Methodology of FPGA-Based Mathematical Error-Based Tuning Sliding Mode Controller

Farzin Piltan, Iman Nazari, Sobhan Siamak and Payman Ferdosali
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
SSP.ROBOTIC@yahoo.com

Abstract

Most of nonlinear controllers need real time mobility operation so one of the most important devices which can be used to solve this challenge is Field Programmable Gate Array (FPGA). FPGA can be used to design a controller in a single chip Integrated Circuit (IC).Design a nonlinear controller for second order nonlinear uncertain dynamical systems is one of the most important challenging works. This paper focuses on the design of a FPGA-based chattering free mathematical error-based tuning sliding mode controller (MTSMC) for highly nonlinear dynamic robot manipulator, in presence of uncertainties. In order to provide high performance nonlinear methodology, sliding mode controller is selected. Pure sliding mode controller can be used to control of partly known nonlinear dynamic parameters of robot manipulator. Conversely, pure sliding mode controller is used in many applications; it has an important drawback namely; chattering phenomenon which it can causes some problems such as saturation and heat the mechanical parts of robot manipulators or drivers. In order to reduce the chattering this research is used the switching function in presence of mathematical error-based method instead of switching function method in pure sliding mode controller. The results demonstrate that the FPGA-based sliding mode controller with switching function is a model-based controllers which works well in certain and partly uncertain system. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. To solve this problem applied mathematical model-free tuning method to FPGA-based sliding mode controller for adjusting the sliding surface gain (λ). Since the sliding surface gain (λ) is adjusted by mathematical model free-based tuning method, it is nonlinear and continuous. In this research new λ is obtained by the previous λ multiple sliding surface slopes updating factor (α). FPGA-based Chattering free mathematical error-based tuning sliding mode controller is stable controller which eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in mathematical error-based tuning sliding mode controller with 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). To have higher implementation speed with good performance TVSC is implemented on Spartan 3E FPGA using Xilinx software (controller computation time=30.2 ns, Max frequency=63.7 MHz and controller action frequency=33 MHZ).

***Keywords**: real time mobility operation, Field Programmable Gate Array (FPGA), nonlinear controller, FPGA-based chattering free mathematical error-based tuning sliding mode controller, uncertainties, chattering phenomenon, robot arm, sliding mode controller, adaptive methodology*

1. Introduction

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). 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). 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 [1-10]. 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. 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 (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 [11-30]. 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. The chattering phenomenon problem can be reduced by using linear saturation boundary layer function in sliding mode control law [31-50]. Lyapunov stability is proved in pure sliding mode controller based on switching (sign) function. The nonlinear equivalent dynamic formulation problem in uncertain system can be solved by using artificial intelligence theorem or online tuning methodology. Fuzzy logic theory is used to estimate the system dynamic. However fuzzy logic controller is used to control complicated nonlinear dynamic systems, but it cannot guarantee stability and robustness. Pure 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 [51-61]. To have high implementation speed, small size device and high speed processing, FPGA is introduced to design FPGA-based mathematical error-based tuning sliding mode controller because FPGA has parallel architecture. FPGAs Xilinx Spartan 3E families are one of the most powerful flexible Hardware Language Description (HDL) programmable IC's. To have the high speed processing FPGA based sliding mode controller in Xilinx ISE 9.1 is designed and implemented [62-63].

2. Literature Review

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]. Control of robot arms using conventional controllers are based on robot arm dynamic modelling. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of robot arms. When the system model is unknown or when it is known but complicated, it is difficult or impossible to use conventional mathematics to process this model [32]. 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 [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. FPGA can be used in wide range area because it is a flexible, technology independent, higher performance, high operation speed, low cost hardware, and fast prototyping [62]. However FPGA has many advantages but one of the most important drawbacks is capacity limitation. In this works design FPGA-based sliding mode controller by Xilinx ISE 9.2 is presented. Research on FPGA-based control of systems is considerably growing as their applications such as industrial automation, robotic surgery and space station's robot arm demand more accuracy, reliability, high performance. For instance, the FPGA-based controls of robot manipulator have been reported in [62-69]. Shao and Sun [63]have proposed an adaptive control algorithm based on FPGA for control of SCARA robot manipulator. They are designed this controller into two micro base controller, the linear part controller is implemented in the FPGA and the nonlinear estimation controller is implemented in DSP. Moreover this controller is implemented in a Xilinx-FPGA XC3S400 with the 20 KHz position loop frequency. The FPGA based servo control and inverse kinematics for Mitsubishi RV-M1 micro robot is presented in [64, 66] which to reduce the limitation of FPGA capacitance they are used 42 steps finite state machine (FSM) in 840 n second. Meshram and Harkare [67-68] have presented a multipurpose FPGA-based 5 DOF robot manipulator using VHDL coding in Xilinx ISE 11.1. This controller has two most important advantages: easy to implement and flexible. Zeyad Assi Obaid et al. [70] have proposed a digital PID fuzzy logic controller using FPGA for tracking tasks that yields semi-global stability of all closed-loop signals. The basic information about FPGA have been reported in [62, 68-72]. A review of design and implementation of FPGA-based systems has been presented in [62]. The FPGA-based sliding mode control of systems has been reported in [73-76]. Lin et al. [73] have presented low cost and high performance FPGA-based fuzzy sliding mode controller for linear induction motor with 80% of flip flops. The fuzzy inference system has 2 inputs (S & $\dot{S}$) and one output K_f with nine rules. Ramos et al. [74] have reported FPGA-based fixed frequency quasi sliding mode control algorithm to control of power inverter. Their proposed controller is implemented in XC4010E-3-PC84 FPGA from XILINX with acceptable experimental and theoretical performance. FPGA-based robust adaptive backstepping sliding mode control for verification of induction motor is reported in [73].

A FPGA chip is programmed by Hardware Description Language (HDL) which contains two types of languages, Very High Description Language (VHDL) and Verilog. VHDL is one of the powerful programming languages that can be used to describe the hardware design. VHDL was developed by the Institute of Electrical and Electronics Engineers (IEEE) in 1987 and Verilog was developed by Gateway Design Automation in 1984 [62, 71]. This research focuses on FPGA-based sliding mode control of robot manipulator and it is implemented in XA3S1600E FPGA from Xilinx in Xilinx-ISE 9.2i software using VHDL code.

Problem Statements: One of the significant challenges in control algorithms is a linear behavior controller design for nonlinear systems (e.g., robot manipulator). Some of robot manipulators which work in industrial processes are controlled by linear PID controllers, but the design of linear controller for robot manipulators is extremely difficult because they are hardly nonlinear and uncertain [1-2, 6]. To reduce the above challenges, the nonlinear robust controller is used to control of robot manipulator. 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 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 this research the nonlinear equivalent dynamic formulation problem and chattering phenomenon in uncertain system is solved by using on-line tuning theorem [8]. To estimate the system dynamics, mathematical error-based sliding mode controller is designed. Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical error-based tuning. This method is based on resolve the on line sliding surface gain (λ) as well as improve the output performance by tuning the sliding surface slope updating factor (α). Mathematical error-based tuning sliding mode controllers is stable model-free controller and eliminates the chattering phenomenon without to use the boundary layer saturation function. Lyapunov stability is proved in mathematical error-based tuning fuzzy sliding mode controller based on switching (sign) function. To have high implementation speed, small size device and high speed processing, FPGA is introduced to design FPGA-based mathematical error-base sliding mode controller because FPGA has parallel architecture. FPGAs Xilinx Spartan 3E families are one of the most powerful flexible Hardware Language Description (HDL) programmable IC's. To have the high speed processing FPGA based sliding mode controller in Xilinx ISE 9.1 is designed and implemented. Section 2, is served as an introduction to the sliding mode controller formulation algorithm and its application to control of robot manipulator. Part 3, introduces and describes the methodology (design PFGA-based mathematical error-based sliding mode 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.

3. Theorem: Dynamic Formulation of Robotic Manipulator, Sliding Mode Formulation Applied to Robot Arm, Proof of Stability and Applied to XILINX-FPGA

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

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

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 [1-29]:

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

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, 41-62]:

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

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]:

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

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]:

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

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

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

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

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

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].

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

where ζ is positive constant.

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

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

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

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 $S(t)$.

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

suppose S is defined as

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

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

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

suppose the second order system is defined as;

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

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

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

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

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

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

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

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

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

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

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

in this method the approximation of $\boldsymbol{U}$ is computed as [6]

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

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

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

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

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

and $\boldsymbol{\tau_{dis}}$ is computed as [1];

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

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

$$\boldsymbol{\tau} = \boldsymbol{\tau_{eq}} + \boldsymbol{K}.\,\mathbf{sgn}(\boldsymbol{S}) \tag{26}$$

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

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

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,

$$\dot{\boldsymbol{V}} = \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{S^T}.\dot{\boldsymbol{M}}.\boldsymbol{S} + \boldsymbol{S^T}\,\boldsymbol{M}\dot{\boldsymbol{S}} \tag{29}$$

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

$$\boldsymbol{M}\dot{\boldsymbol{S}} = -\boldsymbol{V}\boldsymbol{S} + \boldsymbol{M}\dot{\boldsymbol{S}} + \boldsymbol{B} + \boldsymbol{C} + \boldsymbol{G} \tag{30}$$

it is assumed that

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

by substituting (30) in (29)

$$\dot{\boldsymbol{V}} = \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{S^T}\dot{\boldsymbol{M}}\boldsymbol{S} - \boldsymbol{S^T}\boldsymbol{B} + \boldsymbol{C}\boldsymbol{S} + \boldsymbol{S^T}\left(\boldsymbol{M}\dot{\boldsymbol{S}} + \boldsymbol{B} + \boldsymbol{C}\boldsymbol{S} + \boldsymbol{G}\right) = \boldsymbol{S^T}\left(\boldsymbol{M}\dot{\boldsymbol{S}} + \boldsymbol{B} + \boldsymbol{C}\boldsymbol{S} + \boldsymbol{G}\right) \tag{32}$$

suppose the control input is written as follows

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

by replacing the equation (33) in (32)

$$\dot{\boldsymbol{V}} = \boldsymbol{S^T}(\boldsymbol{M}\dot{\boldsymbol{S}} + \boldsymbol{B} + \boldsymbol{C} + \boldsymbol{G} - \widehat{\boldsymbol{M}}\dot{\boldsymbol{S}} - \widehat{\boldsymbol{B}+\boldsymbol{C}}\boldsymbol{S} + \boldsymbol{G} - \boldsymbol{K}\boldsymbol{sgn}(\boldsymbol{S}) = \boldsymbol{S^T}\left(\widetilde{\boldsymbol{M}}\dot{\boldsymbol{S}} + \right. \tag{34}$$

$$\boldsymbol{B}+\boldsymbol{C}\boldsymbol{S}+\boldsymbol{G}-\boldsymbol{K}\boldsymbol{sgn}\boldsymbol{S}$$

it is obvious that

$$\left|\widetilde{\boldsymbol{M}}\dot{\boldsymbol{S}} + \widetilde{\boldsymbol{B}+\boldsymbol{C}}\boldsymbol{S} + \boldsymbol{G}\right| \le \left|\widetilde{\boldsymbol{M}}\dot{\boldsymbol{S}}\right| + \left|\widetilde{\boldsymbol{B}+\boldsymbol{C}}\boldsymbol{S} + \boldsymbol{G}\right| \tag{35}$$

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

$$\boldsymbol{K_u} = \left[\left|\widetilde{\boldsymbol{M}}\dot{\boldsymbol{S}}\right| + |\boldsymbol{B} + \boldsymbol{C}\boldsymbol{S} + \boldsymbol{G}| + \boldsymbol{\eta}\right]_{\boldsymbol{i}}, \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}, \ldots \tag{36}$$

the equation (11) can be written as

$$\boldsymbol{K_u} \ge \left|\left[\widetilde{\boldsymbol{M}}\dot{\boldsymbol{S}} + \boldsymbol{B} + \boldsymbol{C}\boldsymbol{S} + \boldsymbol{G}\right]_{\boldsymbol{i}}\right| + \boldsymbol{\eta_i} \tag{37}$$

therefore, it can be shown that

$$\dot{V} \leq -\sum_{i=1}^{n} \eta_i \, |S_i| \tag{38}$$

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

Field Programmable Gate Array (FPGA): A Field Programmable Gate Array (FPGAs) is a small Field Programmable Device (FPD) that supports thousands of logic gates. Research on FPGA is significantly growing as their applications such as Fast Fourier Transforms (FFT), Discrete Cosine Transforms (DCT), convolution, robotic system, Finite Impulse Response (FIR) filters, power electronics and nonlinear control systems. FPGA is a high speed, low cost, short time to market and small device size. Several semiconductor vendors provide a wide range of FPGAs such as Xilinx, Altera, Atmel and Lattice that each one has own unique architecture. Xilinx and Altera are two well known companies in the silicon technology market. These two companies are the best indicator of FPGA for universities and for giant companies developing digital technology [62]. FPGA was invented by Xilinx Company which this company is the leaders in this technology and the biggest name in the FPGA world. However Xilinx simply can support intermediate generated files but most of FPGA based designer worried about the generated the following files i.e. generated by IP cores, SOF, POF and RBF file programming, and Xilinx settings file which they can settings all part of project [82]. Altera tools are used as a Graphical User Interface (GUI) more than Xilinx which GUI area is more user-friendly for users. Check the delay on the segment in Altera is based on cheap viewer which this part is very helpful. A FPGA chip is programmed by Hardware Description Language (HDL) which contains two types of languages, Very High Description Language (VHDL) and Verilog. VHDL is one of the powerful programming languages that can be used to describe the hardware design. VHDL was developed by the Institute of Electrical and Electronics Engineers (IEEE) in 1987 and Verilog was developed by Gateway Design Automation in 1984 [62, 71]. This research focuses on FPGA-based sliding mode control of robot manipulator and it is implemented in XA3S1600E FPGA from Xilinx in Xilinx-ISE 9.2i software using VHDL code. The introduction of language and architecture of Xilinx FPGA such as VHDL or Verilog in sliding mode control of robot arm will be investigated in this section. The Xilinx Spartan 3E FPGAs has 5 major blocks: Configurable Logic Blocks (CLBs), standard and high speed Input/output Blocks (IOBs), Block RAM’s (BRAMs), Multipliers Blocks, and Digital Clock Managers (DCMs). CLBs is include flexible look up tables (LUTs) to implement memory (storage element) and logic gates. There are 4 slices per CLB each slice has two LUT’s. IOB does control the rate of data between input/output pins and the internal logic gates or elements. It supports bidirectional data with three state operation and multiplicity of signal standards. BRAMs require the data storage including 18-Kbit dual-port blocks. Product two 18-bit binary numbers is done by multiplier blocks. Self-calibrating, digital distributing solution, delaying, multiplying, dividing and phase-shift clock signal are done by DCM [72].

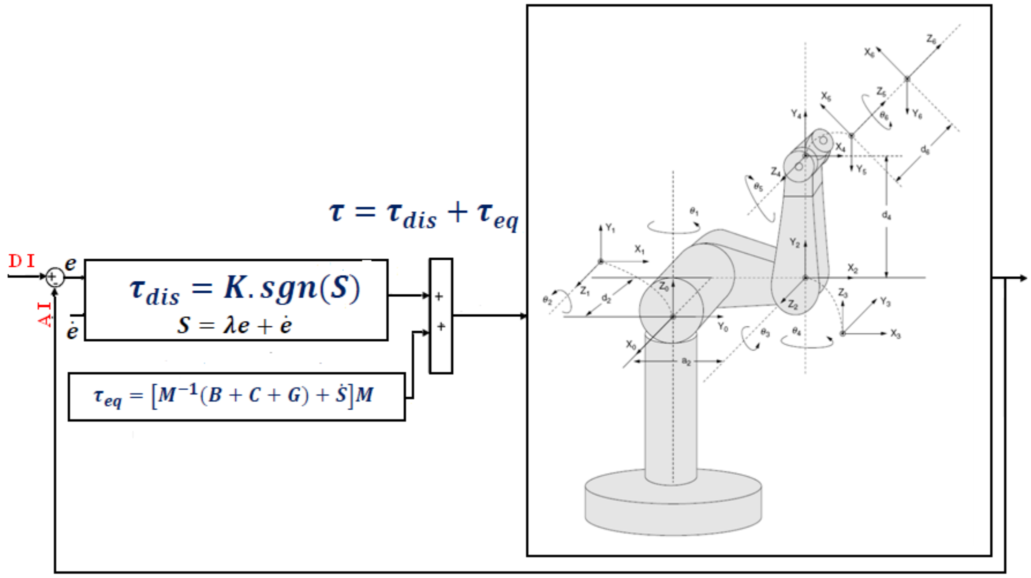


Figure 1: Block Diagram of a Sliding Mode Controller: Applied to Robot Arm

4. Methodology: Design FPGA-based Mathematical Error-based Chattering Free Sliding Mode Controller with Switching Function

Sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical 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 mathematical error-based tuning theorem. In this method mathematical error-based theorem is applied to sliding mode controller to estimate the nonlinear equivalent part. 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 mathematical error-based tuning method and sliding mode controller which this methodology can help to improve system's tracking performance by on-line tuning (mathematical error-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 classical sliding mode controller. Figure 2 shows the mathematical 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.

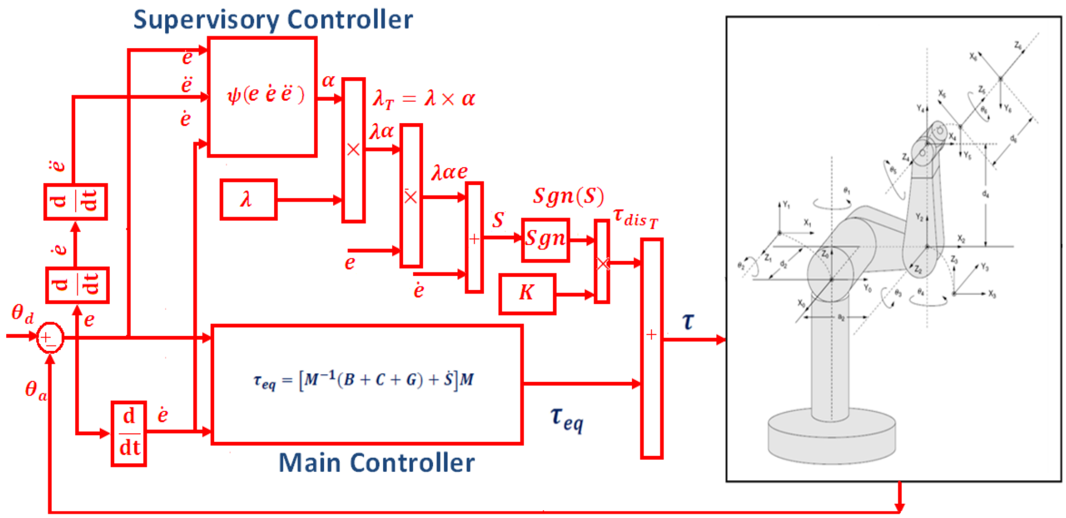


Figure 2. Block Diagram of a Mathematical Error-based Sliding Mode Controller: Applied to Robot Arm

$$\hat{f}(x|\lambda) = \lambda^T\alpha \tag{39}$$

If minimum error (λ^*) is defined by;

$$\lambda^* = arg\ min\ [(Sup|\hat{f}(x|\lambda) - f(x)|)] \tag{40}$$

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 mathematical error-based tuning sliding mode controller is used to adjust the sliding surface slope coefficient. Mathematical 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 second derivative of error ($\ddot{e}$) and an output namely; gain updating factor(α). As a summary design a mathematical error-based tuning is based on the following formulation:

$$\alpha = e^2 - \frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C\right)^5}{1+|e|} + C \tag{41}$$

$$\dot{e}(*) = \dot{e}(t) \quad if\ \dot{e}(t) \geq \dot{e}(t-1)$$

$$\dot{e}(*) = \dot{e}(t-1) \quad if\ \dot{e}(t-1) > \dot{e}(t)$$

Where (α) is gain updating factor, ($\ddot{e}$) is the second derivative of error, ($\dot{e}$) is change of error, (e) is error and C is a coefficient. FPGA supports thousands of gates; it has high operational speed, accurate in response, low cost, short time to market and small size device. As shown in Figure 2, FPGA-based mathematical error-based sliding mode controller divided into two main parts: switching part and equivalent part. Figure 3 shows the discontinuous RTL and VHDL schematic. The first part is focoused on the error compute after that derivation of error is calculated and error multiple by constant coefficient moreover these two parts are added to compute the sliding surface slope (S). Last part is focused on compute the discontinuous torque with respect to switching function to eliminate the chattering.

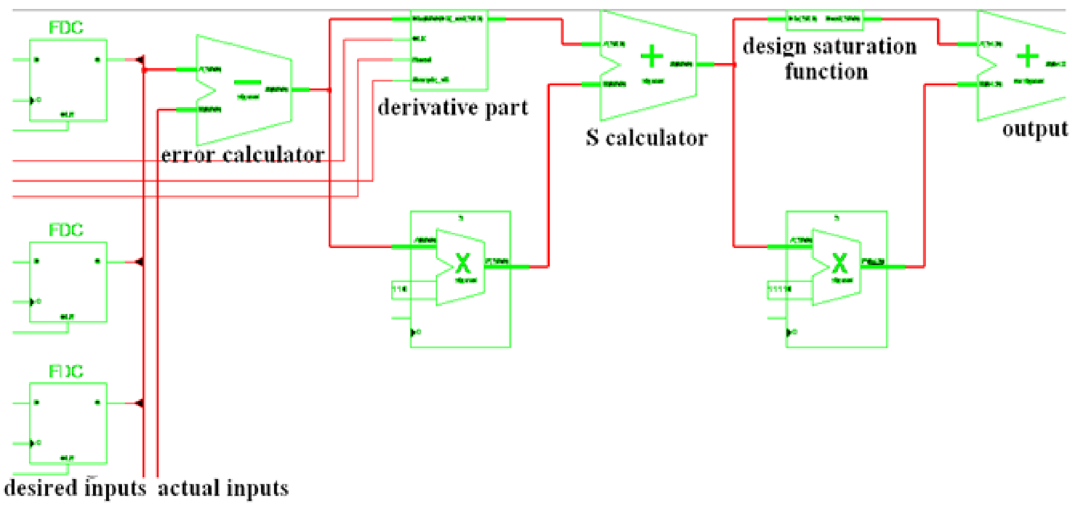


Figure 3. RTL to Design Discontinuous Part

The table 1 indicates the Summary of XA Spartan-3E FPGA Attributes. After implementation mathematical error-based tuning sliding mode controller the most significant resources are; the LUT's (610 out of 29504), CLB (77 out of 3688), Slice (305 out of 14752), Multipliers (27 out of 36), registers (397), and Block RAM memory (648 K) which there are 4 slices per CLB, each slice has two LUT's. So, Number of 4 input LUTs=610, $\frac{610}{2} = 305$ slices, $\frac{305}{4} \cong 77$ CLB's, 610 registers and as a Map report Peak memory usage is 175 MB and registers in the XA3S1600E FPGA. Table 2 demonstrates the FPGA-based sliding mode controller operation summary applied to XA3S1600E-spartan.

Table 1. Summary of XA Spartan-3E FPGA Attributes [82]

Device	System Gates	Equivalent Logic Cells	CLB Array (One CLB = Four Slices)				Distributed RAM bits(1)	Block RAM bits(1)	Dedicated Multipliers	DCMs	Maximum User I/O	Maximum Differential I/O Pairs
			Rows	Columns	Total CLBs	Total Slices						
XA3S100E	100K	2,160	22	16	240	960	15K	72K	4	2	108	40
XA3S250E	250K	5,508	34	26	612	2,448	38K	216K	12	4	172	68
XA3S500E	500K	10,476	46	34	1,164	4,656	73K	360K	20	4	190	77
XA3S1200E	1200K	19,512	60	46	2,168	8,672	136K	504K	28	8	304	124
XA3S1600E	1600K	33,192	76	58	3,688	14,752	231K	648K	36	8	376	156

Notes:

1. By convention, one Kb is equivalent to 1,024 bits.

Table 2. XA3S1600E Device Utilization Summaries

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	216	29,504	1%	
Number of 4 input LUTs	610	29,504	2%	
Logic Distribution				
Number of occupied Slices	342	14,752	2%	
Number of Slices containing only related logic	342	342	100%	
Number of Slices containing unrelated logic	0	342	0%	
Total Number of 4 input LUTs	622	29,504	2%	
Number used as logic	610			
Number used as a route-thru	12			
Number of bonded IOBs	288	376	76%	
IOB Flip Flops	181			
Number of GCLKs	2	24	8%	
Number of MULT18X18SIOs	27	36	75%	
Total equivalent gate count for design	10,334			
Additional JTAG gate count for IOBs	13,824			

The most important contributions in this section are: the successful implementation of pure sliding mode controller for PUMA 560 robot manipulator in a XA3S1600E FPGA and the successful application of FPGA-based controller to position control of robot manipulator.

Proof of Stability: The Lyapunov function in this design is defined as

$$\boldsymbol{V} = \frac{1}{2}\boldsymbol{S^T M S} + \frac{1}{2}\sum_{J=1}^{M}\frac{1}{\boldsymbol{\Upsilon_{sj}}}\boldsymbol{\phi^T}.\boldsymbol{\phi_j} \tag{42}$$

where γ_{sj} is a positive coefficient, $\boldsymbol{\phi} = \boldsymbol{\lambda}^* - \boldsymbol{\lambda}, \boldsymbol{\theta}^*$ is minimum error and λ is adjustable parameter. Since $\dot{M} - 2V$ is skew-symetric matrix;

$$\boldsymbol{S^T M\dot{S}} + \frac{1}{2}\boldsymbol{S^T \dot{M} S} = \boldsymbol{S^T}(\boldsymbol{M\dot{S}} + \boldsymbol{VS}) \tag{43}$$

If the dynamic formulation of robot manipulator defined by

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) \tag{44}$$

the controller formulation is defined by

$$\boldsymbol{\tau} = \widehat{\boldsymbol{M}}\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + \widehat{\boldsymbol{V}}\dot{\boldsymbol{q}}_{\boldsymbol{r}} + \widehat{\boldsymbol{G}} - \boldsymbol{\lambda S} - \boldsymbol{K} \tag{45}$$

According to (43) and (44)

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \widehat{\boldsymbol{M}}\ddot{\boldsymbol{q}}_{\boldsymbol{r}} + \widehat{\boldsymbol{V}}\dot{\boldsymbol{q}}_{\boldsymbol{r}} + \widehat{\boldsymbol{G}} - \boldsymbol{\lambda S} - \boldsymbol{K} \tag{46}$$

Since $\dot{\boldsymbol{q}}_{\boldsymbol{r}} = \dot{\boldsymbol{q}} - \boldsymbol{S}$ and $\ddot{\boldsymbol{q}}_{\boldsymbol{r}} = \ddot{\boldsymbol{q}} - \dot{\boldsymbol{S}}$

$$\boldsymbol{M\dot{S}} + (\boldsymbol{V} + \boldsymbol{\lambda})\boldsymbol{S} = \Delta\boldsymbol{f} - \boldsymbol{K} \tag{47}$$

$$\boldsymbol{M\dot{S}} = \boldsymbol{\Delta f} - \boldsymbol{K} - \boldsymbol{VS} - \boldsymbol{\lambda S}$$

The derivation of V is defined

$$\dot{\boldsymbol{V}} = \boldsymbol{S^T M \dot{S}} + \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{S^T \dot{M} S} + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}} \tag{48}$$

$$\dot{\boldsymbol{V}} = \boldsymbol{S^T}(\boldsymbol{M\dot{S}} + \boldsymbol{VS}) + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}}$$

Based on (46) and (47)

$$\dot{\boldsymbol{V}} = \boldsymbol{S^T}(\boldsymbol{\Delta f} - \boldsymbol{K} - \boldsymbol{VS} - \boldsymbol{\lambda S} + \boldsymbol{VS}) + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}} \tag{49}$$

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

$$\dot{\boldsymbol{V}} = \sum_{J=1}^{M} \left[\boldsymbol{S_j}(\boldsymbol{\Delta f}_{\mathbf{j}} - \boldsymbol{K_j})\right] - \boldsymbol{S^T \lambda S} + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}}$$

suppose α is defined as follows

$$\boldsymbol{\alpha_j} = \boldsymbol{e^2} - \frac{(\left(\frac{\ddot{\boldsymbol{e}}(\boldsymbol{t})}{\dot{\boldsymbol{e}}(*)}\right) - \boldsymbol{C})^{\mathbf{5}}}{\mathbf{1} + |\boldsymbol{e}|} + \mathbf{C} \tag{50}$$

according to 48 and 49;

$$\dot{\boldsymbol{V}} = \sum_{J=1}^{M} \left[\boldsymbol{S_j}(\boldsymbol{\Delta f}_{\mathbf{j}} - \boldsymbol{\lambda^T}[\boldsymbol{e^2} - \frac{\left(\left(\frac{\ddot{\boldsymbol{e}}(\boldsymbol{t})}{\dot{\boldsymbol{e}}(*)}\right) - \boldsymbol{C}\right)^{\mathbf{5}}}{\mathbf{1} + |\boldsymbol{e}|} + \mathbf{C}]\right] - \boldsymbol{S^T \lambda S} + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}} \tag{51}$$

Based on $\boldsymbol{\phi} = \boldsymbol{\theta}^* - \boldsymbol{\theta} \rightarrow \boldsymbol{\theta} = \boldsymbol{\theta}^* - \boldsymbol{\phi}$

$$\dot{\boldsymbol{V}} = \sum_{J=1}^{M} \left[\boldsymbol{S_j}(\boldsymbol{\Delta f}_{\mathbf{j}} - \boldsymbol{\theta^{*T}}[\boldsymbol{e^2} - \frac{\left(\left(\frac{\ddot{\boldsymbol{e}}(\boldsymbol{t})}{\dot{\boldsymbol{e}}(*)}\right) - \boldsymbol{C}\right)^{\mathbf{5}}}{\mathbf{1} + |\boldsymbol{e}|} + \mathbf{C}] + \boldsymbol{\phi^T}[\boldsymbol{e^2} - \frac{\left(\left(\frac{\ddot{\boldsymbol{e}}(\boldsymbol{t})}{\dot{\boldsymbol{e}}(*)}\right) - \boldsymbol{C}\right)^{\mathbf{5}}}{\mathbf{1} + |\boldsymbol{e}|} + \mathbf{C}]\right] - \boldsymbol{S^T \lambda S} + \sum_{J=1}^{M} \frac{\mathbf{1}}{\gamma_{sj}} \boldsymbol{\phi^T}.\dot{\boldsymbol{\phi}}_{\boldsymbol{j}} \tag{52}$$

$$\dot{V} = \sum_{J=1}^{M}\left[S_j(\Delta f_j - (\lambda^*)^T[e^2 - \frac{\left(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C\right)^5}{1+|e|} + \mathrm{C}]\right] - S^T\lambda S + \sum_{J=1}^{M}\frac{1}{\gamma_{sj}}\phi_j^T[e^2 - \frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C)^5}{1+|e|} + \mathrm{C} + \dot{\phi}_j])$$

where $\dot{\theta}_j = e^2 - \frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C)^5}{1+|e|} + \mathrm{C}$ is adaption law, $\dot{\emptyset}_j = -\dot{\theta}_j = -[e^2 - \frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C)^5}{1+|e|} + \mathrm{C}]$
$\dot{V}$ is considered by

$$\dot{V} = \sum_{j=1}^{m}[S_j\,\Delta f_j - \left((\lambda_j^*)^T e^2 - \frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C)^5}{1+|e|} + \mathrm{C}\right)] - S^T\lambda S \tag{53}$$

The minimum error is defined by

$$e_{mj} = \Delta f_j - \left((\lambda_j^*)^T e^2 - \frac{(\left(\frac{\ddot{e}(t)}{\dot{e}(*)}\right) - C)^5}{1+|e|} + \mathrm{C}\right) \tag{54}$$

Therefore $\dot{V}$ is computed as

$$\dot{V} = \sum_{j=1}^{m}[S_j\, e_{mj}] - S^T\lambda S \tag{55}$$

$$\le \sum_{j=1}^{m} |S_j\,||e_{mj}| - S^T\lambda S$$

$$= \sum_{j=1}^{m} |S_j\,||e_{mj}| - \lambda_j {S_j}^2$$

$$= \sum_{j=1}^{m} |S_j\,|\big(|e_{mj}| - \lambda_j S_j\big) \tag{56}$$

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

$$Sup_{x\in U}|f(x) - g(x)| < \epsilon \tag{57}$$

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

$$\boldsymbol{if\ \lambda_j = \alpha \quad that \quad \alpha|S_j| > e_{mj}\,(S_j \neq 0) \qquad then \quad \dot{V} < 0\, for\,(S_j \neq 0)} \tag{58}$$

5. Results

Pure sliding mode controller has difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining sliding mode controller and mathematical error-based tuning in a single IC chip or combining sliding mode controller by fuzzy logic method (FSMC). These methods can improve the system's tracking performance by online tuning method or soft computing method. Proposed 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), change of error ($\dot{e}$) and power two of derivative of error ($\ddot{e}$) by sliding surface slope updating factor (α). Fuzzy sliding mode controller is based on applied fuzzy logic in sliding mode controller to estimate the dynamic formulation in equivalent part. Mathematical error-based tuning sliding mode 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.

This section is focused on compare between Sliding Mode Controller (SMC), Fuzzy Sliding Mode Controller (FSMC) and FPGA-based mathematical error-based tuning Sliding Mode Controller (MTSMC). 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 10%, 20% and 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.

Trajectory: Tracking performances: Based on (27) 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 FSMC 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 mathematical 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 second derivation of error ($\ddot{e}$) by sliding surface slope updating factor (α). Figure 4 shows tracking performance in mathematical error-based tuning sliding mode controller (MTSMC), fuzzy sliding mode controller (FSMC) and SMC without disturbance for step trajectory.

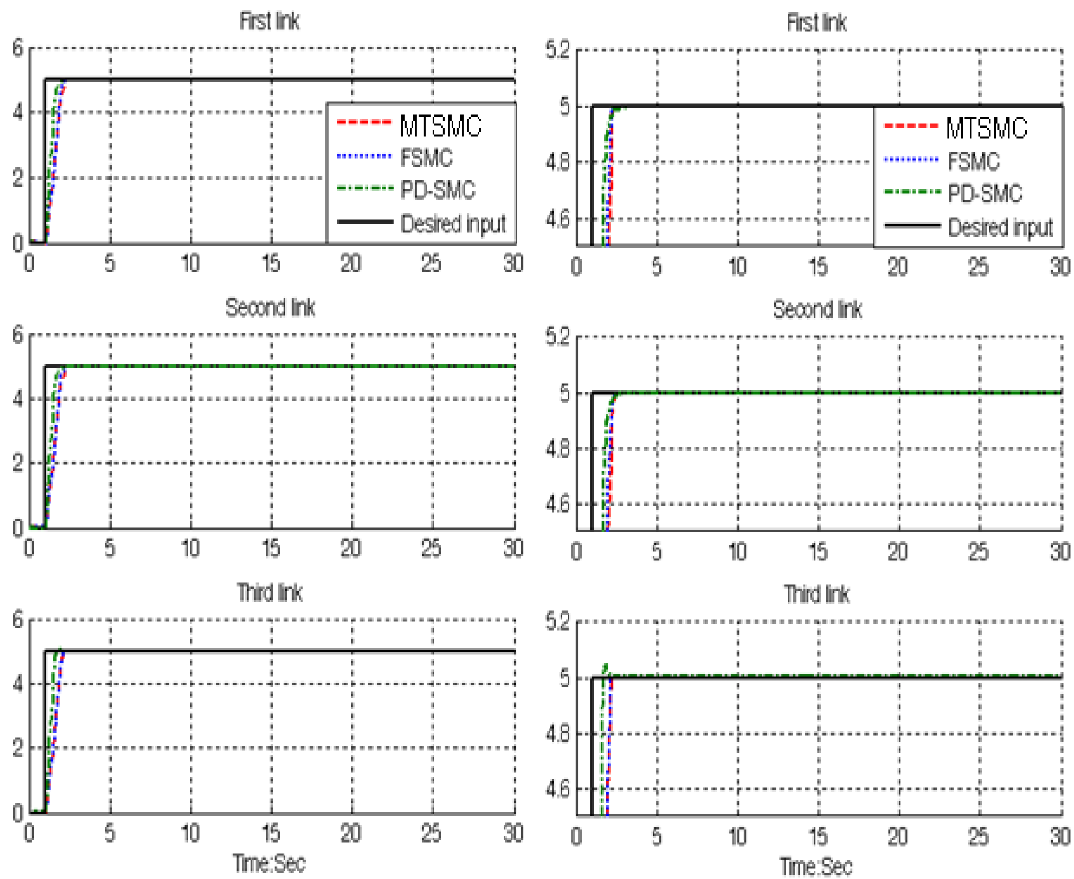


Figure 4. FSMC, MTSMC, Desired Input and SMC for First, Second and Third Link Step Trajectory Performance without Disturbance

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

Disturbance rejection: Figures 5 to 7 show the power disturbance elimination in MTSMC, SMC and FSMC 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 10%, 20% and 40% the power of input signal value is applied to the step trajectory. It found fairly fluctuations in SMC and FSMC trajectory responses.

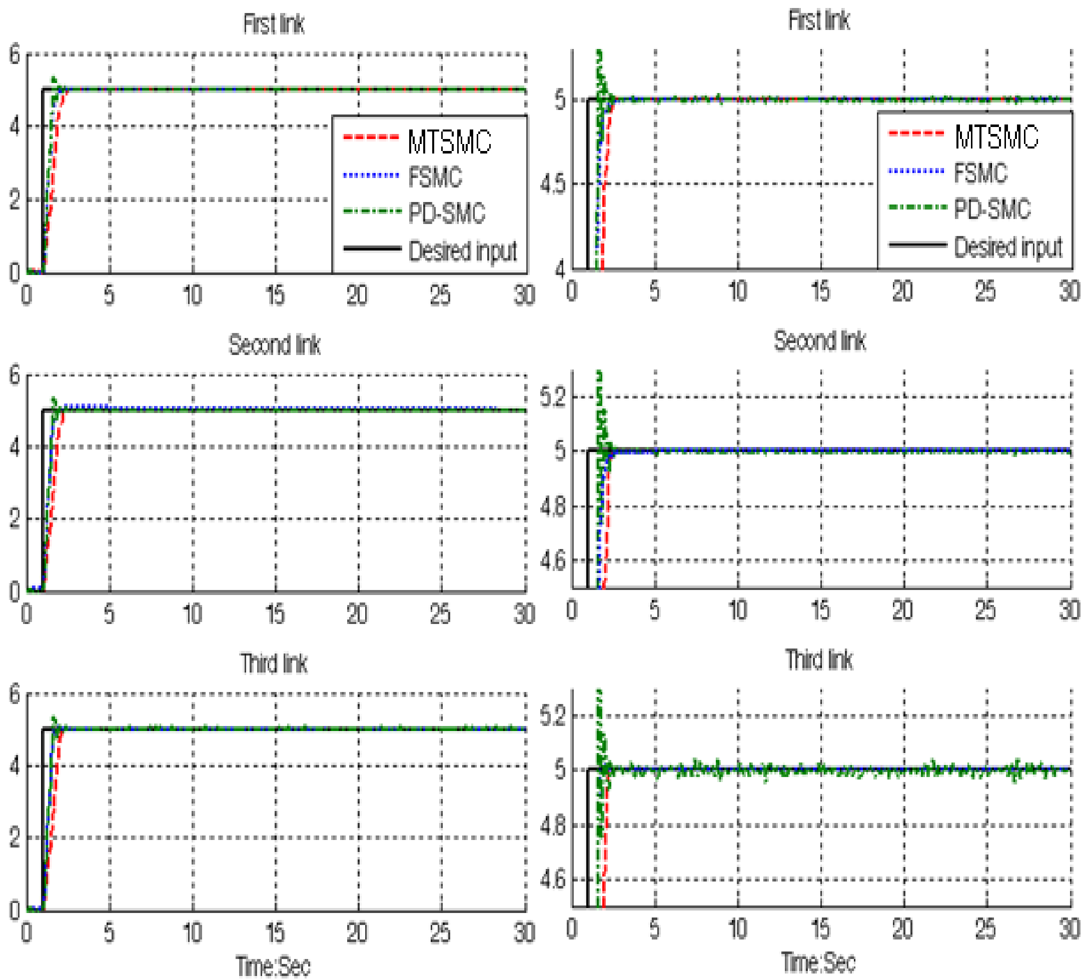


Figure 5. Desired Input, MTSMC, FSMC and SMC for First, Second and Third Link Trajectory with 10% External Disturbance: Step Trajectory

Based on Figure 5; by comparing step response trajectory with 10% disturbance of relative to the input signal amplitude in MTSMC, FSMC and SMC, MTSMC's overshoot about (0%) is lower than FSMC's **(0.5%)** and SMC's (**1%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.63 second)** and MTSMC's **(0.65 second).** Besides the Steady State and RMS error in MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =1.08e-12 and RMS error=1.5e-12)** are bout lower than FSMC **(Steady State error =1.08e-6 and RMS error=1.5e-6)** and SMC's **(Steady State error=1.6e-6 and RMS error=1.9e-6)**.

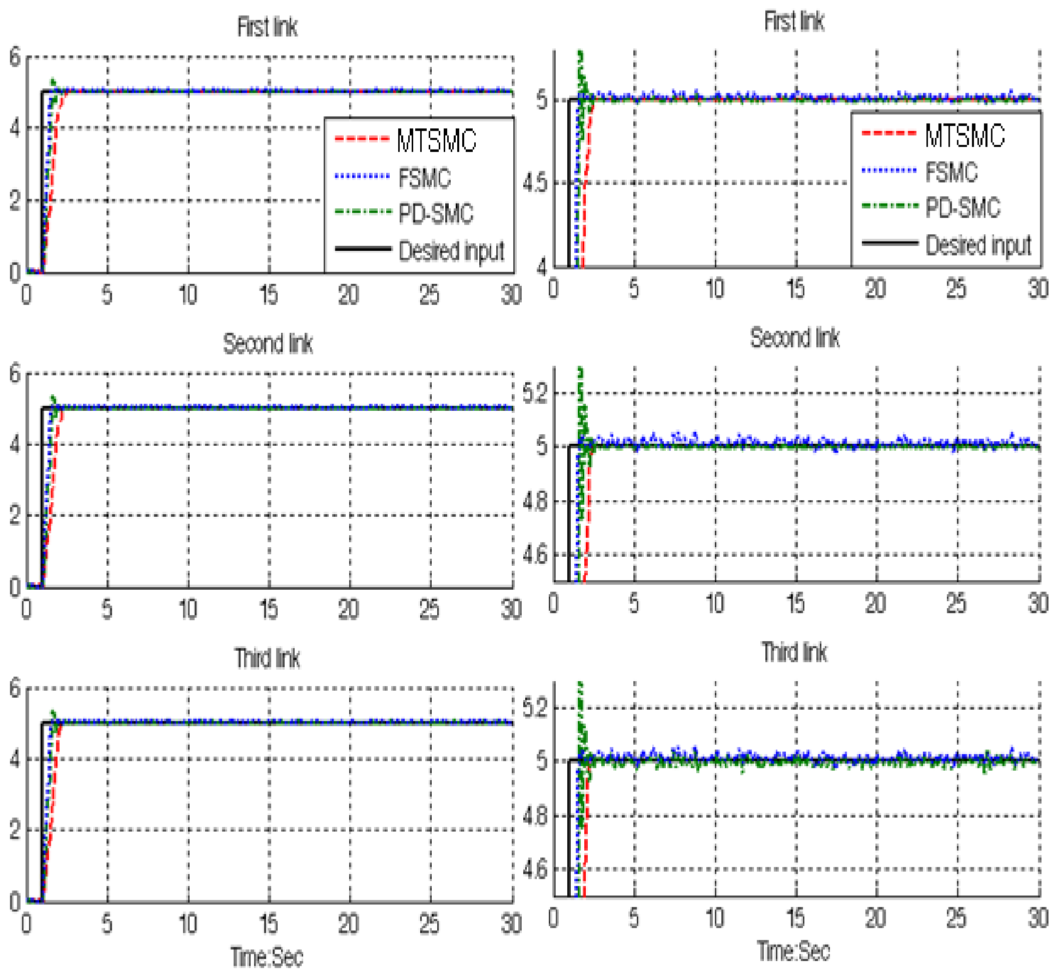


Figure 6. Desired Input, MTSMC, FSMC and SMC for First, Second and Third Link Trajectory with 20% External Disturbance: Step Trajectory

Based on Figure 6; by comparing step response trajectory with 20% disturbance of relative to the input signal amplitude in MTSMC, FSMC and SMC, MTSMC's overshoot about **(0%)** is lower than FSMC's **(1.8%)** and SMC's (**2.1%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.63 second)** and MTSMC's **(0.66 second).** Besides the Steady State and RMS error in FTFSMC, FSMC and PD-SMC it is observed that, error performances in MTSMC **(Steady State error =1.2e-12 and RMS error=1.8e-12)** are about lower than FSMC **(Steady State error =1.7e-5 and RMS error=2e-5)** and SMC's **(Steady State error=1.8e-5 and RMS error=2e-5)**. Based on Figure 6, it was seen that, MTSMC's performance is better than FSMC and SMC because MTSMC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas FSMC and SMC cannot.

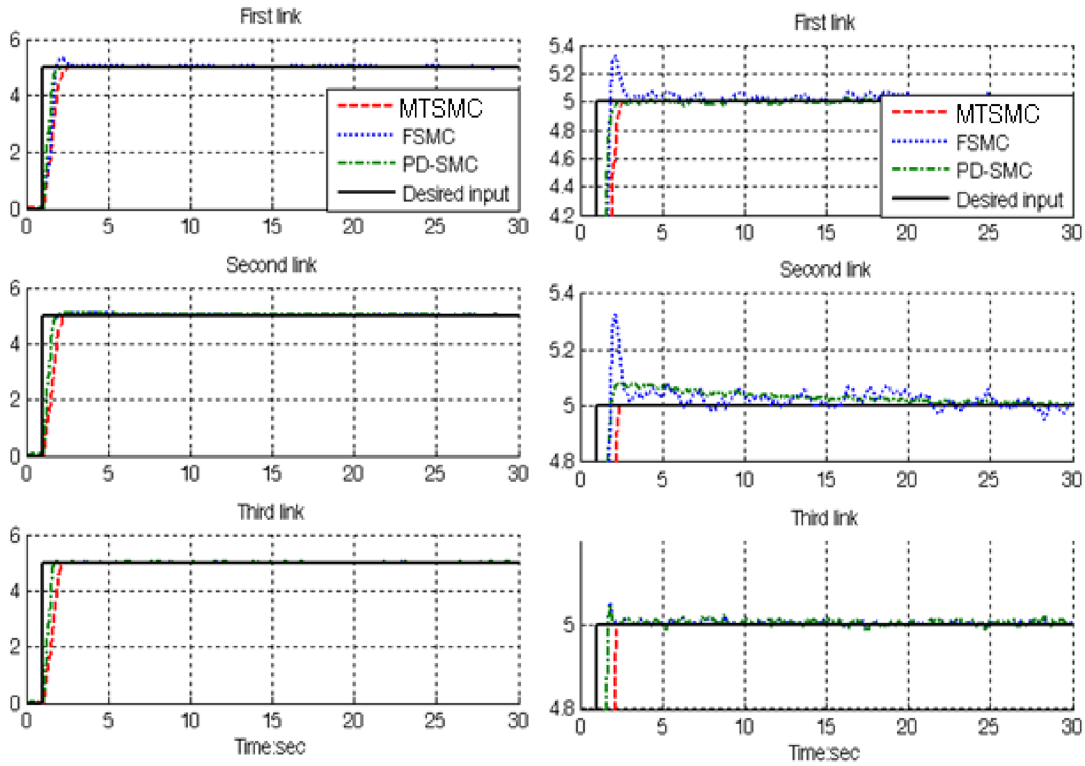


Figure 7. Desired Input, MTSMC, FSMC and SMC for First, Second and Third Link Trajectory with 40% External Disturbance: Step Trajectory

Based on Figure 7; by comparing step response trajectory with 40% disturbance of relative to the input signal amplitude in MTSMC, SMC and FSMC, MTSMC's overshoot about **(0%)** is lower than FSMC's **(6%)** and PD-SMC's (**8%**)**.** SMC's rise time **(0.5 seconds)** is lower than FSMC's **(0.7 second)** and MTSMC's **(0.8 second).** Besides the Steady State and RMS error in MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =1.3e-12 and RMS error=1.8e-12)** are about lower than FSMC **(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 7, FSMC and SMC have moderately oscillation in trajectory response with regard to 40% of the input signal amplitude disturbance but MTSMC has stability in trajectory responses in presence of uncertainty and external disturbance. Based on Figure 7 in presence of 40% unstructured disturbance, MTSMC's is more robust than FSMC and SMC because MTSMC can auto-tune the sliding surface slope coefficient as the dynamic manipulator parameter's change and in presence of external disturbance whereas FSMC and SMC cannot.

Torque performance: Figures 8 and 9 have indicated the power of chattering rejection in MTSMC, SMC and FSMC with 40% disturbance and without disturbance.

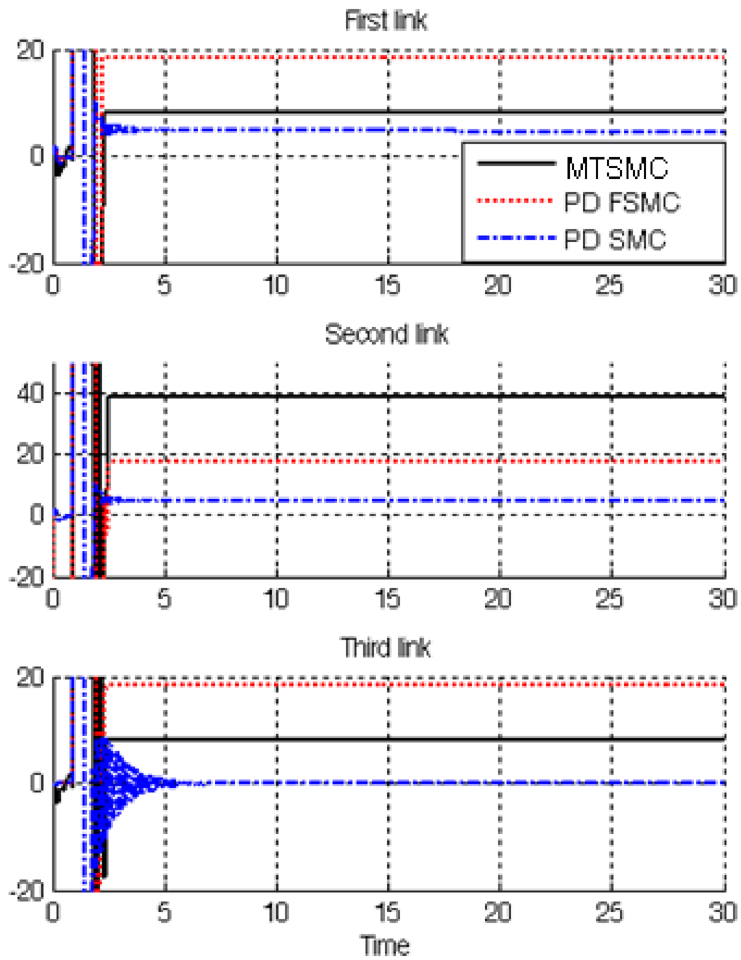


Figure 8. MTSMC, SMC and FSMC for First, Second and third Link Torque Performance without Disturbance

Figure 8 shows torque performance for first three links PUMA robot manipulator in MTSMC, SMC and FSMC without disturbance. Based on Figure 8, MTSMC, SMC and FSMC give considerable torque performance in certain system and all three of controllers eliminate the chattering phenomenon in certain system. Figure 9 has indicated the robustness in torque performance for first three links PUMA robot manipulator in MTSMC, SMC and FSMC in presence of 40% disturbance. Based on Figure 9, it is observed that SMC and FSMC controllers have oscillation but MTSMC has steady in torque performance. This is mainly because pure SMC with saturation function and fuzzy sliding mode controller with saturation function are robust but they have limitation in presence of external disturbance. The MTSMC gives significant chattering elimination when compared to FSMC 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 research.

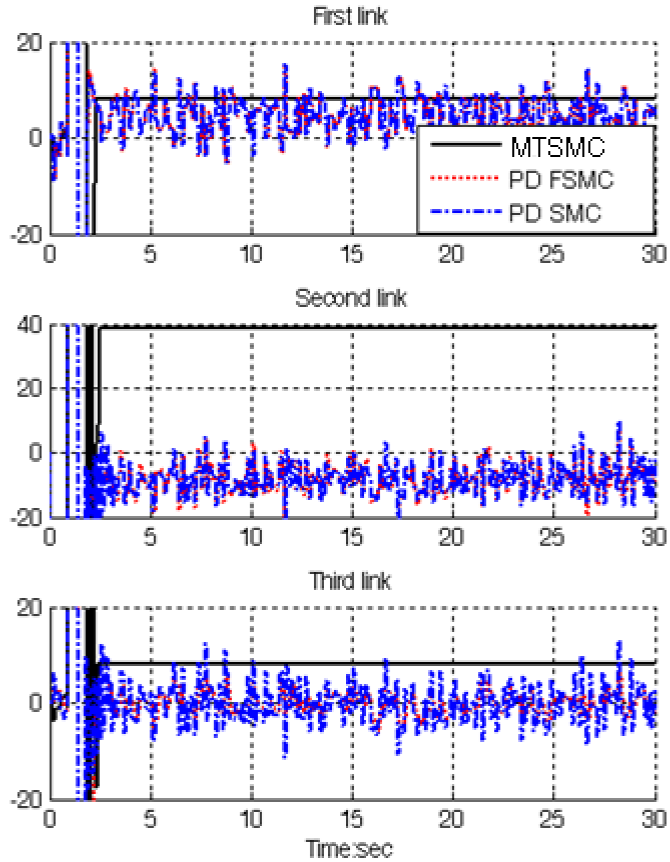


Figure 9. MTSMC, SMC and FSMC for First, Second and Third Link Torque Performance with40% Disturbance

Based on Figure 9 it is observed that, however mathematical tuning error-based sliding mode controller (MTSMC) is a model-based controller that estimate the nonlinear dynamic equivalent formulation by system's performance but it has significant torque performance (chattering phenomenon) in presence of uncertainty and external disturbance. SMC and FSMC have limitation to eliminate the chattering in presence of highly external disturbance (e.g., 40% disturbance) but MTSMC is a robust against to highly external disturbance.

Steady state error: Figure 10 is shown the error performance in MTSMC, SMC and FSMC 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 4, MTSMC's rise time is about 0.6 second, SMC's rise time is about 0.483 second and FSMC'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 MTSMC, FSMC and SMC it is observed that, error performances in MTSMC **(Steady State error =0.9e-12 and RMS error=1.1e-12)** are about lower than FSMC **(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)**.

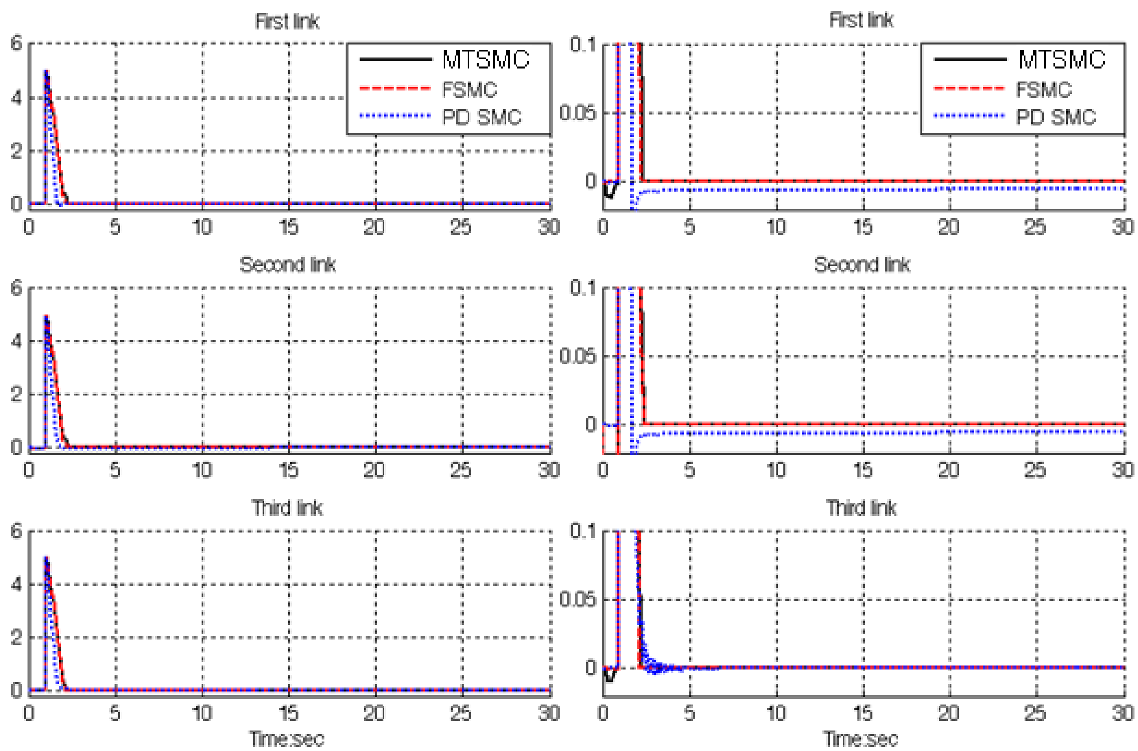


Figure 10. MTSMC, SMC and FSMC for First, Second and Third Link Steady State Error without Disturbance: Step Trajectory

The MTSMC gives significant steady state error performance when compared to FSMC and SMC. When applied 40% disturbances in MTSMC the RMS error increased approximately 0.0164% (percent of increase the MTSMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbanceRMS\ error} = \frac{1.8e-12}{1.1e-12} = 0.0164\%$), in FSMC the RMS error increased approximately 6.9% (percent of increase the FSMC 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 PD-SMC RMS error=$\frac{(40\%\ disturbance\ RMS\ error}{no\ disturbance\ RMS\ error} = \frac{11e-4}{1.2e-6} = 9.17\%$). In this part MTSMC, SMC and FSMC have been comparatively evaluation through MATLAB simulation, for PUMA robot manipulator control. It is observed that however MTSMC 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.

Timing Detail: As a simulation result in XILINX ISE 9.1, it is observed that; this controller is able to make as a fast response at 15.716 ns clock period with 63.29 MHZ of a maximum frequency and 4.407 ns for minimum input arrival time after clock. From investigation and synthesis summary, 30.286 ns for maximum input arrival time after clock with 33.018 MHZ frequencies, this design has 15.716 ns delays for each controller to 46 logic elements and also the offset before CLOCK is 55.773 ns for 132 logic gates. Figure 11 has indicated the displacement, error performance, teta discontinuous (torque performance) at different time. All three joint's inputs are step function with the same step time (step time= 6 micro second), the same initial value (initial value=0) and the same final value (final value=5).

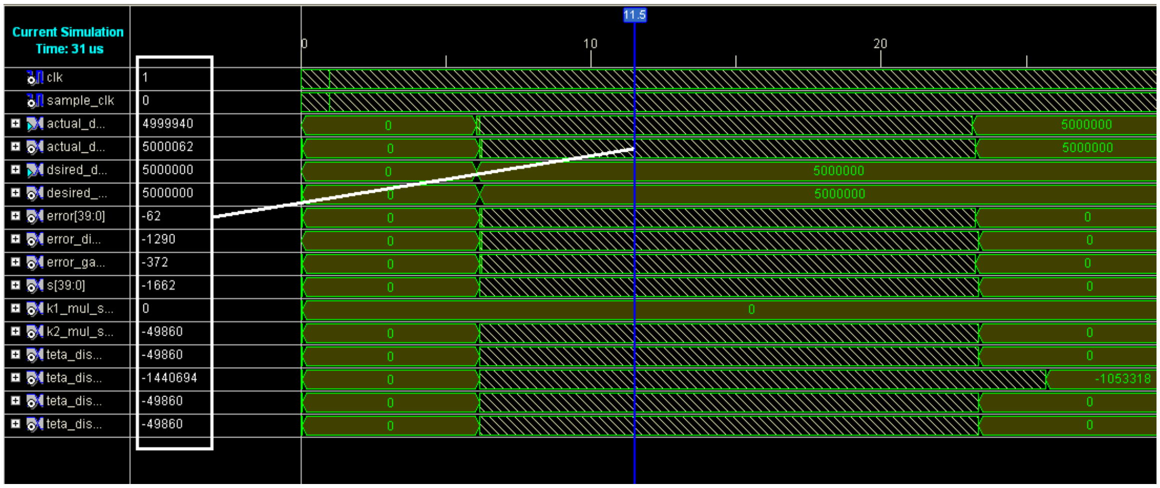


Figure 11. FPGA-based MTSMC for First, Second and Third Link for Desired and Actual Inputs, Error Performance, and Torque Performance in 11.5 μs.

As shown in Figure 11 at 11.5 μs (steady state response), the actual response is close to desired response therefore the FPGA-based MTSMC steady State error is 6×10^{-5}, the desired displacement is 5, the actual displacement is 5.000062 and the torque performance is 0.78 N.m.

6. Conclusion

Refer to this research, a position FPGA-based mathematical error-based tuning sliding mode controller (MTSMC) is proposed for PUMA robot manipulator. Pure sliding mode controller with saturation function and fuzzy sliding mode controller with saturation function have difficulty in handling unstructured model uncertainties. It is possible to solve this problem by combining fuzzy sliding mode controller and mathematical error-based tuning. Since the sliding surface gain (λ) is adjusted by mathematical error-based tuning method, it is nonlinear and continuous. The sliding surface slope updating factor (α) of mathematical error-based tuning part can be changed with the changes in error, change of error and the second change of error. Sliding surface gain is adapted on-line by sliding surface slope updating factor. In pure sliding mode controller and fuzzy sliding mode controller the sliding surface gain is chosen by trial and error, which means pure sliding mode controller and error-based fuzzy sliding mode 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. In mathematical error-based tuning sliding mode controller the sliding surface gain are updated on-line to compensate the system unstructured uncertainty. The stability and convergence of the mathematical error-based tuning sliding mode controller based on switching function is guarantee and proved by the Lyapunov method. The simulation results exhibit that the mathematical error-based tuning sliding mode controller works well in various situations. Based on theoretical and simulation results, it is observed that mathematical error-based tuning sliding mode controller is a model-based stable control for robot manipulator. It is a best solution to eliminate chattering phenomenon with saturation function in structure and unstructured uncertainties. To have the high speed processing, FPGA based mathematical error-based sliding mode controller is designed and implemented. As a timing report after simulation result in XILINX ISE 9.1, it is observed that this controller is able to make as a fast response at 15.716 ns clock period with 63.29 MHZ of a maximum frequency and 4.407 ns for minimum input arrival time after clock. From investigation and synthesis summary, 30.286 ns for maximum input arrival time after clock with 33.018 MHZ frequencies, this design has 15.716 ns delays for each controller to 46 logic elements and also the offset before CLOCK is 55.773 ns for 132 logic gates.

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,* vol. 52, no. 11, **(2007)**, pp. 2085-2102.

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

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

[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,* vol. 27, no. 2, **(2002)**, pp. 332-343.

[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,* vol. 22, no. 2, **(2002)**, pp. 212-222.

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

[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,* vol. 48, no. 1, **(2001)**, pp. 2-3.

[21] J. J. Slotine and S. Sastry, "Tracking control of non-linear systems using sliding surfaces, with application to robot manipulators†," *International Journal of Control,* vol. 38, no. 2, **(1983)**, pp. 465-492.

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

[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,* vol. 10, no. 1, **(2000)**, pp. 239-263.

[26] P. Kachroo and M. Tomizuka, "Chattering reduction and error convergence in the sliding-mode control of a class of nonlinear systems," *Automatic Control, IEEE Transactions on,* vol. 41, no. 7, **(2002)**, pp. 1063-1068.

[27] H. Elmali and N. Olgac, "Implementation of sliding mode control with perturbation estimation (SMCPE)," *Control Systems Technology, IEEE Transactions on,* vol. 4, no. 1, **(2002)**, pp. 79-85.

[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,* vol. 18, no. 4, **(2010)**, pp. 798-810.

[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] F. 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, vol. 2, no. 3, **(2011)**, pp. 205-220.

[32] F. 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), vol. 13, no. 5, **(2011)**, pp. 1085-1092.

[33] F. 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, vol. 2, no. 3, **(2011)**, pp. 146-156.

[34] F. Piltan , A. Zare, Nasri B. Sulaiman, M. H. Marhaban and R. Ramli, , “A Model Free Robust Sliding Surface Slope Adjustment in Sliding Mode Control for Robot Manipulator,” World Applied Science Journal, vol. 12, no. 12, **(2011)**, pp. 2330-2336.

[35] F. 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, vol. 12, no. 12, **(2011)**, pp. 2317-2329.

[36] F. 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, vol. 2, no. 4, **(2011)**, pp. 184-195.

[37] F. Piltan, Nasri Sulaiman, M. H. Marhaban and R. Ramli, “Design On-Line Tunable Gain Artificial Nonlinear Controller,” Journal of Advances In Computer Research, vol. 2, no. 4, **(2011)**, pp. 75-83.

[38] F. Piltan, N. Sulaiman, Payman Ferdosali, Iraj Assadi Talooki, “ Design Model Free Fuzzy Sliding Mode Control: Applied to Internal Combustion Engine,” International Journal of Engineering, vol. 5, no. 4, **(2011)**, pp. 302-312.

[39] F. 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 , vol. 1, no. 1, **(2011)**, pp. 115-123.

[40] F. 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, vol. 1, no. 1, **(2011)**, pp. 1-22.

[41] F. 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, vol. 2, no. 4, **(2011)**, pp. 232-244.

[42] F. 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, vol. 2, no. 5, **(2011)**, pp. 360-380,.

[43] F. 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, vol. 5, no. 5, **(2011)**, pp. 380-398.

[44] F. 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, vol. 5, no. 5, **(2011)**, pp. 399-418.

[45] F. 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, vol. 2, no. 5, **(2011)**, pp. 208-228.

[46] F. 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, vol. 2, no. 5, **(2011)**, pp. 401-425.

[47] F. 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 , vol. 2, no. 5, **(2011)**, pp. 283-297.

[48] F. 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, vol. 2, no. 5, **(2011)**, pp. 298-316.

[49] F. 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, vol. 2, no. 5, **(2011)**, pp. 317-343.

[50] F. Piltan, N. Sulaiman, S. Roosta, Atefeh Gavahian, Samira Soltani, “Evolutionary Design of Backstepping Artificial Sliding Mode Based Position Algorithm: Applied to Robot Manipulator,” International Journal of Engineering, vol. 5, no. 5, **(2011)**, pp. 419-434.

[51] F. 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 , vol. 4, no. 3, **(2011)**, pp. 65-76.

[52] F. 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, vol. 2, no. 5, **(2011)**, pp. 265-282.

[53] F. 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, vol. 4, no. 4, **(2011)**, pp. 91-110.

[54] F. 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, vol. 2, no. 5, **(2011)**, pp. 381-400.

[55] F. 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, vol. 5, no. 5, **(2011)**, 360-379.

[56] F. 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, vol. 3, no. 10, **(2011)**, pp. 7301-7308..

[57] F. 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), vol. 15, no. 3, **(2011)**, pp. 422-428.

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

[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,* vol. 1, no. 1, **(2006)**, pp. 23-29.

[60] Y. Guo and P. Y. Woo, "An adaptive fuzzy sliding mode controller for robotic manipulators," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* vol. 33, no. 2, **(2003)**, pp. 149-159.

[61] C. M. Lin and C. F. Hsu, "Adaptive fuzzy sliding-mode control for induction servomotor systems," *Energy Conversion, IEEE Transactions on,* vol. 19, no. 2, **(2004)**, pp. 362-368.

[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,* vol. 3, no. 4, **(2009)**, pp. 3575-3596.

[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,* **(2006)**, pp. 1163–1168.

[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,* vol. 3, no. 3, **(2009)**, pp. 2724-2740.

[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, vol. 152, no. 5, **(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,* vol. 18, no. 1, **(2003)**, pp. 344-355.

[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, vol. 2, no. 3, **(2011)**, pp. 183-204.

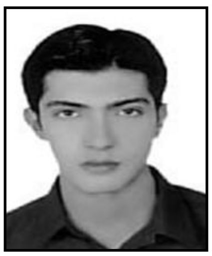
[83] I. Eksin, M. Guzelkaya and S. Tokat, "Sliding surface slope adjustment in fuzzy sliding mode controller," *Mediterranean Conference,* **(2002)**, pp. 160-168.

Authors

Farzin Piltan is research head manager of Sanaat Kade Sabz Pasargad (SSP. Co) research center, IRAN, which this company persuades to research and development of innovative technology through excellence in education, research and development. His current research is interested including nonlinear control, artificial control system and applied nonlinear controller in FPGA. He also has more than 30 publications in 2011.

Iman Nazari is an expert control and automation engineer of SSP Co. research center. His current research interests including control & automation and robotics.

Sobhan Siamak is an expert artificial intelligence engineer of SSP Co. research center. His current research interests including artificial intelligence, nonlinear control and optimization.



Payman Ferdosali is an expert control and automation engineer of SSP Co. research center. His current research interests including real time systems, on-line tuning, Control and optimization.