le -\zeta(t_{reach}) \to t_{reach} \le \frac{|S(0)|}{\eta} \tag{10}$$

Equation (2.41) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \to error(x-x_d)=0 \tag{11}$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt}+\lambda)\quad \tilde{x} = (\dot{x}-\dot{x}_d)+\lambda(x-x_d) \tag{12}$$

The derivation of S, namely, $\dot{S}$ can be calculated as the following;

$$\dot{S} = (\ddot{x}-\ddot{x}_d)+\lambda(\dot{x}-\dot{x}_d) \tag{13}$$

suppose the second order system is defined as;

$$\ddot{x} = f+u \ \to \dot{S} = f+U-\ddot{x}_d+\lambda(\dot{x}-\dot{x}_d) \tag{14}$$

Where f is the dynamic uncertain, and also since $S=0$ *and* $\dot{S}=0$, to have the best approximation ,$\hat{U}$ is defined as

$$\hat{U} = -\hat{f}+\ddot{x}_d-\lambda(\dot{x}-\dot{x}_d) \tag{15}$$

A simple solution to get the sliding condition when the dynamic parameters have uncertainty is the switching control law:

$$U_{dis} = \hat{U}-K(\vec{x},t)\cdot \mathrm{sgn}(s) \tag{16}$$

where the switching function $\mathrm{sgn}(S)$ is defined as

$$sgn(s) = \begin{cases} 1 & s>0 \\ -1 & s<0 \\ 0 & s=0 \end{cases} \tag{17}$$

and the $K(\vec{x},t)$ is the positive constant. Suppose by (16) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S}\cdot S = \left[f-\hat{f}-K\mathrm{sgn}(s)\right]\cdot S = \left(f-\hat{f}\right)\cdot S - K|S| \tag{18}$$

and if the equation (17) instead of (18) the sliding surface can be calculated as

$$s(x,t) = (\frac{d}{dt}+\lambda)^2 \left(\int_0^t \tilde{x}\, dt\right) = (\dot{x}-\dot{x}_d)+2\lambda(\dot{x}-\dot{x}_d)-\lambda^2(x-x_d) \tag{19}$$

in this method the approximation of U is computed as

$$\hat{U} = -\hat{f}+\ddot{x}_d-2\lambda(\dot{x}-\dot{x}_d)+\lambda^2(x-x_d) \tag{20}$$

To reduce or eliminate the chattering it is used the boundary layer method; in boundary layer method the basic idea is replace the discontinuous method by saturation (linear) method with small neighborhood of the switching surface. Figure 3 is shown the control law in boundary layer.

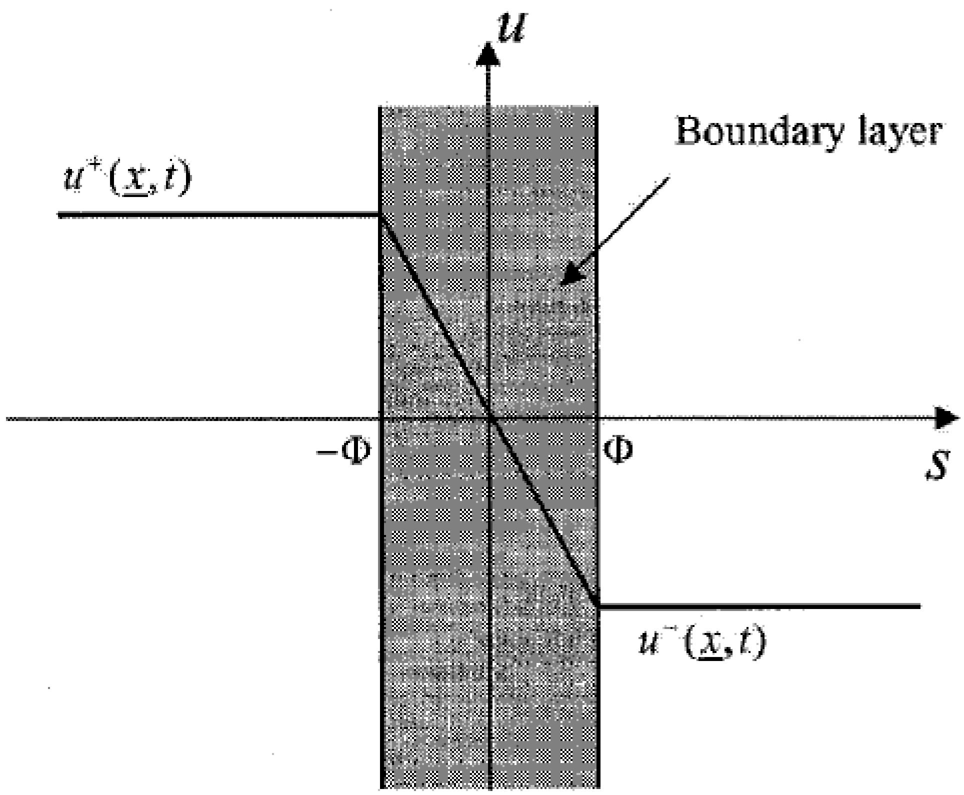


FIGURE 3: Boundary layer applied in discontinuous part [2]

$$B(t) = \{x, |S(t)| \le \emptyset\}; \emptyset > 0 \qquad \textbf{(21)}$$

Where $\emptyset$ is the boundary layer thickness. Therefore the saturation function $\mathbf{Sat}(S/_\emptyset)$ is added to the control law as

$$U = K(\vec{x},t) \cdot \mathrm{Sat}\left(S/_\emptyset\right) \qquad \textbf{(22)}$$

Where $\mathbf{Sat}\left(S/_\emptyset\right)$ can be defined as

$$\mathrm{sat}\left(S/_\emptyset\right) = \begin{cases} 1 & (S/_\emptyset > 1) \\ -1 & (s/_\emptyset < 1) \\ s/_\emptyset & (-1 < s/_\emptyset < 1) \end{cases} \qquad \textbf{(23)}$$

Based on above discussion, the control law for a multi degrees of freedom robot manipulator is written as:

$$\tau = \tau_{eq} + \tau_{sat} \qquad \textbf{(24)}$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems and τ_{eq} can be calculate as follows:

$$\tau_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \qquad \textbf{(25)}$$

Where τ is $n\times 1$ vector of actuation torque, M (q) is $n\times n$ symmetric and positive define inertia matrix, $B(q)$ 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: $[\dot{q}_1.\dot{q}_2,\dot{q}_1.\dot{q}_3,\dots,\dot{q}_1.\dot{q}_n,\dot{q}_2.\dot{q}_3,\dots]^T$ and $[\dot{q}]^2$ is vector, that it can given by: $[\dot{q}_1^{\,2},\dot{q}_2^{\,2},\dot{q}_3^{\,2},\dots]^T$and $G(q)$ is Gravity terms,

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]

$$M(q)=\begin{bmatrix} M_{11} & M_{12} & \dots & \dots & \dots & M_{1n} \\ M_{21} & \dots & \dots & \dots & \dots & M_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ M_{n.1} & \dots & \dots & \dots & \dots & M_{n.n} \end{bmatrix} \quad \textbf{(26)}$$

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

$$B(q)=\begin{bmatrix} b_{112} & b_{113} & \dots & b_{11n} & b_{123} & \dots & b_{12n} & \dots & \dots & b_{1,n-1,n} \\ b_{212} & \dots & \dots & b_{21n} & b_{223} & \dots & \dots & \dots & \dots & b_{2,n-1,n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{n,1,2} & \dots & \dots & b_{n,1,n} & \dots & \dots & \dots & \dots & \dots & b_{n,n-1,n} \end{bmatrix} \quad \textbf{(27)}$$

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

$$C(q)=\begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \quad \textbf{(28)}$$

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

$$G(q)=\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \quad \textbf{(29)}$$

and τ_{sat} is computed as;

$$\tau_{sat}=K\cdot \mathrm{sat}\left(S/_{\emptyset}\right) \quad \textbf{(30)}$$

the control output can be written as;

$$\tau=\tau_{eq}+K.\mathrm{sat}\left(S/_{\emptyset}\right)=\begin{cases}\tau_{eq}+K.\mathrm{sgn}(S) & ,|S|\geq\emptyset \\ \tau_{eq}+K.S/_{\emptyset} & ,|S|<\emptyset\end{cases} \quad \textbf{(31)}$$

Figure 4 shows the position classical sliding mode control for robot manipulator. By (30) and (31) the sliding mode control of robot manipulator is calculated as;

$$\tau=\left[M^{-1}(B+C+G)+\dot{S}\right]M+K\cdot \mathrm{sat}\left(S/_{\emptyset}\right) \quad \textbf{(32)}$$

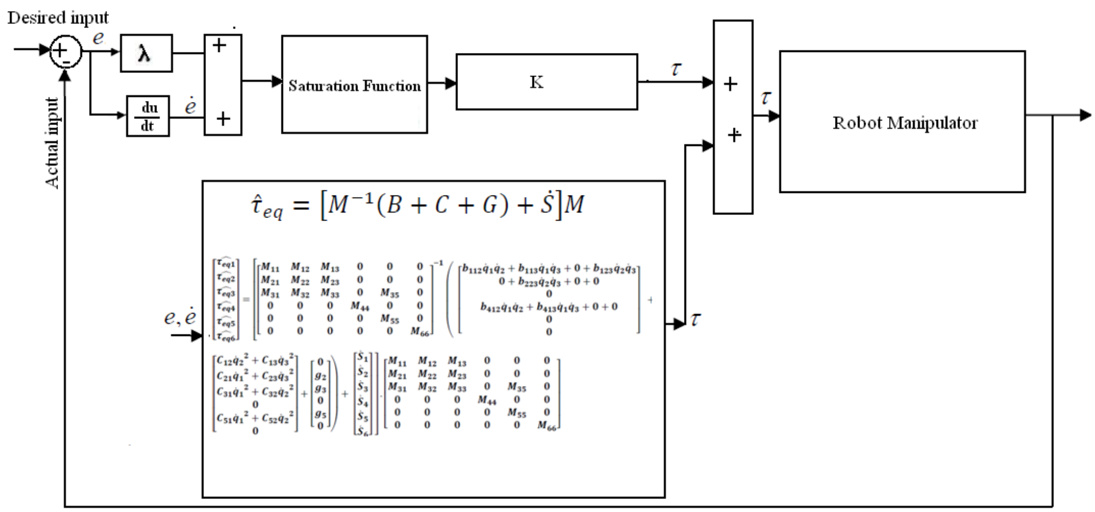


FIGURE 4: diagram of classical sliding mode controller [3, 10-14, 64]

3 INTRODUCTION TO FUZZY CONTROL AND ITS APPLICATION TO SMC

In recent years, artificial intelligence theory has been used in sliding mode control systems. 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. The complete fuzzy rule base for conventional fuzzy controller is shown in Table 1.

Control robot arm manipulators using classical controllers are based on manipulator dynamic model. These controllers often have many problems for modelling. 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].

TABLE 1: Rule table for 2 DOF robot manipulator [40]

$\dot{e}/e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

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].

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) [5, 15-18].

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

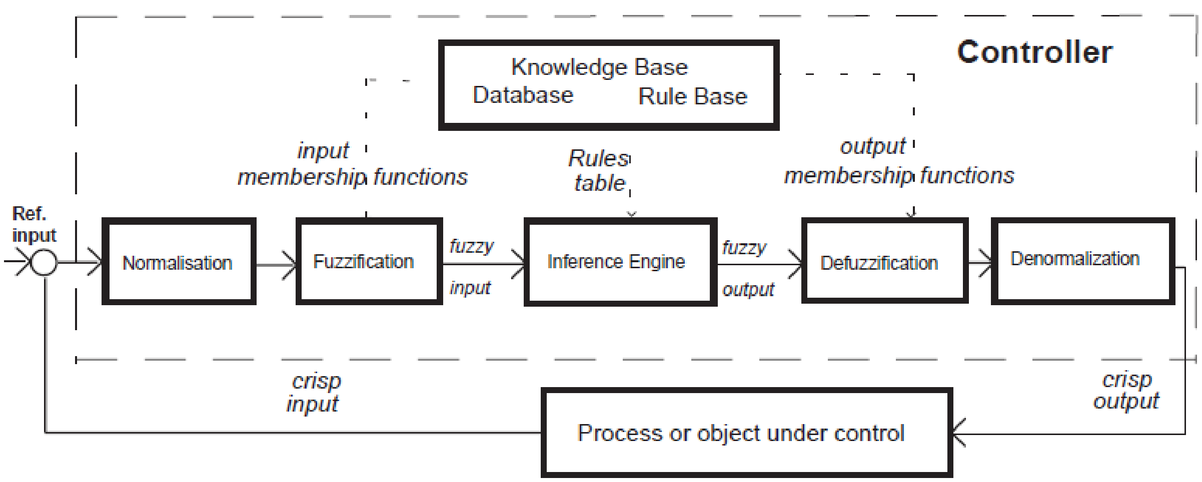


FIGURE 5: Block diagram of a fuzzy controller with details.

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 [24, 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 (Figure 6). 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].

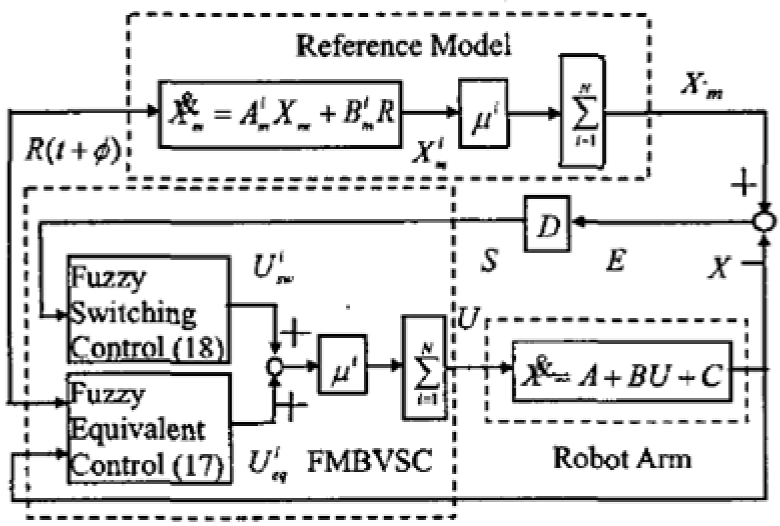


FIGURE 6: Fuzzy model base sliding mode controller [47]

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. Figure 7 is shown the sliding mode fuzzy rule table and the block diagram of SMFC.

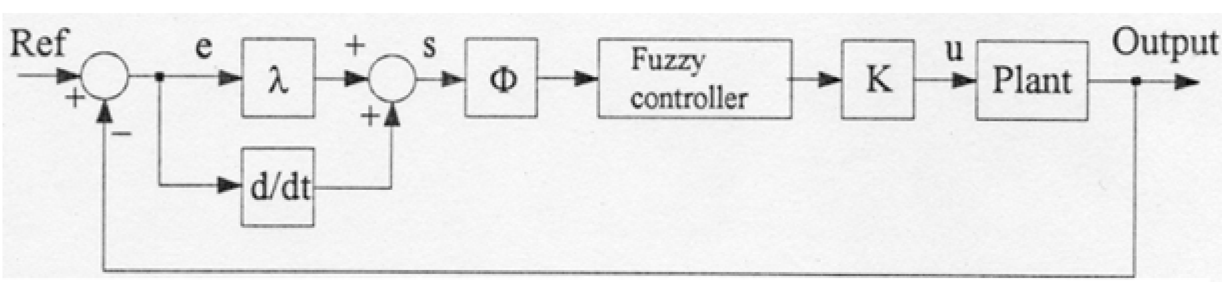


FIGURE 7: The structure of sliding mode fuzzy controller [52]

However the application of FSMC and SMFC are growing but the main SMFC drawback compared to FSMC is calculation the value of sliding surface λ pri-defined very carefully. Moreover, the advantages of SMFC compared to FLC reduce the number of fuzzy rule base and increase the robustness and stability. At last FSMC compare to the SMFC is more suitable for implementation action.

In various dynamic parameters systems that need to be training on-line adaptive control methodology is used. 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. 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 $n-DOF$ robot manipulator with k membership function for each input variable, the number of fuzzy rules for each joint is equal to $3k^{2n}$ that causes to high computation load and also this controller has chattering. This method can only tune the consequence part of the fuzzy rules (Figure 8).

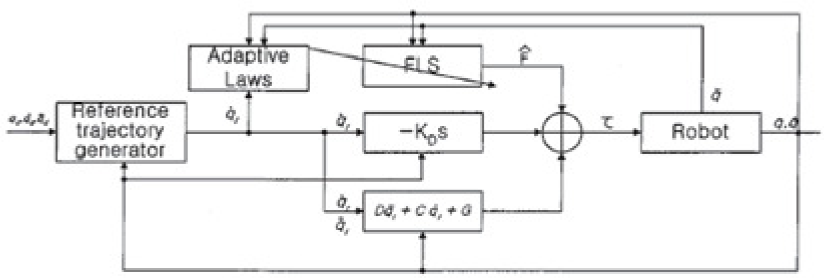


FIGURE 8: The structure of adaptive fuzzy compensate control of robot [58]

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 K_2 membership functions, the number of fuzzy rules for each joint is $(m+1)K_2^m + K_2$. 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 (Figure 9). 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 is illustrated a comparison between computed torque controller[9, 15-16], sliding mode controller[1, 3, 6, 10-14, 17-23, 26], fuzzy logic controller (FLC)[31-40], applied sliding mode in fuzzy logic controller (SMFC)[23, 14, 48-50], applied fuzzy logic method in sliding mode controller (FSMC)[10-14, 54-55, 60-61, 64] and adaptive fuzzy sliding mode controller [3, 10-14, 64].

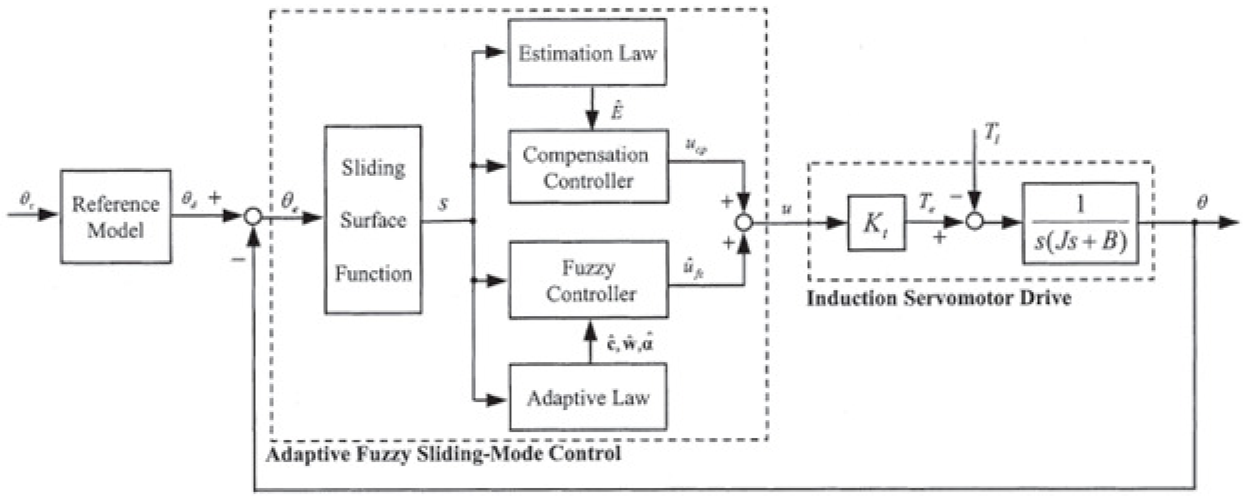


FIGURE 9: The structure of adaptive fuzzy sliding mode controller [61]

4 CONCLUSIONS

Refer to the research, review of sliding mode control and application to robot manipulator has explained in order to design high performance nonlinear controller in the presence of uncertainties. Sliding mode controller (SMC) is a significant nonlinear controller in certain and uncertain dynamic parameters systems. This controller is used to present a systematic solution for stability and robustness which they can play important role to select the best controller. Conversely, pure sliding mode controller is used in many applications; it has two important drawbacks namely; chattering phenomenon, and nonlinear equivalent dynamic formulation in uncertain dynamic parameter. One of the most important techniques to reduce or remove above two challenges is applying non-classical (artificial intelligence) method in robust classical such as sliding mode controller method. In order to solve the uncertain dynamic parameters and complex parameters systems with an artificial intelligence theory, fuzzy logic is one of the best choice which it is used in this research. However fuzzy logic method is useful to control complicated nonlinear dynamic mathematical models but the response quality may not always be so high. This controller can be used in main part of controller (e.g., pure fuzzy logic controller), it can be used to design adaptive controller (e.g., adaptive fuzzy controller), tuning parameters and finally applied to the classical controllers. As mentioned above to improve stability and reduce the fuzzy rule base in fuzzy logic controller one of the major methods is applied sliding mode controller in fuzzy logic methodology. Sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most excellent stability and robustness. Conversely SMFC has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. As mentioned above sliding mode controller has some limitations which applied fuzzy logic in sliding mode controller can causes to reduce the limitations. To compensate the nonlinearity of dynamic equivalent control, several researchers are used model base fuzzy controller instead classical equivalent controller. However FSMC has an acceptable performance but calculate the sliding surface slope by experience knowledge is difficult, particularly when system has structure or unstructured uncertainties, self tuning sliding surface slope fuzzy sliding mode controller is recommended.

TABLE 2: Comparison of six important algorithms [3, 10-14, 64]

Type of method	Advantages	Disadvantages	What to do?
1. CTC	• Higher tracking accuracy • Lower energy • Easy to implement	• That is required accurate dynamic formulation • It has problem under condition of : uncertain system and external disturbance	Design computed torque like controller or the other robust controller such as SMC
2. SMC	• Good control performance for nonlinear systems • In MIMO systems • In discrete time circuit	• Equivalent dynamic formulation • Chattering • It has limitation under condition of : uncertain system and external disturbance	Applied artificial intelligent method in SMC (e.g., FSMC or SMFC)
3. FLC	• Used in unclear and uncertain systems • Flexible • Easy to understand • Shortened in design	• Quality of design • Should be to defined fuzzy coefficient very carefully • Cannot guarantee the stability • reliability	Applied adaptive method in FLC, tuning parameters and applied to classical linear or nonlinear controller
4. SMFC	• Reduce the rule base • Reduce the chattering • Increase the stability and robustness	• Equivalent part • Defined sliding surface slope coefficient very carefully • Difficult to implement • Limitation in noisy and uncertain system	Applied adaptive method, self learning and self organizing method in SMFC
5. FSMC	• More robust • Reduce the chattering • Estimate the equivalent • Easy to implement	• Model base estimate the equivalent part • Limitation in noisy and uncertain system	Design fuzzy error base like equivalent controller and applied adaptive method
6. Adaptive FSMC	• More robust • eliminate the chattering • Estimate the equivalent	• Model base estimate the equivalent part	

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] 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.

[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] 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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[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] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

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

Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm

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

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

Abbas Zare 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

Sadeq Allahdadi 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

Mohammadali Dialame 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

The developed control methodology can be used to build more efficient intelligent and precision mechatronic systems. Three degrees of freedom robot arm is controlled by adaptive sliding mode fuzzy algorithm fuzzy sliding mode controller (SMFAFSMC). This plant has 3 revolute joints allowing the corresponding links to move horizontally. Control of robotic manipulator is very important in field of robotic, because robotic manipulators are Multi-Input Multi-Output (MIMO), nonlinear and most of dynamic parameters are uncertainty. Design strong mathematical tools used in new control methodologies to design adaptive nonlinear robust controller with acceptable performance in this controller is the main challenge. Sliding mode methodology is a nonlinear robust controller which can be used in uncertainty nonlinear systems, but pure sliding mode controller has chattering phenomenon and nonlinear equivalent part in uncertain system therefore the first step is focused on eliminate the chattering and in second step controller is improved with regard to uncertainties. Sliding function is one of the most important challenging in artificial sliding mode algorithm which this problem in order to solved by on-line tuning method. This paper focuses on adjusting the sliding surface slope in fuzzy sliding mode controller by sliding mode fuzzy algorithm.

Keywords: Mechatronic System, Robot arm, Adaptive Method, Sliding mode Fuzzy Algorithm, Fuzzy Sliding Mode Algorithm, Robust Controller, Fuzzy Logic, Sliding Mode Controller, Sliding Surface Slope, Sliding Function, Adaptive Sliding Mode Fuzzy algorithm Fuzzy Sliding Mode Controller.

1. INTRODUCTION

A robot system without any controllers does not to have any benefits, because controller is the main part in this sophisticated system. The main objectives to control of robot manipulators are stability and robustness. Lots of researchers work on design the controller for robotic manipulators to have the best performance. Control of any systems divided in two main groups: linear and nonlinear controller [1].

However, one of the important challenging in control algorithms is design linear behavior controller to easier implementation for nonlinear systems but these algorithms have some limitation such as controller working area must to be near the system operating point and this adjustment is very difficult specially when the dynamic system parameters have large variations and when the system has hard nonlinearities [1-4]. Most of robot manipulators which work in industry are usually controlled by linear PID controllers. But the robot manipulator dynamic functions are, nonlinear with strong coupling between joints (low gear ratio), structure and unstructured uncertainty and Multi- Inputs Multi-Outputs (MIMO) which, design linear controller is very difficult especially if the velocity and acceleration of robot manipulator be high and also when the ratio between joints gear be small [2]. To eliminate above problems in physical systems most of control researcher go toward to select nonlinear robust controller.

One of the most important powerful nonlinear robust controllers is Sliding Mode Controller (SMC). However Sliding mode control methodology was first proposed in the 1950 but this controller has been analyzed by many researchers in recent years. The main reason to select this controller in wide range area is have an acceptable control performance and solve some main challenging topics in control such as resistivity to the external disturbance and uncertainty. However, this controller has above advantages but, pure sliding mode controller has following disadvantages i.e.chattering problem, sensitive and equivalent dynamic formulation [3, 5-15].

After the invention of fuzzy logic theory in 1965 by Zadeh [16], this theory was used in wide range area because Fuzzy Logic Controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control of nonlinear, uncertain and noisy systems. However pure FLC works in many engineering applications but, it cannot guarantee two most important challenges in control, namely, stability and acceptable performance [16-25]. Some researchers applied fuzzy logic methodology in sliding mode controllers (FSMC) to reduce the chattering and solve the nonlinear dynamic equivalent problems in pure sliding mode controller and the other researchers applied sliding mode methodology in fuzzy logic controller (SMFC) to improve the stability of systems, therefore FSMC is a controller based on SMC but SMFC works based on FLC [26-40].

Adaptive control used in systems whose dynamic parameters are varying and/or have unstructured disturbance and need to be training on line. Adaptive fuzzy inference system provide a good knowledge tools for adjust a complex uncertain nonlinear system with changing dynamics to have an acceptable performance. Combined adaptive method to artificial sliding mode controllers can help to controllers to have a better performance by online tuning the nonlinear and time variant parameters [26-40].

This paper is organized as follows: In section 2, dynamic formulation of robot manipulator is presented. Detail of mamdani fuzzy inference estimator is introduce and applied to sliding mode methodology is presented in section 3. In section 4, design adaptive methodology and applied to propose method is presented. In section 5, the simulation result is presented and finally in section 6, the conclusion is presented.

2. APPLICATION: ROBOT MANIPULATOR DYNAMIC FORMULATION

The equation of an *n-DOF* robot manipulator governed by the following equation [1, 4, 33-40]:

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

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 [33-40]:

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

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 variable q_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3, 33-40]:

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

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

3. MAMDANI FUZZY INFERENCE ESTIMATOR SLIDING MODE METHODOLOGY

Sliding mode controller (SMC) is a influential nonlinear, stable and robust controller which it was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices[1, 5-11, 33-40]. A time-varying sliding surface $s(x,t)$ is given by the following equation:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\tilde{x} = 0 \quad (4)$$

where λ is the constant and it is positive. The derivation of S, namely, $\dot{S}$ can be calculated as the following formulation [5-11, 33-40]:

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \quad (5)$$

The control law for a multi degrees of freedom robot manipulator is written as:

$$U = U_{eq} + U_r \quad (6)$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and it can be calculate as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \quad (7)$$

Where $M(q)$ is an inertia matrix which it is symmetric and positive, $V(q,\dot{q}) = B + C$ is the vector of nonlinearity term and $G(q)$ is the vector of gravity force and U_r with minimum chattering based on [33-40] is computed as;

$$U_r = K \cdot (\mathrm{mu} + \mathrm{b})\left({S}/{\emptyset}\right) \quad (8)$$

Where $\emptyset_u = mu + b = saturation_{fuction}$ is a dead zone (saturation) function and, u and b are unlimited coefficient, by replace the formulation (5) in (3) the control output can be written as;

$$U = U_{eq} + K.(\mathrm{mu} + \mathrm{b})\left({S}/{\emptyset}\right) = \begin{cases} U_{eq} + K.\mathrm{sgn}(S) & ,|S| \geq \emptyset \\ U_{eq} + K.{S}/{\emptyset} & ,|S| < \emptyset \end{cases} \quad (9)$$

Where the function of $sgn(S)$ defined as;

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \quad (10)$$

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) [16-25, 33-40].

The basic structure of a fuzzy controller is shown in Figure 1.

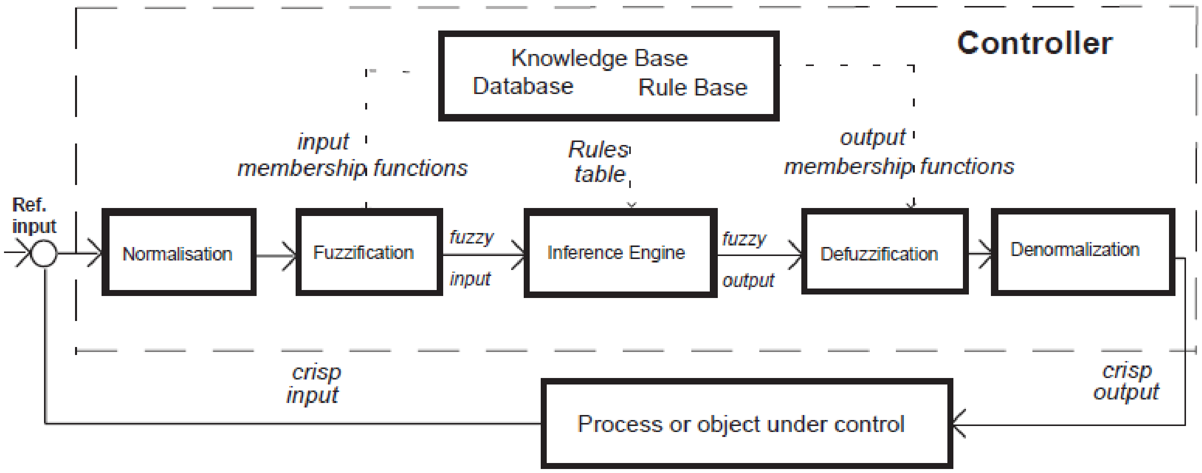


FIGURE 1: Block diagram of a fuzzy controller with details.

The fuzzy system can be defined as below [40]

$$f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(S) \tag{11}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{12}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (8) and $\mu_{(xi)}$ is membership function.

error base fuzzy controller can be defined as

$$U_{fuzzy} = \psi(S) \tag{13}$$

In this work the fuzzy controller has one input which names; sliding function. Fuzzy controller with one input is difficult to implementation, because it needs large number of rules, to cover equivalent part estimation [16-25]. In this fuzzy inference system researcher is defined 7 linguistic variables. As a summary the design of fuzzy inference system estimator 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 deffuzzification [16-18].

Fuzzification: the first step in fuzzification is determine inputs and outputs which, it has one input; namely sliding function (S) and one output ($\emptyset$). The sliding function ($S = \lambda e + \dot{e}$) which related to the difference between desired and actual output position and the difference between desired and actual velocity. The second step is chosen an appropriate membership function for inputs and output which, for simplicity in implementation and also to have an acceptable performance the researcher is selected the triangular membership function which it is shown in Figure 2. The third step is chosen the correct labels for each fuzzy set which, in this research namely as linguistic variable. The linguistic variables for input are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), 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).

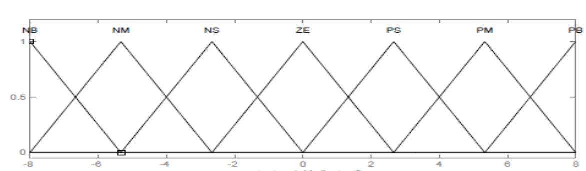


FIGURE 2: Membership function: triangular

Fuzzy Rule Base and Rule Evaluation: the first step in rule base and evaluation is 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 suppose that two fuzzy rules in this controller are;

F.R^1: ***IF S is NB, THEN ∅ is LL*** **(14)**
F.R^2: ***IF e is PS THEN ∅ is ML***

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 inference system. This part is used AND/OR fuzzy operation in antecedent part which AND operation is used.

TABLE 1: Rule table (Mamdani'sFIS)

S	NB	NM	NS	ZE	PS	PM	PB
∅	LL	ML	SL	Z	SR	MR	LR

Aggregation of the Rule Output (Fuzzy inference): Max-Min aggregation is used to this work which the calculation is defined as follows;

$$\mu_U(x_k, y_k, U) = \mu_{\cup_{i=1}^{r} FR^i}(x_k, y_k, U) = max\left\{min_{i=1}^{r}\left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U)\right]\right\} \quad (15)$$

Deffuzzification: The last step to design fuzzy inference in our 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. In this design the Center of gravity method (COG) is used and calculated by the following equation [25];

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

The fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min\,[Sup_{x\in U}|\sum_{l=1}^{M}\theta^T\zeta(x) - U_{equ}|] \quad (17)$$

Where $\theta^*$is the minimum error, $sup_{x\in U}|\sum_{l=1}^{M}\theta^T\zeta(x) - \tau_{equ}|$ is the minimum approximation error.

suppose K_j is defined as follows

$$K_j = \frac{\sum_{l=1}^{M}\theta_j^l[\mu_A(S_j)]}{\sum_{l=1}^{M}[\mu_A(S_j)]} = \theta_j^T\zeta_j(S_j) \quad (18)$$

Where $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad (19)$$

where the $\gamma_{sj}$is the positive constant.

According to the nonlinear dynamic equivalent formulation of robot manipulator the nonlinear equivalent part is estimated by (8)

$$[M^{-1}(B + C + G) + \dot{S}]M = \sum_{l=1}^{M}\theta^T\zeta(x) - \lambda S - K \quad (20)$$

Based on (3) the formulation of proposed fuzzy sliding mode controller can be written as;

$$U = U_{eq_{fuzzy}} + U_r \quad (21)$$

Where $U_{eq_{fuzzy}} = [M^{-1}(B + C + G) + \dot{S}]M + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

Figure 3 is shown the proposed fuzzy sliding mode controller.

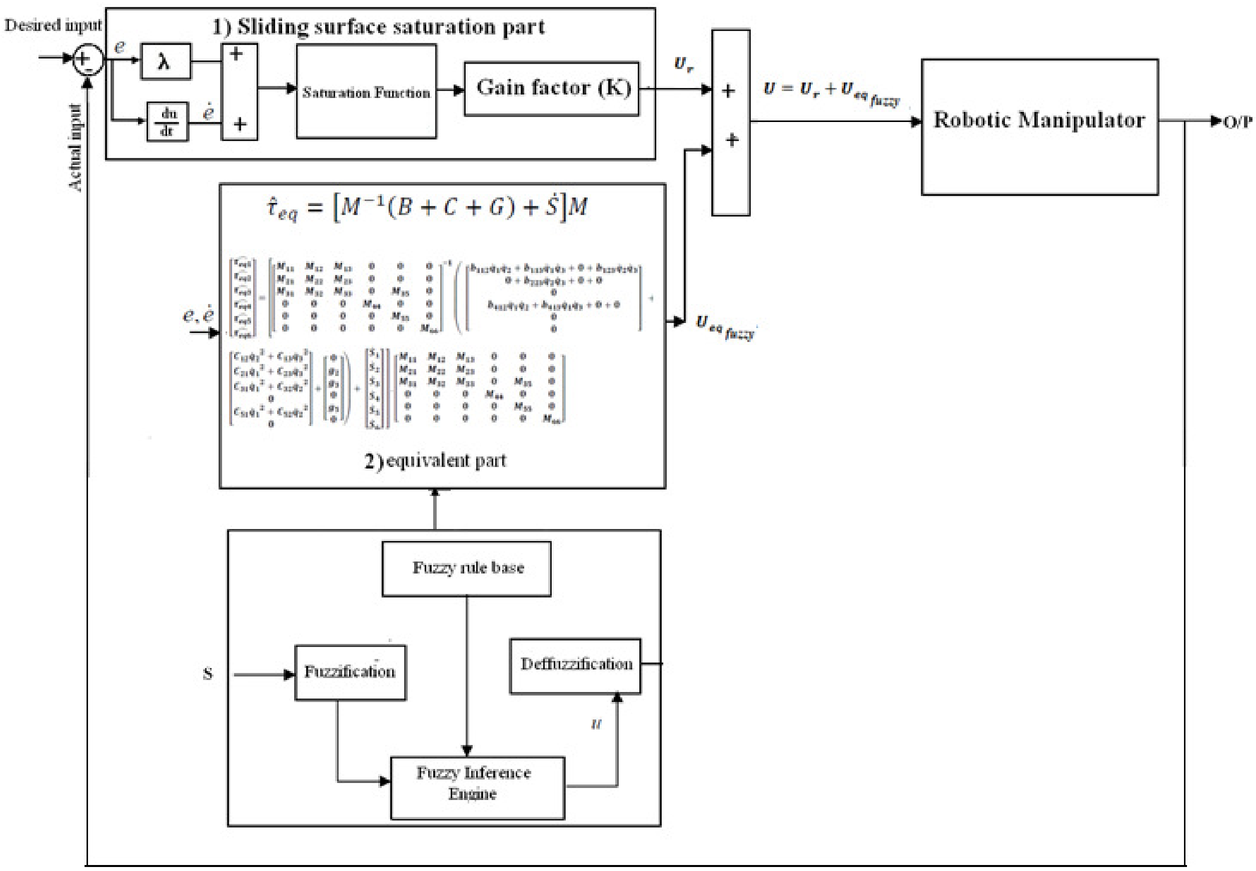


FIGURE 3: Proposed fuzzy sliding mode algorithm: applied to robot manipulator

4. DESIGN ADAPTIVE SLIDING MODE FUZZY ALGORITM: APPLIED TO PROPOSED METHODOLOGY

However proposed FSMC has satisfactory performance but calculate the sliding surface slope by try and error or experience knowledge is very difficult, particularly when system has uncertainties; sliding mode fuzzy self tuning sliding function fuzzy sliding mode controller is recommended.

$$U_{SF} = \psi\left(K \cdot (mx + b) \cdot \left({}^{S}/_{\emptyset}\right)\right) \quad (22)$$

Where U_{SF} is sliding mode fuzzy output function. The adaption low is defined as

$$\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j) \quad (23)$$

where the γ_{sj} is the positive constant and $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad (24)$$

As a result proposed method is very stable with a good performance. Figure 4 is shown the block diagram of proposed adaptive sliding mode fuzzy applied to fuzzy sliding mode controller.

The fuzzy system can be defined as below

$$f(x) = \tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(e, \dot{e}) \tag{25}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{26}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (25) and $\mu_{(xi)}$ is membership function. error base fuzzy controller can be defined as

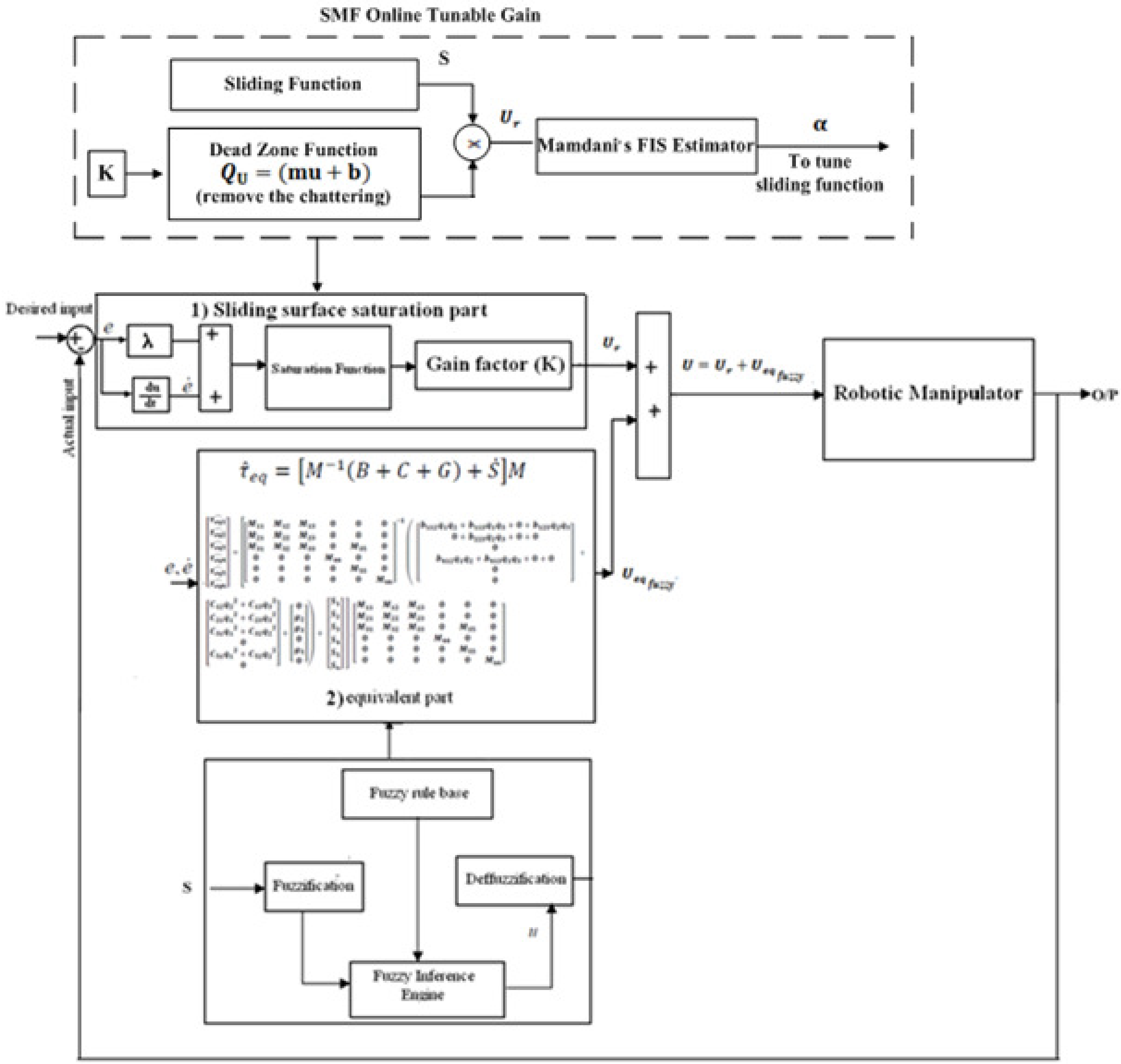


FIGURE 4: Proposed adaptive sliding fuzzy tune FSMC algorithm: applied to robot arm

$$\tau_{fuzzy} = \psi(e, \dot{e}) \tag{27}$$

According to the formulation in sliding mode algorithm

$$if\ S = 0\ then\ \dot{-e} = \lambda e \quad (28)$$

the fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min\left[Sup_{x\in U}\left|\sum_{l=1}^{M}\theta^T\zeta(x) - \tau_{equ}\right|\right] \quad (29)$$

Where $\theta^*$is the minimum error, $sup_{x\in U}\left|\sum_{l=1}^{M}\theta^T\zeta(x) - \tau_{equ}\right|$ is the minimum approximation error. The adaptive controller is used to find the minimum errors of $\theta - \theta^*$.

suppose K_j is defined as follows

$$K_j = \frac{\sum_{l=1}^{M}\theta_j^l[\mu_A(S_j)]}{\sum_{l=1}^{M}[\mu_A(S_j)]} = \theta_j^T\zeta_j(S_j) \quad (30)$$

Where $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad (31)$$

the adaption low is defined as

$$\dot{\theta}_j = \gamma_{sj}S_j\zeta_j(S_j) \quad (32)$$

where the $\gamma_{sj}$is the positive constant.

According to the formulation in fuzzy sliding and also sliding mode fuzzy

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) = \sum_{l=1}^{M}\theta^T\zeta(x) - \lambda S - K \quad (33)$$

The dynamic equation of robot manipulator can be written based on the sliding surface as;

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \quad (34)$$

It is supposed that

$$S^T(\dot{M} - 2V)S = 0 \quad (35)$$

it can be shown that

$$M\dot{S} + (V + \lambda)S = \Delta f - K \quad (36)$$

where $\Delta f = [M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q)] - \sum_{l=1}^{M}\theta^T\zeta(x)$

as a result $\dot{V}$ is became

$$\dot{V} = \frac{1}{2}S^T\dot{M}S - S^TVS + \sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j$$

$$= S^T(-\lambda S + \Delta f - K) + \sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j$$

$$= \sum_{j=1}^{m}[S_j(\Delta f_j - K_j)] - S^T\lambda S + \sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j$$

$$= \sum_{j=1}^{m}[S_j(\Delta f_j - \theta_j^T\zeta_j(S_j))] - S^T\lambda S + \sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j$$

$$= \sum_{j=1}^{m}[S_j(\Delta f_j - (\theta_j^*)^T\zeta_j(S_j) + \phi_j^T\zeta_j(S_j))] - S^T\lambda S + \sum\frac{1}{\gamma_{sj}}\phi_j^T\dot{\phi}_j$$

$$= \sum_{j=1}^{m}[S_j(\Delta f_j - ((\theta_j^*)^T\zeta_j(S_j))] - S^T\lambda S)] + \sum(\frac{1}{\gamma_{sj}}\phi_j^T[\gamma_{sj}\zeta_j(S_j)S_j + \dot{\phi}_j])$$

where $\dot{\theta}_j = \gamma_{sj}S_j\zeta_j(S_j)$ is adaption law, $\dot{\phi}_j = -\dot{\theta}_j = -\gamma_{sj}S_j\zeta_j(S_j)$, consequently $\dot{V}$ can be considered by

$$\dot{V} = \sum_{j=1}^{m} [S_j \, \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right)] - S^T \lambda S \quad \textbf{(37)}$$

the minimum error can be defined by

$$e_{mj} = \Delta f_j - \left((\theta_j^*)^T \zeta_j(S_j)\right) \quad \textbf{(38)}$$

$\dot{V}$ is intended as follows

$$\begin{aligned} \dot{V} &= \sum_{j=1}^{m} [S_j \, e_{mj}] - S^T \lambda S \\ &\leq \textstyle\sum_{j=1}^{m} |S_j \,||e_{mj}| - S^T \lambda S \\ &= \sum_{j=1}^{m} |S_j \,||e_{mj}| - \lambda_j {S_j}^2 \\ &= \sum_{j=1}^{m} |S_j \,|(|e_{mj}| - \lambda_j S_j) \end{aligned} \quad \textbf{(39)}$$

For continuous function $g(x)$, and suppose $\varepsilon > 0$ it is defined the fuzzy logic system in form of (27) such that

$$Sup_{x \in U} |f(x) - g(x)| < \epsilon \quad \textbf{(40)}$$

the minimum approximation error (e_{mj}) is very small.

$$if \;\; \lambda_j = \alpha \quad that \quad \alpha|S_j| > e_{mj} \; (S_j \neq 0) \quad then \quad \dot{V} < 0 \; for \; (S_j \neq 0) \quad \textbf{(41)}$$

5 SIMULATION RESULTS

PD sliding mode controller (PD-SMC), proposed PD fuzzy sliding mode controller (PD FSMC) and proposed adaptive sliding mode fuzzy algorithm Fuzzy Sliding Mode Controller (SMFAFSMC) were tested to Step response trajectory. This simulation applied to three degrees of freedom robot arm therefore the first, second, and third joints are moved from home to final position without and with external disturbance. The simulation was implemented in Matlab/Simulink environment. Trajectory performance, torque performance, disturbance rejection, steady state error and RMS error are compared in these controllers. It is noted that, these systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude which the sample time is equal to 0.1. This type of noise is used to external disturbance in continuous and hybrid systems.

Tracking Performances: Figure 5 is shown tracking performance for first, second and third link in SMC, FSMC and SMFAFSMC without disturbance for sinus trajectories. By comparing sinus response trajectory without disturbance in SMC, FSMC and SMFAFSMC it is found that the SMC's and FSMC's overshoot (4%) is higher than SMFAFSMC (0%), although all of them have about the same rise time.

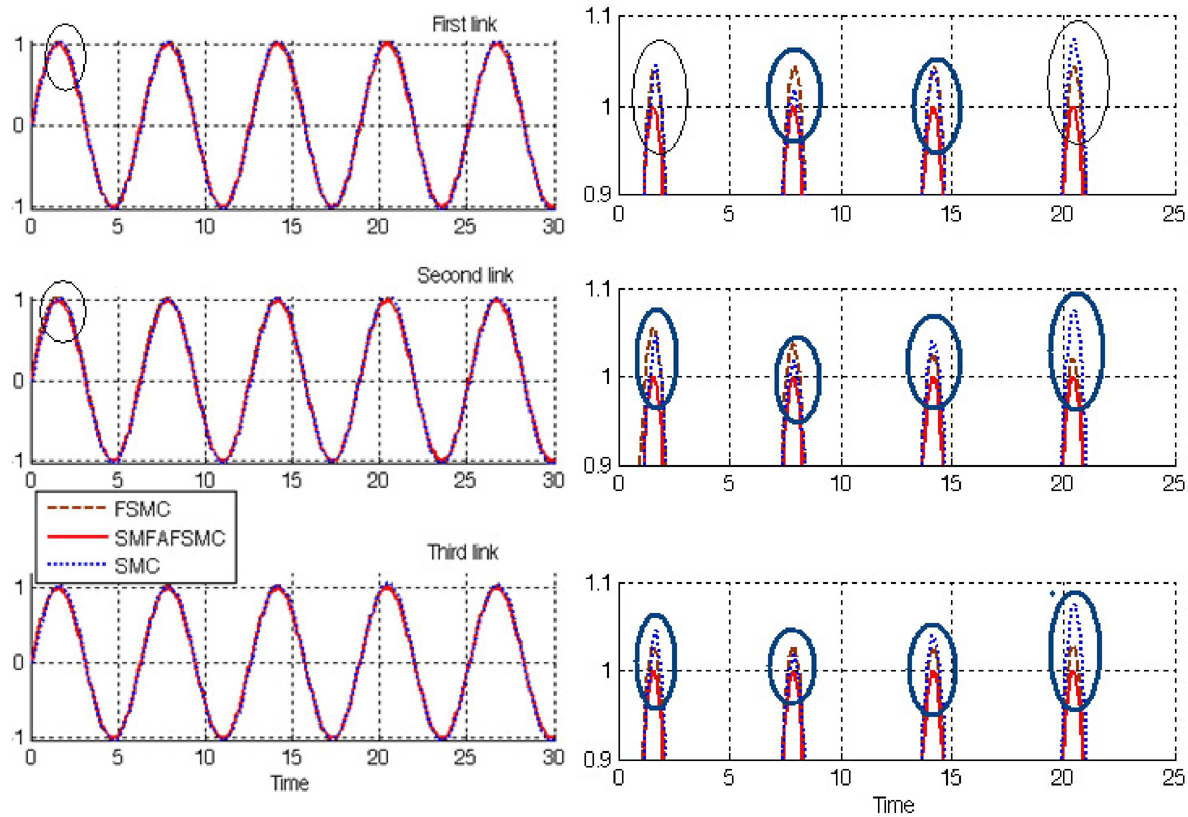


FIGURE 5: FSMC, SMFAFSMC and SMC: applied to robot manipulator.

Disturbance Rejection: Figure 6 has shown the power disturbance elimination in SMC, FSMC and SMFAFSMC. The main target in these controllers is disturbance rejection as well as reduces the chattering. A band limited white noise with predefined of 40% the power of input signal is applied to above controllers. It found fairly fluctuations in SMC and FSMC trajectory responses.

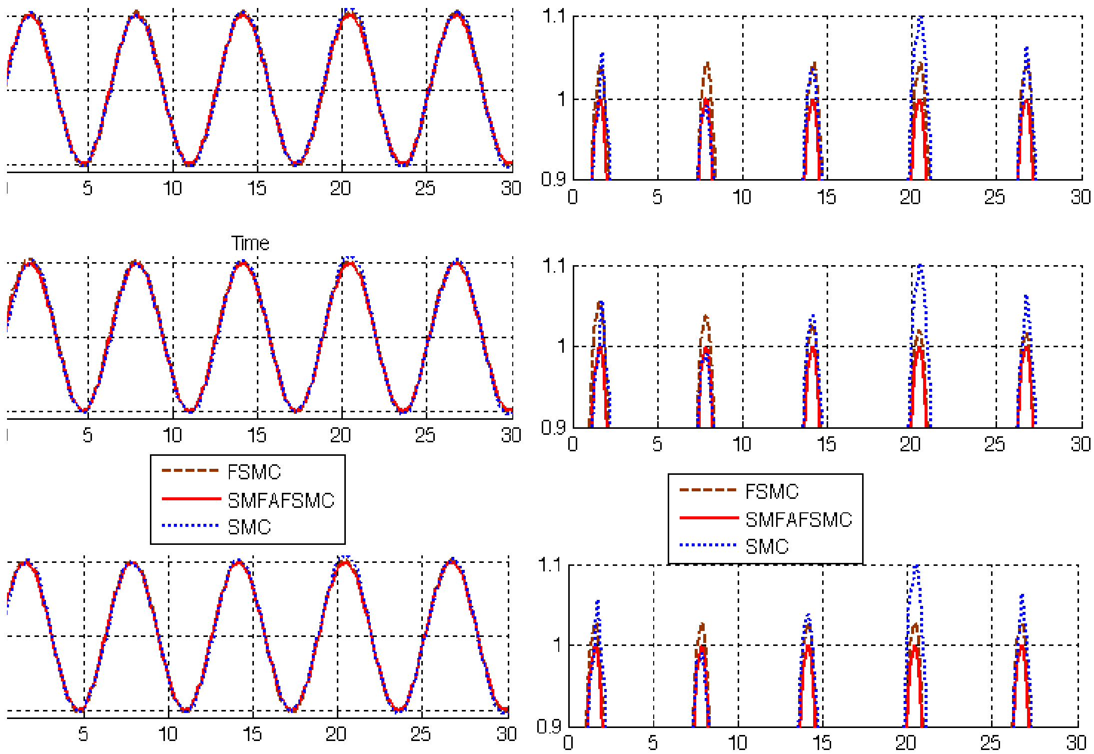


FIGURE 6: FSMC, SMFAFSMC and SMC with disturbance: applied to robot manipulator.

Among above graph relating to trajectory following with external disturbance, FLC has fairly fluctuations. By comparing some control parameters such as overshoot and rise time it found that the SMC's and FSMC's overshoot (9%) is higher than SMFAFSMC (0%).

Error Calculation: Although SMC and FSMC have the same error rate but SMFAFSMC has a better performance in presence of external disturbance (refer to Table 2 and Figure 7), they have oscillation tracking which causes chattering phenomenon. As it is obvious in Table.2 FSMC is a SMC which estimate the equivalent part so FSMC have acceptable performance with regard to SMC in presence of certain and uncertainty. Figure 7 is shown steady state and RMS error in SMC and FSMC and SMFAFSMC in presence of external disturbance.

TABLE 2: RMS Error Rate of Presented controllers

RMS Error Rate	**SMC**	**FSMC**	**SMFAFSMC**
Without Noise	**1e-3**	**0.6e-3**	**0.6e-6**
With Noise	**0.012**	**0.0012**	**0.65e-6**

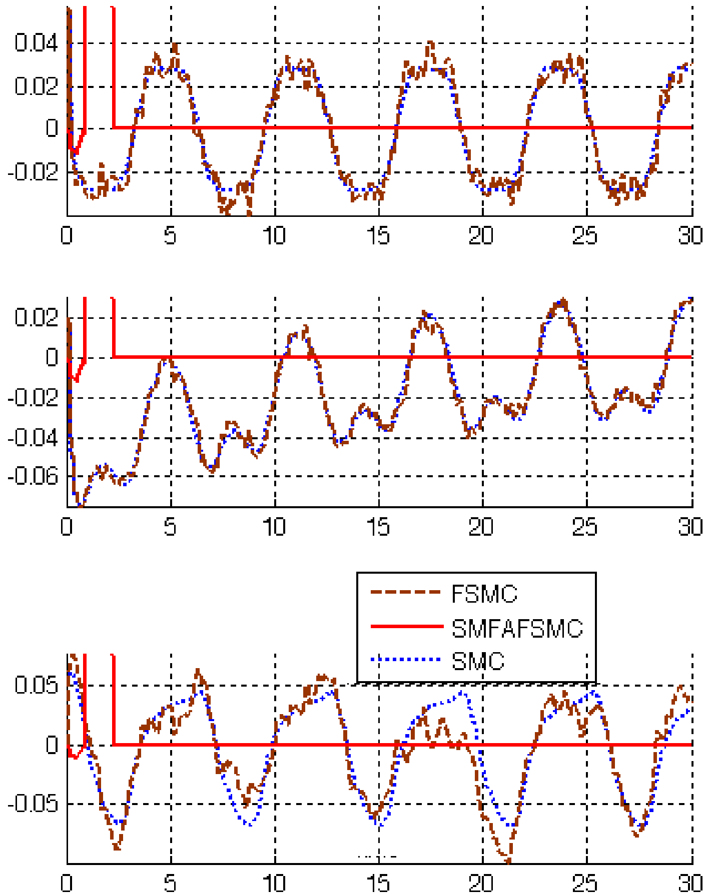


FIGURE 7: FSMC, SMFAFSMC and SMC steady state error with external disturbance: applied to robot manipulator

In these methods if integration absolute error (IAE) is defined by (15), table 3 is shown comparison between these two methods.

$$IAE = \int_0^\infty |e(t)|\, dt \tag{42}$$

Table 3: Calculate IAE

Method	Traditional SMC	FSMC	SMFAFSMC
IAE	490.1	409	210

6 CONCLUSIONS

In this research, a sliding mode fuzzy adaptive fuzzy sliding mode algorithm is proposed in order to design high performance robust controller in presence of structure unstructured uncertainties. The performance is improved by using the advantages of sliding mode algorithm, artificial intelligence method and adaptive algorithm while the disadvantages removed by added each method to previous method. Fuzzy logic method by adding to the sliding mode controller has covered negative points in fuzzy and sliding algorithms. Obviously robot manipulator dynamic parameters are nonlinear therefore design nonlinear robust model free controller is a main goal with regard to sliding mode and fuzzy logic methodology. This paper focuses on comparison between sliding mode controller, fuzzy sliding mode controller and adaptive sliding mode algorithm fuzzy sliding mode controller, to opt the best control method for the uncertain second order system (e.g., robot manipulator). Higher implementation quality of response and model free controller versus an acceptable performance in chattering, trajectory and error is reached by designing adaptive sliding mode fuzzy algorithm fuzzy sliding mode controller. This implementation considerably removes the chattering phenomenon and error in the presence of uncertainties. As a result, this controller will be able to control a wide range of nonlinear second order uncertain system (e.g., robot manipulator) with a high sampling rates because its easy to implement.

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

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

[5] 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.

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

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

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

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

[10] 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.

[11] 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.

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

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

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

[15] 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.

[16] 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.

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

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

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

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

[21] 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.

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

[23] 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.

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

[25] 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.

[26] 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.

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

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

[29] 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.

[30] 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.

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

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

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

[34] 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.

[35] 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.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

[40] 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.

Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm

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

Amin Jalali Max.jalali@gmail.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

Atefeh Gavahian a.gavahian.j@gmail.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

This research is focused on novel particle swarm optimization (PSO) SISO Lyapunov based fuzzy estimator sliding mode algorithms derived in the Lyapunov sense. The stability of the closed-loop system is proved mathematically based on the Lyapunov method. PSO SISO fuzzy compensate sliding mode method design a SISO fuzzy system to compensate for the dynamic model uncertainties of the nonlinear dynamic system and chattering also solved by nonlinear fuzzy saturation like method. Adjust the sliding function is played important role to reduce the chattering phenomenon and also design acceptable estimator applied to nonlinear classical controller so PSO method is used to off-line tuning. Classical sliding mode control is robust to control model uncertainties and external disturbances. A sliding mode method with a switching control low guarantees the stability of the certain and/or uncertain system, but the addition of the switching control low introduces chattering into the system. One way to reduce or eliminate chattering is to insert a nonlinear (fuzzy) boundary like layer method inside of a boundary layer around the sliding surface. Classical sliding mode control method has difficulty in handling unstructured model uncertainties. One can overcome this problem by applied fuzzy inference system into sliding mode algorithm to design and estimate model-free nonlinear dynamic equivalent part. To approximate a time-varying nonlinear dynamic system, a fuzzy system requires a large amount of fuzzy rule base. This large number of fuzzy rules will cause a high computation load. The addition of PSO method to a fuzzy sliding mode controller to tune the parameters of the fuzzy rules in use will ensure a moderate computational load. The PSO method in this algorithm is designed based on the PSO stability theorem. Asymptotic stability of the closed loop system is also proved in the sense of Lyapunov.

Keywords: Particle Swarm Optimization, Lyapunov Based Fuzzy Estimator Sliding Mode Algorithms, Nonlinear Fuzzy Saturation like Method, Sliding Mode Controller, Chattering Phenomenon.

1. INTRODUCTION

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. Control part is used to adjust the timing between the subparts of robot manipulator to reach the best performance (trajectory) [1].

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. 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) [1-8].

Sliding mode controller (SMC) is one of the influential nonlinear controllers in certain and uncertain systems which are used to present a methodical solution for two main important controllers' challenges, which named: stability and robustness. Conversely, this controller is used in different applications; sliding mode controller has subsequent drawbacks i.e. chattering phenomenon, and nonlinear equivalent dynamic formulation in uncertain systems[1-2]. The main reason to opt for this controller is its acceptable control performance in wide range and solves two most important challenging topics in control which names, stability and robustness [7, 9-20]. Sliding mode controller is divided into two main sub controllers: discontinues controller$(\boldsymbol{U_{dis}})$ and equivalent controller$(\boldsymbol{U_{eq}})$. Discontinues controller causes an acceptable tracking performance at the expense of very fast switching. In the theory of infinity fast switching can provide a good tracking performance but it also can provide some problems (e.g., system instability and chattering phenomenon). After going toward the sliding surface by discontinues term, equivalent term help to the system dynamics match to the sliding surface[1, 6]. However, this controller used in many applications but, pure sliding mode controller has following challenges: chattering phenomenon, and nonlinear equivalent dynamic formulation [20]. Chattering phenomenon can causes some problems such as saturation and heat the mechanical parts of robot manipulators or drivers. To reduce or eliminate the chattering, various papers have been reported by many researchers which classified into two most important methods: boundary layer saturation method and estimated uncertainties method [1, 10-14]. In boundary layer saturation method, the basic idea is the discontinuous method replacement by saturation (linear) method with small neighborhood of the switching surface. This replacement caused to increase the error performance against with the considerable chattering reduction. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode with boundary layer to improve the industry application [22]. R. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. As mentioned [24]sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most exceptional stability and robustness. Conversely above method has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. To eliminate the above problems control researchers have applied artificial intelligence method (e.g., fuzzy logic) in nonlinear robust controller (e.g., sliding mode controller) besides this technique is very useful in order to implement easily. Estimated uncertainty method used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [25-28]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Their design had three main parts i.e. equivalent, discontinuous and fuzzy estimator tuning part which has reduced the chattering very well. Elmali et al. [27]and Li and Xu [29]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering. This method was tested for the tracking control of the first two links of a SCARA type HITACHI robot. In this technique, digital controller is used to increase the system's response quality. Conversely this method has the following advantages; increasing the controller's response speed and reducing dependence on dynamic system model by on-line control, the main disadvantage are chattering phenomenon and need to improve the performance.

In recent years, artificial intelligence theory has been used in sliding mode control systems. 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 [31-40]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. 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 [24, 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]; [48-62]. 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. However the application of FSMC and SMFC are growing but the main SMFC drawback compared to FSMC is calculation the value of sliding surface λ pri-defined very carefully. Moreover, the advantages of SMFC compared to FLC reduce the number of fuzzy rule base and increase the robustness and stability. At last FSMC compare to the SMFC is more suitable for implementation action.
The tuning-gain block has been designed at each input/output stage. The PSO has been used to obtain the optimal value of the sliding surface slope. Off-line control method (e.g., PSO) is used in systems whose dynamic parameters are varying and need to be trained off line. In general states PSO algorithm can be tuned classical controller coefficient and fuzzy coefficient or membership function. PSO fuzzy inference system provide a good knowledge tools to adjust a complex uncertain nonlinear system with changing dynamics to have an acceptable performance [63-65] Combined PSO method to artificial sliding mode controllers can help the controllers to have a better performance by off-line tuning the nonlinear and time variant parameters [63-65].

In this research we will highlight the SISO PSO fuzzy sliding mode algorithm with estimates the equivalent part derived in the Lyapunov sense. This algorithm will be analyzed and evaluated on robotic manipulators. Section 2, serves as an introduction to the classical sliding mode control algorithm and its application to a two degree of-freedom robot manipulator, describe the objectives and problem statements. Part 3, introduces and describes the methodology algorithms and proves Lyapunov stability. Section 4 presents the simulation results of this algorithm applied to a 2 degree-of-freedom robot manipulator and the final section is describe the conclusion.

2. OBJECTIVES, PROBLEM STATEMENTS AND SLIDING MODE FORMULATION

When system works with various parameters and hard nonlinearities design linear controller technique is very useful in order to be implemented easily but it has some limitations such as working near the system operating point[2-20]. Sliding mode controller is used in wide range areas such as in robotics, in control process, in aerospace applications and in power converters because it has an acceptable control performance and solve some main challenging topics in control such as resistivity to the external disturbance. Even though, this controller is used in wide range areas but, pure sliding mode controller has the following disadvantages: chattering problem; which caused the high frequency oscillation in the controllers output and equivalent dynamic formulation; calculate the equivalent control formulation is difficult because it depends on the dynamic equation [20]. Conversely pure FLC works in many areas, it cannot guarantee the basic requirement of stability and acceptable performance[30-40]. Although both SMC and FLC have been applied successfully in many applications but they also have some limitations. The linear boundary layer method is used to reduce or eliminate the chattering and fuzzy estimator is used instead of dynamic equivalent equation to implement easily and avoid mathematical model base controller. To reduce the effect of uncertainty in proposed method, self tuning sliding mode fuzzy method is applied in fuzzy sliding mode controller in robot manipulator in order to solve above limitation.
The dynamic equation of an n-link robot manipulator is define as [53-62]

$$M(q)\ddot{q} + c(q,\dot{q}) + G(q) = \tau \tag{1}$$

Where $q \in R^n$ is the vector of joint position, $M(q) \in R^{n\times n}$ is the inertial matrix, $C(q,\dot{q}) \in R^n$ is the matrix of Coriolis and centrifugal forces, $G(q) \in R^n$ is the gravity vector and $\tau \in R^n$ is the vector of joint torques.
This work focuses on two-degree-of-freedom robot manipulator.
The dynamics of this robotic manipulator is given by [1, 6, 9-14]

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

Where

$$M(q) = \begin{bmatrix} m_1 l^2 + 2m_2 l^2 + 2m_2 l^2 \cos q_2 & m_2 l^2 + m_2 l^2 \cos q_2 \\ m_2 l^2 + m_2 l^2 \cos q_2 & m_2 l^2 \end{bmatrix} \tag{3}$$

$$C(q,\dot{q}) = \begin{bmatrix} -2m_2 l^2 \dot{q}_1 \dot{q}_2 \sin q_2 - m_2 l^2 \dot{q}_2^2 \sin q_2 \\ m_2 l^2 \dot{q}_1^2 \sin q_2 \end{bmatrix} \quad (4)$$

Our target is to track the desired trajectories q_d of the robotic manipulators (2) by using a sliding mode controller. We extract $\dot{q}$ from $C(q,\dot{q})$ in (2) and rewrite (2) as

$$\tau = M(q)\ddot{q} + c(q,\dot{q})\dot{q} \quad (5)$$

Where

$$C(q,\dot{q}) = \begin{bmatrix} -m_2 l^2 \dot{q}_2 \sin q_2 - m_2 l^2 \dot{q}_1 \sin q_2 - m_2 l^2 \dot{q}_2 \sin q_2 \\ m_2 l^2 \dot{q}_1 \sin q_2 & 0 \end{bmatrix} \quad (6)$$

We define the tracking error as

$$e = q - q_d \quad (7)$$

Where $q = [q_1, q_2]^T$, $q_d = [q_{1d}, q_{2d}]^T$. The sliding surface is expressed as

$$s = \dot{e} + \lambda e \quad (8)$$

Where $\lambda = diag[\lambda_1, \lambda_2]$, λ_1 and λ_2 are chosen as the bandwidth of the robot controller.
We need to choose τ to satisfy the sufficient condition (9). We define the reference state as

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S}\cdot S = [f - \hat{f} - K\mathrm{sgn}(s)]\cdot S = (f - \hat{f})\cdot S - K|S| \quad (9)$$

$$\dot{q}_e = \dot{q} - s = \dot{q}_d - \lambda e \quad (10)$$

Now we pick the control input τ as

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r - As - Ksgn(s) \quad (11)$$

Where $M^{\wedge}$ and $C_1^{\wedge}$ are the estimations of $M(q)$ and $C_1(q,\dot{q})$; $A = diag[a_1, a_2]$ and $K = diag[k_1, k_2]$ are diagonal positive definite matrices. From (7) and (11), we can get

$$M\dot{s} + (C_1 + A)s = \Delta f - Ksgn(s) \quad (12)$$

Where $\Delta f = \Delta M \ddot{q}_r + \Delta C_1 \dot{q}_r$, $\Delta M = M^{\wedge} - M$ and $\Delta C_1 = C_1^{\wedge} - C_1$. We assume that the bound $|\Delta f_i|_{bound}$ of $\Delta f_i (i = 1,2)$ is known. We choose K as

$$K_i \geq |\Delta f_i|_{bound} \quad (13)$$

We pick the Lyapunov function candidate to be

$$V = \frac{1}{2}s^T M s \quad (14)$$

Since M is positive symmetric definite, $V > 0$ for $s \neq 0$. Take the derivative of M with respect to time in (6) and we get

$$\dot{M} = \begin{bmatrix} -2] m_2 l^2 \dot{q}_2 \sin q_2 - m_2 l^2 \dot{q}_2 \sin q_2 \\ -m_2 l^2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \quad (15)$$

From (11) and (15) we get

$$\dot{M} - 2C_1 = \begin{bmatrix} 0 & 2m_2 l^2 \dot{q}_1 \sin q_2 + m_2 l^2 \dot{q}_2 \sin q_2 \\ -2m_2 l^2 \dot{q}_1 \sin q_2 - m_2 l^2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \quad (16)$$

Which is a skew-systemmetric matrix satisfying

$$s^T(\dot{M} - 2C_1)s = 0 \quad (17)$$

Then $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= s^T M\dot{s} + \frac{1}{2}s^T \dot{M} s \\ &= s^T(M\dot{s} + C_1 s) \\ &= s^T[-As + \Delta f - Ksgn(s)] \\ &= \sum_{i=1}^{2}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T As \end{aligned} \quad (18)$$

For $K_i \geq |\Delta f_i|$, we always get $s_i[\Delta f_i - K_i sgn(s_i)] \leq 0$. We can describe $\dot{V}$ as

$$\dot{V} = \sum_{i=1}^{2}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T A s \leq -s^T\ As < 0 \quad (s \neq 0) \tag{19}$$

To attenuate chattering problem, we introduce a saturation function in the control law instead of the sign function in (9). The control law becomes

$$\tau = \hat{M}\ \ddot{q}_r + \hat{C}_1\dot{q}_r - As - Ksat(s/\Phi) \tag{20}$$

In this classical sliding mode control method, the model of the robotic manipulator is partly unknown. To attenuate chattering, we use the saturation function described in (20). Our control law changes to

$$\tau = \hat{M}\ \ddot{q}_r + \hat{C}_1\dot{q}_r - As - Ksat(s) \tag{21}$$

The main goal is to design a position controller for robot manipulator with acceptable performances (e.g., trajectory performance, torque performance, disturbance rejection, steady state error and RMS error). Robot manipulator has nonlinear dynamic and uncertain parameters consequently; following objectives have been pursuit in the mentioned study.

- To develop a chattering in a position pure sliding mode controller against uncertainties.
- To design and implement a position fuzzy estimator sliding mode controller in order to solve the equivalent problems in the pure sliding mode control.
- To develop a position sliding mode fuzzy adaptive fuzzy sliding mode controller in order to solve the disturbance rejection.

Figure 1 is shown the classical sliding mode methodology with linear saturation function to eliminate the chattering.

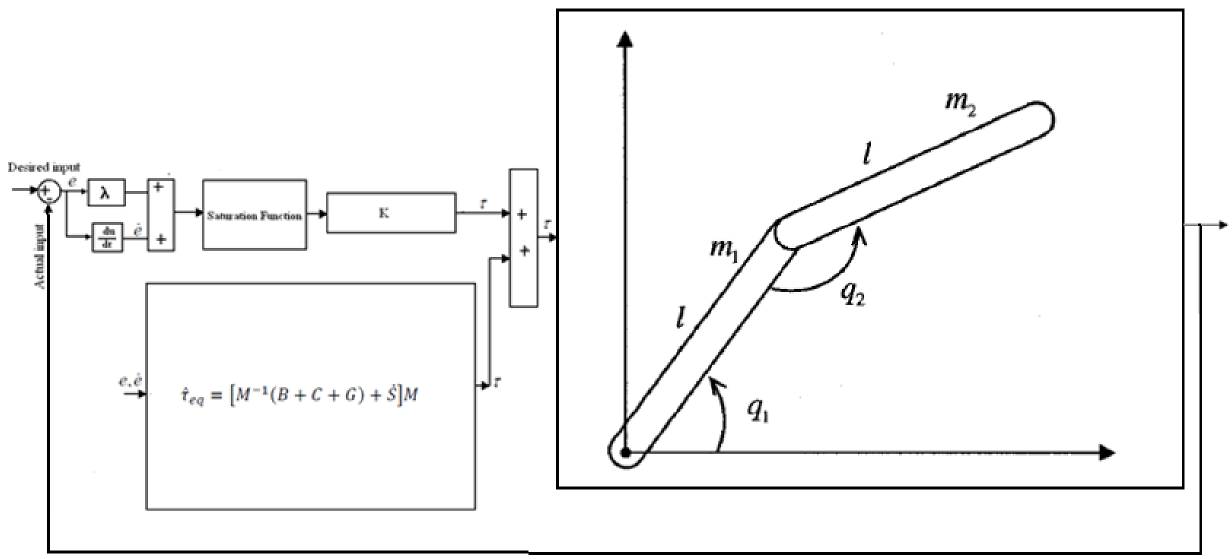


FIGURE 1: Classical sliding mode controller: applied to two-link robotic manipulator

3. METHODOLOGY: DESIGN A NOVEL PSO SISO LYAPUNOV BASED FUZZY ESTIMATOR SLIDING MODE ALGORITHM

First part is focuses on design chattering free sliding mode methodology using nonlinear saturation like algorithm. A time-varying sliding surface $s(x,t)$ is given by the following equation:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\ \tilde{x} = 0 \tag{22}$$

where λ is the constant and it is positive. The derivation of S, namely, $\dot{S}$ can be calculated as the following formulation [5-16, 41-62]:

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \tag{23}$$

The control law for a multi degrees of freedom robot manipulator is written as:

$$U = U_{eq} + U_r \quad (24)$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and it can be calculate as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \quad (25)$$

Where $M(q)$ is an inertia matrix which it is symmetric and positive, $V(q,\dot{q}) = B + C$ is the vector of nonlinearity term and $G(q)$ is the vector of gravity force and U_r with minimum chattering based on [9-16] is computed as;

$$U_r = K \cdot (\mathrm{mu} + \mathrm{b}) \left({}^{S}/_{\emptyset} \right) \quad (26)$$

Where $\emptyset_u = mu + b = Linear\ saturation_{fuction}$ is a dead zone (saturation) function and, u and b are unlimited coefficient, by replace the formulation (5) in (3) the control output can be written as;

$$U = U_{eq} + K.(\mathrm{mu} + \mathrm{b}) \left({}^{S}/_{\emptyset} \right) = \begin{cases} U_{eq} + K.\mathrm{sgn}(S) & ,|S| \geq \emptyset \\ U_{eq} + K.{}^{S}/_{\emptyset} & ,|S| < \emptyset \end{cases} \quad (27)$$

Where the function of $sgn(S)$ defined as;

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \quad (28)$$

Fuzzy logic is a multivalued logic, which can transfer mathematical equation of nonlinear dynamic parameter to expert mathematical knowledge [31]. A block diagram of fuzzy controller is shown in Figure 2. Even though the application area for fuzzy logic control is really wide, the basic form for all command types of controllers still consists of: Input fuzzification (binary-to-fuzzy [B/F] conversion), Fuzzy rule base, Inference engine, and Output deffuzzification (fuzzy-to-binary [F/B] conversion) [32-35].

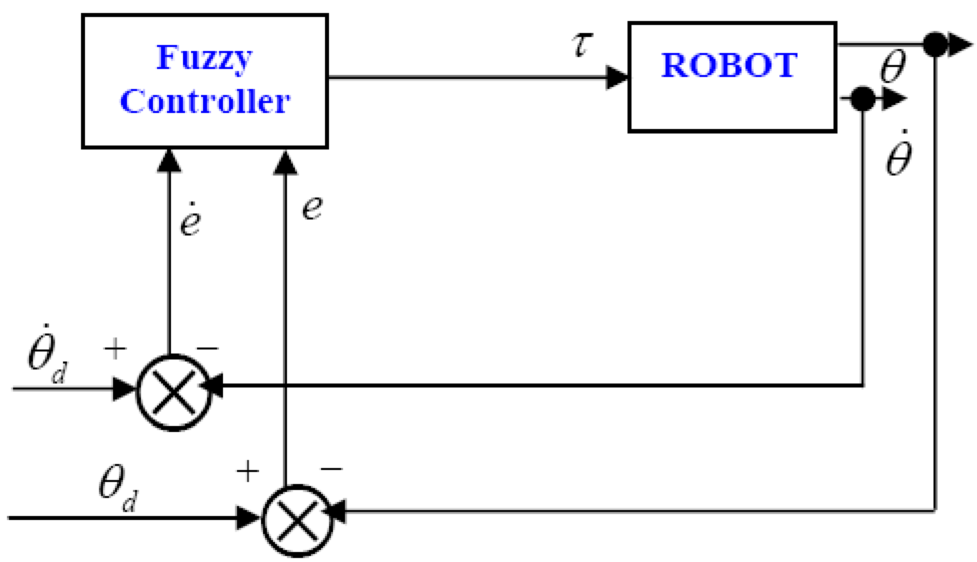


FIGURE 2: Fuzzy controller block diagram [40]

The basic structure of a fuzzy controller is shown in Figure 3.

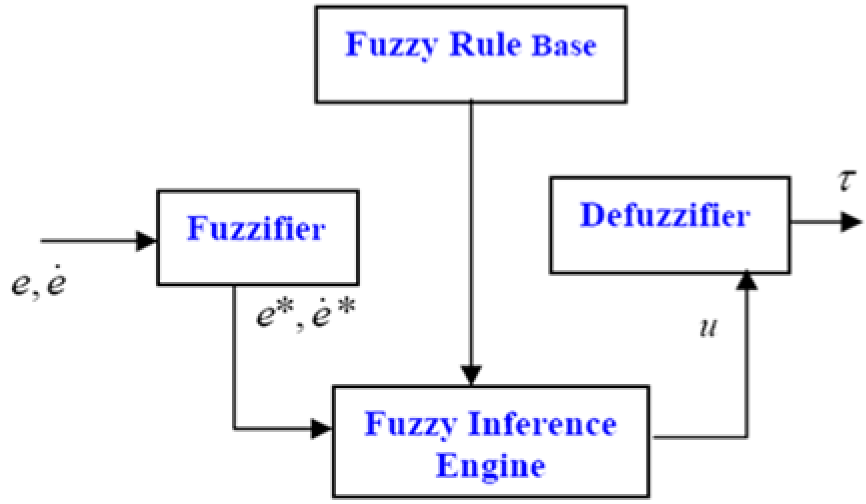


FIGURE 3: Structure of fuzzy logic controller [40]

To eliminate the chattering fuzzy inference system is used instead of linear saturation function to design nonlinear sliding function which as a summary the design of fuzzy logic controller for SMC has five steps:
Determine inputs and outputs: This controller has one input (S) and one output (α). The input is sliding function (S) and the output is coefficient which estimate the saturation function (α).
Find membership function and linguistic variable: The linguistic variables for sliding surface (S) are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and the linguistic variables to find the saturation coefficient (α) are; Large Left (LL), Medium Left (ML), Small Left (SL), Zero (Z), Small Right (SR), Medium Right (MR), Large Right (LR).
Choice of shape of membership function: In this work triangular membership function was selected.

Design fuzzy rule table: design the rule base of fuzzy logic controller can play important role to design best performance FSMC, suppose that two fuzzy rules in this controller are

F.R^1: *IF S is Z, THEN α is Z.* **(29)**
F.R^2: *IF S is (PB) THEN α is (LR).*

The complete rule base for this controller is shown in Table 1.

TABLE 1: Rule table for proposed FSMC

S	NB	NM	NS	Z	PS	PM	PB
τ	LL	ML	SL	Z	SR	MR	LR

The control strategy that deduced by Table1 are
- If sliding surface (S) is N.B, the control applied is N.B for moving S to S=0.
- If sliding surface (S) is Z, the control applied is Z for moving S to S=0.

Defuzzification: The final step to design fuzzy logic controller is deffuzification , there are many deffuzzification methods in the literature, in this controller the COG method will be used, where this is given by

$$COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)} \quad (30)$$

Second part is focuses on design fuzzy estimator to estimate nonlinear equivalent part. The fuzzy system can be defined as below [38-40]

$$f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(S) \quad (31)$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)^T, \zeta(x) = (\zeta^1(x), \zeta^2(x), \zeta^3(x), \dots\dots, \zeta^M(x))^T$

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \quad (32)$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots\dots, \theta^M)$ is adjustable parameter in (8) and $\mu_{(xi)}$ is membership function.

error base fuzzy controller can be defined as

$$U_{fuzzy} = \psi(S) \quad (33)$$

In this work the fuzzy controller has one input which names; sliding function. Fuzzy controller with one input is difficult to implementation, because it needs large number of rules, to cover equivalent part estimation [16-25]. Proposed method is used to a SISO fuzzy system which can approximate the residual coupling effect and alleviate the chattering. The robotic manipulator used in this algorithm is defined as below: the tracking error and the sliding surface are defined as:

$$e = q - q_d \quad (34)$$

$$s = \dot{e} + \lambda_e \quad (35)$$

We introduce the reference state as

$$\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e \quad (36)$$

$$\ddot{q}_r = \ddot{q} - \dot{s} = \ddot{q}_d - \lambda \dot{e} \quad (37)$$

The control input is given by

$$\tau = M^{\wedge} \ddot{q}_r + C_1^{\wedge} \dot{q}_r - As - K \quad (38)$$

Where $A = diag[a_1, \dots, a_m]$ and $a_1, \dots, a_m$ are positive constants; $K = [k_1, \dots, k_m]^T$ and K_j is defined as the fuzzy gain estimated by fuzzy systems.

The fuzzy if-then rules for the jth joint of the robotic manipulator are defined as

$$R^{(l)}: if\ s_j\ is\ A_1^l\ , then\ y\ is\ B_j^l \quad (39)$$

Where $j = 1, \dots, m$ and $l = 1, \dots, M$.

We define K_j by

$$K_j = \frac{\sum_{l=1}^{M} \theta_j^l \left[\mu_{A_j^l}(s_j)\right]}{\sum_{l=1}^{M} \left[\mu_{A_j^l}(s_j)\right]} = \theta_j^T \varepsilon_j(s_j) \quad (40)$$

Where

$$\varepsilon_j(s_j) = \left[\varepsilon_j^1(s_j), \varepsilon_j^2(s_j), \dots, \varepsilon_j^M(s_j)\right]^T, \quad (41)$$

$$\varepsilon_j^l(s_j) = \frac{\sum_{l=1}^{M} \mu_{A_j^l}(s_j)}{\sum_{l=1}^{M} \left[\mu_{A_j^l}(s_j)\right]} \quad (42)$$

The membership function $\mu_{A_j^l}(s_j)$ is a Gaussian membership function defined in bellows:

$$\mu_{A_j^l}(s_j) = \exp\left[-\left(\frac{s_j - \alpha_j^l}{\delta_j^l}\right)^2\right] \quad (j = 1, \dots, m). \quad (43)$$

The Lyapunov function candidate is given by

$$V = \frac{1}{2} s^T M s + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \emptyset_j^T \emptyset_j \quad (44)$$

Where $\phi_j = \theta_j^* - \theta_j$. The derivative of V is

$$\dot{V} = s^T M\dot{s} + \frac{1}{2}s^T \dot{M} s + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \tag{45}$$

Since $\dot{M} - 2C_1$ is a skew-symmetric matrix, we can get $s^T M\dot{s} + \frac{1}{2} s^T \dot{M} s = s^T (M\dot{s} + C_1 s)$. From (2) and (36), we get

$$\tau = M(q)\ddot{q} + c(q,\dot{q})\dot{q} + G(q) = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r + G^{\wedge} - As - K \tag{46}$$

Since $\dot{q}_r = \dot{q} - s$ and $\ddot{q}_r = \ddot{q} - \dot{s}$ in (44) and (45), we get

$$M\dot{s} + (C_1 + A)s = \Delta F - K \tag{47}$$

Where $\Delta F = \Delta M\ddot{q}_r + \Delta C_1\dot{q}_r + \Delta G$, $\Delta M = M^{\wedge} - M$, $\Delta C_1 = C_1^{\wedge} - C_1$ *and* $G = G^{\wedge} - G$. then $\dot{V}$ becomes

$$\begin{aligned}
\dot{V} &= s^T (M\dot{s} + C_1 s) + \textstyle\sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \\
&= -s^T (-As + \Delta f - K) + \textstyle\sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \\
&= \textstyle\sum_{j=1}^{m} [s_j(\Delta f_j - K_j)] - s^T As + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \\
&= \textstyle\sum_{j=1}^{m} [s_j(\Delta f_j - \theta_j^T \varepsilon_j(s_j))] - s^T As + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \\
&= \textstyle\sum_{j=1}^{m} (s_j [\Delta f_j - (\theta_j^*)\varepsilon_j(s_j) + \phi_j^T \varepsilon_j(s_j)]) - s^T As + \sum_{j=1}^{m} \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j^1 \\
&= \textstyle\sum_{j=1}^{m} (s_j [\Delta f_j - (\theta_j^*)\varepsilon_j(s_j)]) - s^T As + \sum_{j=1}^{m} \left(\frac{1}{\gamma_{sj}} \phi_j^T [\gamma_{sj} s_j \varepsilon_j(s_j) + \dot{\phi}_j] \right)
\end{aligned}$$

We choose the adaptation law $\dot{\theta}_j = \gamma_{sj} s_j \varepsilon_j(s_j)$. Since $\dot{\phi}_j = -\dot{\theta}_j = -\gamma_{sj} s_j \varepsilon_j(s_j)$, $\dot{V}$ becomes

$$\dot{V} = \textstyle\sum_{j=1}^{m} \left(s_j \left[\Delta f_j - (\theta_j^*)^T \varepsilon_j(s_j)\right] \right) - s^T As \tag{48}$$

We define the minimum approximation error as

$$\omega_j = \Delta f_j - (\theta_j^*)^T \varepsilon_j(s_j) \tag{49}$$

Then $\dot{V}$ change to

$$\begin{aligned}
\dot{V} &= \textstyle\sum_{j=1}^{m} s_j \omega_j - s^T As \\
&\le \sum_{j=1}^{m} |s_j||\omega_j| - s^T As \\
&= \sum_{j=1}^{m} (|s_j||\omega_j| - a_j s_j^2) \\
&= \sum_{j=1}^{m} \left(|s_j|(|\omega_j| - a_j |s_j|) \right)
\end{aligned} \tag{50}$$

According to Universal Approximation theorem in sliding mode algorithm, the minimum approximation error ω_j is as small as possible. We can simply pick a_j to make $a_j|s_j| > |\omega_j|$ $(s_j \neq 0)$. Then we get $\dot{V} < 0$ for $s \neq 0$.

The fuzzy division can be reached the best state when $S.\dot{S} < 0$ and the error is minimum by the following formulation

$$\theta^* = \arg\min \left[Sup_{x\in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - U_{equ} \right| \right] \tag{51}$$

Where $\theta^*$is the minimum error, $sup_{x\in U} \left| \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \right|$ is the minimum approximation error.

suppose K_j is defined as follows

$$K_j = \frac{\sum_{l=1}^{M} \theta_j^l [\mu_A(S_j)]}{\sum_{l=1}^{M} [\mu_A(S_j)]} = \theta_j^T \zeta_j(S_j) \quad \textbf{(52)}$$

Where $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad \textbf{(53)}$$

where the γ_{sj} is the positive constant.

According to the nonlinear dynamic equivalent formulation of robot manipulator the nonlinear equivalent part is estimated by (8)

$$[M^{-1}(B + C + G) + \dot{S}]M = \sum_{l=1}^{M} \theta^T \zeta(x) - \lambda S - K \quad \textbf{(54)}$$

Based on (3) the formulation of proposed fuzzy sliding mode controller can be written as;

$$U = U_{eq_{fuzzy}} + U_r \quad \textbf{(55)}$$

Where $U_{eq_{fuzzy}} = [M^{-1}(B + C + G) + \dot{S}]M + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

Figure 4 is shown the proposed fuzzy sliding mode controller.

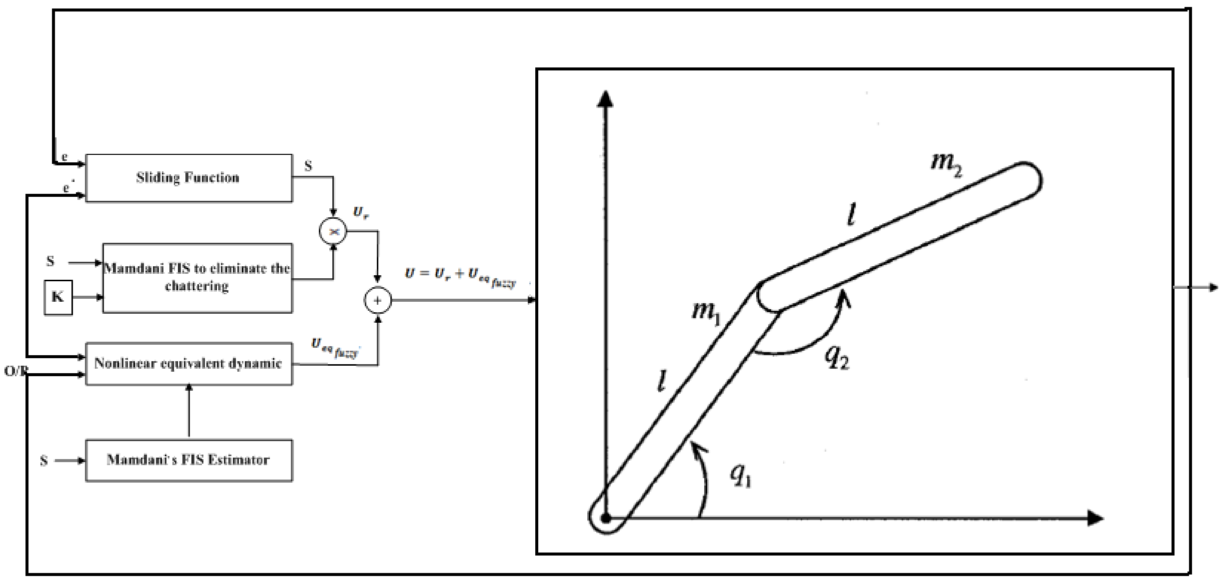


FIGURE 4: Proposed fuzzy estimator sliding mode algorithm: applied to robot manipulator

Figure 5 is shown the fuzzy instead of saturation function.

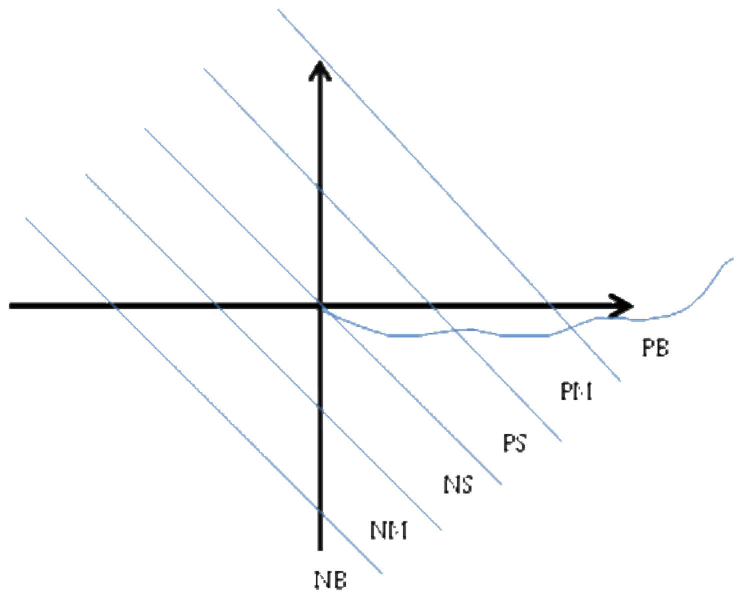


FIGURE 5: Nonlinear fuzzy inference system applied to linear saturation function

Parts three is focused on PSO algorithm and tune the coefficient of sliding function. Particle Swarm Optimization (PSO) is one of the evolutionary optimization algorithms in the branch of swarm intelligence developed by Eberhart and Kennedy in 1995[4]. This algorithm was inspired by the social movement behavior of the birds in the flock searching for food. Compared to the other evolutionary algorithms, the main excellences of this algorithm are: Simple concept, easy to implement, robustness in tuning parameters, minimum storage space and both global and local exploration capabilities. These birds in a flock are symbolically described as particles. These particles are supposed to a swarm "flying" through the problem space. Each particle has a position and a velocity. Any particle's position in the problem space has one solution for the problem. When a particle transfers from one place to another, a different problem solution is generated. Cost function evaluated the solution in order to provide the fitness value of a particle. "Best location" of each particle which has experienced up to now, is recorded in their memory, in order to determine the best fitness value. Particles of a swarm transmit the best location with each other to adapt their own location according to this best location to find the global minimum point. For every generation, the new location is computed by adding the particle's current velocity to its location. PSO is initialized with a random population of solutions in N-dimensional problem space, the i_{th} particle changes and updates its position and velocity according to the following formula:

$$V_{id} = w \times (V_{id} + C_1 \times rand_1 * (P_{id} - X_{id}) + C_2 \times rand_2 \times (P_{gd} - X_{id})) \quad \textbf{(56)}$$

Where X_{id} is calculated by

$$X_{id} = X_{id} + V_{id} \quad \textbf{(57)}$$

Where V_{id} is the inertia weight implies the speed of the particle moving along the dimensions in a problem space. C_1 and C_2 are acceleration parameters, called the cognitive and social parameters; $rand_1$ and $rand_2$ are functions that create random values in the range of (0, 1). X_{id} is the particle's current location; P_{id} (personal best) is the location of the particle experienced its personal best fitness value; P_{gd} (global best) is the location of the particle experienced the highest best fitness value in entire population; ***d*** is the number of dimensions of the problem space; . ***W*** is the momentum part of the particle or constriction coefficient [5] and it is calculated based on the following equation;

$$W = 2 / (2 - \varphi - \sqrt{\varphi^2 - 4\varphi}) \quad \textbf{(58)}$$

$$\varphi = C_1 + C_2 \quad , \quad \varphi > 4 \tag{59}$$

Equation 56 needs each particle to record its location X_{id}, its velocity V_{id}, its personal best fitness value P_{id}, and the whole population's best fitness value P_{gd}. On the basis of following equation the best fitness value X_i is updated at each generation, where the sign $f(.)$ represents the cost function; $X_i(.)$ indicated the best fitness values; and t denotes the generation step.

$$X_i(t+1) = \begin{cases} X_i(t) & f(P_d(t+1)) \le X_i(t) \\ f(P_d(t+1)) & f(P_d(t+1)) > X_i(t) \end{cases} \tag{60}$$

In PSO, the knowledge of each particle will not be substituted until the particle meets a new position vector with a higher competence value than the currently recorded value in its memory [6]. External disturbances influence on tracking trajectory, error rate and torque which result in chattering. But the values are not such a great values and these oscillations are in all physical systems. So, the sliding mode controller can reject perturbations and external disturbances if these parameters adjust properly. So the methodology which is applied in this paper in order to select the best values for these deterministic coefficients to accomplish high performance control is the particle swarm optimization algorithm. This algorithm tunes the gains and determines the appropriate values for these parameters in harmony with the system which was introduced in rear part.

4. RESULTS

Sliding mode controller (PD-SMC) and PSO SISO Lyapunov based fuzzy estimator sliding mode (PSO-FSMC) algorithms were tested to desired response trajectory. In this research the first and second joints are moved from home to final position without and with external disturbance. The simulation was implemented in Matlab/Simulink environments. Trajectory performance, torque performance and disturbance rejection are compared in these controllers. It is noted that, these systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude. This type of noise is used to external disturbance in continuous and hybrid systems.

Tracking performances: From the simulation for first and second trajectory without any disturbance with sinus trajectory, it was seen that both of controllers almost have the same performance, because these controllers are adjusted and worked on certain environment. Figure 6 is shown tracking performance in certain system and without external disturbance these two controllers.

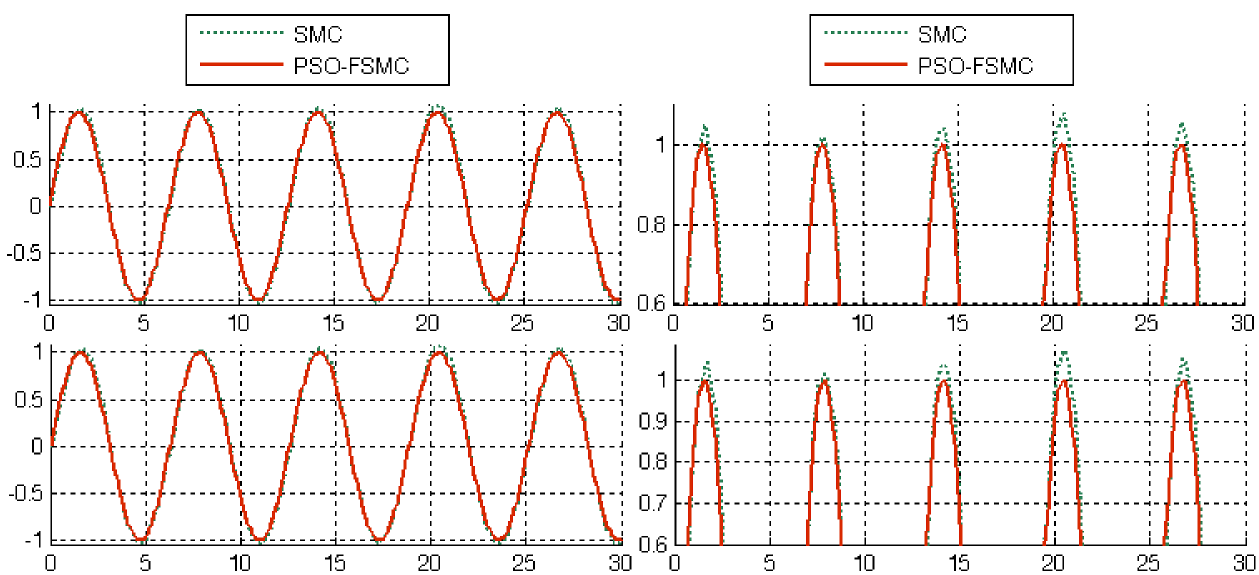


FIGURE 6: SMC Vs. PSO-FSMC: applied to 2-DOF serial robot manipulator

By comparing trajectory response in above graph it is found that the PSOFSMC undershoot **(0%)** is lower than SMC **(8.8%)**, although both of them have about the same overshoot.

Disturbance Rejection: Figure 7 has shown the power disturbance elimination in SMC and PSOFSMC. The main target in these controllers is disturbance rejection as well as reduces the chattering. A band limited white noise with predefined of 40% the power of input signal is applied to above controllers. It found fairly fluctuations in SMC trajectory responses.

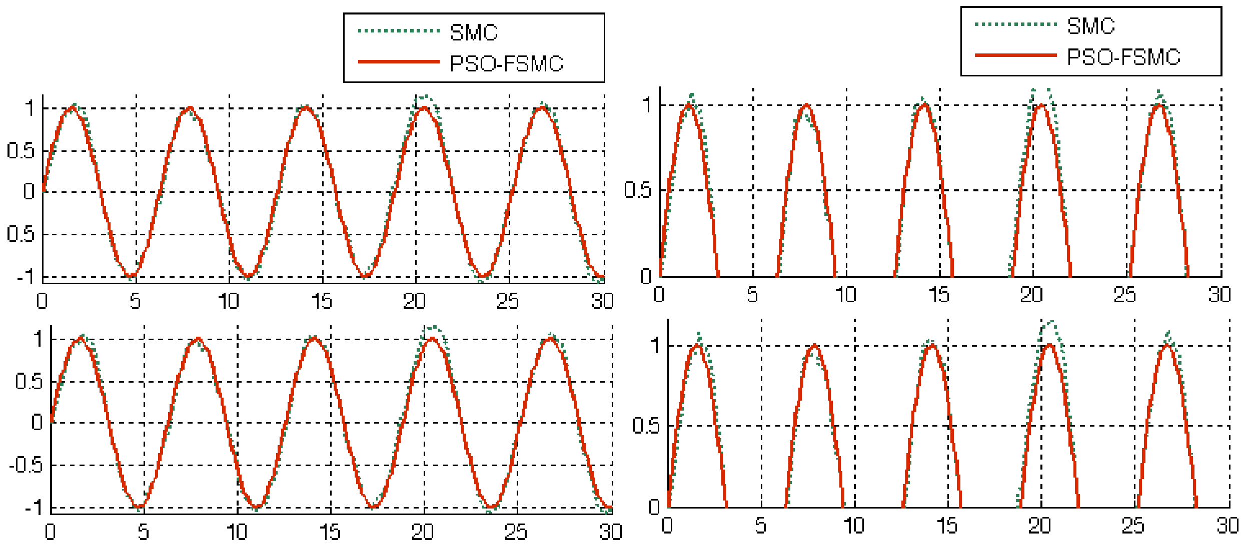


FIGURE 7: SMC Vs. PSO-FSMC with disturbance: applied to 2-DOF serial robot manipulator

Among above graph, relating to sinus trajectory following with external disturbance, pure SMC has slightly fluctuation. By comparing overshoot, rise time, and settling time; PSO SMC's overshoot **(0%)** is lower than SMC's **(11%)** and SMC's rise time **(0.66 sec)** is considerably the same as PSO SMC's **(0.61 sec)**.

Chattering Phenomenon: Chattering is one of the most important challenges in sliding mode controller for this reason the major objectives in this research is eliminate the chattering in controller's output. Figure 8 has shown the power of boundary layer (saturation) method to reduce the chattering in SMC.

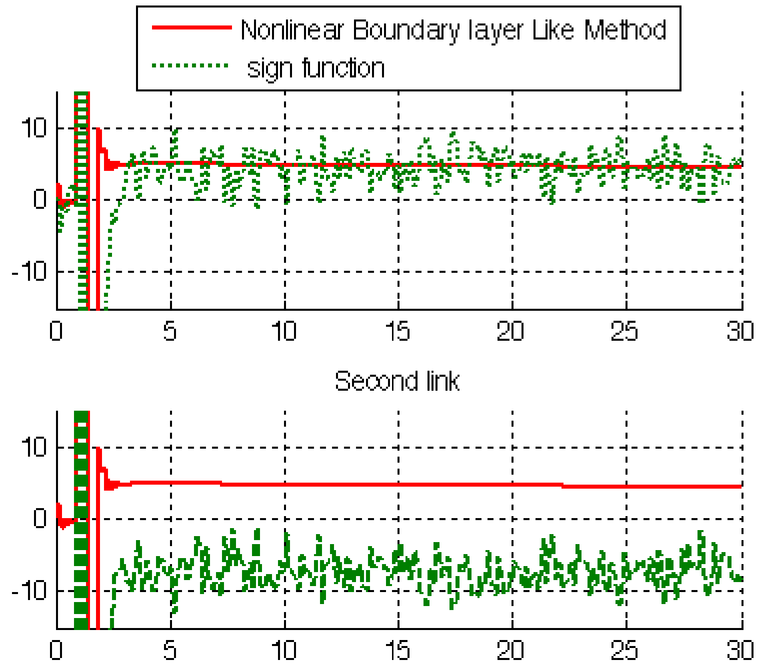


FIGURE 8: Sign Vs. Nonlinear boundary layer like method in PSO-FSMC: applied to 2-DOF serial robot manipulator

Figure 8 has indicated the power of chattering rejection in PSO-FSMC, with and without nonlinear boundary layer like method using fuzzy algorithm. Overall in this research with regard to the sinus response, PSO SMC has the steady chattering compared to the pure SMC in uncertain or/and external disturbance area with regard to nonlinear boundary layer like method using fuzzy inference system.

Error Calculation: Table 2 and Table 3are shown error performance in SMC and PSO-FSMC in presence of external disturbance. SMC has oscillation in tracking which causes chattering phenomenon.

TABLE 2: RMS Error Rate of Presented controllers

RMS Error Rate	SMC	PSO-FSMC
Without Noise	1e-3	0.9e-3
With Noise	0.012	0.00012

In these methods if integration absolute error (IAE) is defined by (61), table 2 is shown comparison between these two methods.

$$IAE = \int_0^\infty |e(t)|\, dt \tag{61}$$

TABLE 3: Calculate IAE

Method	Traditional SMC	PSO-Fuzzy Estimator SMC
IAE	430	209.1

5. CONCLUSION

In this work, a SISO PSO fuzzy estimate sliding mode controller is design, analysis and applied to robot manipulator. This method focuses on design PSO-FSMC algorithm with the formulation derived in the Lyapunov sense. The stability of the closed-loop system is proved mathematically based on the Lyapunov method. The first objective in proposed method is removed the chattering which non linear boundary layer method using fuzzy inference system is used to solve this challenge. The second target in this work is compensate the model uncertainty by SISO fuzzy inference system, in the case of the m-link robotic manipulator, if we define k_1 membership functions for each input variable, the number of fuzzy rules applied for each joint is K_1 which will result in a low computational load. In finally part PSO algorithm is used to off-line tuning and adjusted the sliding function and eliminates the chattering with minimum computational load. In this case the performance is improved by using the advantages of sliding mode algorithm, artificial intelligence compensate method and PSO algorithm while the disadvantages removed by added each method to previous method. Fuzzy logic method by adding to the sliding mode controller has covered negative points in fuzzy and sliding algorithms.

6. 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] 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.

[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] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

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

[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] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] Arquilla, J. and Ronfeldt, D., “Swarming and the Future of Conflict”. DB-311, RAND, SantaMonica, CA, 2000.

[64] Kennedy, J., “The Particle Swarm: Social Adaptation of Knowledge”, IEEE International conference on Evolutionary Computation, Indianapolis, Indiana, USA, pp. 303-308, 1997.

[65] Sun Tzu, “The Art of War” (tr. S.B Griffith), Oxford University Press 1963.

Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator

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

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

Amin Jalali Max.Jalali@gmail.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

Koorosh Aslansefat koo.ec2008@gmail.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

In this research, a Multi Input Multi Output (MIMO) position Field Programmable Gate Array (FPGA)-based fuzzy estimator sliding mode control (SMC) design with the estimation laws derived in Lyapunov sense and application to robotic manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties. Regarding to the positive points in sliding mode controller, fuzzy inference methodology and Lyapunov based method, the controllers output has improved. The main target in this research is analyses and design of the position MIMO artificial Lyapunov FPGA-based controller for robot manipulator in order to solve uncertainty, external disturbance, nonlinear equivalent part, chattering phenomenon, time to market and controller size using FPGA. Robot manipulators are nonlinear, time variant and a number of parameters are uncertain therefore design robust and stable controller based on Lyapunov based is discussed in this research. Studies about classical sliding mode controller (SMC) show that: although this controller has acceptable performance with known dynamic parameters such as stability and robustness but there are two important disadvantages as below: chattering phenomenon and mathematical nonlinear dynamic equivalent controller part. The first challenge; nonlinear dynamic part; is applied by inference estimator method in sliding mode controller in order to solve the nonlinear problems in classical sliding mode controller. And the second challenge; chattering phenomenon; is removed by linear method. Asymptotic stability of the closed loop system is also proved in the sense of Lyapunov. In the last part it can find the implementation of MIMO fuzzy estimator sliding mode controller on FPGA; FPGA-based fuzzy estimator sliding mode controller has many advantages such as high speed, low cost, short time to market and small device size. One of the most important drawbacks is limited capacity of available cells which this research focuses to solve this challenge. FPGA can be used to design a controller in a single chip Integrated Circuit (IC). In this research the SMC is designed using Very High Description Language (VHDL) for implementation on FPGA device (XA3S1600E-Spartan-3E), with minimum chattering.

Keywords: Mathematical Tunable, FPGA, MIMO Fuzzy Estimator, Fuzzy Sliding Mode Based Lyapunov Algorithm, Robot Manipulator, Chattering Phenomenon, VHDL language.

1. INTRODUCTION, MOTIVATION AND BACKGROUND

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. 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. 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 sentences because these topics played important roles to select and design the best acceptable performance controllers[1-6]. 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[7-12]. 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 Single Input Single Output (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].

There are several methods for controlling a robot manipulator, which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes[1]. Sliding mode controller (SMC) is a significant nonlinear controller in certain and uncertain dynamic parameters systems. This controller is used to present a systematic solution for stability and robustness which they can play important role to select the best controller. Conversely, pure sliding mode controller is used in many applications; it has two important drawbacks namely; chattering phenomenon, and nonlinear equivalent dynamic formulation in uncertain dynamic parameter [1, 6, and 20]. To reduce or eliminate the chattering this research is used linear saturation boundary layer function [13-20]. In boundary layer saturation method, the basic idea is the discontinuous method replacement by saturation (linear) method with small neighborhood of the switching surface. This replacement caused to increase the error performance against with the considerable chattering reduction. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode with boundary layer to improve the industry application [22]. R. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. As mentioned [24]sliding mode fuzzy controller (SMFC) is fuzzy controller based on sliding mode technique to simple implement, most exceptional stability and robustness. Conversely above method has the following advantages; reducing the number of fuzzy rule base and increasing robustness and stability, the main disadvantage of SMFC is need to define the sliding surface slope coefficient very carefully. To eliminate the above problems control researchers have applied artificial intelligence method (e.g., fuzzy logic) in nonlinear robust controller (e.g., sliding mode controller) besides this technique is very useful in order to implement easily. One of the most important techniques to reduce or remove above two challenges is applying non-classical (artificial intelligence) method in robust classical such as sliding mode controller method. Estimated uncertainty method used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [25-28]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Their design had three main parts i.e. equivalent, discontinuous and fuzzy estimator tuning part which has reduced the chattering very well. Elmali et al. [27]and Li and Xu [29]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering. This method was tested for the tracking control of the first two links of a SCARA type HITACHI robot. In this technique, digital controller is used to increase the system's response quality. Conversely this method has the following advantages; increasing the controller's response speed and reducing dependence on dynamic system model by on-line control, the main disadvantage are chattering phenomenon and need to improve the performance. In order to solve the uncertain dynamic parameters and complex parameters systems with an artificial intelligence theory, fuzzy logic is one of the best choice which it is used in this research. However fuzzy logic method is useful to control complicated nonlinear dynamic mathematical models but the response quality may not always be so high. This controller can be used in main part of controller (e.g., pure fuzzy logic controller), it can be used to design adaptive controller (e.g., adaptive fuzzy controller), tuning parameters and finally applied to the classical controllers [31-40]. 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 [24, 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].

As mentioned above sliding mode controller has some limitations which applied fuzzy logic in sliding mode controller can causes to reduce the limitations [48-53]. However FSMC has an acceptable performance but calculate the sliding surface slope by experience knowledge is difficult, particularly when system has structure or unstructured uncertainties, mathematical model free on-line tunable gain is recommended. 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.

Commonly, most of nonlinear controllers in robotic applications need a real time operation. FPGA-based controller has been used in this application because it is small device in size, high speed, low cost, and short time to market. Therefore FPGA-based controller can have a short execution time because it has parallel architecture. Research on FPGA-based control of systems is considerably growing as their applications such as industrial automation, robotic surgery, and space station's robot arm demand more accuracy, reliability, high performance. For instance, the FPGA-based controls of robot manipulator have been reported in [63-70]. Shao and Sun [64]have proposed an adaptive control algorithm based on FPGA for control of SCARA robot manipulator. They are designed this controller into two micro base controller, the linear part controller is implemented in the FPGA and the nonlinear estimation controller is implemented in DSP. Moreover this controller is implemented in a Xilinx-FPGA XC3S400 with the 20 KHz position loop frequency. The FPGA based servo control and inverse kinematics for Mitsubishi RV-M1 micro robot is presented in[65, 67] which to reduce the limitation of FPGA capacitance they are used 42 steps finite state machine (FSM) in 840 n second. Meshram and Harkare [68-69] have presented a multipurpose FPGA-based 5 DOF robot manipulator using VHDL coding in Xilinx ISE 11.1. This controller has two most important advantages: easy to implement and flexible. Zeyad Assi Obaid et al. [71] have proposed a digital PID fuzzy logic controller using FPGA for tracking tasks that yields semi-global stability of all closed-loop signals. The basic information about FPGA have been reported in [63, 69-73]. A review of design and implementation of FPGA-based systems has been presented in [63]. The FPGA-based sliding mode control of systems has been reported in [74-77]. Lin et al. [74] have presented low cost and high performance FPGA-based fuzzy sliding mode controller for linear induction motor with 80% of flip flops. The fuzzy inference system has 2 inputs $(S \,\&\, \dot{S})$ and one output K_f with nine rules. Ramos et al. [75] have reported FPGA-based fixed frequency quasi sliding mode control algorithm to control of power inverter. Their proposed controller is implemented in XC4010E-3-PC84 FPGA from XILINX with acceptable experimental and theoretical performance. FPGA-based robust adaptive backstepping sliding mode control for verification of induction motor is reported in [76]. A FPGA chip has programmed by Hardware Description Language (HDL) which contains two types of languages, Very High Description Language (VHDL) and Verilog. VHDL is one of the powerful programming languages that can be used to describe the hardware design. VHDL was developed by the Institute of Electrical and Electronics Engineers (IEEE) in 1987 and Verilog was developed by Gateway Design Automation in 1984 [63, 72]. This research focuses on FPGA-based sliding mode control of robot manipulator and it is implemented in XA3S1600E FPGA from Xilinx in Xilinx-ISE 9.2i software using VHDL code.

In this research we will highlight the MIMO mathematical model-free adaptive fuzzy estimator sliding mode algorithm with estimates the equivalent part derived in the Lyapunov sense. This algorithm will be analyzed and evaluated on robotic manipulators. Section 2, serves as an introduction to problem formulation of controller and its application to a three degree of-freedom robot manipulator, describe the objectives and problem statements. Part 3, introduces and describes the methodology algorithms and proves Lyapunov stability. Section 4 presents the simulation results of this algorithm applied to a 3 degree-of-freedom robot manipulator and the final section is describe the conclusion.

2. CONTROL FORMULATION, OBJECTIVES AND PROBLEM STATEMENTS

Fist part is focused on nonlinear dynamic of PUMA 560 robot manipulator. 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]:

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

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 1 if vector of nonlinearity term derive as Centrifugal, Coriolis and Gravity terms, as a result robot manipulator dynamic equation can also be written as [78-79]:

$$N(q,\dot{q}) = V(q,\dot{q}) + G(q) \quad \textbf{(2)}$$

$$V(q,\dot{q}) = B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 \quad \textbf{(3)}$$

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) \quad \textbf{(4)}$$

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 (4) it can be written as [1, 6, 80-81];

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \quad \textbf{(5)}$$

To implementation (5) 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;

$$M(\theta) = m_1 J_{v1}^T J_{v1} + J_{\omega 1}^{TC1} I_1 J_{\omega 1} + m_2 J_{v2}^T J_{v2} + J_{\omega 2}^{TC2} I_2 J_{\omega 2} + m_3 J_{v3}^T J_{v3} + J_{\omega 3}^{TC3} I_3 J_{\omega 3} + m_4 J_{v4}^T J_{v4} \quad \textbf{(6)}$$
$$m_5 J_{v5}^T J_{v5} + J_{\omega 5}^{TC5} I_5 J_{\omega 5} + m_6 J_{v6}^T J_{v6} + J_{\omega 6}^{TC6} I_6 J_{\omega 6}$$

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]

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & \cdots & \cdots & \cdots & M_{1n} \\ M_{21} & \cdots & \cdots & \cdots & \cdots & M_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ M_{n,1} & \cdots & \cdots & \cdots & \cdots & M_{n,n} \end{bmatrix} \tag{7}$$

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

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & \cdots & b_{11n} & b_{123} & \cdots & b_{12n} & \cdots & \cdots & b_{1,n-1,n} \\ b_{212} & \cdots & \cdots & b_{21n} & b_{223} & \cdots & \cdots & \cdots & \cdots & b_{2,n-1,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ b_{n,1,2} & \cdots & \cdots & b_{n,1,n} & \cdots & \cdots & \cdots & \cdots & \cdots & b_{n,n-1,n} \end{bmatrix} \tag{8}$$

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

$$C(q) = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \tag{9}$$

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

$$G(q) = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \tag{10}$$

Robotic control is one of the most active research areas in the field of robotics and one of the well-known robot manipulator in the field of academic and industries is PUMA 560 robot manipulator. To position control of robot manipulator, the second three axes are locked the dynamic equation of PUMA robot manipulator is given by [78-79];

$$M(\theta) \begin{bmatrix} \ddot{\theta}1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + B(\theta) \begin{bmatrix} \dot{\theta}_1\dot{\theta}_2 \\ \dot{\theta}_1\dot{\theta}_3 \\ \dot{\theta}_2\dot{\theta}_3 \end{bmatrix} + C(\theta) \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + G(\theta) = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{11}$$

Where

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

M is computed as

$$M_{11} = I_{m1} + I_1 + I_3 \times \cos(\theta_2)\cos(\theta_2) + I_7\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + I_{10}\sin(\theta_2+\theta_3) \\ I_{11}\sin(\theta_2)\cos(\theta_2) + I_{21}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + 2 + [I_5\cos(\theta_2)\sin(\theta_2+\theta_3) + I_{12}\cos \\ \theta_3) + I_{15}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + I_{16}\cos(\theta_2)\sin(\theta_2+\theta_3) + I_{22}\sin(\theta_2+\theta_3)\cos(\theta_2 + \tag{13}$$

$$M_{12} = I_4\sin(\theta_2) + I_8\cos(\theta_2+\theta_3) + I_9 cos(\theta_2) + I_{13}\sin(\theta_2+\theta_3) - I_{18}\cos(\theta_2+\theta_3) \tag{14}$$

$$M_{13} = I_8\cos(\theta_2+\theta_3) + I_{13}\sin(\theta_2+\theta_3) - I_{18}\cos(\theta_2+\theta_3) \tag{15}$$

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

$$M_{23} = I_5\sin(\theta_3) + I_6 + I_{12}\cos(\theta_3) + I_{16}\sin(\theta_3) + 2I_{15} \quad (17)$$

$$M_{33} = I_{m3} + I_6 + 2I_{15} \quad (18)$$

$$M_{35} = I_{15} + I_{17} \quad (19)$$

$$M_{44} = I_{m4} + I_{14} \quad (20)$$

$$M_{55} = I_{m5} + I_{17} \quad (21)$$

$$M_{66} = I_{m6} + I_{23} \quad (22)$$

$$M_{21} = M_{12}\ , M_{31} = M_{13}\ and\ M_{32} = M_{23} \quad (23)$$

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

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (24)$$

Where,

$$b_{112} = 2[-I_3\sin(\theta_2)\cos(\theta_2) + I_5\cos(\theta_2 + \theta_2 + \theta_3) + I_7\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) - I_{12}s \quad (25)$$
$$\theta_3) - I_{15}2\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) + I_{16}\cos(\theta_2 + \theta_2 + \theta_3) + I_{21}\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta$$
$$2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3))] + I_{10}(1 - 2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)) + I_{11}(1 - 2\sin(\theta_2)si$$

$$b_{113} = 2[I_5\cos(\theta_2)\cos(\theta_2 + \theta_3) + I_7\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) - I_{12}\cos(\theta_2)\sin(\theta_2 + \theta_2 \quad (26)$$
$$\theta_3)\cos(\theta_2 + \theta_3) + I_{16}\cos(\theta_2)\cos(\theta_2 + \theta_3) + I_{21}\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) + I_{22}(1 - 2\sin$$
$$\theta_3)\sin(\theta_2 + \theta_3))] + I_{10}(1 - 2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3))$$

$$b_{115} = 2[-\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) + I_{15}2\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) + I_{16}\cos(\theta_2)\cos(\theta \quad (27)$$
$$I_{22}\cos(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3)]$$

$$b_{123} = 2[-I_8\sin(\theta_2 + \theta_3) + I_{13}\cos(\theta_2 + \theta_3) + I_{18}\sin(\theta_2 + \theta_3)] \quad (28)$$

$$b_{214} = I_{14}\sin(\theta_2 + \theta_3) + I_{19}\sin(\theta_2 + \theta_3) + 2I_{20}\sin(\theta_2 + \theta_3)(1 - 0.5) \quad (29)$$

$$b_{223} = 2[-I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3)] \quad (30)$$

$$b_{235} = 2[I_{16}\cos(\theta_3) + I_{22}] \quad (31)$$

$$b_{314} = 2[I_{20}\sin(\theta_2 + \theta_3)(1 - 0.5)] + I_{14}\sin(\theta_2 + \theta_3) + I_{19}\sin(\theta_2 + \theta_3) \quad (32)$$

$$b_{412} = b_{214} = -[I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta_3) + 2I_{20}\sin(\theta_2+\theta_3)(1-0.5)] \quad (33)$$

$$b_{413} = -b_{314} = -2[I_{20}\sin(\theta_2+\theta_3)(1-0.5)] + I_{14}\sin(\theta_2+\theta_3) + I_{19}\sin(\theta_2+\theta_3) \quad (34)$$

$$b_{415} = -I_{20}\sin(\theta_2+\theta_3) - I_{17}\sin(\theta_2+\theta_3) \quad (35)$$

$$b_{514} = -b_{415} = I_{20}\sin(\theta_2+\theta_3) + I_{17}\sin(\theta_2+\theta_3) \quad (36)$$

consequently coriolis matrix is shown as bellows;

$$B(q).\dot{q}\dot{q} = \begin{bmatrix} b_{112}.\dot{q}_1\dot{q}_2 + b_{113}.\dot{q}_1\dot{q}_3 + 0 + b_{123}.\dot{q}_2\dot{q}_3 \\ 0 + b_{223}.\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}.\dot{q}_1\dot{q}_2 + b_{413}.\dot{q}_1\dot{q}_3 + 0 \\ 0 \\ 0 \end{bmatrix} \quad (37)$$

Moreover Centrifugal (C) matrix is demonstrated as

$$C(q) = \begin{bmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{21} & 0 & C_{23} & 0 & 0 & 0 \\ C_{31} & C_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ C_{51} & C_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (38)$$

Where,

$$c_{12} = I_4\cos(\theta_2) - I_8\sin(\theta_2+\theta_3) - I_9\sin(\theta_2) + I_{13}\cos(\theta_2+\theta_3) + I_{18}\sin(\theta_2+\theta_3) \quad (39)$$

$$c_{13} = 0.5b_{123} = -I_8\sin(\theta_2+\theta_3) + I_{13}\cos(\theta_2+\theta_3) + I_{18}\sin(\theta_2+\theta_3) \quad (40)$$

$$c_{21} = -0.5b_{112} = I_3\sin(\theta_2)\cos(\theta_2) - I_5\cos(\theta_2+\theta_2+\theta_3) - I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_{\cdots} \quad (41)$$
$$\theta_2+\theta_3) + I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2+\theta_2+\theta_3) - I_{21}\sin(\theta_2+\theta_3)\cos(\cdots$$
$$I_{22}(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)) - 0.5I_{10}(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)) -$$
$$0.5I_{11}(1-2\sin(\theta_2)\sin(\theta_2))$$

$$c_{22} = 0.5b_{223} = -I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3) \quad (42)$$

$$c_{23} = -0.5b_{113} = -I_5\cos(\theta_2)\cos(\theta_2+\theta_3) - I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{12}\cos(\theta_2)\cdots \quad (43)$$
$$I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2)\cos(\theta_2+\theta_3) - I_{21}\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3)\cdots$$
$$2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)) - 0.5I_{10}(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3))$$

$$c_{31} = -c_{23} = I_{12}\sin(\theta_3) - I_5\cos(\theta_3) - I_{16}\cos(\theta_3) \quad (44)$$

$$c_{32} = -0.5b_{115} = \sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_{\cdots} \quad (45)$$
$$I_{22}\cos(\theta_2+\theta_3)\cos(\theta_2+\theta_3)$$

$$c_{52} = -0.5b_{225} = -I_{16}\cos(\theta_3) - I_{22} \quad (46)$$

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

$$C(q).\dot{q}^2 = \begin{bmatrix} c_{112}.\dot{q}_2^2 + c_{13}.\dot{q}_3^2 \\ c_{21}.\dot{q}_1^2 + c_{23}.\dot{q}_3^2 \\ c_{13}.\dot{q}_1^2 + c_{32}.\dot{q}_2^2 \\ 0 \\ c_{51}.\dot{q}_1^2 + c_{52}.\dot{q}_2^2 \\ 0 \end{bmatrix} \tag{47}$$

Gravity (G) Matrix can be written as

$$G(q) = \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{48}$$

Where,

$$G_2 = g_1 \cos(\theta_2) + g_2 \sin(\theta_2 + \theta_3) + g_3 \sin(\theta_2) + g_4 \cos(\theta_2 + \theta_3) + g_5 \sin(\theta_2 + \theta_3) \tag{49}$$

$$G_3 = g_2 \sin(\theta_2 + \theta_3) + g_4 \cos(\theta_2 + \theta_3) + g_5 \sin(\theta_2 + \theta_3) \tag{50}$$

$$G_5 = g_5 \sin(\theta_2 + \theta_3) \tag{51}$$

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

$$\ddot{q} = M^{-1}(q).\{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{52}$$

and K is introduced as

$$K = \{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \tag{53}$$

$\ddot{q}$ can be written as

$$\ddot{q} = M^{-1}(q).K \tag{54}$$

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

$$K_1 = \tau_1 - [\, b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\,\dot{q}_2\dot{q}_3] - [\, C_{12}\,\dot{q}_2{}^2 + C_{13}\,\dot{q}_3{}^2] - g_1 \tag{55}$$

$$K_2 = \tau_2 - [\, b_{223}\,\dot{q}_2\dot{q}_3] - [\, C_{21}\,\dot{q}_1{}^2 + C_{23}\,\dot{q}_3{}^2] - g_2 \tag{56}$$

$$K_3 = \tau_3 - [C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2] - g_3 \tag{57}$$

$$K_4 = \tau_4 - [\, b_{412}\,\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3] - g_4 \tag{58}$$

$$K_5 = \tau_5 - [\, C_{51}\,\dot{q}_1{}^2 + C_{52}\,\dot{q}_2{}^2] - g_5 \tag{59}$$

$$K_6 = \tau_6 \tag{60}$$

An information about inertial constant and gravitational constant are shown in Tables 1 and 2 based on [78-79].

TABLE 1: Inertial constant reference (*Kg.m²*)

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

TABLE 2: Gravitational constant (*N.m*)

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

Second part is focused on sliding mode formulation and its challenge. We define the tracking error as

$$e = q - q_d \tag{61}$$

Where $q = [q_1, q_2]^T$, $q_d = [q_{1d}, q_{2d}]^T$. The sliding surface is expressed as

$$s = \dot{e} + \lambda e \tag{62}$$

Where $\lambda = diag[\lambda_1, \lambda_2]$, λ_1 and λ_2 are chosen as the bandwidth of the robot controller.
We need to choose τ to satisfy the sufficient condition (63). We define the reference state as

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S} \cdot S = [f - \hat{f} - K\operatorname{sgn}(s)] \cdot S = (f - \hat{f}) \cdot S - K|S| \tag{63}$$

$$\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e \tag{64}$$

Now we pick the control input τ as

$$\tau = M^{\wedge}\ddot{q}_r + C^{\wedge}\dot{q}_r + \hat{B}[\dot{q}\dot{q}] + \hat{G} - As - Ksgn(s) \quad (65)$$

Where $M^{\wedge}$ and $C_1^{\wedge}$ are the estimations of $M(q)$ and $C_1(q,\dot{q})$; $A = diag[a_1, a_2]$ and $K = diag[k_1, k_2]$ are diagonal positive definite matrices. From (61) and (65), we can get

$$M\dot{s} + (C + B + G + A)s = \Delta f - Ksgn(s) \quad (66)$$

Where $\Delta f = \Delta M\ddot{q}_r + \Delta C\dot{q}_r + \Delta B[\dot{q}\dot{q}] + G$, $\Delta M = M^{\wedge} - M$ and $\Delta C = C^{\wedge} - C$. We assume that the bound $|\Delta f_i|_{bound}$ of $\Delta f_i (i = 1, 2, ..., N)$ is known. We choose K as

$$K_i \geq |\Delta f_i|_{bound} \quad (67)$$

We pick the Lyapunov function candidate to be

$$V = \frac{1}{2}s^T Ms \quad (68)$$

Which is a skew-systemmetric matrix satisfying

$$s^T\left(\dot{M} - 2(C + B + G + A)\right)s = 0 \quad (69)$$

Then $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= s^T M\dot{s} + \frac{1}{2}s^T\dot{M}s \\ &= s^T(M\dot{s} + (C + B + G + A)s) \\ &= s^T[-As + \Delta f - Ksgn(s)] \\ &= \sum_{i=1}^{N}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T As \end{aligned} \quad (70)$$

For $K_i \geq |\Delta f_i|$, we always get $s_i[\Delta f_i - K_i sgn(s_i)] \leq 0$. We can describe $\dot{V}$ as

$$\dot{V} = \sum_{i=1}^{N}(s_i[\Delta f_i - K_i sgn(s_i)]) - s^T As \leq -s^T As < 0 \quad (s \neq 0) \quad (71)$$

To attenuate chattering problem, we introduce a saturation function in the control law instead of the sign function in (63). The control law becomes

$$\tau = M^{\wedge}\ddot{q}_r + C^{\wedge}\dot{q}_r + \hat{B}[\dot{q}\dot{q}] + \hat{G} - As - Ksat(s/\Phi) \quad (72)$$

In this classical sliding mode control method, the model of the robotic manipulator is partly unknown. To attenuate chattering, we use the saturation function described in (72). Our control law changes to

$$\tau = M^{\wedge}\ddot{q}_r + C^{\wedge}\dot{q}_r + \hat{B}[\dot{q}\dot{q}] + \hat{G} - As - Ksat(s) \quad (73)$$

The control law for a multi degrees of freedom robot manipulator is written as:

$$U = U_{eq} + U_r \quad (74)$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and it can be calculate as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \quad (75)$$

Where $M(q)$ is an inertia matrix which it is symmetric and positive, $V(q,\dot{q}) = B + C$ is the vector of nonlinearity term and $G(q)$ is the vector of gravity force and U_r with minimum chattering based on [5-11] is computed as;

$$U_r = K \cdot (\mathrm{mu} + \mathrm{b})\left({S}/{\emptyset}\right) \quad (76)$$

Where $\emptyset_u = mu + b = saturation_{fuction}$ is a dead zone (saturation) function and, u and b are unlimited coefficient, by replace the formulation (76) in (74) the control output can be written as;

$$U = U_{eq} + K.(\mathrm{mu} + \mathrm{b})\left({S}/{\emptyset}\right) = \begin{cases} U_{eq} + K.\mathrm{sgn}(S) & ,|S| \geq \emptyset \\ U_{eq} + K.{S}/{\emptyset} & ,|S| < \emptyset \end{cases} \quad \textbf{(77)}$$

Where the function of $sgn(S)$ defined as;

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \quad \textbf{(78)}$$

The main goal is to design a FPGA based adaptive mathematical model free MIMO fuzzy estimator sliding mode controller. Based on above robot manipulator has nonlinear and highly uncertain parameters consequently; following objectives have been pursuit in this paper.

- To develop a chattering in a position pure sliding mode controller against uncertainties via linear boundary layer method.
- To design and implement a MIMO fuzzy estimator sliding mode controller in order to solve the equivalent problems in the pure sliding mode control with minimum rule base based on Lyapunov formulation.
- To develop a mathematical model free adaptive fuzzy estimator sliding mode controller in order to solve the disturbance rejection and reduce the fuzzy rule base.
- To design and implement a FPGA based mathematical model free adaptive fuzzy estimator sliding mode controller

Figure 1 is shown the classical sliding mode methodology with linear saturation function to eliminate the chattering.

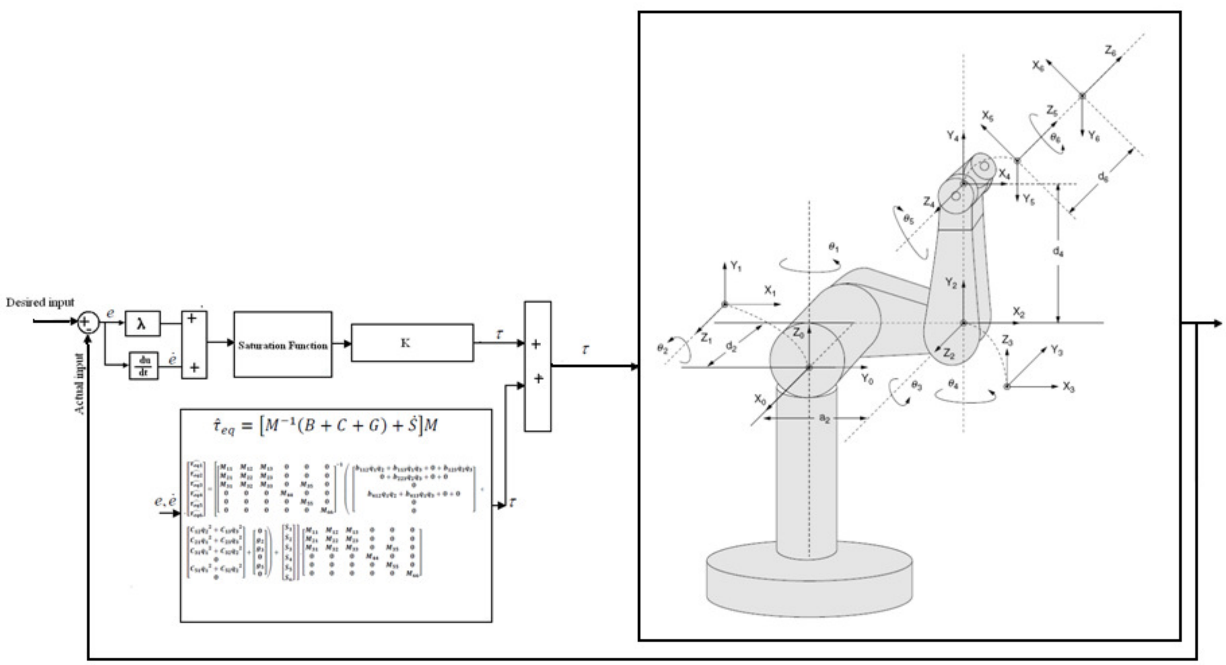


FIGURE 1: Classical sliding mode controller: applied to 6-link robotic manipulator

Zadeh introduced fuzzy sets in 1965. After 40 years, fuzzy systems have been widely used in different fields, especially on control problems. Fuzzy systems transfer expert knowledge to mathematical models. Fuzzy systems used fuzzy logic to estimate dynamics of our systems. Fuzzy controllers including fuzzy if-then rules are used to control our systems. 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) [30-40].

The basic structure of a fuzzy controller is shown in Figure 2.

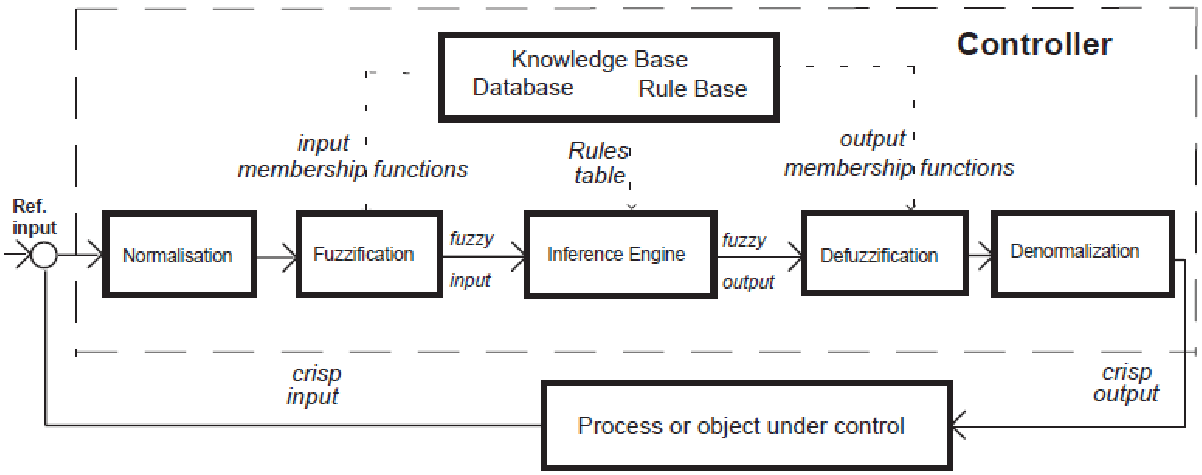


FIGURE 2: Block diagram of a fuzzy controller with details.

Conventional control methods use mathematical models to controls systems. Fuzzy control methods replace the mathematical models with fuzzy if then-rules and fuzzy membership function to controls systems. Both fuzzy and conventional control methods are designed to meet system requirements of stability and convergence. When mathematical models are unknown or partially unknown, fuzzy control models can used fuzzy systems to estimate the unknown models. This is called the model-free approach [31, 35].

3. METHODOLOGY: DESIGN A NOVEL FPGA BASED MIMO ADAPTIVE MATHEMATICAL MODEL FREE LYAPUNOV BASED FUZZY ESTIMATE SLIDING MODE CONTROL

Conventional control models can use adaptive control methods to achieve the model-free approach. When system dynamics become more complex, nonlinear systems are difficult to handle by conventional control methods. Fuzzy systems can approximate arbitrary nonlinear systems. In practical problems, systems can be controlled perfectly by expert. Experts provide linguistic description about systems. Conventional control methods cannot design controllers combined with linguistic information. When linguistic information is important for designing controllers, we need to design fuzzy controllers for our systems. Fuzzy control methods are easy to understand for designers. The design process of fuzzy controllers can be simplified with simple mathematical models. Adaptive control uses a learning method to self-learn the parameters of systems. For system whose dynamics are varying, adaptive control can learn the parameters of system dynamics. In traditional adaptive control, we need some information about our system such as the structure of system or the order of the system. In adaptive fuzzy control we can deal with uncertain systems. Due to the linguistic characteristic, adaptive fuzzy controllers behave like operators: adaptively controlling the system under various conditions. Adaptive fuzzy control provides a good tool for making use of expert knowledge to adjust systems. This is important for a complex unknown system with changing dynamics. We divide adaptive fuzzy control into two categories: direct adaptive fuzzy control and indirect adaptive fuzzy control. A direct adaptive fuzzy controller adjusts the parameters of the control input. An indirect adaptive fuzzy controller adjusts the parameters of the control system based on the estimated dynamics of the plant.

We define fuzzy systems as two different types. The firs type of fuzzy systems is given by

$$f(x) = \sum_{l=1}^{M} \theta^l \mathcal{E}^l(x) = \theta^T \mathcal{E}(x) \tag{79}$$

Where $\theta = (\theta^1, \dots, \theta^M)^T, \mathcal{E}(x) = (\mathcal{E}^1(x), \dots, \mathcal{E}^M(x))^T, and\ \mathcal{E}^l(x) =: \prod_{i=1}^{n} \frac{\mu_{A_i^l}(x_i)}{} \sum_{l=1}^{M}(\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$. $\theta^1, \dots, \theta^M$ are adjustable parameters in (18) . $\mu_{A_1^1}(x_1), \dots, \mu_{A_n^m}(x_n)$ are given membership functions whose parameters will not change over time.

The second type of fuzzy systems is given by

$$f(x) = \frac{\sum_{l=1}^{M} \theta^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]} \tag{80}$$

Where θ^l , α_i^l and δ_i^l are all adjustable parameters.
From the universal approximation theorem, we know that we can find a fuzzy system to estimate any continuous function. For the first type of fuzzy systems, we can only adjust θ^l in (79). We define $f^{\wedge}(x|\theta)$ as the approximator of the real function $f(x)$.

$$f^{\wedge}(x|\theta) = \theta^T \varepsilon(x) \tag{81}$$

We define θ^* as the values for the minimum error:

$$\theta^* = \arg \min_{\theta \in \Omega} \left[\sup_{x \in U} |f^{\wedge}(x|\theta) - g(x)|\right] \tag{82}$$

Where Ω is a constraint set for θ. For specific x ,$sup_{x\in U} |f^{\wedge}(x|\theta^*) - f(x)|$ is the minimum approximation error we can get.

We used the first type of fuzzy systems (79) to estimate the nonlinear system (75) the fuzzy formulation can be write as below;

$$f(x|\theta) = \theta^T \varepsilon(x) = \frac{\sum_{l=1}^{n} \theta^l \left[\mu_{A^l}(x)\right]}{\sum_{l=1}^{n} [\mu_{A^l}(x)]} \tag{83}$$

Where $\theta^1, \dots, \theta^n$ are adjusted by an adaptation law. The adaptation law is designed to minimize the parameter errors of $\theta - \theta^*$. A MIMO (multi-input multi-output) fuzzy system is designed to compensate the uncertainties of the robotic manipulator. The parameters of the fuzzy system are adjusted by adaptation laws. The tracking error and the sliding surface state are defined as (58-62)

$$e = q - q_d \tag{84}$$

$$s = \dot{e} + \lambda_e \tag{85}$$

We define the reference state as

$$\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e \tag{86}$$

$$\ddot{q}_r = \ddot{q} - \dot{s} = \ddot{q}_d - \lambda \dot{e} \tag{87}$$

The general MIMO if-then rules are given by

$$R^l: if\ x_1\ is\ A_1^l\ , x_2\ is\ A_2^l\ , \dots, x_n\ is\ A_n^l\ , then\ y_1 is\ B_1^l\ , \dots, y_m\ is\ B_m^l \tag{88}$$

Where $l = 1,2,\dots,M$ are fuzzy if-then rules; $x = (x_1,\dots,x_n)^T$ and $y = (y_1,\dots,y_n)^T$ are the input and output vectors of the fuzzy system. The MIMO fuzzy system is define as

$$f(x) = \ominus^T \varepsilon(x) \quad \textbf{(89)}$$

Where

$$\ominus^T = (\theta_1,\dots,\theta_m)^T = \begin{bmatrix} \theta_1^1, \theta_1^2, \dots, \theta_1^M \\ \theta_2^1, \theta_2^2, \dots, \theta_2^M \\ \vdots \\ \theta_m^1, \theta_m^2, \dots, \theta_m^M \end{bmatrix} \quad \textbf{(90)}$$

$\varepsilon(x) = (\varepsilon^1(x),\dots,\varepsilon^M(x))^T$, $\varepsilon^1(x) = \prod_{i=1}^{n} \mu_{A_i^l}(x_i) / \sum_{l=1}^{M}(\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$, and $\mu_{A_i^l}(x_i)$ is defined in (82). To reduce the number of fuzzy rules, we divide the fuzzy system in to three parts:

$$F^1(q,\dot{q}) = \ominus^{1^T} \varepsilon(q,\dot{q}) = \left[\theta_1^{1^T} \varepsilon(q,\dot{q}), \dots, \theta_m^{1^T} \varepsilon(q,\dot{q})\right]^T \quad \textbf{(91)}$$

$$F^2(q,\ddot{q}_r) = \ominus^{2^T} \varepsilon(q,\ddot{q}_r) = \left[\theta_1^{2^T} \varepsilon(q,\ddot{q}_r), \dots, \theta_m^{2^T} \varepsilon(q,\ddot{q}_r)\right]^T \quad \textbf{(92)}$$

$$F^3(q,\ddot{q}) = \ominus^{3^T} \varepsilon(q,\ddot{q}) = \left[\theta_1^{3^T} \varepsilon(q,\dot{q}), \dots, \theta_m^{3^T} \varepsilon(q,\ddot{q})\right]^T \quad \textbf{(93)}$$

The control input is given by

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r + G^{\wedge} + F^1(q,\dot{q}) + F^2(q,\ddot{q}_r) + F^3(q,\ddot{q}) - K_D s - W sgn(s) \quad \textbf{(94)}$$

Where $M^{\wedge}$, $C_1^{\wedge}$ are the estimations of $M(q)$ and $C_1(q,\dot{q})$; $K_D = diag\,[K_{D1},\dots,K_{Dm}]$ and $K_{D1},\dots,K_{Dm}$ are positive constants; $W = diag\,[W_1,\dots,W_m]$ and $W_1,\dots,W_m$ are positive constants. The adaptation law is given by

$$\begin{aligned} \dot{\theta}_j^1 &= -\Gamma_{1j} s_j \varepsilon(q,\dot{q}) \\ \dot{\theta}_j^2 &= -\Gamma_{2j} s_j \varepsilon(q,\ddot{q}_r) \\ \dot{\theta}_j^3 &= -\Gamma_{3j} s_j \varepsilon(q,\ddot{q}) \end{aligned} \quad \textbf{(95)}$$

Where $j = 1,\dots,m$ and $\Gamma_{1j} - \Gamma_{3j}$ are positive diagonal matrices.
The Lyapunov function candidate is presented as

$$V = \frac{1}{2}s^T M s + \frac{1}{2}\sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\emptyset_j^{1^T}\emptyset_j^1 + \frac{1}{2}\sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\emptyset_j^{2^T}\emptyset_j^2 + \frac{1}{2}\sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\emptyset_j^{13^T}\emptyset_j^3 \quad \textbf{(96)}$$

Where $\emptyset_j^1 = \emptyset_j^{1^*} - \emptyset_j^1$, $\emptyset_j^2 = \emptyset_j^{2^*} - \emptyset_j^2$ and $\emptyset_j^3 = \emptyset_j^{3^*} - \emptyset_j^3$ we define

$$F(q,\dot{q},\ddot{q}_r,\ddot{q}) = F^1(q,\dot{q}) + F^2(q,\ddot{q}_r) + F^3(q,\ddot{q}) \quad \textbf{(97)}$$

From (83) and (82), we get

$$M(q)\ddot{q} + C_1(q,\dot{q})\dot{q} + G(q) = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r + G^{\wedge} + F(q,\dot{q},\ddot{q}_r,\ddot{q}) - K_D s - W sgn(s) \quad \textbf{(98)}$$

Since $\dot{q}_r = \dot{q} - s$ and $\ddot{q}_r = \ddot{q} - \dot{s}$, we get

$$M\dot{s} + (C_1 + K_D)s + W sgn(s) = -\Delta F + F(q,\dot{q},\ddot{q}_r,\ddot{q}) \quad \textbf{(99)}$$

Then $M\dot{s} + C_1 s$ can be written as

$$M\dot{s} + C_1 s = -\Delta F + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) - K_D s - W sgn(s) \quad \textbf{(100)}$$

Where $\Delta F = \tilde{M}\ddot{q}_r + \tilde{C}_1\dot{q}_r + \tilde{G}$,$\tilde{M} = M - M^{\wedge}$,$\tilde{C}_1 = C_1 - \tilde{C}_1$ *and* $\tilde{G} = G - G^{\wedge}$.
The derivative of V is

$$\dot{V} = s^T M\dot{s} + \frac{1}{2}s^T \dot{M}s + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\phi_j^{1^T}\dot{\phi}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\phi_j^{2^T}\dot{\phi}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\phi_j^{13^T}\dot{\phi}_j^3 \quad \textbf{(101)}$$

We know that $s^T M\dot{s} + \frac{1}{2}s^T\dot{M}s = s^T(M\dot{s} + C_1 s)$ from (100). Then

$$\dot{V} = -s^T[-K_D s + W sgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\phi_j^{1^T}\dot{\phi}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\phi_j^{2^T}\dot{\phi}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\phi \quad \textbf{(102)}$$

We define the minimum approximation error as

$$\omega = \Delta F - [F^1(q, \dot{q} | \ominus^{1^*}) + F^2(q, \ddot{q}_r | \ominus^{2^*}) + F^3(q, \ddot{q} | \ominus^{3^*})] \quad \textbf{(103)}$$

We plug (103) in to (102)

$$\dot{V} = -s^T[-K_D s + W sgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\phi_j^{1^T}\dot{\phi}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\phi_j^{2^T}\dot{\phi}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\phi_j^{13^T}\dot{\phi}_j^3$$

$$= -s^T[-K_D s + W sgn(s) + \omega + F^1(q, \dot{q} | \ominus^{1^*}) + F^2(q, \ddot{q}_r | \ominus^{2^*}) + F^3(q, \ddot{q} | \ominus^{3^*}) - F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q})] + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\phi_j^{1^T}\dot{\phi}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\phi_j^{2^T}\dot{\phi}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\phi_j^{13^T}\dot{\phi}_j^3$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m} s_j \phi_j^{1^T}\varepsilon(q, \dot{q}) - \sum_{j=1}^{m} s_j \phi_j^{2^T}\varepsilon(q, \ddot{q}_r) - \sum_{j=1}^{m} s_j \phi_j^{3^T}\varepsilon(q, \ddot{q}) + \sum_{j=1}^{m}\frac{1}{\Gamma_{1j}}\phi_j^{1^T}\dot{\phi}_j^1 + \sum_{j=1}^{m}\frac{1}{\Gamma_{2j}}\phi_j^{2^T}\dot{\phi}_j^2 + \sum_{j=1}^{m}\frac{1}{\Gamma_{3j}}\phi_j^{3^T}\dot{\phi}_j^3$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m}\phi_j^{1^T}(s_j\varepsilon(q, \dot{q}) - \frac{1}{\Gamma_{1j}}\dot{\phi}_j^1) - \sum_{j=1}^{m}\phi_j^{2^T}(s_j\varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{2j}}\dot{\phi}_j^2) - \sum_{j=1}^{m}\phi_j^{3^T}(s_j\varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{3j}}\dot{\phi}_j^3)$$

$$= -s^T K_D s - s^T W sgn(s) - s^T\omega - \sum_{j=1}^{m}\phi_j^{1^T}(s_j\varepsilon(q, \dot{q}) + \frac{1}{\Gamma_{1j}}\dot{\phi}_j^1) - \sum_{j=1}^{m}\phi_j^{2^T}(s_j\varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{2j}}\dot{\phi}_j^2) - \sum_{j=1}^{m}\phi_j^{3^T}(s_j\varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{3j}}\dot{\phi}_j^3)$$

This section focuses on, self tuning gain updating factor for sliding function in SMC, namely, sliding surface slope (λ). The block diagram for this method is shown in Figure 2. In this controller the actual sliding surface gain (λ) is obtained by multiplying the sliding surface with gain updating factor (α). The gain updating factor (α) is calculated on-line by fuzzy dynamic model independent which has sliding surface (S) as its inputs. The gain updating factor is independent of any dynamic model of robotic manipulator parameters. It is a basic fact that the system performance in SMC is sensitive to gain updating factor, λ. Thus, determination of an optimum λ value for a system is an important problem. If the system parameters are unknown or uncertain, the problem becomes more highlighted. This problem is solved by adjusting the sliding function of the sliding mode controller continuously in real-time. In this way, the performance of the overall system is improved with respect to the classical sliding mode controller. Gain tuning-SMC has strong resistance and solves the uncertainty problems. In this controller the actual sliding function (λ_{new}) is obtained by multiplying the old sliding function (λ_{old}) with the output of supervisory mathematical free model controller(α).

$$\lambda^{new} = \lambda^{old} \times \alpha \quad \textbf{(104)}$$

Tuning FPGA based SMC method can tune automatically the scale parameters using new method. To keep the structure of the controller as simple as possible and to avoid heavy computation, a mathematical supervisor tuner is selected [13-14]. In this method the tuneable controller tunes the input scaling factors using gain updating factors. In this method the sliding function, λ, is updated by a new coefficient factor, α, Where α is a function of system error. Figure 3 is shown the proposed method.

$$\alpha = e^2 - \frac{(r_v - r_{vmin})^2}{1+|e|} + r_{vmin} \tag{105}$$

$$r_v = \frac{\dot{e}(t) - e^{\cdot}(t-1)}{\dot{e}(0)} \qquad if\ e^{\cdot}(0) = \begin{cases} \dot{e}(t) & if\ \ e^{\cdot}(t) \geq \dot{e}(t-1) \\ \dot{e}(t-1) & if\ \ e^{\cdot}(t) < \dot{e}(t-1) \end{cases} \tag{106}$$

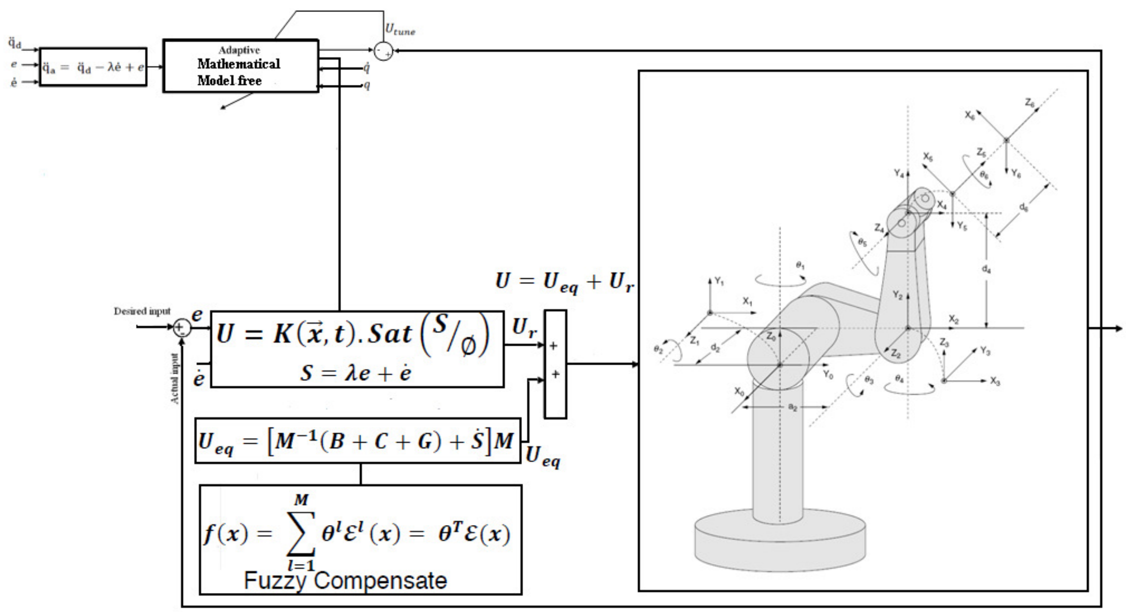


FIGURE 3: Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm

FPGA supports thousands of gates, it is a high operational speed, accurate in response, low cost, short time to market and small size device, research on FPGA is considerably growing as the application of nonlinear (e.g., robotic) systems. The block diagram and part of VHDL code of the FPGA-based sliding mode control systems for a robot manipulator is shown in Figure 4.

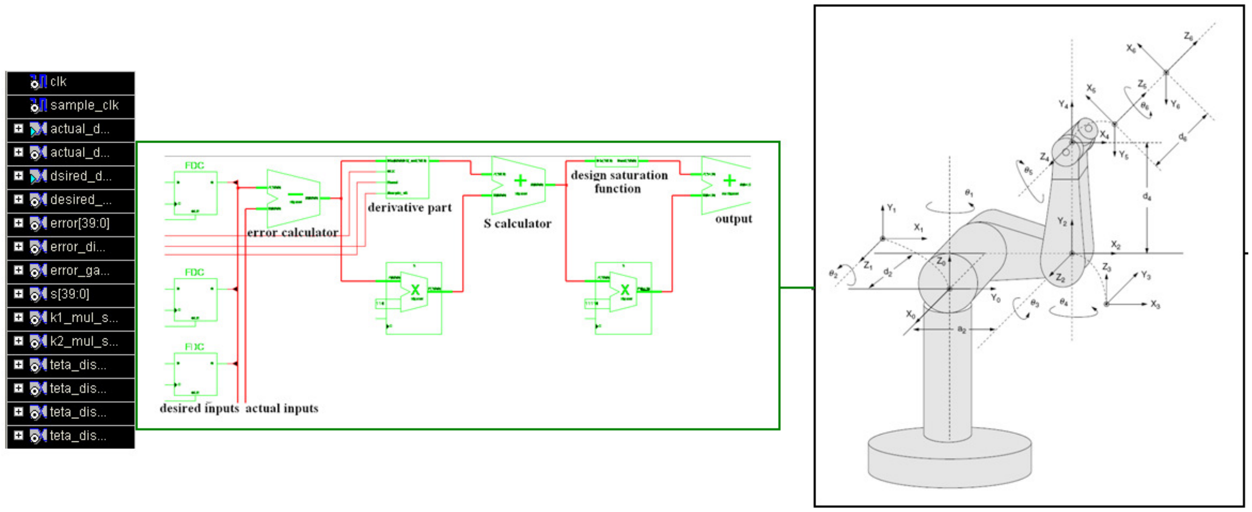


FIGURE 4: FPGA-based Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm

FPGAs Xilinx Spartan 3E families are one of the most powerful flexible Hardware Language Description (HDL) programmable IC's. The last part is focused on the design FPGA based sliding mode controller in Xilinx ISE 9.1. As a result the number of fundamental programmable functional element are used in the XA3S1600E FPGA equal: the LUT's (610 out of 29504), CLB (77 out of 3688), Slice (305 out of 14752), Multipliers (27 out of 36), registers (397), Block RAM memory (648 K) and as a Map report Peak memory usage is 175 MB.

4. RESULTS AND DISCUSSION

Sliding mode controller (SMC) and adaptive MIMO fuzzy compensate SMC and FPGA-based adaptive MIMO fuzzy compensate were tested to Step response trajectory. In this simulation the first, second, and third joints are moved from home to final position without and with external disturbance. The simulation was implemented in Matlab/Simulink and Xilinx-ISE 9.1 environments. It is noted that, these systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude which the sample time is equal to 0.1. This type of noise is used to external disturbance in continuous and hybrid systems.

Tracking performances: Figure 5 is shown the tracking performance in SMC and adaptive MIMO fuzzy compensate SMC without disturbance for Step trajectories. The best possible coefficients in Step SMC are; $K_p = K_v = K_i = 30,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 3, \lambda_2 = 6, \lambda_3 = 6$. From the simulation for first, second, and third links, different controller gains have the different result. Tuning parameters of SMC and adaptive MIMO fuzzy compensate SMC for this type trajectories in PUMA robot manipulator are shown in Table 3.

TABLE 3: Tuning parameters of Step SMC

	λ_1	k_1	ϕ_1	λ_2	k_2	ϕ_2	λ_3	k_3	ϕ_3	SS error $_1$	SS error $_2$	SS error $_3$	RMS error
data1	3	30	0.1	6	30	0.1	6	30	0.1	0	0	-5.3e-15	0
data2	30	30	0.1	60	30	0.1	60	30	0.1	-5.17	14.27	-1.142	0.05
data3	3	300	0.1	6	300	0.1	6	300	0.1	2.28	0.97	0.076	0.08

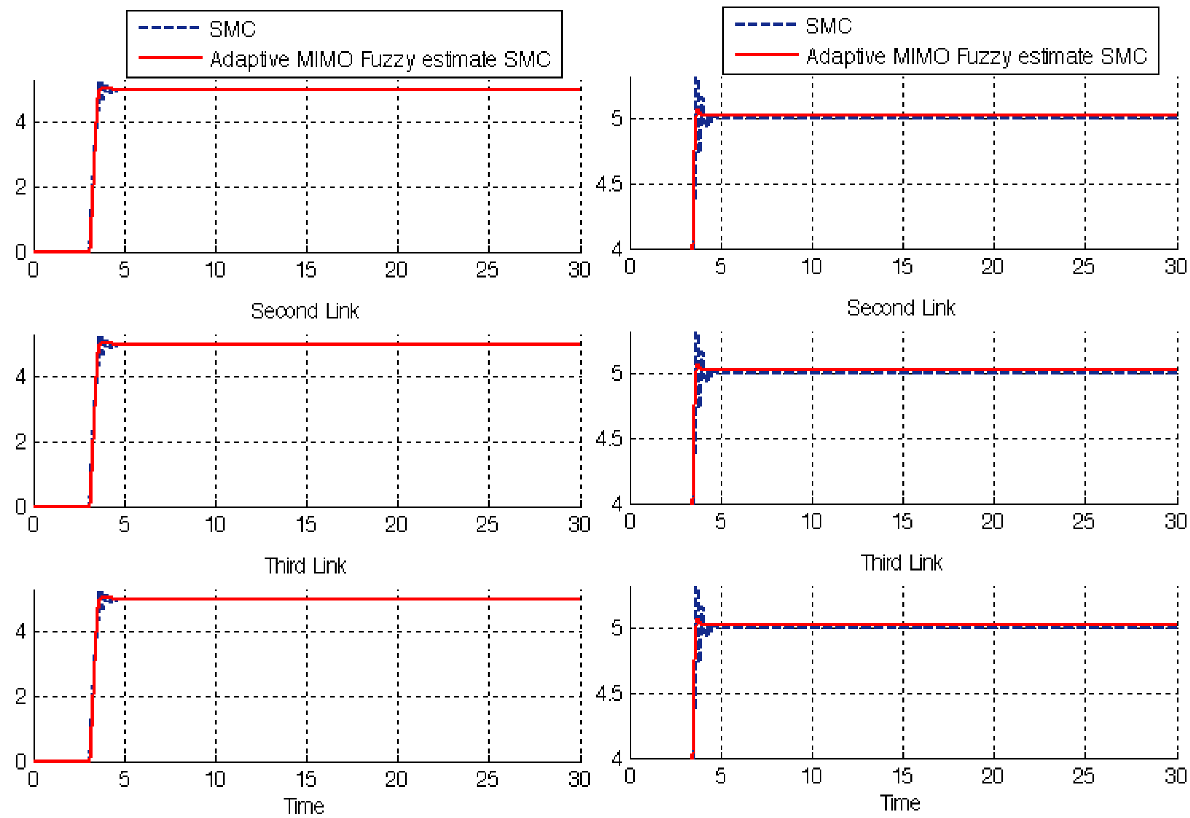


FIGURE 5: SMC Vs. Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: Trajectory performance.

Figure 6 is shown the tracking performance in FPGA-based adaptive MIMO fuzzy compensate SMC without disturbance for Step trajectory.

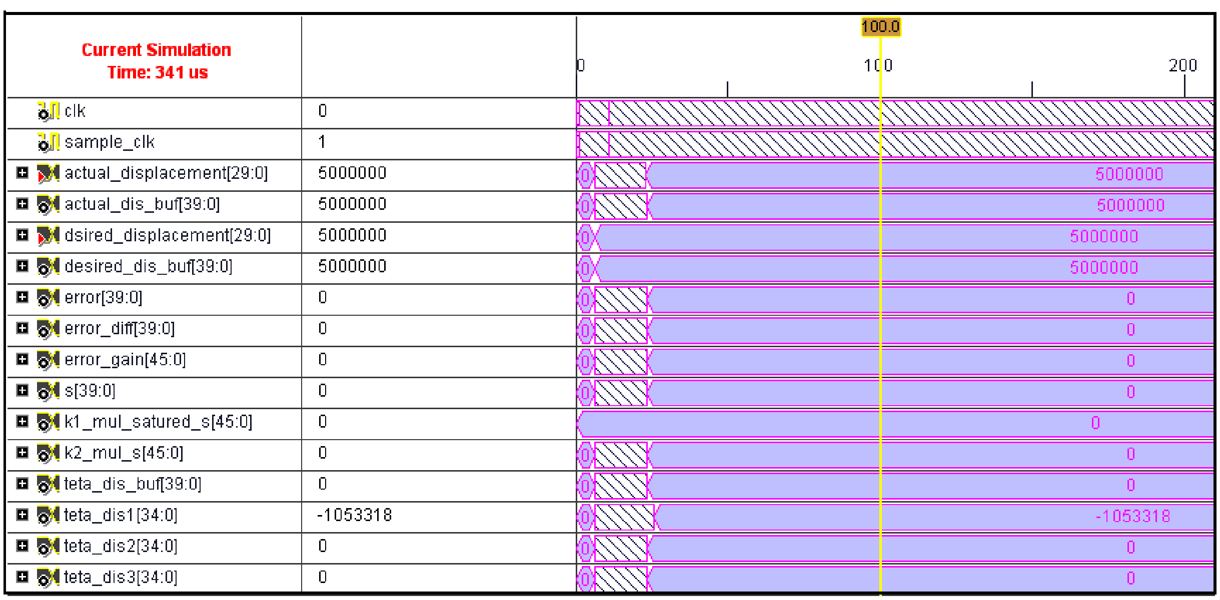


FIGURE 6: FPGA based Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: Trajectory performance.

As mentioned above graphs (Figure 5 and 6), in certain (structured and unstructured) environment it is shown that both of sliding mode controller and adaptive MIMO fuzzy compensate sliding mode algorithm removed the chattering because these controller are used linear boundary layer saturation method.

Disturbance Rejection: Figure 7 is indicated the power disturbance removal in SMC and adaptive MIMO fuzzy compensate sliding mode algorithm. As mentioned before, SMC is one of the most important robust nonlinear controllers. Besides a band limited white noise with predefined of 40% the power of input signal is applied to the step SMC and adaptive MIMO fuzzy compensate sliding mode algorithm; it found slight oscillations in SMC trajectory responses. As a result, by comparing SMC and adaptive MIMO fuzzy compensate sliding mode algorithm, it found that adaptive MIMO fuzzy compensate sliding mode algorithm is more robust than SMC with regards to the same external disturbance.

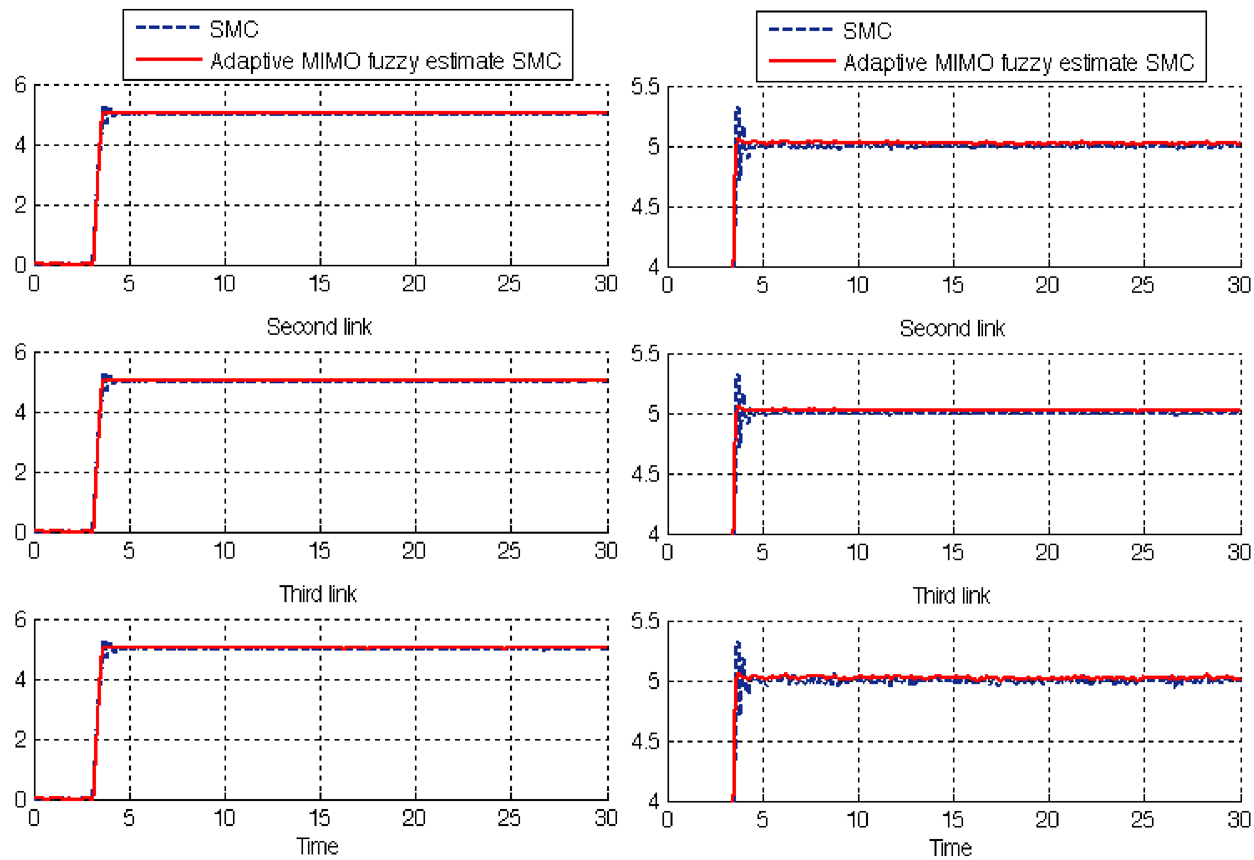


FIGURE 7: SMC Vs. Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: Disturbance rejection.

Figure 8 is shown the tracking performance in FPGA-based adaptive MIMO fuzzy compensate SMC with external disturbance for Step trajectory.

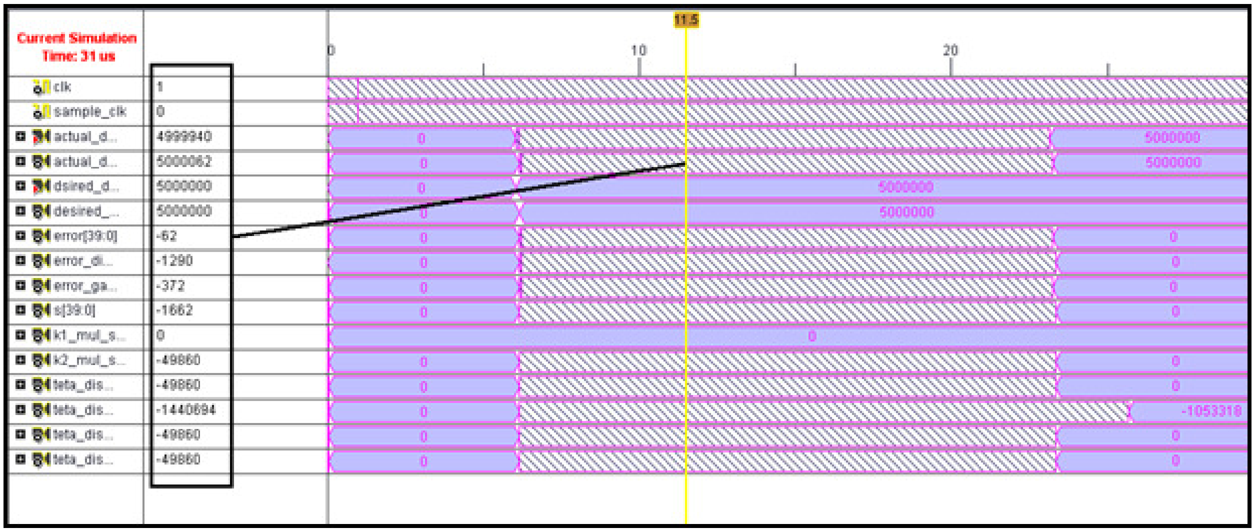


FIGURE 8: FPGA based Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: Disturbance rejection

Chattering Phenomenon: Figure 9 has presented the power of adaptive MIMO fuzzy compensate SMC. These figures have illustrated the power chattering elimination in SMC as well as in adaptive MIMO fuzzy compensate SMC, with external disturbance. By comparing these controllers, conversely SMC has slight fluctuations; adaptive MIMO fuzzy compensate SMC is steadily stabilized. As a result, with respect to the external disturbance adaptive MIMO fuzzy compensate SMC has an acceptable performance.

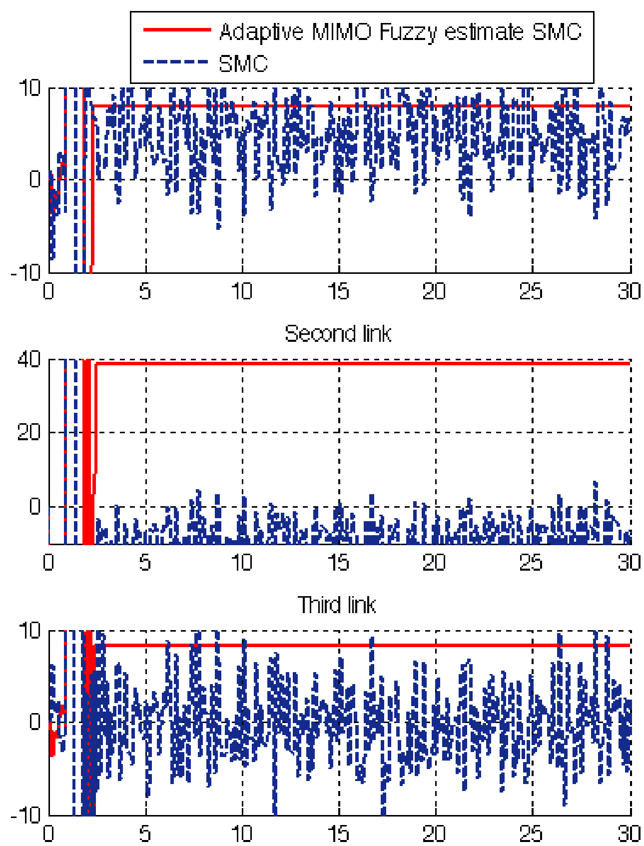


FIGURE 9: SMC Vs. Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: Chattering.

Figure 10 is shown the chattering phenomenon in FPGA-based adaptive MIMO fuzzy compensate SMC with external disturbance for Step trajectory.

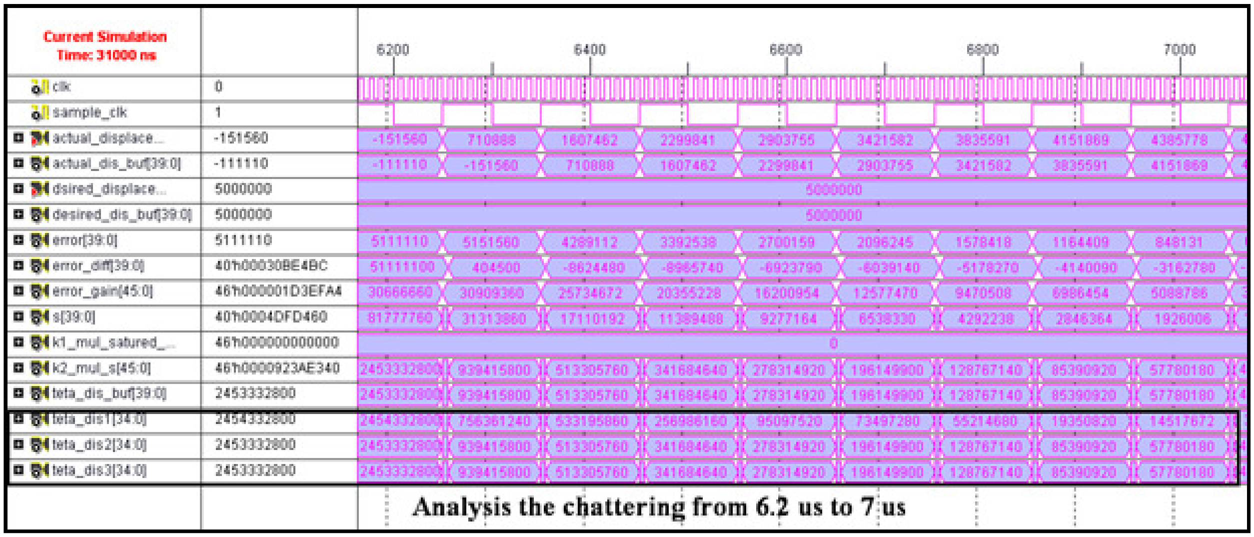


FIGURE 10: FPGA based Adaptive MIMO Fuzzy Compensate Sliding Mode Algorithm: chattering rejection

5. CONCLUSION AND EXTENSION

Refer to the research, a position FPGA-based mathematical model free adaptive fuzzy estimator sliding mode control Lyapunov based design and application to PUMA robot manipulator has proposed in order to design high performance robust and stable FPGA based nonlinear controller in the presence of structure and unstructured uncertainties. The stability of the closed-loop system is proved mathematically based on the Lyapunov method. The first objective in proposed method is removed the chattering which linear boundary layer method is used to solve this challenge. The second target in this work is compensate the model uncertainty by MIMO fuzzy inference system, in the case of the m-link robotic manipulator, if we define k_1 membership functions for each input variable, the number of fuzzy rules applied for each joint is K_1 which will result in a low computational load. The third target in this research is applied mathematical model free to MIMO fuzzy estimator sliding mode algorithm and eliminate the chattering with minimum computational load and the final main goal is design FPGA based proposed methodology which in this case the performance is improved by using the advantages of sliding mode algorithm, artificial intelligence compensate method, adaptive algorithm and FPGA while the disadvantages removed by added each method to previous method. Fuzzy logic method by adding to the sliding mode controller has covered negative points in fuzzy and sliding algorithms. Higher implementation speed and small chip size versus an acceptable performance is reached by designing FPGA-based sliding mode controller. This implementation considerably reduces the chattering phenomenon and error in the presence of certainties. The controller works with a maximum clock frequency of 63.29 MHz and the computation time (delay in activation) of this controller is 0.1μs. As a result, this controller will be able to control a wide range of robot manipulators with a high sampling rates because its small size versus high speed markets.

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] 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.

[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] 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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

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

[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] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] N. Sulaiman*, et al.*, "Design and Implementation of FPGA-Based Systems-A Review," *Australian Journal of Basic and Applied Sciences,* vol. 3, pp. 3575-3596, 2009.

[64] X. Shao and D. Sun, "Development of an FPGA-based motion control ASIC for robotic manipulators," 2006, pp. 8221-8225.

[65] Y. S. Kung*, et al.*, "FPGA-implementation of inverse kinematics and servo controller for robot manipulator," *Proc. IEEE Int. on Robotics and Biomimetics,* pp. 1163–1168, 2006.

[66] X. Shao*, et al.*, "A new motion control hardware architecture with FPGA-based IC design for robotic manipulators," 2006, pp. 3520-3525.

[67] Y. S. Kung*, et al.*, "Design and Implementation of a Servo System for Robotic Manipulator," ed: CACS, 2005.

[68] U. D. Meshram and R. Harkare, "FPGA Based Five Axis Robot Arm Controller."

[69] U. Meshram*, et al.*, "Robot arm controller using FPGA," 2009, pp. 8-11.

[70] Y. S. Kung and G. S. Shu, "Development of a FPGA-based motion control IC for robot arm," 2006, pp. 1397-1402.

[71] Z. A. Obaid*, et al.*, "Developed Method of FPGA-based Fuzzy Logic Controller Design with the Aid of Conventional PID Algorithm," *Australian Journal of Basic and Applied Sciences,* vol. 3, pp. 2724-2740, 2009.

[72] S. T. Karris, *Digital circuit analysis and design with Simulink modeling and introduction to CPLDs and FPGAs*: Orchard Pubns, 2007.

[73] K. D. Rogers, "ACCELERATION AND IMPLEMENTION OF A DSP PHASE-BASED FREQUENCY ESTIMATION ALGORITHM: MATLAB/SIMULINK TO FPGA VIA XILINX SYSTEM GENERATOR," Citeseer, 2004.

[74] F. J. Lin*, et al.*, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive," 2005, pp. 1137-1148.
[75] R. R. Ramos*, et al.*, "A fixed-frequency quasi-sliding control algorithm: application to power inverters design by means of FPGA implementation," *Power Electronics, IEEE Transactions on,* vol. 18, pp. 344-355, 2003.
[76] F. J. Lin*, et al.*, "FPGA-based adaptive backstepping sliding-mode control for linear induction motor drive," *Power Electronics, IEEE Transactions on,* vol. 22, pp. 1222-1231, 2007.
[77] S. Lentijo*, et al.*, "FPGA based sliding mode control for high frequency power converters," 2004, pp. 3588-3592.
[78] B. Armstrong*, et al.*, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," 2002, pp. 510-518.
[79] P. I. Corke and B. Armstrong-Helouvry, "A search for consensus among model parameters reported for the PUMA 560 robot," 2002, pp. 1608-1613.

Simultaneous State and Actuator Fault Estimation With Fuzzy Descriptor PMID and PD Observers for Satellite Control Systems

Rajab Challoo *kfrc000@tamuk.edu*
Dept. of Electrical Engineering &Computer Science
Texas A&M University-Kingsville
Kingsville, 78363-8202, USA

Sunny Dubey *sunny.dubey@students.tamuk.edu*
Dept. of Electrical Engineering &Computer Science
Texas A&M University-Kingsville
Kingsville, 78363-8202, USA

Abstract

In this paper, Takagi-Sugeno (T-S) fuzzy descriptor proportional multiple-integral derivative (PMID) and proportional derivative (PD) observer methods that can estimate the system states and actuator faults simultaneously are proposed. T-S fuzzy model is obtained by linearsing satellite/spacecraft attitude dynamics at suitable operating points. For fault estimation, actuator fault is introduced as state vector to develop augmented descriptor system and robust fuzzy PMID and PD observers are developed. Stability analysis is performed using Lyapunov direct method. The convergence conditions of state estimation error are formulated in the form of LMI (linear matrix inequality). Derivative gain, obtained using singular value decomposition of descriptor state matrix (E), gives more design degrees of freedom together with proportional and integral gains obtained from LMI. Simulation study is performed for our proposed methods.

Keywords: Fault, Descriptor Systems, Estimation, Fuzzy Model, Observers, Robustness, Linear Matrix Inequality, Quadratic Stability.

1. INTRODUCTION

A fault is termed as an unexpected change in the system's behavior that deteriorates the normal functioning of the system. The process of estimating the magnitude of the fault occurring in the system is coined as fault estimation [3]. In order to accomplish successful space missions the safety of satellite/spacecraft attitude control systems is crucial. Actuators and sensors are essential components of satellite control systems. If they get faulty, fault diagnosis must be carried out in order to avert the danger involved in space missions. Using sliding mode observers [6, 7] and adaptive observers [9], fault diagnosis is carried out extensively.

Integral actions are helpful to achieve steady-state accuracy in control systems. The design of proportional-integral (PI) observers were established [10] with the introduction of integral action in observer design. Till now, such observers have attracted many researchers.

The problem of constructing the observers for descriptor linear systems has been studied by many researchers parallel to the standard linear control systems. The design of full-order observers and reduced-order observers for descriptor linear systems can be found in fault diagnosis literature.

In real sense of words, the satellite attitude dynamics show non-linearity. So, Takagi-Sugeno (T-S) fuzzy model [11] can be used to linearise the satellite attitude dynamics at suitable operating points. [4] introduced fuzzy proportional multiple-integral observer method for robust actuator fault estimation. The idea of generalized proportional integral derivative (GPID) and proportional multiple-integral derivative (PMID) observers is proposed by [1, 2].

In this paper, we propose fuzzy PMID and PD observer based methods for robust actuator fault estimation. In our design, derivative gain gives more design degrees of freedom as compared to fuzzy PMI observers. The design constraints in our *fuzzy* PMID method are not strict for observer gains as compared to PMID or PID observers mentioned above.

In Section 2, we formulate the problem for actuator fault estimation of satellite control systems. The T-S fuzzy PMID & PD descriptor observers are designed in Section 3 and Section 4 respectively. Simulation studies are performed in order to validate the proposed methods in Section 5.

2. PROBLEM FORMULATION

2.1 ATTITUDE DYNAMICS

The equation of rotational motion of rigid satellite/spacecraft body is:

$$M = J\frac{\partial \omega}{\partial t} + \omega \times (J\omega + h_w) \tag{1}$$

$$M = J\frac{\partial \omega}{\partial t} + S(\omega)(J\omega + h_w) \tag{2}$$

In (2) we define as follows:

$$\frac{\partial \omega}{\partial t} = \begin{Bmatrix} \frac{d\omega_x}{dt} \\ \frac{d\omega_y}{dt} \\ \frac{d\omega_z}{dt} \end{Bmatrix} = \begin{Bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{Bmatrix}, \quad S(\omega) = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \text{ and } M = T_u + T_d .$$

Therefore,

$$J\dot{\omega} + S(\omega)(J\omega + h_w) = T_u + T_d \tag{3}$$

where inertia matrix of satellite is J , ω is angular velocity with respect to inertial frame, T_u is output torque of the flywheels, T_d is the disturbance from environment and h_w is the angular momentum of three flywheels [4].

With ω as the state variable and actuator fault f_a , the state space system for (3) is represented as:

$$\begin{aligned} \dot{x} &= -J^{-1}S(\omega)(J\omega + h_w) + J^{-1}T_u + J^{-1}T_d + J^{-1}f_a, \\ y &= Cx. \end{aligned} \tag{4}$$

Now (4) can be written as [4]:

$$\begin{aligned} \dot{x} &= f(x) + Bu + Dd + Ff_a, \\ y &= Cx \end{aligned} \tag{5}$$

where $f(x) = -J^{-1}S(\omega)(J\omega + h_w), B = D = F = J^{-1}$ and $C = I_{3\times3}$

2.2 T-S Fuzzy Model

The T-S fuzzy model consists of an *if-then* rule base. The antecedent of each rule [8] comprises of scheduling variables and fuzzy sets. The consequent of each rule is a simple functional expression.

The *i*-th rule is described as

Model rule *i*:
If z_1 is Z_1^i and … and z_p is Z_p^i then $y = F_i(z)$

where the vector *z* has *p* components, z_j *j* =1, 2… *p,* and stands for the vector of scheduling variables and their values determine the degree to which rules are active. The sets, Z_j^i, *j*=1, 2, … , *p; i* = 1, 2, …, *m,* where *m* is the number of the rules, are the antecedents fuzzy sets. The values of a scheduling variable z_j belong to a fuzzy set Z_j^i with a truth value given by the membership function $\lambda_{ij} : \mathbb{R} \to [0, 1]$. The truth value for an entire rule is determined based on the individual premise variables, using a conjunction operator as:

$$\varphi_i(z) = \prod_{j=1}^{p} \lambda_{ij}(z_j) \tag{6}$$

The fuzzy weights are determined as

$$w_i(z) = \varphi_i(z) \Big/ \sum_{j=1}^{m} \varphi_j(z)$$

$$w_i(z) \geq 0 \text{ and } \sum_i^p w_i(z) = 1 \tag{7}$$

The T-S fuzzy system for (5) can be written as:

$$\dot{x} = \sum_{i=1}^{p} w_i(z)(A_i x + B_i u + Dd + F_i f_a),$$
$$y = \sum_{i=1}^{p} w_i(z) C_i x, \tag{8}$$

where $F_i \in R^{n \times k}$.

Each linearised model for satellite can be obtained as [4]:

$$\dot{x} = A(\omega_0)x + Bu + Dd + Ff_a,$$
$$y = Cx. \tag{9}$$

where $A(\omega_0) = -J^{-1}S(\omega_0)J, B = D = F = J^{-1}$, $C = I_{3\times3}$ and ω_0 is the operating point.

So we [4] have

$$\dot{x} = \sum_{i=1}^{p} w_i(z)(A_i x + Bu + Dd + Ff_a)$$
$$y = \sum_{i=1}^{p} w_i(z) C_i x \tag{10}$$

3. T-S FUZZY DESCRIPTOR PMID OBSERVER

Consider the following fuzzy descriptor system

$$E\dot{x} = \sum_{i=1}^{p} w_i(z)(A_i x + Bu + Dd + Ff_a)$$
$$y = \sum_{i=1}^{p} w_i(z) C_i x. = Cx \tag{11}$$

where $x \in R^n$ is the descriptor state vector, $u \in R^m$ and $y \in R^p$ are, respectively, the control input and output vectors, $f_a \in R^k$ is unknown actuator fault. The matrix *E* may be singular.

The q-th derivative of the fault is assumed to be bounded. The fault considered in this paper allows $q \geq 1$ as the first derivatives of faults with time are bounded.

Consider the following system with proportional, multiple integral and derivative weights of the output estimation error

$$\begin{aligned} E\dot{\hat{x}} &= \sum_{i=1}^{p} w_i(z)\left(A_i\hat{x} + Bu + L_p(y - C\hat{x}) + L_d(\dot{y} - C\dot{\hat{x}})\right) + F\hat{f}_a, \\ \hat{y} &= C\hat{x}, \\ \dot{\hat{y}} &= C\dot{\hat{x}}, \end{aligned} \tag{12}$$

$$\begin{aligned} \dot{\hat{f}}_a^{\,q} &= \sum_{i=1}^{p} w_i(z) L_i^q (y - C\hat{x}) + \hat{f}_a^{\,q-1} \\ \dot{\hat{f}}_a^{\,q-1} &= \sum_{i=1}^{p} w_i(z) L_i^{q-1} (y - C\hat{x}) + \hat{f}_a^{\,q-2} \\ &\vdots \\ &\vdots \\ &\vdots \\ \dot{\hat{f}}_a^{\,2} &= \sum_{i=1}^{p} w_i(z) L_i^2 (y - C\hat{x}) + \hat{f}_a^{\,1} \\ \dot{\hat{f}}_a^{\,1} &= \sum_{i=1}^{p} w_i(z) L_i^1 (y - C\hat{x}). \end{aligned} \tag{13}$$

Here, $\hat{x} \in R^n$ is an estimation of the descriptor state vector x, and $\hat{f}_a^{\,i} \in R^k \, (i = 1,2,\ldots,q)$ is an estimation of $(q-i)$-th derivative of the fault; the proportional gain $L_p \in R^{n \times p}$, the derivative gain $L_d \in R^{n \times p}$ and the integral gain $L_i \in R^{k \times p}$ are design matrices.

In order to estimate the actuator fault, we construct an augmented descriptor system as follows:

Let $\xi_i = f^{(q-i)}, (i = 1,2,\ldots,q)$ (14)

Using (12), (13) and (14), we get:

$$\begin{aligned} \overline{E}\dot{\overline{x}} &= \sum_{i=1}^{p} w_i(z)\overline{A}_i\overline{x} + \overline{B}u + \overline{D}d + Gf_a^{\,q}, \\ y &= \overline{C}\overline{x} \end{aligned} \tag{15}$$

where,

$$\overline{x} = \left[x^T, \xi_1^T, \ldots.., \xi_q^T\right]^T, B = \left[B^T, 0,0\ldots,0\right]^T, \overline{D} = \left[D^T, 0,0..0\right]^T$$

$$G = [0, I_k, 0,..,0]^T, \overline{C} = [C,0,0..0],$$

$$\overline{A}_i = \begin{bmatrix} A_i & 0 & \cdots & 0 & F \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 \end{bmatrix} \quad \overline{E} = \begin{bmatrix} E & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 \\ 0 & 0 & \cdots & 0 & I \end{bmatrix}$$

Now we develop observer for augmented system (15) as:

$$\begin{aligned}\dot{\Phi} &= \sum_{i=1}^{p} w_i(z)\left(\overline{A}_i - \overline{L}_p\overline{C}\right)\hat{\overline{x}} + \overline{B}u + \overline{L}_p y,\\ \hat{\overline{x}} &= \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\left(\Phi + \overline{L}_d y\right),\\ \dot{\hat{\overline{x}}} &= \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\left(\dot{\Phi} + \overline{L}_d \dot{y}\right),\\ &= \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\cdot\left(\sum_{i=1}^{p} w_i(z)\left(\left(\overline{A}_i - \overline{L}_p\overline{C}\right)\hat{\overline{x}} + \overline{B}u + \overline{L}_p y + \overline{L}_d \dot{y}\right)\right)\end{aligned} \tag{16}$$

where,

$$\overline{L}_p = \left[L_p^T, \left(L_i^1\right)^T, \ldots \left(L_i^q\right)^T\right]^T,$$

$$\overline{L}_d = \left[L_d^T, 0, 0 \ldots 0\right]^T$$

Add $\overline{L}_d\,\dot{y}$ to the both sides of (15), and then we have

$$\dot{\overline{x}} = \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\cdot\left(\sum_{i=1}^{p} w_i(z)\left(\left(\overline{A}_i - \overline{L}_p\overline{C}\right)\overline{x} + \overline{B}u + \overline{D}d + Gf_a^q + \overline{L}_p y + \overline{L}_d \dot{y}\right)\right) \tag{17}$$

The dynamic state error equation is:

$$\begin{aligned}\dot{\overline{e}} &= \dot{\hat{\overline{x}}} - \dot{\overline{x}}\\ &= \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\left(\sum_{i=1}^{p} w_i(z)\left(\left(\overline{A}_i - \overline{L}_p\overline{C}\right)\overline{e} - \overline{D}d - Gf_a^q\right)\right)\\ &= \left(\overline{E} + \overline{L}_d\overline{C}\right)^{-1}\left(\sum_{i=1}^{p} w_i(z)\left(\left(\overline{A}_i - \overline{L}_p\overline{C}\right)\overline{e} - M\overline{d}\right)\right)\end{aligned} \tag{18}$$

where $M\overline{d} = \begin{bmatrix}\overline{D} & G\end{bmatrix}\begin{bmatrix} d \\ f_a^q\end{bmatrix}$

Condition 1: The trio (E, A_i, C) is completely observable if

$$rank\begin{bmatrix} E \\ C\end{bmatrix} = n, \quad rank\begin{bmatrix} sE - A_i \\ C\end{bmatrix} = n \tag{19}$$

Condition 2: $rank\begin{bmatrix} A_i & F \\ C & 0\end{bmatrix} = n + k$ (20)

Theorem 1: If conditions 1 and 2 are satisfied, there exists a robust fuzzy observer in the form of (16) for the plant (15), which can make the steady estimator error dynamics as small as any desired accuracy. The derivative gain is such that $S = \overline{E} + \overline{L}_d\overline{C}$ is non-singular and the gain $\overline{L}_p$ is solved from the following linear matrix inequalities if there exists a common positive definite matrix $P \in R^{(n+k)\times(n+k)}$ and matrix Y_i such that

$$\begin{bmatrix}\overline{A}_i S^{-T} P + PS^{-1}\overline{A}_i - \overline{C}^T Y_i^T - Y_i\overline{C} + I & -PS^{-1}M \\ -M^T S^{-T} P & -\gamma^2 I\end{bmatrix} < 0 \tag{21}$$

with $\gamma > 0$ then $\overline{L}_p = SP^{-1}Y_i$

Proof: Under the conditions 1 and 2, the trio $\left(\overline{E}, \overline{A}_i, \overline{C}\right)$ is completely observable. Then derivative gain must be chosen such that S is non-singular.

For $rank(E) = m,$ singular value decomposition of descriptor state matrix *E* gives two orthogonal matrices Γ and Ξ such that

$$E = \Gamma \begin{bmatrix} \Delta_m & 0 \\ 0 & 0 \end{bmatrix} \Xi^T = \Gamma \begin{bmatrix} I_m & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta_m & 0 \\ 0 & I_{n-m} \end{bmatrix} \Xi^T$$

where $\Delta_m = diag(\Psi_1, \Psi_2, ..., \Psi_m)$ with $\Psi_k > 0, k = 1,2,....m.$

Let

$$\Theta = \Xi \begin{bmatrix} \Delta_m^{-1} & 0 \\ 0 & I_{n-m} \end{bmatrix}$$

Then, we get

$$\Gamma^T E\, \Theta = \begin{bmatrix} I_m & 0 \\ 0 & 0 \end{bmatrix},$$

$$C\,\Theta = \begin{bmatrix} C_1 & C_2 \end{bmatrix},$$

In order that trio (*E, A, C*) is completely observable,

$$rank(C_2) = n - m$$

Thus, one has derivative gain as below [12]:

$$L_d = \Gamma \begin{bmatrix} 0 \\ \gamma_d \left(C_2^T C_2\right)^{-1} C_2^T \end{bmatrix} \tag{22}$$

where γ_d is any positive number.

We can compute as below [12]:

$$E + L_d C = \Gamma \begin{bmatrix} I_m & 0 \\ \gamma_d \left(C_2{}^T C_2\right)^{-1} C_2{}^T C_1 & \gamma_d I_{n-m} \end{bmatrix} \tag{23}$$

which implies that $E + L_d C$ is non-singular. Now by using $\overline{E}, \overline{L}_d$, and $\overline{C}$, we can say that $S = \overline{E} + \overline{L}_d \overline{C}$ is non-singular too.

Stability analysis is performed using Lyapunov direct method. The convergence conditions of state estimation error are formulated in the form of LMI (linear matrix inequality).The proportional and integral gains are determined from obtained LMI (21).

When a system is quadratically stable it implies that it is stable. However, the reverse is not necessarily true. So, the conditions obtained using the Lyapunov function are only sufficient. The unforced T-S model is quadratically stable if the Lyapunov function decreases and tends to zero when time approaches to infinity for all trajectories of error in our case.

The goal of robust actuator fault estimation is to determine proportional and integral gains (together with derivative gains) that cause the asymptotic convergence of error towards zero as time tends to infinity in case of disturbances and perturbations.

Consider the following Lyapunov function candidate,

$$V(\overline{e}) = \overline{e}^T P \overline{e},$$

$$\dot{V}(\bar{e}) = \bar{e}^T P\dot{\bar{e}} + \dot{\bar{e}}^T P\bar{e}$$

$$= \bar{e}^T P\left[\left(\sum_{i=1}^{p} w_i(z) S^{-1}\left(\bar{A}_i - \bar{L}_p\bar{C}\right)\bar{e} - S^{-1}M\bar{d}\right)\right] + \left[\sum_{i=1}^{p} w_i(z)\left(\bar{e}^T\left(\left(\bar{A}_i - \bar{L}_p\bar{C}\right)^T S^{-T}\right)\right) - \bar{d}^T M^T S^{-T}\right] P\bar{e}$$

$$= \sum_{i=1}^{p} w_i(z)\bar{e}^T PS^{-1}\left(\bar{A}_i - \bar{L}_p\bar{C}\right)\bar{e} - \bar{e}^T PS^{-1}M\bar{d} + \sum_{i=1}^{p} w_i(z)\bar{e}^T\left(\bar{A}_i - \bar{L}_p\bar{C}\right)^T S^{-T}P\bar{e} - \bar{d}^T M^T S^{-T}P\bar{e}$$

$$= \sum_{i=1}^{p} w_i(z)\bar{e}^T\left[PS^{-1}\left(\bar{A}_i - \bar{L}_p\bar{C}\right) + \left(\bar{A}_i - \bar{L}_p\bar{C}\right)^T S^{-T}P\right]\bar{e} - 2\bar{e}^T PS^{-1}M\bar{d}$$

Define $\bar{L}_p = SP^{-1}\bar{Y}_i$ and consider the Lyapunov function V such that

$$\dot{V}(\bar{e}) + \bar{e}^T\bar{e} - \gamma^2\bar{d}^T\bar{d} \le 0$$

Thus integrating this expression leads to

$$V(\bar{e}(\infty)) - V(\bar{e}(0)) \le \int_0^\infty \gamma^2\bar{d}^T\bar{d} - \bar{e}^T\bar{e}\,dt.$$

Since T-S model is asymptotically stable and with zero initial conditions, we obtain

$$0 < \int_0^\infty \gamma^2\bar{d}^T\bar{d} - \bar{e}^T\bar{e}\,dt$$

which is equivalent to

$$\int_0^\infty \bar{e}^T\bar{e} < \gamma^2\int_0^\infty \bar{d}^T\bar{d} \text{ Or } \|\bar{e}\|_{T_f} < \gamma\|\bar{d}\|_{T_f} \tag{24}$$

Now, the stability conditions are obtained as in (25) below

$$\begin{aligned}\dot{V}(\bar{e}) + \bar{e}^T\bar{e} - \gamma^2\bar{d}^T\bar{d} &= \sum_{i=1}^{p} w_i(z)\bar{e}^T\left[PS^{-1}\left(\bar{A}_i - \bar{L}_p\bar{C}\right) + \left(\bar{A}_i - \bar{L}_p\bar{C}\right)^T S^{-T}P\right]\bar{e} - 2\bar{e}^T PS^{-1}M\bar{d} + \bar{e}^T\bar{e} - \gamma^2\bar{d}^T\bar{d} \\ &= \sum_{i=1}^{p} w_i(z)\begin{bmatrix}\bar{e}^T & \bar{d}^T\end{bmatrix}\begin{bmatrix}\bar{A}_i S^{-T}P + PS^{-1}\bar{A}_i - \bar{C}^T Y_i^T - Y_i\bar{C} + I & -PS^{-1}M \\ -M^T S^{-T}P & -\gamma^2 I\end{bmatrix}\begin{bmatrix}\bar{e} \\ \bar{d}\end{bmatrix}\end{aligned} \tag{25}$$

4. T-S FUZZY DESCRIPTOR PD OBSERVER

Consider the following fuzzy descriptor system with proportional and derivative weights of the output estimation error

$$\begin{aligned}E\dot{\hat{x}} &= \sum_{i=1}^{p} w_i(z)\left(A_i\hat{x} + Bu + L_p(y - C\hat{x}) + L_d(\dot{y} - C\dot{\hat{x}})\right), \\ \hat{y} &= C\hat{x}, \\ \dot{\hat{y}} &= C\dot{\hat{x}}\end{aligned} \tag{26}$$

where L_p and L_d are respectively the proportional and derivative gain matrices. Derivative gain is determined using (22).

In order to obtain the estimation of actuator fault, we introduce fault as an auxiliary state vector in (11) and get the following augmented system,

$$\begin{aligned}\bar{E}\dot{\bar{x}} &= \sum_{i=1}^{p} w_i(z)\bar{A}_i\bar{x} + \bar{B}u + \bar{D}d, \\ y &= \bar{C}\bar{x}\end{aligned} \tag{27}$$

where $x_a = Ff_{a,}\ \bar{x} = \begin{bmatrix} x \\ x_a \end{bmatrix}, \bar{A}_i = \begin{bmatrix} A_i & I_3 \\ 0 & 0 \end{bmatrix}, \bar{E} = \begin{bmatrix} I_3 & 0 \\ 0 & I_3 \end{bmatrix}, F = I_3, \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \bar{D} = \begin{bmatrix} D & 0 \end{bmatrix}$.

If there exists an observer as (28) for the plant (27), then actuator fault and the states of the system can be estimated simultaneously.

$$\begin{aligned}
\dot{\Phi} &= \sum_{i=1}^{p} w_i(z)(\bar{A}_i - L_p\bar{C})\hat{\bar{x}} + \bar{B}u + L_p y, \\
\hat{\bar{x}} &= (\bar{E} + L_d\bar{C})^{-1}(\Phi + L_d y), \\
\dot{\hat{\bar{x}}} &= (\bar{E} + L_d\bar{C})^{-1}(\dot{\Phi} + L_d \dot{y}), \\
&= (\bar{E} + L_d\bar{C})^{-1} \cdot \left(\sum_{i=1}^{p} w_i(z)((\bar{A}_i - L_p\bar{C})\hat{\bar{x}} + \bar{B}u + L_p y + L_d \dot{y})\right)
\end{aligned} \tag{28}$$

Add $L_d\,\dot{y}$ to the both sides of (27), and then we have

$$\dot{\bar{x}} = (\bar{E} + L_d\bar{C})^{-1} \cdot \left(\sum_{i=1}^{p} w_i(z)((\bar{A}_i - L_p\bar{C})\bar{x} + \bar{B}u + \bar{D}d + L_p y + L_d \dot{y})\right) \tag{29}$$

The dynamic state error equation is:

$$\begin{aligned}
\dot{\bar{e}} &= \dot{\bar{x}} - \dot{\hat{\bar{x}}} \\
&= (\bar{E} + \bar{L}_d\bar{C})^{-1}\left(\sum_{i=1}^{p} w_i(z)((\bar{A}_i - \bar{L}_p\bar{C})\bar{e} + \bar{D}d)\right)
\end{aligned}$$

Theorem 2: If conditions 1 and 2 are satisfied, there exists a robust fuzzy observer in the form of (28) for the plant (27), which can make the steady estimator error dynamics as small as any desired accuracy. The derivative gain is such that $S = \bar{E} + L_d\bar{C}$ is non-singular and the gain L_p is solved from the following linear matrix inequalities if there exists a common positive definite matrix $P \in R^{(n+k)\times(n+k)}$ and matrix Y_i such that

$$\begin{bmatrix} \bar{A}_i S^{-T} P + PS^{-1}\bar{A}_i - \bar{C}^T Y_i^T - Y_i\bar{C} + I & -PS^{-1}\bar{D} \\ -\bar{D}^T S^{-T} P & -\gamma^2 I \end{bmatrix} < 0 \tag{30}$$

with $\gamma > 0$ then $L_p = SP^{-1}Y_i$

Proof: Proof is similar to theorem 1. Derivative gain and proportional gains are obtained in the same fashion as in theorem 1. The only difference is that we have removed multiple integrals.

Remark: As we have introduced the actuator fault as an auxiliary state vector in the plant (11), the matrix $F = I_3$ and not the inverse of inertia matrix *J.* The actuator fault can be directly isolated from estimated state $\hat{\bar{x}}$.

Since $x_a = Ff_{a,}\ \bar{x} = \begin{bmatrix} x \\ x_a \end{bmatrix}$

Therefore,

$\hat{x}_a = [0 \quad I_3]\hat{\bar{x}}$ then $\hat{f}_a = F^{-1}[0 \quad I_3]\hat{\bar{x}}$

5. SIMULATION RESULTS

In order to obtain the T-S fuzzy model, suitable operating points are chosen in the vicinity of zero and we employ triangular membership function in this case. The actuator faults [4] along three axes are

$$f_{ax} = \begin{cases} 0 & t \le 0 \\ 0.01(t-60) + 0.4 & 60 < t \le 80 \\ 0.8 & t > 80 \end{cases}$$

$$f_{ay} = \begin{cases} 0 & 0 < t < 30 \\ 0.03(t-30) & 30 \le t \le 50 \\ -0.02(t-50) + 0.6 & 50 < t \le 80 \\ 0 & t > 80 \end{cases} \tag{31}$$

$$f_{az} = \begin{cases} 0 & t \le 40 \\ 0.2\sin((0.523t) - 1.57) & t > 40 \end{cases}$$

The simulation data is borrowed from [4]. The proportional and integral gains are obtained using proper index γ for fuzzy PMID. In case of fuzzy PD, only proportional gains are determined using (30).

The T-S Fuzzy PMID Descriptor Observer shows satisfactory performance when $q \ge 1$. The simulation results shown below are obtained using MATLAB/SIMULINKsoftware.

In order to get the T-S fuzzy model, linearization method is employed using suitably choosen 8 different operating points in the vicinity of equilibrium point (0,0,0). The reasonable index γ is choosen in such a way that feasible solution is obtained for LMIs in (21) & (30).

Derivative gain is obtained from (22) and proportional-integral gains are determined from (21) for fuzzy descriptor PMID observer. The fuzzy descriptor system (15) and fuzzy descriptor observer (16) are simulated in SIMULINK to get the outputs shown in this section. Similar procedure is followed for fuzzy descriptor PD observer using (22), (27), (28), and (30).

Actuator fault estimation using T-S fuzzy descriptor PMID and PD observers is shown in figures 1 – 6. It can be infered from these figures that proposed fuzzy descriptor PMID observer outperforms fuzzy descriptor PD observer. So, the multiple integral actions introduced in observer estimate the fault more better.

In order to have better idea, the figures for estimated error are also obtained that clearly represents that fuzzy descriptor PMID performs better than fuzzy descriptor PD observer.

The derivative gain can give us more design degrees of freedom. It can make us obtain fuzzy PMID and PD observer only with original coefficient matrices together with proportional or proportional-integral gains. Further, the effects of faults and disturbances are reduced with smaller values of $\bar{S}^{-1}$ as we increase the derivative gain. Due to such effect, faults are estimated more accurately.

In the present design, we have to take only the original system matrices into consideration which clearly indicates that the simultaneous observer is state-space observer. So it is more easy in computation and reliable in many applications [12].

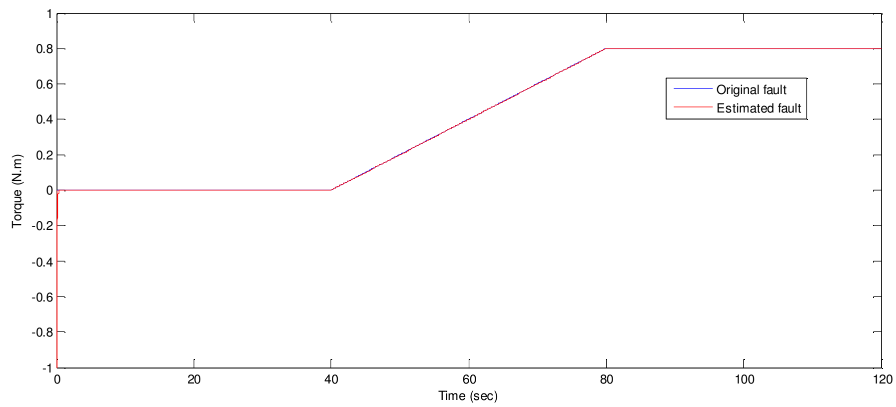


FIGURE 1: Actuator fault estimation along X-axis using T-S fuzzy descriptor PMID observer

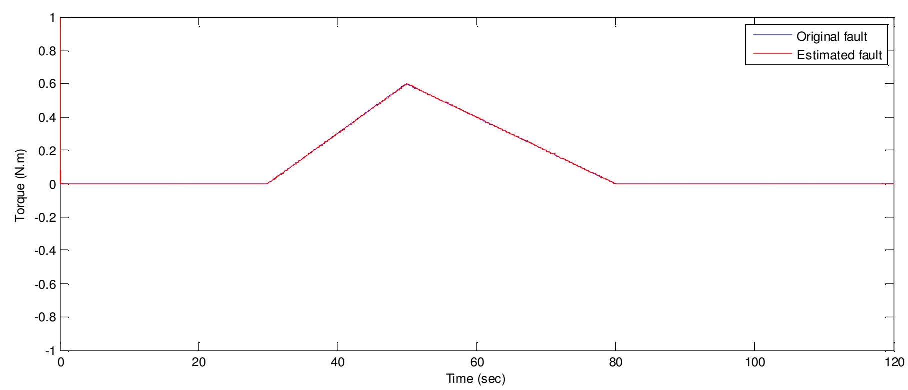


FIGURE 2: Actuator fault estimation along Y-axis using T-S fuzzy descriptor PMID observer

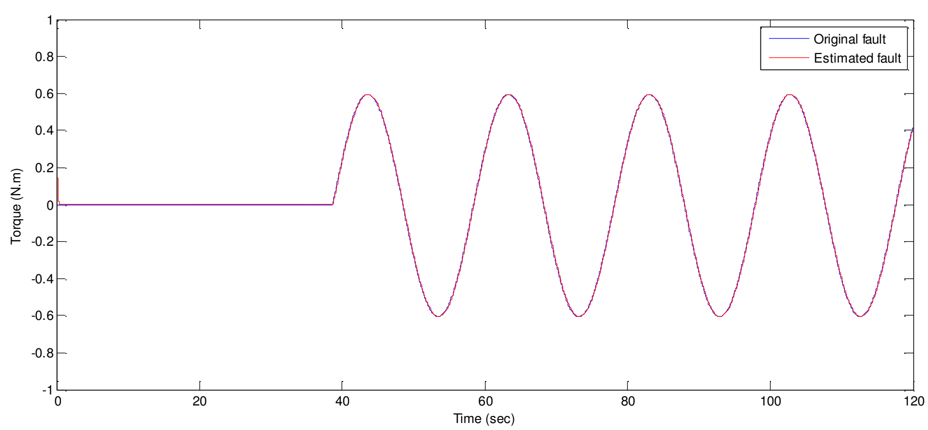


FIGURE 3: Actuator fault estimation along Z-axis using T-S fuzzy descriptor PMID observer

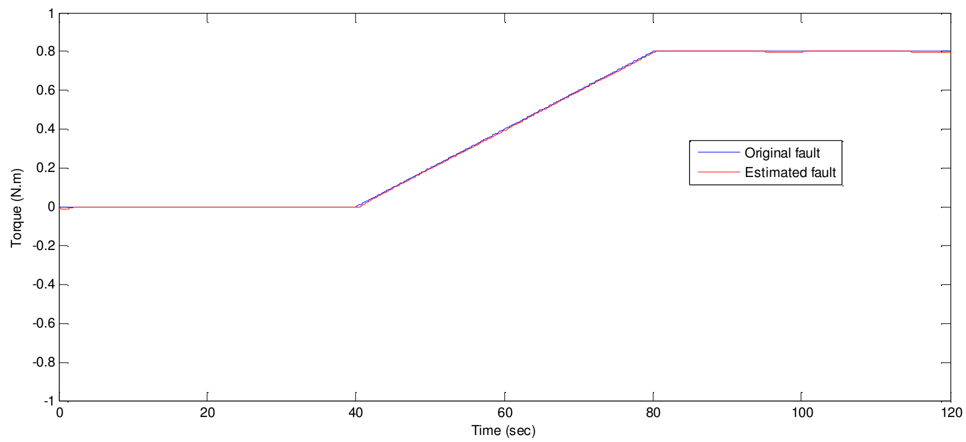


FIGURE 4: Actuator fault estimation along X-axis using T-S fuzzy descriptor PD observer

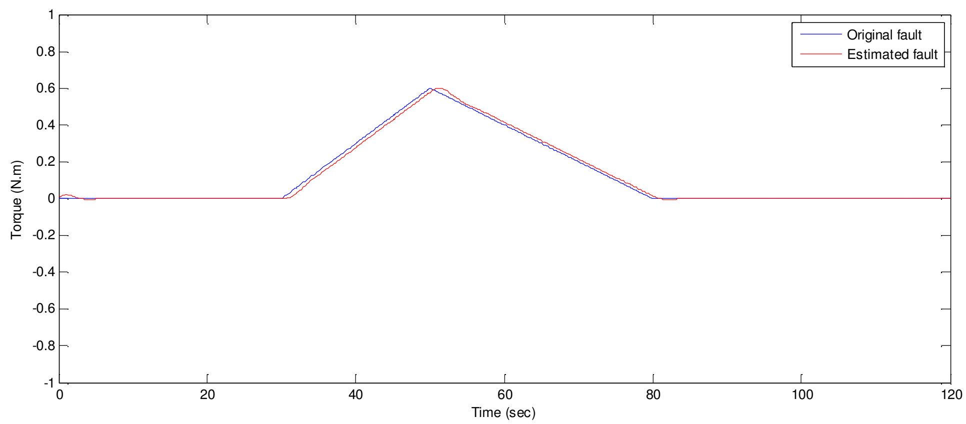


FIGURE 5: Actuator fault estimation along Y-axis using T-S fuzzy descriptor PD observer

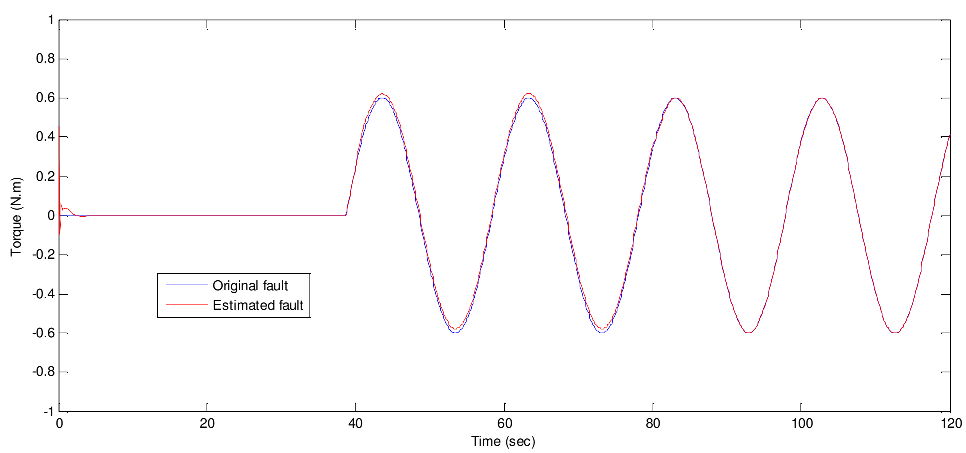


FIGURE 6: Actuator fault estimation along Z-axis using T-S fuzzy descriptor PD observer

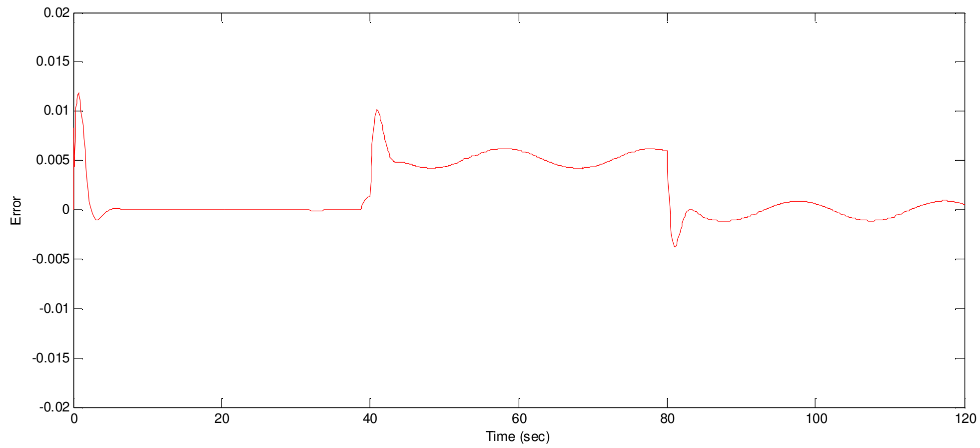


FIGURE 7: Estimated error for fault along X-axis using T-S fuzzy descriptor PD observer

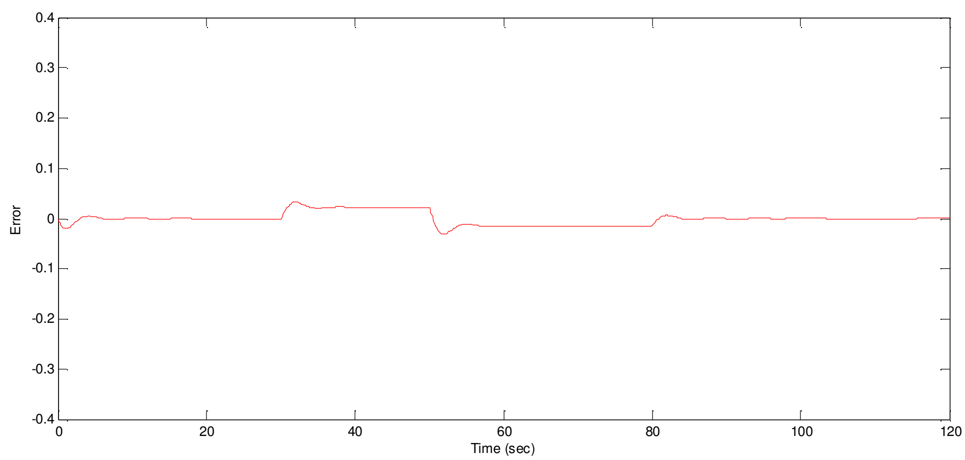


FIGURE 8: Estimated error for fault along Y-axis using T-S fuzzy descriptor PD observer

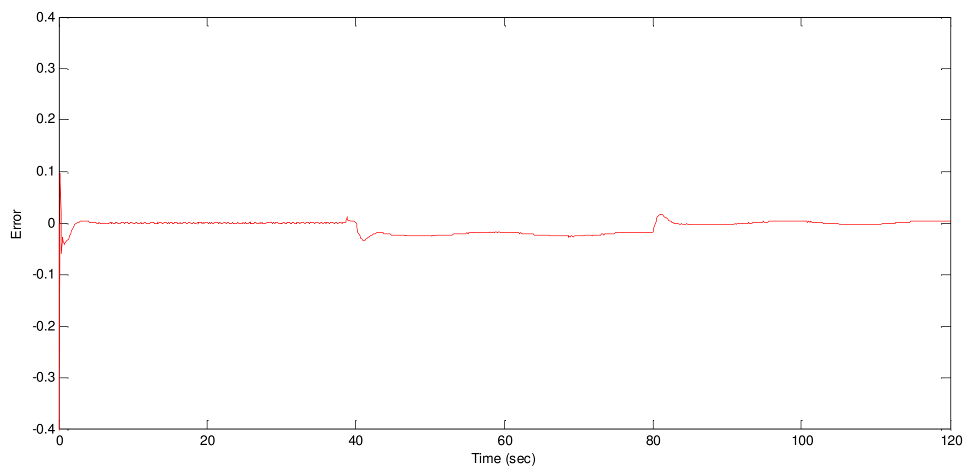


FIGURE 9: Estimated error for fault along Z-axis using T-S fuzzy descriptor PD observer

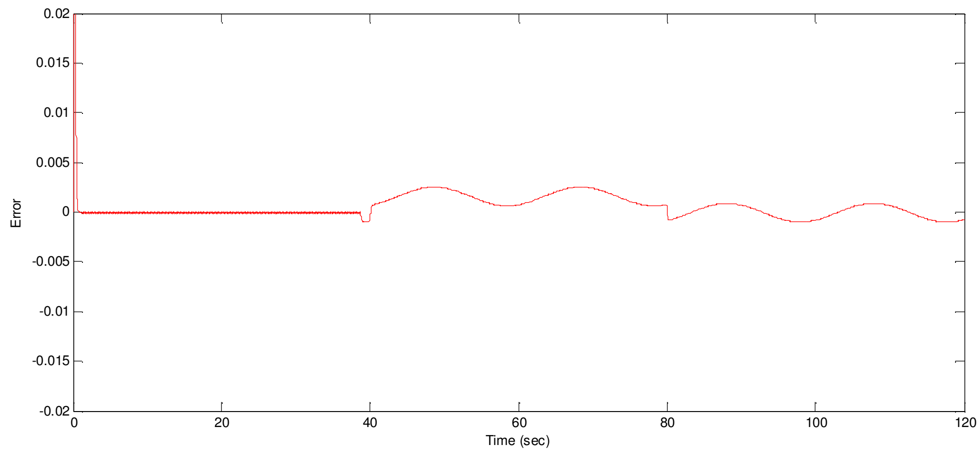


FIGURE 10: Estimated error for fault along X-axis using T-S fuzzy descriptor PMID observer

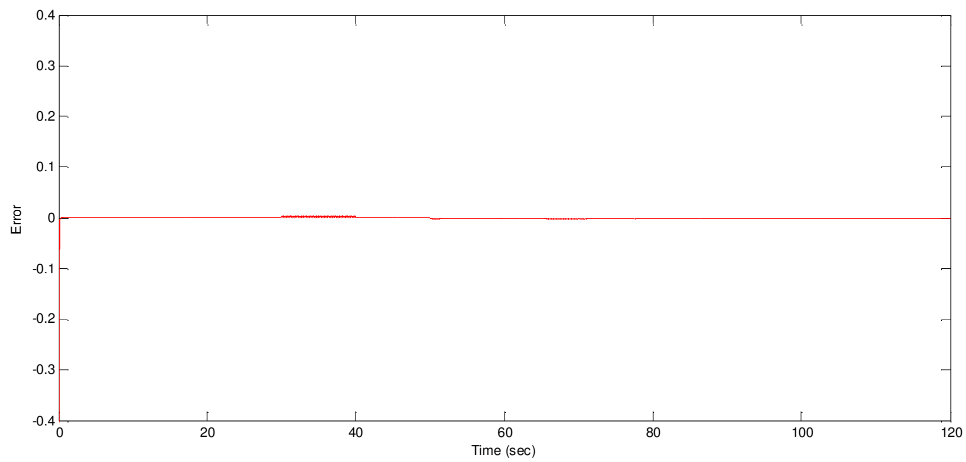


FIGURE 11: Estimated error for fault along Y-axis using T-S fuzzy descriptor PMID observer

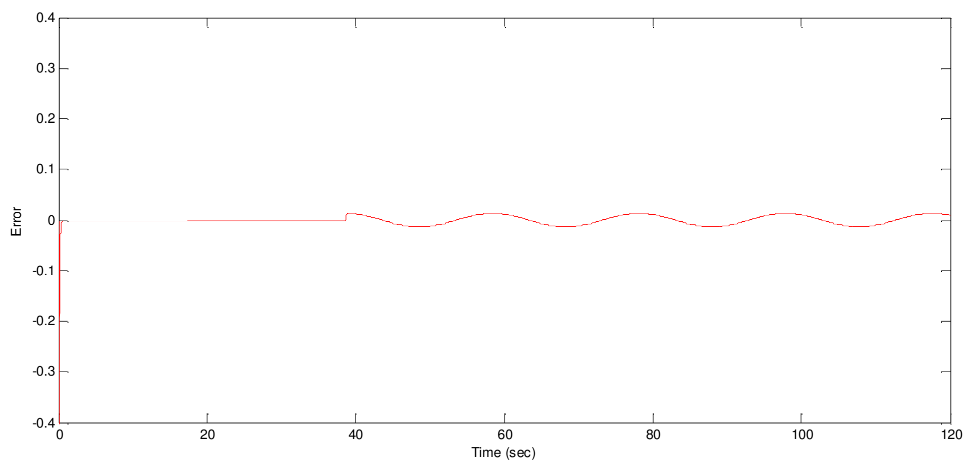


FIGURE 12: Estimated error for fault along Z-axis using T-S fuzzy descriptor PMID observer

In our proposed work, the multiple integral actions assumed for fault estimation reduced as compared to other observers in the literature available. We have assumed q=1 and as it can be noticed that [4] has assumed q=2. The less number of integral actions considered in our work is sufficient and better enough for observer design to reduce computational cost. As we increase the order of the faults, less integral actions would be required as per proposed work. Such difference arises due to derivative gain added in previously available fuzzy PMI observer.

The more design degrees of freedom given by derivative gain reduces the multiple integral actions and make observer design simpler. It should also be noticed that together with γ, we have γ_d index that makes fault estimation better. The main contribution of this paper is design of fuzzy PMID and fuzzy PD observers.

The results shown above in figures 1-6 support the proposed methods. It can be noticed that the estimated error in the fuzzy PMID observer is less than the fuzzy PD observer. The comparison of fig.7 and fig. 10 gives better picture that error in fig.7 is more. Similar conclusions can be made about figures for estimated error.

The figures 1-3 clearly show the good estimation of original fault along three dimensions. In order to get better notion, fig. 2 and fig. 5 should be compared. In fig. 2 original fault is estimated far better than in fig. 5. Thus, the conclusion can be drawn that the fuzzy PMID is better than fuzzy PD observer in terms of estimation of time varying faults.

Artificial neural networks are better approximator of nonlinear systems as compared to fuzzy logic methods. The extension of this research would include the construction of neuro-fuzzy observer. The fuzzy weights determined using linearization method can act as weights of neural networks and by choosing suitable activation function, artificial neural networks can be brought into play.

A better fault tolerant scheme can be designed for such observers. We have considered only the fault estimation. Fault diagnosis is another essential extension.

The continuous time-invariant system is considered for fault estimation here, the discrete time or continuous time variant systems can also be considered giving better application in real world problems.

The time delay systems (A (t + Δt), B (t + Δt), etc) with reduced order observer can also be designed providing less computational costs for observer design. While considering nonlinear models, modeling uncertainty should be taken into consideration which is of importance in the field of fault diagnosis.

6. CONCLUSION

The fuzzy descriptor proportional multiple integral derivative (PMID) and proportional derivative (PD) observers are proposed to estimate the actuator fault of satellite attitude control systems. The convergence condition of state estimation error is formulated in the form of LMI. The proposed observers are robust since they have been synthesized to decouple and attenuate both the effects of disturbances and fault approximated error. The main contribution can be noticed in terms of more design degrees of freedom added by derivative gain which enhances the system response. Simulation study reveals fuzzy descriptor PMID outperforms fuzzy descriptor PD observer in terms of robust actuator fault estimation.

7. REFERENCES

[1] W. Ai-Guo, D. Guang-Ren and F. Yan-Ming, "Generalized PID observer design for descriptor linear systems," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 37, No. 5, pp. 1390-1395, 2007.

[2] W. Ai-Guo and D. Guang-Ren, "PMID observer design of descriptor linear systems," In Proceedings of the 26th Chinese Control Conference, 2007, pp. 161-165.

[3] S. Ding. (2008), Model-based fault diagnosis techniques (1st edition), [Online]. Available: http://0www.springerlink.com.oasis.lib.tamuk.edu/content/?k=model+based+fault+diagnosis [Feb., 2011]

[4] Y. Fu, Y. Cheng and B. Jiang, "Robust actuator fault estimation for satellite control system via fuzzy proportional multiple-integral observer method," in Proceedings of the 29th Chinese Control Conference, 2010, pp. 3942-3946.

[5] J. Gun-Ping, W. Suo-Ping and S. Wen-Zhong, "Design of observer with integrators for linear systems with unknown input disturbances," IEE Electronic Letters, Vol. 36, No. 13, pp. 1168-1169, 2000.

[6] Y. Jinyong and L. Zhiyuan, "Fault reconstruction based on sliding mode observer for linear descriptor systems," In Proceedings of the 7th Asian Control Conference, 2009, pp. 1132-1137.

[7] J. Lia-hui, T. Fa, and L. Chen, "Fault reconstruction based on sliding mode observers," World Congress on Computer Science and Information Engineering, IEEE Computer Society, pp. 479-484, 2009.

[8] Z. Lendek, T.M. Guerra, R. Babuska, and B. De Schutter (2011), Stabilty Analysis and Nonlinear Observer Design using Takagi-Sugeno Fuzzy Models (1st edition), [Online]. Available: http://0-www.springerlink.com [April, 2011]

[9] K. Meinghed, J. Yame, and C. Aubrun, "Modifica-tion of an adaptive observer to improve its performance in actuator fault detection," In Proceedinds of 18th Mediterranean Conference on Control & Automation, Morocco, 2008,pp. 75-81.

[10] B. Shafai and R L. Carroll, "Design of proportional integral observer for linear time- varying multivariable systems," In Proceedings of 24th IEEE conference on Decision Control, 1985, pp. 597-599.

[11] T. Takagi and M. Sugeno, "Fuzzy identification of the systems and its application to modeling and control," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-15, No. 1, pp. 116-132, 1985.

[12] Z. Gao and S.X.Ding, "Fault estimation and fault-tolerant control for descriptor systems via proportional, multiple-integral and derivative observer design," IET Control Theory Appl., Vol. 1, No. 5, pp. 1208-1218, 2007.

[13] Christopher Edwards, Thomas Lombaerts and Hafid Smaili, "Fault Tolerant Flight Control," Lecture notes in Control and Information Sciences, 2010.

[14] S. X. Ding, P. Zhang, B. Huang, E. L. Ding, and P. M. Frank, "An approach to norm and statistical methods based residual evaluation," In Proceedings of the 10th IEEE International Conference on Methods and Models in Automation and Robotics, pp. 777 - 780, 2004.

[15] X. Ding and P. M. Frank, "Frequency domain approach and threshold selector for robust model-based fault detection and isolation," In Proceedings of the 1st IFAC Symp. Safe-process, 1991.

[16] X. Ding and P. M. Frank, "An adaptive observer-based fault detection scheme for nonlinear systems," in Proceedings of the 12th IFAC World Congress, Sydney, pp. 307-312, 1993.

[17] P. M. Frank and X. Ding, "Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis," Automatica, vol. 30, pp. 789-904, 1994.

[18] E. Frisk and L. Nielsen, "Robust residual generation for diagnosis including a reference model for residual behavior," Automatica, vol. 42, pp. 437-445, 2006

[19] Ashish Teewari, "Atmospheric and Space Flights," Second edition, 2007, pp. 370-372.

[20] Ajit K. Mandal, "Introduction to Control Engineering," Third edition, 2006, pp. 255-275.

[21] Guido Herrmann, Matthew C. Turner and Ian Postlethwaite, "Mathematical Methods for Robust & Nonlinear Control," Second edition, 2007, pp. 170-175.

[22] Kevin M. Passino and Stephen Yurkovich, "Fuzzy Control," First edition, 1998, pp. 121-130.

[23] Nguyen Tan Tien, "Applied Nonlinear Control-Fundamentals of Lyapunov Theory," Second edition, 2002, pp. 68-79.

[24] Stephen Boyd, "Linear Matrix Inequalities in System and Control Theory," 1994, pp. 25-33.

[25] S. Linder, B. Shafai and M. Saif, "Estimating and Accomodating Unknown Actuator Faults with PI Observer," in Proceedings of the 1988 IEEE International Conference on Control Applications, 1998.

[26] G. Duan and G. Liu, "Eigenstructure assignment design for proportional integral observers: continuous time case," in Proceedings of IEE Control Theory Appl. Vol. 148, No. 3, May 2001.

[27] K. Krishna and P. Kabore, "On the design of integral and proportional integral observers," in Proceedings of the American Control Conference, June 2009.

[28] D. Ichalal, B. Marx and J. Ragot, "Observer based actuator fault tolerant control for nonlinear Takagi-Sugeno systems: an LMI approach," in Proceedings of Mediterranean Conference on Control & Automation (MED), pp. 1278-1283, June 2010.

[29] S.R. Beale and B. Shafai, "Robust control system design with the proportional integral observer," in Proceedings of the 27th conference on Decision and Control," pp. 554-557, 1988.

[30] Q. Wu and M. Saif, "Robust fault Diagnosis for a Satellite using Neural Sliding Mode Observer," in Proceedings of the 44th IEEE Conference on Decision and Control, pp. 7668-7673, 2005.

[31] B. Marx, D. Koeing and D. Georges, "Robust fault diagnosis for linear descriptor systems using proportional and integral observers," in Proceedings of the 42nd IEEE Conference on Decision and Control, pp. 457-462, 2003.

Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control

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

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

Iraj Asadi Talooki 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

Payman Ferdosali 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

This paper expands a Multi Input Multi Output (MIMO) fuzzy estimator variable structure control (VSC) which controller coefficient is on-line tuned by fuzzy backstepping algorithm. The main goal is to guarantee acceptable trajectories tracking between the internal combustion engine (IC engine) air to fuel ratio and the desired input. The fuzzy controller in proposed fuzzy estimator variable structure controller is based on Lyapunov fuzzy inference system (FIS) with minimum model based rule base. The input represents the function between variable structure function, error and the rate of error. The outputs represent fuel ratio, respectively. The fuzzy backstepping methodology is on-line tune the variable structure function based on adaptive methodology. The performance of the MIMO fuzzy estimator VSC which controller coefficient is on-line tuned by fuzzy backstepping algorithm (FBAFVSC) is validated through comparison with VSC and proposed method. Simulation results signify good performance of fuel ratio in presence of uncertainty and external disturbance.

Keywords: Internal Combustion Engine, Variable Structure Controller, Fuzzy Backstepping Controller, Chattering Phenomenon, Adaptive Methodology, Proposed Fuzzy Estimator Variable Structure Controller, Lyapunov Based Controller.

1. MOTIVATION, INTRODUCTION AND BACKGROUND

Motivation: Internal combustion (IC) engines are optimized to meet exhaust emission requirements with the best fuel economy. Closed loop combustion control is a key technology that is used to optimize the engine combustion process to achieve this goal. In order to conduct research in the area of closed loop combustion control, a control oriented cycle-to-cycle engine model, containing engine combustion information for each individual engine cycle as a function of engine crank angle, is a necessity. Air-to-fuel (A/F) ratio is the mass ratio of air to fuel trapped inside a cylinder before combustion begins, and it affects engine emissions, fuel economy, and other performances. In this research, a fuzzy backstepping adaptive MIMO fuzzy estimator variable structure control scheme is used to simultaneously control the mass flow rate of port fuel injection (PFI) systems to regulate the A/F ratio of PFI to desired levels. One of the most important challenges in the field of IC engine is IC engine control, because this system is MIMO, nonlinear, time variant parameter and uncertainty [63-71]. Presently, IC engines 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. Classical and non-classical methods are two main categories of nonlinear plant 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 [1-2].

Introduction: Modeling of an entire IC engine is a very important and complicated process because engines are nonlinear, multi inputs-multi outputs and time variant. One purpose of accurate modeling is to save development costs of real engines and minimizing the risks of damaging an engine when validating controller designs. Nevertheless, developing a small model, for specific controller design purposes, can be done and then validated on a larger, more complicated model. [63-71]. Dynamic modeling of IC engines is used to describe the behavior of this system, design of model based controller, and for simulation. The dynamic modeling describes the relationship between nonlinear output formulation to electrical or mechanical source and also it can be used to describe the particular dynamic effects to behavior of system[1]. In an internal combustion engine, a piston moves up and down in a cylinder and power is transferred through a connecting rod to a crank shaft. The continual motion of the piston and rotation of the crank shaft as air and fuel enter and exit the cylinder through the intake and exhaust valves is known as an engine cycle. The first and most significant engine among all internal combustion engines is the Otto engine, which was developed by Nicolaus A. Otto in 1876 (Figure 1). In his engine, Otto created a unique engine cycle that consisted of four piston strokes. These strokes are: intake stroke, compression stroke, expansion stroke and exhaust stroke [63-71].

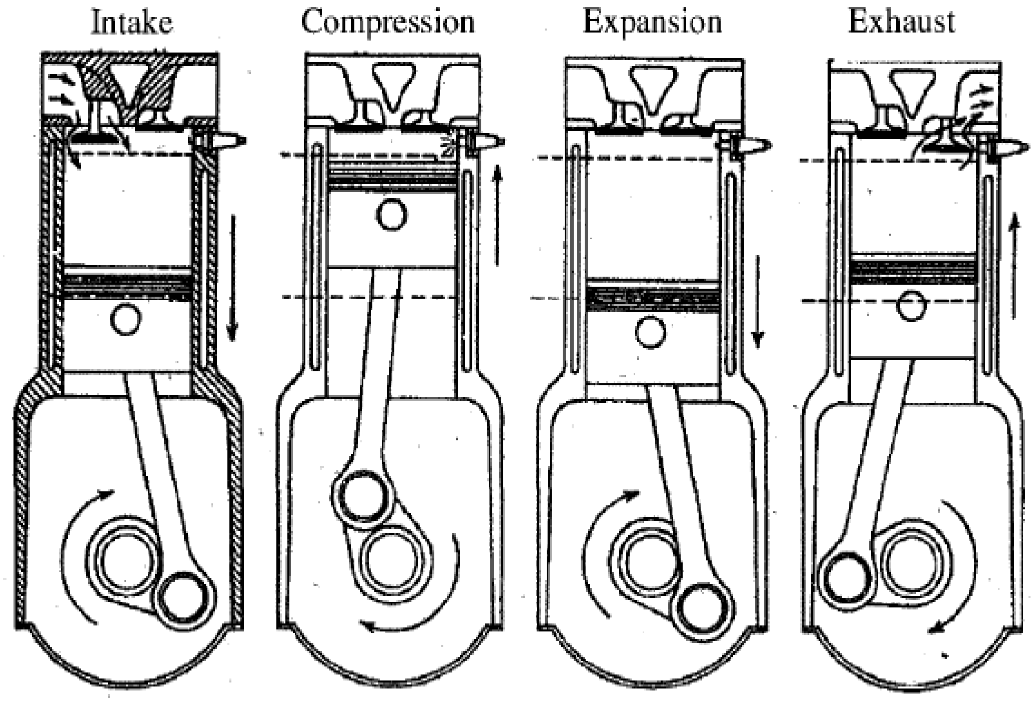


FIGURE 1: The Four Stroke Engine Cycle [1]

Controller (control system) is a device which can sense information from linear or nonlinear system (e.g., IC engine) to improve the systems performance [3-20]. There are two control systems: open-loop and close-loop (feedback), by using the open-loop system the control action is independent of the output and in close-loop system the control action depends on the output. In feedback control system considering that there are many disturbances and also variable dynamic parameters something that is really necessary is keeping plant variables close to the desired value. Feedback control system development is the most important thing in many different fields of engineering. The feedback controllers also have classified into two general groups'; positive feedback and negative feedback, which in positive the feedback signal can amplify the effect of the input signal and in negative the feedback signal can reduce the effect of the input signal. Figure 2 shows a basic block diagram of a simple negative feedback system with single-input, single-output (SISO). The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[5-29]. 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 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., parameter variations). At present, in some applications IC engines 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). One of the best nonlinear robust controllers is variable structure control which is used in nonlinear uncertain systems.

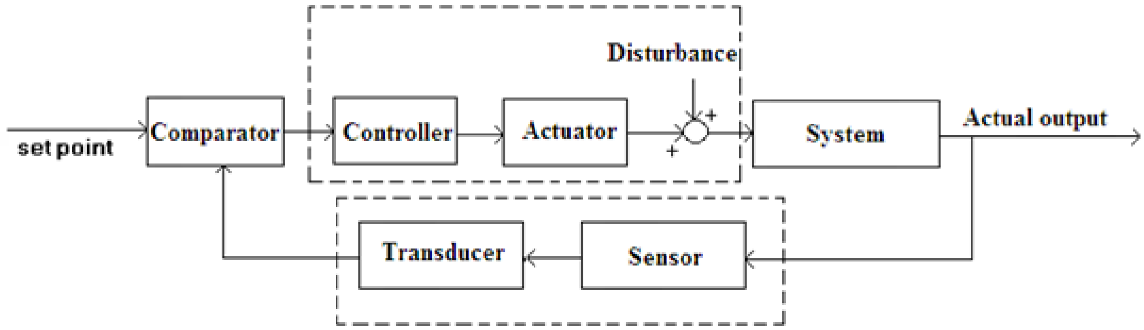


FIGURE 2: Block diagram of a feedback control[3]

One of the nonlinear robust controllers is variable structure controller, although this controller has been analyzed by many researchers but the first proposed was in the 1950 [41-62].This controller is used in wide range areas such as in robotics, in control process, in aerospace applications and in IC engines because this methodology can solve some main challenging topics in control such as resistivity to the external disturbance and stability. Even though, this controller is used in wide range areas but, pure variable structure controller has two drawbacks: Firstly, output oscillation (chattering); which caused the heating in the mechanical parameters. Secondly, nonlinear dynamic formulation of nonlinear systems which applied in nonlinear dynamic nonlinear controller; calculate this control formulation is absolutely difficult because it depends on the dynamic nonlinear system's equation [20-23]. Chattering phenomenon can causes some problems such as saturation and heats the mechanical parts of IC engine or drivers. To reduce or eliminate the oscillation, various papers have been reported by many researchers which one of the best method is; boundary layer saturation method [1]. In boundary layer linear saturation method, the basic idea is the discontinuous method replacement by linear continuous saturation method with small neighborhood of the switching surface. This replacement caused to considerable chattering reduction. Estimated uncertainty method used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to reduce the chattering phenomenon and also solve nonlinear dynamic formulation. This research introduced novel applied MIMO fuzzy inference engine to variable structure controller to estimate the nonlinear control formulation with low computation load. In recent years, artificial intelligence theory has been used in variable structure control systems. 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., IC engines). Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory [30-41]. 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 IC engines using classical controllers are based on IC engines dynamic modelling. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of IC engines, but these models are MIMO, non-linear and calculate accurate dynamic modelling is definitely 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]. Research on applied fuzzy logic methodology in variable structure controller (FVSC) to reduce or eliminate the high frequency oscillation (chattering) and to compensate the unknown system dynamics pure variable structure controller considerably improves the nonlinear plant control process [42-43]. Investigation on applied variable structure methodology in fuzzy logic controller (VSFC) 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 nonlinear plant control [23]; [48-50]. However the application of FVSC and VSFC are growing but the main VSFC drawback compared to FVSC is calculation the value of structure surface λ pri-defined very carefully. The advantages of VSFC compared to fuzzy logic controller (FLC) is reduce the number of fuzzy rule base and increase the robustness and stability. At last FVSC compare to the VSFC is more suitable for implementation action. In various dynamic parameters systems that need to be training on-line adaptive control methodology is used. 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 variable structure controller.

Background: Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented variable structure method with boundary layer to improve the industry application [22]. R. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Their design had three main parts i.e. equivalent, discontinuous and fuzzy estimator tuning part which has reduced the chattering very well. Elmali et al. [27]and Li and Xu [29]have addressed variable structure control with perturbation estimation method (VSCPE) to reduce the classical variable structure chattering. 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. H.Temeltas [46] has proposed fuzzy adaption techniques for VSC to achieve robust tracking of nonlinear systems and solves the chattering problem. Conversely system's performance is better than variable structure controller; it is depended on nonlinear dynamic equqation. C. L. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based variable structure control based on *N* fuzzy based linear state-space to estimate the uncertainties. A MIMO FVSC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a nonlinear system [42]. Yoo and Ham [58]have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. This method can only tune the consequence part of the fuzzy rules. Medhafer et al. [59] have proposed an indirect adaptive fuzzy variable structure controller to control nonlinear system. This MIMO algorithm, applies to estimate the nonlinear dynamic parameters. Compared with the previous algorithm the numbers of fuzzy rules have reduced by introducing the variable structure surface as inputs of fuzzy systems. Y. Guo and P. Y. Woo [60]have proposed a SISO fuzzy system compensate and reduce the chattering. C. M. Lin and C. F. Hsu [61] can tune both systems by fuzzy rules. Shahnazi et al., have proposed a SISO PI direct adaptive fuzzy variable structure controller based on Lin and Hsu algorithm to reduce or eliminate chattering. The bounds of PI controller and the parameters are online adjusted by low adaption computation [44].
In this research we will highlight the MIMO adaptive backstepping fuzzy variable structure algorithm with estimates the nonlinear dynamic part derived in the Lyapunov sense. This

algorithm will be analyzed and evaluated on IC engine. Section 2, serves as an introduction to the variable structure formulation algorithm and its application to an IC engine, describe the objectives and problem statements. Part 3, introduces and describes the methodology algorithms and proves Lyapunov stability. Section 4 presents the simulation results of this algorithm applied to an IC engine and the final section is describe the conclusion.

2. OBJECTIVES, PROBLEM STATEMENTS AND VARIABLE STRUCTURE FORMULATION APPLIED TO IC ENGINE

Dynamic of IC engine: In developing a valid engine model, the concept of the combustion process, abnormal combustion, and cylinder pressure must be understood. The combustion process is relatively simple and it begins with fuel and air being mixed together in the intake manifold and cylinder. This air-fuel mixture is trapped inside cylinder after the intake valve(s) is closed and then gets compressed [13]. When the air-fuel mixture is compressed it causes the pressure and temperature to increase inside the cylinder. Unlike normal combustion, the cylinder pressure and temperature can rise so rapidly that it can spontaneously ignite the air-fuel mixture causing high frequency cylinder pressure oscillations. These oscillations cause the metal cylinders to produce sharp noises called knock, which it caused to abnormal combustion. The pressure in the cylinder is a very important physical parameter that can be analyzed from the combustion process. Since cylinder pressure is very important to the combustion event and the engine cycle in spark ignition engines, the development of a model that produces the cylinder pressure for each crank angle degree is necessary. A cylinder pressure model that calculates the total cylinder pressure over 720 crank angle degrees was created based upon the following formulation [63-71]:

$$P_{cyl}(\theta) = P_m(\theta) + P_{net}(\theta) \tag{1}$$

where $P_{cyl}(\theta)$ is pressure in cylinder, $P_m(\theta)$ is Wiebe function, and $P_{net}(\theta)$ is motoring pressure of a cylinder. Air fuel ratio is the mass ratio of air and fuel trapped inside the cylinder before combustion starts. Mathematically it is the mass of the air divided by the mass of the fuel as shown in the equation below:

$$Air\ to\ Fuel = \frac{\dot{m}_{air}}{\dot{m}_{fuel}} \tag{2}$$

If the ratio is too high or too low, it can be adjusted by adding or reducing the amount of fuel per engine cycle that is injected into the cylinder. The fuel ratio can be used to determine which fuel system should have a larger impact on how much fuel is injected into the cylinder. Since a direct fuel injector has immediate injection of its fuel with significant charge cooling effect, it can have a quicker response to the desired amount of fuel that is needed by an engine [66].

Variable structure methodology: Based on variable structure discussion, the control law for a multi degrees of freedom robot manipulator is written as [18-24]:

$$U = U_{Nonlinear} + U_{dis} \tag{3}$$

Where, the model-based component $U_{Nonlinear}$ is compensated the nominal dynamics of systems. Therefore $U_{Nonlinear}$ can calculate as follows:

$$U_{Nonlinear} = [M^{-1}(P_m(\theta) + P_{net}(\theta)) + \dot{S}]M \tag{4}$$

Where

$$M^{-1} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}^{-1} \quad M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

A simple solution to get the variable structure condition when the dynamic parameters have uncertainty is the switching control law:

$$U_{dis} = K(\vec{x}, t) \cdot \mathrm{sgn}(s) \tag{5}$$

where the switching function $\mathbf{sgn(S)}$ is defined as

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{6}$$

and the $K(\vec{x}, t)$ is the positive constant.
the lyapunov formulation can be written as follows,

$$V = \frac{1}{2} S^T . M . S \tag{7}$$

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2} S^T . \dot{M} . S + S^T \ M\dot{S} \tag{8}$$

the dynamic equation of IC engine can be written based on the structure variable surface as

$$M\dot{S} = -VS + M\dot{S} + P_m(\theta) + P_{net}(\theta) \tag{9}$$

it is assumed that

$$S^T \left(\dot{M} - 2P_m(\theta) + P_{net}(\theta) \right) S = 0 \tag{10}$$

by substituting (15) in (14)

$$\dot{V} = \frac{1}{2} S^T \dot{M} S - S^T P_m(\theta) + P_{net}(\theta) S + S^T \left(M\dot{S} + P_m(\theta) + P_{net}(\theta) S \right) = S^T \left(M\dot{S} + P_m(\theta) + P_{\text{,}} \right. \tag{11}$$

suppose the control input is written as follows

$$\hat{U} = \widehat{U_{Nonlinear}} + \widehat{U_{dis}} = [\widehat{M^{-1}}(P_m(\theta) + P_{net}(\theta)) + \dot{S}]\hat{M} + K.sgn(S) + P_m(\theta) + P_{net}(\theta) S \tag{12}$$

by replacing the equation (18) in (17)

$$\dot{V} = S^T (M\dot{S} + P_m(\theta) + P_{net}(\theta) - \hat{M}\dot{S} - P_m(\theta) \widehat{+ P}_{net}(\theta) S - Ksgn(S) = \\ S^T \left(\tilde{M}\dot{S} + P_m(\theta) \widetilde{+ P}_{net}(\theta) S - Ksgn(S) \right) \tag{13}$$

it is obvious that

$$\left| \tilde{M}\dot{S} + P_m(\theta) \widetilde{+ P}_{net}(\theta) S \right| \le \left| \tilde{M}\dot{S} \right| + \left| P_m(\theta) \widetilde{+ P}_{net}(\theta) S \right| \tag{14}$$

the Lemma equation in IC engine system can be written as follows

$$K_u = \left[\left| \tilde{M}\dot{S} \right| + \left| P_m(\theta) + P_{net}(\theta) S \right| + \eta \right]_i , i = 1, 2, 3, 4, \ldots \tag{15}$$

the equation (16) can be written as

$$K_u \ge \left| \left[\tilde{M}\dot{S} + P_m(\theta) + P_{net}(\theta) S + \right]_i \right| + \eta_i \tag{16}$$

therefore, it can be shown that

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i \left| S_i \right| \tag{17}$$

Consequently the equation (17) guaranties the stability of the Lyapunov equation. Figure 3 is shown the integration of mean value engine and cylinder pressure model.

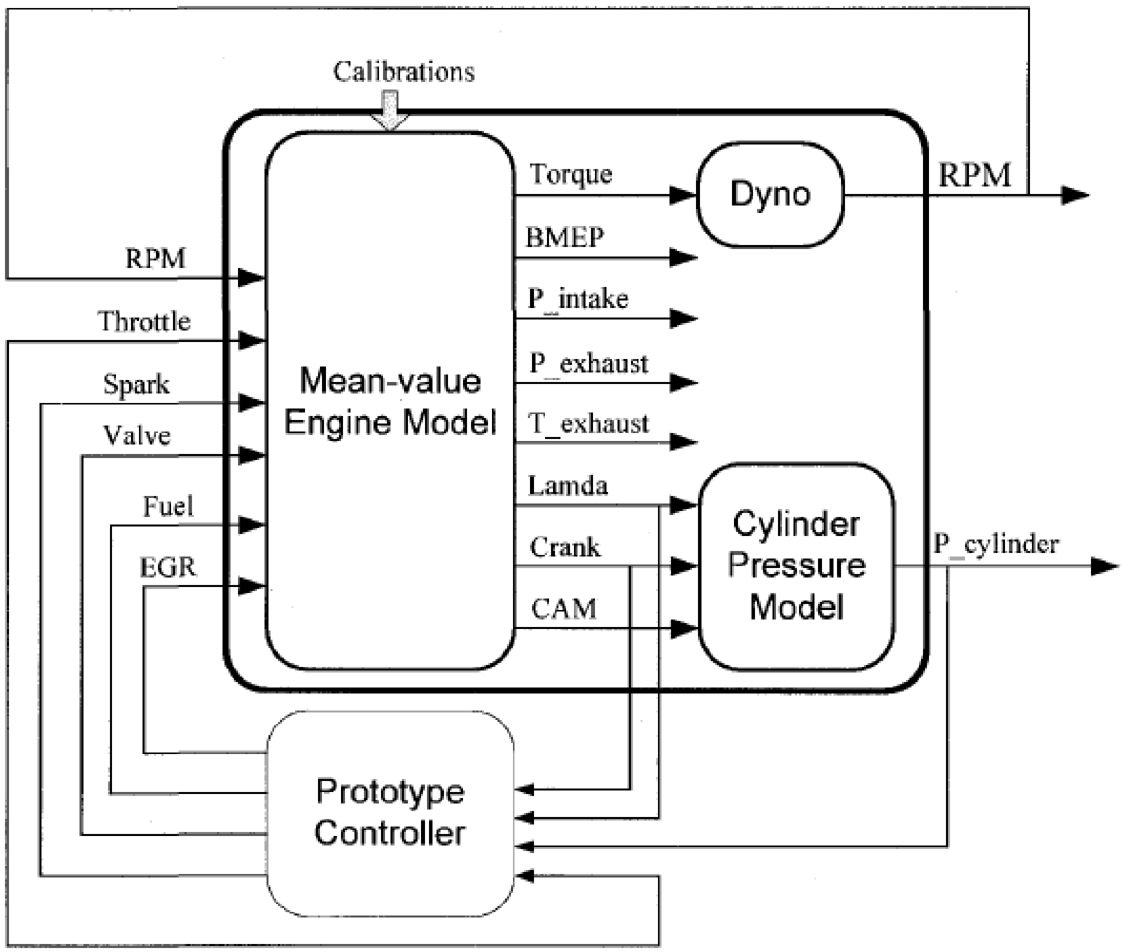


FIGURE 3: Block diagram of an integration of mean value engine and cylinder pressure model

Figure 4 is shown pure variable structure controller, applied to IC engine.

Problem Statements

Even though, variable structure controller is used in wide range areas but, pure it also has chattering problem and nonlinear dynamic part challenges. On the other hand, fuzzy logic controller has been used for nonlinear and uncertain (e.g., IC engine) systems controlling. Conversely pure FLC works in many areas, it cannot guarantee the basic requirement of stability and acceptable performance[8]. Although both VSC and FLC have been applied successfully in many applications but they also have some limitations. The boundary layer method is used to reduce or eliminate the chattering and proposed fuzzy Lyapunov estimator method focuses on substitution fuzzy logic system instead of dynamic nonlinear equation to implement easily and avoid mathematical model base controller. To reduce the effect of uncertainty in proposed method, MIMO novel adaptive method is applied in fuzzy variable structure controller in IC engine.

Objectives

The main goal is to design a MIMO fuzzy backstepping adaptive fuzzy estimation variable structure methodology which applied to internal combustion engine with easy to design and implement. IC engine has nonlinear dynamic and uncertain parameters consequently; following objectives have been pursuit in the mentioned research: To develop a chattering in a position pure variable structure controller against uncertainties, to design and implement a Lyapunov MIMO fuzzy structure variable controller in order to solve the equivalent problems with minimum rule base and finally to develop a position fuzzy backstepping adaptive fuzzy estimation variable structure controller in order to solve the disturbance rejection and reduce the computation load.

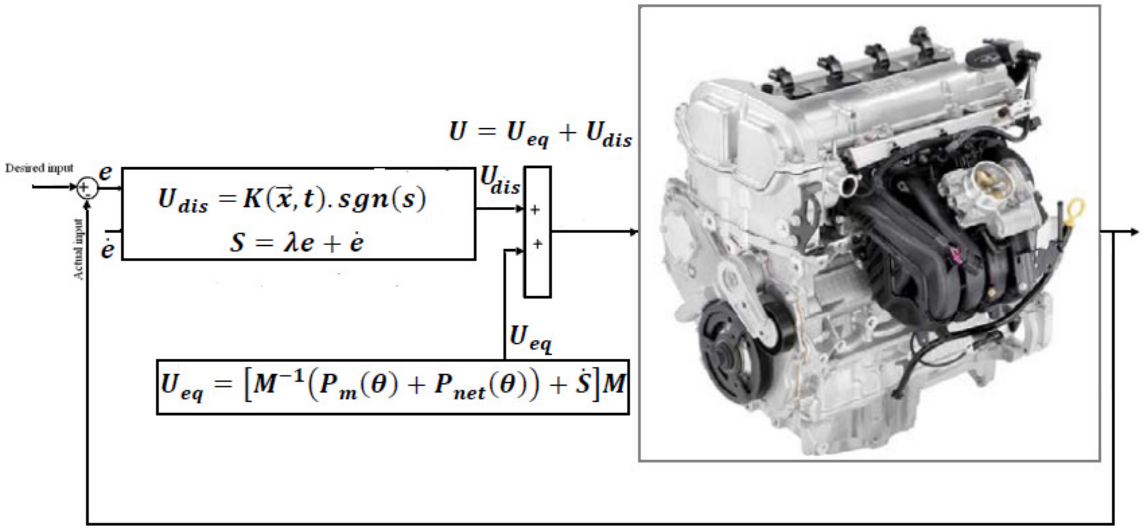


FIGURE 4: Block diagram of a variable structure controller: applied to IC engine

3. METHODOLOGY: DESIGN A NOVEL MIMO FUZZY BACKSTEPPING ADAPTIVE FUZZY ESTIMATION VARIABLE STRUCTURE CONTROL

First part is focused on eliminate the oscillation (chattering) in pure variable structure controller based on linear boundary layer method. To reduce or eliminate the chattering it is used the boundary layer method; in boundary layer method the basic idea is replace the discontinuous method by saturation (linear) method with small neighborhood of the switching surface. This replace is caused to increase the error performance [20-24].

$$B(t) = \{x, |S(t)| \leq \emptyset\}; \emptyset > 0 \quad \textbf{(18)}$$

Where $\emptyset$ is the boundary layer thickness. Therefore, to have a smote control law, the saturation function $Sat(S/\emptyset)$ added to the control law:

$$U = K(\vec{x},t).Sat\left(S/\emptyset\right) \quad \textbf{(19)}$$

Where $Sat\left(S/\emptyset\right)$ can be defined as

$$sat\left(S/\emptyset\right) = \begin{cases} 1 & (S/\emptyset > 1) \\ -1 & (S/\emptyset < 1) \\ S/\emptyset & (-1 < S/\emptyset < 1) \end{cases} \quad \textbf{(20)}$$

Based on above discussion, the control law for an IC engine is written as [18-24]:

$$U = U_{eq} + U_r \quad \textbf{(21)}$$

Figure 5 is shown classical variable structure which eliminates the chattering using linear boundary layer method.

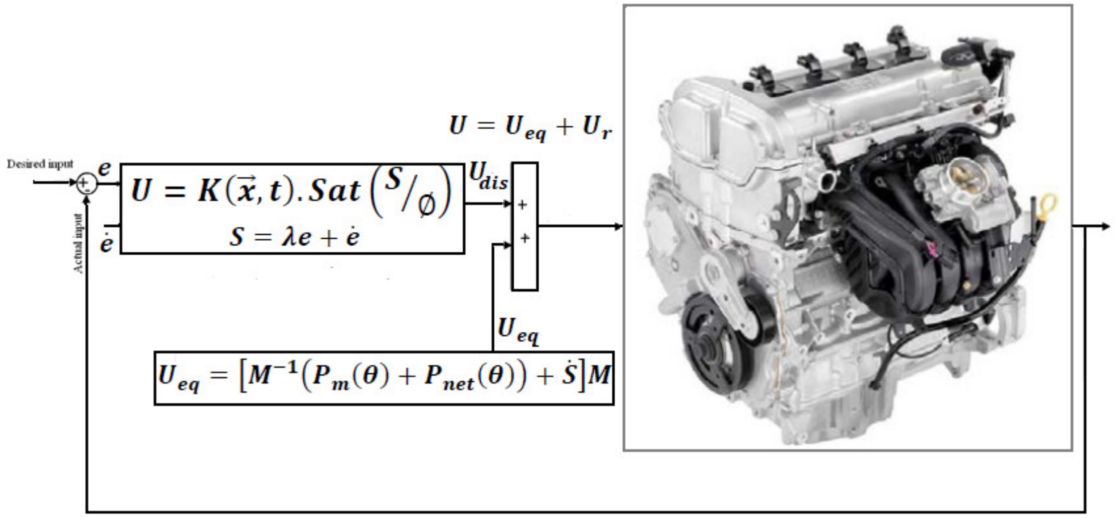


FIGURE 5: Chattering free Block diagram of a variable structure controller: applied to IC engine

Second step is focused on design MIMO fuzzy estimation variable structure based on Lyapunov formulation. The firs type of fuzzy systems is given by

$$f(x) = \sum_{l=1}^{M} \theta^l \mathcal{E}^l(x) = \theta^T \mathcal{E}(x) \quad \textbf{(22)}$$

Where $\theta = (\theta^1, \dots, \theta^M)^T, \mathcal{E}(x) = (\mathcal{E}^1(x), \dots, \mathcal{E}^M(x))^T, and\ \mathcal{E}^l(x) =: \prod_{i=1}^{n} \frac{\mu_{A_i^l}(x_i)}{} \sum_{l=1}^{M}(\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$. $\theta^1, \dots, \theta^M$ are adjustable parameters in (28) . $\mu_{A_1^1}(x_1), \dots, \mu_{A_n^m}(x_n)$ are given membership functions whose parameters will not change over time.

The second type of fuzzy systems is given by

$$f(x) = \frac{\sum_{l=1}^{M} \theta^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]} \quad \textbf{(23)}$$

Where $\theta^l, \alpha_i^l\ and\ \delta_i^l$ are all adjustable parameters. From the universal approximation theorem, we know that we can find a fuzzy system to estimate any continuous function. For the first type of fuzzy systems, we can only adjust θ^l in (28). We define $f^{\wedge}(x|\theta)$ as the approximator of the real function $f(x)$.

$$f^{\wedge}(x|\theta) = \theta^T \varepsilon(x) \quad \textbf{(24)}$$

We define θ^* as the values for the minimum error:

$$\theta^* = \arg\min_{\theta \in \Omega} \left[\sup_{x \in U} |f^{\wedge}(x|\theta) - g(x)|\right] \quad \textbf{(25)}$$

Where Ω is a constraint set for θ. For specific $x, sup_{x \in U}|f^{\wedge}(x|\theta^*) - f(x)|$ is the minimum approximation error we can get.

We used the first type of fuzzy systems (23) to estimate the nonlinear system (10) the fuzzy formulation can be write as below;

$$f(x|\theta) = \theta^T \varepsilon(x) = \frac{\sum_{l=1}^{n} \theta^l [\mu_{A^l}(x)]}{\sum_{l=1}^{n} [\mu_{A^l}(x)]} \quad (26)$$

Where $\theta^1, \dots, \theta^n$ are adjusted by an adaptation law. The adaptation law is designed to minimize the parameter errors of $\theta - \theta^*$. A MIMO (multi-input multi-output) fuzzy system is designed to compensate the uncertainties of the nonlinear system. The parameters of the fuzzy system are adjusted by adaptation laws. The tracking error and the sliding surface state are defined as:

$$e = q - q_d \quad (27)$$

$$s = \dot{e} + \lambda_e \quad (28)$$

We define the reference state as

$$\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e \quad (29)$$

$$\ddot{q}_r = \ddot{q} - \dot{s} = \ddot{q}_d - \lambda \dot{e} \quad (30)$$

The general MIMO if-then rules are given by

$$R^l: if\ x_1\ is\ A_1^l\ , x_2\ is\ A_2^l\ , \dots , x_n\ is\ A_n^l\ , then\ y_1 is\ B_1^l\ , \dots , y_m\ is\ B_m^l \quad (31)$$

Where $l = 1,2,\dots,M$ are fuzzy if-then rules; $x = (x_1, \dots, x_n)^T$ and $y = (y_1, \dots, y_n)^T$ are the input and output vectors of the fuzzy system. The MIMO fuzzy system is define as

$$f(x) = \ominus^T \varepsilon(x) \quad (32)$$

Where

$$\ominus^T = (\theta_1, \dots, \theta_m)^T = \begin{bmatrix} \theta_1^1, \theta_1^2, \dots, \theta_1^M \\ \theta_2^1, \theta_2^2, \dots, \theta_2^M \\ \vdots \\ \theta_m^1, \theta_m^2, \dots, \theta_m^M \end{bmatrix} \quad (33)$$

$\varepsilon(x) = (\varepsilon^1(x), \dots, \varepsilon^M(x))^T$, $\varepsilon^1(x) = \prod_{i=1}^{n} \mu_{A_i^l}(x_i) / \sum_{l=1}^{M} (\prod_{i=1}^{n} \mu_{A_i^l}(x_i))$, and $\mu_{A_i^l}(x_i)$ is defined in (24). To reduce the number of fuzzy rules, we divide the fuzzy system in to three parts:

$$F^1(q,\dot{q}) = \ominus^{1^T} \varepsilon(q,\dot{q}) = \left[\theta_1^{1^T} \varepsilon(q,\dot{q}), \dots, \theta_m^{1^T} \varepsilon(q,\dot{q})\right]^T \quad (34)$$

$$F^2(q,\ddot{q}_r) = \ominus^{2^T} \varepsilon(q,\ddot{q}_r) = \left[\theta_1^{2^T} \varepsilon(q,\ddot{q}_r), \dots, \theta_m^{2^T} \varepsilon(q,\ddot{q}_r)\right]^T \quad (35)$$

$$F^3(q,\ddot{q}) = \ominus^{3^T} \varepsilon(q,\ddot{q}) = \left[\theta_1^{3^T} \varepsilon(q,\dot{q}), \dots, \theta_m^{3^T} \varepsilon(q,\ddot{q})\right]^T \quad (36)$$

The control input is given by

$$\tau = M^{\wedge} \ddot{q}_r + P_m(\theta) + P_{net}(\theta) + F^1(q,\dot{q}) + F^2(q,\ddot{q}_r) + F^3(q,\ddot{q}) - K_D s - W sgn(s) \quad (37)$$

Where $M^{\wedge}$, $P_m(\theta) + P_{net}(\theta)$ are the estimations of $M(q)$ and and are positive constants; $W = diag\ [W_1, \dots, W_m]$ and $W_1, \dots, W_m$ are positive constants. The adaptation law is given by

$$\dot{\theta}_j^1 = -\Gamma_{1j} s_j \varepsilon(q,\dot{q})$$
$$\dot{\theta}_j^2 = -\Gamma_{2j} s_j \varepsilon(q,\ddot{q}_r) \quad (38)$$

$$\dot{\theta}_j^3 = -\Gamma_{3j} s_j \varepsilon(q, \ddot{q})$$

Where $j = 1, \dots, m$ and $\Gamma_{1j} - \Gamma_{3j}$ are positive diagonal matrices.
The Lyapunov function candidate is presented as

$$V = \frac{1}{2} s^T M s + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \emptyset_j^1 + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \emptyset_j^2 + \frac{1}{2} \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{13^T} \emptyset_j^3 \qquad (39)$$

Where $\emptyset_j^1 = \emptyset_j^{1^*} - \emptyset_j^1, \emptyset_j^2 = \emptyset_j^{2^*} - \emptyset_j^2$ and $\emptyset_j^3 = \emptyset_j^{3^*} - \emptyset_j^3$ we define

$$F(q, \dot{q}, \ddot{q}_r, \ddot{q}) = F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q}) \qquad (40)$$

From (22) and (23), we get

$$M(q)\ddot{q} + P_m(\theta) + P_{net}(\theta) = M^{\wedge} \ddot{q}_r + \overline{P_m(\theta) + P_{net}(\theta)} + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) - K_D s - W s \qquad (41)$$

Since $\dot{q}_r = \dot{q} - s$ and $\ddot{q}_r = \ddot{q} - \dot{s}$, we get

$$M\dot{s} + (P_m(\theta) + P_{net}(\theta) + K_D)s + Wsgn(s) = -\Delta F + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) \qquad (42)$$

Then $M\dot{s} + P_m(\theta) + P_{net}(\theta)s$ can be written as

$$M\dot{s} + P_m(\theta) + P_{net}(\theta)s = -\Delta F + F(q, \dot{q}, \ddot{q}_r, \ddot{q}) - K_D s - Wsgn(s) \qquad (43)$$

Where $\Delta F = \tilde{M}\ddot{q}_r + P_m(\theta) + P_{net}(\theta)$, $\tilde{M} = M - M^{\wedge}$, $\tilde{C}_1 = P_m(\theta) + P_{net}(\theta) - \overline{P_m(\theta) + P_{net}(\theta)}$ The derivative of V is

$$\dot{V} = s^T M \dot{s} + \frac{1}{2} s^T \dot{M} s + \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \dot{\emptyset}_j^1 + \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \dot{\emptyset}_j^2 + \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{13^T} \dot{\emptyset}_j^3 \qquad (44)$$

We know that $s^T M \dot{s} + \frac{1}{2} s^T \dot{M} s = s^T (M\dot{s} + P_m(\theta) + P_{net}(\theta)s)$ from (44). Then

$$\dot{V} = -s^T[-K_D s + Wsgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \dot{\emptyset}_j^1 + \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \dot{\emptyset}_j^2 + \sum_{j=1}^{m} \frac{1}{\Gamma} \qquad (45)$$

We define the minimum approximation error as

$$\omega = \Delta F - [F^1(q, \dot{q} | \ominus^{1^*}) + F^2(q, \ddot{q}_r | \ominus^{2^*}) + F^3(q, \ddot{q} | \ominus^{3^*})] \qquad (46)$$

We plug (51) in to (52)

$$\dot{V} = -s^T[-K_D s + Wsgn(s) + \Delta F - F(q, \dot{q}, \ddot{q}_r, \ddot{q})] + \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \dot{\emptyset}_j^1 + \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \dot{\emptyset}_j^2 + \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{13^T} \dot{\emptyset}_j^3$$

$$= -s^T[-K_D s + Wsgn(s) + \omega + F^1(q, \dot{q} | \ominus^{1^*}) + F^2(q, \ddot{q}_r | \ominus^{2^*}) + F^3(q, \ddot{q} | \ominus^{3^*}) - F^1(q, \dot{q}) + F^2(q, \ddot{q}_r) + F^3(q, \ddot{q})] + \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \dot{\emptyset}_j^1 + \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \dot{\emptyset}_j^2 + \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{13^T} \dot{\emptyset}_j^3$$

$$= -s^T K_D s - s^T Wsgn(s) - s^T \omega - \sum_{j=1}^{m} s_j \emptyset_j^{1^T} \varepsilon(q, \dot{q}) - \sum_{j=1}^{m} s_j \emptyset_j^{2^T} \varepsilon(q, \ddot{q}_r) - \sum_{j=1}^{m} s_j \emptyset_j^{3^T} \varepsilon \sum_{j=1}^{m} \frac{1}{\Gamma_{1j}} \emptyset_j^{1^T} \dot{\emptyset}_j^1 + \sum_{j=1}^{m} \frac{1}{\Gamma_{2j}} \emptyset_j^{2^T} \dot{\emptyset}_j^2 + \sum_{j=1}^{m} \frac{1}{\Gamma_{3j}} \emptyset_j^{3^T} \dot{\emptyset}_j^3$$

$$= -s^T K_D s - s^T Wsgn(s) - s^T \omega - \sum_{j=1}^{m} \emptyset_j^{1^T} (s_j \varepsilon(q, \dot{q}) - \frac{1}{\Gamma_{1j}} \dot{\emptyset}_j^1) - \sum_{j=1}^{m} \emptyset_j^{2^T} (s_j \varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{2j}} \dot{\emptyset}_j^2) - \sum_{j=1}^{m} \emptyset_j^{3^T} (s_j \varepsilon(q, \ddot{q}_r) - \frac{1}{\Gamma_{3j}} \dot{\emptyset}_j^3)$$

$$= -s^T K_D s - s^T W sgn(s) - s^T \omega - \sum_{j=1}^{m} \phi_j^{1^T} \left(s_j\, \varepsilon(q, \dot{q}) + \frac{1}{\Gamma_{1j}} \dot{\phi}_j^1\right) - \sum_{j=1}^{m} \phi_j^{2^T} \left(s_j \varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{2j}} \dot{\phi}_j^2\right) - \sum_{j=1}^{m} \phi_j^{3^T} \left(s_j\, \varepsilon(q, \ddot{q}_r) + \frac{1}{\Gamma_{3j}} \dot{\phi}_j^3\right)$$

Then $\dot{V}$ becomes

$$\dot{V} = -s^T K_D s - s^T W sgn(s) - s^T \omega \qquad \textbf{(47)}$$

$$= -\sum_{j=1}^{m} \left(s_j^2 K_{Dj} + W_j |s_j| + s_j \omega_j\right)$$

$$= -\sum_{j=1}^{m} [\, s_j \,(s_j K_{Dj} + \omega_j) + W_j |s_j| \,]$$

Since ω_j can be as small as possible, we can find K_{Dj} that $|s_j^2 K_{Dj}| > |\omega_j| (s_j \neq 0)$.
Therefore, we can get $s_j(s_j K_{Dj} + \omega_j) > 0$ for $s_j \neq 0$ and $\dot{V} < 0 \ (s \neq 0)$. Figure 6 is shown the fuzzy estimator variable structure.

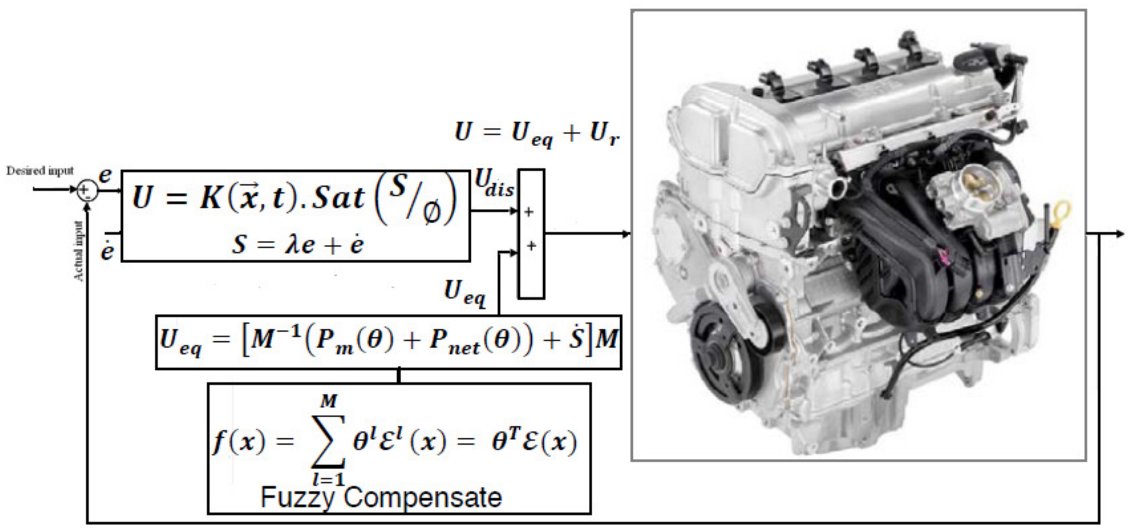


FIGURE 6: Chattering free Block diagram of a fuzzy estimator variable structure controller

Third step is focused on design Mamdani's fuzzy [30-40] backstepping adaptive fuzzy estimator variable structure. As mentioned above pure variable structure controller has nonlinear dynamic equivalent limitations in presence of uncertainty and external disturbances in order to solve these challenges this work applied Mamdani's fuzzy inference engine estimator in variable structure controller. However proposed MIMO fuzzy estimator variable structure has satisfactory performance but calculate the variable structure surface slope by try and error or experience knowledge is very difficult, particularly when system has structure or unstructured uncertainties; MIMO Mamdani's fuzzy backstepping variable structure function fuzzy estimator variable structure controller is recommended. The backstepping method is based on mathematical formulation which this method is introduced new variables into it in form depending on the dynamic equation of IC engine. This method is used as feedback linearization in order to solve nonlinearities in the system. To use of nonlinear fuzzy filter this method in this research makes it possible to create dynamic nonlinear backstepping estimator into the adaptive fuzzy estimator variable structure process to eliminate or reduce the challenge of uncertainty in this part. The backstepping controller is calculated by;

$$U_{B.S} = U_{eq_{B.S}} + M.I \qquad \textbf{(48)}$$

Where $U_{B.S}$ is backstepping output function, $U_{eq_{B.S}}$ is backstepping nonlinear equivalent function which can be written as (55) and I is backstepping control law which calculated by (49)

$$U_{eq_{B.S}} = [(P_m(\theta) + P_{net}(\theta))] \quad (49)$$
$$I = [\ddot{\theta} + K_1(K_1 - 1) \cdot e + (K_1 + K_2) \cdot \dot{e}] \quad (50)$$

Based on (10) and (47) the fuzzy backstepping filter is considered as

$$(P_m(\theta) + P_{net}(\theta)) = \sum_{l=1}^{M} \theta^T \zeta(x) - \lambda S - K \quad (51)$$

Based on (48) the formulation of fuzzy backstepping filter can be written as;

$$U = U_{eqB.S_{fuzzy}} + MI \quad (52)$$

Where $U_{eq_{B.Sfuzzy}} = [(P_m(\theta) + P_{net}(\theta))] + \sum_{l=1}^{M} \theta^T \zeta(x) + K$

The adaption low is defined as

$$\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j) \quad (53)$$

where the γ_{sj} is the positive constant and $\zeta_j(S_j) = [\zeta_j^1(S_j), \zeta_j^2(S_j), \zeta_j^3(S_j), \ldots\ldots, \zeta_j^M(S_j)]^T$

$$\zeta_j^1(S_j) = \frac{\mu_{(A)}{}_j^l(S_j)}{\sum_i \mu_{(A)}{}_j^l(S_j)} \quad (54)$$

The dynamic equation of IC engine can be written based on the variable structure surface as;

$$M\dot{S} = -VS + M\dot{S} + VS \quad (55)$$

It is supposed that

$$S^T(\dot{M} - 2V)S = 0 \quad (56)$$

The derivation of Lyapunov function ($\dot{V}$) is written as

$$\dot{V} = \frac{1}{2} S^T \dot{M} S - S^T V S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j$$
$$= S^T(-\lambda S + \Delta f - K) + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j$$
$$= \sum_{j=1}^{m} [S_j(\Delta f_j - K_j)] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j$$
$$= \sum_{j=1}^{m} [S_j(\Delta f_j - \theta_j^T \zeta_j(S_j))] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j$$
$$= \sum_{j=1}^{m} [S_j(\Delta f_j - (\theta_j^*)^T \zeta_j(S_j) + \phi_j^T \zeta_j(S_j))] - S^T \lambda S + \sum \frac{1}{\gamma_{sj}} \phi_j^T \dot{\phi}_j$$
$$= \sum_{j=1}^{m} [S_j(\Delta f_j - ((\theta_j^*)^T \zeta_j(S_j))] - S^T \lambda S)] + \sum (\frac{1}{\gamma_{sj}} \phi_j^T [\gamma_{sj} \zeta_j(S_j) S_j + \dot{\phi}_j])$$

Where $\dot{\theta}_j = \gamma_{sj} S_j \zeta_j(S_j)$ is adaption law and $\dot{\phi}_j = -\dot{\theta}_j = -\gamma_{sj} S_j \zeta_j(S_j)$, consequently $\dot{V}$ can be considered by

$$\dot{V} = \sum_{j=1}^{m} [S_j \Delta f_j - ((\theta_j^*)^T \zeta_j(S_j))] - S^T \lambda S \quad (57)$$

The minimum error can be defined by

$$e_{mj} = \Delta f_j - ((\theta_j^*)^T \zeta_j(S_j)) \quad (58)$$

$\dot{V}$ is intended as follows

$$\dot{V} = \sum_{j=1}^{m} [S_j e_{mj}] - S^T \lambda S$$

$$\leq \sum_{j=1}^{m} |S_j||e_{mj}| - S^T\lambda S$$
$$= \sum_{j=1}^{m} |S_j||e_{mj}| - \lambda_j {S_j}^2$$
$$= \sum_{j=1}^{m} |S_j|(|e_{mj}| - \lambda_j S_j) \quad (59)$$

For continuous function $U_{eqB\_S_{fuzzy}}$ and suppose $\varepsilon > 0$ it is defined the fuzzy backstepping controller in form of (58) such that

$$Sup_{x \in U}\left|U_{eqB\,S_{fuzzy}} + MI\right| < \epsilon \quad (60)$$

As a result MIMO fuzzy backstepping adaptive fuzzy estimation variable structure is very stable which it is one of the most important challenges to design a controller with suitable response. Figure 7 is shown the block diagram of proposed MIMO fuzzy backstepping adaptive fuzzy estimation variable structure.

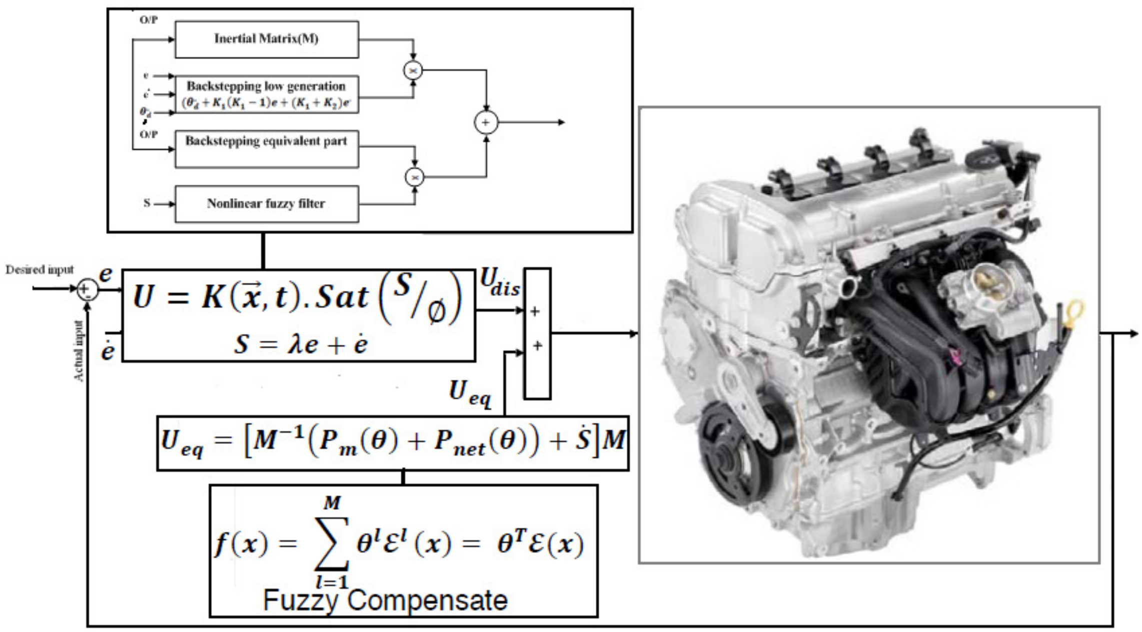


FIGURE 7: Chattering free Block diagram of a MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller

4. RESULTS

Variable structure controller (VSC) and proposed MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller were tested to sinus response trajectory. The simulation was implemented in Matlab/Simulink environment. Fuel ratio trajectory, disturbance rejection and error are compared in these controllers. It is noted that, these systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude. This type of noise is used to external disturbance in continuous and hybrid systems.

Fuel Ratio Trajectory: Figure 8 shows the fuel ratio in VSC and proposed MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller without disturbance for sinus trajectory.

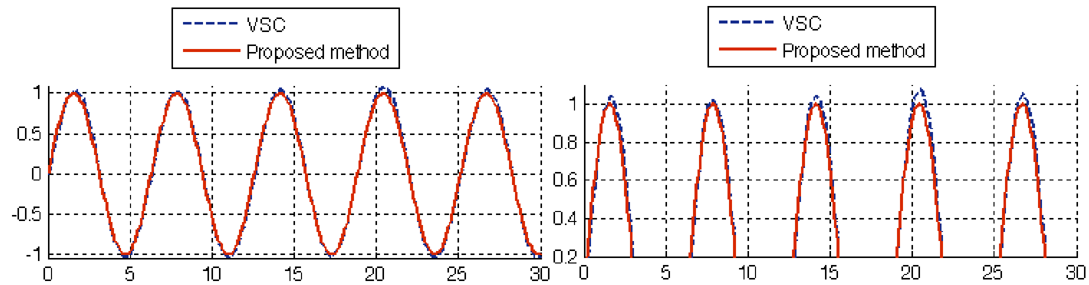


FIGURE 8: VSC Vs. MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller: fuel ratio

By comparing sinus response, Figure 8, in SMC and MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller, conversely the MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller 's overshoot **(0%)** is lower than VSC's **(3%)**.

Disturbance Rejection

Figure 9 is indicated the power disturbance removal in VSC and MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller. As mentioned before, VSC is one of the most important robust nonlinear controllers. Besides a band limited white noise with predefined of 40% the power of input signal is applied to the sinus VSC and MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller; it found slight oscillations in VSC trajectory responses.

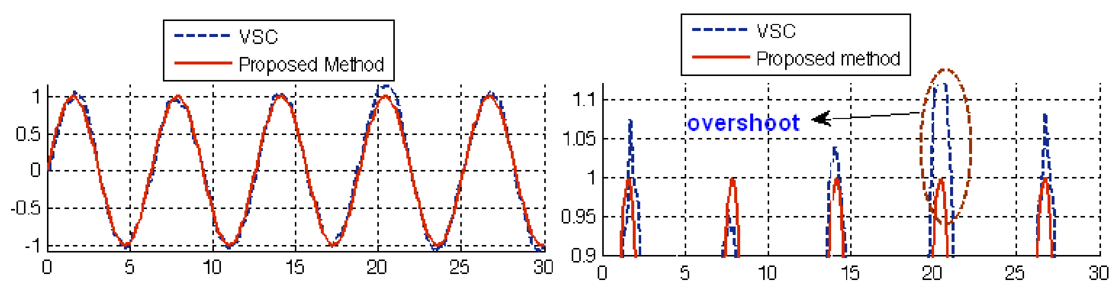


FIGURE 9: VSC Vs. MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller: fuel ratio with external disturbance

Among above graph, relating to sinus trajectory following with external disturbance, VSC has slightly fluctuations. By comparing overshoot; MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller 's overshoot **(0%)** is lower than VSC's **(12%).**

Errors in the Model: MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller has lower error rate (refer to Table.1), VSC has oscillation tracking which causes chattering phenomenon at the presence of disturbances. Figure 10 is shown steady state and RMS error in VSC and MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller in presence of external disturbance.

RMS Error Rate	**VSC**	**Proposed method**
Without Noise	**1e-3**	**0.6e-9**
With Noise	**0.01**	**0.1e-8**

TABLE 1: RMS Error Rate of Presented controllers

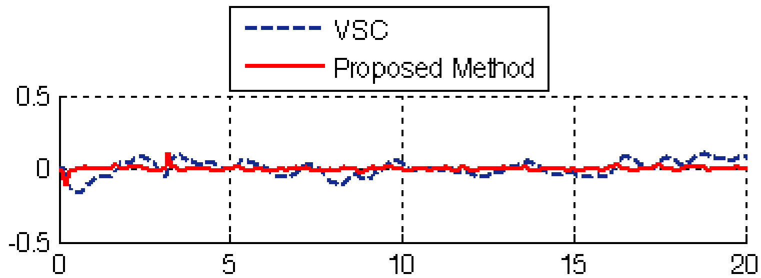


FIGURE 10: VSC Vs. MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller: Steady state error in presence of external disturbance

In these methods if integration absolute error (IAE) is defined by (67), table 2 is shown comparison between these two methods.

$$IAE = \int_0^\infty |e(t)|\, dt \tag{67}$$

Method	**VSC**	**Proposed Method**
IAE	**442.1**	**214.8**

TABLE 2: Calculate IAE

5. CONCLUSION

Refer to the research, a MIMO fuzzy backstepping adaptive fuzzy estimator variable structure controller design and application to IC engine has proposed in order to design high performance nonlinear controller in the presence of uncertainties, external disturbances and Lyapunov based. Regarding to the positive points in variable structure controller, fuzzy inference system and adaptive fuzzy backstepping methodology it is found that the adaptation laws derived in the Lyapunov sense. The stability of the closed-loop system is proved mathematically based on the Lyapunov method. The first objective in proposed method is removed the chattering which linear boundary layer method is used to solve this challenge. The second target in this work is compensate the model uncertainty by MIMO fuzzy inference system, in the case of the IC engine, if we define k_1 membership functions for each input variable, the number of fuzzy rules applied for each joint is K_1 which will result in a low computational load. In finally part fuzzy backstepping methodology with minimum rule base is used to online tuning and adjusted the fuzzy variable structure method and eliminates the chattering with minimum computational load. In this case the performance is improved by using the advantages of variable structure, artificial intelligence compensate method and adaptive algorithm while the disadvantages removed by added each method to previous method.

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] 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.

[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] 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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

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

[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] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] Heywood, J., “Internal Combustion Engine Fundamentals”, McGraw-Hill, New York, 1988.

[64] Ferguson, C., “Internal Combustion Engines: Applied Thermosciences”, John Wiley & Sons, Inc., New York, 2001.

[65] Guzzella, L., “Introduction to Modeling and Control of Internal Combustion Engine Systems” Springer, New York, 2004.

[66] Ramos, J., “Internal Combustion Engine Modeling”, Hemisphere Publishing corporation, New York, 1989.

[67] Blair, G., “Design and Simulation of Four Stroke Engines”, Society of Automotive Engineers, Warrendale, Pa, 1999.

[68] G. Zhu, et al, "Closed-Loop Ignition Timing Control for SI Engines Using Ionization Current Feedback," IEEE Trans on Control Systems, pp. 416-427, May 2007.

[69] I. Haskara, et al, "On Combustion Invariants For MBT Timing Estimation and Control," in ASME Internal Combustion Engine Division, 2004.

[70] Dawson, J., “An experimental and Computational Study of Internal Combustion Engine Modeling for Controls Oriented Research” Ph.D. Dissertation, The Ohio State University, 2005.

[71] Lee, B., “Methodology for the Static and Dynamic Model Based Engine Calibration and Optimization” Ph.D. Dissertation, The Ohio State University, 2005.

On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory

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

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

Atefeh Gavahian *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

Samaneh Roosta *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

Samira Soltani *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

Classical sliding mode controller is robust to model uncertainties and external disturbances. A sliding mode control method with a switching control low guarantees asymptotic stability of the system, but the addition of the switching control law introduces chattering in to the system. One way of attenuating chattering is to insert a saturation function inside of a boundary layer around the sliding surface. Unfortunately, this addition disrupts Lyapunov stability of the closed-loop system. Classical sliding mode control method has difficulty in handling unstructured model uncertainties. One can overcome this problem by combining a sliding mode controller and fuzzy system together. Fuzzy rules allow fuzzy systems to approximate arbitrary continuous functions. To approximate a time-varying nonlinear system, a fuzzy system requires a large amount of fuzzy rules. This large number of fuzzy rules will cause a high computation load. The addition of an adaptive law to a fuzzy sliding mode controller to online tune the parameters of the fuzzy rules in use will ensure a moderate computational load. Refer to this research; tuning methodology can online adjust both the premise and the consequence parts of the fuzzy rules. Since this algorithm for is specifically applied to a robot manipulator.

Keywords: Classical Sliding Mode Controller, Robust, Uncertainties, Chattering Phenomenon, Lyapunov Theory, Fuzzy Sliding Mode Controller, Tuning Fuzzy Sliding Mode Controller, Robotic System.

1. INTRODUCTION AND MOTIVATION

Sliding mode controller (SMC) is one of the influential nonlinear controllers in certain and uncertain systems which are used to present a methodical solution for two main important controllers' challenges, which named: stability and robustness. Conversely, this controller is used in different applications; sliding mode controller has consequent disadvantages ; chattering phenomenon and nonlinear equivalent dynamic formulation; which chattering is caused to some difficulties such as saturation and heat for mechanical parts of robot manipulators or drivers and nonlinear equivalent dynamic formulation in uncertain systems is most important challenge in highly nonlinear uncertain system[1, 5-29]. In order to solve the chattering in the systems output, boundary layer method should be applied so beginning able to recommended model in the main motivation which in this method the basic idea is replace the discontinuous method by saturation (linear) method with small neighbourhood of the switching surface. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode with boundary layer to improve the industry application [22]. R. Palm has presented a fuzzy method to nonlinear approximation instead of linear approximation inside the boundary layer to improve the chattering and control the result performance[23]. Moreover, C. C. Weng and W. S. Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. To remove the above setbacks, control researchers have applied artificial intelligence method (e.g., fuzzy logic, neural network and genetic algorithm) in nonlinear robust controller (e.g., sliding mode controller, backstepping and feedback linearization) besides this technique is very useful in order to implement easily. Estimated uncertainty method is used in term of uncertainty estimator to compensation of the system uncertainties. It has been used to solve the chattering phenomenon and also nonlinear equivalent dynamic. If estimator has an acceptable performance to compensate the uncertainties, the chattering is reduced. Research on estimated uncertainty to reduce the chattering is significantly growing as their applications such as industrial automation and robot manipulator. For instance, the applications of artificial intelligence, neural networks and fuzzy logic on estimated uncertainty method have been reported in [22-30]. Wu et al. [30] have proposed a simple fuzzy estimator controller beside the discontinuous and equivalent control terms to reduce the chattering. Elmali et al. [27]and Li and Xu [29]have addressed sliding mode control with perturbation estimation method (SMCPE) to reduce the classical sliding mode chattering.

In recent years, artificial intelligence theory has been used in sliding mode control systems [31-40, 68]. 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 [33-38]. The main reasons to use fuzzy logic technology are able to give approximate recommended solution for uncertain and complex 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. The applications of artificial intelligence, neural networks and fuzzy logic, on robot arm control have reported in [29-31]. 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. The applications of fuzzy logic on sliding mode controller have reported in [11-16, 23-30]. 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 [42-47]. 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 equation. 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 (MIMO) FSMC reduces the chattering phenomenon and reconstructs the approximate the unknown system has been presented for a robot manipulator [48]. 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 [10-15]; [49-55]. Lhee et al. [49]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. However the application of FSMC and SMFC are growing but the main SMFC drawback compared to FSMC is calculation the value of sliding surface λ pri-defined very carefully. Moreover, the advantages of SMFC compared to FLC reduce the number of fuzzy rule base and increase the robustness and stability. At last FSMC compare to the SMFC is more suitable for implementation action.

In various dynamic parameters systems that need to be training on-line tuneable gain control methodology is used. On-line tuneable 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, on-line tuneable method is applied to artificial sliding mode controller. 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, on-line control technique can train the dynamic parameter to have satisfactory performance. Calculate sliding surface slope is common challenge in classical sliding mode controller and fuzzy sliding mode controller. Research on adaptive (on-line tuneable) fuzzy control is significantly growing, for instance, different adaptive fuzzy controllers have been reported in [40, 56-68]. The adaptive sliding mode controller is used to estimate the unknown dynamic parameters and external disturbances. The applications of adaptive fuzzy sliding mode controller to control the robot manipulators have been reported in [11-20, 67]. Yoo and Ham [58]have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. In $n-DOF$ robot manipulator with k membership function for each input variable, the number of fuzzy rules for each joint is equal to $3k^{2n}$ 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 K_2 membership functions, the number of fuzzy rules for each joint is $(m+1)K_2^m+K_2$. 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 [51]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 reduce. 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. Piltan et al., have proposed a adaptive fuzzy inference sliding mode controller to reduce or eliminate chattering for robot manipulator [67].
This paper is organized as follows:

In section 2, detail of dynamic equation of robot arm, problem statements, objectives and introduced the sliding mode controller are presented. Detail of tuning methodology which online adjusted both the premise and the consequence parts of the fuzzy rules is presented in section 3. In section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. ROBOT MANIPULATOR DYNAMICS, OBJECTIVES, PROBLEM STATEMENTS AND SLIDING MODE METHODOLOGY

Robot manipulator dynamic: 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[1]. 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 nonlinear robust controller design is major subject in this work. The equation of a multi degrees of freedom (DOF) robot manipulator is calculated by the following equation[6]:

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

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 1 if vector of nonlinearity term derive as Centrifugal, Coriolis and Gravity terms, as a result robot manipulator dynamic equation can also be written as [10-16]:

$$N(q,\dot{q}) = V(q,\dot{q}) + G(q) \quad \textbf{(2)}$$

$$V(q,\dot{q}) = B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 \quad \textbf{(3)}$$

$$\tau = M(q)\ddot{q} + B(q)[\dot{q}\,\dot{q}] + C(q)[\dot{q}]^2 + G(q) \quad \textbf{(4)}$$

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: $[\dot{q}_1 \cdot \dot{q}_2, \dot{q}_1 \cdot \dot{q}_3, \dots, \dot{q}_1 \cdot \dot{q}_n, \dot{q}_2 \cdot \dot{q}_3, \dots]^T$, and $[\dot{q}]^2$ is vector, that it can given by: $[\dot{q}_1^{\,2}, \dot{q}_2^{\,2}, \dot{q}_3^{\,2}, \dots]^T$. In robot manipulator dynamic part the inputs are torques and the outputs are actual displacements, as a result in (4) it can be written as [1, 6, 10-16];

$$\ddot{q} = M^{-1}(q).\{\tau - N(q,\dot{q})\} \quad \textbf{(5)}$$

To implementation (5) 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;

$$M(\theta) = m_1 J_{v1}^T J_{v1} + J_{\omega1}^{TC1} I_1 J_{\omega1} + m_2 J_{v2}^T J_{v2} + J_{\omega2}^{TC2} I_2 J_{\omega2} + m_3 J_{v3}^T J_{v3} + J_{\omega3}^{TC3} I_3 J_{\omega3} + m_4 J_{v4}^T J_{v4} - m_5 J_{v5}^T J_{v5} + J_{\omega5}^{TC5} I_5 J_{\omega5} + m_6 J_{v6}^T J_{v6} + J_{\omega6}^{TC6} I_6 J_{\omega6} \quad (6)$$

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]

$$M(q) = \begin{bmatrix} M_{11} & M_{12} & \dots & \dots & \dots & M_{1n} \\ M_{21} & \dots & \dots & \dots & \dots & M_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ M_{n.1} & \dots & \dots & \dots & \dots & M_{n.n} \end{bmatrix} \quad (7)$$

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

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & \dots & b_{11n} & b_{123} & \dots & b_{12n} & \dots & \dots & b_{1.n-1.n} \\ b_{212} & \dots & \dots & b_{21n} & b_{223} & \dots & \dots & \dots & \dots & b_{2.n-1.n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{n.1.2} & \dots & \dots & b_{n.1.n} & \dots & \dots & \dots & \dots & \dots & b_{n.n-1.n} \end{bmatrix} \quad (8)$$

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

$$C(q) = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \quad (9)$$

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

$$G(q) = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \quad (10)$$

Sliding mode controller (SMC): SMC is a powerful nonlinear controller which has been analyzed by many researchers especially in recent years. This theory was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices [2]. The main reason to opt for this controller is its acceptable control performance in wide range and solves two most important challenging topics in control which names, stability and robustness [7, 17-20]. Sliding mode control theory for control of robot manipulator was first proposed in 1978 by Young to solve the set point problem ($\dot{q}_d = 0$) by discontinuous method in the following form [19, 3];

$$\tau_{(q,t)} = \begin{cases} \tau_i^+(q,t) & if \ S_i > 0 \\ \tau_i^-(q,t) & if \ S_i < 0 \end{cases} \quad (11)$$

where S_i is sliding surface (switching surface), $i = 1, 2, \dots, n$ for *n*-DOF robot manipulator, $\tau_i(q,t)$ is the i^{th} torque of joint. Sliding mode controller is divided into two main sub controllers: discontinues controller(τ_{dis}) and equivalent controller(τ_{eq}).

Robot manipulators are one of the highly nonlinear and uncertain systems which caused to needed to robust controller. This section provides introducing the formulation of sliding mode controller to robot manipulator based on [1, 6]Consider a nonlinear single input dynamic system of the form [6]:

$$x^{(n)} = f(\vec{x}) + b(\vec{x})u \quad (12)$$

Where u is the vector of control input, $x^{(n)}$ is the n^{th} derivation of x, $x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T$ is the state vector, $f(x)$ is unknown or uncertainty, and $b(x)$ is of known *sign* function. The control problem is truck to the desired state; $x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d^{(n-1)}]^T$, and have an acceptable error which is given by:

$$\tilde{x} = x - x_d = [\tilde{x}, \dots, \tilde{x}^{(n-1)}]^T \quad (13)$$

A time-varying sliding surface $s(x, t)$ is given by the following equation:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\tilde{x} = 0 \quad (14)$$

where λ is the positive constant. To further penalize tracking error integral part can be used in sliding surface part as follows:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\left(\int_0^t \tilde{x}\,dt\right) = 0 \quad (15)$$

The main target in this methodology is kept the sliding surface slope $s(x, t)$ near to the zero. Therefore, one of the common strategies is to find input U outside of $s(x, t)$.

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)| \quad (16)$$

where ζ is positive constant and in equation (16) forces tracking trajectories is towards sliding condition.

$$\text{If } \mathbf{S(0)>0} \rightarrow \frac{d}{dt}S(t) \le -\zeta \quad (17)$$

To eliminate the derivative term, it is used an integral term from t=0 to t=t_{reach}

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt}S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \quad (18)$$

Where t_{reach} is the time that trajectories reach to the sliding surface so, suppose S($t_{reach} = 0$) defined as

$$0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta} \quad (19)$$

and

$$if\ S(0) < 0 \rightarrow 0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta} \quad (20)$$

Equation (20) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0 \quad (21)$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt} + \lambda)\quad \tilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \quad (22)$$

The derivation of S, namely, $\dot{S}$ can be calculated as the following;

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \quad (23)$$

suppose the second order system is defined as;

$$\ddot{x} = f + u \rightarrow \dot{S} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \quad (24)$$

Where f is the dynamic uncertain, and also since $S = 0$ *and* $\dot{S} = 0$, to have the best approximation ,$\hat{U}$ is defined as

$$\hat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \quad (25)$$

A simple solution to get the sliding condition when the dynamic parameters have uncertainty is the switching control law:

$$U_{dis} = \hat{U} - K(\vec{x},t) \cdot \mathrm{sgn}(s) \quad \textbf{(26)}$$

where the switching function $\mathrm{sgn}(S)$ is defined as

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \quad \textbf{(27)}$$

and the $K(\vec{x},t)$ is the positive constant. Suppose by (26) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S} \cdot S = \left[f - \hat{f} - K\mathrm{sgn}(s)\right] \cdot S = \left(f - \hat{f}\right) \cdot S - K|S| \quad \textbf{(28)}$$

and if the equation (27) instead of (28) the sliding surface can be calculated as

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^2 \left(\int_0^t \tilde{x}\, dt\right) = (\dot{\mathrm{x}} - \dot{\mathrm{x}}_{\mathrm{d}}) + 2\lambda(\dot{\mathrm{x}} - \dot{\mathrm{x}}_{\mathrm{d}}) - \lambda^2(\mathrm{x} - \mathrm{x}_{\mathrm{d}}) \quad \textbf{(29)}$$

in this method the approximation of U is computed as

$$\hat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{\mathrm{x}} - \dot{\mathrm{x}}_{\mathrm{d}}) + \lambda^2(\mathrm{x} - \mathrm{x}_{\mathrm{d}}) \quad \textbf{(30)}$$

Therefore the switching function $\mathrm{sgn}(S)$ is added to the control law as

$$U = K(\vec{x},t) \cdot \mathrm{Sgn}(S) \quad \textbf{(31)}$$

Based on above discussion, the control law for a multi degrees of freedom robot manipulator is written as:

$$\tau = \tau_{eq} + \tau_{dis} \quad \textbf{(32)}$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems and τ_{eq} can be calculate as follows:

$$\tau_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \quad \textbf{(33)}$$

and τ_{dis} is computed as;

$$\tau_{dis} = K \cdot \mathrm{Sgn}(S) \quad \textbf{(34)}$$

the control output can be written as;

$$\tau = \tau_{eq} + K.\mathrm{Sgn}(S) \quad \textbf{(35)}$$

Figure 1 shows the position classical sliding mode control for robot manipulator. By (34) and (35) the sliding mode control of robot manipulator is calculated as;

$$\tau = \left[M^{-1}(B + C + G) + \dot{S}\right]M + K \cdot \mathrm{Sgn}(S) \quad \textbf{(36)}$$

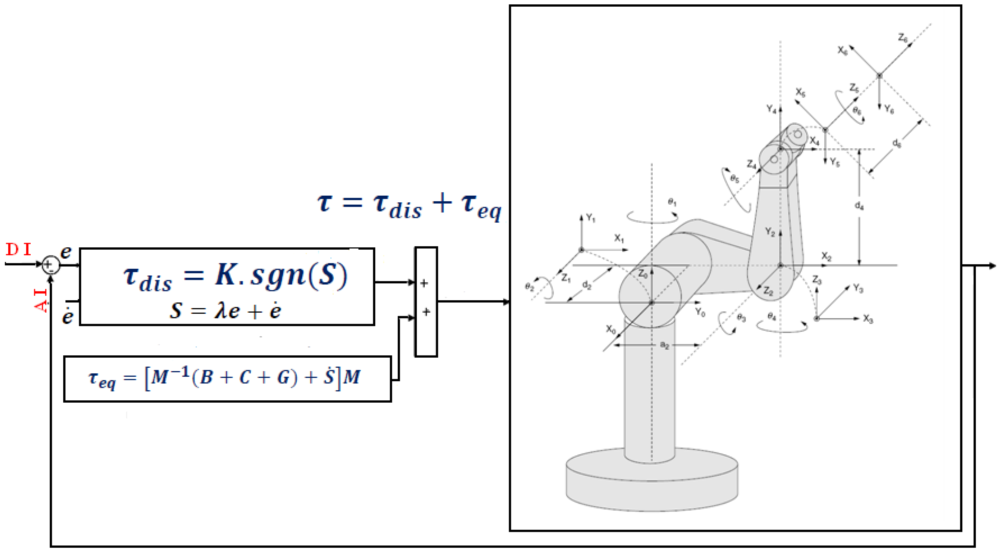


FIGURE 1: Diagram of classical sliding mode controller [3, 9-16, 64-67]

The Lyapunov formulation can be written as follows,

$$V = \frac{1}{2} S^T . M . S \tag{37}$$

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2} S^T . \dot{M} . S + S^T \; M\dot{S} \tag{38}$$

the dynamic equation of robot manipulator can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + VS + G - \tau \tag{39}$$

it is assumed that

$$S^T\left(\dot{M} - 2V\right)S = 0 \tag{40}$$

by substituting (39) in (38)

$$\dot{V} = \frac{1}{2} S^T \dot{M} S - S^T VS + S^T\left(M\dot{S} + VS + G - \tau\right) = S^T\left(M\dot{S} + VS + G - \tau\right) \tag{41}$$

suppose the control input is written as follows

$$\hat{\tau} = \widehat{\tau_{eq}} + \widehat{\tau_{dis}} = \left[\widehat{M}^{-1}\left(\widehat{V} + \widehat{G}\right) + \dot{S}\right]\widehat{M} + K.sgn(S) + K_v S \tag{42}$$

by replacing the equation (42) in (41)

$$\dot{V} = S^T(M\dot{S} + VS + G - \widehat{M}\dot{S} - \widehat{V}S - \widehat{G} - K_v S - Ksgn(S) = S^T\left(\widetilde{M}\dot{S} + \widetilde{V}S + \widetilde{G} - K_v S - Ksgn\right. \tag{43}$$

it is obvious that

$$\left|\widetilde{M}\dot{S} + \widetilde{V}S + \widetilde{G} - K_v S\right| \le \left|\widetilde{M}\dot{S}\right| + \left|\widetilde{V}S\right| + \left|\widetilde{G}\right| + \left|K_v S\right| \tag{44}$$

the Lemma equation in robot manipulator system can be written as follows

$$K_u = \left[\left|\widetilde{M}\dot{S}\right| + \left|VS\right| + \left|G\right| + \left|K_v S\right| + \eta\right]_i, i = 1, 2, 3, 4, \ldots \tag{45}$$

the equation (40) can be written as

$$K_u \ge \left|\left[\widetilde{M}\dot{S} + VS + G - K_v S\right]_i\right| + \eta_i \tag{46}$$

therefore, it can be shown that

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i |S_i| \tag{47}$$

Consequently the equation (47) guaranties the stability of the Lyapunov equation

Problem statements: Even though, sliding mode controller is used in wide range areas but, pure it has chattering problem and nonlinear dynamic part challenges. On the other hand, fuzzy logic controller has been used for nonlinear and uncertain systems controlling. Conversely pure fuzzy logic controller (FLC) works in many areas, it cannot guarantee the basic requirement of stability and acceptable performance[8]. Although both SMC and FLC have been applied successfully in many applications but they have some limitations. The boundary layer method is used to reduce or eliminate the chattering and proposed fuzzy Lyapunov estimator method focuses on substitution fuzzy logic system instead of dynamic nonlinear equation to implement easily and avoid mathematical model base controller. To reduce the effect of uncertainty in proposed method, novel antecedent and consequent adaptive method is applied to fuzzy sliding mode controller in robot manipulator.

Objectives: The main goal is to design a novel fuzzy adaptive fuzzy estimation sliding mode methodology which applied to robot manipulator with easy to design and implement. Robot manipulator has nonlinear dynamic and uncertain parameters consequently; following objectives have been pursuit in the mentioned research: To develop a chattering in a position pure variable structure controller against uncertainties, to design and implement a Lyapunov fuzzy structure variable controller in order to solve the equivalent problems with minimum rule base and finally to develop a position fuzzy (antecedent and consequent) adaptive fuzzy estimation sliding mode controller in order to solve the disturbance rejection and reduce the computation load.

3. METHODOLOGY: DESIGN A NOVEL FUZZY (ANTECEDENT AND CONSEQUENT) ADAPTIVE FUZZY ESTIMATION SLIDING MODE CONTROLLER

First part is focused on eliminate the oscillation (chattering) in pure SMC based on linear boundary layer method. To reduce or eliminate the chattering it is used the boundary layer method; in boundary layer method the basic idea is replace the discontinuous method by saturation (linear) method with small neighborhood of the switching surface. This replace is caused to increase the error performance [20-24].

$$B(t) = \{x, |S(t)| \le \emptyset\}; \emptyset > 0 \tag{48}$$

Where $\emptyset$ is the boundary layer thickness. Therefore, to have a smote control law, the saturation function $Sat(S/\emptyset)$ added to the control law:

$$U = K(\vec{x},t).Sat\left(S/\emptyset\right) \tag{49}$$

Where $Sat\left(S/\emptyset\right)$ can be defined as

$$sat\left(S/\emptyset\right) = \begin{cases} 1 & (S/\emptyset > 1) \\ -1 & (S/\emptyset < 1) \\ S/\emptyset & (-1 < S/\emptyset < 1) \end{cases} \tag{50}$$

Based on above discussion, the control law for a robot manipulator is written as [10-24]:

$$U = U_{eq} + U_r \tag{51}$$

Where, the model-based component U_{eq} is the nominal dynamics of systems and U_{eq} can be calculate as follows:

$$U_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{52}$$

and U_{sat} is computed as;

$$U_{sat} = K \cdot \mathrm{sat}\left(S/\emptyset\right) \tag{53}$$

the control output can be written as;

$$U = U_{eq} + K.\mathrm{sat}\left({S}/{\emptyset}\right) = \begin{cases} U_{eq} + K.\mathrm{sgn}(S) & ,|S| \geq \emptyset \\ U_{eq} + K.{S}/{\emptyset} & ,|S| < \emptyset \end{cases} \quad \textbf{(54)}$$

Figure 2 is shown classical variable structure which eliminates the chattering using linear boundary layer method.

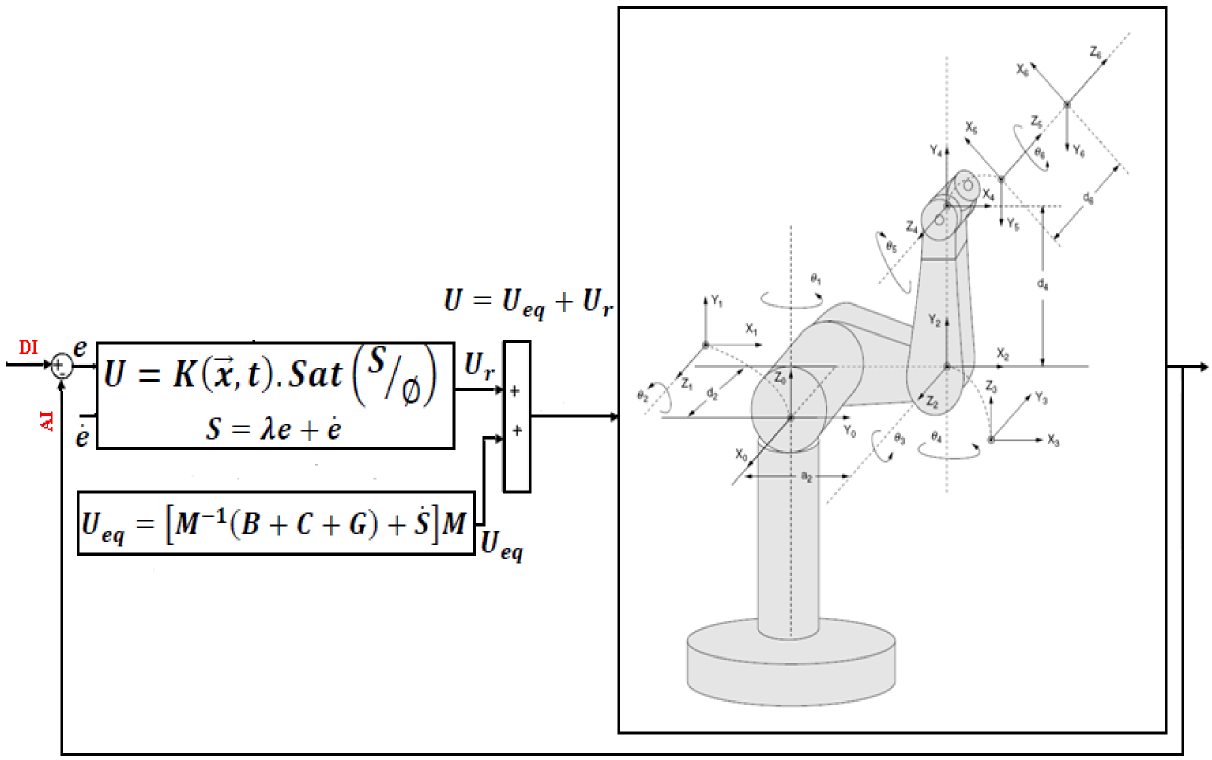


FIGURE 2: Chattering free Block diagram of a SMC: applied to robot manipulator

Second step is focused on design fuzzy estimation variable structure based on Lyapunov formulation. The firs type of fuzzy systems is given by

$$f(x) = \sum_{l=1}^{M} \theta^l \varepsilon^l(x) = \theta^T \varepsilon(x) \quad \textbf{(55)}$$

Where $\theta = (\theta^1, \dots, \theta^M)^T, \varepsilon(x) = (\varepsilon^1(x), \dots, \varepsilon^M(x))^T, and\ \varepsilon^l(x) =: \prod_{i=1}^{n} \frac{\mu_{A_i^l}(x_i)}{\sum_{l=1}^{M}(\prod_{i=1}^{n} \mu_{A_i^l}(x_i))}$. $\theta^1, \dots, \theta^M$ are adjustable parameters in (55) . $\mu_{A_1^1}(x_1), \dots, \mu_{A_n^m}(x_n)$ are given membership functions whose parameters will not change over time.

The second type of fuzzy systems is given by

$$f(x) = \frac{\sum_{l=1}^{M} \theta^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \alpha_i^l}{\delta_i^l}\right)^2\right)\right]} \quad \textbf{(56)}$$

Where θ^l , α_i^l *and* δ_i^l are all adjustable parameters. From the universal approximation theorem, we know that we can find a fuzzy system to estimate any continuous function. For the first type of fuzzy systems, we can only adjust θ^l in (55). We define $f^{\wedge}(x|\theta)$ as the approximator of the real function $f(x)$.

$$f^{\wedge}(x|\theta) = \theta^T \varepsilon(x) \quad \textbf{(57)}$$

We define θ^* as the values for the minimum error:

$$\theta^* = \arg\min_{\theta\in\Omega}\left[\sup_{x\in U} |f^{\wedge}(x|\theta) - g(x)|\right] \quad \textbf{(58)}$$

Where Ω is a constraint set for θ. For specific x ,$sup_{x\in U}|f^{\wedge}(x|\theta^*) - f(x)|$ is the minimum approximation error we can get.

We used the first type of fuzzy systems (56) to estimate the nonlinear system (20) the fuzzy formulation can be write as below;

$$f(x|\theta) = \theta^T \varepsilon(x) = \frac{\sum_{l=1}^{n} \theta^l [\mu_{A^l}(x)]}{\sum_{l=1}^{n} [\mu_{A^l}(x)]} \quad \textbf{(59)}$$

Where $\theta^1, \ldots, \theta^n$ are adjusted by an adaptation law. The adaptation law is designed to minimize the parameter errors of $\theta - \theta^*$. A fuzzy system is designed to compensate the uncertainties of the nonlinear system. The control input is given by

$$\tau = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r + G^{\wedge} - F^{\wedge}(s) - F_{cp}(s) \quad \textbf{(60)}$$

Where $F^{\wedge}(s) = [f_1^{\wedge}(s_1), \ldots, f_m^{\wedge}(s_m)]^T$ and $F_{cp}(S) = [F_{cp1}(s_1), \ldots, F_{cp_m}(s_m)]^T$. We define $F = \Delta M\ddot{q}_r + \Delta C_1\dot{q}_r + \Delta G$ where $F = [f_1, \ldots, f_m]^T$, $\Delta M = M^{\wedge} - M$, $\Delta C_1 = C_1^{\wedge} - C_1$ and $\Delta G_1 = G^{\wedge} - G$.
From Universal Approximation Theorem in (58), there exists an optimal fuzzy controller $f^{\wedge *}_j(s_j)$ such that:

$$f_j = f^{\wedge *}_j(e, \dot{e}) + \Delta_J \quad \textbf{(61)}$$

Where Δ_J is the minimum approximation error.

The fuzzy if-then rules are given by fuzzy rule base. In (59), we assume $\sum_{L=1}^{M} \mu_{A_j^L}(e, \dot{e}) = 1$ and $\varepsilon_j^L(e, \dot{e})$ becomes

$$\varepsilon_j^L(s_j) = \frac{\mu_{A_j^L}(e, \dot{e})}{\sum_{L=1}^{M}[\mu_{A_j^L}(e, \dot{e})]} = \mu_{A_j^L}(e, \dot{e}) \quad \textbf{(62)}$$

Where we define $\varphi_j^L = \mu_{A_j^L}(e, \dot{e})$. The membership function $\mu_{A_j^L}(e, \dot{e})$ is a Gaussian membership function represented by

$$\mu_{A_j^L}(s_j) = \exp\left[-(\sigma_j^l((e, \dot{e}) - \alpha_j^l))^2\right] \quad \textbf{(63)}$$

Then the fuzzy estimator $f^{\wedge}_j(e, \dot{e})$ is given as

$$f^{\wedge}_j(e, \dot{e}) = \theta_j^T \varphi_j \quad \textbf{(64)}$$

where $\theta_j = [\theta_j^1, \ldots, \theta_j^M]^T$, $\varphi_j = [\varphi_j^1, \varphi_j^2, \ldots, \varphi_j^M]^T$. We define $\tilde{f}_j$ such that

$$\tilde{f}_j = f_j - f^{\wedge}_j(e, \dot{e})$$

$$= f^{\wedge *}_{j}(e,\dot{e}) - f^{\wedge}_{j}(e,\dot{e}) + \Delta_j$$
$$= \theta_j^{*T}\varphi_j^* - \theta_j^T\varphi_j + \Delta_j \qquad \textbf{(65)}$$

where θ_j^* and φ_j^* are the optimal values based on Universal Approximation Theorem in (58). We define $\tilde{\theta}_j = \theta_j^* - \theta_j$, $\tilde{\varphi}_j = \varphi_j^* - \varphi_j$ and (6) is rewritten as

$$\tilde{f}_j = (\theta_j + \tilde{\theta}_j)^T(\varphi_j + \tilde{\varphi}_j) - \theta_j^T\varphi_j + \Delta_j$$
$$= \theta_j^T\tilde{\varphi}_j + \tilde{\theta}_j^T\varphi_j + \tilde{\theta}_j^T\tilde{\varphi}_j + \Delta_j \qquad \textbf{(66)}$$

We take Taylor series expansion of φ_j around two vectors α_j and σ_j where $\alpha_j = [\alpha_j^1,\dots,\alpha_j^M]^T$ and $\alpha_j = [\sigma_j^1,\dots,\sigma_j^M]^T$ (α_j^l and σ_j^l are defined in (63)):

$$\varphi_j^* = \varphi_j + \frac{\partial\varphi_j}{\partial\alpha_j}\tilde{\alpha}_j + \frac{\partial\varphi_j}{\partial\alpha_j}\tilde{\sigma}_j + h.o.t. \qquad \textbf{(67)}$$

where $\tilde{\alpha}_j = \alpha_j^* - \alpha_j$, $\tilde{\varphi}_j = \varphi_j^* - \varphi_j$ and $h.o.t.$ denotes the higher order terms. We rewrite (67) as

$$\tilde{\varphi}_j = \frac{\partial\varphi_j}{\partial\alpha_j}\tilde{\alpha}_j + \frac{\partial\varphi_j}{\partial\alpha_j}\tilde{\sigma}_j + h.o.t.$$
$$= B_j\tilde{\alpha}_j + C_j\tilde{\sigma}_j + h.o.t. \qquad \textbf{(68)}$$

where

$$B_j = \begin{bmatrix} \frac{\partial\varphi_j^1}{\partial\alpha_j^1} & \frac{\partial\varphi_j^2}{\partial\alpha_j^1} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^1} \\ \frac{\partial\varphi_j^1}{\partial\alpha_j^2} & \frac{\partial\varphi_j^2}{\partial\alpha_j^2} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^2} \\ \vdots & & & \\ \frac{\partial\varphi_j^1}{\partial\alpha_j^M} & \frac{\partial\varphi_j^2}{\partial\alpha_j^M} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^M} \end{bmatrix}, \qquad C_j = \begin{bmatrix} \frac{\partial\varphi_j^1}{\partial\alpha_j^1} & \frac{\partial\varphi_j^2}{\partial\alpha_j^1} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^1} \\ \frac{\partial\varphi_j^1}{\partial\alpha_j^2} & \frac{\partial\varphi_j^2}{\partial\alpha_j^2} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^2} \\ \vdots & & & \\ \frac{\partial\varphi_j^1}{\partial\alpha_j^M} & \frac{\partial\varphi_j^2}{\partial\alpha_j^M} & \cdots & \frac{\partial\varphi_j^M}{\partial\alpha_j^M} \end{bmatrix} \qquad \textbf{(69)}$$

We substitute (68) into (66):

$$\tilde{f}_j = \theta_j^T(B_j\tilde{\alpha}_j + C_j\tilde{\sigma}_j + h.o.t.) + \tilde{\theta}_j^T\varphi_j + \tilde{\theta}_j^T\tilde{\varphi}_j + \Delta_j$$
$$= \theta_j^T B_j\tilde{\alpha}_j + \theta_j^T C_j\tilde{\sigma}_j + \tilde{\theta}_j^T\varphi_j + \theta_j^T h.o.t. + \tilde{\theta}_j^T\tilde{\varphi}_j + \Delta_j$$
$$= \theta_j^T B_j\tilde{\alpha}_j + \theta_j^T C_j\tilde{\sigma}_j + \tilde{\theta}_j^T\varphi_j + \varepsilon_j \qquad \textbf{(70)}$$

where $\varepsilon_j = \theta_j^T h.o.t. + \tilde{\theta}_j^T\tilde{\varphi}_j + \Delta_j$ is assumed to be bounded by $|\varepsilon_j| \le E_j$. E_j is a constant and the value of E_j uncertain to the designer. We define E^* as the real value and the estimation error is given by

$$\tilde{E}_j = E_j^* - E_j \qquad \textbf{(71)}$$

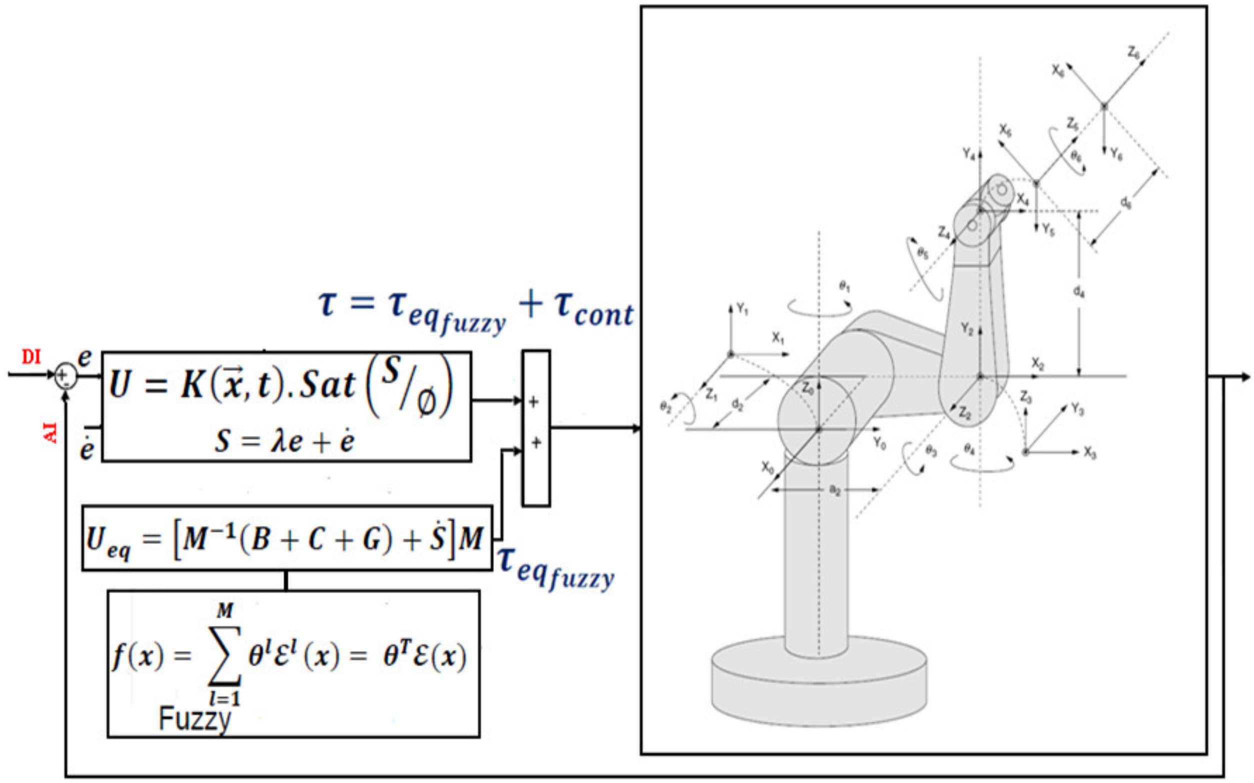


FIGURE 3: Chattering free Block diagram of a fuzzy estimator SMC: applied to robot manipulator

Third step is focused on design fuzzy (antecedent and consequent) adaptive fuzzy estimation sliding mode based on Lyapunov formulation. We produce an adaptation law to online tune the following parameters: θ_j in (64), σ_j^l , α_j^l in (63) and the bound E_j in (71). The adaptation laws are expressed as

$$\dot{\theta}_j = \eta_{j2} s_j \varphi_j \quad \textbf{(72)}$$

$$\dot{\alpha}_j = \eta_{j3} s_j B_j^T \theta_j \quad \textbf{(73)}$$

$$\dot{\sigma}_j = \eta_{j4} s_j C_j^T \theta_j \quad \textbf{(74)}$$

$$f_{cpj}(s_j) = E_j sgn(s_j) \quad \textbf{(75)}$$

$$\dot{E}_j = \eta_{j1} |s_j| \quad \textbf{(76)}$$

where $\eta_{j1}, \eta_{j2}, \eta_{j3}$ and η_{j4} are positive constants; $\theta_j = [\theta_j^1, \theta_j^2, \dots, \theta_j^M]^T$, $\alpha_j = [\alpha_j^1, \alpha_j^2, \dots, \alpha_j^M]^T$, $\sigma_j = [\sigma_j^1, \sigma_j^2, \dots, \sigma_j^M]^T$; B_j , C_j are given in (69); $f_{cpj}(s_j)$ is the compensation term defined in (60).

If the following Lyapunov function candidate defined by:

$$V = \frac{1}{2} s^T M s + \frac{1}{2} \sum_{j=1}^{m} \left(\frac{\tilde{E}_j^2}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \tilde{\theta}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \tilde{\alpha}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \tilde{\sigma}}{\eta_{j4}} \right) \quad \textbf{(77)}$$

The derivative of V is defined by:

$$\dot{V} = s^T M\dot{s} + \frac{1}{2} s^T \dot{M} s + \sum_{j=1}^{m}\left(\frac{\tilde{E}_j \dot{\tilde{E}}_j}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \dot{\tilde{\theta}}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \dot{\tilde{\alpha}}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \dot{\tilde{\sigma}}}{\eta_{j4}}\right) \quad \textbf{(78)}$$

where $\tilde{E}_j = E_j^* - E_j$, $\tilde{\theta}_j = \theta_j^* - \theta_j$, $\tilde{\alpha}_j = \alpha_j^* - \alpha_j$, $\tilde{\sigma}_j = \sigma_j^* - \sigma_j$. from robot manipulator formulation and (1):

$$M(q)\ddot{q} + C_1(q,\dot{q})\dot{q} + G(q) = M^{\wedge}\ddot{q}_r + C_1^{\wedge}\dot{q}_r + G^{\wedge} - F^{\wedge}(s) - F_{cp}(s) \quad \textbf{(79)}$$

Since $\dot{M} - 2C_1$ is a skew-symmetric matrix, we can get $s^T M\dot{s} + \frac{1}{2}s^T \dot{M} s = s^T(M\dot{s} + C_1 s)$. From $\dot{q}_r = \dot{q} - s = \dot{q}_d - \lambda e$ and $\ddot{q}_r = \ddot{q} - \dot{s} = \ddot{q}_d - \lambda\dot{e}$

$$M\dot{s} + C_1 s = F - F^{\wedge}(s) - F_{cp}(s) \quad \textbf{(80)}$$

where $= \Delta M\ddot{q}_r + \Delta C_1\dot{q}_r + \Delta G$, $\Delta M = M^{\wedge} - M$, $\Delta C_1 = C_1^{\wedge} - C_1$ and $\Delta G_1 = G^{\wedge} - G$. Then $\dot{V}$ becomes

$$\dot{V} = s^T(M\dot{s} + C_1 s) + \sum_{j=1}^{m}\left(\frac{\tilde{E}_j \dot{\tilde{E}}_j}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \dot{\tilde{\theta}}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \dot{\tilde{\alpha}}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \dot{\tilde{\sigma}}}{\eta_{j4}}\right)$$

$$= s^T(F - F^{\wedge}(s) - F_{cp}(s)) + \sum_{j=1}^{m}\left(\frac{\tilde{E}_j \dot{\tilde{E}}_j}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \dot{\tilde{\theta}}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \dot{\tilde{\alpha}}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \dot{\tilde{\sigma}}}{\eta_{j4}}\right)$$

$$= \sum_{j=1}^{m} s_j(f_j - f^{\wedge}_j(s_j) - f_{cpj}) + \sum_{j=1}^{m}\left(\frac{\tilde{E}_j \dot{\tilde{E}}_j}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \dot{\tilde{\theta}}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \dot{\tilde{\alpha}}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \dot{\tilde{\sigma}}}{\eta_{j4}}\right)$$

$$= \sum_{j=1}^{m} s_j(\theta_j^T B_j\tilde{\alpha}_j + \theta_j^T C_j\tilde{\sigma}_j + \tilde{\theta}_j^T\varphi_j + \varepsilon_j - f_{cpj}) - \sum_{j=1}^{m}\left(\frac{\tilde{E}_j \dot{\tilde{E}}_j}{\eta_{j1}} + \frac{\tilde{\theta}_j^T \dot{\tilde{\theta}}}{\eta_{j2}} + \frac{\tilde{\alpha}_j^T \dot{\tilde{\alpha}}}{\eta_{j3}} + \frac{\tilde{\sigma}_j^T \dot{\tilde{\sigma}}}{\eta_{j4}}\right)$$

$$= \sum_{j=1}^{m}\left[\tilde{\theta}_j^T\left(s_j\varphi_j - \frac{\dot{\theta}_j}{\eta_{j2}}\right) + \tilde{\alpha}_j^T\left(s_j B_j^T\theta_j - \frac{\dot{\alpha}_j}{\eta_{j3}}\right) + \tilde{\sigma}_j^T\left(s_j C_j^T\theta_j - \frac{\dot{\sigma}_j}{\eta_{j4}}\right)\right] + \sum_{j=1}^{m}\left(s_j\varepsilon_j - s_j f_{cpj}\frac{\tilde{E}_j\dot{E}_j}{\eta_{j1}}\right) \quad \textbf{(81)}$$

We substitute the adaption law (72)-(76) in to (81):

$$\dot{V} = \sum_{j=1}^{m}[s_j\varepsilon_j - s_j E_j sgn(s_j) - \tilde{E}_j|s_j|]$$

$$= \sum_{j=1}^{m}[s_j\varepsilon_j - s_j E_j sgn(s_j) - (E_j^* - E_j)s_j sgn(s_j)]$$

$$= \sum_{j=1}^{m}[s_j\varepsilon_j - E_j^* s_j sgn(s_j)]$$

$$= \sum_{j=1}^{m}[|s_j||\varepsilon_j| - E_j^*|s_j|]$$

$$= \sum_{j=1}^{m}[|s_j|(|\varepsilon_j| - E_j^*)] \leq 0 \quad \textbf{(82)}$$

where $\dot{V}$ is negative semidefinite . We define $\dot{V}_j = |s_j(t)|(|\varepsilon_j| - E_j^*)$. From $\dot{V}_j \leq 0$, we can get $s_j(t)$ is bounded. We assume $|s_j(t)| \leq \eta_s$ and rewrite $|s_j(t)|(E_j^* - |\varepsilon_j|) \leq -\dot{V}_j$ as

$$s_j(t) \leq \frac{1}{E_j^*}|s_j(t)||\varepsilon_j| - \frac{1}{E_j^*}\dot{V}_j \leq \frac{\eta_s}{E_j^*}|\varepsilon_j| - \frac{1}{E_j^*}\dot{V}_j \quad \textbf{(83)}$$

Then we take the integral on both sides of (83):

$$\int_0^t |s_j(v)|dv \leq \frac{\eta_s}{E_j^*}\int_0^t |\varepsilon_j|dv + \frac{1}{E_j^*}\left(V_j(0) - V_j(t)\right) \leq \frac{\eta_s}{E_j^*}\int_0^t |\varepsilon_j|dv + \frac{1}{E_j^*}\left(|V_j(0)| - |V_j(t)|\right) \quad \textbf{(84)}$$

If $\varepsilon_j \in L_1$, we can get $s_j \in L_1$ from (84). Since we can prove $\dot{s}_j$ is bounded (see proof in [15]), we have $\dot{s}_j \in L_\infty$. We introduce the following Barbalat's Lemma[16].

Lemma: let $\emptyset: R \to R$ be a uniformly continues function on $[0,\infty)$. Suppose that $lim_{t\to\infty}\int_0^t \emptyset(\tau)d\tau$ exists and is finite. Then,

$lim_{t\to\infty}\emptyset(t) = 0$

Therefore, by using Barbalat's Lemma, we can get $lim_{t\to\infty}s_j(t) = 0$ and $lim_{t\to\infty}e_j(t) = 0$. Figure 4 is shown fuzzy (antecedent and consequent) adaptive fuzzy estimator sliding mode controller.

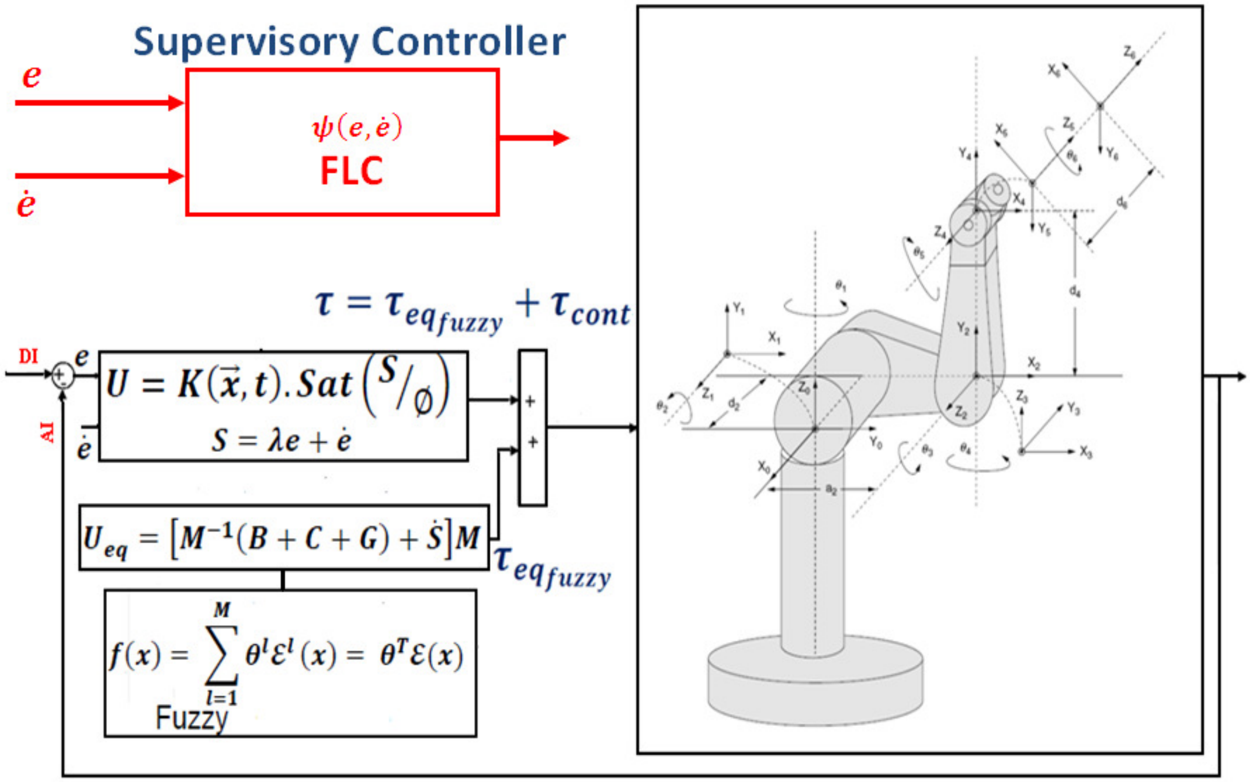


FIGURE 4: Chattering free Block diagram of a fuzzy adaptive (consequent and antecedent) fuzzy estimator SMC: applied to robot manipulator

4. RESULTS

This methodology can online adjust both the premise and the consequent parts of the fuzzy rules. In this method we choose $\hat{M}, \hat{B}$ and $\hat{C}$ for compensation. We define five membership functions for each input variable based on

$$\mu_{A_j^l}(S_j) = \exp[-(\sigma_j^l(S_j - \alpha_j^l))^2] \quad \textbf{(85)}$$

Sliding mode controller (SMC) and proposed fuzzy (premise and the consequent) adaptive fuzzy estimator sliding mode controller were tested to sinus response trajectory. The simulation was implemented in Matlab/Simulink environment. Link/joint trajectory and disturbance rejection are compared in these controllers. These systems are tested by band limited white noise with a predefined 40% of relative to the input signal amplitude.

Link/joint Trajectory: Figure 5 shows the link/joint trajectory in SMC and proposed methodology without disturbance for sinus trajectory. (This techniques is applied to first two link of PUMA robot manipulator)

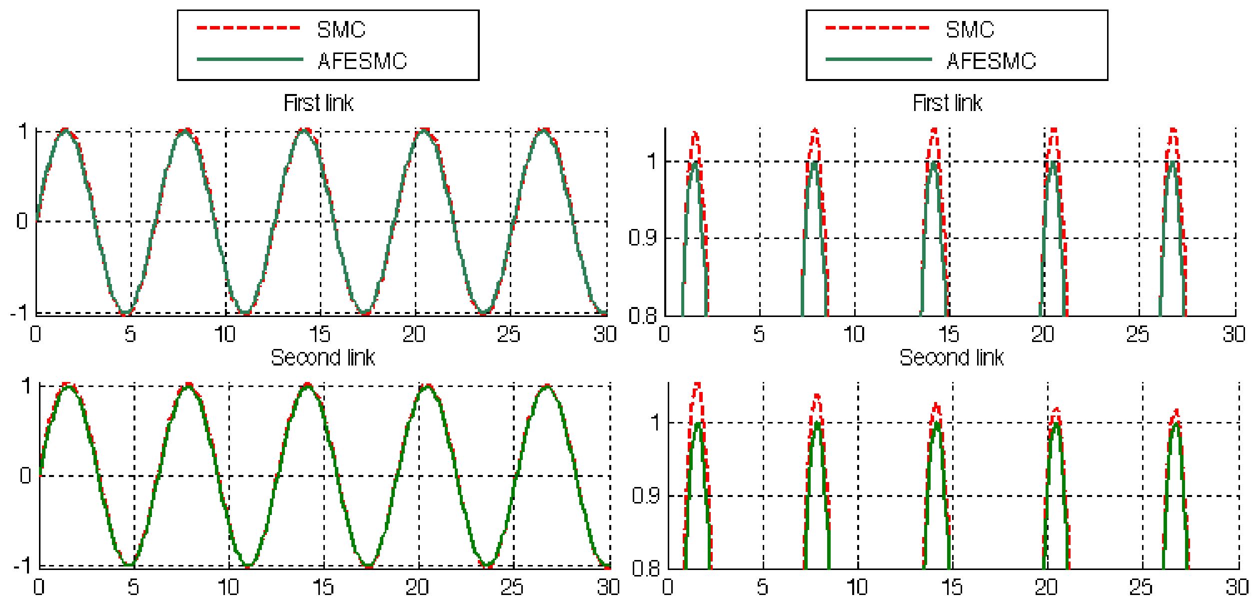


FIGURE 5: Proposed method **(**AFESMC) and SMC trajectory: applied to robot manipulator

By comparing sinus response, Figure 5, in SMC and proposed controller, the proposed controller's overshoot **(0%)** is lower than SMC's **(3%).**

Disturbance rejection: Figure 6 is indicated the power disturbance removal in SMC and proposed controller. Besides a band limited white noise with predefined of 40% the power of input signal is applied to the sinus SMC and proposed controller; it found slight oscillations in SMC trajectory responses.

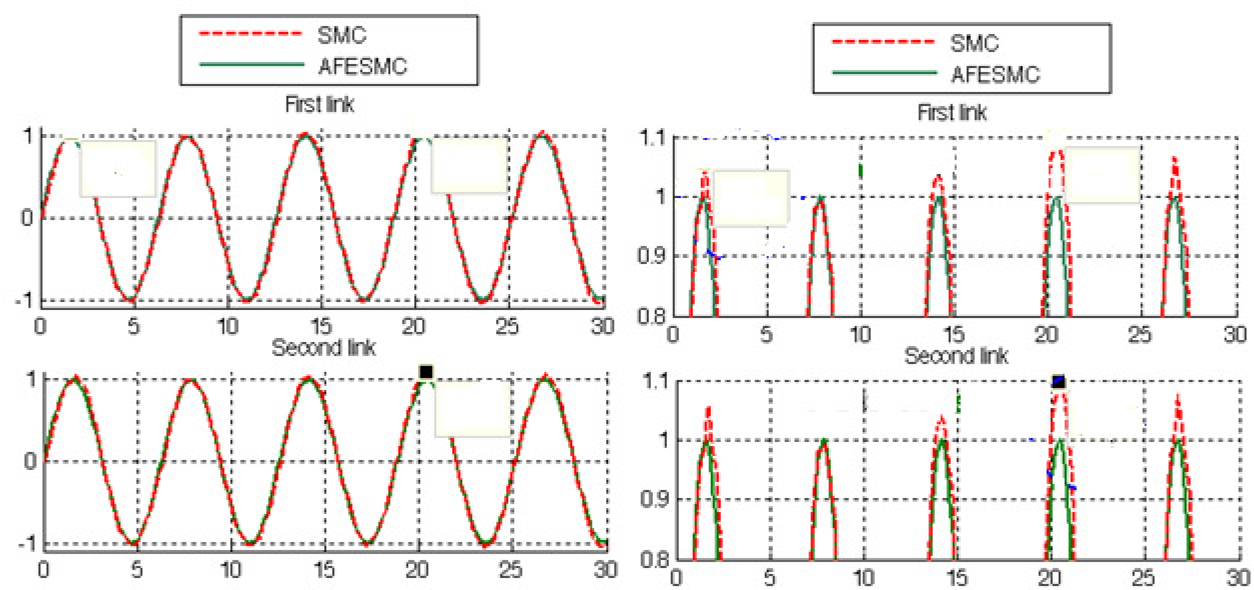


FIGURE 6: Proposed controller (AFESMC) and SMC trajectory with external disturbance: applied to robot manipulator

5 CONCLUSIONS

In this paper, a fuzzy (premise and consequent) adaptive robust control fuzzy estimator sliding mode method is proposed in order to design a high performance robust controller in the presence of structured uncertainties and unstructured uncertainties. The approach improves performance by using the advantages of sliding mode control, fuzzy logic estimation method and adaptive control while the disadvantages attributed to these methods are remedied by each other. This is achieved without increasing the complexities of the overall design and analysis of the controller. The proposed controller attenuates the effort of model uncertainties from both structured uncertainties and unstructured uncertainties. Thus, transient performance and final tracking accuracy is guaranteed by proper design of the controller. The results revealed that adaption of fuzzy rules weights reduce the model uncertainties significantly, and hence farther improvements of the tracking performance can be achieved. This algorithm created a methodology of learning both the premise and the consequence part of fuzzy rules. In this method chattering is eliminated.

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] 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.

[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] 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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] Piltan, F., et al., "Artificial control of PUMA robot manipulator: A-review of fuzzy inference engine and application to classical controller," International Journal of Robotic and Automation, 2 (5), 2011.

[16] Piltan, F., et al., "Design and implementation of sliding mode algorithm: applied to robot manipulator-A review," International Journal of Robotic and Automation, 2 (5), 2011.

[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] Iordanov, H. N., B. W. Surgenor, 1997. Experimental evaluation of the robustness of discrete sliding mode control versus linear quadratic control, IEEE Trans. On control system technology, 5(2):254-260.

[63] Harashima F., Hashimoto H., and Maruyama K, 1986. Practical robust control of robot arm using variable structure system, IEEE conference, P.P:532-539

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

[65] Piltan, F., et al., "An adaptive sliding surface slope adjustment in PD sliding mode fuzzy control for robot manipulator," International Journal of Control and Automation, 4 (3): 65-76, 2011.

[66] Piltan, F., et al., "Novel artificial control of nonlinear uncertain system: design a novel modified PSO SISO Lyapunov based fuzzy sliding mode algorithm," International Journal of Robotic and Automation, 2 (5), 2011.

[67] Piltan, F., et al., "Design adaptive fuzzy inference sliding mode algorithm: applied to robot arm," International Journal of Robotic and Automation, 2 (5), 2011.

[68] Piltan, F., et al., "Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine and Application to Classical Controller ," International Journal of Robotic and Automation, 2 (5), 2011.

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.
- Determine α_i, that α_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;

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

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

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

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

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

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

$$R_n^0 = (U_1V_1)(U_2V_2) \dots \dots \dots (U_nV_n) \quad (5)$$

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

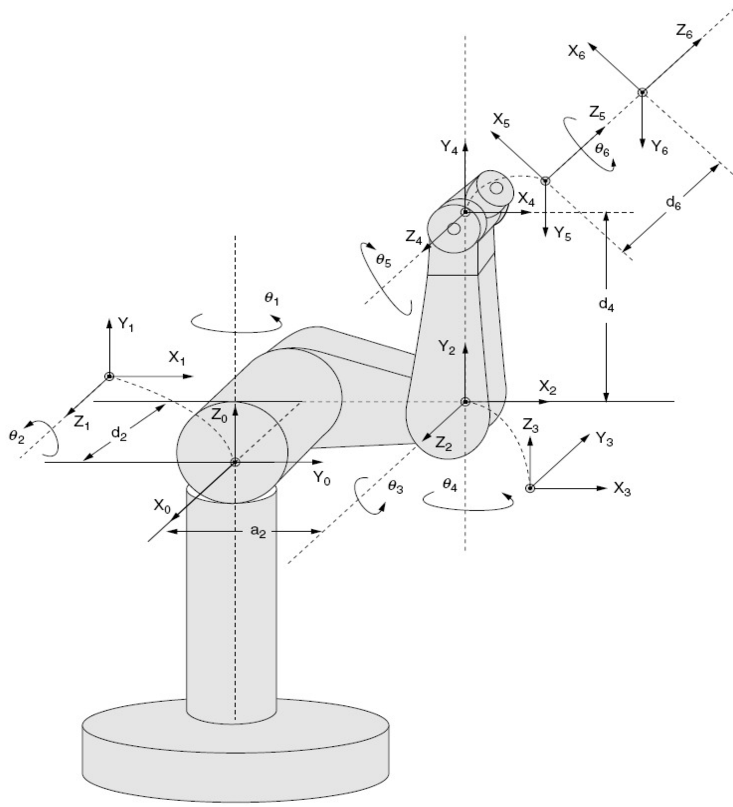


FIGURE 1: D-H notation for a six-degrees-of-freedom PUMA 560 robot manipulator[2]

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]

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

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

$${}^0_nT = {}^0_1T.{}^1_2T.{}^2_3T \dots \dots .{}^{n-1}_{\ \ n}T = \begin{bmatrix} R_n^0 & d_n^0 \\ 0 & 1 \end{bmatrix} \tag{7}$$

In PUMA robot manipulator the final transformation matrix is given by

$${}^0_6T = {}^0_1T.{}^1_2T.{}^2_3T \dots \dots .{}^5_6T = \begin{bmatrix} R_6^0 & d_6^0 \\ 0 & 1 \end{bmatrix} \tag{8}$$

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 ${}^0_6T$ can be determined by

$$ {}^0_6T = \begin{bmatrix} N_x & B_x & T_x & P_x \\ N_y & B_y & T_y & P_y \\ N_z & B_z & T_z & P_z \end{bmatrix} \quad (10) $$

Table 2 shows the PUMA 560 D-H parameters.

Link i	θ_i(rand)	α_i(rand)	a_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:

$$ Nx = cos(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) + sin(\theta_1) \times sin(\; sin(\theta_2+\theta_3) \times cos(\theta_1)) + sin(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) - cos(\quad (11) $$

$$ Ny = cos(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos \; sin(\theta_5) \times sin(\theta_2+\theta_3) \times sin(\theta_1)) + sin(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times si \; cos(\theta_4) \times cos(\theta_1)) \quad (12) $$

$$ Nz = cos(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2+\theta_3) - sin(\theta_5) \times cos(\theta_2+\theta_3)) + \; sin(\theta_4) \times sin(\theta_2+\theta_3) \quad (13) $$

$$ Bx = -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times cos(\theta_1) + sin(\theta_4) \; sin(\theta_5) \times sin(\theta_2+\theta_3) \times cos(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times co \; cos(\theta_4) \times sin(\theta_1)) \quad (14) $$

$$ By = -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\theta_1) - sin(\theta_4) \times co \; sin(\theta_5) \times sin(\theta_2+\theta_3) \times sin(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2+\theta_3) \times sin(\; cos(\theta_1)) \quad (15) $$

$$ Bz = -sin(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2+\theta_3) - sin(\theta_5) \times cos(\theta_2+\theta_3)) \; sin(\theta_4) \times sin(\theta_2+\theta_3) \quad (16) $$

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

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]

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

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

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

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];

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

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

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

$$M_{12} = I_4\sin(\theta_2)+I_8\cos(\theta_2+\theta_3)+I_9\cos(\theta_2)+I_{13}\sin(\theta_2+\theta_3)-I_{18}\cos(\theta_2+ \tag{36}$$

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

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

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

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

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

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

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

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

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

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,

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

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

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

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

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

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

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

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

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

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

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

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

consequently coriolis matrix is shown as bellows;

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

Moreover Centrifugal (C) matrix is demonstrated as

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

Where,

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

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

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

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

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

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

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

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

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

$$C(q).q^{\dot{2}} = \begin{bmatrix} c_{112}.\dot{q}_2^2 + c_{13}.\dot{q}_3^2 \\ c_{21}.\dot{q}_1^2 + c_{23}.\dot{q}_3^2 \\ c_{13}.\dot{q}_1^2 + c_{32}.\dot{q}_2^2 \\ 0 \\ c_{51}.\dot{q}_1^2 + c_{52}.\dot{q}_2^2 \\ 0 \end{bmatrix} \quad (69)$$

Gravity (G) Matrix can be written as

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

Where,

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

$$G_3 = g_2\sin(\theta_2+\theta_3) + g_4\cos(\theta_2+\theta_3) + g_5\sin(\theta_2+\theta_3) \quad (72)$$

$$G_5 = g_5\sin(\theta_2+\theta_3) \quad (73)$$

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

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

and K is introduced as

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

$\ddot{q}$ can be written as

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

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

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

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

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

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

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

$$K_6 = \tau_6 \quad (82)$$

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 $n-DOF$ robot manipulator with k membership function for each input variable, the number of fuzzy rules for each joint is equal to $3k^{2n}$ 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 K_2 membership functions, the number of fuzzy rules for each joint is $(m+1)K_2^m+K_2$. 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;

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

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

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

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

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

A Gaussian membership function of fuzzy set is defined by

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

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

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

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

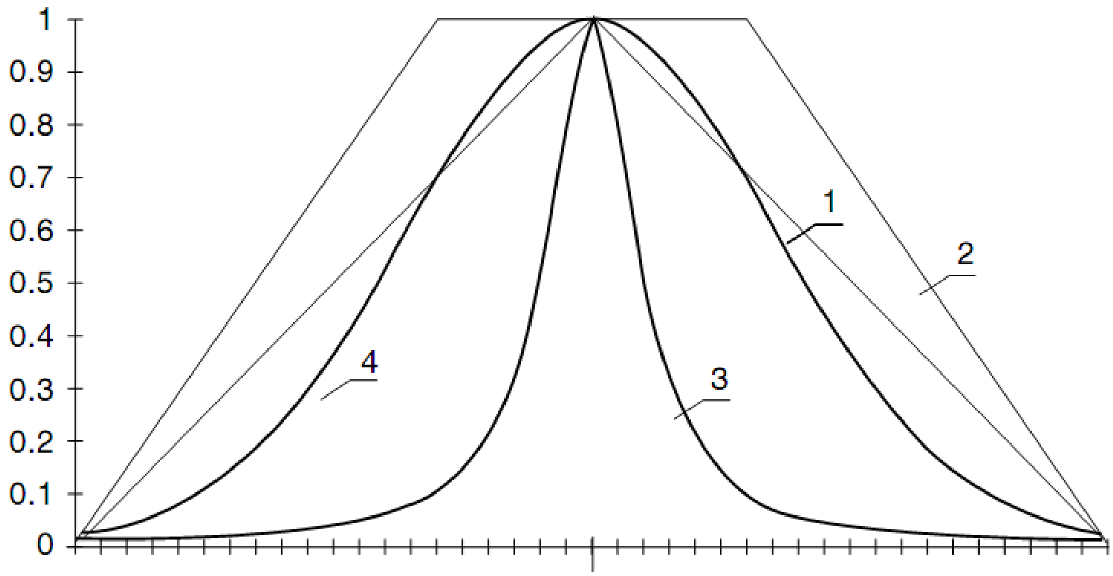


FIGURE 2: Most important membership functions in fuzzy set: 1-Trianglar, 2-Trapezoidal, 3-Gaussian, 4-Bell-shaped [53].

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

$$T(a,b)=\mu_{A\cap B(u)}=\min\{\mu_{A(u)},\mu_{B(u)}\}=\mu_{A(u)}\cdot\mu_{B(u)} \tag{89}$$

$$=\max\left(0,\ \mu_{A(u)}+\mu_{B(u)}-1\right)=\begin{cases}\mu_{A(u)} & , if\ \mu_{B(u)}=1\\ \mu_{B(u)} & , if\ \mu_{A(u)}=1\\ 0 & , if \mu_{B(u)},\mu_{A(u)}<1\end{cases}$$

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:

$$if\ e\ is\ NB\ and\ \dot{e}\ is\ ML\ then\ T\ is\ LL \quad (93)$$

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.

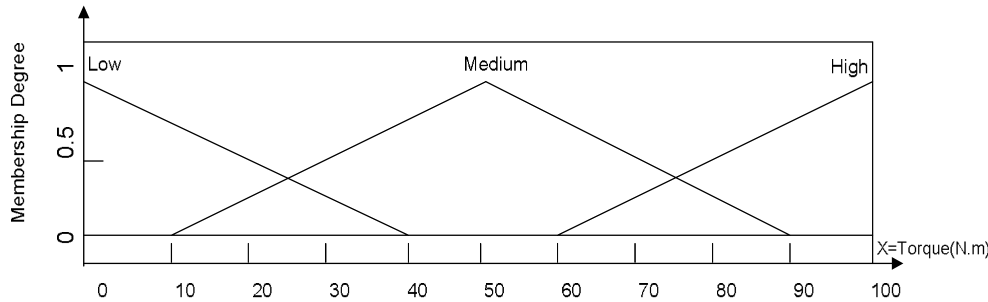


FIGURE 3: Linguistic variable and Linguistic value

$If-then$ 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

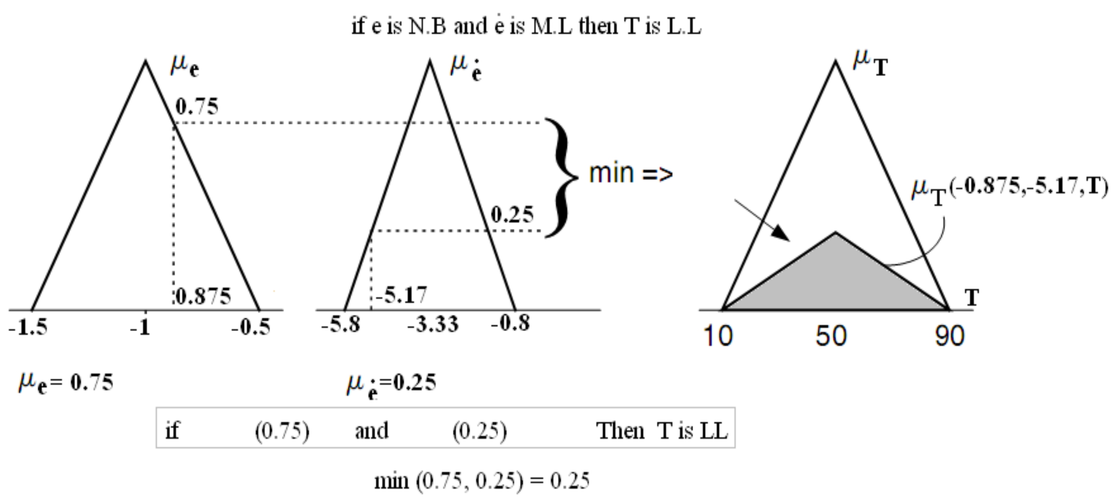


FIGURE 4: Main three parts in IF-THEN rules in fuzzy set

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

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 $F.R^2$:

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

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

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:

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

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

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.

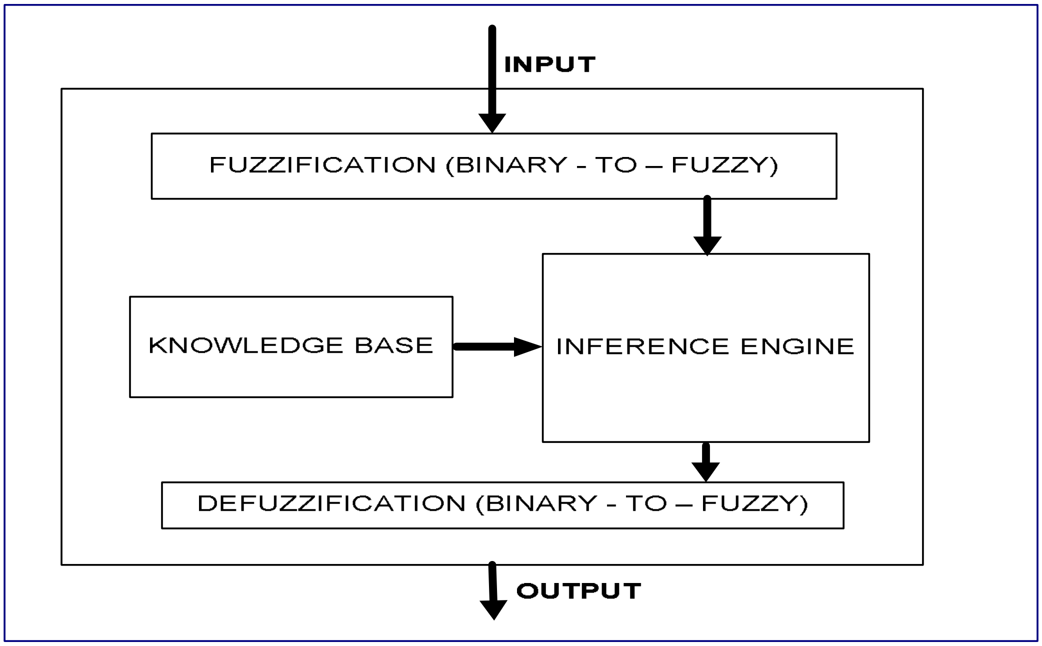


FIGURE 5: Structure of Fuzzy Logic Controller (F.L.C)

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.

INSTRUCTIONS TO CONTRIBUTORS

Robots are becoming part of people's everyday social lives - and will increasingly become so. In future years, robots may become caretaking assistants for the elderly or academic tutors for our children, or medical assistants, day care assistants, or psychological counselors. Robots may become our co-workers in factories and offices, or maids in our homes.

The International Journal of Robotics and Automation (IJRA), a refereed journal aims in providing a platform to researchers, scientists, engineers and practitioners throughout the world to publish the latest achievement, future challenges and exciting applications of intelligent and autonomous robots. IJRA is aiming to push the frontier of robotics into a new dimension, in which motion and intelligence play equally important roles. IJRA scope includes systems, dynamics, control, simulation, automation engineering, robotics programming, software and hardware designing for robots, artificial intelligence in robotics and automation, industrial robots, automation, manufacturing, and social implications.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJRA.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with Volume 3, 2012, IJRA appear with more focused issues. Besides normal publications, IJRA intends to organize special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

IJRA LIST OF TOPICS

The realm of International Journal of Robotics and Automation (IJRA) extends, but not limited, to the following:

- Automation Control
- Automation Engineering
- Autonomous Robots
- Biotechnology & Robotics
- Emergence of The Thinking Machine
- Forward Kinematics
- Household Robots and Automation
- Inverse Kinematics
- Jacobian and Singularities
- Methods for Teaching Robots
- Nanotechnology & Robotics (Nanobots)
- Orientation Matrices
- Robot Controller
- Robot Structure and Workspace
- Robotic & Automation Software Development
- Robotic Exploration
- Robotic Surgery
- Robotic Surgical Procedures
- Robotic Welding
- Robotics Applications
- Robotics Programming
- Robotics Technologies
- Robots Society and Ethics
- Software and Hardware Designing for Robots
- Spatial Transformations
- Trajectory Generation
- Unmanned (Robotic) Vehicles

CALL FOR PAPERS

Volume: 3 - **Issue: 1**

i. Paper Submission: December 31, 2011 **ii. Author Notification:** January 31, 2012

iii. Issue Publication: February 2012

CONTACT INFORMATION

Computer Science Journals Sdn BhD
M-3-19 Plaza Damas, Sri Hartamas
50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6207 1607
006 03 2782 6991

Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2011
COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
006 03 2782 6991

FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org