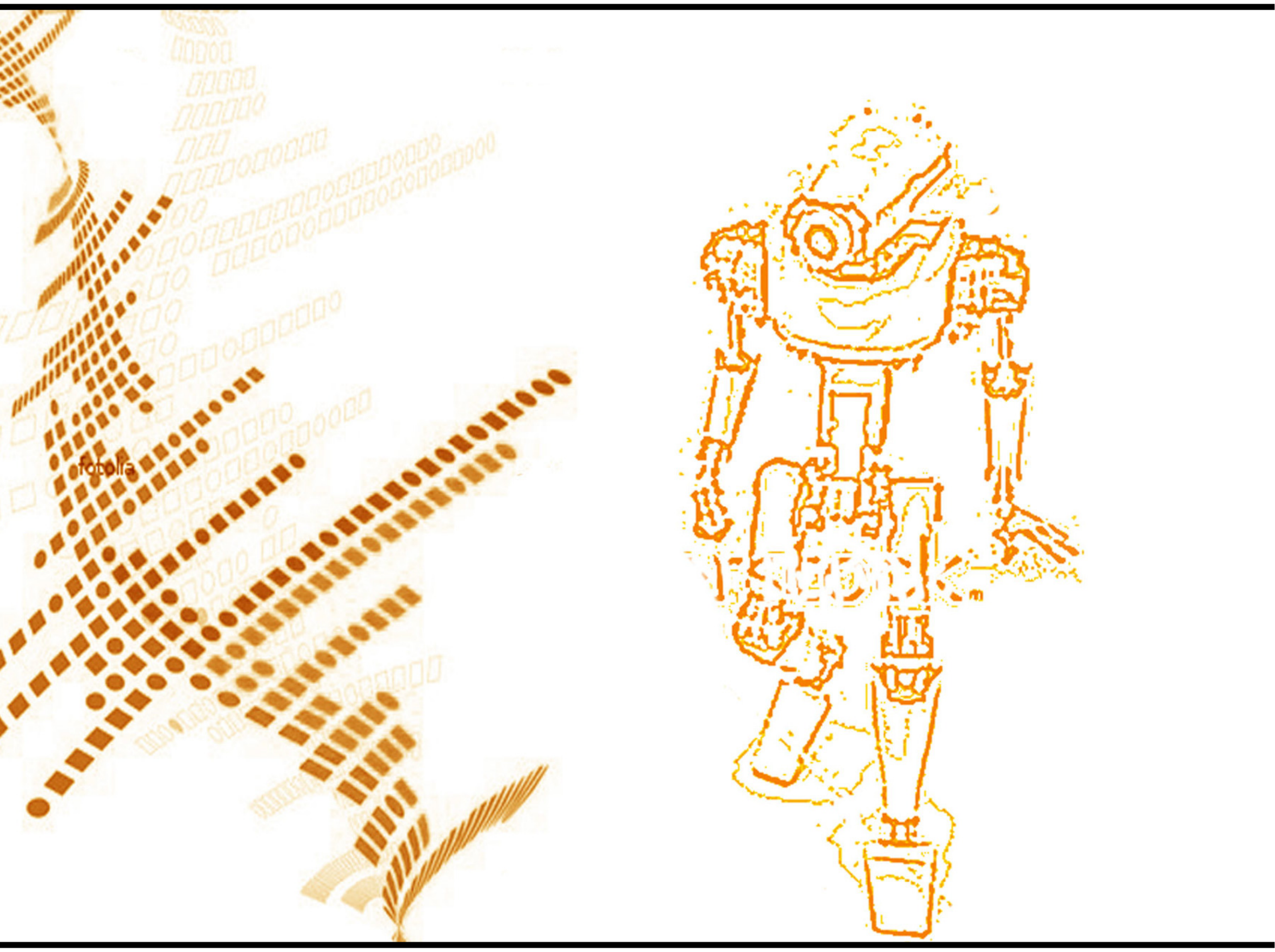
Volume 3 ▪ Issue 3 ▪ May / June 2012

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 3, ISSUE 3, 2012

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 3, Issue 3, May / June 2012
Publishing Date: 20-06- 2012
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, 2012

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 *Third* Issue of Volume *Three* 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. Started with Volume 3, 2012, IJRA appears 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)

TABLE OF CONTENTS

Volume 3, Issue 3, May / June 2012

Pages

Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach

Farzin Piltan 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

Ali Hosainpour 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

Ebrahim Mazlomian 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

Mohammad Shamsodini 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

Mohammad Hossein Yarmahmoudi 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 sliding mode fuzzy controller which sliding surface gain is on-line tuned by minimum fuzzy inference algorithm. The main goal is to guarantee acceptable trajectories tracking between the second order nonlinear system (robot manipulator) actual and the desired trajectory. The fuzzy controller in proposed sliding mode fuzzy controller is based on Mamdani's fuzzy inference system (FIS) and it has one input and one output. The input represents the function between sliding function, error and the rate of error. The outputs represent torque, respectively. The fuzzy inference system methodology is on-line tune the sliding surface gain based on error-based fuzzy tuning methodology. Pure sliding mode fuzzy controller has difficulty in handling unstructured model uncertainties. To solve this problem applied fuzzy-based tuning method to sliding mode fuzzy controller for adjusting the sliding surface gain (λ). Since the sliding surface gain (λ) is adjusted by fuzzy-based tuning method, it is nonlinear and continuous. Fuzzy-based tuning sliding mode fuzzy controller is stable model-free controller which eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in fuzzy-based tuning sliding mode fuzzy controller based on switching (sign) function.

This controller has acceptable performance in presence of uncertainty (e.g., overshoot=0%, rise time=0.8 second, steady state error = 1e-9 and RMS error=1.8e-12).

Keywords: robot manipulator, sliding mode controller, sliding mode fuzzy controller, fuzzy on-line tune sliding mode fuzzy controller, Lyapunov- based, fuzzy inference system.

1. INTRODUCTION, BACKGROUND and MOTIVATION

Motivation and background: PUMA 560 robot manipulator is a 6 DOF serial robot manipulator. From the control point of view, robot manipulator divides into two main parts i.e. kinematics and dynamic parts. The dynamic parameters of this system are highly nonlinear [1-10]. Sliding mode controller (SMC) is a significant nonlinear controller under condition of partly uncertain dynamic parameters of system [1, 6-30]. This controller is used to control of highly nonlinear systems especially for robot manipulators, because this controller is a robust and stable [41-51]. 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 [52-60]. The chattering phenomenon problem can be reduced by using linear saturation boundary layer function in sliding mode control law. Lyapunov stability is proved in pure sliding mode controller based on switching (sign) function [59-60]. The nonlinear equivalent dynamic formulation problem in uncertain system can be solved by using artificial intelligence theorem. Fuzzy logic theory is used to estimate the system dynamic [31-40]. However fuzzy logic controller is used to control complicated nonlinear dynamic systems, but it cannot guarantee stability and robustness [31-40]. Fuzzy logic controller is used in adaptive methodology and this method is also can applied to nonlinear conventional control methodology to improve the stability, increase the robustness, reduce the fuzzy rule base and estimate the system's dynamic parameters [31-40]. To reduce the fuzzy rule base with regards to improve the stability and robustness sliding mode fuzzy controller is introduced. In sliding mode fuzzy controller sliding mode controller is applied to fuzzy logic controller to reduce the fuzzy rules and increase the stability and robustness [61-80]. The main drawback in sliding mode fuzzy controller is calculation the value of sliding surface slope coefficient pri-defined very carefully. To estimate the system dynamics, fuzzy inference system is introduced. Most of researcher is applied fuzzy logic theorem in sliding mode controller to design a model free controller. Pure sliding mode controller and sliding mode fuzzy controller have difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining fuzzy sliding mode controller or sliding mode fuzzy controller and adaption law which this method can helps improve the system's tracking performance by online tuning method [61-82]. It is possible to solve this problem by combining sliding mode fuzzy controller and fuzzy-based tuning which this method can helps improve the system's tracking performance by online tuning method. This method is based on resolve the on line sliding surface slope as well as improve the output performance by tuning the sliding surface slope coefficient. The sliding surface gain ($\boldsymbol{\lambda}$) of this controller is adjusted online depending on the last values of error (e) and change of error ($\dot{e}$) by sliding surface slope updating factor (α). Fuzzy-based tuning sliding mode fuzzy controller is stable model-free controller which does not need to limits the dynamic model of robot manipulator and eliminate the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in fuzzy-based tuning sliding mode fuzzy controller based on switching (sign) function.

Literature Review

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]. 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]. 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, Weng and 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 most exceptional stability and robustness. Sliding mode fuzzy controller has the two most important advantages: reduce the number of fuzzy rule base and increase robustness and stability. Conversely sliding mode fuzzy controller has the above advantages, define the sliding surface slope coefficient very carefully is the main disadvantage of this controller. 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. However this controller's response is very fast and robust but it has chattering phenomenon. 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 control of robot manipulator 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: artificial intelligence controller (neuro-fuzzy) 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 in 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 controller in presence of nonlinear dynamic part is significantly growing as nonlinear control applications. For instance, the applications of fuzzy logic on sliding mode controller have reported in [24, 41-45]. Fuzzy sliding mode controller (FSMC) is a nonlinear controller based on sliding mode method when fuzzy logic methodology applied to sliding mode controller to reduce the high frequency oscillation (chattering) and compensate the dynamic model of uncertainty based on nonlinear dynamic model [42-43]. Temeltas [46] has proposed fuzzy adaption techniques and applied to SMC to have robust controller and solves the chattering problem. In this method however system's performance is better than sliding mode controller but it is depended on nonlinear dynamic equations. Hwang and Chao [47] have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode controller based on neuro-fuzzy based linear state-space to estimate the uncertainties. A MIMO fuzzy sliding mode controller to reduces the chattering phenomenon and estimate the nonlinear equivalent part has been presented for a robot manipulator [42]. Sliding mode fuzzy controller (SMFC) is an artificial intelligence controller based on fuzzy logic methodology when, sliding mode controller is applied to fuzzy logic controller to reduce the fuzzy rules and refine the stability of close loop system in fuzzy logic controller [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. In comparison, to reduce the number of fuzzy rule base, increase the robustness and stability sliding mode fuzzy controller is more suitable than fuzzy logic controller [52]. 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 [75]. 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. Hsu et al. [54] have presented traditional adaptive fuzzy sliding mode control which can update fuzzy rules to compensate nonlinear parameters and guarantee the stability robot manipulator controller. Hsueh et al. [43] have presented traditional self tuning sliding mode controller which can resolve the chattering problem without 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 pure sliding mode controller and fuzzy logic controller, as a result it is used to adjust and tune coefficient. Adaptive fuzzy sliding mode controller is used to many applications. This controller is based on online tuning the parameters and caused to improve the trajectory [40, 55-57]. The adaptive sliding mode controller is used to estimate the unknown dynamic parameters and external disturbances. For instance, the applications of adaptive fuzzy sliding mode controller to control the robot manipulators have been reported in [24, 29, 45]. Generally, adaptive fuzzy sliding mode control of robot manipulator is classified into two main groups' i.e. multi-input multi-output (MIMO) and single-input single-output (SISO) fuzzy systems. Yoo and Ham [58] have proposed a MIMO fuzzy system to help the compensation and estimation the torque coupling. In $\boldsymbol{n-DOF}$ robot manipulator with $\boldsymbol{k}$ membership function for each input variable, the number of fuzzy rules for each joint is equal to $\boldsymbol{3k^{2n}}$ that causes to high computation load and also this controller has chattering. This method can only tune the consequence part of the fuzzy rules. Guo and Woo [60] have proposed a SISO fuzzy controller to compensate the switching terms. The number of fuzzy rules is reduced ($\boldsymbol{K_2}$) with regard to reduce the chattering. Lin and Hsu [61] have proposed a methodology to tuning consequence and premise part of fuzzy rules to reduce the chattering based on tuning the membership function. In this method the number of fuzzy rules equal to $\boldsymbol{K_2}$ with low computational load but chattering is expected. 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 $\boldsymbol{K_2}$ fuzzy rules numbers. The bounds of PI controller and the parameters are online adjusted by low adaption computation and tune the membership function[44]. Medhafer et al. [59] have proposed an indirect adaptive fuzzy sliding mode controller to control robot manipulator. This MIMO algorithm, applies to partly estimate the nonlinear dynamic parameters.

Contributions

Sliding mode controller is used to control of highly nonlinear systems especially for robot manipulators. The first problem of the pure sliding mode controller with switching function was chattering phenomenon in certain and uncertain systems. The nonlinear equivalent dynamic problem in uncertain system is the second challenge in pure sliding mode controller. To eliminate the PUMA robot manipulator's dynamic of system, 7 rules Mamdani inference system is design and applied to sliding mode methodology with switching function. This methodology is worked based on applied fuzzy logic in equivalent nonlinear dynamic part to eliminate unknown dynamic parameters. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. This research is solved this problem by combining sliding mode fuzzy controller and fuzzy-based tuning. It is based on resolve the on line sliding surface gain ($\boldsymbol{\lambda}$) as well as improve the output performance. The sliding surface gain ($\boldsymbol{\lambda}$) of this controller is adjusted online depending on the last values of error (e) and change of error ($\dot{e}$) by sliding surface slope updating factor (α). Fuzzy-based tuning sliding mode fuzzy controller is stable model-free controller which does not need to limits the dynamic model of robot manipulator and eliminate the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in fuzzy-based tuning sliding mode fuzzy controller based on switching (sign) function.

Paper Outline

Section 2, is served as an introduction to the sliding mode controller formulation algorithm and its application to control of robot manipulator, dynamic of robot manipulator and proof of stability. Part 3, introduces and describes the methodology (design fuzzy-based tuningerror-based sliding mode fuzzy controller) algorithms and proves Lyapunov stability. Section 4 presents the simulation results and discussion of this algorithm applied to a robot arm and the final section is describing the conclusion.

2. THEOREM

Dynamic formulation: The equation of an *n-DOF* robot manipulator governed by the following equation [1, 4, 15-29, 63-74]:

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

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 [1-29]:

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

Where B(q) is the matrix of coriolios torques, C(q) is the matrix of centrifugal torques, and G(q) is the vector of gravity force. The dynamic terms in equation (2) are only manipulator position. This is a decoupled system with simple second order linear differential dynamics. In other words, the component $\ddot{q}$ influences, with a double integrator relationship, only the joint variableq_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3, 41-62]:

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

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

Sliding mode methodology: Consider a nonlinear single input dynamic system is defined by [6]:

$$\boldsymbol{x^{(n)} = f(\vec{x}) + b(\vec{x})u} \qquad \textbf{(4)}$$

Where u is the vector of control input, $\boldsymbol{x^{(n)}}$ is the $\boldsymbol{n^{th}}$ derivation of $\boldsymbol{x}$, $\boldsymbol{x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T}$ is the state vector, $\boldsymbol{f(x)}$ is unknown or uncertainty, and $\boldsymbol{b(x)}$ is of known *sign* function. The main goal to design this controller is train to the desired state; $\boldsymbol{x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d{}^{(n-1)}]^T}$, and trucking error vector is defined by [6]:

$$\boldsymbol{\widetilde{x} = x - x_d = [\widetilde{x}, \dots, \widetilde{x}^{(n-1)}]^T} \qquad \textbf{(5)}$$

A time-varying sliding surface $\boldsymbol{s(x,t)}$ in the state space $\boldsymbol{R^n}$ is given by [6]:

$$\boldsymbol{s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\widetilde{x} = 0} \qquad \textbf{(6)}$$

where λ is the positive constant. To further penalize tracking error, integral part can be used in sliding surface part as follows [6]:

$$\boldsymbol{s(x,t)} = (\frac{d}{dt} + \boldsymbol{\lambda})^{\boldsymbol{n-1}} \left(\int_0^t \boldsymbol{\widetilde{x}\,dt}\right) = \boldsymbol{0} \qquad \textbf{(7)}$$

The main target in this methodology is kept the sliding surface slope $\boldsymbol{s(x,t)}$ near to the zero. Therefore, one of the common strategies is to find input $\boldsymbol{U}$ outside of $\boldsymbol{s(x,t)}$ [6].

$$\boldsymbol{\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)|} \qquad \textbf{(8)}$$

where ζ is positive constant.

$$\text{If } \mathbf{S(0)>0} \rightarrow \frac{\mathbf{d}}{\mathbf{dt}}\mathbf{S(t)} \le -\boldsymbol{\zeta} \qquad \textbf{(9)}$$

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

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt} S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S\,(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \tag{10}$$

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

$$0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta} \tag{11}$$

and

$$if\ S(0) < 0 \rightarrow 0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta} \tag{12}$$

Equation (12) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0 \tag{13}$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt} + \lambda) \quad \tilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \tag{14}$$

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

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

suppose the second order system is defined as;

$$\ddot{x} = f + u \ \rightarrow \dot{S} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \tag{16}$$

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

$$\widehat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \tag{17}$$

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

$$U_{dis} = \widehat{U} - K(\vec{x}, t) \cdot \mathrm{sgn}(s) \tag{18}$$

where the switching function $\mathbf{sgn}(S)$ is defined as [1, 6]

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{19}$$

and the $K(\vec{x}, t)$ is the positive constant. Suppose by (8) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt} s^2(x,t) = \dot{S} \cdot S = \left[f - \hat{f} - K\mathrm{sgn}(s)\right] \cdot S = \left(f - \hat{f}\right) \cdot S - K|S| \tag{20}$$

and if the equation (12) instead of (11) the sliding surface can be calculated as

$$s(x,t) = (\frac{d}{dt} + \lambda)^2 \left(\int_0^t \tilde{x}\, dt\right) = (\dot{x} - \dot{x}_d) + 2\lambda(\dot{x} - \dot{x}_d) - \lambda^2(x - x_d) \tag{21}$$

in this method the approximation of U is computed as [6]

$$\widehat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{x} - \dot{x}_d) + \lambda^2(x - x_d) \tag{22}$$

Based on above discussion, the sliding mode control law for multi degrees of freedom robot manipulator is written as [1, 6]:

$$\tau = \tau_{eq} + \tau_{dis} \tag{23}$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems and τ_{eq} for first 3 DOF PUMA robot manipulator can be calculate as follows [1]:

$$\tau_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{24}$$

and τ_{dis} is computed as [1];

$$\tau_{dis} = K \cdot \mathrm{sgn}(S) \quad (25)$$

by replace the formulation (25) in (23) the control output can be written as;

$$\tau = \tau_{eq} + K.\mathrm{sgn}(S) \quad (26)$$

By (26) and (24) the sliding mode control of PUMA 560 robot manipulator is calculated as;

$$\tau = \left[M^{-1}(B + C + G) + \dot{S}\right]M + K \cdot \mathrm{sgn}(S) \quad (27)$$

where $S = \lambda e + \dot{e}$ in PD-SMC and $S = \lambda e + \dot{e} + (\frac{\lambda}{2})^2 \sum e$ in PID-SMC.

Proof of Stability: the lyapunov formulation can be written as follows,

$$V = \frac{1}{2} S^T.M.S \quad (28)$$

the derivation of V can be determined as,

$$\dot{V} = \frac{1}{2} S^T.\dot{M}.S + S^T\, M\dot{S} \quad (29)$$

the dynamic equation of IC engine can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + B + C + G \quad (30)$$

it is assumed that

$$S^T\left(\dot{M} - 2B + C + G\right)S = 0 \quad (31)$$

by substituting (30) in (29)

$$\dot{V} = \frac{1}{2} S^T\dot{M}S - S^T B + CS + S^T\left(M\dot{S} + B + CS + G\right) = S^T\left(M\dot{S} + B + CS + G\right) \quad (32)$$

suppose the control input is written as follows

$$\hat{U} = \widehat{U_{Nonlinear}} + \widehat{U_{dis}} = \left[\widehat{M^{-1}}(B + C + G) + \dot{S}\right]\widehat{M} + K.sgn(S) + B + CS + G \quad (33)$$

by replacing the equation (33) in (32)

$$\dot{V} = S^T(M\dot{S} + B + C + G - \widehat{M}\dot{S} - \widehat{B + CS} + G - Ksgn(S)) = S^T\left(\widetilde{M}\dot{S} + \widetilde{B + CS} + G - Ksgn(S)\right) \quad (34)$$

it is obvious that

$$\left|\widetilde{M}\dot{S} + \widetilde{B + CS} + G\right| \le \left|\widetilde{M}\dot{S}\right| + \left|\widetilde{B + CS} + G\right| \quad (35)$$

the Lemma equation in robot arm system can be written as follows

$$K_u = \left[\left|\widetilde{M}\dot{S}\right| + |B + CS + G| + \eta\right]_i, i = 1, 2, 3, 4, \dots \quad (36)$$

the equation (11) can be written as

$$K_u \ge \left|\left[\widetilde{M}\dot{S} + B + CS + G\right]_i\right| + \eta_i \quad (37)$$

therefore, it can be shown that

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i\, |S_i| \quad (38)$$

Consequently the equation (38) guaranties the stability of the Lyapunov equation. Figure 1 is shown pure sliding mode controller, applied to robot arm.

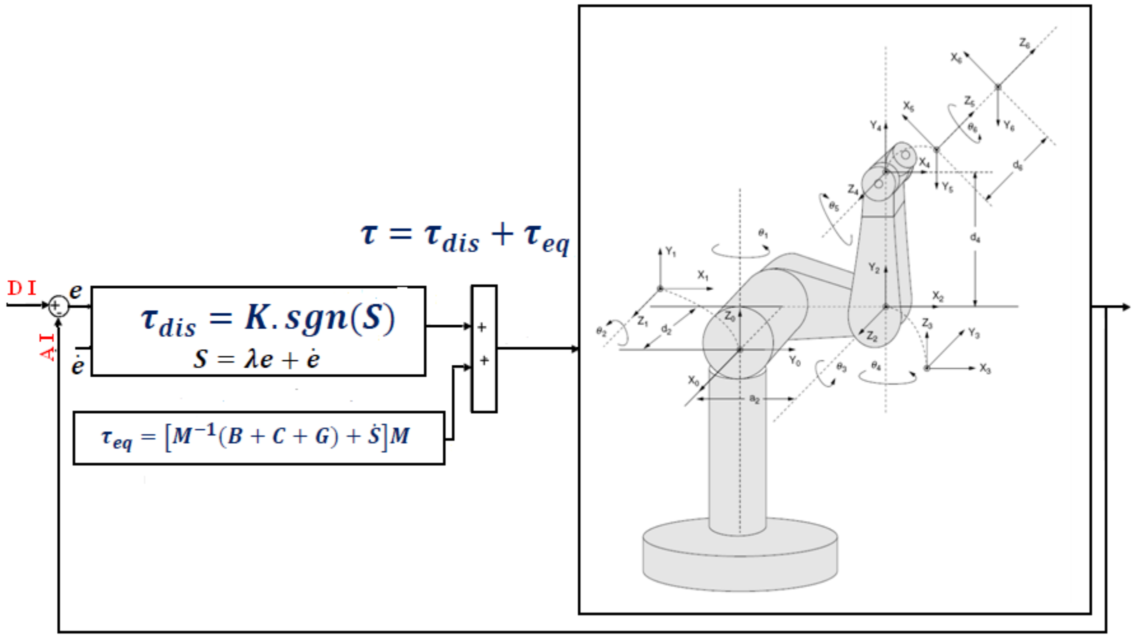


FIGURE 1: Block diagram of a sliding mode controller: applied to robot arm

3. METHODOLOGY

As shown in Figure 1, sliding mode controller is divided into two main parts: discontinuous part and equivalent part. Discontinuous part is based on switching function which this method is used to good following trajectory. Equivalent part is based on robot manipulator's dynamic formulation which these formulations are nonlinear; MIMO and some of them are unknown. Equivalent part of sliding mode controller is based on nonlinear dynamic formulations of robot manipulator. Robot manipulator's dynamic formulations are highly nonlinear and some of parameters are unknown therefore design a controller based on dynamic formulation is complicated. To solve this challenge fuzzy logic methodology is applied to sliding mode controller. Based on literature [43-44, 58-61], most of researchers are designed fuzzy model-based sliding mode controller and model-based sliding mode fuzzy controller. In this research fuzzy logic method is applied to SMC to reduce the fuzzy rule base, improve the stability and robustness. Figure 2 shows sliding mode fuzzy controller.

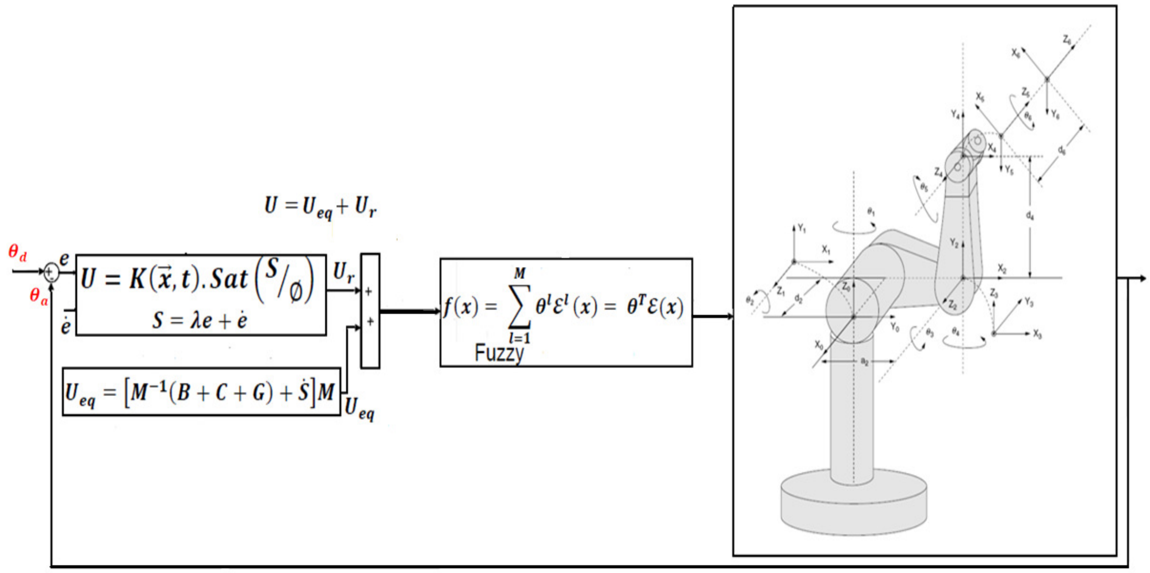


FIGURE 2: Sliding Mode Fuzzy Controller

To solve the challenge of sliding mode controller based on nonlinear dynamic formulation this research is focused on estimate the nonlinear equivalent formulation based on fuzzy logic methodology in feed forward way in this system. In this method; dynamic nonlinear equivalent part is estimated by performance/error-based fuzzy logic controller. In sliding mode fuzzy controller; error based Mamdani's fuzzy inference system has considered with one input, one output and totally 5 rules to estimate the dynamic equivalent part. In this method a model free Mamdani's fuzzy inference system has considered based on error-based fuzzy logic controller to estimate the nonlinear equivalent part. For both sliding mode controller and sliding mode fuzzy controller applications the system performance is sensitive to the sliding surface slope coefficient(λ). For instance, if large value of λ is chosen the response is very fast the system is unstable and conversely, if small value of λ is considered the response of system is very slow but system is stable. Therefore to have a good response, compute the best value sliding surface slope coefficient is very important. Eksin et. al [83] have designed mathematical model-free sliding surface slope in fuzzy sliding mode controller. In above method researchers are used saturation function instead of switching function therefore the proof of stability is very difficult. In sliding mode fuzzy controller based on (27) the PD-sliding surface is defined as follows:

$$S = \dot{e} + \lambda_1 e \tag{39}$$

where $\lambda_1 = diag[\lambda_{11}, \lambda_{12}, \lambda_{13}]$. The time derivative of S is computed;

$$\dot{S} = \ddot{q}_d + \lambda_1 \dot{e} \tag{40}$$

Based on Figure 3.5, the fuzzy error-based sliding mode controller's output is written;

$$\hat{\tau} = (\tau_{eq} + \tau_{cont})_{fuzzy\ estimator} \tag{41}$$

Based on fuzzy logic methodology

$$f(x) = U_{fuzzy} = \sum_{l=1}^{M} \theta^T\, \zeta(x) \tag{42}$$

where θ^T is adjustable parameter (gain updating factor) and $\zeta(x)$ is defined by;

$$\zeta(x) = \frac{\sum_i \mu(x_i) x_i}{\sum_i \mu(x_i)} \tag{43}$$

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

$$\tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \left[M^{-1}(B + C + G) + \dot{S}\right] M \text{ +K.sgn (S)} \quad \textbf{(44)}$$

Based on [80-81] to compute dynamic parameters of PUMA560;

$$\tau_{fuzzy} = \begin{bmatrix} \tau_{1fuzzy} \\ \tau_{2fuzzy} \\ \tau_{3fuzzy} \end{bmatrix}, \; M^{-1} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix}^{-1}$$

$$B + C + G = \begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2^{\,2} + C_{13}\dot{q}_3^{\,2} \\ C_{21}\dot{q}_1^{\,2} + C_{23}\dot{q}_3^{\,2} \\ C_{31}\dot{q}_1^{\,2} + C_{32}\dot{q}_2^{\,2} \\ 0 \\ C_{51}\dot{q}_1^{\,2} + C_{52}\dot{q}_2^{\,2} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix}$$

$$\dot{S} = \begin{bmatrix} \dot{S}_1 \\ \dot{S}_2 \\ \dot{S}_3 \\ \dot{S}_4 \\ \dot{S}_5 \\ \dot{S}_6 \end{bmatrix} \text{ and } M = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix}$$

Therefore, the error-based fuzzy sliding mode controller for PUMA robot manipulator is calculated by the following equation;

$$\begin{bmatrix} \widehat{\tau_1} \\ \widehat{\tau_2} \\ \widehat{\tau_3} \end{bmatrix} = \left(\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} + \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \; \mathbf{sgn} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} \right)_{fuzzy\ estimate} \quad \textbf{(45)}$$

As mentioned in Figure 2, the design of error-based fuzzy to estimate the equivalent part based on Mamdani's fuzzy inference method has four steps, namely, fuzzification, fuzzy rule base and rule evaluation, aggregation of the rule output (fuzzy inference system) and defuzzification. In most of industrial robot manipulators, controllers are still usually classical linear, but the manipulator dynamics is highly nonlinear and have uncertain or variation in parameters (e.g., structure and unstructured), as a result design a classical linear controllers for this system is very difficult and sometimes impossible. The first solution is to make the robust algorithm in order to reduce the uncertainty problems in a limit variation (e.g., sliding mode controller and computed torque like controller). Conversely the first solution is used in many applications it has some limitations such as nonlinear dynamic part in controller. The second solution is applied artificial intelligence method (e.g., fuzzy logic) in conventional nonlinear method to reduce or eliminate the challenges. However the second solution is a superior to reduce or eliminate the dynamic nonlinear part with respect to have stability and fairly good robustness but it has a robust in a limit variation. The third solution is used the on-line sliding mode fuzzy controller (e.g., fuzzy-based tuning sliding surface slope in sliding mode fuzzy controller). Adaptive (on-line) control is used in systems whose dynamic parameters are varying and need to be training on line. Sliding mode fuzzy controller has difficulty in handling unstructured model uncertainties and this controller's performance is sensitive to sliding surface slope coefficient. It is possible to solve above challenge by combining fuzzy-based tuning method and sliding mode fuzzy controller which this methodology can help to improve system's tracking performance by on-line tuning (fuzzy-based tuning) method. Based on above discussion, compute the best value of sliding surface slope coefficient has played important role to improve system's tracking performance especially the system parameters are unknown or uncertain. This problem is solved by tuning the surface slope coefficient ($\boldsymbol{\lambda}$) of the error-based fuzzy sliding mode controller continuously in real-time. In this methodology, the system's performance is improved with respect to the classical sliding mode controller and sliding mode fuzzy controller. Figure 3 shows the fuzzy-based tuning sliding mode fuzzy controller.

Based on (44) to adjust the sliding surface slope coefficient we define $\hat{f}(x|\lambda)$ as the fuzzy based tuning.

$$\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) = \boldsymbol{\lambda}^{T}\boldsymbol{\zeta}(\boldsymbol{x}) \quad \textbf{(46)}$$

If minimum error ($\boldsymbol{\lambda}^*$) is defined by;

$$\boldsymbol{\lambda}^* = \boldsymbol{arg\ min}\ [(\boldsymbol{Sup}|\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{x}))] \quad \textbf{(47)}$$

Where λ^T is adjusted by an adaption law and this law is designed to minimize the error's parameters of $\boldsymbol{\lambda} - \boldsymbol{\lambda}^*$. adaption law in fuzzy-based tuning sliding mode fuzzy controller is used to adjust the sliding surface slope coefficient. Fuzzy-based tuning part is a supervisory controller based on Mamdani's fuzzy logic methodology. This controller has two inputs namely; error (e) and change of error $(\dot{e})$ and an output namely; gain updating factor(α). As a summary design a fuzzy-based tuning based on fuzzy logic method in fuzzy based tuning sliding mode fuzzy controller has five steps:

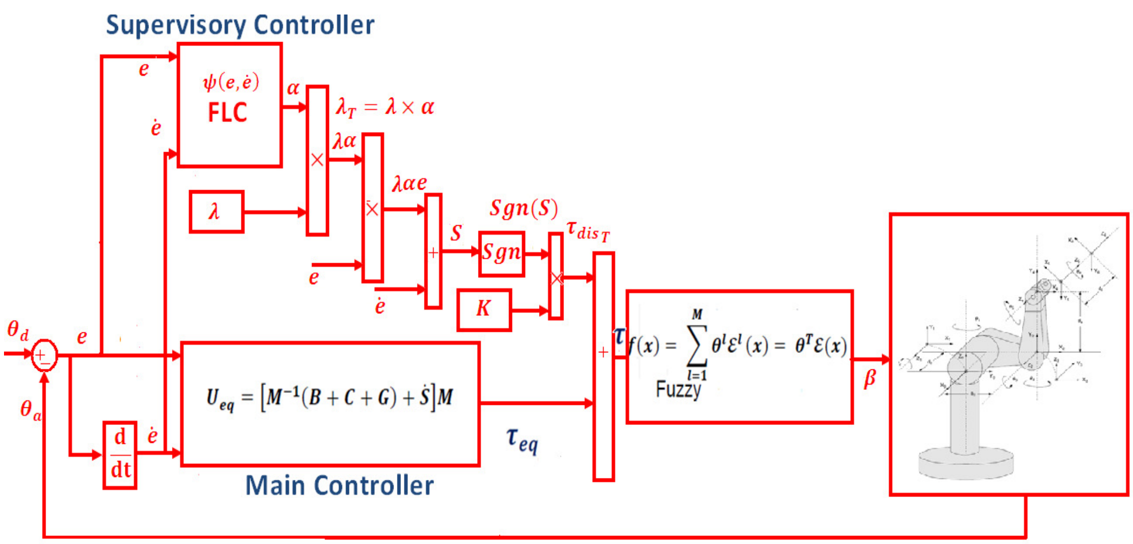


FIGURE 3 : Fuzzy based tuning Sliding Mode Fuzzy Controller

1. Determine Inputs and Outputs

it has two inputs error and change of error $(e, \dot{e})$ and the output name's is sliding surface slope updating factor (α).

2. Find linguistic Variable

The linguistic variables for error(e) are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and it is quantized into thirteen levels represented by: -1, -0.83, -0.66, -0.5, -0.33, -0.16, 0, 0.16, 0.33, 0.5, 0.66, 0.83, 1, the linguistic variables for change of error($\dot{e}$) are ;Fast Left (FL), Medium Left (ML), Slow Left (SL),Zero (Z), Slow Right (SR), Medium Right (MR), Fast Right (FR), and it is quantized in to thirteen levels represented by: -6, -5, -0.4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, and the linguistic variables for sliding surface slope updating factor (α) are; Zero (ZE), Very Small (VS), Small (S), Small Big (SB), Medium Big (MB), Big (B), and Very Big (VB) and they are defined on $[0.5,1]$ and quantized into thirteen levels respected by: 0.5, 0.5417, 0.583, 0.625, 0.667, 0.7087, 0.7503, 0.792, 0.834, 0.876, 0.917, 0.959, 1.

3. Type of membership function: In this research triangular membership function is selected because it has linear equation with regard to has a high-quality response.

4. Design fuzzy rule table: the rule base for sliding surface slope updating factor of fuzzy-based tuning error-based fuzzy sliding mode controller is based on

F.R[1]: *IF e is NB and $\dot{e}$ is NB, THEN α is VB.* **(48)**

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

5. Defuzzification: COG method is used to defuzzification in this research.

e \ $\dot{e}$	FL	ML	SL	Z	SR	MR	FR
NB	VB	VB	VB	B	SB	S	ZE
NM	VB	VB	B	B	MB	S	VS
NS	VB	MB	B	VB	VS	S	VS
Z	S	SB	MB	ZE	MB	SB	S
PS	VS	S	VS	VB	B	MB	VB
PM	VS	S	MB	B	B	VB	VB
PB	ZE	S	SB	B	VB	VB	VB

TABLE 1: Fuzzy rule base for sliding surface slope updating factor (α)

Based on Figure 3, supervisory controller is a controller to solve the unstructured uncertainties and tuning the sliding surface slope coefficient. This controller consists of two parts: fuzzy logic controller and scaling factor. Fuzzy logic controller is a Mamdani's error base inference system which has error (e) and change of error ($\dot{e}$) as inputs and sliding surface slope updating factor (α) as output. Each inputs has seven linguistic variables thus the controller's output has 49 rules, the output is defined between [0.5 1] and it is quantized into thirteen levels. Sliding surface slope updating factor (α) is used to tuning the main controller to give the best possible results. It is required because the robot manipulator's dynamic equations are highly nonlinear, the rules formulated in fuzzy sliding mode controller through user experience are not always correct under defined and also to unstructured uncertainties. It is independent of robot manipulator dynamic parameters and depends only on current system's performance; it is based on error and change of error. In this method the actual sliding surface slope coefficient ($\boldsymbol{\lambda}$) is obtained by multiplying the sliding surface with sliding surface slope updating factor (α). The sliding surface slope updating factor (α) is calculated on-line by 49 rules Mamdani's error-based fuzzy logic methodology. To limitation the error between [-1 1] and change of error between [-6 6], the best values for scaling factors are; $K_\alpha = 1.5$ and $K_\beta = 3$ based on Table 2.

k_α	k_β	e_{band}	$\dot{e}_{band}$	α_{band}
0.25	0.5	$[-0.166 \quad 0.18]$	$[-0.98 \quad 1.1]$	$[-0.1 \quad 0.1]$
0.5	1	$[-0.332 \quad 0.37]$	$[-1.96 \quad 2.2]$	$[-0.05 \quad 0.2]$
0.75	1.5	$[-0.498 \quad 0.54]$	$[-2.94 \quad 3.3]$	$[0 \quad 0.3]$
1	2	$[-0.664 \quad 0.733]$	$[-3.92 \quad 4.4]$	$[0.12 \quad 0.38]$
1.25	2.5	$[-0.83 \quad 0.9]$	$[-4.9 \quad 5.5]$	$[0.3 \quad 0.85]$
1.5	3	$[-0.995 \quad 1.1]$	$[-5.88 \quad 6.6]$	$[0.49 \quad 1.01]$
1.75	3.5	$[-1.162 \quad 1.26]$	$[-6.86 \quad 7.7]$	$[0.58 \quad 1.38]$
2	4	$[-1.328 \quad 1.44]$	$[-7.84 \quad 8.8]$	$[0.63 \quad 1.6]$

TABLE 2: Best value of k_α and k_β to tuning the α

Table 3 shows the sliding surface slope updating factor (α) lookup table in fuzzy-based tuning part by COG defuzzification method. It has 169 cells to shows the fuzzy-based tuning to on-line tuning the sliding surface slope coefficient. For instance if $e = -1$ and $\dot{e} = -3.92$ then the output=0.5. Based on Table 3.3 if two fuzzy rules are defined by

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

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

If all input fuzzy activated by crisp input values $e = -1$ and $\dot{e} = -3.92$ and fuzzy set to compute NB, ML and FL are defined as

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

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

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

$\dot{e}$ → / e ↓	**Membership Function (τ)**												
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-1	-92	-92	-92	-92	-90	-90	-80	-77	-70	-55	-50	-20	0
-0.83	-92	-92	-92	-92	-80	-77	70	-55	-50	-30	-25	0	0
-0.66	-92	-92	-92	-80	-77	-77	-70	-55	-50	-25	0	0	25
-0.5	-92	-92	-71	-57	-43	-28	-28	-28	-25	0	0	25	43
-0.33	-92	-71	-57	-43	-28	-25	-25	-25	0	0	25	43	57
-0.16	-71	-57	-43	-28	-25	-25	0	0	0	14	28	43	57
0	-71	-57	-43	-28	-25	0	0	0	14	28	43	43	71
0.16	-43	-28	-25	0	0	0	0	14	14	28	43	57	70
0.33	-28	-25	0	0	0	14	14	14	28	43	57	71	84
0.5	-25	0	0	0	14	28	28	28	43	57	70	84	92
0.66	0	0	0	14	28	43	43	43	57	70	84	92	92
0.83	0	0	14	28	43	57	57	57	70	92	92	92	92
1	0	14	28	43	57	71	92	92	92	92	92	92	92

TABLE 3: Sliding surface slope updating factor (α): Fuzzy-based tuning fuzzy sliding mode controller lookup table by COG method

while $\alpha_{(VB)} = \{(0,0.4165),(0.25,0.4403),(0.5,0.4641),(0.75,0.4879),(1,0.5),$
$(0.75,0.5238),(0.5,0.5476),(0.25,0.5714),(0,0.5834)\}$

In this controller AND fuzzy operation is used therefore the output fuzzy set is calculated by using individual rule-base inference. The activation degrees is computed as

Table 4 shows the fuzzy equivalent torque performance ($\tau_{eq\ fuzzy}$) lookup table in fuzzy-based tuning error-based fuzzy sliding mode controller by COG defuzzification method.

$\dot{e}$ → / e ↓	Membership Function												
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-1	-85	-85	-85	-85	-84	-84	-83	-71	-57	-43	-28	-14	0
-0.83	-85	-85	-85	-85	-71	-57	-56	-57	-43	-28	-14	0	0
-0.66	-85	-85	-85	-71	-57	-43	-43	-43	-28	-14	0	0	14
-0.5	-85	-85	-71	-57	-43	-28	-28	-28	-14	0	0	14	28
-0.33	-85	-71	-57	-43	-28	-14	-14	-14	0	0	14	28	43
-0.16	-71	-57	-43	-28	-14	-14	0	0	0	14	28	43	57
0	-71	-57	-43	-28	-14	0	0	0	14	28	43	43	71
0.16	-43	-28	-14	0	0	0	0	14	14	28	43	57	70
0.33	-28	-14	0	0	0	14	14	14	28	43	57	71	84
0.5	-14	0	0	0	14	28	28	28	43	57	70	84	85
0.66	0	0	0	14	28	43	43	43	57	70	84	85	85
0.83	0	0	14	28	43	57	57	57	70	85	85	85	85
1	0	14	28	43	57	71	85	85	85	85	85	85	85

TABLE 4: Fuzzy equivalent torque performance ($\tau_{eq\,fuzzy}$): Fuzzy-based tuning fuzzy sliding mode controller lookup table by COG method

$$\mu_{FR_1} = \min[\mu_{e(N.B)}(-1), \mu_{\dot{e}(M.L)}(-3.92)] = \min[1,0.75] = 0.75$$
$$\mu_{FR_2} = \min[\mu_{e(N.B)}(-1), \mu_{\dot{e}(F.L)}(-3.92)] = \min[1,0.5] = 0.5$$

The activation degrees of the consequent parts for $F.R^1$ and $F.R^2$ are computed as:

$$\mu_{FR_1}(-1,-3.92,\alpha) = \min[\mu_{FR_1}(-1,-3.92), \mu_{\alpha(VB)}] = \min[0.75, \mu_{\alpha(VB)}]$$
$$\mu_{FR_2}(-1,-3.92,\alpha) = \min[\mu_{FR_2}(-1,-3.92), \mu_{\alpha(VB)}] = \min[0.5, \mu_{\alpha(VB)}]$$

Fuzzy set $\alpha_{L.L(1)}$ and $\alpha_{L.L(2)}$ have nine elements:

$$F.F^1(-1,-3.92,\alpha) = \{(0,0.4165),(0.25,0.4403),(0.5,0.4641),(0.75,0.4879),(1,0.5),(0.75,0.5238),(0.5,0.5476),(0.25,0.5714),(0,0.5834)\}$$

$$F.F^2(-1,-3.92,T) = \{(0,0.4165),(0.25,0.4403),(0.5,0.4641),(0.75,0.4879),(1,0.5),(0.75,0.5238),(0.5,0.5476),(0.25,0.5714),(0,0.5834)\}$$

Max-min aggregation is used to find the output of fuzzy set:

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

$$U_{12} = \{(0,0.4165),(0.25,0.4403),(0.5,0.4641),(0.75,0.4879),(0.75,0.5),(0.75,0.5238),(0.5,0.5476),(0.25,0.5714),(0,0.5834)\}$$

The COG defuzzification is selected as;

$$COG = (0.25 \times 0.4403) + (0.5 \times 0.4641) + (0.75 \times 0.4879) + (0.75 \times 0.5) + (0.75 \times 0.5238) + (0.5 \times 0.5476) + (0.25 \times 0.5714)[0.25 + 0.5 + 0.75 + 0.75 + 0.75 + 0.5 + 0.25]^{-1} = \frac{1.875}{3.75} = 0.5$$

Based on Tables 1, 3 and 4 where $e = -1$ and $\dot{e} = -3.92$ and fuzzy set to compute NB, ML and FL are defined as

$e_{(NB)} = \{(0,-1.5),(0.25,-1.375),(0.5,-1.25),(0.75,-1.125),(1,-1),(0.75,-0.875),$
$(0.5,-0.75),(0.25,-0.625),(0,-0.5)$
$\dot{e}_{(ML)} = \{(0,-5.8),(0.25,-5.17),(0.5,-4.55),(0.75,-3.92),(1,-3.3),(0.75,-2.67),$
$(0.5,-2.05),(0.25,-1.42),(0,-0.83)$
$\dot{e}_{(FL)} = \{(0,-7.5),(0.25,-6.88),(0.5,-6.25),(0.75,-5.57),(1,-5),(0.75,-4.30),$
$(0.5,-3.92),(0.25,-3.12),(0,-2.5)$
while $T_{(LL)} = \{(0,-123),(0.25,-113.5),(0.5,-104),(0.75,-94.5),(1,-85),(0.75,-75.5),$
$(0.5,-66),(0.25,-56.5),(0,-47)$. Based on 2.63 the activation degrees is computed as

$$\mu_{FR_1} = \min\left[\mu_{e(N.B)}(-1),\mu_{\dot{e}(M.L)}(-3.92)\right] = \min[1,0.75] = 0.75$$
$$\mu_{FR_2} = \min\left[\mu_{e(N.B)}(-1),\mu_{\dot{e}(F.L)}(-3.92)\right] = \min[1,0.5] = 0.5$$

The activation degrees of the consequent parts for $F.R^1$ and $F.R^2$ are computed as:

$$\mu_{FR_1}(-1,-3.92,T) = \min\left[\mu_{FR_1}(-1,-3.92),\mu_{T(L.L)}\right] = \min\left[0.75,\mu_{T(L.L)}\right]$$
$$\mu_{FR_2}(-1,-3.92,T) = \min\left[\mu_{FR_2}(-1,-3.92),\mu_{T(L.L)}\right] = \min\left[0.5,\mu_{T(L.L)}\right]$$

Fuzzy set $T_{L.L(1)}$ and $T_{L.L(2)}$ have nine elements:

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

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

Based on 2.69, Max-min aggregation is used to find the output of fuzzy set:

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

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

Based on (2.71) the COG defuzzification is selected as;

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

Based on Figures 3.8, torque performance is calculated by;

$$\boldsymbol{\tau} = [\boldsymbol{\tau}_{\boldsymbol{equ}} + (\boldsymbol{\tau}_{\boldsymbol{dis}})_{\boldsymbol{tuning}}]_{\boldsymbol{fuzzy}} \tag{49}$$

where $(\tau_{dis})_{tuning}$ is a discontinuous part which tuning by fuzzy-based tuning method. Table 5 shows the torque performance in fuzzy-based tuning error-based fuzzy sliding mode controller look up table.

$\dot{e}$ → / e ↓	**Membership Function (τ)**												
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-1	-92	-92	-92	-92	-90	-90	-80	-77	-70	-55	-50	-20	0
-0.83	-92	-92	-92	-92	-80	-77	70	-55	-50	-30	-25	0	0
-0.66	-92	-92	-92	-80	-77	-77	-70	-55	-50	-25	0	0	25
-0.5	-92	-92	-71	-57	-43	-28	-28	-28	-25	0	0	25	43
-0.33	-92	-71	-57	-43	-28	-25	-25	-25	0	0	25	43	57
-0.16	-71	-57	-43	-28	-25	-25	0	0	0	14	28	43	57
0	-71	-57	-43	-28	-25	0	0	0	14	28	43	43	71
0.16	-43	-28	-25	0	0	0	0	14	14	28	43	57	70
0.33	-28	-25	0	0	0	14	14	14	28	43	57	71	84
0.5	-25	0	0	0	14	28	28	28	43	57	70	84	92
0.66	0	0	0	14	28	43	43	43	57	70	84	92	92
0.83	0	0	14	28	43	57	57	57	70	92	92	92	92
1	0	14	28	43	57	71	92	92	92	92	92	92	92

TABLE 5: Torque (τ) performance: Fuzzy-based tuning fuzzy sliding mode controller lookup table by COG method

Based on Figure 5, fuzzy-based tuning error-based fuzzy sliding mode controller for PUMA560 robot manipulator is calculated by the following equation;

$$\begin{bmatrix}\widehat{\tau_1}\\ \widehat{\tau_2}\\ \widehat{\tau_3}\end{bmatrix} = [\begin{bmatrix}\tau_{1eq}\\ \tau_{2eq}\\ \tau_{3eq}\end{bmatrix} + \begin{bmatrix}\lambda_1\times\alpha_1\\ \lambda_2\times\alpha_2\\ \lambda_3\times\alpha_3\end{bmatrix}\begin{bmatrix}K_1\\ K_2\\ K_3\end{bmatrix} sgn\begin{bmatrix}S_1\\ S_2\\ S_3\end{bmatrix}]_{fuzzy} \quad \textbf{(50)}$$

Where $\alpha = [\alpha_1, \alpha_2, \alpha_3]$ is sliding surface slope updating factor and it is calculated based on error-based fuzzy logic methodology.

4. RESULTS AND DISCUSSION

Sliding mode controller (SMC), sliding mode fuzzy controller (SMFC) and fuzzy-based tuning sliding mode fuzzy controller (FTSMFC) were tested to Step response trajectory. In this simulation, to control position of PUMA robot manipulator 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. These controllers 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 and applied to nonlinear dynamic of these controllers. Pure sliding mode controller and sliding mode fuzzy controller have difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode fuzzy controller and fuzzy-based tuning in a single controller method. This method can improve the system's tracking performance by online tuning method. This method is based on resolve the on line sliding surface slope as well as improve the output performance by tuning the sliding surface slope coefficient. The sliding surface gain ($\boldsymbol{\lambda}$) of this controller is adjusted online depending on the last values of error ($\boldsymbol{e}$) and change of error ($\dot{\boldsymbol{e}}$) by sliding surface slope updating factor ($\boldsymbol{\alpha}$). Fuzzy-based tuning sliding mode fuzzy controller is stable model-based controller which does not need to limits the dynamic model of robot manipulator and eliminate the chattering phenomenon without to use the boundary layer saturation function.

Tracking performances: Based on (44) in sliding mode fuzzy controller and based on (10) in sliding mode controller; controllers performance are depended on the gain updating factor (K) and sliding surface slope coefficient (λ). These two coefficients are computed by trial and error in PD-SMC and FSMC. The best possible coefficients in step SMFC are; $K_p = K_v = K_i = 18,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 3, \lambda_2 = 6, \lambda_3 = 6$ and the best possible coefficients in step SMC are; $\lambda_1 = 1\ , \lambda_2 = 6, \lambda_3 = 8$; $K_p = K_v = K_i = 10$; $\emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1$. In fuzzy-based tuning sliding mode fuzzy controller the sliding surface gain is adjusted online depending on the last values of error (e) and change of error $(\dot{e})$ by sliding surface slope updating factor (α). Figure 4 shows tracking performance in fuzzy-based tuning sliding mode fuzzy controller (FTSMFC), sliding mode fuzzy controller (SMFC) and SMC without disturbance for step trajectory.

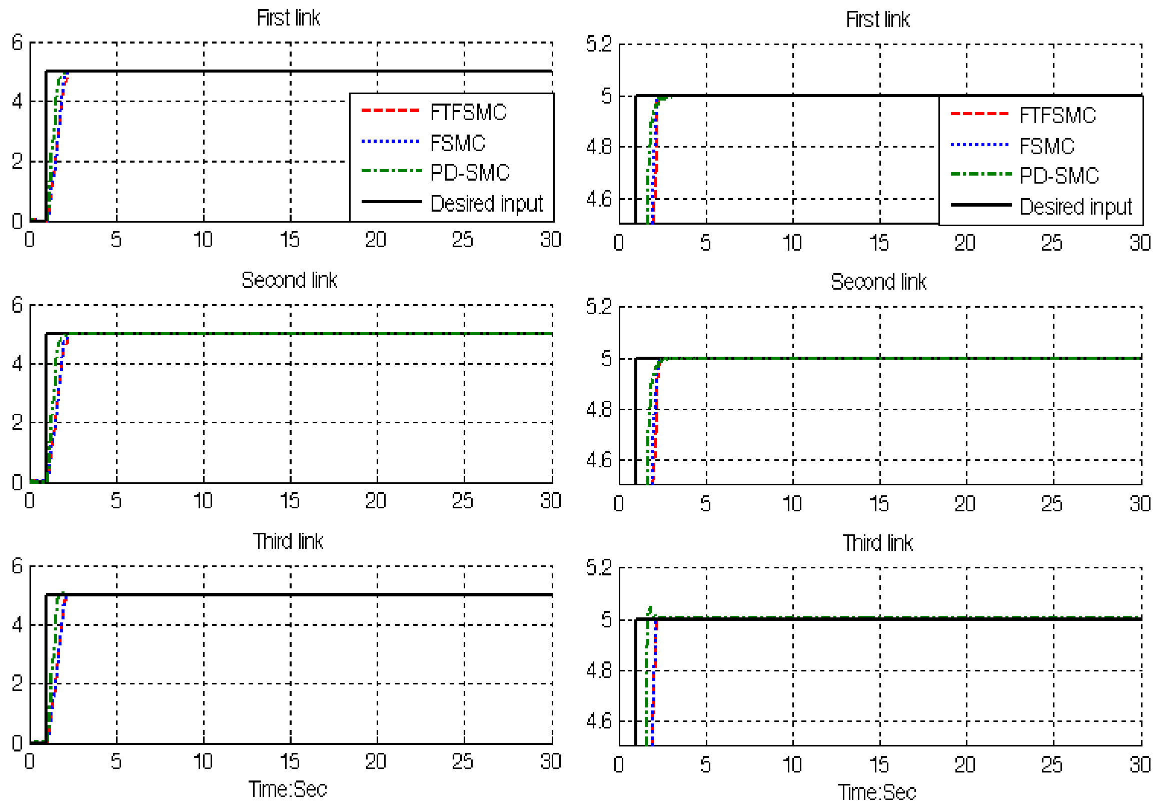


FIGURE 4: SMFC, FTSMFC, desired input and SMC for first, second and third link step trajectory performance without disturbance

Based on Figure 4 it is observed that, the overshoot in FTSMFC is 0%, in SMC's is 1% and in SMFC's is 0%, and rise time in FTSMFC's is 0.6 seconds, in SMC's is 0.483 second and in SMFC's is about 0.6 seconds. From the trajectory MATLAB simulation for FTSMFC, SMC and SMFC in certain system, it was seen that all of three controllers have acceptable performance.

Disturbance Rejection

Figure 5 shows the power disturbance elimination in FTSMFC, SMC and SMFC with disturbance for step trajectory. The disturbance rejection is used to test the robustness comparisons of these three controllers for step trajectory. A band limited white noise with predefined of 40% the power of input signal value is applied to the step trajectory. Based on Figure 5, it was seen that, STSMFC's performance is better than SMFC and SMC because FTSMFC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas SMFC and SMC cannot.

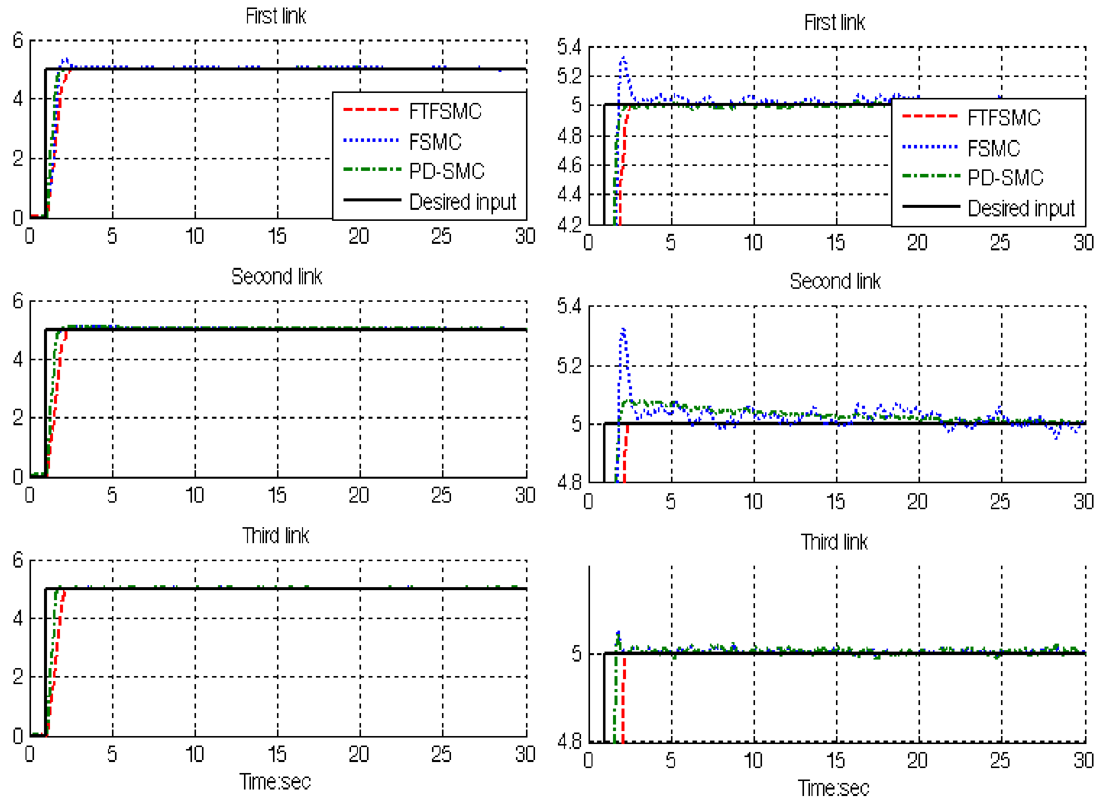


FIGURE 5: Desired input, FTSMFC, SMFC and SMC for first, second and third link trajectory with 40%external disturbance: step trajectory

Based on Figure 5; by comparing step response trajectory with 40% disturbance of relative to the input signal amplitude in FTSMFC, SMC and SMFC, FTSMFC's overshoot about **(0%)** is lower than FTSMFC's **(6%)** and SMC's (**8%**)**.** SMC's rise time **(0.5 seconds)** is lower than SMFC's **(0.7 second)** and FTSMFC's **(0.8 second).** Besides the Steady State and RMS error in FTSMFC, SMFC and SMC it is observed that, error performances in FTSMFC **(Steady State error =1.3e-12 and RMS error=1.8e-12)** are about lower than SMFC **(Steady State error =10e-4 and RMS error=0.69e-4)** and SMC's **(Steady State error=10e-4 and RMS error=11e-4).** Based on Figure 5, SMFC and SMC have moderately oscillation in trajectory response with regard to 40% of the input signal amplitude disturbance but FTSMFC has stability in trajectory responses in presence of uncertainty and external disturbance. Based on Figure 5 in presence of 40% unstructured disturbance, STSMFC's is more robust than SMFC and SMC because FTSMFC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas SMFC and SMC cannot.

Torque Performance

Figures 6 and 7 have indicated the power of chattering rejection in FTSMFC, SMC and SMFC with 40% disturbance and without disturbance.

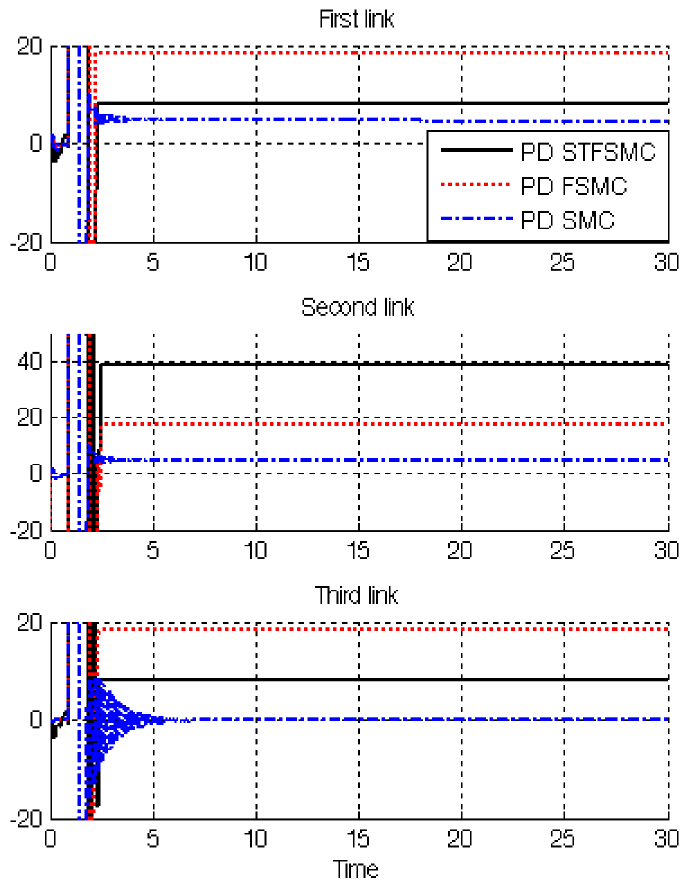


FIGURE 6: FTSMFC, SMC and SMFC for first, second and third link torque performance without disturbance

Figure 6 shows torque performance for first three links PUMA robot manipulator in FTSMFC, SMC and SMFC without disturbance. Based on Figure 6, FTSMFC, SMC and SMFC give considerable torque performance in certain system and all three of controllers eliminate the chattering phenomenon in certain system. Figure 7 has indicated the robustness in torque performance for first three links PUMA robot manipulator in FTSMFC, SMC and SMFC in presence of 40% disturbance. Based on Figure 7, it is observed that SMC and SMFC controllers have oscillation but FTSMFC has steady in torque performance. This is mainly because pure SMC and sliding mode fuzzy controller are robust but they have limitation in presence of external disturbance. The FTSMFC gives significant chattering elimination when compared to SMFC and SMC. This elimination of chattering phenomenon is very significant in presence of 40% disturbance. This challenge is one of the most important objectives in this thesis.

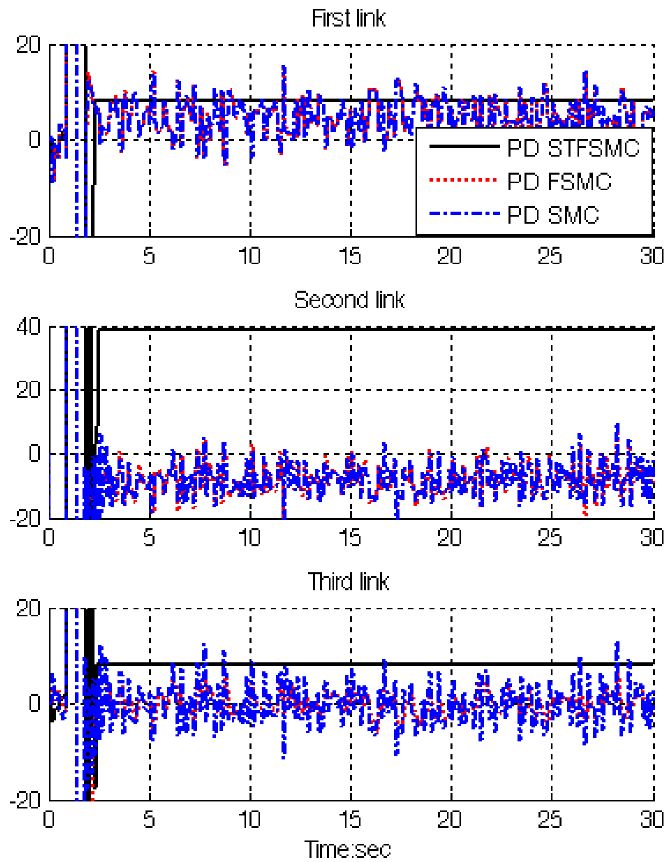


FIGURE 7: FTSMFC, SMC and SMFC for first, second and third link torque performance with40% disturbance

Based on Figure 7 it is observed that, however fuzzy tuning sliding mode fuzzy controller (FTSMFC) is a model-based controller that estimate the nonlinear dynamic equivalent formulation by fuzzy rule base but it has significant torque performance (chattering phenomenon) in presence of uncertainty and external disturbance. SMC and SMFC have limitation to eliminate the chattering in presence of highly external disturbance (e.g., 40% disturbance) but FTSMFC is a robust against to highly external disturbance.

Steady state error: Figure 8 is shown the error performance in FTSMFC, SMC and SMFC for first three links of PUMA robot manipulator**.** The error performance is used to test the disturbance effect comparisons of these controllers for step trajectory. All three joint's inputs are step function with the same step time (step time= 1 second), the same initial value (initial value=0) and the same final value (final value=5). Based on Figure 5, FTSMFC's rise time is about 0.6 second, SMC's rise time is about 0.483 second and SMFC's rise time is about 0.6 second which caused to create a needle wave in the range of 5 (amplitude=5) and the different width. In this system this time is transient time and this part of error introduced as a transient error. Besides the Steady State and RMS error in STSMFC, SMFC and SMC it is observed that, error performances in FTSMFC **(Steady State error =0.9e-12 and RMS error=1.1e-12)** are bout lower than SMFC **(Steady State error =0.7e-8 and RMS error=1e-7)** and SMC's **(Steady State error=1e-8 and RMS error=1.2e-6)**.

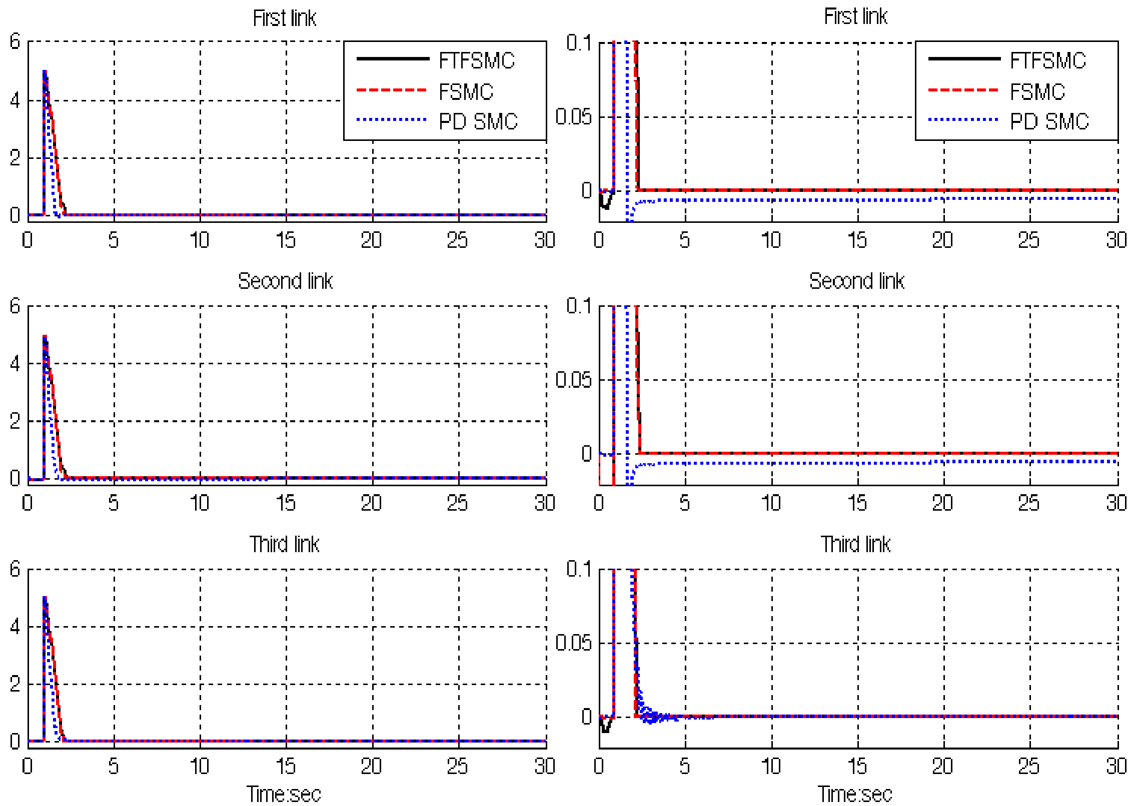


FIGURE 8: FTSMFC, SMC and SMFC for first, second and third link steady state error without disturbance: step trajectory

The FTSMFC gives significant steady state error performance when compared to SMFC and SMC. When applied 40% disturbances in FTSMFC the RMS error increased approximately 0.0164% (percent of increase the FTSMFC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{1.8e-12}{1.1e-12} = 0.0164\%$), in SMFC the RMS error increased approximately 6.9% (percent of increase the SMFC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{0.69e-4}{1e-7} = 6.9\%$)in SMC the RMS error increased approximately 9.17% (percent of increase the SMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbance\ RMS\ error} = \frac{11e-4}{1.2e-6} = 9.17\%$). In this part FTSMFC, SMC and SMFC have been comparatively evaluation through MATLAB simulation, for PUMA robot manipulator control. It is observed that however FTSMFC is independent of nonlinear dynamic equation of PUMA 560 robot manipulator but it can guarantee the trajectory following and eliminate the chattering phenomenon in certain systems, structure uncertain systems and unstructured model uncertainties by online tuning method.

5. CONCLUSION

Refer to this research, a position fuzzy-based tuning sliding mode fuzzy controller (FTSMFC) is proposed for PUMA robot manipulator. The nonlinear equivalent dynamic problem in uncertain system is estimated by using fuzzy logic theory. To estimate the PUMA robot manipulator system's dynamic, 5 rules Mamdani inference system is design and applied to sliding mode methodology. This methodology is based on applied fuzzy logic in equivalent nonlinear dynamic part to estimate unknown parameters. The results demonstrate that the sliding mode fuzzy controller is a model-based controllers which works well in certain and partly uncertain system. Pure sliding mode controller and sliding mode fuzzy controller have difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode fuzzy controller and fuzzy-based tuning. Since the sliding surface gain (λ) is adjusted by fuzzy-based tuning method, it is nonlinear and continuous. The sliding surface slope updating factor (α) of fuzzy-based tuning part can be changed with the changes in error and change of error rate between half to one. Sliding surface gain is adapted on-line by sliding surface slope updating factor. In pure sliding mode controller and sliding mode fuzzy controller the sliding surface gain is chosen by trial and error, which means pure sliding mode controller and sliding mode fuzzy controller have to have a prior knowledge of the system uncertainty. If the knowledge is not available error performance and chattering phenomenon are go up. The stability and convergence of the fuzzy-based tuning sliding mode fuzzy controller based on switching function is guarantee and proved by the Lyapunov method. The simulation results exhibit that the fuzzy-based tuning sliding mode fuzzy controller works well in various situations. Based on theoretical and simulation results, it is observed that fuzzy-based tuning sliding mode fuzzy controller is a model-free stable control for robot manipulator. It is a best solution to eliminate chattering phenomenon with switching function in structure and unstructured uncertainties.

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] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, *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, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* No. 3, vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, No. 2,* 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," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," *IEEE conference proceeding,* 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE conference proceeding,* 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* No. 1, 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,* No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* No. 2, vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics Journal,* No. 1, 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,* No. 7, 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,* No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," *IEEE conference proceeding,* 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,* No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," *IEEE conference proceeding,* 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, “Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain,” International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] 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.

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, “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.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , “A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator,” World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli “Design Adaptive Fuzzy Robust Controllers for Robot Manipulator,” World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , “Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line Tunable Gain: Applied to Robot Manipulator,” International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, “Design On-Line Tunable Gain Artificial Nonlinear Controller,” Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, “ Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine,” International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, “Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller,” Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, “Novel Sliding Mode Controller for robot manipulator using FPGA,” Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, “Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System,” International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, “Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, “Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System,” International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, "Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method," International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, "Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control," International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, "Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ," International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, "Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm," International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, "Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ," International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, "Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator," International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, "Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator," International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, "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.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, "Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ," International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, "Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ," International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, "On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory," International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, "Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator," International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, “Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine,” International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.,” Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine,” world applied science journal (WASJ), 15 (3): 422-428, 2011

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* No. 1, 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,* No. 2, 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,* No. 2, vol. 19, pp. 362-368, 2004.

[62] N. Sulaiman, Z. A. Obaid, M. Marhaban and M. Hamidon , "Design and Implementation of FPGA-Based Systems-A Review," *Australian Journal of Basic and Applied Sciences,* No. 4, vol. 3, pp. 3575-3596, 2009.

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

[64] Y. S. Kung, K. Tseng, C. Chen, H. Sze and A. Wang, "FPGA-implementation of inverse kinematics and servo controller for robot manipulator," *Proc. IEEE Int. on Robotics and Biomimetics,* pp. 1163–1168, 2006.

[65] X. Shao, D. Sun and J. K. Mills, "A new motion control hardware architecture with FPGA-based IC design for robotic manipulators," *IEEE Conference,* 2006, pp. 3520-3525.

[66] Y. S. Kung, C. S. Chen and G. S. Shu, "Design and Implementation of a Servo System for Robotic Manipulator," CACS, 2005.

[67] U. D. Meshram and R. Harkare, "FPGA Based Five Axis Robot Arm Controller," *IEEE Conference,* 2005, pp. 3520-3525.

[68] U. Meshram, P. Bande, P. Dwaramwar and R. Harkare, "Robot arm controller using FPGA," *IEEE Conference,* 2009, pp. 8-11.

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

[70] Z. A. Obaid, N. Sulaiman and M. Hamidon, "Developed Method of FPGA-based Fuzzy Logic Controller Design with the Aid of Conventional PID Algorithm," *Australian Journal of Basic and Applied Sciences,* No. 3, vol. 3, pp. 2724-2740, 2009.

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

[72] K. D. Rogers, "Acceleration and implementation of a DSP phase-based frequency estimation algorithm: MATLAB/SIMULINK to FPGA via XILINX system generator," Citeseer, 2004.

[73] F. J. Lin, D. H. Wang and P. K. Huang, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive," IEEE journal of electrical power application, No. 5, Vol. 152, 2005, pp. 1137-1148.

[74] R. R. Ramos, D. Biel, E. Fossas and F. Guinjoan, "A fixed-frequency quasi-sliding control algorithm: application to power inverters design by means of FPGA implementation," *Power Electronics, IEEE Transactions on,* No. 1, vol. 18, pp. 344-355, 2003.

[75] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," *Carleton university Ottawa,*2007.

[76] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," *IEEE Conference,* 2004, pp. 3588-3592.

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

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

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

[80] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," *IEEE Conference,* 2002, pp. 510-518.

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

[82] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[83] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* 2002, pp. 160-168.

[84] Samira Soltani & Farzin Piltan, "*Design Artificial Nonlinear Controller Based on Computed Torque like Controller with Tunable Gain*". World Applied Science Journal,14 (9): 1306-1312, 2011.

[85] Farzin Piltan, I. Nazari, S. Siamak, P. Ferdosali ,"Methodology of FPGA-based mathematical error-based tuning sliding mode controller" International Journal of Control and Automation, 5(1): 89-110, 2012.

[86] Farzin Piltan, M. A. Dialame, A. Zare, A. Badri ,"Design Novel Lookup table changed Auto Tuning FSMC: Applied to Robot Manipulator" International Journal of Engineering, 6(1): 25-40, 2012.

[87] Farzin Piltan, B. Boroomand, A. Jahed, H. Rezaie ,"Methodology of Mathematical Error-Based Tuning Sliding Mode Controller" International Journal of Engineering, 6(2): 96-112, 2012.

[88] Farzin Piltan, F. Aghayari, M. R. Rashidian, M. Shamsodini, "A New Estimate Sliding Mode Fuzzy Controller for Robotic Manipulator" International Journal of Robotics and Automation, 3(1): 45-58, 2012.

[89] Farzin Piltan, M. Keshavarz, A. Badri, A. Zargari , "Design novel nonlinear controller applied to robot manipulator: design new feedback linearization fuzzy controller with minimum rule base tuning method" International Journal of Robotics and Automation, 3(1): 1-18, 2012.

[90] 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.

[91] Farzin Piltan , M.H. Yarmahmoudi, M. Shamsodini, E.Mazlomian, A.Hosainpour. " PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[92] Farzin Piltan, R. Bayat, F. Aghayari, B. Boroomand. "Design Error-Based Linear Model-Free Evaluation Performance Computed Torque Controller" International Journal of Robotics and Automation, 3(3): 2012.

[93] Farzin Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari & Mina Mirazaei . " PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[94] Farzin Piltan, J. Meigolinedjad, S. Mehrara, S. Rahmdel. " Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology" International Journal of Robotics and Automation, 3(3): 2012.

[95] Farzin Piltan, S. Rahmdel, S. Mehrara, R. Bayat." Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses" International Journal of Engineering, 3(3): 2012.

[96] Farzin Piltan, M. Mirzaie, F. Shahriyari, Iman Nazari & S. Emamzadeh." Design Baseline Computed Torque Controller" International Journal of Engineering, 3(3): 2012.

PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses

Farzin Piltan *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

Sara Emamzadeh *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 Hivand *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

Forouzan Shahriyari *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

Mina Mirzaei *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 describes the MATLAB/SIMULINK realization of the PUMA 560 robot manipulator position control methodology. This paper focuses on two main areas, namely robot manipulator analysis and implementation, and design, analyzed and implement nonlinear sliding mode control (SMC) methods. These simulation models are developed as a part of a software laboratory to support and enhance graduate/undergraduate robotics courses, nonlinear control courses and MATLAB/SIMULINK courses at research and development company (SSP Co.) research center, Shiraz, Iran.

Keywords: MATLAB/SIMULINK, PUMA 560 Robot Manipulator, Position Control Method, Sliding Mode Control, Robotics, Nonlinear Control.

1. INTRODUCTION

Computer modeling, simulation and implementation tools have been widely used to support and develop nonlinear control, robotics, and MATLAB/SIMULINK courses. MATLAB with its toolboxes such as SIMULINK [1] is one of the most accepted software packages used by researchers to enhance teaching the transient and steady-state characteristics of control and robotic courses [3_7]. In an effort to modeling and implement robotics, nonlinear control and advanced MATLAB/SIMULINK courses at research and development SSP Co., authors have developed MATLAB/SIMULINK models for learn the basic information in field of nonlinear control and industrial robot manipulator [8, 9].

The international organization defines the robot 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 a 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 n degrees of freedom serial link 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 and the axis number seven to n use to reach the avoid the difficult conditions (e.g., surgical robot and space robot manipulator). 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 such as trajectory planning, essential prerequisite for some dynamic description such as Newton's equation for motion of point mass, 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 such as linear or nonlinear dynamic behavior, design of model based controller such as pure sliding mode controller and pure computed torque controller which design these controller are based on nonlinear dynamic equations, and for simulation. The dynamic modeling describes the relationship between joint motion, velocity, and accelerations to force/torque or current/voltage and also it can be used to describe the particular dynamic effects (e.g., inertia, coriolios, centrifugal, and the other parameters) to behavior of system[1]. The Unimation PUMA 560 serially links robot manipulator was used as a basis, because this robot manipulator is 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 [1-15].

Controller is a device which can sense information from linear or nonlinear system (e.g., robot manipulator) to improve the systems performance [3]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[5]. Several industrial robot manipulators are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when robot manipulator works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, parameter variations). In some applications robot manipulators are used in an 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. Sliding mode controller is a powerful nonlinear robust controller under condition of partly uncertain dynamic parameters of system [7]. This controller is used to control of highly nonlinear systems especially for robot manipulators. Chattering phenomenon and nonlinear equivalent dynamic formulation in uncertain dynamic parameter are two main drawbacks in pure sliding mode controller [20]. 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 controller is divided into two main sub controllers: discontinues controller($\boldsymbol{\tau}_{dis}$) and equivalent controller($\boldsymbol{\tau}_{eq}$). Discontinues controller causes an acceptable tracking performance at the expense of very fast switching. Conversely in this theory good trajectory following is based on fast switching, fast switching is caused to have system instability and chattering phenomenon. Fine tuning the sliding surface slope is based on nonlinear equivalent part [1, 6]. However, this controller is used in many applications but, pure sliding mode controller has two most important challenges: chattering phenomenon and nonlinear equivalent dynamic formulation in uncertain parameters[20]. Chattering phenomenon (Figure 1) 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].

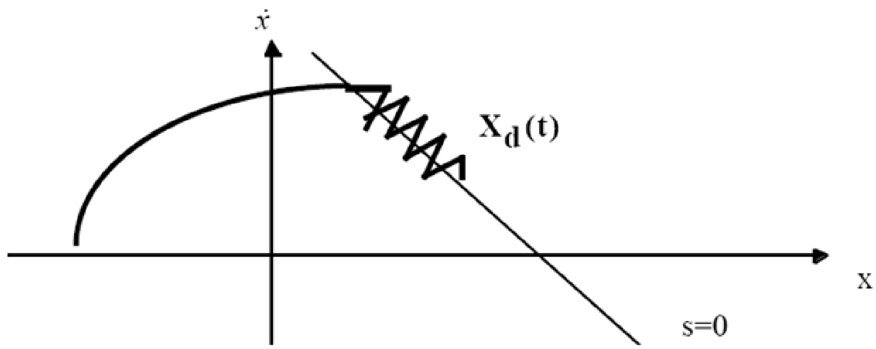


FIGURE 1: Chattering as a result of imperfect control switching [1].

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]. 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, Weng and 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 most exceptional stability and robustness. Sliding mode fuzzy controller has the two most important advantages: reduce the number of fuzzy rule base and increase robustness and stability. Conversely sliding mode fuzzy controller has the above advantages, define the sliding surface slope coefficient very carefully is the main disadvantage of this controller. 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. However this controller's response is very fast and robust but it has chattering phenomenon. Design a robust controller for robot manipulator is essential because robot manipulator has highly nonlinear dynamic parameters.

This paper is organized as follows:
In section 2, dynamic and kinematics formulation of robot manipulator and methodology of implemented of them are presented. Detail of classical SMC and MATLAB/SIMULINK implementation of this controller is presented in section 3. In section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. PUMA 560 ROBOT MANIPULATOR FORMULATION: DYNAMIC FORMULATION OF ROBOTIC MANIPULATOR AND KINEMATICS FORMULATION OF ROBOTIC MANIPULATOR

Rigid-body kinematics: one of the main concern among robotic and control engineers is positioning the manipulator's End-effector to the most accurate place and transparent the effect of disturbance and errors which will affect on manipulator's final result. As a matter of fact, controlling manipulators are hard and expensive because they are multi-input, multi-output, time variant and non-linear, so it has been a topic for researchers to design the most sufficient controller to help the manipulator to achieve to the desired expectation under any circumstance. PUMA 560 is a good instance for manipulators, because it is widely used in both industry and academic, and the dynamic parameters for this robot arm have been identified and documented in literature. One of the main parts of a manipulator's controller is its kinematics which can be divided into two parts; forward kinematics and inverse kinematics. Implementation of inverse kinematic is hard and expensive. In this work we will aim on implementation of PUMA 560 robot manipulator kinematics. Study of robot manipulators is classified into two main groups: kinematics and dynamics. Calculate the relationship between rigid bodies and end-effector without any forces is called Robot manipulator Kinematics. Study of this part is pivotal to calculate accurate dynamic part, to design with an acceptable performance controller, and in real situations and practical applications. As expected the study of manipulator kinematics is divided into two main parts: forward and inverse kinematics. Forward kinematics has been used to find the position and orientation of task (end-effector) frame when angles and/or displacement of joints are known. Inverse kinematics has been used to find possible joints variable (displacements and angles) when all position and orientation of end-effector be active [1].

The main target in forward kinematics is calculating the following function:

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

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.

The Denavit-Hartenberg (D-H) convention is a method of drawing robot manipulators free body diagrams. Denvit-Hartenberg (D-H) convention study is necessary to calculate forward kinematics in serial robot manipulator. The first step to calculate the serial link robot manipulator forward kinematics is link description; the second step is finding the D-H convention after the frame attachment and finally finds the forward kinematics. Forward kinematics is a 4×4 matrix which 3×3 of them shows the rotation matrix, 3×1 of them is shown the position vector and last four cells are scaling factor[1, 6]. 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 =[$q_1, q_2, \ldots\ldots, q_n$], 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 are 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, Zhang and Paul have worked on particular way of robot kinematics solution to reduce the computation[10]. Kieffer has proposed a simple iterative solution to computation of inverse kinematics[11]. Ahmad and Guez 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].

Singularity is a location in the robot manipulator's workspace which the robot manipulator loses one or more degrees of freedom in Cartesian space. Singularities are one of the most important challenges in inverse kinematics which Cheng et al., have proposed a method to solve this problem [13]. A systematic Forward Kinematics of robot manipulator solution is the main target of this part. The first step to compute Forward Kinematics (F.K) of robot manipulator is finding the standard D-H parameters. Figure 2 shows the schematic of the PUMA 560 robot manipulator. The following steps show the systematic derivation of the standard D-H parameters.

1. Locate the robot arm
2. Label joints
3. Determine joint rotation or translation (θ or d)
4. Setup base coordinate frames.
5. Setup joints coordinate frames.
6. Determineα_i, thatα_i, link twist, is the angle between Z_i and Z_{i+1} about an X_i.
7. Determine d_i and a_i , that a_i, link length, is the distance between Z_i and Z_{i+1} along X_i. d_i, offset, is the distance between X_{i-1} and X_i along Z_i axis.
8. Fill up the D-H parameters table. Table 1 shows the standard 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;

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

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

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

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

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

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

$$R_n^0 = (U_1V_1)(U_2V_2) \dots \dots \dots (U_nV_n) \tag{5}$$

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

TABLE 1: The Denavit Hartenberg parameter

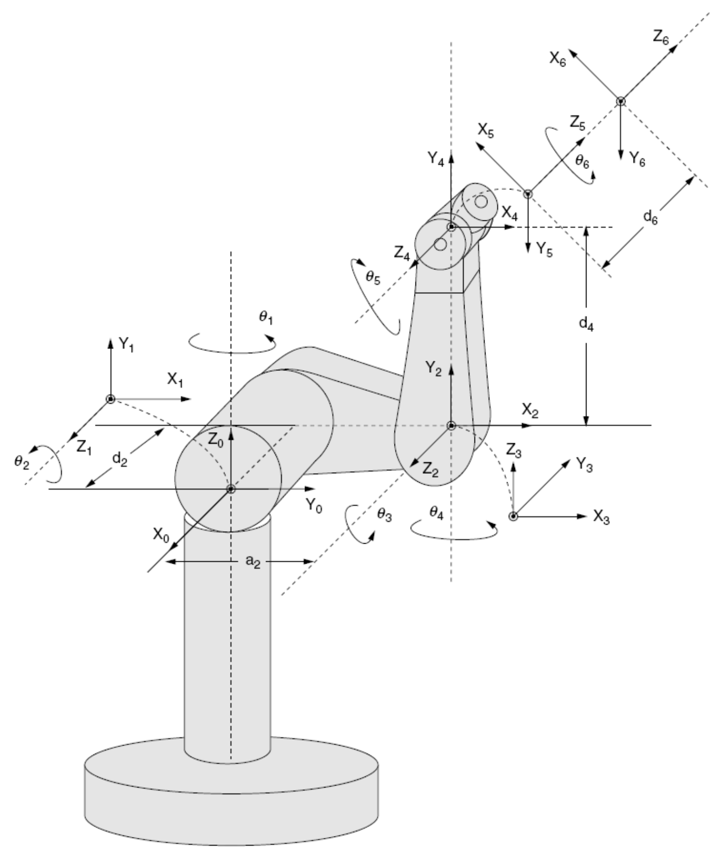


FIGURE 2: D-H notation for a six-degrees-of-freedom PUMA 560 robot manipulator[2]

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]

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

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

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

Kinematics of PUMA 560 robot manipulator: In PUMA robot manipulator the final transformation matrix is given by

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

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

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

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

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

Table 2 shows the PUMA 560 D-H parameters.

Link i	θ_i(rad)	α_i(rad)	a_i(m)	d_i(m)
1	θ_1	$-\pi/2$	0	0
2	θ_2	0	0.4318	0.14909
3	θ_3	$\pi/2$	0.0203	0
4	θ_4	$-\pi/2$	0	0.43307
5	θ_5	$\pi/2$	0	0
6	θ_6	0	0	0.05625

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:

$$\boldsymbol{Nx} = \boldsymbol{cos(\theta\_6)} \times (\boldsymbol{cos(\theta\_5)} \times (\boldsymbol{cos(\theta\_4)} \times \boldsymbol{cos(\theta\_2 + \theta\_3)} \times \boldsymbol{cos(\theta\_1)} + \boldsymbol{sin(\theta\_1)} \times \boldsymbol{sin(\theta\_1)}) + \boldsymbol{sin(\theta\_5)} \times \boldsymbol{sin(\theta\_2 + \theta\_3)} \times \boldsymbol{cos(\theta\_1)}) + \boldsymbol{sin(\theta\_6)} \times (\boldsymbol{sin(\theta\_4)} \times \boldsymbol{cos(\theta\_2 + \theta\_3)} \times \boldsymbol{cos(\theta\_1)} - \boldsymbol{cos(\theta\_4)} \times \boldsymbol{sin(\theta\_1)}) \tag{11}$$

$$Ny = cos(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos(\theta_1)) + sin(\theta_5) \times sin(\theta_2 + \theta_3) \times sin(\theta_1)) + sin(\theta_6) \times (sin(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) + cos(\theta_4) \times cos(\theta_1)) \quad (12)$$

$$Nz = cos(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2 + \theta_3) - sin(\theta_5) \times cos(\theta_2 + \theta_3)) + sin(\theta_6) \times sin(\theta_4) \times sin(\theta_2 + \theta_3) \quad (13)$$

$$Bx = -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times cos(\theta_1) + sin(\theta_4) \times sin(\theta_1)) + sin(\theta_5) \times sin(\theta_2 + \theta_3) \times cos(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2 + \theta_3) \times cos(\theta_1) - cos(\theta_4) \times sin(\theta_1)) \quad (14)$$

$$By = -sin(\theta_6) \times (cos(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos(\theta_1)) + sin(\theta_5) \times sin(\theta_2 + \theta_3) \times sin(\theta_1)) + cos(\theta_6) \times (sin(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) + cos(\theta_4) \times cos(\theta_1)) \quad (15)$$

$$Bz = -sin(\theta_6) \times (cos(\theta_5) \times cos(\theta_4) \times sin(\theta_2 + \theta_3) - sin(\theta_5) \times cos(\theta_2 + \theta_3)) + cos(\theta_6) \times sin(\theta_4) \times sin(\theta_2 + \theta_3) \quad (16)$$

$$Tx = sin(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times cos(\theta_1) + sin(\theta_4) \times sin(\theta_1)) - cos(\theta_5) \times sin(\theta_2 + \theta_3) \times cos(\theta_1) \quad (17)$$

$$Ty = sin(\theta_5) \times (cos(\theta_4) \times cos(\theta_2 + \theta_3) \times sin(\theta_1) - sin(\theta_4) \times cos(\theta_1)) - cos(\theta_5) \times sin(\theta_2 + \theta_3) \times sin(\theta_1) \quad (18)$$

$$Tz = sin(\theta_5) \times cos(\theta_4) \times sin(\theta_2 + \theta_3) + cos(\theta_5) \times cos(\theta_2 + \theta_3) \quad (19)$$

$$Px = 0.4331 \times sin(\theta_2 + \theta_3) \times cos(\theta_1) + 0.0203 \times cos(\theta_2 + \theta_3) \times cos(\theta_1) - 0.1491 \times sin(\theta_1) + 0.4318 \times cos(\theta_2 cos(\theta_1) \quad (20)$$

$$Py = 0.4331 \times sin(\theta_2 + \theta_3) \times sin(\theta_1) + 0.0203 \times cos(\theta_2 + \theta_3) \times sin(\theta_1) + 0.1491 \times cos(\theta_1) + 0.4312 \times cos(\theta_2) \times sin(\theta_1) \quad (21)$$

$$Pz = -0.4331 \times cos(\theta_2 + \theta_3) + 0.0203 \times sin(\theta_2 + \theta_3) + 0.4318 \times sin(\theta_2) \quad (22)$$

PUMA-560 Kinematics Implementation Using MATLAB/SIMULINK

robot manipulator kinematics is essential part to calculate the relationship between rigid bodies and end-effector without any forces. Study of this part is fundamental to calculate accurate dynamic part, to design a controller with acceptable performance, and finally in real situations and particular applications.

In forward kinematics, variables of joints (revolute or prismatic) is given and position and orientation (pose) of rigid body is desired *(*Figure 3*).* In revolute joints the variables are θ_i which means it's joint angle with its neighbor joint. If the joint is prismatic, the variable is d_i which means link offset between joints. In forward kinematics the final result is a 4×4 matrix which 3 factors of it, is end-effector's position and 9 is it's orientation as shown in Figure 10.

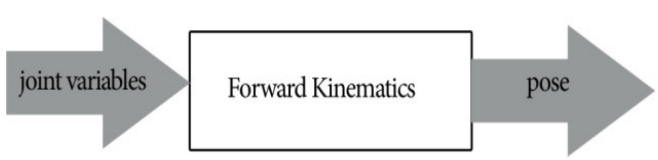


FIGURE3: Forward kinematics block diagram

Desired input is our goal. It means that we are expecting our End-effector to reach at that point.

Sometimes the result that manipulator is reaching at is not what we were expecting for. The main cause of this problem is the disturbances which effects on our system. Nonetheless to say, these disturbances are unwanted, affect on the result and they are the main reason for controller designing. Actual input means the point that end-effector has reached as a result. Actual input, if the disturbance does not affect on our system is the same as desired input and if it affect, is far from the desired input. Clearly saying, if the desired input and actual input become different, the meaning is that the end-effector has not reached to the expected point. At the very first place we must define our system .The system that we are working on is PUMA 560 which has 6 degrees of freedom (6 DOF) and it's joints are RRR. It means that all joints are revolute. As mentioned before, due to type of joints, desired inputs are varied. The joints of the system we are working on are RRR which means they are revolute. So, system's variables are θ_i.as shown in Figure 4, we have 12 inputs and 24 outputs. Inputs are both desired inputs and actual inputs. Our goal is testing if the actual result has reached to our desired. Also its outputs are two 4×4 matrixes. In every matrix, three of factors, show position and nine factors show orientation (Figure 4).

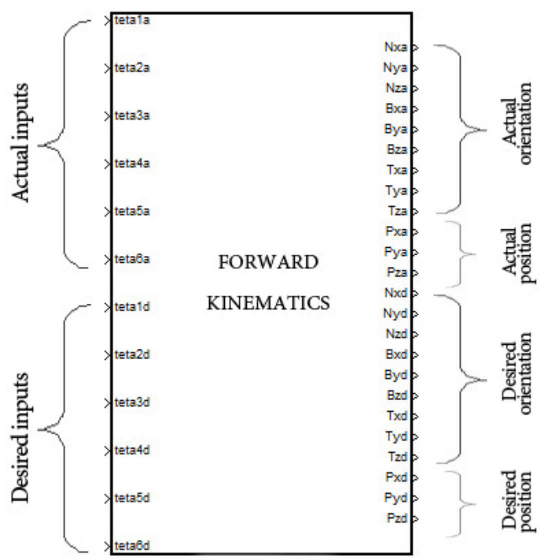


FIGURE 4: Forward kinematics block diagram: inputs and outputs

Note that we aim on controlling the position and we do not work on orientation. So in this system, we do not work on actual orientation and desired orientation. In the next step, we must implement the block diagram of our kinematics. Table 3 shows the input and outputs used in our kinematics block diagram. Also Table 4 and Table 5 show formulation for each variable used in kinematics.

Nxa,Nya,Nza,Bxa,Bya,Bza,Txa,Tya,Tza,Pxa,Pya, Pza	teta1a,teta2a,teta3a,teta4d,teta5d,teta6a,te ta1d
Nxd,Nyd,Nzd,Bxd,Byd,Bzd,Txd,Tyd,Tzd,Pxd,Py d,Pzd	teta2d,teta3d,teta4d,teta5d,teta6d

TABLE3: Inputs and outputs of kinematics

Nxa	cos(teta6a)*(cos(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*cos(teta1a)+sin(teta4d)*sin(teta1a))+ sin(teta5d)*sin(teta2a+teta3a)*cos(teta1a))+sin(teta6a)*(sin(teta4d)*cos(teta2a+teta3a)*cos(teta1a)-cos(teta4d) * sin(teta1a))
Nya	cos(teta6a)*(cos(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*sin(teta1a)- sin(teta4d)*cos(teta1a))+sin(teta5d)*sin(teta2a+teta3a)*sin(teta1a))+sin(teta6a)*(sin(teta4d)*cos(teta2a+teta3a)*sin(teta1a)+cos(teta4d)* cos(teta1a))
Nza	cos(teta6a)*(cos(teta5d)*cos(teta4d)*sin(teta2a+teta3a)- sin(teta5d)* cos(teta2a+teta3a))+sin(teta6a)*sin(teta4d)*sin(teta2a+teta3a)
Bxa	-sin(teta6a)*(cos(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*cos(teta1a) +sin(teta4d)*sin(teta1a))+sin(teta5d)*sin(teta2a+teta3a)*cos(teta1a))+cos(teta6a)*(sin(teta4d)* cos(teta2a+teta3a)*cos(teta1a)-cos(teta4d)*sin(teta1a))
Bya	-sin(teta6a)*(cos(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*sin(teta1a) - sin(teta4d)*cos(teta1a))+sin(teta5d)*sin(teta2a+teta3a)*sin(teta1a)) +cos(teta6a)*(sin(teta4d)*cos(teta2a+teta3a)*sin(teta1a)+cos(teta4d)*cos(teta1a))
Bza	-sin(teta6a)*(cos(teta5d)*cos(teta4d)*sin(teta2a+teta3a)-sin(teta5d)* cos(teta2a+teta3a))+cos(teta6a)*sin(teta4d)*sin(teta2a+teta3a)
Txa	sin(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*cos(teta1a)+sin(teta4d)* sin(teta1a))- cos(teta5d)*sin(teta2a+teta3a)*cos(teta1a)
Tya	sin(teta5d)*(cos(teta4d)*cos(teta2a+teta3a)*sin(teta1a)-sin(teta4d) *cos(teta1a))- cos(teta5d)*sin(teta2a+teta3a)*sin(teta1a)
Tza	sin(teta5d)*cos(teta4d)*sin(teta2a+teta3a)+cos(teta5d)* cos(teta2a+teta3a)
Pxa	0.4331*sin(teta2a+teta3a)*cos(teta1a)+0.0203*cos(teta2a+teta3a)* cos(teta1a)- 0.1491*sin(teta1a)+0.4318*cos(teta2a)*cos(teta1a)
Pya	0.4331*sin(teta2a+teta3a)*sin(teta1a)+0.0203*cos(teta2a+teta3a)* sin(teta1a)+0.1491*cos(teta1a)+0.4312*cos(teta2a)*sin(teta1a)
Pza	-0.4331*cos(teta2a+teta3a)+0.0203*sin(teta2a+teta3a)+0.4318* sin(teta2a)

TABLE 4: Actual input formulas

Nxd	cos(teta6d)*(cos(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*cos(teta1d)+sin(teta4d)*sin(teta1d))+ sin(teta5d)*sin(teta2d+teta3d)*cos(teta1d))+sin(teta6d)*(sin(teta4d)*cos(teta2d+teta3d)*cos(teta1d)-cos(teta4d) * sin(teta1d))
Nyd	cos(teta6d)*(cos(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*sin(teta1d)-sin(teta4d)*cos(teta1d))+sin(teta5d)*sin(teta2d+teta3d)*sin(teta1d))+sin(teta6d)*(sin(teta4d)*cos(teta2d+teta3d)*sin(teta1d)+cos(teta4d)* cos(teta1d))
Nzd	cos(teta6d)*(cos(teta5d)*cos(teta4d)*sin(teta2d+teta3d)- sin(teta5d)* cos(teta2d+teta3d))+sin(teta6d)*sin(teta4d)*sin(teta2d+teta3d)
Bxd	-sin(teta6d)*(cos(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*cos(teta1d) +sin(teta4d)*sin(teta1d))+sin(teta5d)*sin(teta2d+teta3d)*cos(teta1d))+cos(teta6d)*(sin(teta4d)* cos(teta2d+teta3d)*cos(teta1d)-cos(teta4d)*sin(teta1d))
Byd	-sin(teta6d)*(cos(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*sin(teta1d) - sin(teta4d)*cos(teta1d))+sin(teta5d)*sin(teta2d+teta3d)*sin(teta1d)) +cos(teta6d)*(sin(teta4d)*cos(teta2d+teta3d)*sin(teta1d)+cos(teta4d)*cos(teta1d))
Bzd	-sin(teta6d)*(cos(teta5d)*cos(teta4d)*sin(teta2d+teta3d)-sin(teta5d)* cos(teta2d+teta3d))+cos(teta6d)*sin(teta4d)*sin(teta2d+teta3d)
Txd	sin(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*cos(teta1d)+sin(teta4d)* sin(teta1d))-cos(teta5d)*sin(teta2d+teta3d)*cos(teta1d)
Tyd	sin(teta5d)*(cos(teta4d)*cos(teta2d+teta3d)*sin(teta1d)-sin(teta4d) *cos(teta1d))-cos(teta5d)*sin(teta2d+teta3d)*sin(teta1d)
Tzd	sin(teta5d)*cos(teta4d)*sin(teta2d+teta3d)+cos(teta5d)* cos(teta2d+teta3d)
Pxd	0.4331*sin(teta2d+teta3d)*cos(teta1d)+0.0203*cos(teta2d+teta3d)* cos(teta1d)-0.1491*sin(teta1d)+0.4318*cos(teta2d)*cos(teta1d)
Pyd	0.4331*sin(teta2d+teta3d)*sin(teta1d)+0.0203*cos(teta2d+teta3d)* sin(teta1d)+0.1491*cos(teta1d)+0.4312*cos(teta2d)*sin(teta1d)
Pzd	-0.4331*cos(teta2d+teta3d)+0.0203*sin(teta2d+teta3d)+0.4318* sin(teta2d)

TABLE 5: Desired input formulas

As mentioned before, we aim on position controlling. so we must connect desired position and actual position to RMS error block diagram to find out whether the end-effector has reached to expected point or not. Kinematics of our system is shown in Figure 5.

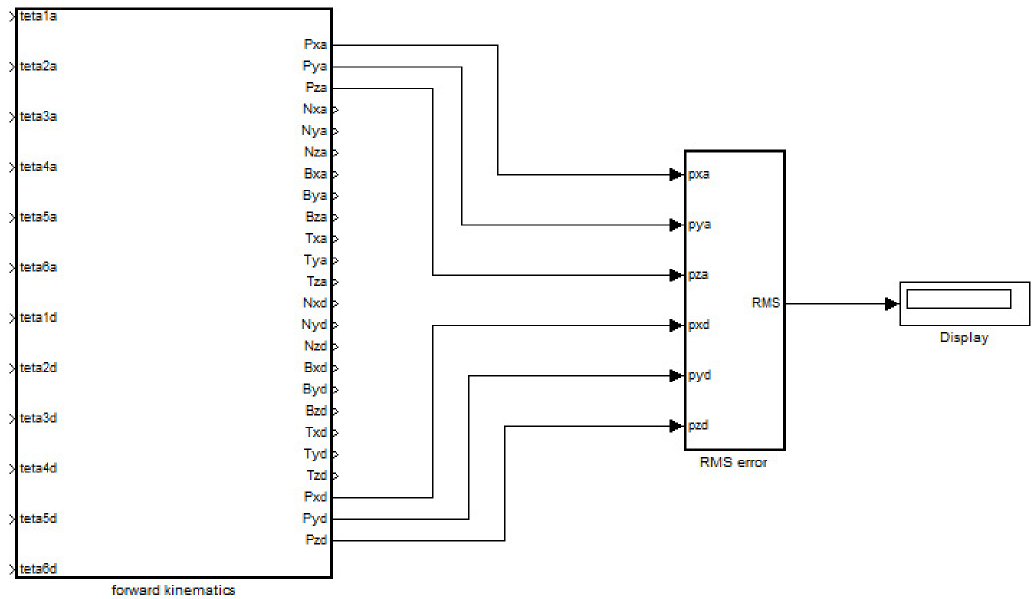


FIGURE 5: Kinematics of PUMA 560

Dynamic of 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.

i. **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.

ii. **Forward dynamics** used to compute the joint acceleration from joint actuators. This algorithm is required for simulations.

iii. **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.

iv. **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, 14].

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, 14]. 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]. 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]:

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

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 2.8 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]:

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

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

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

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

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

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;

$$\boldsymbol{M(\theta)=m_1J_{v1}^TJ_{v1}+J_{\omega1}^{TC1}I_1J_{\omega1}+m_2J_{v2}^TJ_{v2}+J_{\omega2}^{TC2}I_2J_{\omega2}+m_3J_{v3}^TJ_{v3}+J_{\omega3}^{TC3}I_3J_{\omega3}+ m_4J_{v4}^TJ_{v4}+J_{\omega4}^{TC4}I_4J_{\omega4}+m_5J_{v5}^TJ_{v5}+J_{\omega5}^{TC5}I_5J_{\omega5}+m_6J_{v6}^TJ_{v6}+J_{\omega6}^{TC6}I_6J_{\omega6}} \tag{28}$$

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]

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

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

$$\boldsymbol{B(q)} = \begin{bmatrix} b_{112} & b_{113} & \dots & b_{11n} & b_{123} & \dots & b_{12n} & \dots & \dots & b_{1.n-1.n} \\ b_{212} & \dots & \dots & b_{21n} & b_{223} & \dots & \dots & \dots & \dots & b_{2.n-1.n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{n.1.2} & \dots & \dots & b_{n.1.n} & \dots & \dots & \dots & \dots & \dots & b_{n.n-1.n} \end{bmatrix} \quad \textbf{(30)}$$

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

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

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

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

Dynamics of PUMA560 Robot Manipulator

To position control of robot manipulator, the second three axes are locked the dynamic equation of PUMA robot manipulator is given by [77-80];

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

Where

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

$\boldsymbol{M}$ is computed as

$$M_{11} = I_{m1} + I_1 + I_3 \times \cos(\theta_2)\cos(\theta_2) + I_7\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + I_{10}\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{11}\sin(\theta_2)\cos(\theta_2) + I_{21}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + 2 + [I_5\cos(\theta_2)\sin(\theta_2+\theta_3) + I_{12}\cos(\theta_2)\cos(\theta_2+\theta_3) + I_{15}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3) + I_{16}\cos(\theta_2)\sin(\theta_2+\theta_3) + I_{22}\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) \quad \textbf{(35)}$$

$$M_{12} = I_4\sin(\theta_2) + I_8\cos(\theta_2+\theta_3) + I_9\mathit{cos}(\theta_2) + I_{13}\sin(\theta_2+\theta_3) - I_{18}\cos(\theta_2+\theta_3) \quad \textbf{(36)}$$

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

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

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

$$M_{33} = I_{m3} + I_6 + 2I_{15} \quad (40)$$

$$M_{35} = I_{15} + I_{17} \quad (41)$$

$$M_{44} = I_{m4} + I_{14} \quad (42)$$

$$M_{55} = I_{m5} + I_{17} \quad (43)$$

$$M_{66} = I_{m6} + I_{23} \quad (44)$$

$$M_{21} = M_{12}\ , M_{31} = M_{13}\ and\ M_{32} = M_{23} \quad (45)$$

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

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

Where,

$$b_{112} = 2\big[- I_3\sin(\theta_2)\cos(\theta_2) + I_5\cos(\theta_2+\theta_2+\theta_3) + I_7\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{12}\sin(\theta_2+\theta_2+\theta_3) - I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{16}\cos(\theta_2+\theta_2+\theta_3) + I_{21}\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{22}\big(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\big)\big] + I_{10}\big(1-2\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)\big) + I_{11}\big(1-2\sin(\theta_2)\sin(\theta_2)\big) \quad (47)$$

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

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

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

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

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

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

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

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

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

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

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

consequently coriolis matrix is shown as bellows;

$$\boldsymbol{B(q).\dot q\dot q = \begin{bmatrix} b_{112}.\dot q_1\dot q_2 + b_{113}.\dot q_1\dot q_3 + 0 + b_{123}.\dot q_2\dot q_3 \\ 0 + b_{223}.\dot q_2\dot q_3 + 0 + 0 \\ 0 \\ b_{412}.\dot q_1\dot q_2 + b_{413}.\dot q_1\dot q_3 + 0 \\ 0 \\ 0 \end{bmatrix}} \tag{59}$$

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

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

Where,

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

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

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

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

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

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

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

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

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

$$C(q).\dot{q}^2 = \begin{bmatrix} c_{112}.\dot{q}_2^2 + c_{13}.\dot{q}_3^2 \\ c_{21}.\dot{q}_1^2 + c_{23}.\dot{q}_3^2 \\ c_{13}.\dot{q}_1^2 + c_{32}.\dot{q}_2^2 \\ 0 \\ c_{51}.\dot{q}_1^2 + c_{52}.\dot{q}_2^2 \\ 0 \end{bmatrix} \tag{69}$$

Gravity (G) Matrix can be written as

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

Where,

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

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

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

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

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

and K is introduced as

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

$\ddot{q}$ can be written as

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

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

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

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

$$K_3 = \tau_3 - \left[C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2\right] - g_3 \tag{79}$$

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

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

$$K_6 = \tau_6 \tag{82}$$

An information about inertial constant and gravitational constant are shown in Tables 6 and 7 based on the studies carried out by Armstrong [80] and Corke and Armstrong [81].

$\mathrm{I}_1 = 1.43 \pm 0.05$	$I_2 = 1.75 \pm 0.07$
$I_3 = 1.38 \pm 0.05$	$I_4 = 0.69 \pm 0.02$
$I_5 = 0.372 \pm 0.031$	$\mathrm{I}_6 = 0.333 \pm 0.016$
$I_7 = 0.298 \pm 0.029$	$\mathrm{I}_8 = -0.134 \pm 0.014$
$I_9 = 0.0238 \pm 0.012$	$I_{10} = -0.0213 \pm 0.0022$
$I_{11} = -0.0142 \pm 0.0070$	$I_{12} = -0.011 \pm 0.0011$
$I_{13} = -0.00379 \pm 0.0009$	$I_{14} = 0.00164 \pm 0.000070$
$I_{15} = 0.00125 \pm 0.0003$	$I_{16} = 0.00124 \pm 0.0003$
$I_{17} = 0.000642 \pm 0.0003$	$\mathrm{I}_{18} = 0.000431 \pm 0.00013$
$I_{19} = 0.0003 \pm 0.0014$	$I_{20} = -0.000202 \pm 0.0008$
$\mathrm{I}_{21} = -0.0001 \pm 0.0006$	$I_{22} = -0.000058 \pm 0.000015$
$\mathrm{I}_{23} = 0.00004 \pm 0.00002$	$I_{m1} = 1.14 \pm 0.27$
$I_{m2} = 4.71 \pm 0.54$	$I_{m3} = 0.827 \pm 0.093$
$I_{m4} = 0.2 \pm 0.016$	$I_{m5} = 0.179 \pm 0.014$
$I_{m6} = 0.193 \pm 0.016$	

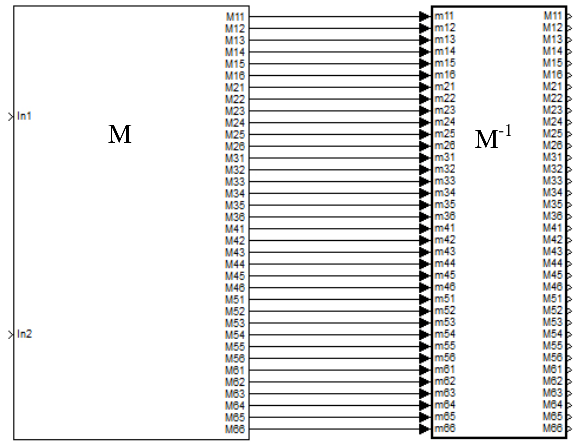
TABLE 6: Inertial constant reference (*$Kg.m^2$*)

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

TABLE 7: Gravitational constant (*N.m*)

Formulation and implementation of Matrix Entries: As mentioned before, every matrix entry has its own formula. Below you can find them:

Finding inverse matrix for kinetic energy: Kinetic energy has illustrated by **M**. The kinetic energy matrix is a **6 x 6** matrix [10]. In MATLAB, the command ***"inv(matrix)"*** will inverse a n x n matrix .what is more, the results must be taken into a separate matrix in order to be used in Dynamic equation. Both **M** and $\mathbf{M}^{-1}$ must be implemented in a separate *Matlab Embedded Function*. The outputs of **M** will be linked to inputs of $\mathbf{M}^{-1}$. The block diagram will be shown as Figure 6.



Coriolis Effect Matrix: The Coriolis Effect is a **15 x 6** matrix. The block diagram of $B(q).[\dot{q}\dot{q}]$ could be shown as Figure 7. We set $q_4 = q_5 = q_6 = 0$

:

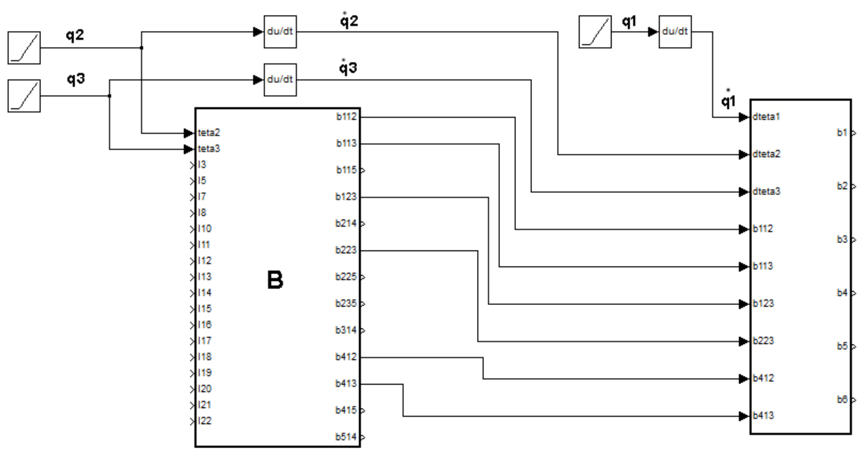


FIGURE7: Block diagram for coriolis effect

Centrifugal Force Matrix

Centrifugal force has illustrated as **C** and is a **6 x 6** matrix. In PUMA 560, the centrifugal force is a **6 x 6** matrix.after implementing centrifugal force in a block diagram, its time to implement $C(q)\dot{q}^2$. We set $q_4 = q_5 = q_6 = 0$ the block diagram for this part could be illustrated as Figure 8.

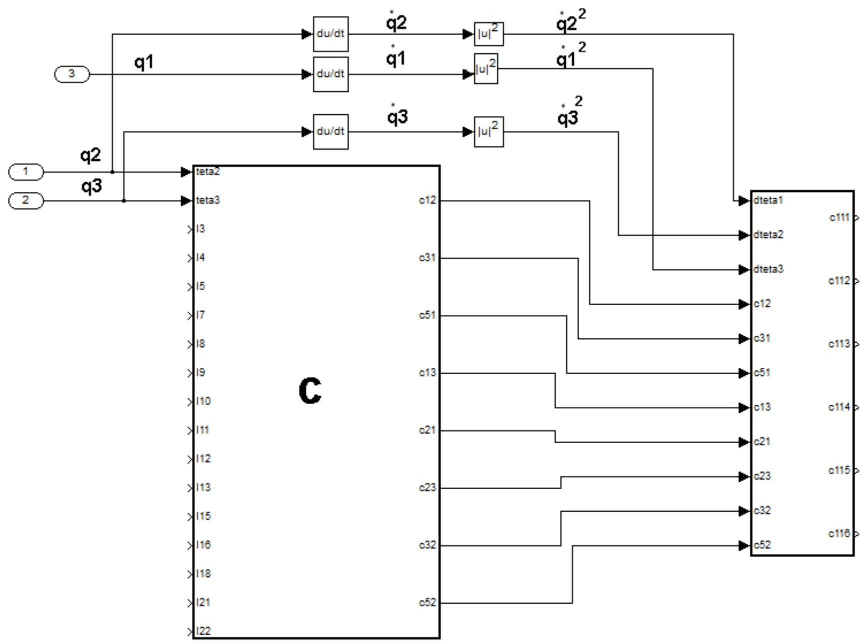


FIGURE8: Block diagram for Centrifugal force

Gravity Matrix
Gravity is shown as **g** and is a **6 x 1** matrix. In PUMA 560, the Gravity is a **6 x 1** matrix. The block diagram is presented as Figure 9.

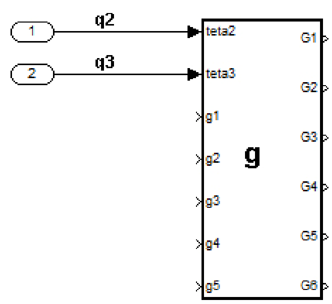


FIGURE9: Block diagram for Gravity

Implement Dynamic Formula in SIMULINK

$\boldsymbol{I}$ is summation between Coriolis Effect Matrix, Centrifugal force Matrix and Gravity matrix. $\boldsymbol{K}$ could be find in equation (77). Figure 10 is shown I and K implementation.

The block diagram I can be made as below:

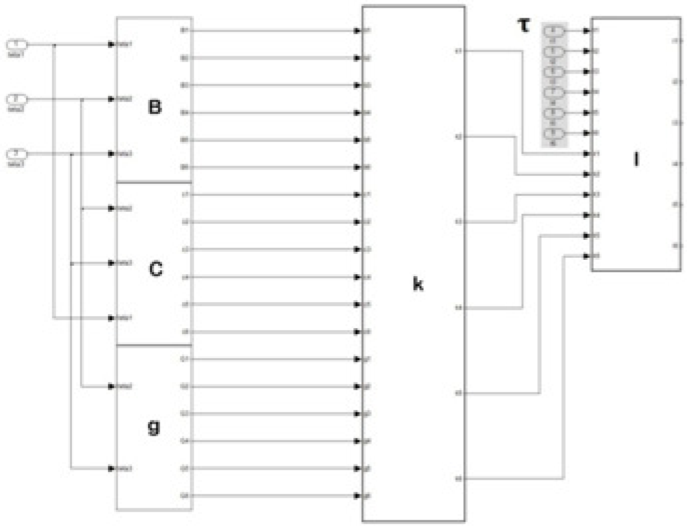


FIGURE 10: Block diagram for K and I

After masking **I** the block diagram will be making as Figure 11.

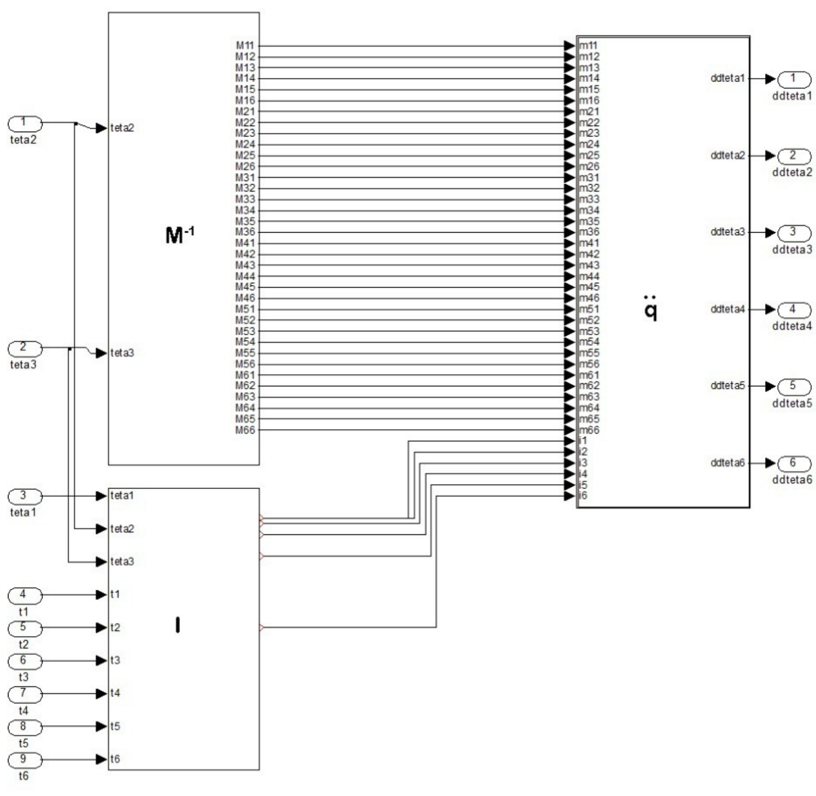


FIGURE 11: Block diagram for $\ddot{q}$

After implementing, block diagram should be masked. The block diagram shown in Figure 11, counts $\ddot{q}$.to count q, block diagram below must be implemented as Figure 12.

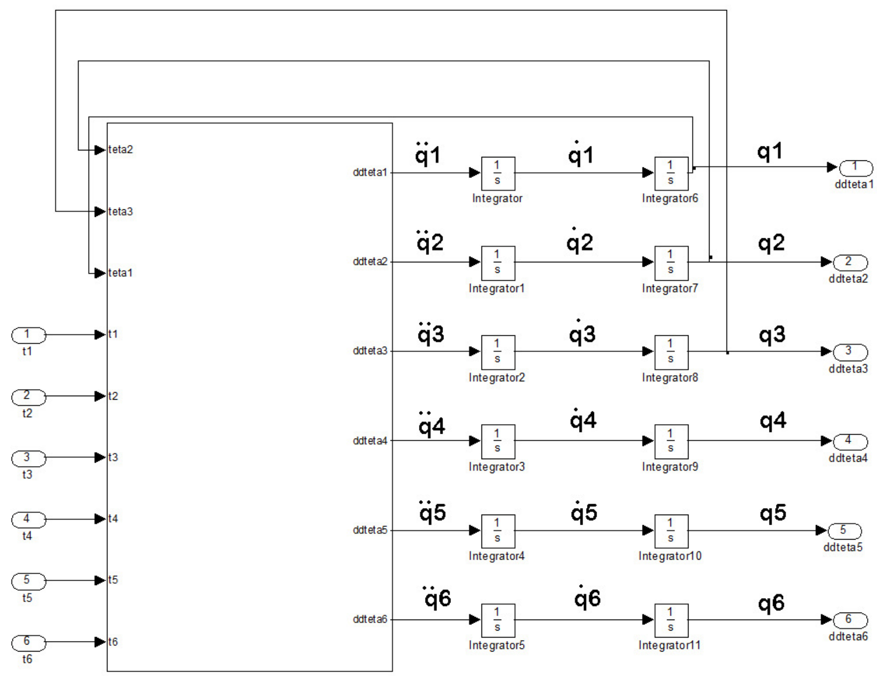


FIGURE 12: Block diagram for q

Now, everything should be masked and constants shown in Table1 and Table2 must be defined. At the end the final block diagram could be illustrated as Figure 13.

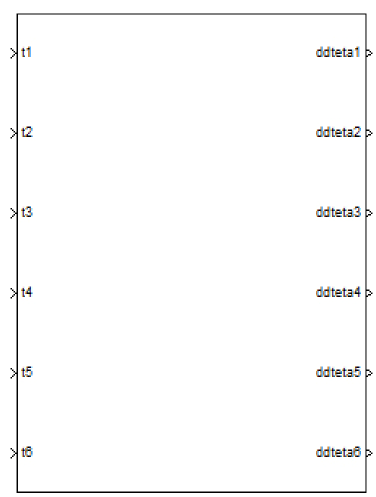


FIGURE 13: final block diagram for Dynamic model

3. CONTROL: SLIDING MODE CONTROLLER ANALYSIS, MODELLING AND IMPLEMENTATION ON PUMA 560 ROBOT MANIPULATOR

In this section formulations of sliding mode controller for robot manipulator is presented based on [1, 6]. Consider a nonlinear single input dynamic system is defined by [6]:

$$x^{(n)} = f(\vec{x}) + b(\vec{x})u \quad (83)$$

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 main goal to design this controller is train to the desired state; $x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d{}^{(n-1)}]^T$, and trucking error vector is defined by [6]:

$$\tilde{x} = x - x_d = [\tilde{x}, \dots, \tilde{x}^{(n-1)}]^T \quad (84)$$

A time-varying sliding surface $s(x,t)$ in the state space R^n is given by [6]:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\tilde{x} = 0 \quad (85)$$

where λ is the positive constant. To further penalize tracking error, integral part can be used in sliding surface part as follows [6]:

$$s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\left(\int_0^t \tilde{x}\,dt\right) = 0 \quad (86)$$

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 $s(x,t)$ [6].

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)| \quad (87)$$

where ζ is positive constant.

$$\text{If } \mathbf{S(0)>0} \rightarrow \frac{d}{dt}S(t) \le -\zeta \quad (88)$$

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

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt}S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \quad (89)$$

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

$$0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta} \quad (90)$$

and

$$if\ S(0) < 0 \rightarrow 0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta} \quad (91)$$

Equation (91) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0 \quad (92)$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt} + \lambda)\quad \tilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \quad (93)$$

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

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \quad (94)$$

suppose the second order system is defined as;

$$\ddot{x} = f + u \rightarrow \dot{S} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \tag{95}$$

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

$$\widehat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \tag{96}$$

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

$$U_{dis} = \widehat{U} - K(\vec{x}, t) \cdot \mathrm{sgn}(s) \tag{97}$$

where the switching function $\mathbf{sgn}(S)$ is defined as [1, 6]

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \tag{98}$$

and the $K(\vec{x}, t)$ is the positive constant. Suppose by (90) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt} s^2(x,t) = \dot{S} \cdot S = \left[f - \hat{f} - K\mathrm{sgn}(s)\right] \cdot S = \left(f - \hat{f}\right) \cdot S - K|S| \tag{99}$$

and if the equation (94) instead of (93) the sliding surface can be calculated as

$$s(x,t) = (\frac{d}{dt} + \lambda)^2 \left(\int_0^t \tilde{x} \, dt\right) = (\dot{x} - \dot{x}_d) + 2\lambda(\dot{x} - \dot{x}_d) - \lambda^2(x - x_d) \tag{100}$$

in this method the approximation of U is computed as [6]

$$\widehat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{x} - \dot{x}_d) + \lambda^2(x - x_d) \tag{101}$$

Based on above discussion, the sliding mode control law for a multi degrees of freedom robot manipulator is written as [1, 6]:

$$\tau = \tau_{eq} + \tau_{dis} \tag{102}$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems and τ_{eq} for first 3 DOF PUMA robot manipulator can be calculate as follows [1]:

$$\tau_{eq} = \left[M^{-1}(B + C + G) + \dot{S}\right]M \tag{103}$$

and τ_{dis} is computed as [1];

$$\tau_{dis} = K \cdot \mathrm{sgn}(S) \tag{104}$$

by replace the formulation (104) in (102) the control output can be written as;

$$\tau = \tau_{eq} + K.\mathrm{sgn}(S) \tag{105}$$

Figure 14 shows the position classical sliding mode control for PUMA 560 robot manipulator. By (105) and (103) the sliding mode control of PUMA 560 robot manipulator is calculated as;

$$\tau = \left[M^{-1}(B + C + G) + \dot{S}\right]M + K \cdot \mathrm{sgn}(S) \tag{106}$$

where $S = \lambda e + \dot{e}$ in PD-SMC and $S = \lambda e + \dot{e} + (\frac{\lambda}{2})^2 \sum e$ in PID-SMC.

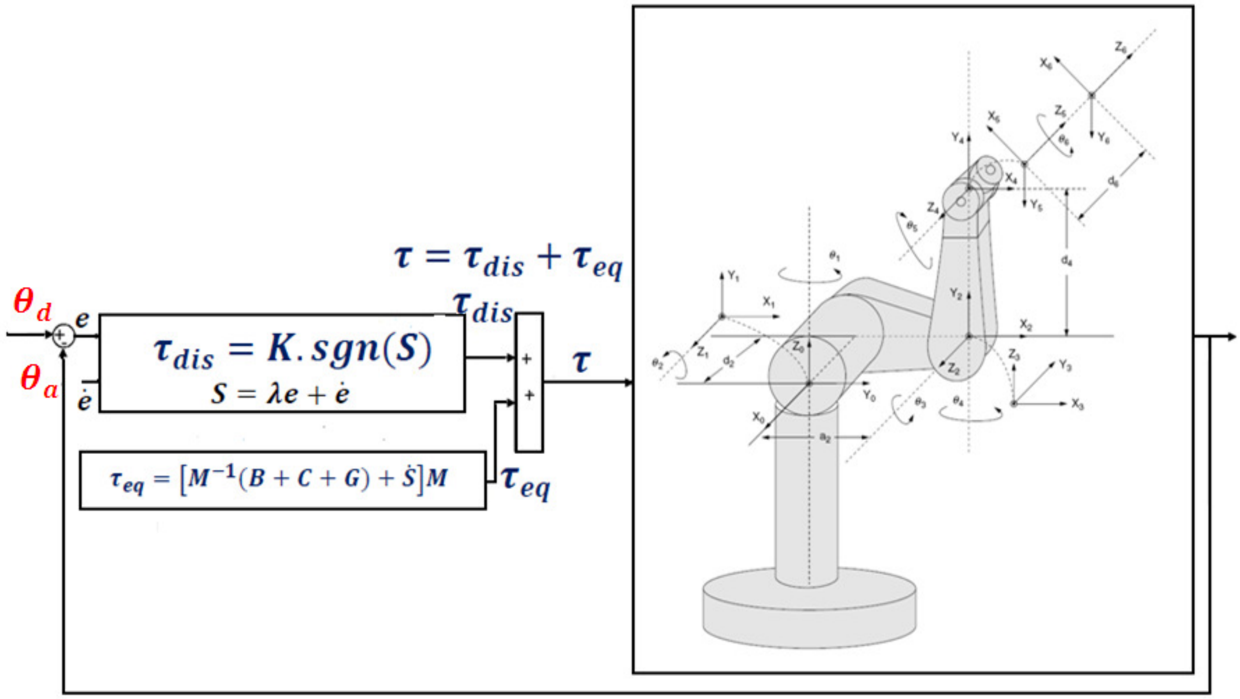


FIGURE 14: Block diagram of pure sliding mode controller with switching function

Implemented Sliding Mode Controller

The main object is implementation of controller block. According to T $_{dis}$ equation which is T $_{dis}$=K * sign(s), this part will be created like figure 15. As it is obvious, the parameter e is the difference of actual and desired values and $\dot{e}$ is the change of error. Luanda (l1) and k are coefficients which are affected on discontinuous component and the saturation function accomplish the switching progress. A sample of discontinuous torque for one joint is like figure 15.

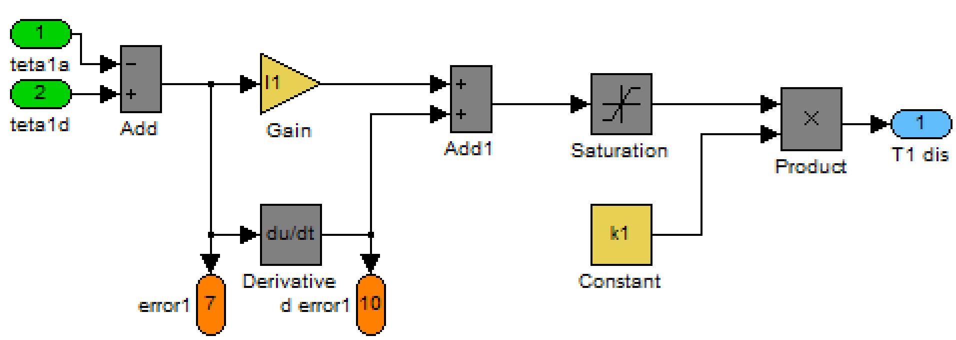


FIGURE 15: Discontinuous part of torque for one joint variable

As it is seen in figure 15 the error value and the change of error were chosen to exhibit in measurement center. In this block by changing gain and coefficient values, the best control system will be applied. In the second step according to torque formulation in SMC mode, the equivalent part should constructed. Based on equivalent formulation $\tau_{eq} = [M^{-1}(B + C + G) + \dot{S}]M$ all constructed blocks just connect to each other as Figure 16. In this figure the N $(q, \dot{q})$ is the dynamic parameters block (i.e., A set of Coriolis, Centrifugal and

Gravity blocks) and the derivative of S is apparent. Just by multiplication and summation, the output which is equivalent torque will be obtained.

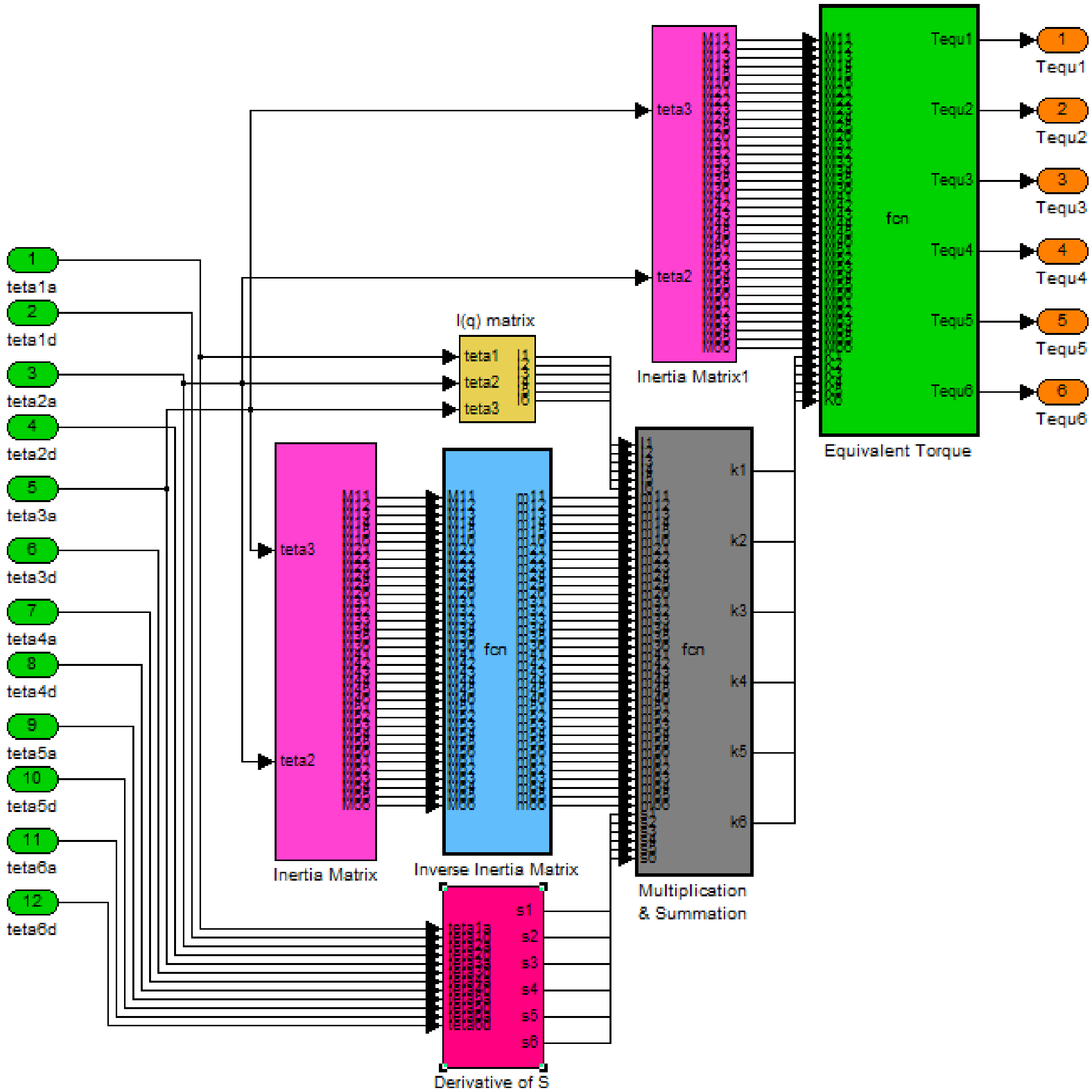


FIGURE 16: the equivalent part of torque with required blocks

The inputs are thetas and the final outputs are equivalent torque values. The relations between other blocks are just multiplication and summation as mentioned in torque equation. The next phase is calculation of the summation of equivalent part and discontinuous part which make the total torque value. This procedure is depicted in Figure 17.

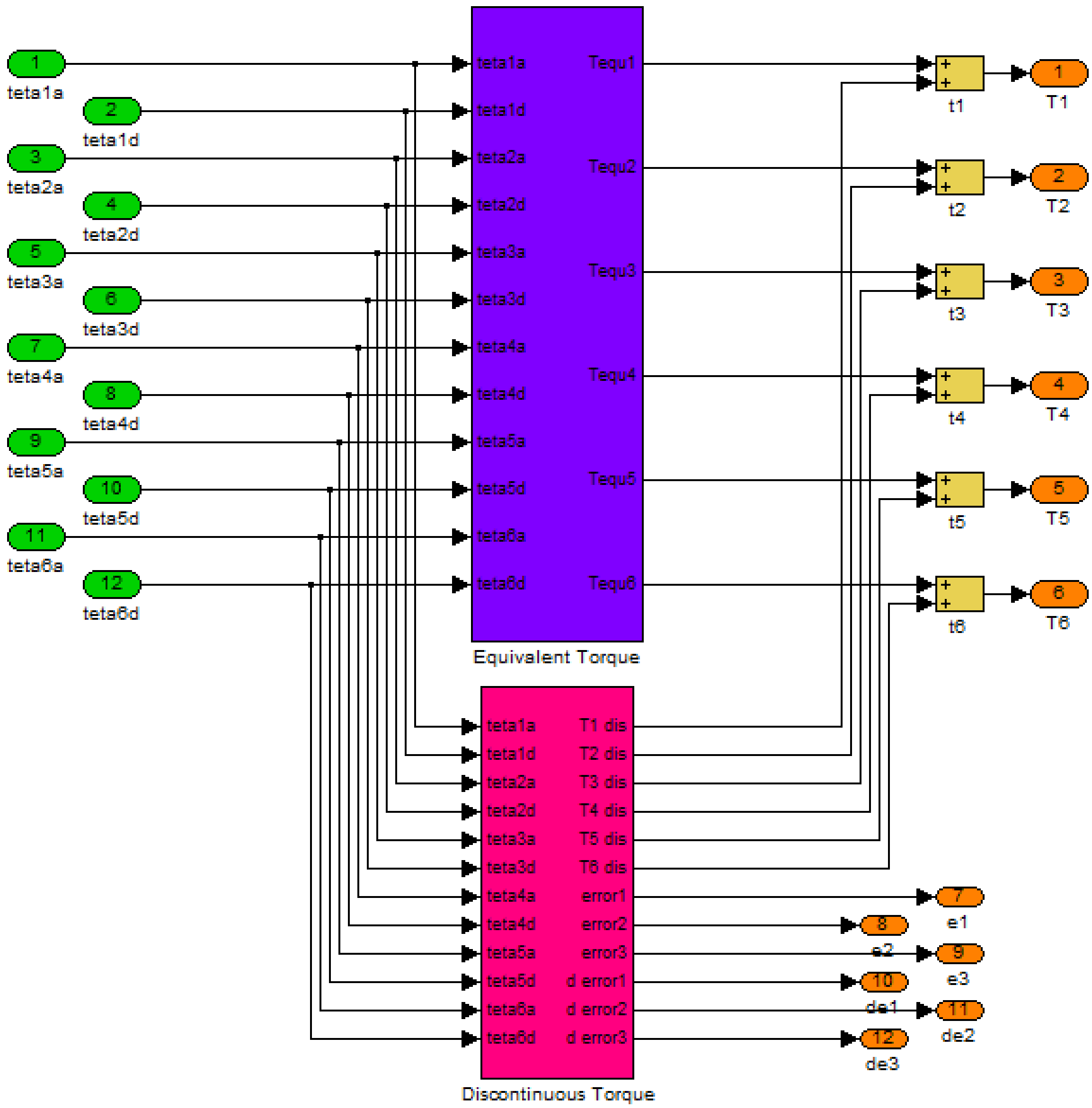


FIGURE 17: the total value of torque which is summation of equivalent & discontinuous blocks

In the next step transform our subsystems into a general system to form controller block and the outputs will be connected to the plant, in order to execute controlling process. Then, trigger the main inputs with power supply to check validity and performance. In Figure 18 Dynamics, Kinematics, Controller and the measurement center blocks are shown.

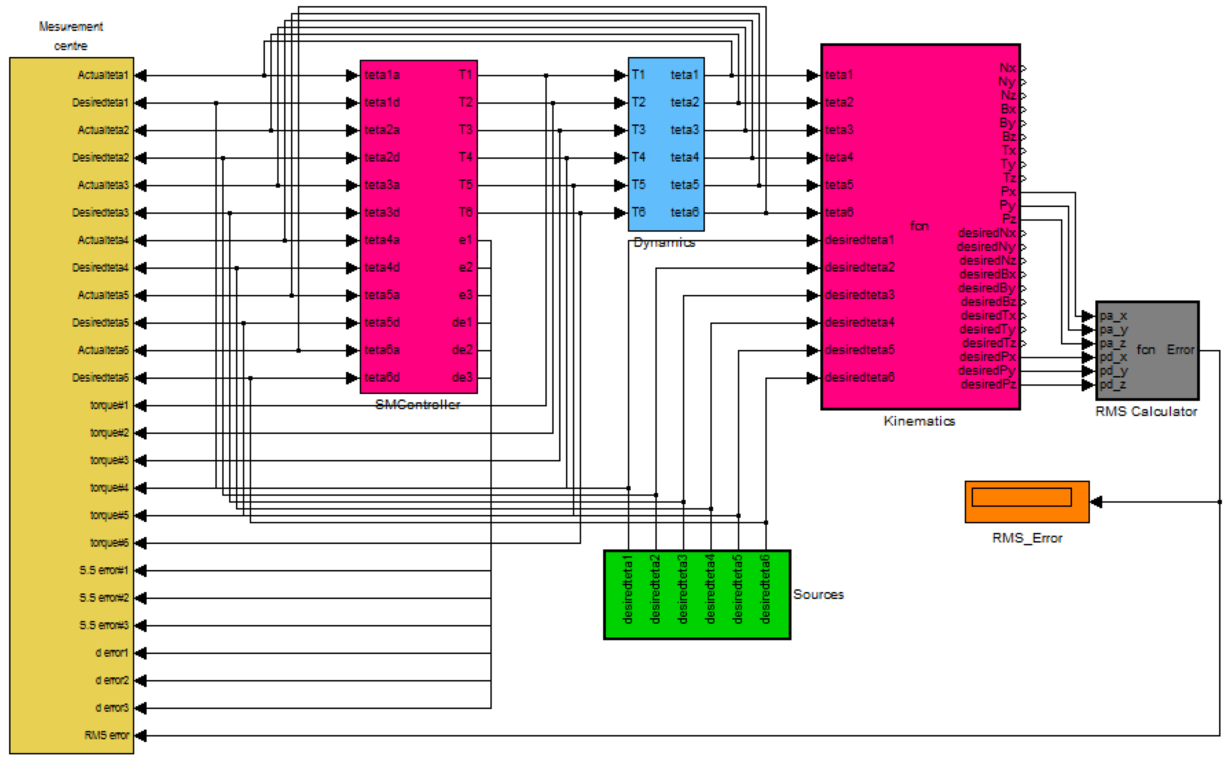


FIGURE 18: Measurement center, Controller, Dynamics and Kinematics Blocks

4. RESULTS

PD-sliding mode controller (PD-SMC) and PID-sliding mode controller (PID-SMC) were tested to Step and Ramp responses. 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 environment. 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

Figures 19 and 20 show the tracking performance in PD-SMC and PID SMC without disturbance for Step and Ramp trajectories. The best possible coefficients in Step PID-SMC are; $K_p = K_v = K_i = 30,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 3, \lambda_2 = 6, \lambda_3 = 6$ and in Ramp PID-SMC are; $K_p = K_v = K_i = 5,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 15, \lambda_2 = 15, \lambda_3 = 10$ as well as similarly in Step PD-SMC are; $K_p = K_v = 10,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 1, \lambda_2 = 6, \lambda_3 = 8$; and at last in Ramp PD-SMC are; $K_p = K_v = 5,\ \emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1, and\ \lambda_1 = 15, \lambda_2 = 15, \lambda_3 = 10$. From the simulation for first, second, and third links, different controller gains have the different result. Tuning parameters of PID-SMC and PD-SMC for two type trajectories in PUMA 560 robot manipulator are shown in Table 8 to 11.

	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
data 1	3	30	0.1	6	30	0.1	6	30	0.1	0	0	-5.3e-15	0
data 2	30	30	0.1	60	30	0.1	60	30	0.1	-5.17	14.27	-1.142	0.05
data 3	3	300	0.1	6	300	0.1	6	300	0.1	2.28	0.97	0.076	0.08

TABLE 8: Tuning parameters of Step PID-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	15	5	0.1	15	5	0.1	10	5	0.1	4.6e-12	-3.97e-12	-3.87e-12	0.0002441
data2	150	5	0.1	150	5	0.1	100	5	0.1	1005	1108	436.5	0.8
data3	15	50	0.1	15	50	0.1	10	50	0.1	-0.1877	-0.1	-0.03	0.0006579

TABLE 9: Tuning parameters of a Ramp PID-SMC

	k_1	λ_1	ϕ_1	k_2	λ_2	ϕ_2	k_3	λ_3	ϕ_3	SS error $_1$	SS error $_2$	SS error $_3$	RMS error
data1	10	1	0.1	10	6	0.1	10	8	0.1	1e-6	1e-6	1e-6	1.2e-6
data2	100	1	0.1	100	6	0.1	100	8	0.1	0.2	0.05	-0.02	-0.037
data3	10	10	0.1	10	60	0.1	10	80	0.1	0.22	-0.21	-0.19	0.09

TABLE 10: Tuning parameters of a Step PD-SMC

	k_1	λ_1	ϕ_1	k_2	λ_2	ϕ_2	k_3	λ_3	ϕ_3	SS error $_1$	SS error $_2$	SS error $_3$	RMS error
data1	5	15	0.1	5	15	0.1	5	10	0.1	-6e-12	-8.5e-11	-1.7e-11	8.3e-5
data2	50	15	0.1	50	15	0.1	50	10	0.1	0.09	0.06	0.02	0.00162
data3	5	150	0.1	5	150	0.1	5	100	0.1	377.7	377	272	0.732

TABLE 11: Tuning parameters of a Ramp PD-SMC

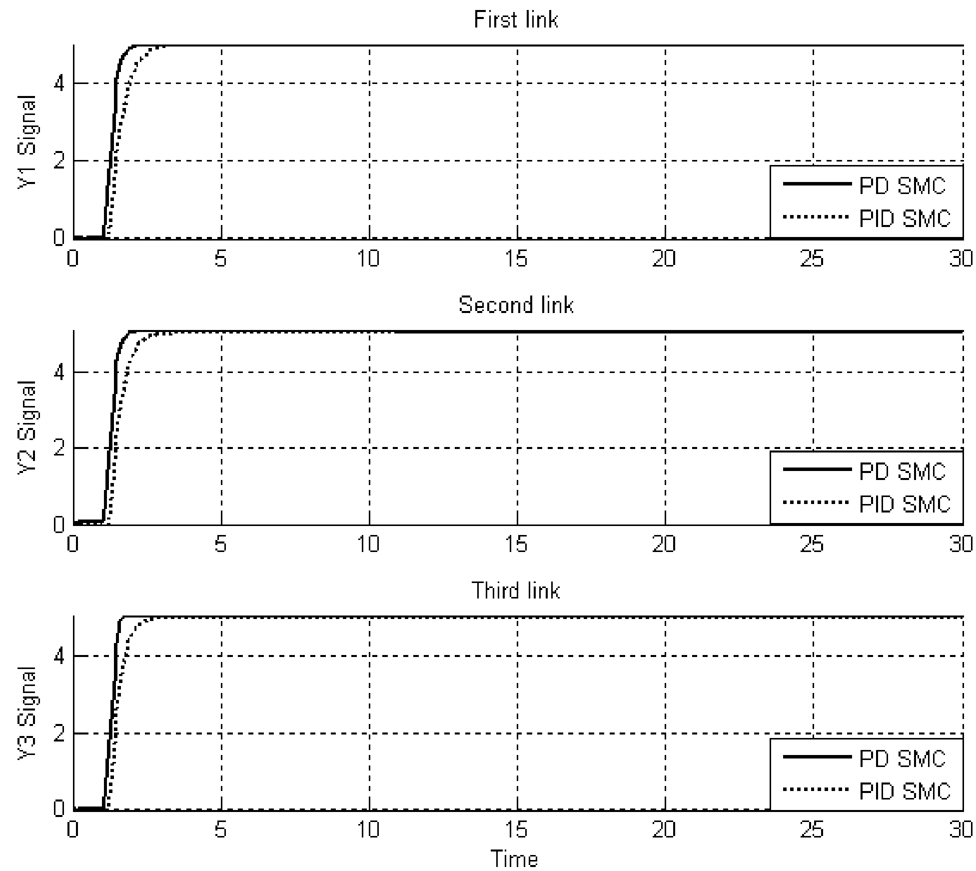


FIGURE 19: Step PD-SMC and PID-SMC for First, second and third link trajectory without any disturbance

By comparing step response, Figure 19, in PD and PID-SMC, conversely the PID's overshoot **(0%)** is lower than PD's **(1%)**, the PD's rise time **(0.483 Sec)** is dramatically lower than PID's **(0.9 Sec)**; in addition the Settling time in PD **(Settling time=0.65 Sec)** is fairly lower than PID **(Settling time=1.4 Sec)**.

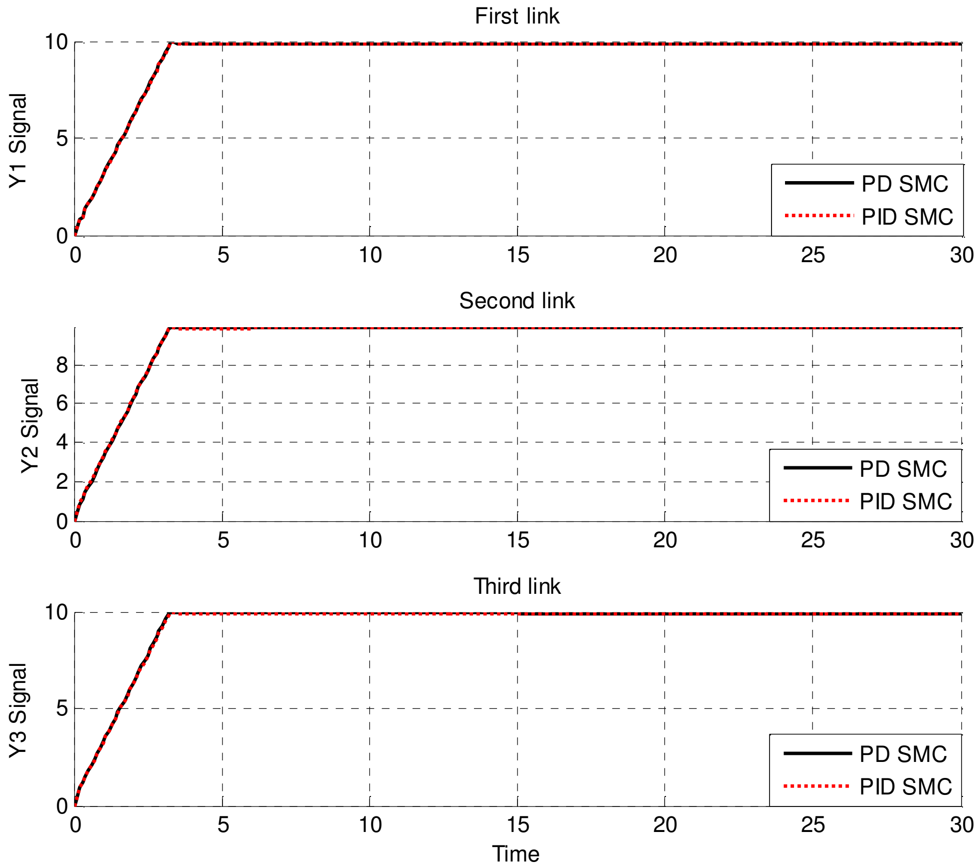


FIGURE 20: Ramp PD SMC and PID SMC for First, second and third link trajectory without any disturbance.

Figure 20 shows that, the trajectories response process that in the first 3.3 seconds rise to 10 then they are on a stable state up to the second 30.

Disturbance Rejection

Figures 21 and 22 are indicated the power disturbance removal in PD and PID-SMC. 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 and ramp PD and PID-SMC; it found slight oscillations in trajectory responses.

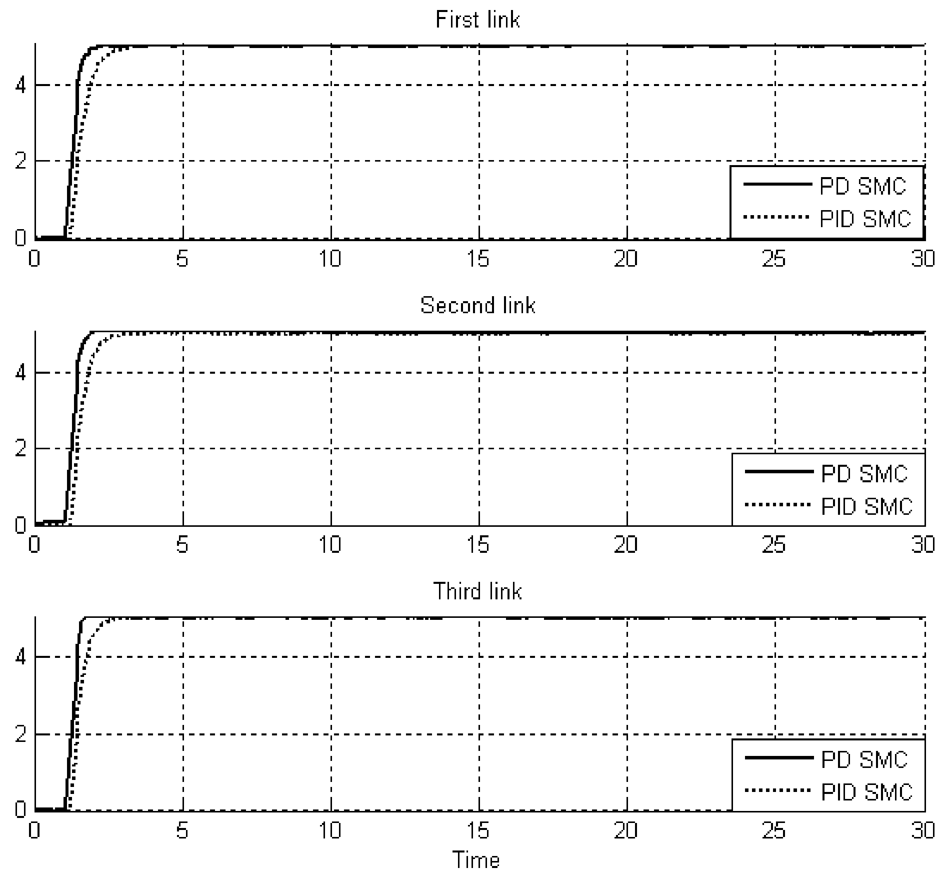


FIGURE 21: Step PD SMC and PID SMC for First, second and third link trajectory with external disturbance

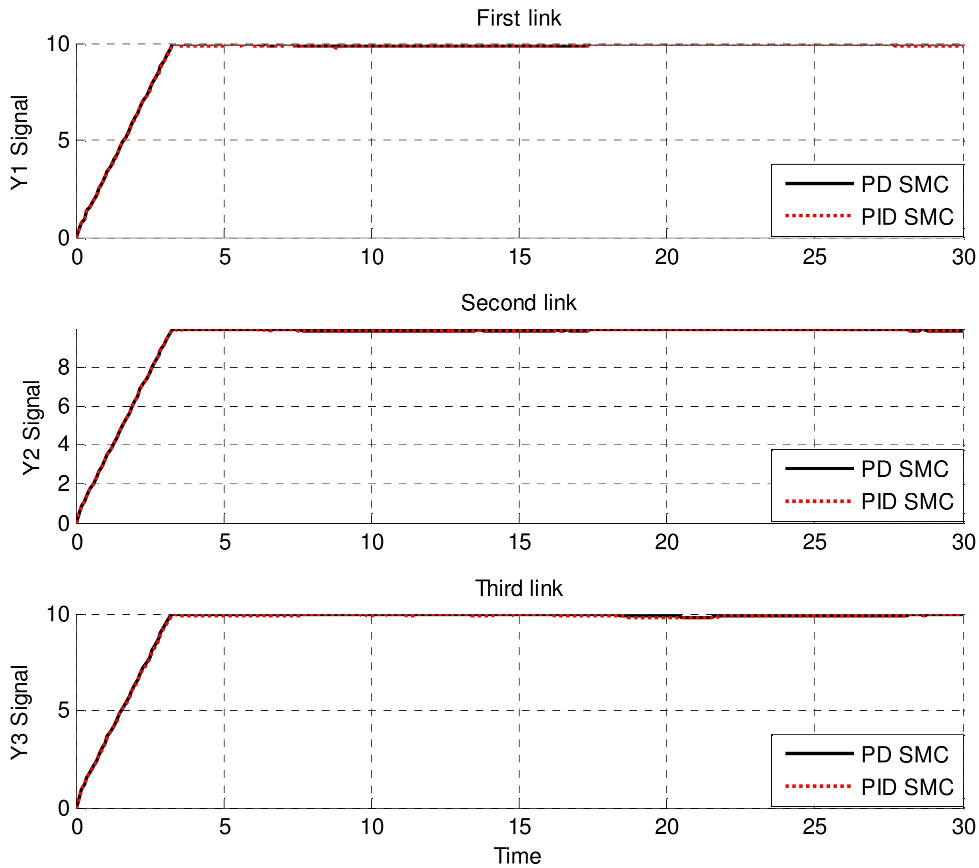


FIGURE 22: Ramp PD SMC and PID SMC for first, second and third link trajectory with external disturbance

Among above graphs (21 and 22), relating to step and ramp trajectories following with external disturbance, PID SMC and PD SMC have slightly fluctuations. By comparing overshoot, rise time, and settling time; PID's overshoot **(0.9%)** is lower than PD's **(1.1%)**, PD's rise time **(0.48 sec)** is considerably lower than PID's **(0.9 sec)** and finally the Settling time in PD **(Settling time=0.65 Sec)** is quite lower than PID **(Settling time=1.5 Sec)**.

Chattering phenomenon: As mentioned in previous chapter, chattering is one of the most important challenges in sliding mode controller which one of the major objectives in this research is reduce or remove the chattering in system's output. To reduce the chattering researcher is used $saturation$ function instead of $switching$ function. Figure 23 has shown the power of boundary layer (saturation) method to reduce the chattering in PD-SMC.

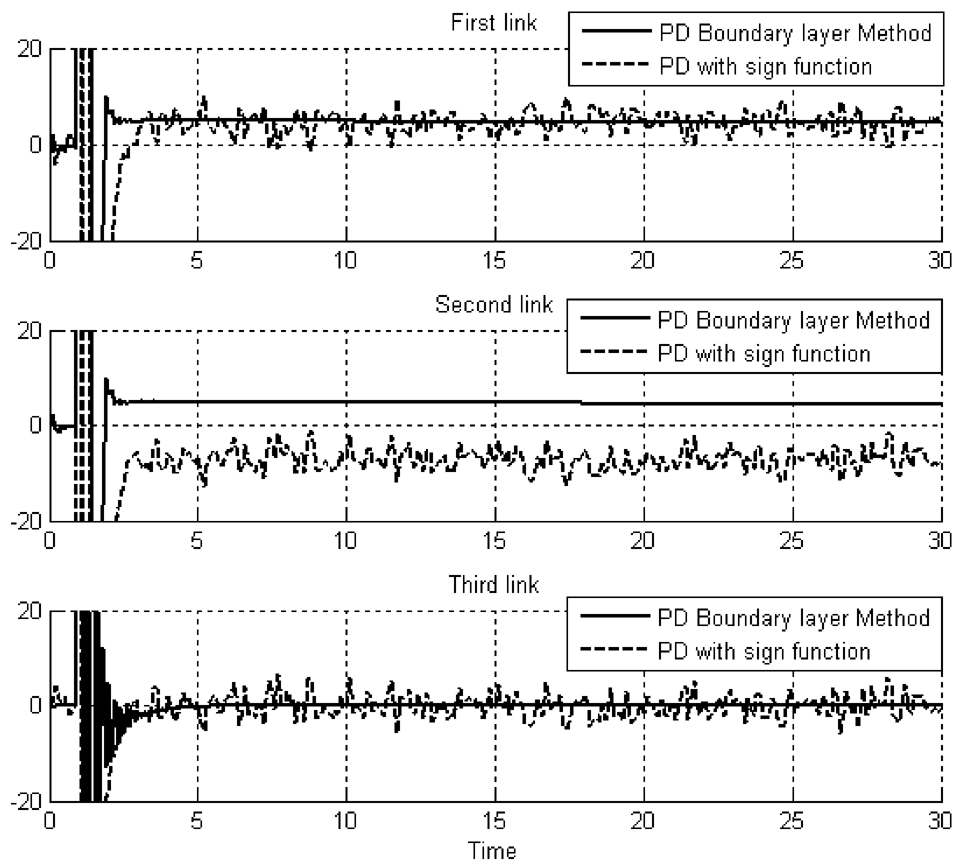


FIGURE 23: PD-SMC boundary layer methods Vs. PD-SMC with discontinuous (Sign) function

Figures 24 and 25 have indicated the power of chattering rejection in PD and PID-SMC, with and without disturbance. As mentioned before, chattering can caused to the hitting in driver and mechanical parts so reduce the chattering is more important. Furthermore band limited white noise with predefined of 40% the power of input signal is applied the step and ramp PD and PID-SMC, it seen that slight oscillations in third joint trajectory responses. Overall in this research with regard to the step response, PD-SMC has the steady chattering compared to the PID-SMC.

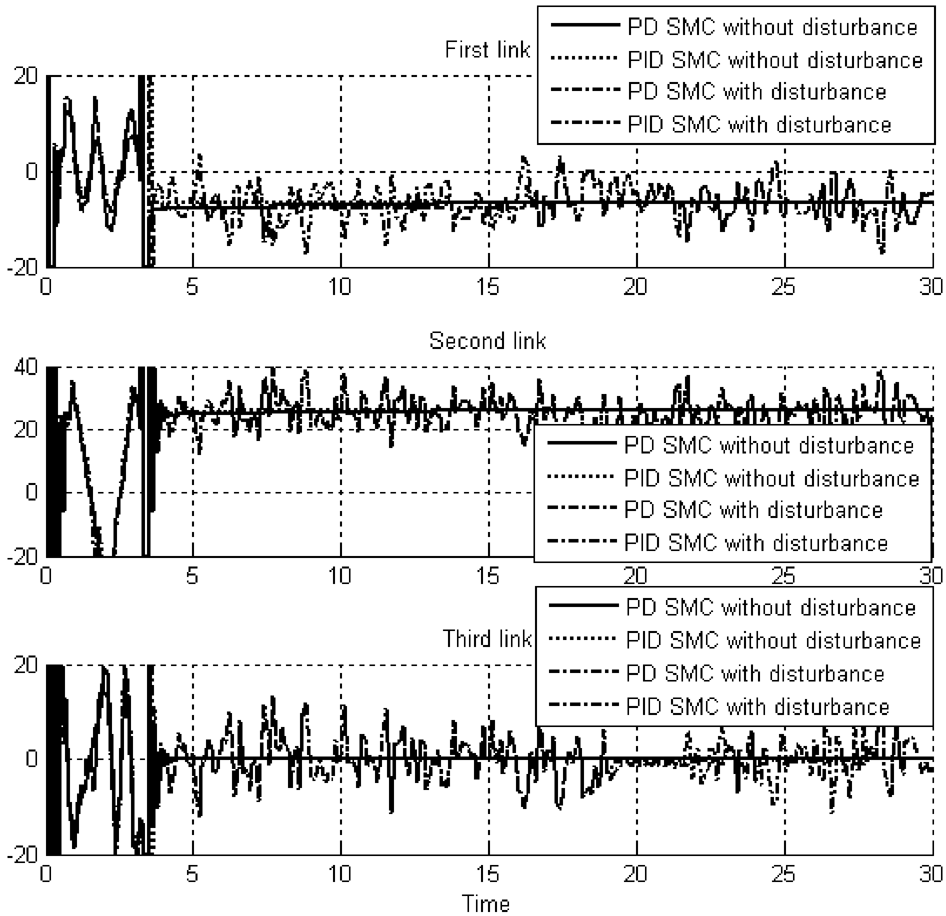


FIGURE 24: Step PID SMC and PD SMC for First, second and third link chattering without and with disturbance.

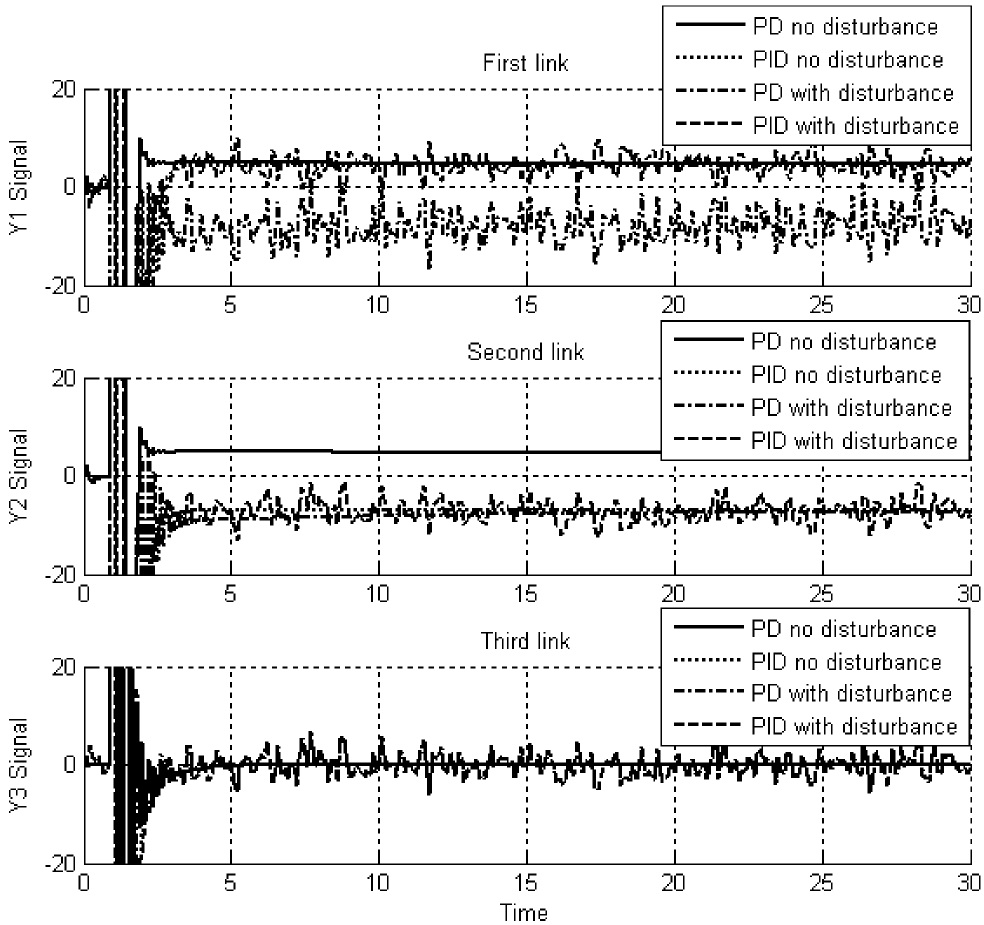


FIGURE 25: Ramp PID SMC and PD SMC for First, second and third link chattering without and with disturbance.

Errors in the model: Figures 26 and 27 have shown the error disturbance in PD and PID SMC. The controllers with no external disturbances have the same error response, but PID SMC has the better steady state error when the robot manipulator has an external disturbance. By comparing steady and RMS error in a system with no disturbance it found that the PID's errors **(Steady State error = 0 and RMS error=1e-8)** are approximately less than PD's **(Steady State error $\cong 1e-6$ and RMS error=$1.2e-6$)**.

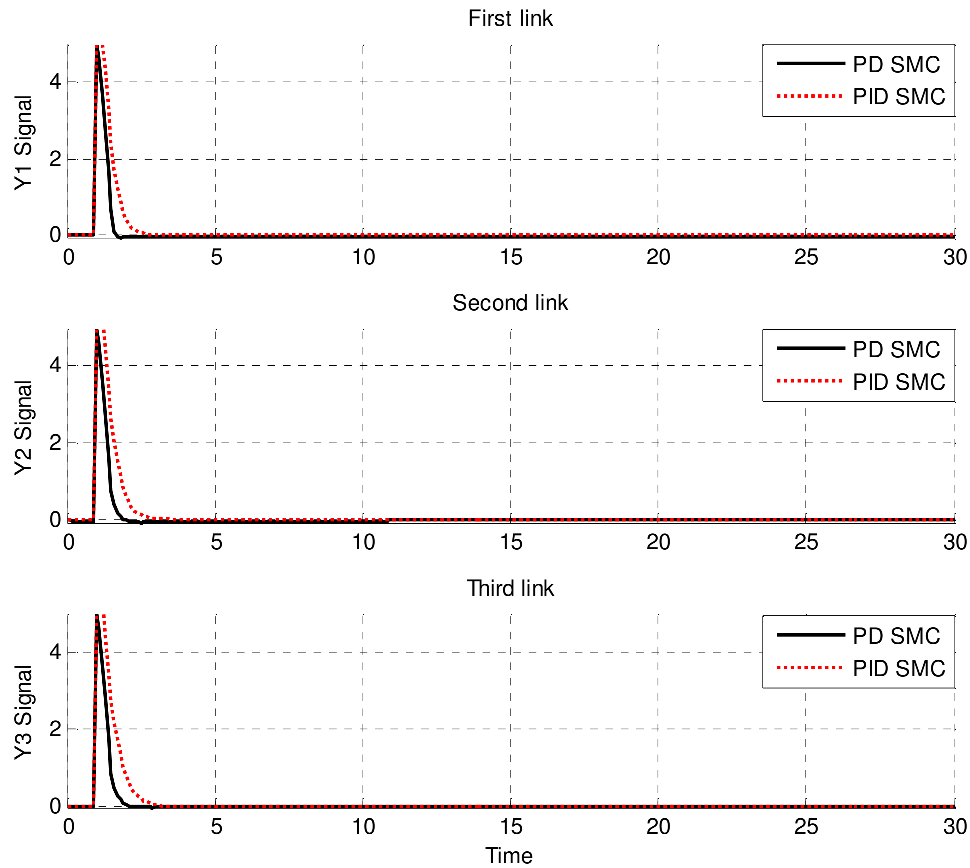


FIGURE 26: Step PID SMC and PD SMC for First, second and third link steady state error performance.

Above graphs (26 and 27) show that in first seconds; PID SMC and PD SMC are increasing very fast. By comparing the steady state error and RMS error it found that the PID's errors **(Steady State error = -0.0007 and RMS error=0.0008)** are fairly less than PD's **(Steady State error** $\cong$ **0.0012 and RMS error=0.0018)**, When disturbance is applied to PD and PID SMC the errors are about 13% growth.

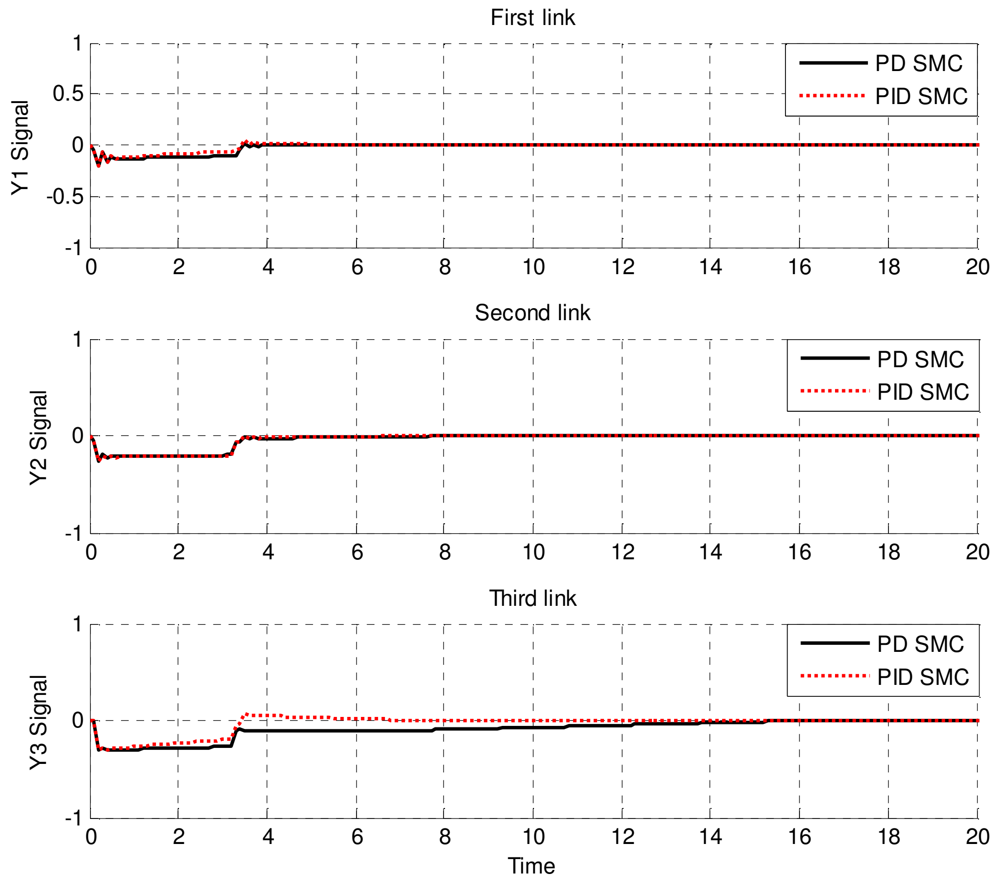


FIGURE 27: Ramp PID SMC and PD SMC for First, second and third link steady state error performance

5. CONCLUSION

In this research we introduced, basic concepts of robot manipulator (e.g., PUMA 560 robot manipulator) and nonlinear control methodology. PUMA 560 robot manipulator is a 6 DOF serial robot manipulator. One of the most active research areas in the field of robotics is robot manipulators control, because these systems are multi-input multi-output (MIMO), nonlinear, and uncertainty. At present, robot manipulators are used in unknown and unstructured situation and caused to provide complicated systems, consequently strong mathematical tools are used in new control methodologies to design nonlinear robust controller with satisfactory performance (e.g., minimum error, good trajectory, disturbance rejection). Sliding mode controller (SMC) is a significant nonlinear controller under condition of partly uncertain dynamic parameters of system. This controller is used to control of highly nonlinear systems especially for robot manipulators, because this controller is a robust and stable. Conversely, pure sliding mode controller is used in many applications; it has an important drawback namely; chattering phenomenon. The chattering phenomenon problem can be reduced by using linear saturation boundary layer function in sliding mode control law. Lyapunov stability is proved in pure sliding mode controller based on switching (sign) function.

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] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, 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, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," IEEE Transactions on Automatic Control, No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," Fuzzy Sets and Systems, No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," IEEE Journal of Robotics and Automation, No. 3, vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, No. 2, 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," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," IEEE conference proceeding, 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," Automatic Control, IEEE Transactions on, No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," Proceedings of the IEEE, No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," IEEE conference proceeding, 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," IEEE Transactions on Industrial Electronics, No. 1, 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, No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," International Journal of Control, No. 2, vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," Mechatronics Journal, No. 1, 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, No. 7, 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, No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," IEEE conference proceeding, 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, No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," IEEE conference proceeding, 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, “Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain,” International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] 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.

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, “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.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , “A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator,” World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli “Design Adaptive Fuzzy Robust Controllers for Robot Manipulator,” World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , “Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line

Tunable Gain: Applied to Robot Manipulator," International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, "Design On-Line Tunable Gain Artificial Nonlinear Controller," Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, " Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine," International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, "Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller," Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, "Novel Sliding Mode Controller for robot manipulator using FPGA," Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, "Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System," International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, "Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ," International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, "Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System," International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, "Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method," International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, "Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control," International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, "Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ," International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, "Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm," International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, "Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ," International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, "Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov

Algorithm: Applied to Robot Manipulator," International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, "Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator," International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, "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.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, "Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ," International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, "Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ," International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, "On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory," International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, "Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator," International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, "Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine," International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.," Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine," world applied science journal (WASJ), 15 (3): 422-428, 2011.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," Fuzzy Systems, IEEE Transactions on, No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," International Journal of Modelling, Identification and Control, No. 1, 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, No. 2, 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, No. 2, vol. 19, pp. 362-368, 2004.

[62] N. Sulaiman, Z. A. Obaid, M. Marhaban and M. Hamidon , "Design and Implementation of FPGA-Based Systems-A Review," Australian Journal of Basic and Applied Sciences, No. 4, vol. 3, pp. 3575-3596, 2009.

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

[64] Y. S. Kung, K. Tseng, C. Chen, H. Sze and A. Wang, "FPGA-implementation of inverse kinematics and servo controller for robot manipulator," Proc. IEEE Int. on Robotics and Biomimetics, pp. 1163–1168, 2006.

[65] X. Shao, D. Sun and J. K. Mills, "A new motion control hardware architecture with FPGA-based IC design for robotic manipulators," IEEE Conference, 2006, pp. 3520-3525.

[66] Y. S. Kung, C. S. Chen and G. S. Shu, "Design and Implementation of a Servo System for Robotic Manipulator," CACS, 2005.

[67] U. D. Meshram and R. Harkare, "FPGA Based Five Axis Robot Arm Controller," IEEE Conference, 2005, pp. 3520-3525.

[68] U. Meshram, P. Bande, P. Dwaramwar and R. Harkare, "Robot arm controller using FPGA," IEEE Conference, 2009, pp. 8-11.

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

[70] Z. A. Obaid, N. Sulaiman and M. Hamidon, "Developed Method of FPGA-based Fuzzy Logic Controller Design with the Aid of Conventional PID Algorithm," Australian Journal of Basic and Applied Sciences, No. 3, vol. 3, pp. 2724-2740, 2009.

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

[72] K. D. Rogers, "Acceleration and implementation of a DSP phase-based frequency estimation algorithm: MATLAB/SIMULINK to FPGA via XILINX system generator," Citeseer, 2004.

[73] F. J. Lin, D. H. Wang and P. K. Huang, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive," IEEE journal of electrical power application, No. 5, Vol. 152, 2005, pp. 1137-1148.

[74] R. R. Ramos, D. Biel, E. Fossas and F. Guinjoan, "A fixed-frequency quasi-sliding control algorithm: application to power inverters design by means of FPGA implementation," Power Electronics, IEEE Transactions on, No. 1, vol. 18, pp. 344-355, 2003.

[75] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," Carleton university Ottawa,2007.

[76] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," IEEE Conference, 2004, pp. 3588-3592.

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

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

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

[80] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," IEEE Conference, 2002, pp. 510-518.

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

[82] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[83] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," Mediterranean Conference, 2002, pp. 160-168.

[84] Samira Soltani & Farzin Piltan, "Design Artificial Nonlinear Controller Based on Computed Torque like Controller with Tunable Gain". World Applied Science Journal,14 (9): 1306-1312, 2011.

[85] Farzin Piltan, H. Rezaie, B. Boroomand, Arman Jahed," Design robust back stepping online tuning feedback linearization control applied to IC engine," International Journal of Advance Science and Technology, 42: 183-204, 2012.

[86] Farzin Piltan, I. Nazari, S. Siamak, P. Ferdosali ,"Methodology of FPGA-based mathematical error-based tuning sliding mode controller" International Journal of Control and Automation, 5(1): 89-110, 2012.

[87] Farzin Piltan, M. A. Dialame, A. Zare, A. Badri ,"Design Novel Lookup table changed Auto Tuning FSMC: Applied to Robot Manipulator" International Journal of Engineering, 6(1): 25-40, 2012.

[88] Farzin Piltan, B. Boroomand, A. Jahed, H. Rezaie ,"Methodology of Mathematical Error-Based Tuning Sliding Mode Controller" International Journal of Engineering, 6(2): 96-112, 2012.

[89] Farzin Piltan, F. Aghayari, M. R. Rashidian, M. Shamsodini, "A New Estimate Sliding Mode Fuzzy Controller for Robotic Manipulator" International Journal of Robotics and Automation, 3(1): 45-58, 2012.

[90] Farzin Piltan, M. Keshavarz, A. Badri, A. Zargari , "Design novel nonlinear controller applied to robot manipulator: design new feedback linearization fuzzy controller with minimum rule base tuning method" International Journal of Robotics and Automation, 3(1): 1-18, 2012.

[91] 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.

[92] Farzin Piltan, A. Hosainpour, E. Mazlomian, M.Shamsodini, M.H Yarmahmoudi. "Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach" International Journal of Robotics and Automation, 3(3): 2012.

[93] Farzin Piltan , M.H. Yarmahmoudi, M. Shamsodini, E.Mazlomian, A.Hosainpour. " PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[94] Farzin Piltan, R. Bayat, F. Aghayari, B. Boroomand. "Design Error-Based Linear Model-Free Evaluation Performance Computed Torque Controller" International Journal of Robotics and Automation, 3(3): 2012.

[95] Farzin Piltan, J. Meigolinedjad, S. Mehrara, S. Rahmdel. " Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology" International Journal of Robotics and Automation, 3(3): 2012.

[96] Farzin Piltan, S. Rahmdel, S. Mehrara, R. Bayat." Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses" International Journal of Engineering, 3(3): 2012.

[97] Farzin Piltan, M. Mirzaie, F. Shahriyari, Iman Nazari & S. Emamzadeh." Design Baseline Computed Torque Controller" International Journal of Engineering, 3(3): 2012.

Design Error-based Linear Model-free Evaluation Performance Computed Torque Controller

Farzin Piltan 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

Reza Bayat *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

Farid Aghayari *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

Bamdad Boroomand *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

Design a nonlinear controller for second order nonlinear uncertain dynamical systems is one of the most important challenging works. This research focuses on the design, implementation and analysis of a model-free linear error-based tuning computed torque controller for highly nonlinear dynamic second order system, in presence of uncertainties. In order to provide high performance nonlinear methodology, computed torque controller is selected. Pure computed torque controller can be used to control of partly known nonlinear dynamic parameters of nonlinear systems. Conversely, pure computed torque controller is used in many applications; it has an important drawback namely; nonlinear equivalent dynamic formulation in uncertain dynamic parameter. In order to solve the uncertain nonlinear dynamic parameters, implement easily and avoid mathematical model base controller, model-free performance/error-based linear methodology with three inputs and one output is applied to pure computed torque controller. The results demonstrate that the error-based linear tuning computed torque controller is a model-based controllers which works well in certain and uncertain system. Pure computed torque controller has difficulty in handling unstructured model uncertainties. To solve this problem applied linear model-free error -based tuning method to computed torque controller for adjusting the linear inner loop gain (K). Since the linear inner loop gain (K) is adjusted by linear error-based tuning method, it is linear and continuous. In this research new K is obtained by the previous K multiple gain updating factor (α) which is a coefficient varies between half to two.

Keywords: Computed Torque Controller, Linear on-line Tuning Method, Gain Updating Factor, linear Inner loop Gain, Error-based Tuning Method.

1. INTRODUCTION, BACKGROUND and MOTIVATION

Controller is a device which can sense information from linear or nonlinear system (e.g., robot manipulator) to improve the systems performance [1-3]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[4-5]. Several industrial robot manipulators are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when robot manipulator works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, parameter variations). In some applications robot manipulators are used in an 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). Computed torque controller is an influential nonlinear controller to certain and partly uncertain systems which it is based on system's dynamic model [6-20].

Computed torque controller (CTC) is a powerful nonlinear controller which it widely used in control of robot manipulator. It is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. This controller works very well when all dynamic and physical parameters are known but when the robot manipulator has variation in dynamic parameters, in this situation the controller has no acceptable performance[14]. In practice, most of physical systems (e.g., robot manipulators) parameters are unknown or time variant, therefore, computed torque like controller used to compensate dynamic equation of robot manipulator[1, 6]. Research on computed torque controller is significantly growing on robot manipulator application which has been reported in [1, 6, 15-16]. Vivas and Mosquera [15]have proposed a predictive functional controller and compare to computed torque controller for tracking response in uncertain environment. However both controllers have been used in feedback linearization, but predictive strategy gives better result as a performance. A computed torque control with non parametric regression models have been presented for a robot arm[16]. This controller also has been problem in uncertain dynamic models. Based on [1, 6]and [15-16] computed torque controller is a significant nonlinear controller to certain systems which it is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. When all dynamic and physical parameters are known, computed torque controller works fantastically; practically a large amount of systems have uncertainties, therefore sliding mode controller is one of the best case to solve this challenge.

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 [21-75]. 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, error-based linear adaptive method is applied to computed torque controller. Hsu et al. [54] have presented traditional adaptive methods which can update fuzzy rules to compensate nonlinear parameters and guarantee the stability. Hsueh et al. [43] have presented traditional self tuning method which can resolve the controller problem.

For nonlinear dynamic systems 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 pure computed torque controller, as a result it is used to adjust and tune coefficient.

This paper is organized as follows:
In section 2, detail of classical computed torque controller is presented. In section 3, design error-based linear tuning method is presented and applied to computed torque controller; this method

is used to reduce the error performance and estimation the equivalent part. In section 4, simulation result is presented and finally in section 5, the conclusion is presented.

2. CASE STUDY and COMPUTED TORQUE CONTROLLER FORMULATION

Case study (Robot manipulator dynamic formulation): The equation of an *n-DOF* robot manipulator governed by the following equation [1, 3, 15-29]:

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

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:

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

Where B(q) is the matrix of coriolios torques, C(q) is the matrix of centrifugal torques, and G(q) is the vector of gravity force. The dynamic terms in equation (2) are only manipulator position. This is a decoupled system with simple second order linear differential dynamics. In other words, the component $\ddot{q}$ influences, with a double integrator relationship, only the joint variableq_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3, 10-29]:

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

Computed Torque Controller: The central idea of Computed torque controller (CTC) is feedback linearization so, originally this algorithm is called feedback linearization controller. It has assumed that the desired motion trajectory for the manipulator $\boldsymbol{q_d(t)}$, as determined, by a path planner. Defines the tracking error as:

$$\boldsymbol{e(t) = q_d(t) - q_a(t)} \tag{4}$$

Where e(t) is error of the plant, $\boldsymbol{q_d(t)}$ is desired input variable, that in our system is desired displacement, $\boldsymbol{q_a(t)}$ is actual displacement. If an alternative linear state-space equation in the form $\boldsymbol{\dot{x} = Ax + BU}$ can be defined as

$$\boldsymbol{\dot{x} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} U} \tag{5}$$

With $U = -M^{-1}(q).N(q,\dot{q}) + M^{-1}(q).\tau$ and this is known as the Brunousky canonical form. By equation (4) and (5) the Brunousky canonical form can be written in terms of the state $x = [e^T\ \dot{e}^T]^T$ as [1]:

$$\boldsymbol{\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} . \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} U} \tag{6}$$

With

$$\boldsymbol{U = \ddot{q}_d + M^{-1}(q).\{N(q.\dot{q}) - \tau\}} \tag{7}$$

Then compute the required arm torques using inverse of equation (7), is;

$$\boldsymbol{\tau = M(q)(\ddot{q}_d - U) + N(\dot{q},q)} \tag{8}$$

This is a nonlinear feedback control law that guarantees tracking of desired trajectory. Selecting proportional-plus-derivative (PD) feedback for U(t) results in the PD-computed torque controller [6];

$$\boldsymbol{\tau = M(q)\left(\ddot{q}_d + K_v\dot{e} + K_p e\right) + N(q,\dot{q})} \tag{9}$$

and the resulting linear error dynamics are

$$\boldsymbol{\left(\ddot{q}_d + K_v\dot{e} + K_p e\right) = 0} \tag{10}$$

According to the linear system theory, convergence of the tracking error to zero is guaranteed [6]. Where $\boldsymbol{K_p}$ and $\boldsymbol{K_v}$ are the controller gains. The result schemes is shown in Figure 1, in which two feedback loops, namely, inner loop and outer loop, which an inner loop is a compensate loop and an outer loop is a tracking error loop.

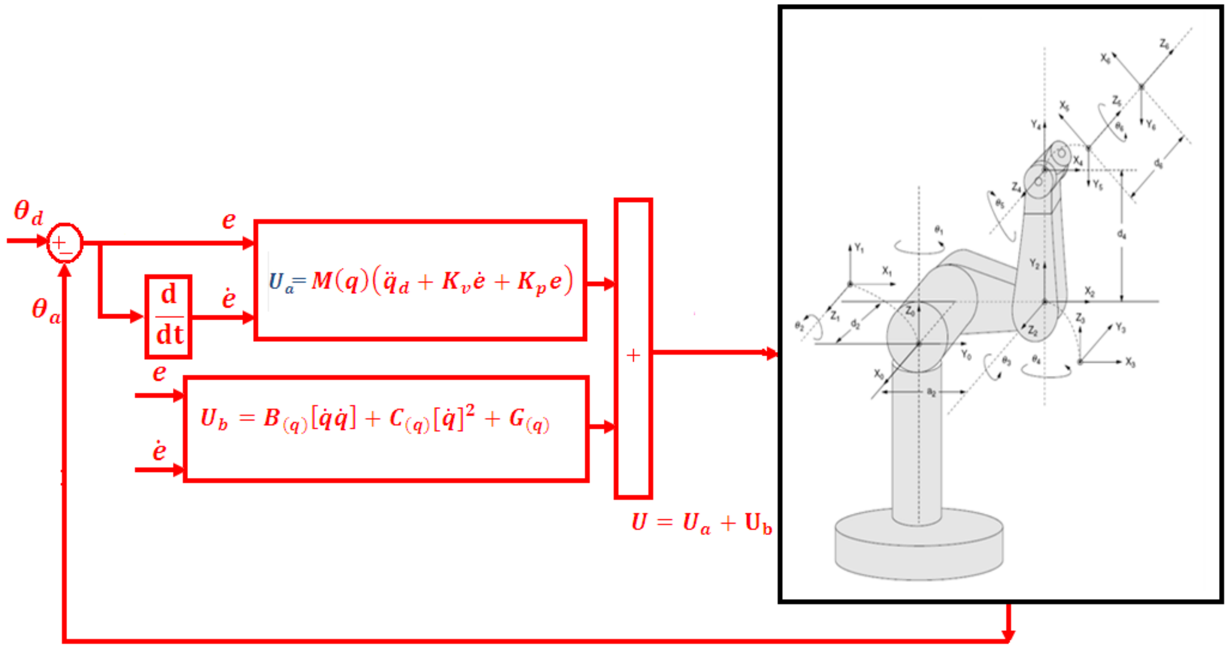


FIGURE 1: Block diagram of PD-computed torque controller (PD-CTC)

The application of proportional-plus-derivative (PD) computed torque controller to control of 6 DOF robot manipulator introduced in this part. Suppose that in (9) the nonlinearity term defined by the following term

$$N(q,\dot{q}) = B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q) = \begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2 \\ C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2 \\ C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2 \\ 0 \\ C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \tag{11}$$

Therefore the equation of PD-CTC for control of robot manipulator is written as the equation of (12);

$$\begin{bmatrix} \widehat{\tau_1} \\ \widehat{\tau_2} \\ \widehat{\tau_3} \\ \widehat{\tau_4} \\ \widehat{\tau_5} \\ \widehat{\tau_6} \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix} \begin{bmatrix} \ddot{q}_{d1} + K_{v1}\dot{e}_1 + K_{p1}e_1 \\ \ddot{q}_{d2} + K_{v2}\dot{e}_2 + K_{p2}e_2 \\ \ddot{q}_{d3} + K_{v3}\dot{e}_3 + K_{p3}e_3 \\ \ddot{q}_{d4} + K_{v4}\dot{e}_4 + K_{p4}e_4 \\ \ddot{q}_{d5} + K_{v5}\dot{e}_5 + K_{p5}e_5 \\ \ddot{q}_{d6} + K_{v6}\dot{e}_6 + K_{p6}e_6 \end{bmatrix} + \tag{12}$$

$$\begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2 \\ C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2 \\ C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2 \\ 0 \\ C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix}$$

The controller based on a formulation (12) is related to robot dynamics therefore it has problems in uncertain conditions.

3. METHODOLOGY: ERROR-BASED LINEAR MODEL-FREE TUNING COMPUTED TORQUE CONTROLLER

Computed torque controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining computed torque controller and linear error-based tuning method which this method can helps to eliminate the error and improves the system's tracking performance by online tuning method. In this research the nonlinear equivalent dynamic (equivalent part) formulation problem in uncertain system is solved by using on-line linear error-based tuning theorem. In this method linear error-based theorem is applied to computed torque controller to adjust the linear inner loop gain. Computed torque controller has difficulty in handling unstructured model uncertainties and this controller's performance is sensitive to linear inner loop gain. It is possible to solve above challenge by combining linear error-based tuning method and computed torque controller which this methodology can help to improve system's tracking performance by on-line tuning (linear error-based tuning) method. Based on above discussion, compute the best value of linear inner loop gain has played important role to improve system's tracking performance especially when the system parameters are unknown or uncertain. This problem is solved by tuning the linear inner loop gain ($\boldsymbol{K}$) of the computed torque controller continuously in real-time. In this methodology, the system's performance is improved with respect to the pure computed torque controller. Figure 2 shows the linear error-based tuning computed torque controller.

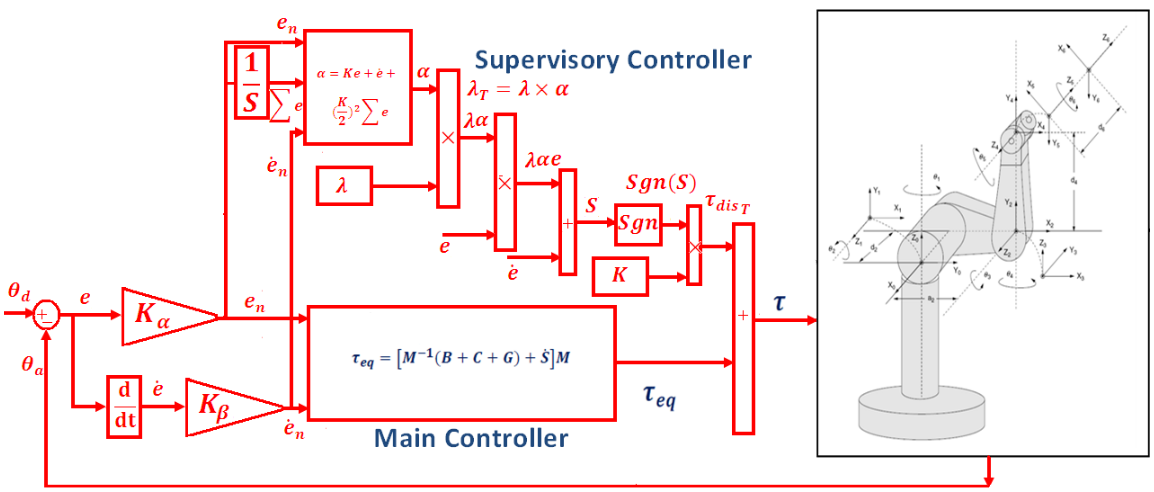


FIGURE 2: Block diagram of a linear error-based computed torque controller: applied to robot arm

$$\hat{f}(x|\lambda) = \lambda^T \alpha \quad (13)$$

If minimum error (λ^*) is defined by;

$$\lambda^* = \arg\min [(Sup|\hat{f}(x|\lambda) - f(x)|)] \quad (14)$$

Where λ^T is adjusted by an adaption law and this law is designed to minimize the error's parameters of $\lambda - \lambda^*$. adaption law in linear error-based tuning computed torque controller is used to adjust the linear inner loop gain. Linear error-based tuning part is a supervisory controller based on the following formulation methodology. This controller has three inputs namely; error (e), change of error $(\dot{e})$ and the integral of error $(\sum e)$ and an output namely; gain updating factor(α). As a summary design a linear error-based tuning is based on the following formulation:

$$\boldsymbol{\alpha = K.e + \dot{e} + \frac{(K)^2}{2}\sum e} \tag{15}$$

$$\boldsymbol{K_{on-line} = \alpha.\lambda e + \dot{e} \Rightarrow K_{on-line} = (K.e + \dot{e} + \frac{(K)^2}{2}\sum e)\lambda e + \dot{e}}$$

$$\boldsymbol{\lambda_{Tune} = \lambda.\alpha \Rightarrow \lambda_{Tune} = \lambda(K.e + \dot{e} + \frac{(K)^2}{2}\sum e)}$$

Where (α) is gain updating factor, $(\sum e)$ is the integral of error, $(\dot{e})$ is change of error, (e) is error and K is a coefficient.

4 Simulation Results

Pure computed torque controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining computed torque controller and linear error-based tuning in a single controller method. This method can improve the system's tracking performance by online tuning method. This method is based on resolve the on line linear inner loop gain as well as improve the output performance by tuning the linear inner loop coefficient. The linear inner loop gain $(\boldsymbol{K})$ of this controller is adjusted online depending on the last values of error (e), change of error $(\dot{e})$ and integral of error by gain updating factor (α).

Tracking performances: Based on (12) in computed torque controller; controller performance is depended on the linear inner loop gain updating factor (K). These two coefficients are computed by trial and error in CTC. The best possible coefficients in step CTC are; $K_p = K_v = K_i = 18$. In linear error-based tuning computed torque controller the linear inner loop gain is adjusted online depending on the last values of error (e), change of error $(\dot{e})$ and integral of error by gain updating factor (α). Figure 3 shows tracking performance in computed torque controller (CTC) and linear tuning computed torque controller (LTCTC) without disturbance for step trajectory.

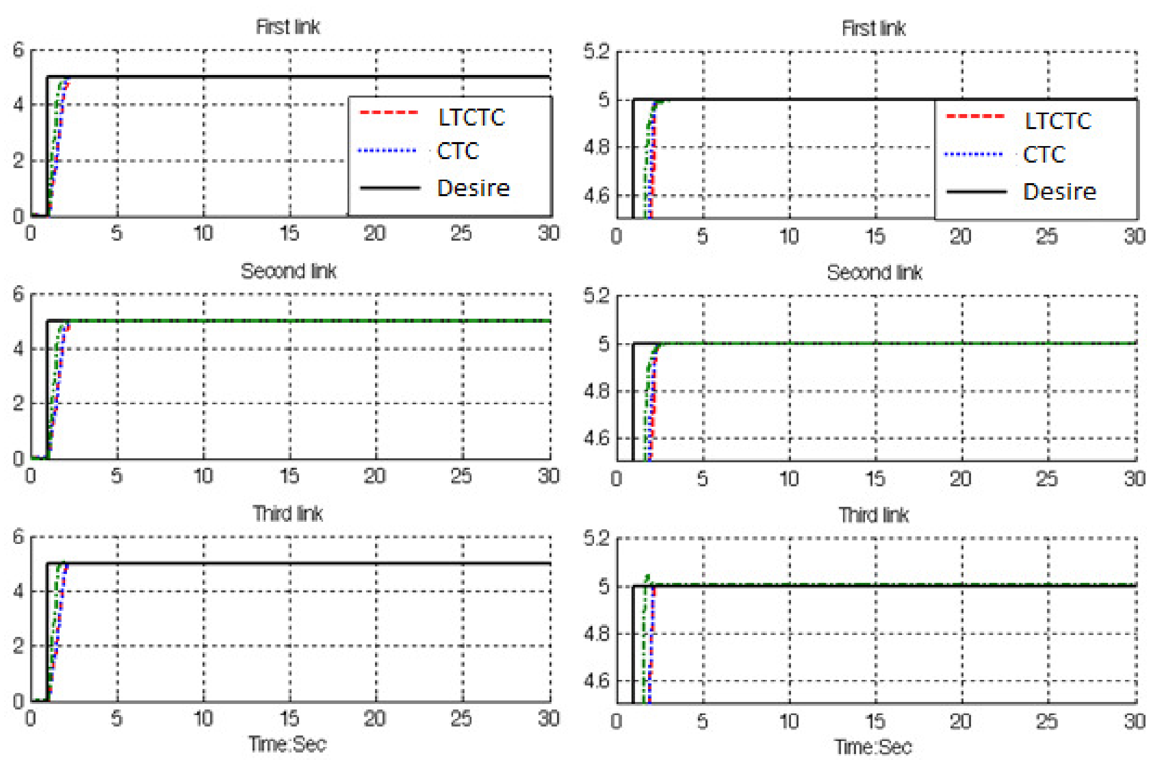


FIGURE 3: CTC, LTCTC and desired input for first, second and third link step trajectory performance without disturbance

Based on Figure 3 it is observed that, the overshoot in LTCTC is 0% and in CTC's is 1%, and rise time in LTCTC's is 0.6 seconds and in CTC's is 0.483 second. From the trajectory MATLAB simulation for LTCTC and CTC in certain system, it was seen that all of two controllers have acceptable performance.

Disturbance Rejection: Figures 4 to 6 show the power disturbance elimination in LTCTC and CTC with disturbance for step trajectory. The disturbance rejection is used to test the robustness comparisons of these two controllers for step trajectory. A band limited white noise with predefined of 10%, 20% and 40% the power of input signal value is applied to the step trajectory. It found fairly fluctuations in trajectory responses.

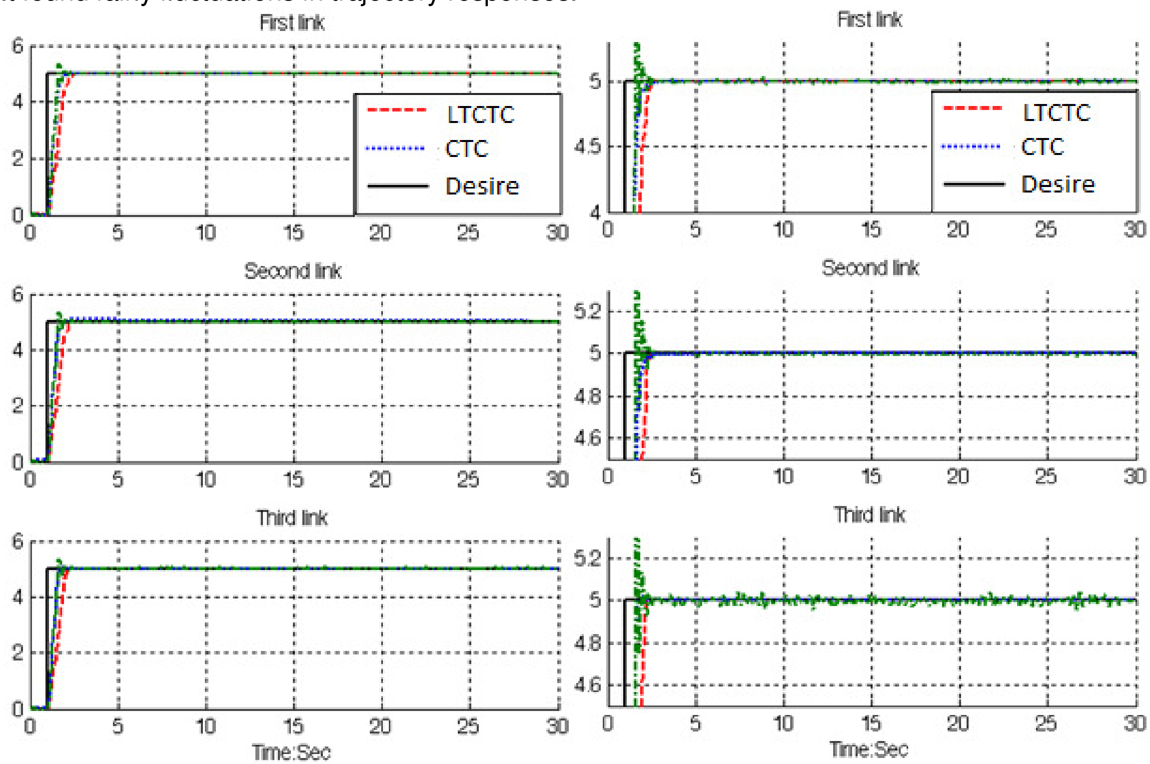


FIGURE 4: Desired input, LTCTC and CTC for first, second and third link trajectory with 10%external disturbance: step trajectory

Based on Figure 4; by comparing step response trajectory with 10% disturbance of relative to the input signal amplitude in LTCTC and CTC, LTCTC's overshoot about (0%) is lower than PD-CTC's (**1%**). CTC's rise time **(0.5 seconds)** is lower than LTCTC's **(0.65 second).** Besides the Steady State and RMS error in LTCTC and CTC it is observed that, error performances in LTCTC **(Steady State error =1.08e-12 and RMS error=1.5e-12)** are bout lower than CTC's **(Steady State error=1.6e-6 and RMS error=1.9e-6)**.

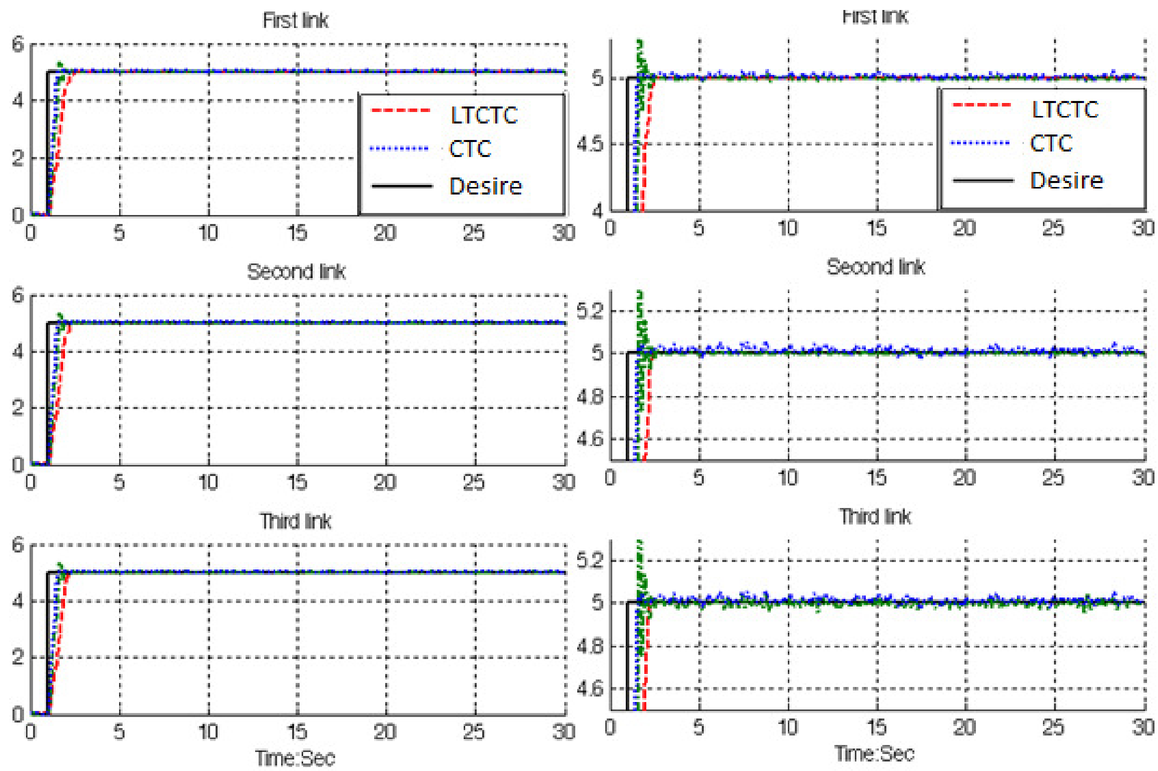


FIGURE 5: Desired input, LTCTC and CTC for first, second and third link trajectory with 20%external disturbance: step trajectory

Based on Figure 5; by comparing step response trajectory with 20% disturbance of relative to the input signal amplitude in LTCTC and CTC, LTCTC's overshoot about **(0%)** is lower than CTC's (**2.1%**)**.** CTC's rise time **(0.5 seconds)** is lower than LTCTC's **(0.66 second).** Besides the Steady State and RMS error in LTCTC and CTC it is observed that, error performances in LTCTC **(Steady State error =1.2e-12 and RMS error=1.8e-12)** are about lower than CTC's **(Steady State error=1.8e-5 and RMS error=2e-5)**. Based on Figure 5, it was seen that, LTCTC's performance is better than CTC because LTCTC can auto-tune the inner loop gain coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas CTC cannot.

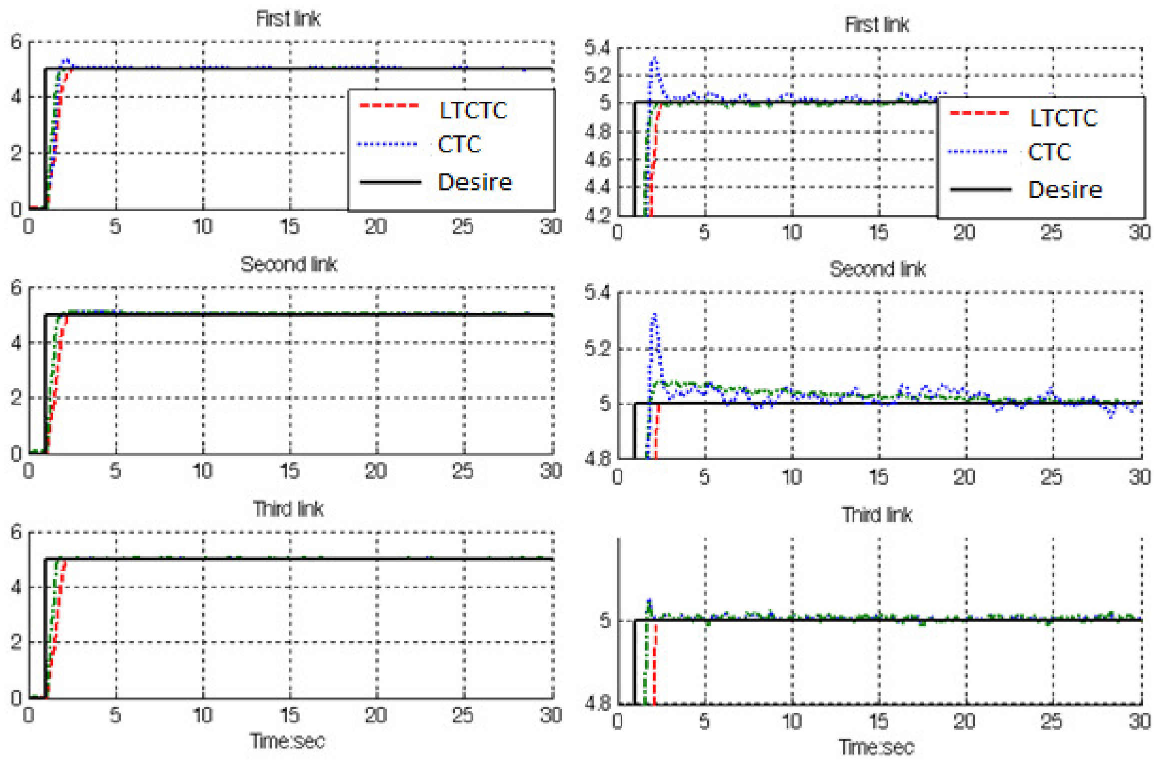


FIGURE 6: Desired input, LTCTC and CTC for first, second and third link trajectory with 40%external disturbance: step trajectory

Based on Figure 6; by comparing step response trajectory with 40% disturbance of relative to the input signal amplitude in LTCTC and CTC, LTCTC's overshoot about **(0%)** is lower than CTC (**8%**). CTC's rise time **(0.5 seconds)** is lower than LTCTC's **(0.8 second).** Besides the Steady State and RMS error in LTCTC and CTC it is observed that, error performances in LTCTC **(Steady State error =1.3e-12 and RMS error=1.8e-12)** are about lower than CTC's **(Steady State error=10e-4 and RMS error=11e-4).** Based on Figure 6, CTC has moderately oscillation in trajectory response with regard to 40% of the input signal amplitude disturbance but LTCTC has stability in trajectory responses in presence of uncertainty and external disturbance. Based on Figure 6 in presence of 40% unstructured disturbance, LTCTC's is more robust than CTC because LTCTC can auto-tune the inner loop coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas CTC cannot.

Steady state error: The error performance is used to test the disturbance effect comparisons of these controllers for step trajectory. All three joint's inputs are step function with the same step time (step time= 1 second), the same initial value (initial value=0) and the same final value (final value=5). LTCTC's rise time is about 0.6 second, and CTC's rise time is about 0.6 second which caused to create a needle wave in the range of 5 (amplitude=5) and the different width. In this system this time is transient time and this part of error introduced as a transient error. Besides the Steady State and RMS error in LTCTC and CTC it is observed that, error performances in LTCTC **(Steady State error =0.9e-12 and RMS error=1.1e-12)** are bout lower than CTC's **(Steady State error=1e-8 and RMS error=1.2e-6)**. The LTCTC gives significant steady state error performance when compared to CTC. When applied 40% disturbances in LTCTC the RMS error increased approximately 0.0164% (percent of increase the LTCTC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{1.8e-12}{1.1e-12} = 0.0164\%)$, in CTC the RMS error increased approximately 9.17% (percent of increase the PD-SMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbance\ RMS\ error} = \frac{11e-4}{1.2e-6} = 9.17\%)$. In this part LTCTC and CTC have been comparatively evaluation through

MATLAB simulation, for PUMA robot manipulator control. It is observed that however LTCTC is dependent of nonlinear dynamic equation of PUMA 560 robot manipulator but it can guarantee the trajectory following in certain systems, structure uncertain systems and unstructured model uncertainties by online tuning method.

5. CONCLUSIONS

Refer to this research, a linear model-free error-based tuning computed torque controller (LTCTC) is proposed for robot manipulator. The first problem of the pure CTC was adjust the linear inner loop gain in certain and uncertain systems. this problem can be reduced in certain system by using trial and error methodology in computed torque control law. The simulation results exhibit that the CTC works well in certain system. The nonlinear equivalent dynamic problem in uncertain system is solved by using on-line tuning method. Pure CTC has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining error-based linear methodology and computed torque controller. Since the linear inner loop gain (K) is adjusted by linear tuning method. The linear inner loop gain updating factor (α) of error-based tuning part can be changed with the changes in error and change of error and integral of error rate between half to two. linear inner loop gain is adapted on-line by linear inner loop gain updating factor. In pure CTC the linear inner loop gain is chosen by trial and error, which means pure CTC has to have a prior knowledge of the system uncertainty. If the knowledge is not available error performance is go up. In LTCTC the linear inner loop gain is updated on-line to compensate the system unstructured uncertainty. The simulation results exhibit that the LTCTC works well in various situations. Based on theoretical and simulation results, it is observed that LTCTC is a model-based stable control for robot manipulator. It is a good solution to reduce the error in structure and unstructured uncertainties.

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] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, *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, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* No. 3, vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, No. 2,* 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," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," *IEEE conference proceeding,* 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE conference proceeding,* 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* No. 1, 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,* No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* No. 2, vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics Journal,* No. 1, 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,* No. 7, 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,* No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," *IEEE conference proceeding,* 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,* No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," *IEEE conference proceeding,* 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] 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

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, "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.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli "Design Adaptive Fuzzy Robust Controllers for Robot Manipulator," World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , "Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line Tunable Gain: Applied to Robot Manipulator," International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, "Design On-Line Tunable Gain Artificial Nonlinear Controller," Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, " Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine," International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, "Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller," Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, "Novel Sliding Mode Controller for robot manipulator using FPGA," Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, "Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System," International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, “Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, “Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System,” International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, “Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method,” International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, “Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control,” International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, “Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ,” International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, “Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm,” International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, “Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ,” International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, “Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator,” International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator,” International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, “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.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, “Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ,” International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, “Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, “On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode

Controller Based on Lyaponuv Theory," International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, "Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator," International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, "Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine," International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.," Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine," world applied science journal (WASJ), 15 (3): 422-428, 2011.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* No. 1, 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,* No. 2, 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,* No. 2, vol. 19, pp. 362-368, 2004.

[62] N. Sulaiman, Z. A. Obaid, M. Marhaban and M. Hamidon , "Design and Implementation of FPGA-Based Systems-A Review," *Australian Journal of Basic and Applied Sciences,* No. 4, vol. 3, pp. 3575-3596, 2009.

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

[64] Y. S. Kung, K. Tseng, C. Chen, H. Sze and A. Wang, "FPGA-implementation of inverse kinematics and servo controller for robot manipulator," *Proc. IEEE Int. on Robotics and Biomimetics,* pp. 1163–1168, 2006.

[65] X. Shao, D. Sun and J. K. Mills, "A new motion control hardware architecture with FPGA-based IC design for robotic manipulators," *IEEE Conference,* 2006, pp. 3520-3525.

[66] Y. S. Kung, C. S. Chen and G. S. Shu, "Design and Implementation of a Servo System for Robotic Manipulator," CACS, 2005.

[67] U. D. Meshram and R. Harkare, "FPGA Based Five Axis Robot Arm Controller," *IEEE Conference,* 2005, pp. 3520-3525.

[68] U. Meshram, P. Bande, P. Dwaramwar and R. Harkare, "Robot arm controller using FPGA," *IEEE Conference,* 2009, pp. 8-11.

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

[70] Z. A. Obaid, N. Sulaiman and M. Hamidon, "Developed Method of FPGA-based Fuzzy Logic Controller Design with the Aid of Conventional PID Algorithm," *Australian Journal of Basic and Applied Sciences,* No. 3, vol. 3, pp. 2724-2740, 2009.

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

[72] K. D. Rogers, "Acceleration and implementation of a DSP phase-based frequency estimation algorithm: MATLAB/SIMULINK to FPGA via XILINX system generator," Citeseer, 2004.

[73] F. J. Lin, D. H. Wang and P. K. Huang, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive," IEEE journal of electrical power application, No. 5, Vol. 152, 2005, pp. 1137-1148.

[74] R. R. Ramos, D. Biel, E. Fossas and F. Guinjoan, "A fixed-frequency quasi-sliding control algorithm: application to power inverters design by means of FPGA implementation," *Power Electronics, IEEE Transactions on,* No. 1, vol. 18, pp. 344-355, 2003.

[75] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," *Carleton university Ottawa,*2007.

[76] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," *IEEE Conference,* 2004, pp. 3588-3592.

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

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

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

[80] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," *IEEE Conference,* 2002, pp. 510-518.

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

[82] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[83] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* 2002, pp. 160-168.

[84] Farzin Piltan, H. Rezaie, B. Boroomand, Arman Jahed," Design robust back stepping online tuning feedback linearization control applied to IC engine," International Journal of Advance Science and Technology, 42: 183-204, 2012.

[85] Farzin Piltan, I. Nazari, S. Siamak, P. Ferdosali ,"Methodology of FPGA-based mathematical error-based tuning sliding mode controller" International Journal of Control and Automation, 5(1): 89-110, 2012.

[86] Farzin Piltan, M. A. Dialame, A. Zare, A. Badri ,"Design Novel Lookup table changed Auto Tuning FSMC: Applied to Robot Manipulator" International Journal of Engineering, 6(1): 25-40, 2012.

[87] Farzin Piltan, B. Boroomand, A. Jahed, H. Rezaie ,"Methodology of Mathematical Error-Based Tuning Sliding Mode Controller" International Journal of Engineering, 6(2): 96-112, 2012.

[88] Farzin Piltan, F. Aghayari, M. R. Rashidian, M. Shamsodini, "A New Estimate Sliding Mode Fuzzy Controller for Robotic Manipulator" International Journal of Robotics and Automation, 3(1): 45-58, 2012.

[89] Farzin Piltan, M. Keshavarz, A. Badri, A. Zargari , "Design novel nonlinear controller applied to robot manipulator: design new feedback linearization fuzzy controller with minimum rule base tuning method" International Journal of Robotics and Automation, 3(1): 1-18, 2012.

[90] Samira Soltani & Farzin Piltan, "*Design Artificial Nonlinear Controller Based on Computed Torque like Controller with Tunable Gain*". World Applied Science Journal,14 (9): 1306-1312, 2011.

[91] 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.

[92] Farzin Piltan, A. Hosainpour, E. Mazlomian, M.Shamsodini, M.H Yarmahmoudi. "Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach" International Journal of Robotics and Automation, 3(3): 2012.

[93] Farzin Piltan , M.H. Yarmahmoudi, M. Shamsodini, E.Mazlomian, A.Hosainpour. " PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[94] Farzin Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari & Mina Mirazaei . " PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[95] Farzin Piltan, J. Meigolinedjad, S. Mehrara, S. Rahmdel. " Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology" International Journal of Robotics and Automation, 3(3): 2012.

[96] Farzin Piltan, S. Rahmdel, S. Mehrara, R. Bayat." Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses" International Journal of Engineering, 3(3): 2012.

[97] Farzin Piltan, M. Mirzaie, F. Shahriyari, Iman nazari & S. Emamzadeh." Design Baseline Computed Torque Controller" International Journal of Engineering, 3(3): 2012.

PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses

Farzin Piltan *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

Mohammad Hossein Yarmahmoudi *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

Mohammad Shamsodini *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

Ebrahim Mazlomian *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

Ali Hosainpour 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 describes the MATLAB/SIMULINK realization of the PUMA 560 robot manipulator position control methodology. This paper focuses on design, analyzed and implements nonlinear computed torque control (CTC) methods. These simulation models are developed as a part of a software laboratory to support and enhance graduate/undergraduate robotics courses, nonlinear control courses and MATLAB/SIMULINK courses at research and development company (SSP Co.) research center, Shiraz, Iran.

Keywords: MATLAB/SIMULINK, PUMA 560 Robot Manipulator, Position Control Method, Computed Torque Control, Robotics, Nonlinear Control.

1. INTRODUCTION

Computer modeling, simulation and implementation tools have been widely used to support and develop nonlinear control, robotics, and MATLAB/SIMULINK courses. MATLAB with its toolboxes such as SIMULINK [1] is one of the most accepted software packages used by researchers to enhance teaching the transient and steady-state characteristics of control and robotic courses [3_7]. In an effort to modeling and implement robotics, nonlinear control and advanced MATLAB/SIMULINK courses at research and development SSP Co., authors have developed MATLAB/SIMULINK models for learn the basic information in field of nonlinear control and industrial robot manipulator [8, 9].

Controller is a device which can sense information from linear or nonlinear system (e.g., robot manipulator) to improve the systems performance [3]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[5]. Several industrial robot manipulators are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when robot manipulator works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, parameter variations). In some applications robot manipulators are used in an 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). Joint space and operational space control are closed loop controllers which they have been used to provide robustness and rejection of disturbance effect. The main target in joint space controller is design a feedback controller that allows the actual motion ($q_a(t)$) tracking of the desired motion ($q_d(t)$). This control problem is classified into two main groups. Firstly, transformation the desired motion $X_d(t)$ to joint variable $q_d(t)$ by inverse kinematics of robot manipulators [6]. Figure 1 shows the main block diagram of joint space controller. The main target in operational space controller is to design a feedback controller to allow the actual end-effector motion $X_a(t)$ to track the desired endeffector motion $X_d(t)$. This control methodology requires a greater algorithmic complexity and the inverse kinematics used in the feedback control loop. Direct measurement of operational space variables are very expensive that caused to limitation used of this controller in industrial robot manipulators[6]. Figure 2 shows the main block diagram of operational space control.

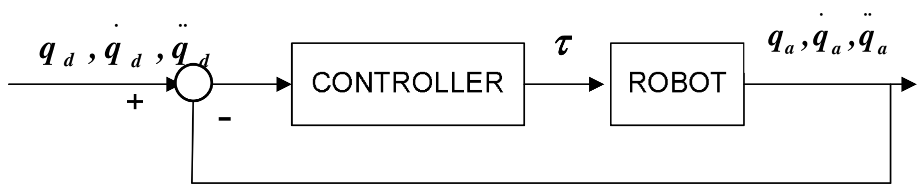


FIGURE 1: Block diagram of joint space control

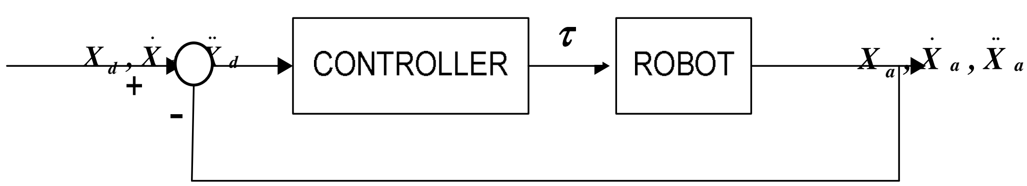


FIGURE 2: Block diagram of operational space control

Computed torque controller (CTC) is a powerful nonlinear controller which it widely used in control of robot manipulator. It is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. This controller works very well when all dynamic and physical parameters are known but when the robot manipulator has variation in dynamic parameters, in this situation the controller has no acceptable performance[14]. In practice, most of physical systems (e.g., robot manipulators) parameters are unknown or time variant, therefore, computed torque like controller used to compensate dynamic equation of robot manipulator[1, 6]. Research on computed torque controller is significantly growing on robot manipulator application which has been reported in [1, 6, 15-16]. Vivas and Mosquera [15]have proposed a predictive functional controller and compare to computed torque controller for tracking response in uncertain environment. However both controllers have been used in feedback linearization, but predictive strategy gives better result as a performance. A computed torque control with non parametric regression models have been presented for a robot arm[16]. This controller also has been problem in uncertain dynamic models. Based on [1, 6]and [15-16] computed torque controller is a significant nonlinear controller to certain systems which it is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law.

This paper is organized as follows:
In section 2, dynamic formulation of robot manipulator is presented. Detail of classical CTC and MATLAB/SIMULINK implementation of this controller is presented in section 3. In section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. PUMA 560 ROBOT MANIPULATOR DYNAMIC FORMULATION

Dynamics of PUMA560 Robot Manipulator: To position control of robot manipulator, the second three axes are locked the dynamic equation of PUMA robot manipulator is given by [77-80];

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

Where

$$M(q)=\begin{bmatrix}M_{11} & M_{12} & M_{13} & 0 & 0 & 0\\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0\\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0\\ 0 & 0 & 0 & M_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & M_{55} & 0\\ 0 & 0 & 0 & 0 & 0 & M_{66}\end{bmatrix} \quad (2)$$

M is computed as

$$M_{11} = I_{m1}+I_1+I_3\times\cos(\theta_2)\cos(\theta_2)+I_7\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)+I_{10}\sin(\theta_2+\theta_3 \; I_{11}\sin(\theta_2)\cos(\theta_2)+I_{21}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)+2+[I_5\cos(\theta_2)\sin(\theta_2+\theta_3) \; I_{12}\cos(\theta_2)\cos(\theta_2+\theta_3)+I_{15}\sin(\theta_2+\theta_3)\sin(\theta_2+\theta_3)+I_{16}\cos(\theta_2)\sin(\theta_2+\theta_3) \; \theta_3)\cos(\theta_2+\theta_3) \quad (3)$$

$$M_{12} = I_4\sin(\theta_2)+I_8\cos(\theta_2+\theta_3)+I_9\cos(\theta_2)+I_{13}\sin(\theta_2+\theta_3)-I_{18}\cos(\theta_2+ \quad (4)$$

$$M_{13} = I_8\cos(\theta_2+\theta_3) + I_{13}\sin(\theta_2+\theta_3) - I_{18}\cos(\theta_2+\theta_3) \tag{5}$$

$$M_{22} = I_{m2} + I_2 + I_6 + 2[I_5\sin(\theta_3) + I_{12}\cos(\theta_2) + I_{15} + I_{16}\sin(\theta_3) \tag{6}$$

$$M_{23} = I_5\sin(\theta_3) + I_6 + I_{12}\cos(\theta_3) + I_{16}\sin(\theta_3) + 2I_{15} \tag{7}$$

$$M_{33} = I_{m3} + I_6 + 2I_{15} \tag{8}$$

$$M_{35} = I_{15} + I_{17} \tag{9}$$

$$M_{44} = I_{m4} + I_{14} \tag{10}$$

$$M_{55} = I_{m5} + I_{17} \tag{11}$$

$$M_{66} = I_{m6} + I_{23} \tag{12}$$

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

and Corilios (B) matrix is calculated as the following

$$B(q) = \begin{bmatrix} b_{112} & b_{113} & 0 & b_{115} & 0 & b_{123} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{214} & 0 & 0 & b_{223} & 0 & b_{225} & 0 & 0 & b_{235} & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{314} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ b_{412} & b_{412} & 0 & b_{415} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b_{514} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{14}$$

Where,

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

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

$$b_{115} = 2[-\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) + I_{16}\cos(\theta \\ \theta_3) + I_{22}\cos(\theta_2+\theta_3)\cos(\theta_2+\theta_3)] \tag{17}$$

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

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

$$b_{223} = 2[-I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3)] \tag{20}$$

$$b_{235} = 2[I_{16}\cos(\theta_3) + I_{22}] \tag{21}$$

$$b_{314} = 2[I_{20}\sin(\theta_2 + \theta_3)(1 - 0.5)] + I_{14}\sin(\theta_2 + \theta_3) + I_{19}\sin(\theta_2 + \theta_3) \tag{22}$$

$$b_{412} = b_{214} = -[I_{14}\sin(\theta_2 + \theta_3) + I_{19}\sin(\theta_2 + \theta_3) + 2I_{20}\sin(\theta_2 + \theta_3)(1 - 0.5)] \tag{23}$$

$$b_{413} = -b_{314} = -2[I_{20}\sin(\theta_2 + \theta_3)(1 - 0.5)] + I_{14}\sin(\theta_2 + \theta_3) + I_{19}\sin(\theta_2 + \theta_3 \tag{24}$$

$$b_{415} = -I_{20}\sin(\theta_2 + \theta_3) - I_{17}\sin(\theta_2 + \theta_3) \tag{25}$$

$$b_{514} = -b_{415} = I_{20}\sin(\theta_2 + \theta_3) + I_{17}\sin(\theta_2 + \theta_3) \tag{26}$$

consequently coriolis matrix is shown as bellows;

$$B(q).\dot{q}\dot{q} = \begin{bmatrix} b_{112}.\dot{q}_1\dot{q}_2 + b_{113}.\dot{q}_1\dot{q}_3 + 0 + b_{123}.\dot{q}_2\dot{q}_3 \\ 0 + b_{223}.\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}.\dot{q}_1\dot{q}_2 + b_{413}.\dot{q}_1\dot{q}_3 + 0 \\ 0 \\ 0 \end{bmatrix} \tag{27}$$

Moreover Centrifugal (C) matrix is demonstrated as

$$C(q) = \begin{bmatrix} 0 & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{21} & 0 & C_{23} & 0 & 0 & 0 \\ C_{31} & C_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ C_{51} & C_{52} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{28}$$

Where,

$$c_{12} = I_4\cos(\theta_2) - I_8\sin(\theta_2 + \theta_3) - I_9\sin(\theta_2) + I_{13}\cos(\theta_2 + \theta_3) + I_{18}\sin(\theta_2 + \theta \tag{29}$$

$$c_{13} = 0.5b_{123} = -I_8\sin(\theta_2 + \theta_3) + I_{13}\cos(\theta_2 + \theta_3) + I_{18}\sin(\theta_2 + \theta_3) \tag{30}$$

$$c_{21} = -0.5b_{112} = I_3\sin(\theta_2)\cos(\theta_2) - I_5\cos(\theta_2 + \theta_2 + \theta_3) - I_7\sin(\theta_2 + \theta_3)\cos \\ I_{12}\sin(\theta_2 + \theta_2 + \theta_3) + I_{15}2\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) - I_{16}\cos(\theta_2 + \theta_2 + \theta_3) - I \\ \theta_3)\cos(\theta_2 + \theta_3) - I_{22}(1 - 2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)) - 0.5I_{10}(1 - 2\sin(\theta_2 + \theta \\ \theta_3)) - 0.5I_{11}(1 - 2\sin(\theta_2)\sin(\theta_2)) \tag{31}$$

$$c_{22} = 0.5b_{223} = -I_{12}\sin(\theta_3) + I_5\cos(\theta_3) + I_{16}\cos(\theta_3) \tag{32}$$

$$c_{23} = -0.5b_{113} = -I_5\cos(\theta_2)\cos(\theta_2 + \theta_3) - I_7\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) + I_{12} \\ \theta_2) - I_{15}2\sin(\theta_2 + \theta_3)\cos(\theta_2 + \theta_3) - I_{16}\cos(\theta_2)\cos(\theta_2 + \theta_3) - I_{21}\sin(\theta_2 + \theta_3) \\ I_{22}(1 - 2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)) - 0.5I_{10}(1 - 2\sin(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)) \tag{33}$$

$$c_{31} = -c_{23} = I_{12}\sin(\theta_3) - I_5\cos(\theta_3) - I_{16}\cos(\theta_3) \quad (34)$$

$$c_{32} = -0.5b_{115} = \sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{15}2\sin(\theta_2+\theta_3)\cos(\theta_2+\theta_3) - I_{16}\cos(\theta_2)\cos(\theta_2 + I_{22}\cos(\theta_2+\theta_3)\cos(\theta_2+\theta_3) \quad (35)$$

$$c_{52} = -0.5b_{225} = -I_{16}\cos(\theta_3) - I_{22} \quad (36)$$

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

$$C(q).q^{\cdot 2} = \begin{bmatrix} c_{112}.q_2^2 + c_{13}.q_3^2 \\ c_{21}.q_1^2 + c_{23}.q_3^2 \\ c_{13}.q_1^2 + c_{32}.q_2^2 \\ 0 \\ c_{51}.q_1^2 + c_{52}.q_2^2 \\ 0 \end{bmatrix} \quad (37)$$

Gravity (G) Matrix can be written as

$$G(q) = \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix} \quad (38)$$

Where,

$$G_2 = g_1\cos(\theta_2) + g_2\sin(\theta_2+\theta_3) + g_3\sin(\theta_2) + g_4\cos(\theta_2+\theta_3) + g_5\sin(\theta_2 + \quad (39)$$

$$G_3 = g_2\sin(\theta_2+\theta_3) + g_4\cos(\theta_2+\theta_3) + g_5\sin(\theta_2+\theta_3) \quad (40)$$

$$G_5 = g_5\sin(\theta_2+\theta_3) \quad (41)$$

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

$$\ddot{q} = M^{-1}(q).\{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \quad (42)$$

and K is introduced as

$$K = \{\tau - [B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q)]\} \quad (43)$$

$\ddot{q}$ can be written as

$$\ddot{q} = M^{-1}(q).K \quad (44)$$

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

$$K_1 = \tau_1 - [b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3] - [C_{12}\dot{q}_2^{\ 2} + C_{13}\dot{q}_3^{\ 2}] - g_1 \quad (45)$$

$$K_2 = \tau_2 - [b_{223}\dot{q}_2\dot{q}_3] - [C_{21}\dot{q}_1^{\ 2} + C_{23}\dot{q}_3^{\ 2}] - g_2 \quad (46)$$

$$K_3 = \tau_3 - [C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2] - g_3 \tag{47}$$

$$K_4 = \tau_4 - [b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3] - g_4 \tag{48}$$

$$K_5 = \tau_5 - [C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2] - g_5 \tag{49}$$

$$K_6 = \tau_6 \tag{50}$$

An information about inertial constant and gravitational constant are shown in Tables 1 and 2 based on the studies carried out by Armstrong [80] and Corke and Armstrong [81].

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

TABLE 1: 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 2: Gravitational constant (*N.m*)

3. CONTROL: COMPUTED TORQUE CONTROLLER ANALYSIS, MODELLING AND IMPLEMENTATION ON PUMA 560 ROBOT MANIPULATOR

Computed torque controller (CTC) is a powerful nonlinear controller which it widely used in control of robot manipulator. It is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. This controller works very well when all dynamic and physical parameters are known but when the robot manipulator has variation in dynamic parameters, in this situation the controller has no acceptable performance[14]. In practice, most of physical systems (e.g., robot manipulators) parameters are unknown or time variant, therefore, computed torque like controller used to compensate dynamic equation of robot manipulator[1, 6]. Research on computed torque controller is significantly growing on robot manipulator application which has been reported in [1, 6, 15-16]. Vivas and Mosquera [15]have proposed a predictive functional controller and compare to computed torque controller for tracking response in uncertain environment. However both controllers have been used in feedback linearization, but predictive strategy gives better result as a performance. A computed torque control with non parametric regression models have been presented for a robot arm[16]. This controller also has been problem in uncertain dynamic models. Based on [1, 6]and [15-16] computed torque controller is a significant nonlinear controller to certain systems which it is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. When all dynamic and physical parameters are known, computed torque controller works fantastically; practically a large amount of systems have uncertainties, therefore sliding mode controller is one of the best case to solve this challenge.

The central idea of Computed torque controller (CTC) is feedback linearization so, originally this algorithm is called feedback linearization controller. It has assumed that the desired motion trajectory for the manipulator $q_d(t)$, as determined, by a path planner. Defines the tracking error as:

$$e(t) = q_d(t) - q_a(t) \tag{51}$$

Where e(t) is error of the plant, $q_d(t)$ is desired input variable, that in our system is desired displacement, $q_a(t)$ is actual displacement. If an alternative linear state-space equation in the form $\dot{x} = Ax + BU$ can be defined as

$$\dot{x} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} U \tag{52}$$

With $U = -M^{-1}(q).N(q,\dot{q}) + M^{-1}(q).\tau$ and this is known as the Brunousky canonical form. By equation (51) and (52) the Brunousky canonical form can be written in terms of the state $x = [e^T \; \dot{e}^T]^T$ as [1]:

$$\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} U \tag{53}$$

With

$$U = \ddot{q}_d + M^{-1}(q).\{N(q,\dot{q}) - \tau\} \tag{54}$$

Then compute the required arm torques using inverse of equation (55), is;

$$\tau = M(q)(\ddot{q}_d - U) + N(\dot{q},q) \tag{55}$$

This is a nonlinear feedback control law that guarantees tracking of desired trajectory. Selecting proportional-plus-derivative (PD) feedback for U(t) results in the PD-computed torque controller [6];

$$\tau = M(q)\left(\ddot{q}_d + K_v\dot{e} + K_p e\right) + N(q,\dot{q}) \tag{56}$$

and the resulting linear error dynamics are

$$(\ddot{q}_d + K_v \dot{e} + K_p e) = 0 \tag{57}$$

According to the linear system theory, convergence of the tracking error to zero is guaranteed [6]. Where K_p and K_v are the controller gains. The result schemes is shown in Figure 3, in which two feedback loops, namely, inner loop and outer loop, which an inner loop is a compensate loop and an outer loop is a tracking error loop.

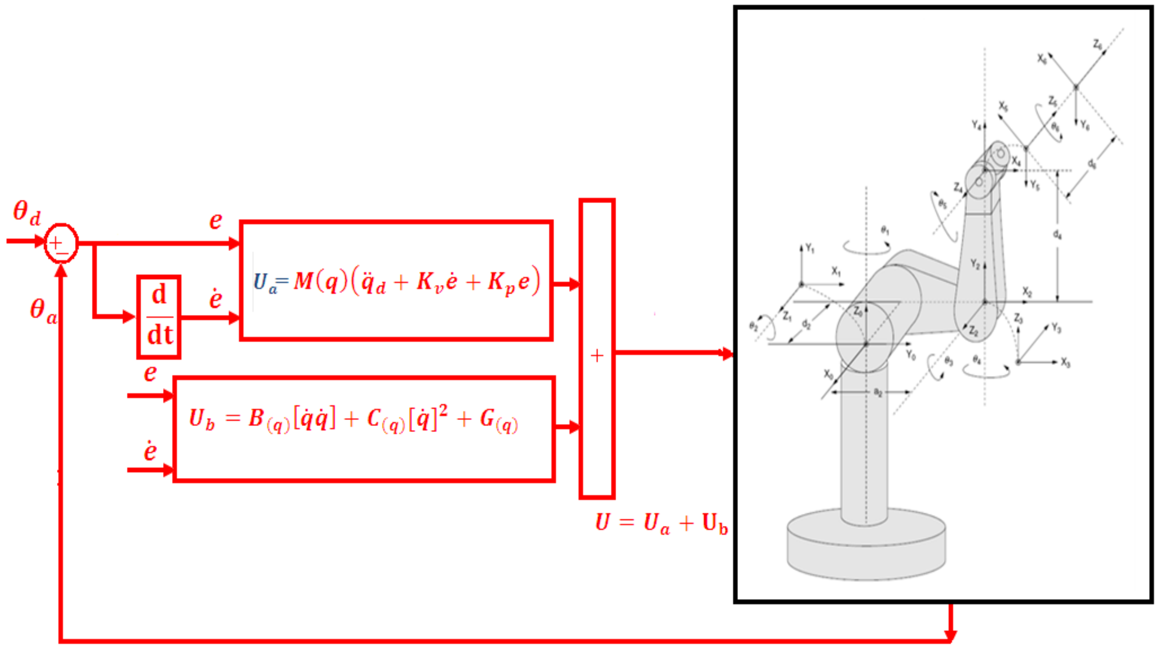


FIGURE 3: Block diagram of PD-computed torque controller (PD-CTC)

The application of proportional-plus-derivative (PD) computed torque controller to control of PUMA 560 robot manipulator introduced in this part. Suppose that in (58) the nonlinearity term defined by the following term

$$N(q,\dot{q}) = B(q)\dot{q}\dot{q} + C(q)\dot{q}^2 + g(q) \tag{58}$$

$$\begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2{}^2 + C_{13}\dot{q}_3{}^2 \\ C_{21}\dot{q}_1{}^2 + C_{23}\dot{q}_3{}^2 \\ C_{31}\dot{q}_1{}^2 + C_{32}\dot{q}_2{}^2 \\ 0 \\ C_{51}\dot{q}_1{}^2 + C_{52}\dot{q}_2{}^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix}$$

Therefore the equation of PD-CTC for control of PUMA 560 robot manipulator is written as the equation of (59);

$$\begin{bmatrix} \hat{\tau}_1 \\ \hat{\tau}_2 \\ \hat{\tau}_3 \\ \hat{\tau}_4 \\ \hat{\tau}_5 \\ \hat{\tau}_6 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & 0 & 0 & 0 \\ M_{21} & M_{22} & M_{23} & 0 & 0 & 0 \\ M_{31} & M_{32} & M_{33} & 0 & M_{35} & 0 \\ 0 & 0 & 0 & M_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & M_{66} \end{bmatrix} \begin{bmatrix} \ddot{q}_{d1} + K_{v1}\dot{e}_1 + K_{p1}e_1 \\ \ddot{q}_{d2} + K_{v2}\dot{e}_2 + K_{p2}e_2 \\ \ddot{q}_{d3} + K_{v3}\dot{e}_3 + K_{p3}e_3 \\ \ddot{q}_{d4} + K_{v4}\dot{e}_4 + K_{p4}e_4 \\ \ddot{q}_{d5} + K_{v5}\dot{e}_5 + K_{p5}e_5 \\ \ddot{q}_{d6} + K_{v6}\dot{e}_6 + K_{p6}e_6 \end{bmatrix} \tag{59}$$

$$+\begin{bmatrix} b_{112}\dot{q}_1\dot{q}_2 + b_{113}\dot{q}_1\dot{q}_3 + 0 + b_{123}\dot{q}_2\dot{q}_3 \\ 0 + b_{223}\dot{q}_2\dot{q}_3 + 0 + 0 \\ 0 \\ b_{412}\dot{q}_1\dot{q}_2 + b_{413}\dot{q}_1\dot{q}_3 + 0 + 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} C_{12}\dot{q}_2^{\,2} + C_{13}\dot{q}_3^{\,2} \\ C_{21}\dot{q}_1^{\,2} + C_{23}\dot{q}_3^{\,2} \\ C_{31}\dot{q}_1^{\,2} + C_{32}\dot{q}_2^{\,2} \\ 0 \\ C_{51}\dot{q}_1^{\,2} + C_{52}\dot{q}_2^{\,2} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \\ 0 \\ g_5 \\ 0 \end{bmatrix}$$

The controller based on a formulation (59) is related to robot dynamics therefore it has problems in uncertain conditions.

Implemented Computed Torque Controller

In first step, constructed dynamics and kinematics blocks (i.e., plant) with power supply will be put in work space. The main object is implementation of controller block. According to PD equation which is $\ddot{q}_d + K_v\dot{e}_1 + K_p e$, the linearized part will be created like Figure 4. The linearized part so called PID. As it is obvious, the parameter e is the difference of actual and desired values and $\dot{e}$ is the change of error. K_p and k_v are proportional and derivative gains and $\ddot{q}_d$ is double derivative of the joint variable. A sample of PD controller block for one joint is like Figure 5.

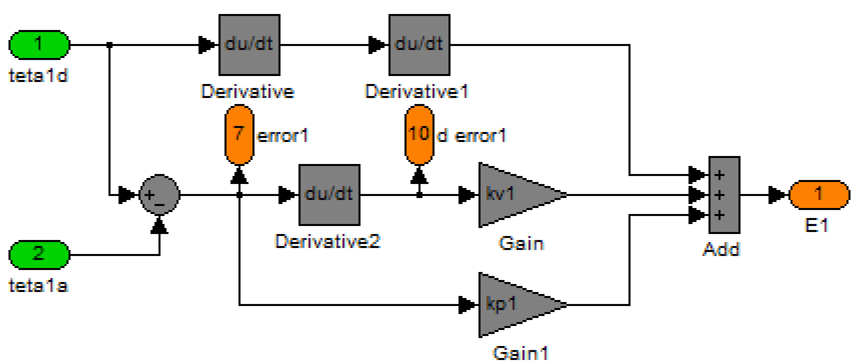


FIGURE 4: PD Controller for a joint variable

As it is seen in Figure 4 the error value and the change of error were chosen to exhibit in measurement center. In this block by changing gain values, the best control system will be applied. In the second step according to torque formulation in CTC mode, all constructed blocks just connect to each other as blew. In Figure 5 the N $(q, \dot{q})$ is the dynamic parameters block (i.e., A set of Coriolis, Centrifugal and Gravity blocks).

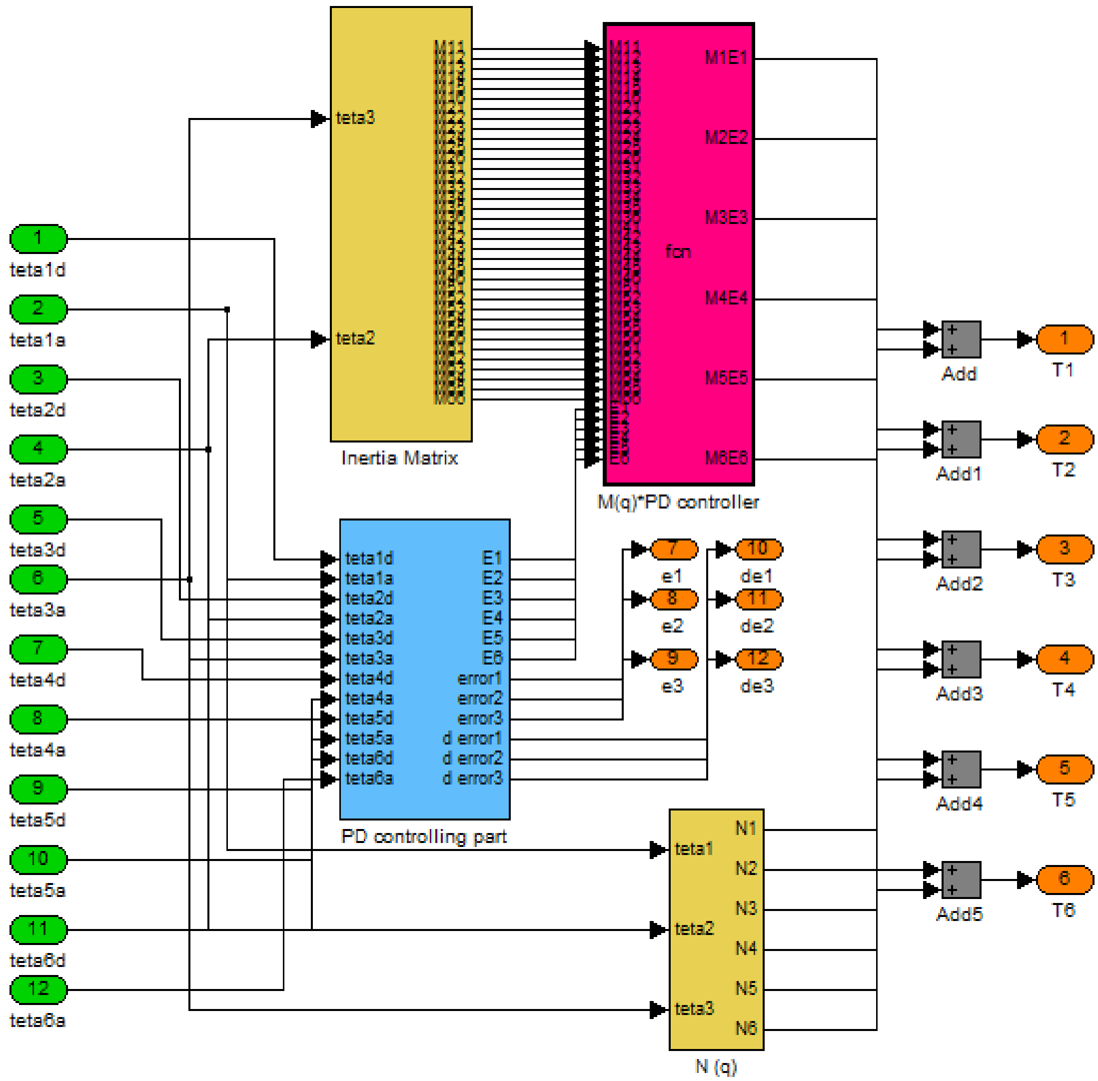


FIGURE 5: The general diagram of controller

The inputs are thetas and the final outputs are torque values. The relations between other blocks are just multiplication and summation as mentioned in torque equation. In the next step transform our subsystems into a general system to form controller block and the outputs will be connected to the plant, in order to execute controlling process. Then, trigger the main inputs with power supply to check validity and performance. In Figure 6, Dynamics, Kinematics and Controller blocks are shown.

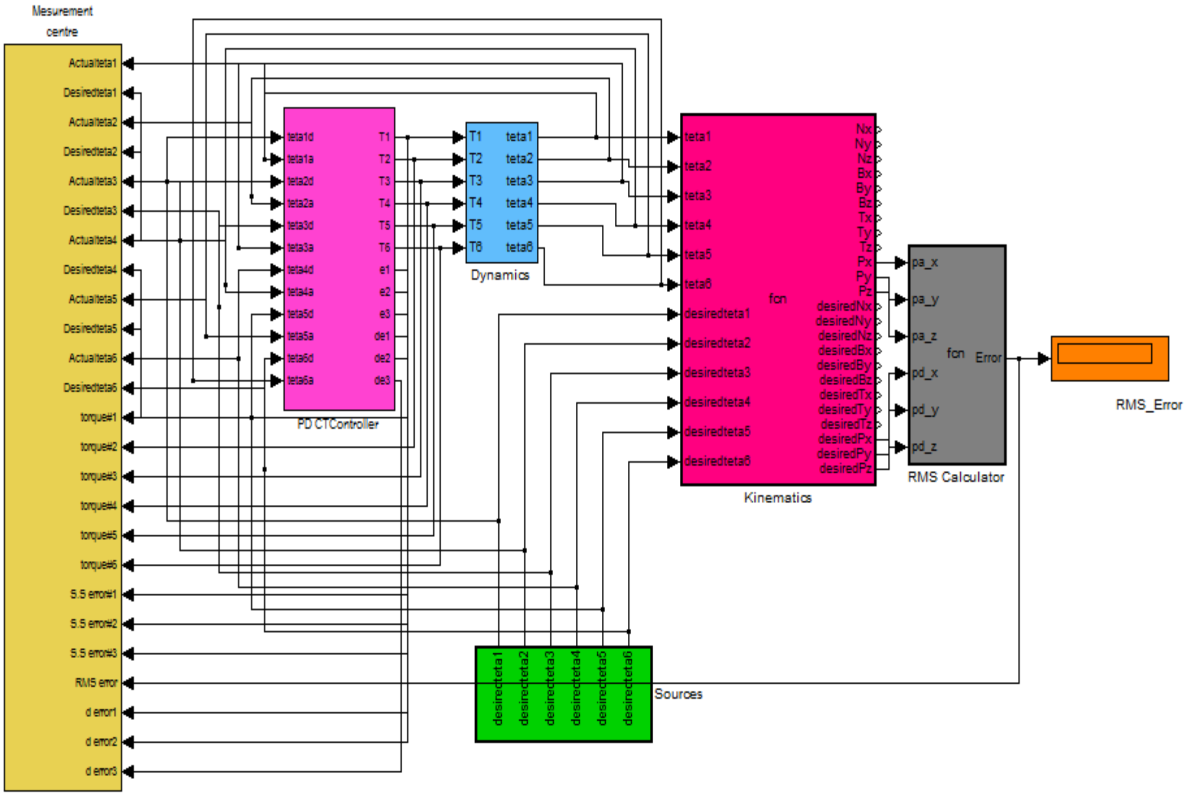


FIGURE 6 : Controller, Dynamics and Kinematics Blocks

4. RESULTS

PD-Computed torque controller (PD-CTC) and PID-Computed torque controller (PID-CTC) were tested to Step and Ramp responses. 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 environment. 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.

Trajectory Performance

Figures 7 and 8 show tracking performance for PD-CTC and PID CTC without disturbance for two type trajectories. The optimal coefficients in Step PID CTC are; $K_p = 70, K_v = 15, and\ K_i = 75$ and in Ramp PID CTC are; $K_p = 50, K_v = 10, and\ K_i = 25$ as well as similarly in Step and Ramp PD CTC are; $K_p = 30\ and\ K_v = 4$; From the simulation for first, second, and third links, it was seen that the different controller gains have the different performance. Tuning parameters of PID-CTC and PD-CTC for PUMA robot manipulator are shown in Tables 3 to 6.

	k_{P_1}	k_{V_1}	k_{I_1}	k_{p_2}	k_{V_2}	k_{I_2}	k_{p_3}	k_{V_3}	k_{I_3}	RMS error	SS error ^{1}	SS error ^{2}	SS error ^{3}
1	70	24	70	70	24	70	70	24	70	2.276e-5	-3.81e-5	-3.81e-5	-3.81e-5
2	50	24	70	50	24	70	50	24	70	3.34e-5	-5.6e-5	-5.6e-5	-5.6e-5
3	70	15	75	70	15	75	70	15	75	0	0	0	0
4	70	24	50	70	24	50	70	24	50	3.7e-5	-6.2e-5	-6.2e-5	-6.2e-5

TABLE 3: Tuning parameters of a step PID-CTC

	k_{P_1}	k_{V_1}	k_{I_1}	k_{p_2}	k_{V_2}	k_{I_2}	k_{p_3}	k_{V_3}	k_{I_3}	RMS error	SS error ^{1}	SS error ^{2}	SS error ^{3}
1	90	10	25	90	10	25	90	10	25	-1.2e-6	-1.6e-6	-1.6e-6	-1.6e-6
2	50	10	25	50	10	25	50	10	25	-4.5e-9	-6e-9	-6e-9	-6e-9
3	90	3	25	90	3	25	90	3	25	-8.8e-7	-1.17e-6	-1.17e-6	-1.17e-6
4	90	10	10	90	10	10	90	10	10	-2.4e-5	-3.2e-5	-3.2e-5	-3.2e-5

TABLE 4: Tuning parameters of a ramp PID-CTC

	k_{P_1}	k_{V_1}	k_{p_2}	k_{V_2}	k_{p_3}	k_{V_3}	RMs error	SS error ^{1}	SS error ^{2}	SS error ^{3}
1	8	4	8	4	8	4	2.276e-5	-3.81e-5	-3.81e-5	-3.81e-5
2	30	4	30	4	30	4	1.34e-5	-3.6e-5	-2.54e-5	-1.6e-5
3	1	4	1	4	1	4	0.0039	0.0065	0.0065	0.0065
4	8	40	8	40	8	40	0.502	5.043	5.043	5.043
5	8	0.5	8	0.5	8	0.5	0.0026	0.0043	0.0043	0.0043

TABLE 5: Tuning parameters of a step PD-CTC

	k_{P_1}	k_{V_1}	k_{p_2}	k_{V_2}	k_{p_3}	k_{V_3}	RMs error	SS error ^{1}	SS error ^{2}	SS error ^{3}
1	8	4	8	4	8	4	3.2e-5	-2.81e-5	-2.6e-5	-2.1e-5
2	30	4	30	4	30	4	-6e-5	-8e-5	-8.6e-5	-8.9e-5
3	1	4	1	4	1	4	0.000305	0.00024	0.00024	0.00024
4	8	40	8	40	8	40	0.6	4.93	4.93	4.93
5	8	0.5	8	0.5	8	0.5	0.5	2.9	2.9	2.9

TABLE 6: Tuning parameters of a ramp PD-CTC

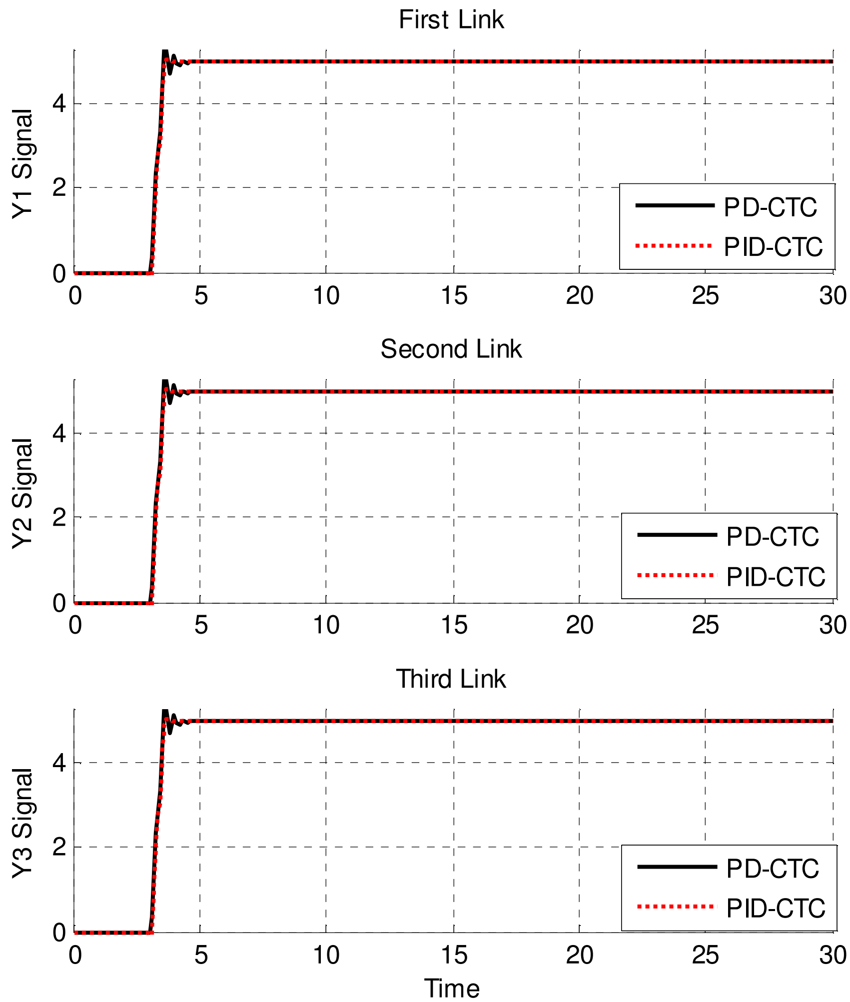


FIGURE 7: Step PID-CTC and PD-CTC for First, second and third link trajectory

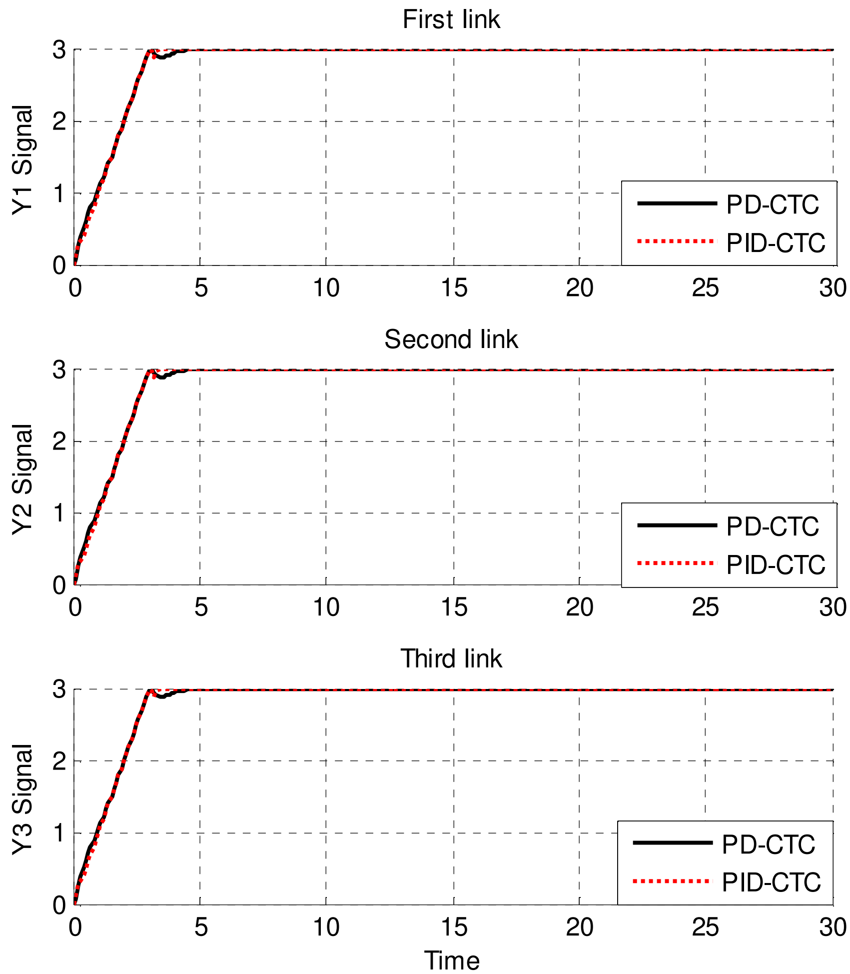


FIGURE 8: Ramp PID-CTC and PD-CTC for First, second and third link trajectory

By comparing step response trajectory without disturbance in PD and PID CTC, it is found that the PID's overshoot **(1.32%)** is lower than PD's **(6.44%)**, although both of them have about the same rise time; PID CTC **(0.5 sec)** and PD CTC **(0.403 sec)**. Besides the Steady State and RMS error in PID **(Steady State error =0 and RMS error=0)** is fairly lower than PD **(Steady State error** $\cong -3^{-5}$ **and RMS error=**-1.6×10^{-5}).

In above graphs over the same period, it is created that the PD and PID CTC are increased slightly, but PD CTC at $t = 3\,sec$ has very low distortion, after $t = 3\,sec$ bought of graphs remain are unchanged as mentioned to the Tables 4 and 6.

Disturbance Rejection: Figures 9 and 10 have shown the power disturbance elimination in PD and PID CTC. The main target in this controller is disturbance rejection as well as the other responses. A band limited white noise with predefined of 40% the power of input signal is applied to the Step and Ramp PD and PID CTC. It found fairly fluctuations in trajectory responses. As mentioned earlier, CTC works very well when all parameters are known, this challenge plays important role to select the SMC as a based robust controller in this research.

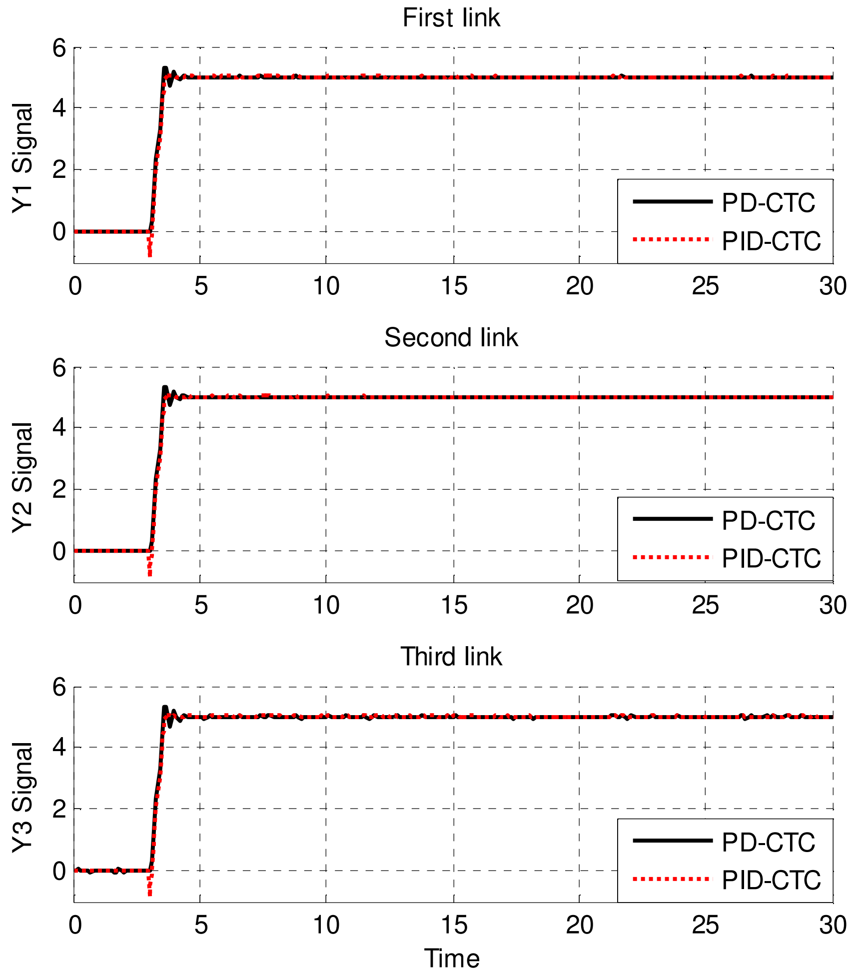


FIGURE 9: Step PID-CTC and PD-CTC for First, second and third link trajectory with disturbance.

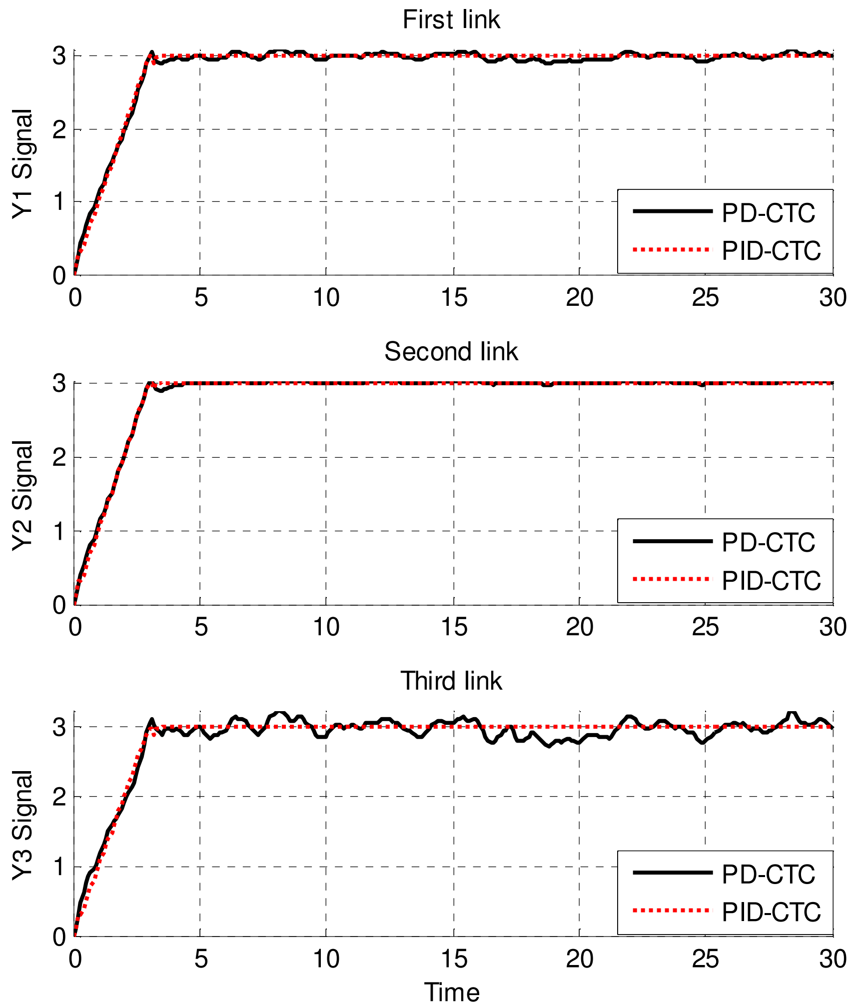


FIGURE 10: Ramp PID-CTC and PD-CTC for First, second and third link trajectory with disturbance.

Among above graphs (9 and 10) relating to Step and Ramp trajectories following with external disturbance, PID CTC and PD CTC have fairly fluctuations. By comparing some control parameters such as overshoot, rise time, steady state and RMS error it computed that the PID's overshoot **(1.8%)** is lower than PD's **(8%)**, although both of them have about the same rise time; PID CTC **(0.5 sec)** and PD CTC **(0.41 sec)**, the Steady State and RMS error in PID **(Steady State error = -0.0019 and RMS error=0.0025)** is fairly lower than PD **(Steady State error $\cong 0.005$ and RMS error=0.0042)**.

Errors in the model: Figures 11 and 12 have shown the error disturbance in PD and PID CTC. The controllers with no external disturbances have the same error response, but PID CTC has the better steady state error when the robot manipulator has external disturbance. Furthermore the RMS error profile for PID CTC is sharply dropped compared to the PD CTC.

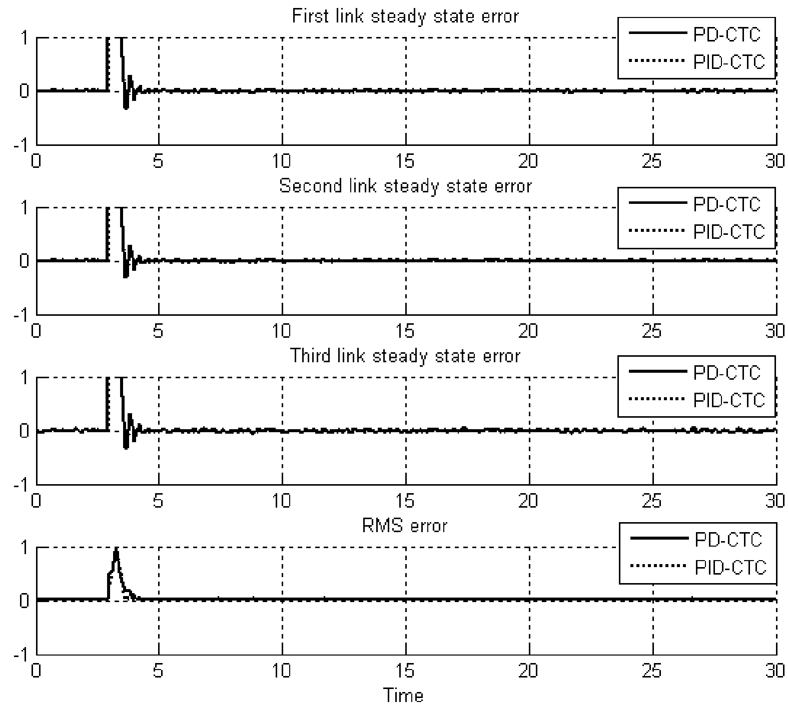


FIGURE 11: Step PID-CTC and PD-CTC for First, second and third link steady state and RMS error with disturbance.

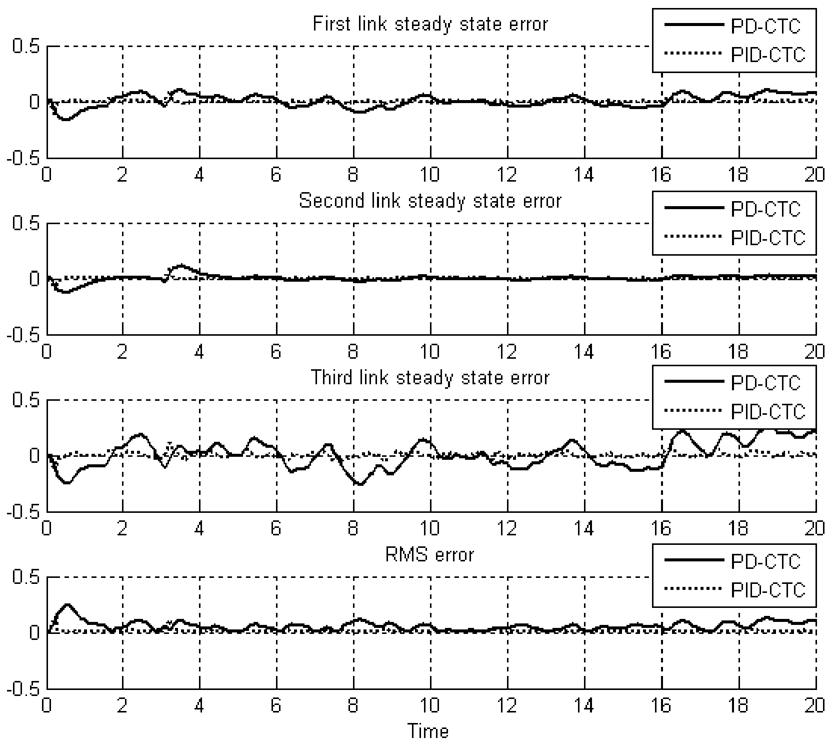


FIGURE 12: Ramp PID-CTC and PD-CTC for First, second and third link steady state and RMS error with disturbance.

The errors in PID CTC and PD CTC is widely increased among of error graphs (11 and 12) relating to Step and Ramp response with external disturbance. By comparing the steady state and RMS error it observed that the PID's State and RMS error **(Steady State error = -0.0019 and RMS error=0.0025)** is lower than PD's **(Steady State error $\cong 0.005$ and RMS error=0.0042)**. When applied disturbance in these controllers it is computed that the steady state and RMS error increased rapidly approximately 130%.

5. CONCLUSION

In this research we introduced, basic concepts of robot manipulator (e.g., PUMA 560 robot manipulator) and control methodology. PUMA 560 robot manipulator is a 6 DOF serial robot manipulator. From the control point of view, robot manipulator divides into two main parts i.e. kinematics and dynamic parts. The dynamic parameters of this system are highly nonlinear. To control of this system nonlinear control methodology (computed torque controller) is introduced. Computed torque controller (CTC) is an influential nonlinear controller to certain and partly uncertain systems which it is based on feedback linearization and computes the required arm torques using the nonlinear feedback control law. When all dynamic and physical parameters are known computed torque controller works superbly.

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] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, *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, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* No. 3, vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, No. 2,* 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," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," *IEEE conference proceeding,* 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE conference proceeding,* 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* No. 1, 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,* No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* No. 2, vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics Journal,* No. 1, 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,* No. 7, 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,* No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," *IEEE conference proceeding,* 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,* No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," *IEEE conference proceeding,* 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, “Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain,” International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] 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.

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, “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.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , “A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator,” World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli “Design Adaptive Fuzzy Robust Controllers for Robot Manipulator,” World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , “Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line Tunable Gain: Applied to Robot Manipulator,” International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, “Design On-Line Tunable Gain Artificial Nonlinear Controller,” Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, “ Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine,” International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, “Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller,” Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, “Novel Sliding Mode Controller for robot manipulator using FPGA,” Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, “Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System,” International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, “Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, “Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System,” International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, “Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method,” International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, “Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control,” International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, “Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ,” International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, “Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm,” International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, “Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ,” International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, “Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator,” International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator,” International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, “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.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, “Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ,” International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, “Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, “On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory,” International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator,” International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, “Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine,” International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.,” Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine,” world applied science journal (WASJ), 15 (3): 422-428, 2011.

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* No. 1, 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,* No. 2, 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,* No. 2, vol. 19, pp. 362-368, 2004.

[62] N. Sulaiman, Z. A. Obaid, M. Marhaban and M. Hamidon , "Design and Implementation of FPGA-Based Systems-A Review," *Australian Journal of Basic and Applied Sciences,* No. 4, vol. 3, pp. 3575-3596, 2009.

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

[64] Y. S. Kung, K. Tseng, C. Chen, H. Sze and A. Wang, "FPGA-implementation of inverse kinematics and servo controller for robot manipulator," *Proc. IEEE Int. on Robotics and Biomimetics,* pp. 1163–1168, 2006.

[65] X. Shao, D. Sun and J. K. Mills, "A new motion control hardware architecture with FPGA-based IC design for robotic manipulators," *IEEE Conference,* 2006, pp. 3520-3525.

[66] Y. S. Kung, C. S. Chen and G. S. Shu, "Design and Implementation of a Servo System for Robotic Manipulator," CACS, 2005.

[67] U. D. Meshram and R. Harkare, "FPGA Based Five Axis Robot Arm Controller," *IEEE Conference,* 2005, pp. 3520-3525.

[68] U. Meshram, P. Bande, P. Dwaramwar and R. Harkare, "Robot arm controller using FPGA," *IEEE Conference,* 2009, pp. 8-11.

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

[70] Z. A. Obaid, N. Sulaiman and M. Hamidon, "Developed Method of FPGA-based Fuzzy Logic Controller Design with the Aid of Conventional PID Algorithm," *Australian Journal of Basic and Applied Sciences,* No. 3, vol. 3, pp. 2724-2740, 2009.

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

[72] K. D. Rogers, "Acceleration and implementation of a DSP phase-based frequency estimation algorithm: MATLAB/SIMULINK to FPGA via XILINX system generator," Citeseer, 2004.

[73] F. J. Lin, D. H. Wang and P. K. Huang, "FPGA-based fuzzy sliding-mode control for a linear induction motor drive," IEEE journal of electrical power application, No. 5, Vol. 152, 2005, pp. 1137-1148.

[74] R. R. Ramos, D. Biel, E. Fossas and F. Guinjoan, "A fixed-frequency quasi-sliding control algorithm: application to power inverters design by means of FPGA implementation," *Power Electronics, IEEE Transactions on,* No. 1, vol. 18, pp. 344-355, 2003.

[75] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," *Carleton university Ottawa,*2007.

[76] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," *IEEE Conference,* 2004, pp. 3588-3592.

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

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

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

[80] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," *IEEE Conference,* 2002, pp. 510-518.

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

[82] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[83] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* 2002, pp. 160-168.

[78] Samira Soltani & Farzin Piltan, "*Design Artificial Nonlinear Controller Based on Computed Torque like Controller with Tunable Gain*". World Applied Science Journal,14 (9): 1306-1312, 2011.

[71] Farzin Piltan, H. Rezaie, B. Boroomand, Arman Jahed," Design robust back stepping online tuning feedback linearization control applied to IC engine," International Journal of Advance Science and Technology, 42: 183-204, 2012.

[72] Farzin Piltan, I. Nazari, S. Siamak, P. Ferdosali ,"Methodology of FPGA-based mathematical error-based tuning sliding mode controller" International Journal of Control and Automation, 5(1): 89-110, 2012.

[73] Farzin Piltan, M. A. Dialame, A. Zare, A. Badri ,"Design Novel Lookup table changed Auto Tuning FSMC: Applied to Robot Manipulator" International Journal of Engineering, 6(1): 25-40, 2012.

[74] Farzin Piltan, B. Boroomand, A. Jahed, H. Rezaie ,"Methodology of Mathematical Error-Based Tuning Sliding Mode Controller" International Journal of Engineering, 6(2): 96-112, 2012.

[75] Farzin Piltan, F. Aghayari, M. R. Rashidian, M. Shamsodini, "A New Estimate Sliding Mode Fuzzy Controller for Robotic Manipulator" International Journal of Robotics and Automation, 3(1): 45-58, 2012.

[76] Farzin Piltan, M. Keshavarz, A. Badri, A. Zargari , "Design novel nonlinear controller applied to robot manipulator: design new feedback linearization fuzzy controller with minimum rule base tuning method" International Journal of Robotics and Automation, 3(1): 1-18, 2012.

[77] 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.

[78] 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.

[79] Farzin Piltan, A. Hosainpour, E. Mazlomian, M.Shamsodini, M.H Yarmahmoudi. "Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach" International Journal of Robotics and Automation, 3(3): 2012.

[80] Farzin Piltan, R. Bayat, F. Aghayari, B. Boroomand. "Design Error-Based Linear Model-Free Evaluation Performance Computed Torque Controller" International Journal of Robotics and Automation, 3(3): 2012.

[81] Farzin Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari & Mina Mirazaei . " PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[82] Farzin Piltan, J. Meigolinedjad, S. Mehrara, S. Rahmdel. " Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology" International Journal of Robotics and Automation, 3(3): 2012.

[83] Farzin Piltan, S. Rahmdel, S. Mehrara, R. Bayat." Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses" International Journal of Engineering, 3(3): 2012.

[84] Farzin Piltan, M. Mirzaie, F. Shahriyari, Iman Nazari & S. Emamzadeh." Design Baseline Computed Torque Controller" International Journal of Engineering, 3(3): 2012.

Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology

Farzin Piltan 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

Javad Meigolinedjad *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

Saleh Mehrara *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

Sajad Rahmdel *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 this research, a baseline error-based tuning sliding mode controller (LTSMC) is proposed for robot manipulator. Sliding mode controller (SMC) is an important nonlinear controller in a partly uncertain dynamic system's parameters. Sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and adaption law which this method can helps improve the system's tracking performance by online tuning method. Since the sliding surface gain (λ) is adjusted by baseline tuning method, it is continuous. In this research new λ is obtained by the previous λ multiple sliding surface slopes updating factor (δ). Baseline error-based tuning sliding mode controller is stable model-based controller which eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in baseline error-based tuning sliding mode controller based on switching (sign) function. This controller has acceptable performance in presence of uncertainty (e.g., overshoot=0%, rise time=0.4 second, steady state error = 1.8e-10 and RMS error=1.16e-12).

Keywords: Baseline Error-based Tuning Sliding Mode Controller, Sliding Mode Controller, Unstructured Model Uncertainties, Adaptive Method, Sliding Surface Gain, Sliding Surface Slopes Updating Factor, Chattering Phenomenon.

1. INTRODUCTION, BACKGROUND and MOTIVATION

Introduction

Controller is a device which can sense information from linear or nonlinear system (e.g., robot manipulator) to improve the systems performance [1-4]. The main targets in designing control systems are stability, good disturbance rejection, and small tracking error[5-6]. Several industrial robot manipulators are controlled by linear methodologies (e.g., Proportional-Derivative (PD) controller, Proportional- Integral (PI) controller or Proportional- Integral-Derivative (PID) controller), but when robot manipulator works with various payloads and have uncertainty in dynamic models this technique has limitations. From the control point of view, uncertainty is divided into two main groups: uncertainty in unstructured inputs (e.g., noise, disturbance) and uncertainty in structure dynamics (e.g., payload, parameter variations). In some applications robot manipulators are used in an 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). Sliding mode controller is a powerful nonlinear robust controller under condition of partly uncertain dynamic parameters of system [7]. This controller is used to control of highly nonlinear systems especially for robot manipulators. Chattering phenomenon in uncertain dynamic parameter is the main drawback in pure sliding mode controller [8-20]. The chattering phenomenon problem in pure sliding mode controller is reduced by using linear saturation boundary layer function but prove the stability is very difficult. 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 [41-70]. Fuzzy adaptive method is used in systems which want to training parameters by expert knowledge [21-38]. 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 [39-40, 71-77].

Robot manipulator is a 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) [15-25]. Parallel robot manipulators have many legs with some links and, where in these robot manipulators base frame has connected to the final frame. Most of industrial robots are serial links, which in n degrees of freedom serial link 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 and the axis number seven to n use to reach the avoid the difficult conditions (e.g., surgical robot and space robot manipulator). 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 such as trajectory planning, essential prerequisite for some dynamic description such as Newton's equation for motion of point mass, 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[1-6]. Dynamic modeling of robot manipulators is used to describe the behavior of robot manipulator such as linear or nonlinear dynamic behavior, design of model based controller such as pure sliding mode controller and pure computed torque controller which design these controller are based on nonlinear dynamic equations, and for simulation. The dynamic modeling describes the relationship between joint motion, velocity, and accelerations to force/torque or current/voltage and also it can be used to describe the particular dynamic effects (e.g., inertia, coriolios, centrifugal, and the other parameters) to behavior of system[1-15].

Background

Chattering phenomenon can causes some problems such as saturation and heats the mechanical parts of robot arm 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. Slotine and Sastry have introduced boundary layer method instead of discontinuous method to reduce the chattering[21]. Slotine has presented sliding mode controller with boundary layer to improve the industry application [22]. 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, Weng and Yu improved the previous method by using a new method in fuzzy nonlinear approximation inside the boundary layer and adaptive method[24]. In various dynamic parameters systems that need to be training on-line, adaptive control methodology is used. Mathematical model free adaptive method is used in systems which want to training parameters by performance knowledge. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, adaptive method is applied to sliding mode controller. Mohan and Bhanot [40] have addressed comparative study between some adaptive fuzzy, and a new hybrid fuzzy control algorithm for robot arm control. They found that self-organizing fuzzy logic controller and proposed hybrid integrator fuzzy give the best performance as well as simple structure. 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 sliding mode controller; it is depended on nonlinear dynamic equqation. Hwang *et al.* [47]have proposed a Tagaki-Sugeno (TS) fuzzy model based sliding mode controller 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 sliding mode 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 sliding surface as inputs of fuzzy systems. Guo and Woo [60]have proposed a SISO fuzzy system compensate and reduce the chattering. Lin and Hsu [61] can tune both systems by fuzzy rules. Eksin et. al [70] have designed mathematical model-free sliding surface slope in fuzzy sliding mode controller.

This paper is organized as follows. In section 2, main subject of sliding mode controller, proof of stability and dynamic formulation of robot manipulator are presented. This section covered the following details, classical sliding mode control, classical sliding for robotic manipulators, proof of stability in pure sliding mode controller, chatter free sliding controller and nonlinear dynamic formulation of system. A methodology of proposed method is presented in section 3, which covered the baseline tuning error-based tuning sliding mode controller and proofs the stability in this method and applied to robot manipulator. In section 4, the sliding mode controller and proposed methodology are compared and discussed. In section 5, the conclusion is presented.

2. THEOREM: DYNAMIC FORMULATION OF ROBOTIC MANIPULATOR, SLIDING MODE FORMULATION APPLIED TO ROBOT ARM, PROOF OF STABILITY

Dynamic of robot arm: The equation of an *n-DOF* robot manipulator governed by the following equation [1, 4, 15-29, 63-70]:

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

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 [1-29]:

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

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 variableq_i, independently of the motion of the other joints. Therefore, the angular acceleration is found as to be [3, 41-62]:

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

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

Sliding Mode methodology: Consider a nonlinear single input dynamic system is defined by [6]:

$$\boldsymbol{x^{(n)} = f(\vec{x}) + b(\vec{x})u} \quad \textbf{(4)}$$

Where u is the vector of control input, $\boldsymbol{x^{(n)}}$ is the $\boldsymbol{n^{th}}$ derivation of $\boldsymbol{x}$, $\boldsymbol{x = [x, \dot{x}, \ddot{x}, \dots, x^{(n-1)}]^T}$ is the state vector, $\boldsymbol{f(x)}$ is unknown or uncertainty, and $\boldsymbol{b(x)}$ is of known *sign* function. The main goal to design this controller is train to the desired state; $\boldsymbol{x_d = [x_d, \dot{x}_d, \ddot{x}_d, \dots, x_d{}^{(n-1)}]^T}$, and trucking error vector is defined by [6]:

$$\boldsymbol{\widetilde{x} = x - x_d = [\widetilde{x}, \dots, \widetilde{x}^{(n-1)}]^T} \quad \textbf{(5)}$$

A time-varying sliding surface $\boldsymbol{s(x,t)}$ in the state space $\boldsymbol{R^n}$ is given by [6]:

$$\boldsymbol{s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\,\widetilde{x} = 0} \quad \textbf{(6)}$$

where λ is the positive constant. To further penalize tracking error, integral part can be used in sliding surface part as follows [6]:

$$\boldsymbol{s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\left(\int_0^t \widetilde{x}\,dt\right) = 0} \quad \textbf{(7)}$$

The main target in this methodology is kept the sliding surface slope $\boldsymbol{s(x,t)}$ near to the zero. Therefore, one of the common strategies is to find input $\boldsymbol{U}$ outside of $\boldsymbol{s(x,t)}$ [6].

$$\boldsymbol{\frac{1}{2}\frac{d}{dt}s^2(x,t) \le -\zeta|s(x,t)|} \quad \textbf{(8)}$$

where ζ is positive constant.

$$\textbf{\textit{If S(0)>0}}\rightarrow \frac{d}{dt}\boldsymbol{S(t)} \le -\zeta \quad \textbf{(9)}$$

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

$$\boldsymbol{\int_{t=0}^{t=t_{reach}} \frac{d}{dt}S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0)} \quad \textbf{(10)}$$

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

$$\boldsymbol{0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta}} \quad \textbf{(11)}$$

and

$$\boldsymbol{if\, S(0)} < 0 \rightarrow \boldsymbol{0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta}} \quad \textbf{(12)}$$

Equation (12) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$\boldsymbol{if\, S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0} \quad \textbf{(13)}$$

suppose S is defined as

$$\boldsymbol{s(x,t) = (\frac{d}{dt} + \lambda)\quad \widetilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d)} \quad \textbf{(14)}$$

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

$$\dot{\mathbf{S}} = (\ddot{\mathbf{x}} - \ddot{\mathbf{x}}_{\mathbf{d}}) + \boldsymbol{\lambda}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_{\mathbf{d}}) \tag{15}$$

suppose the second order system is defined as;

$$\ddot{\mathbf{x}} = \mathbf{f} + \mathbf{u} \ \rightarrow \dot{\mathbf{S}} = \mathbf{f} + \mathbf{U} - \ddot{\mathbf{x}}_{\mathbf{d}} + \boldsymbol{\lambda}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_{\mathbf{d}}) \tag{16}$$

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

$$\widehat{\boldsymbol{U}} = -\widehat{\boldsymbol{f}} + \ddot{\boldsymbol{x}}_{\boldsymbol{d}} - \boldsymbol{\lambda}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_{\mathbf{d}}) \tag{17}$$

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

$$\boldsymbol{U}_{\boldsymbol{dis}} = \widehat{\boldsymbol{U}} - \boldsymbol{K}(\vec{\boldsymbol{x}}, \boldsymbol{t}) \cdot \mathbf{sgn}(\boldsymbol{s}) \tag{18}$$

where the switching function $\mathbf{sgn}(\boldsymbol{S})$ is defined as [1, 6]

$$\boldsymbol{sgn}(\boldsymbol{s}) = \begin{cases} \mathbf{1} & \boldsymbol{s} > 0 \\ -\mathbf{1} & \boldsymbol{s} < 0 \\ \mathbf{0} & \boldsymbol{s} = \mathbf{0} \end{cases} \tag{19}$$

and the $\boldsymbol{K}(\vec{\boldsymbol{x}}, \boldsymbol{t})$ is the positive constant. Suppose by (8) the following equation can be written as,

$$\frac{\mathbf{1}}{\mathbf{2}}\frac{\boldsymbol{d}}{\boldsymbol{dt}}\boldsymbol{s}^2(\boldsymbol{x}, \boldsymbol{t}) = \dot{\mathbf{S}} \cdot \mathbf{S} = \left[\boldsymbol{f} - \widehat{\boldsymbol{f}} - \boldsymbol{K}\mathbf{sgn}(\boldsymbol{s})\right] \cdot \boldsymbol{S} = \left(\boldsymbol{f} - \widehat{\boldsymbol{f}}\right) \cdot \boldsymbol{S} - \boldsymbol{K}|\boldsymbol{S}| \tag{20}$$

and if the equation (12) instead of (11) the sliding surface can be calculated as

$$\boldsymbol{s}(\boldsymbol{x}, \boldsymbol{t}) = (\frac{\boldsymbol{d}}{\boldsymbol{dt}} + \boldsymbol{\lambda})^2 \left(\int_{\mathbf{0}}^{\boldsymbol{t}} \widetilde{\boldsymbol{x}}\, \boldsymbol{dt}\right) = (\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_{\boldsymbol{d}}) + \mathbf{2}\boldsymbol{\lambda}(\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_{\boldsymbol{d}}) - \boldsymbol{\lambda}^2(\boldsymbol{x} - \boldsymbol{x}_{\boldsymbol{d}}) \tag{21}$$

in this method the approximation of $\boldsymbol{U}$ is computed as [6]

$$\widehat{\boldsymbol{U}} = -\widehat{\boldsymbol{f}} + \ddot{\boldsymbol{x}}_{\boldsymbol{d}} - \mathbf{2}\boldsymbol{\lambda}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_{\mathbf{d}}) + \boldsymbol{\lambda}^2(\mathbf{x} - \mathbf{x}_{\mathbf{d}}) \tag{22}$$

Based on above discussion, the sliding mode control law for a multi degrees of freedom robot manipulator is written as [1, 6]:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\boldsymbol{eq}} + \boldsymbol{\tau}_{\boldsymbol{dis}} \tag{23}$$

Where, the model-based component $\boldsymbol{\tau}_{\boldsymbol{eq}}$ is the nominal dynamics of systems and $\boldsymbol{\tau}_{\boldsymbol{eq}}$ for first 3 DOF PUMA robot manipulator can be calculate as follows [1]:

$$\boldsymbol{\tau}_{\boldsymbol{eq}} = \left[\boldsymbol{M}^{-\mathbf{1}}(\boldsymbol{B} + \boldsymbol{C} + \boldsymbol{G}) + \dot{\boldsymbol{S}}\right]\boldsymbol{M} \tag{24}$$

and $\boldsymbol{\tau}_{\boldsymbol{dis}}$ is computed as [1];

$$\boldsymbol{\tau}_{\boldsymbol{dis}} = \boldsymbol{K} \cdot \boldsymbol{sgn}(\boldsymbol{S}) \tag{25}$$

by replace the formulation (25) in (23) the control output can be written as;

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\boldsymbol{eq}} + \boldsymbol{K}.\boldsymbol{sgn}(\boldsymbol{S}) \tag{26}$$

By (26) and (24) the sliding mode control of PUMA 560 robot manipulator is calculated as;

$$\boldsymbol{\tau} = \left[\boldsymbol{M}^{-\mathbf{1}}(\boldsymbol{B} + \boldsymbol{C} + \boldsymbol{G}) + \dot{\boldsymbol{S}}\right]\boldsymbol{M} + \boldsymbol{K} \cdot \boldsymbol{sgn}(\boldsymbol{S}) \tag{27}$$

where $S = \lambda e + \dot{e}$ in PD-SMC and $S = \lambda e + \dot{e} + (\frac{\lambda}{2})^2 \sum e$ in PID-SMC.

Proof of Stability: the lyapunov formulation can be written as follows,

$$\boldsymbol{V} = \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{S}^T.\boldsymbol{M}.\boldsymbol{S} \tag{28}$$

the derivation of V can be determined as,

$$\dot{\boldsymbol{V}} = \frac{1}{2}\boldsymbol{S}^T.\dot{\boldsymbol{M}}.\boldsymbol{S} + \boldsymbol{S}^T\, \boldsymbol{M}\dot{\boldsymbol{S}} \tag{29}$$

the dynamic equation of IC engine can be written based on the sliding surface as

$$M\dot{S} = -VS + M\dot{S} + B + C + G \tag{30}$$

it is assumed that

$$S^T(\dot{M} - 2B + C + G)S = 0 \tag{31}$$

by substituting (30) in (29)

$$\dot{V} = \frac{1}{2}S^T\dot{M}S - S^TB + CS + S^T(M\dot{S} + B + CS + G) = S^T(M\dot{S} + B + CS + G) \tag{32}$$

suppose the control input is written as follows

$$\hat{U} = \widehat{U_{Nonlinear}} + \widehat{U_{dis}} = [\widehat{M^{-1}}(B + C + G) + \dot{S}]\hat{M} + K.sgn(S) + B + CS + G \tag{33}$$

by replacing the equation (33) in (32)

$$\dot{V} = S^T(M\dot{S} + B + C + G - \hat{M}\dot{S} - \widehat{B + CS} + G - Ksgn(S) = S^T\left(\tilde{M}\dot{S} + \widetilde{B + CS} + G - Ksgn(S)\right) \tag{34}$$

it is obvious that

$$|\tilde{M}\dot{S} + \widetilde{B + CS} + G| \le |\tilde{M}\dot{S}| + |\widetilde{B + CS} + G| \tag{35}$$

the Lemma equation in robot arm system can be written as follows

$$K_u = [|\tilde{M}\dot{S}| + |B + CS + G| + \eta]_i , i = 1,2,3,4, \dots \tag{36}$$

the equation (11) can be written as

$$K_u \ge \left|[\tilde{M}\dot{S} + B + CS + G]_i\right| + \eta_i \tag{37}$$

therefore, it can be shown that

$$\dot{V} \le -\sum_{i=1}^{n} \eta_i |S_i| \tag{38}$$

Consequently the equation (38) guaranties the stability of the Lyapunov equation. Figure 1 is shown pure sliding mode controller, applied to robot arm.

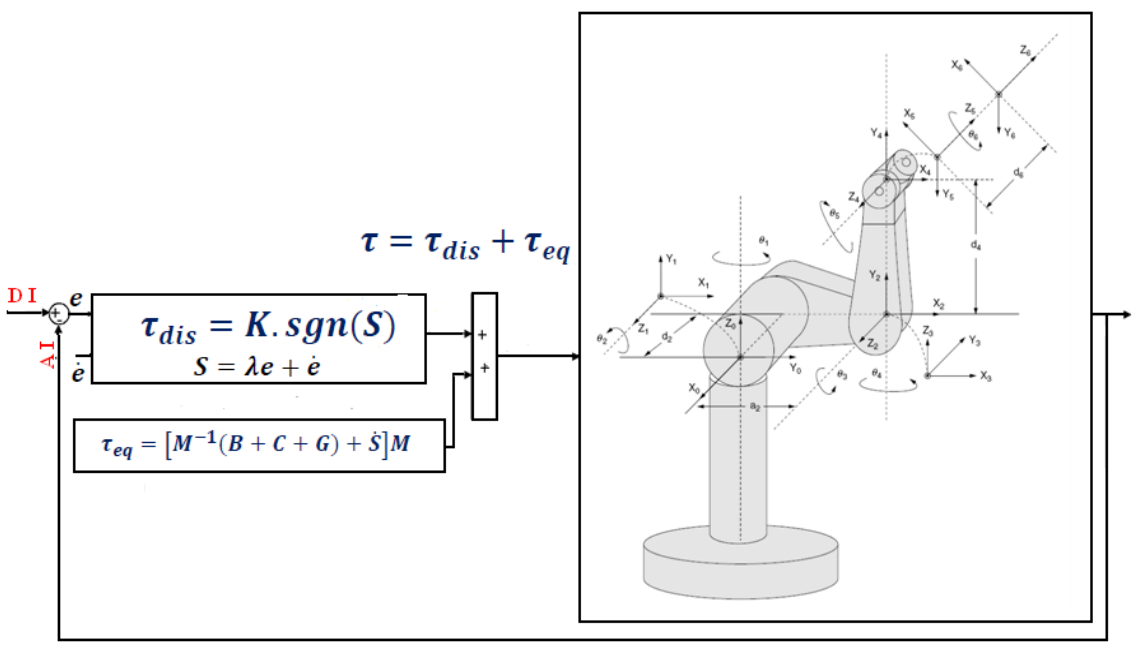


FIGURE 1: Block diagram of a sliding mode controller: applied to robot arm

3. METHODOLOGY: ROBUST BASELINE ON-LINE TUNING FOR STABLE SLIDING MODE CONTROLLER

Sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and baseline error-based tuning method which this method can helps to eliminate the chattering in presence of switching function method and improves the system's tracking performance by online tuning method. In this research the nonlinear equivalent dynamic (equivalent part) formulation problem in uncertain system is solved by using on-line linear error-based tuning theorem. In this method linear error-based theorem is applied to sliding mode controller to adjust the sliding surface slope. Sliding mode controller has difficulty in handling unstructured model uncertainties and this controller's performance is sensitive to sliding surface slope coefficient. It is possible to solve above challenge by combining linear error-based tuning method and sliding mode controller which this methodology can help to improve system's tracking performance by on-line tuning (baseline performance based tuning) method. Based on above discussion, compute the best value of sliding surface slope coefficient has played important role to improve system's tracking performance especially when the system parameters are unknown or uncertain. This problem is solved by tuning the surface slope coefficient ($\boldsymbol{\lambda}$) of the sliding mode controller continuously in real-time. In this methodology, the system's performance is improved with respect to the pure sliding mode controller. Figure 2 shows the baseline error-based tuning sliding mode controller. Based on (23) and (27) to adjust the sliding surface slope coefficient we define $\hat{f}(x|\lambda)$ as the fuzzy based tuning.

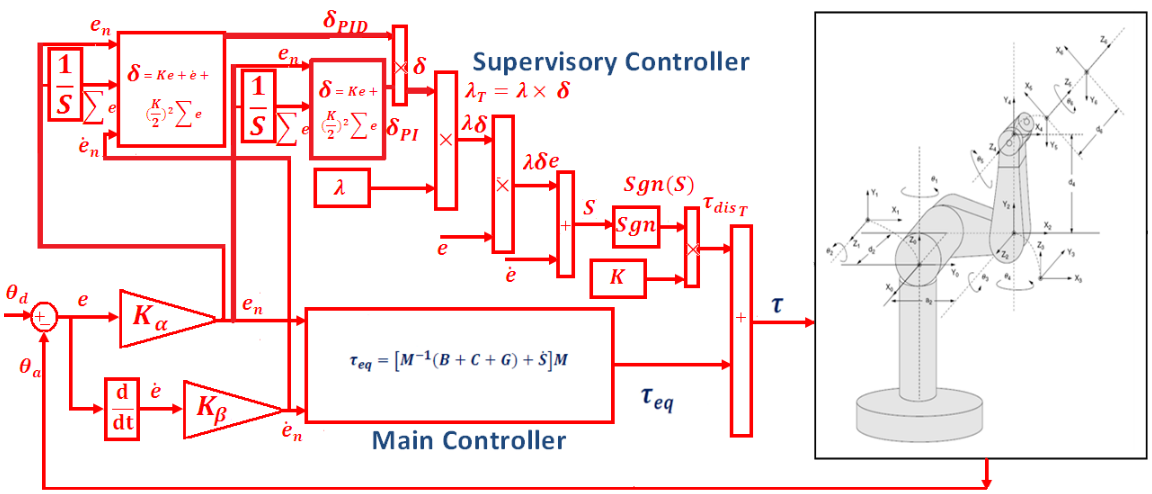


FIGURE 2: Block diagram of a linear error-based sliding mode controller: applied to robot arm

$$\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) = \boldsymbol{\lambda}^{\boldsymbol{T}}\boldsymbol{\delta} \tag{39}$$

If minimum error ($\boldsymbol{\lambda}^*$) is defined by;

$$\boldsymbol{\lambda}^* = \boldsymbol{arg\ min}\ [\left(\boldsymbol{Sup}\left|\hat{\boldsymbol{f}}(\boldsymbol{x}|\boldsymbol{\lambda}) - \boldsymbol{f}(\boldsymbol{x})\right)\right] \tag{40}$$

Where λ^T is adjusted by an adaption law and this law is designed to minimize the error's parameters of $\boldsymbol{\lambda} - \boldsymbol{\lambda}^*$. adaption law in linear error-based tuning sliding mode controller is used to adjust the sliding surface slope coefficient. Linear error-based tuning part is a supervisory controller based on the following formulation methodology. This controller has three inputs namely; error (e), change of error $(\dot{e})$ and the integral of error $(\sum e)$ and an output namely; gain updating factor(δ). As a summary design a linear error-based tuning is based on the following formulation:

$$\boldsymbol{\delta} = (\boldsymbol{K}.\boldsymbol{e} + \dot{\boldsymbol{e}} + \frac{(\boldsymbol{K})^2}{2}\sum \boldsymbol{e}) \times (\boldsymbol{K}.\boldsymbol{e} + \frac{(\boldsymbol{K})^2}{2}\sum \boldsymbol{e}\ ) \tag{41}$$

$$S_{on-line} = \delta.\lambda e + \dot{e} \Rightarrow S_{on-line} = \{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}\lambda e + \dot{e}$$

$$\lambda_{Tune} = \lambda.\delta \Rightarrow \lambda_{Tune} = \lambda\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}$$

Where (δ) is gain updating factor, $(\sum e)$ is the integral of error, $(\dot{e})$ is change of error, (e) is error and K is a coefficient.

Proof of Stability: The Lyapunov function in this design is defined as

$$V = \frac{1}{2}S^T MS + \frac{1}{2}\sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\phi_j \tag{42}$$

where γ_{sj} is a positive coefficient, $\phi = \lambda^* - \lambda$, θ^* is minimum error and λ is adjustable parameter. Since $\dot{M} - 2V$ is skew-symetric matrix;

$$S^T M\dot{S} + \frac{1}{2}S^T \dot{M} S = S^T(M\dot{S} + VS) \tag{43}$$

If the dynamic formulation of robot manipulator defined by

$$\tau = M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) \tag{44}$$

the controller formulation is defined by

$$\tau = \hat{M}\ddot{q}_r + \hat{V}\dot{q}_r + \hat{G} - \lambda S - K \tag{45}$$

According to (43) and (44)

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) = \hat{M}\ddot{q}_r + \hat{V}\dot{q}_r + \hat{G} - \lambda S - K \tag{46}$$

Since $\dot{q}_r = \dot{q} - S$ and $\ddot{q}_r = \ddot{q} - \dot{S}$

$$M\dot{S} + (V + \lambda)S = \Delta f - K \tag{47}$$

$$M\dot{S} = \Delta f - K - VS - \lambda S$$

The derivation of V is defined

$$\dot{V} = S^T M\dot{S} + \frac{1}{2}S^T \dot{M} S + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j \tag{48}$$

$$\dot{V} = S^T(M\dot{S} + VS) + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j$$

Based on (46) and (47)

$$\dot{V} = S^T(\Delta f - K - VS - \lambda S + VS) + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j \tag{49}$$

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

$$\dot{V} = \sum_{J=1}^{M}[S_j(\Delta f_j - K_j)] - S^T\lambda S + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j$$

suppose α is defined as follows

$$\delta_j = \{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\} \tag{50}$$

according to 48 and 49;

$$\dot{V} = \sum_{J=1}^{M}\left[S_j(\Delta f_j - \lambda^T K.e + \dot{e} + \frac{(K)^2}{2}\sum e]\right] - S^T\lambda S + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j \quad (51)$$

Based on $\phi = \theta^* - \theta \to \theta = \theta^* - \phi$

$$\dot{V} = \sum_{J=1}^{M}\left[S_j(\Delta f_j - \theta^{*T}[\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}] + \phi^T[\delta\right. \quad (52)$$
$$\left. = \{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}]\right] - S^T\lambda S$$
$$+ \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi^T.\dot{\phi}_j$$

$$\dot{V} = \sum_{J=1}^{M}\left[S_j(\Delta f_j - (\lambda^*)^T\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}]\right] - S^T\lambda S$$
$$+ \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi_j^T[\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\} + \dot{\phi}_j])$$

where $\dot{\theta}_j = \{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}$ is adaption law, $\dot{\phi}_j = -\dot{\theta}_j = [\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}$ $\dot{V}$ is considered by

$$\dot{V} = \sum_{j=1}^{m}[S_j\,\Delta f_j - \left((\lambda_j^*)^T\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}\right)] - S^T\lambda S \quad (53)$$

The minimum error is defined by

$$e_{mj} = \Delta f_j - \left((\lambda_j^*)^T\{\left(K.e + \dot{e} + \frac{(K)^2}{2}\sum e\right) \times \left(K.e + \frac{(K)^2}{2}\sum e\right)\}\right) \quad (54)$$

Therefore $\dot{V}$ is computed as

$$\dot{V} = \sum_{j=1}^{m}[S_j\,e_{mj}] - S^T\lambda S \quad (55)$$
$$\leq \sum_{j=1}^{m}|S_j||e_{mj}| - S^T\lambda S$$
$$= \sum_{j=1}^{m}|S_j||e_{mj}| - \lambda_j {S_j}^2$$

$$= \sum_{j=1}^{m}|S_j|(|e_{mj}| - \lambda_j S_j) \quad (56)$$

4. RESULTS

This part is focused on compare between Sliding Mode Controller (SMC) and baseline error-based tuning Sliding Mode Controller (LTSMC). These controllers were tested by step responses. In this simulation, to control position of PUMA robot manipulator 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. 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 and applied to nonlinear dynamic of these controllers.

Tracking performances: In sliding mode controller; controllers performance are depended on the gain updating factor (K) and sliding surface slope coefficient (λ). These two coefficients are computed by trial and error in SMC. The best possible coefficients in step SMC are; $\lambda_1 = 1, \lambda_2 = 6, \lambda_3 = 8$; $K_p = K_v = K_i = 10$; $\emptyset_1 = \emptyset_2 = \emptyset_3 = 0.1$. In linear error-based tuning sliding mode controller the sliding surface gain is adjusted online depending on the last values of error (e), change of error $(\dot{e})$ and the integral of error $(\sum e)$ by sliding surface slope updating factor (δ). Figure 3 shows tracking performance in baseline error-based tuning sliding mode controller (LTSMC) and sliding mode controller (SMC) without disturbance for step trajectory.

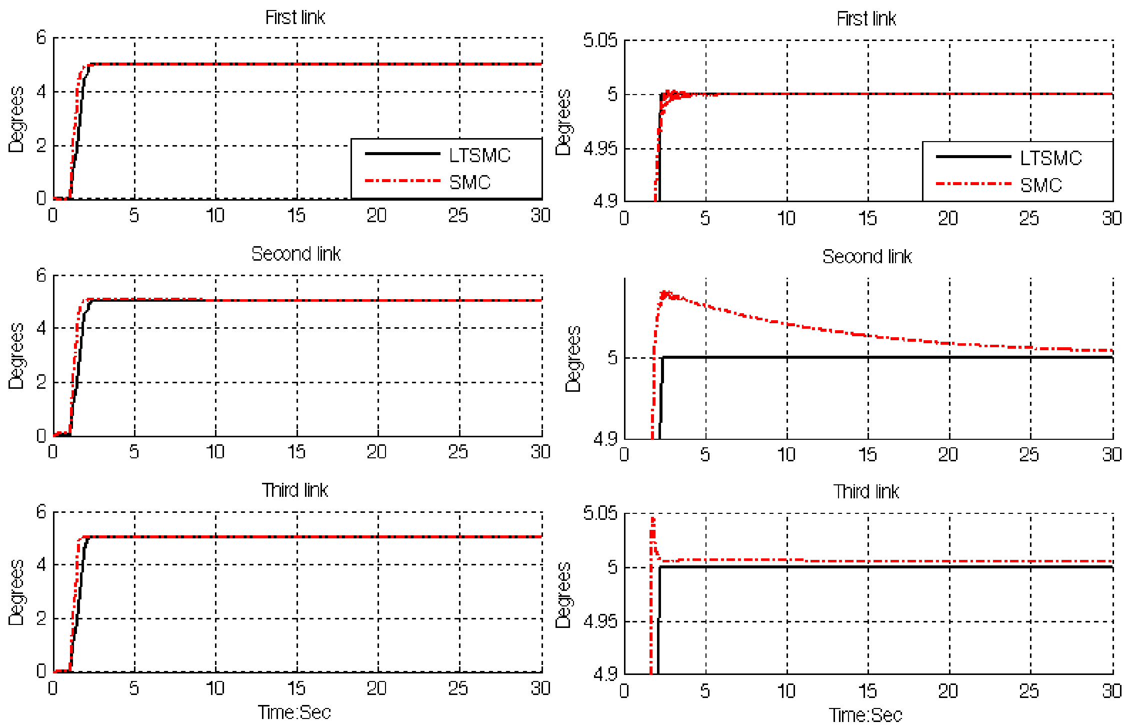


FIGURE 3: LTSMC and SMC for first, second and third link step trajectory performance without disturbance

Based on Figure 3 it is observed that, the overshoot in LTSMC is 0% and in SMC's is 1%, and the rise time in LTSMC's is 0.48 seconds and in SMC's is 0.4 second. From the trajectory MATLAB simulation for LTSMC and SMC in certain system, it was seen that all of two controllers have acceptable performance.

Disturbance Rejection: Figure 4 shows the power disturbance elimination in LTSMC and SMC with disturbance for step trajectory. The disturbance rejection is used to test the robustness comparisons in these two controllers for step trajectory. A band limited white noise with predefined of 40% the power of input signal value is applied to the step trajectory. It found fairly fluctuations in SMC trajectory responses.

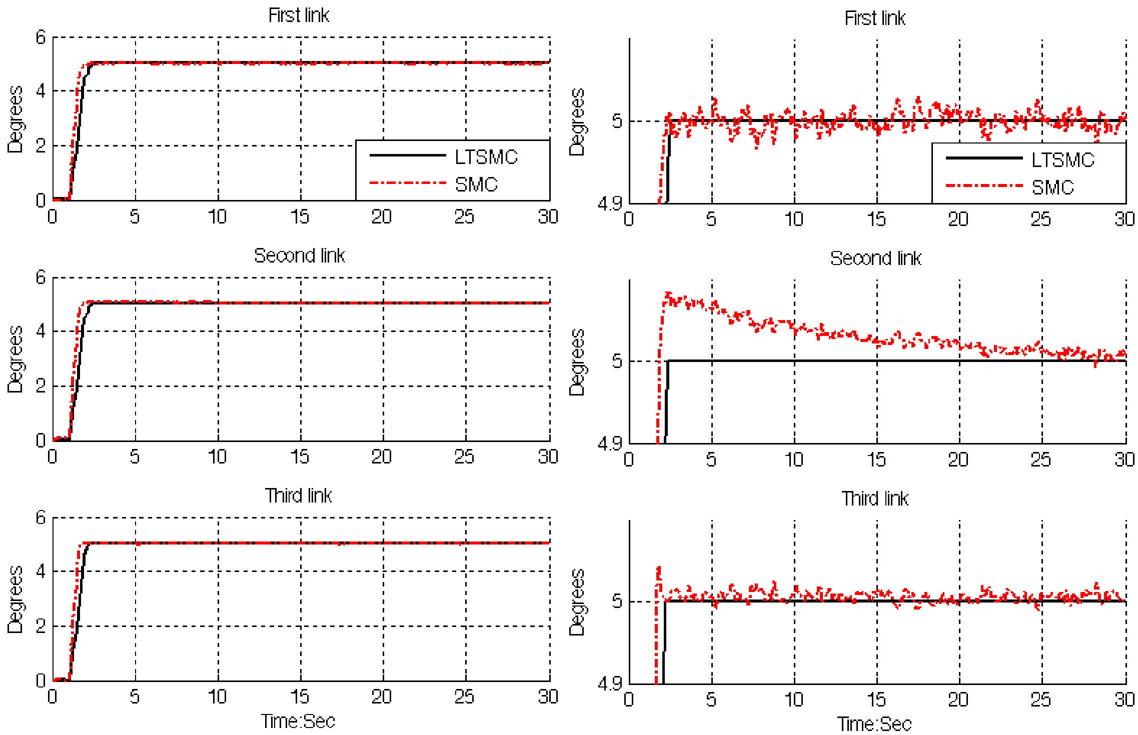


FIGURE 4: LTSMC and SMC for first, second and third link trajectory with 40%external disturbance: step trajectory

Based on Figure 4; by comparing step response trajectory with 40% disturbance of relative to the input signal amplitude in LTSMC and SMC, LTSMC's overshoot about **(0%)** is lower than SMC's (**8%**)**.** SMC's rise time **(0.5 seconds)** is lower than LTSMC's **(0.8 second).** Besides the Steady State and RMS error in LTSMC and SMC it is observed that, error performances in LTSMC **(Steady State error =1.3e-12 and RMS error=1.8e-12)** are about lower than SMC's **(Steady State error=10e-4 and RMS error=11e-4).** Based on Figure 4, SMC has moderately oscillation in trajectory response with regard to 40% of the input signal amplitude disturbance but LTSMC has stability in trajectory responses in presence of uncertainty and external disturbance. Based on Figure 4 in presence of 40% unstructured disturbance, LTSMC's is more robust than SMC because LTSMC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas SMC cannot.

Torque Performance: Figure 5 and 6 have indicated the power of chattering rejection in LTSMC and SMC with 40% disturbance and without disturbance.

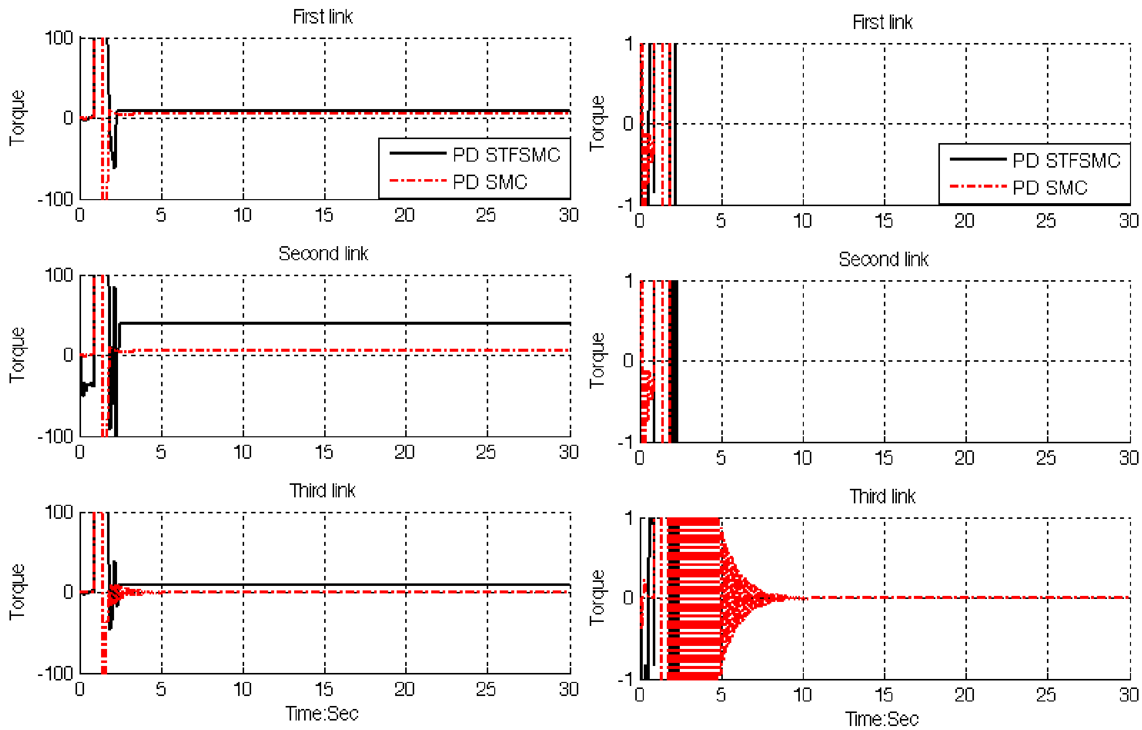


FIGURE 5: LTSMC and SMC for first, second and third link torque performance without disturbance

Figure 5 shows torque performance for first three links robot manipulator in LTSMC and SMC without disturbance. Based on Figure 5, LTSMC and SMC give considerable torque performance in certain system and all two controllers eliminate the chattering phenomenon in certain system. Figure 6 has indicated the robustness in torque performance for three links robot manipulator in LTSMC and SMC in presence of 40% disturbance. Based on Figure 6, it is observed that SMC controller has oscillation but LTSMC has steady in torque performance. This is mainly because pure SMC are robust but they have limitation in presence of external disturbance. The LTSMC gives significant chattering elimination when compared to SMC. This elimination of chattering phenomenon is very significant in presence of 40% disturbance. This challenge is one of the most important objectives in this research.

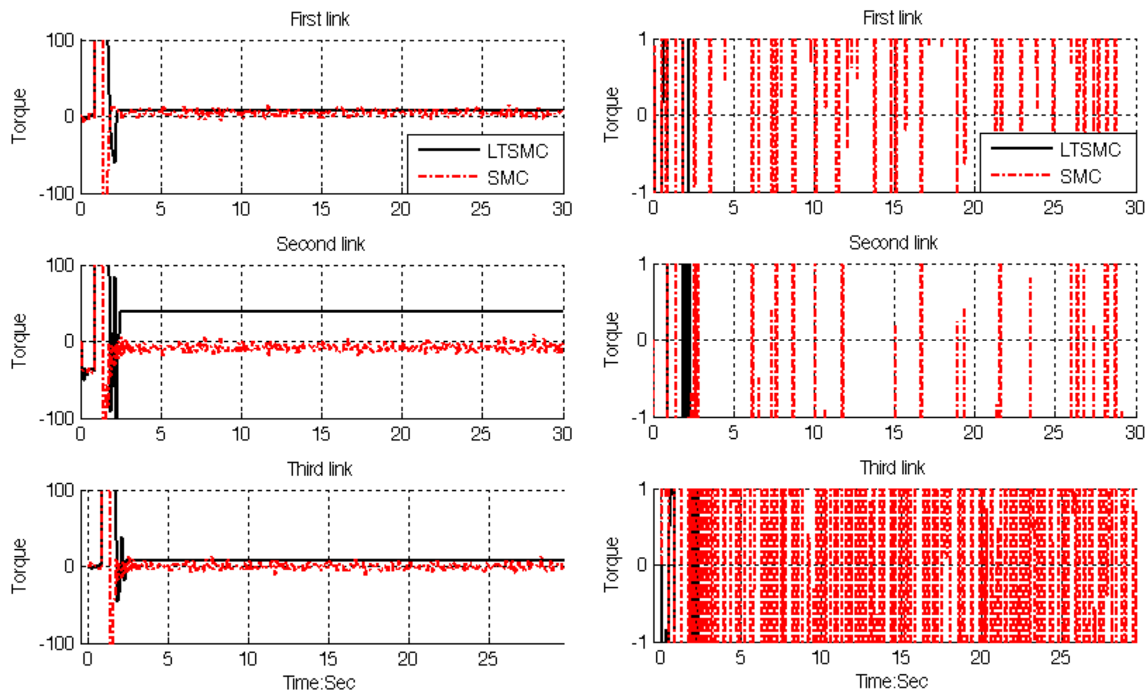


FIGURE 6: LTSMC and SMC for first, second and third link torque performance with40% disturbance

SMC has limitation to eliminate the chattering in presence of highly external disturbance (e.g., 40% disturbance) but LTSMC is a robust against to highly external disturbance.

Steady state error: Figure 7 is shown the error performance in LTSMC and SMC for three links robot manipulator**.** The error performance is used to test the disturbance effect comparisons of these controllers for step trajectory. All three joint's inputs are step function with the same step time (step time= 1 second), the same initial value (initial value=0) and the same final value (final value=5). Based on Figure 3, LTSMC's rise time is about 0.48 second and SMC's rise time is about 0.4 second which caused to create a needle wave in the range of 5 (amplitude=5) and the different width. In this system this time is transient time and this part of error introduced as a transient error. Besides the Steady State and RMS error in LTSMC and SMC it is observed that, error performances in LTSMC **(Steady State error =1.8e-10 and RMS error=1.16e-12)** are about lower than SMC's **(Steady State error=1e-8 and RMS error=1.2e-6)**.

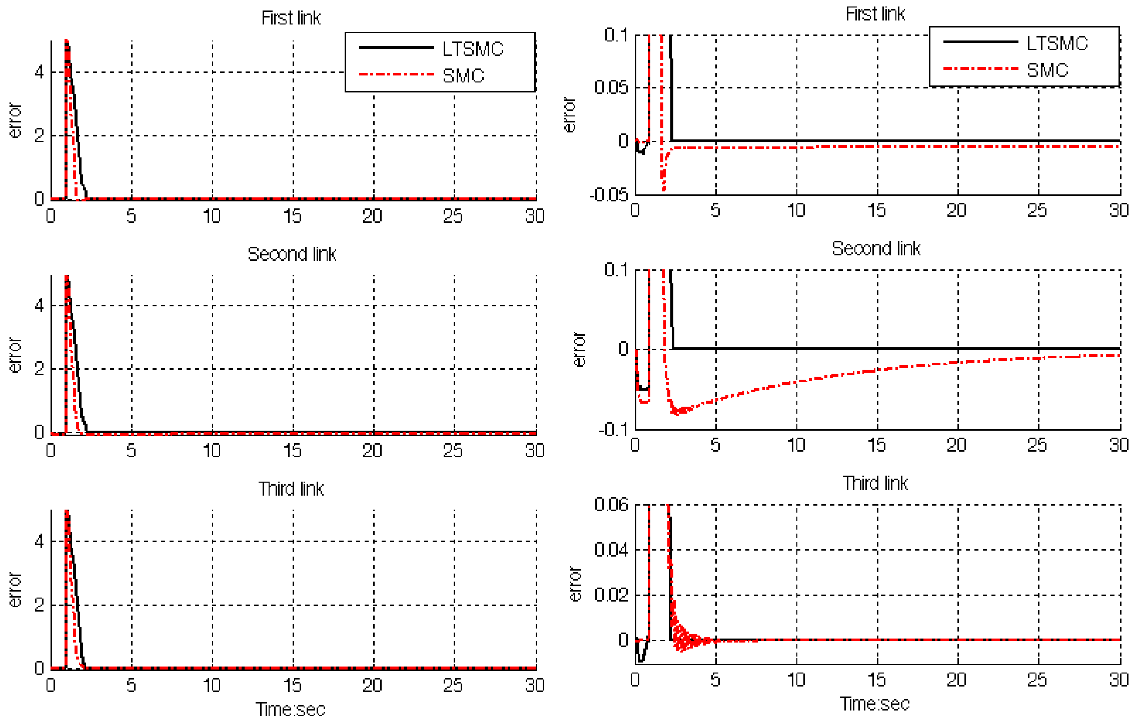


FIGURE 7: LTSMC and SMC for first, second and third link steady state error without disturbance: step trajectory

The LTSMC gives significant steady state error performance when compared to SMC. When applied 40% disturbances in LTSMC the RMS error increased approximately 0.0164% (percent of increase the LTSMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error}=\frac{1.16e-12}{1.1e-12}=0.0122\%)$ and in SMC the RMS error increased approximately 9.17% (percent of increase the PD-SMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbance\ RMS\ error}=\frac{11e-4}{1.2e-6}=9.17\%)$. In this part LTSMC and SMC have been comparatively evaluation through MATLAB simulation, for 3DOF robot manipulator control. It is observed that however LTSMC is dependent of nonlinear dynamic equation of robot manipulator but it can guarantee the trajectory following and eliminate the chattering phenomenon in certain systems, structure uncertain systems and unstructured model uncertainties by online tuning method.

5. CONCLUSION

In this research, a baseline error-based tuning sliding mode controller (LTSMC) is design and applied to robot manipulator. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and linear error-based tuning. The sliding surface gain (λ) is adjusted by linear error-based tuning method. The sliding surface slope updating factor (δ) of linear error-based tuning part can be changed with the changes in error, change of error and the integral (summation) of error. Sliding surface gain is adapted on-line by sliding surface slope updating factor. In pure sliding mode controller the sliding surface gain is chosen by trial and error, which means pure sliding mode controller had to have a prior knowledge of the system uncertainty. If the knowledge is not available error performance and chattering phenomenon are go up. In linear error-based tuning sliding mode controller the sliding surface gain are updated on-line to compensate the system unstructured uncertainty. The simulation results exhibit that the linear error-based tuning sliding mode controller works well in various situations.

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] L. Cheng, Z. G. Hou, M. Tan, D. Liu and A. M. Zou, "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.

[5] J. J. D'Azzo, C. H. Houpis and S. N. Sheldon, *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, L. Fridman, A. Pisano and E. Usai, "Analysis of chattering in systems with second-order sliding modes," *IEEE Transactions on Automatic Control,* No. 11, vol. 52,pp. 2085-2102, 2007.

[8] J. Wang, A. Rad and P. Chan, "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," *Fuzzy Sets and Systems,* No. 1, vol. 122,pp. 21-30, 2001.

[9] C. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal of Robotics and Automation,* No. 3, vol. 2, pp. 171-179, 1986.

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

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

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

[13] F. T. Cheng, T. L. Hour, Y. Y. Sun and T. H. Chen, "Study and resolution of singularities for a 6-DOF PUMA manipulator," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, No. 2,* 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," Conference Proceedings, 2005.

[16] D. Nguyen-Tuong, M. Seeger and J. Peters, "Computed torque control with nonparametric regression models," *IEEE conference proceeding,* 2008, pp. 212-217.

[17] V. Utkin, "Variable structure systems with sliding modes," *Automatic Control, IEEE Transactions on,* No. 2, vol. 22, pp. 212-222, 2002.

[18] R. A. DeCarlo, S. H. Zak and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial," *Proceedings of the IEEE,* No. 3, vol. 76, pp. 212-232, 2002.

[19] K. D. Young, V. Utkin and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE conference proceeding,* 2002, pp. 1-14.

[20] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," *IEEE Transactions on Industrial Electronics,* No. 1, 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,* No. 2, vol. 38, pp. 465-492, 1983.

[22] J. J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of Control,* No. 2, vol. 40, pp. 421-434, 1984.

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

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

[25] M. Ertugrul and O. Kaynak, "Neuro sliding mode control of robotic manipulators," *Mechatronics Journal,* No. 1, 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,* No. 7, 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,* No. 1, vol. 4, pp. 79-85, 2002.

[28] J. Moura and N. Olgac, "A comparative study on simulations vs. experiments of SMCPE," *IEEE conference proceeding,* 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,* No. 4, vol. 18, pp. 798-810, 2010.

[30] B. Wu, Y. Dong, S. Wu, D. Xu and K. Zhao, "An integral variable structure controller with fuzzy tuning design for electro-hydraulic driving Stewart platform," *IEEE conference proceeding,* 2006, pp. 5-945.

[31] Farzin Piltan , N. Sulaiman, Zahra Tajpaykar, Payman Ferdosali, Mehdi Rashidi, "Design Artificial Nonlinear Robust Controller Based on CTLC and FSMC with Tunable Gain," International Journal of Robotic and Automation, 2 (3): 205-220, 2011.

[32] 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

[33] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samira Soltani, Samaneh Roosta, "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.

[34] Farzin Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , "A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator," World Applied Science Journal, 12 (12): 2330-2336, 2011.

[35] Farzin Piltan , A. H. Aryanfar, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli "Design Adaptive Fuzzy Robust Controllers for Robot Manipulator," World Applied Science Journal, 12 (12): 2317-2329, 2011.

[36] Farzin Piltan, N. Sulaiman , Arash Zargari, Mohammad Keshavarz, Ali Badri , "Design PID-Like Fuzzy Controller With Minimum Rule Base and Mathematical Proposed On-line Tunable Gain: Applied to Robot Manipulator," International Journal of Artificial intelligence and expert system, 2 (4):184-195, 2011.

[37] Farzin Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, "Design On-Line Tunable Gain Artificial Nonlinear Controller," Journal of Advances In Computer Research, 2 (4): 75-83, 2011.

[38] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, " Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine," International Journal of Engineering, 5 (4):302-312, 2011.

[39] Farzin Piltan, N. Sulaiman, Samaneh Roosta, M.H. Marhaban, R. Ramli, "Design a New Sliding Mode Adaptive Hybrid Fuzzy Controller," Journal of Advanced Science & Engineering Research , 1 (1): 115-123, 2011.

[40] Farzin Piltan, Atefe Gavahian, N. Sulaiman, M.H. Marhaban, R. Ramli, "Novel Sliding Mode Controller for robot manipulator using FPGA," Journal of Advanced Science & Engineering Research, 1 (1): 1-22, 2011.

[41] Farzin Piltan, N. Sulaiman, A. Jalali & F. Danesh Narouei, "Design of Model Free Adaptive Fuzzy Computed Torque Controller: Applied to Nonlinear Second Order System," International Journal of Robotics and Automation, 2 (4):232-244, 2011.

[42] Farzin Piltan, N. Sulaiman, Iraj Asadi Talooki, Payman Ferdosali, "Control of IC Engine: Design a Novel MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ," International Journal of Robotics and Automation, 2 (5):360-380, 2011.

[43] Farzin Piltan, N. Sulaiman, Payman Ferdosali, Mehdi Rashidi, Zahra Tajpeikar, "Adaptive MIMO Fuzzy Compensate Fuzzy Sliding Mode Algorithm: Applied to Second Order Nonlinear System," International Journal of Engineering, 5 (5): 380-398, 2011.

[44] Farzin Piltan, N. Sulaiman, Hajar Nasiri, Sadeq Allahdadi, Mohammad A. Bairami, "Novel Robot Manipulator Adaptive Artificial Control: Design a Novel SISO Adaptive Fuzzy Sliding Algorithm Inverse Dynamic Like Method," International Journal of Engineering, 5 (5): 399-418, 2011.

[45] Farzin Piltan, N. Sulaiman, Sadeq Allahdadi, Mohammadali Dialame, Abbas Zare, "Position Control of Robot Manipulator: Design a Novel SISO Adaptive Sliding Mode Fuzzy PD Fuzzy Sliding Mode Control," International Journal of Artificial intelligence and Expert System, 2 (5):208-228, 2011.

[46] Farzin Piltan, SH. Tayebi HAGHIGHI, N. Sulaiman, Iman Nazari, Sobhan Siamak, "Artificial Control of PUMA Robot Manipulator: A-Review of Fuzzy Inference Engine And Application to Classical Controller ," International Journal of Robotics and Automation, 2 (5):401-425, 2011.

[47] Farzin Piltan, N. Sulaiman, Abbas Zare, Sadeq Allahdadi, Mohammadali Dialame, "Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm," International Journal of Robotics and Automation , 2 (5): 283-297, 2011.

[48] Farzin Piltan, Amin Jalali, N. Sulaiman, Atefeh Gavahian, Sobhan Siamak, “Novel Artificial Control of Nonlinear Uncertain System: Design a Novel Modified PSO SISO Lyapunov Based Fuzzy Sliding Mode Algorithm ,” International Journal of Robotics and Automation, 2 (5): 298-316, 2011.

[49] Farzin Piltan, N. Sulaiman, Amin Jalali, Koorosh Aslansefat, “Evolutionary Design of Mathematical tunable FPGA Based MIMO Fuzzy Estimator Sliding Mode Based Lyapunov Algorithm: Applied to Robot Manipulator,” International Journal of Robotics and Automation, 2 (5):317-343, 2011.

[50] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator,” International Journal of Engineering, 5 (5):419-434, 2011.

[51] Farzin Piltan, N. Sulaiman, S.Soltani, M. H. Marhaban & R. Ramli, “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.

[52] Farzin Piltan, N. Sulaiman, Mehdi Rashidi, Zahra Tajpaikar, Payman Ferdosali, “Design and Implementation of Sliding Mode Algorithm: Applied to Robot Manipulator-A Review ,” International Journal of Robotics and Automation, 2 (5):265-282, 2011.

[53] Farzin Piltan, N. Sulaiman, Amin Jalali, Sobhan Siamak, and Iman Nazari, “Control of Robot Manipulator: Design a Novel Tuning MIMO Fuzzy Backstepping Adaptive Based Fuzzy Estimator Variable Structure Control ,” International Journal of Control and Automation, 4 (4):91-110, 2011.

[54] Farzin Piltan, N. Sulaiman, Atefeh Gavahian, Samaneh Roosta, Samira Soltani, “On line Tuning Premise and Consequence FIS: Design Fuzzy Adaptive Fuzzy Sliding Mode Controller Based on Lyaponuv Theory,” International Journal of Robotics and Automation, 2 (5):381-400, 2011.

[55] Farzin Piltan, N. Sulaiman, Samaneh Roosta, Atefeh Gavahian, Samira Soltani, “Artificial Chattering Free on-line Fuzzy Sliding Mode Algorithm for Uncertain System: Applied in Robot Manipulator,” International Journal of Engineering, 5 (5):360-379, 2011.

[56] Farzin Piltan, N. Sulaiman and I.AsadiTalooki, “Evolutionary Design on-line Sliding Fuzzy Gain Scheduling Sliding Mode Algorithm: Applied to Internal Combustion Engine,” International Journal of Engineering Science and Technology, 3 (10):7301-7308, 2011.

[57] Farzin Piltan, Nasri B Sulaiman, Iraj Asadi Talooki and Payman Ferdosali.,” Designing On-Line Tunable Gain Fuzzy Sliding Mode Controller Using Sliding Mode Fuzzy Algorithm: Applied to Internal Combustion Engine,” world applied science journal (WASJ), 15 (3): 422-428, 2011

[58] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *Fuzzy Systems, IEEE Transactions on,* No. 2, vol. 8, pp. 186-199, 2002.

[59] H. Medhaffar, N. Derbel and T. Damak, "A decoupled fuzzy indirect adaptive sliding mode controller with application to robot manipulator," *International Journal of Modelling, Identification and Control,* No. 1, 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,* No. 2, 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,* No. 2, vol. 19, pp. 362-368, 2004.

[62] Xiaosong. Lu, "An investigation of adaptive fuzzy sliding mode control for robot manipulator," *Carleton university Ottawa,*2007.

[63] S. Lentijo, S. Pytel, A. Monti, J. Hudgins, E. Santi and G. Simin, "FPGA based sliding mode control for high frequency power converters," *IEEE Conference,* 2004, pp. 3588-3592.

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

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

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

[67] B. Armstrong, O. Khatib and J. Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm," *IEEE Conference,* 2002, pp. 510-518.

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

[69] Farzin Piltan, N. Sulaiman, M. H. Marhaban, Adel Nowzary, Mostafa Tohidian," "Design of FPGA based sliding mode controller for robot manipulator," International Journal of Robotic and Automation, 2 (3): 183-204, 2011.

[70] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* 2002, pp. 160-168.

[71] Farzin Piltan, H. Rezaie, B. Boroomand, Arman Jahed," Design robust back stepping online tuning feedback linearization control applied to IC engine," International Journal of Advance Science and Technology, 42: 183-204, 2012.

[72] Farzin Piltan, I. Nazari, S. Siamak, P. Ferdosali ,"Methodology of FPGA-based mathematical error-based tuning sliding mode controller" International Journal of Control and Automation, 5(1): 89-110, 2012.

[73] Farzin Piltan, M. A. Dialame, A. Zare, A. Badri ,"Design Novel Lookup table changed Auto Tuning FSMC: Applied to Robot Manipulator" International Journal of Engineering, 6(1): 25-40, 2012.

[74] Farzin Piltan, B. Boroomand, A. Jahed, H. Rezaie ,"Methodology of Mathematical Error-Based Tuning Sliding Mode Controller" International Journal of Engineering, 6(2): 96-112, 2012.

[75] Farzin Piltan, F. Aghayari, M. R. Rashidian, M. Shamsodini, "A New Estimate Sliding Mode Fuzzy Controller for Robotic Manipulator" International Journal of Robotics and Automation, 3(1): 45-58, 2012.

[76] Farzin Piltan, M. Keshavarz, A. Badri, A. Zargari , "Design novel nonlinear controller applied to robot manipulator: design new feedback linearization fuzzy controller with minimum rule base tuning method" International Journal of Robotics and Automation, 3(1): 1-18, 2012.

[77] 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.

[78] Samira Soltani & Farzin Piltan, "*Design Artificial Nonlinear Controller Based on Computed Torque like Controller with Tunable Gain*". World Applied Science Journal,14 (9): 1306-1312, 2011.

[79] 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.

[80] Farzin Piltan, A. Hosainpour, E. Mazlomian, M.Shamsodini, M.H Yarmahmoudi. "Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach" International Journal of Robotics and Automation, 3(3): 2012.

[81] Farzin Piltan , M.H. Yarmahmoudi, M. Shamsodini, E.Mazlomian, A.Hosainpour. " PUMA-560 Robot Manipulator Position Computed Torque Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[82] Farzin Piltan, R. Bayat, F. Aghayari, B. Boroomand. "Design Error-Based Linear Model-Free Evaluation Performance Computed Torque Controller" International Journal of Robotics and Automation, 3(3): 2012.

[83] Farzin Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari & Mina Mirazaei . " PUMA-560 Robot Manipulator Position Sliding Mode Control Methods Using MATLAB/SIMULINK and Their Integration into Graduate/Undergraduate Nonlinear Control, Robotics and MATLAB Courses" International Journal of Robotics and Automation, 3(3): 2012.

[84] Farzin Piltan, S. Rahmdel, S. Mehrara, R. Bayat." Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses" International Journal of Engineering, 3(3): 2012.

[85] Farzin Piltan, M. Mirzaie, F. Shahriyari, Iman Nazari & S. Emamzadeh." Design Baseline Computed Torque Controller" International Journal of Engineering, 3(3): 2012.

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. Started 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
- Autonomous Robots
- Emergence of The Thinking Machine
- Household Robots and Automation
- Jacobian and Singularities
- Nanotechnology & Robotics (Nanobots)
- Robot Controller
- Robotic & Automation Software Development
- Robotic Surgery
- Robotic Welding
- Robotics Programming
- Robots Society and Ethics
- Spatial Transformations
- Unmanned (Robotic) Vehicles
- Automation Engineering
- Biotechnology & Robotics
- Forward Kinematics
- Inverse Kinematics
- Methods for Teaching Robots
- Orientation Matrices
- Robot Structure and Workspace
- Robotic Exploration
- Robotic Surgical Procedures
- Robotics Applications
- Robotics Technologies
- Software and Hardware Designing for Robots
- Trajectory Generation

CALL FOR PAPERS

Volume: 3 - **Issue: 4**

i. Paper Submission: October 31, 2012 **ii. Author Notification:** November 30, 2012

iii. Issue Publication: December 2012

CONTACT INFORMATION

Computer Science Journals Sdn BhD
B-5-8 Plaza Mont Kiara, Mont Kiara
50480, Kuala Lumpur, MALAYSIA
Phone: 00603 6204 5627
Fax: 00603 6204 5628
Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2012
COMPUTER SCIENCE JOURNALS SDN BHD
B-5-8 PLAZA MONT KIARA
MONT KIARA
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6204 5627
FAX: 006 03 6204 5628
EMAIL: cscpress@cscjournals.org