Design New Nonlinear Controller with Parallel Fuzzy Inference System Compensator to Control of Continuum Robot Manipulator

Sara Heidari[1,2], Farzin Piltan[2], Mohammad Shamsodini[2], Kamran Heidari[2] and Samaneh Zahmatkesh[2]

[1]*Department of Mathematics, Islamic Azad University, Shiraz, Iran*
[2]*Senior Researcher at Research and Development Unit, Sanatkadehe Sabze Pasargad Company, (SSP. Co), Shiraz, Iran*

SSP.ROBOTIC@iranssp.com; WWW.IRANSSP.COM

Abstract

Sliding Mode controller (SMC) is a highly significant nonlinear controller under condition of certain and partly uncertain dynamic parameters of system. This controller is used to control of highly nonlinear systems especially for continuum robot manipulators, because this controller is a robust and stable. In opposition, conventional sliding mode controller is used in many applications; it has two important drawbacks; chattering phenomenon and nonlinear equivalent dynamic formulation in uncertain dynamic parameter. To reduce or eliminate the chattering two methods is used; modified proportional-derivative boundary derivative sliding function and design a nonlinear sliding surface function based on fuzzy inference system and applied to sliding mode controller. The nonlinear equivalent dynamic formulation problem in uncertain system can be solved by applied artificial intelligence theorem (e.g., fuzzy inference system). Based on this research chattering eliminated based on applied modified sliding function and fuzzy inference engine and this controller is more robust than conventional sliding mode controller and simple fuzzy sliding mode controller.

Keywords*: Sliding mode controller, continuum Robot manipulator, Fuzzy inference system, fuzzy sliding mode controller, 'modified sliding function*

1. Introduction

Continuum robots represent a class of robots that have a biologically inspired form characterized by flexible backbones and high degrees-of-freedom structures [1]. The idea of creating “trunk and tentacle” robots, (in recent years termed continuum robots [1]), is not new [2]. Inspired by the bodies of animals such as snakes [3], the arms of octopi [4], and the trunks of elephants [5-6], researchers have been building prototypes for many years. A key motivation in this research has been to reproduce in robots some of the special qualities of the biological counterparts. This includes the ability to “slither” into tight and congested spaces and (of particular interest in this work) the ability to grasp and manipulate a wide range of objects, via the use of “whole arm manipulation”, *i.e.*, wrapping their bodies around objects, conforming to their shape profiles. Hence, these robots have potential applications in whole arm grasping and manipulation in unstructured environments such as rescue operations. Theoretically, the compliant nature of a continuum robot provides infinite degrees of freedom to these devices. However, there is a limitation set by the practical inability to incorporate infinite actuators in the device. Most of these robots are consequently underactuated (in terms of numbers of independent actuators) with respect to their anticipated tasks. In other words they must achieve a wide range of configurations with relatively few control inputs. This is partly due to the desire to keep the body structures (which, unlike in conventional rigid-link manipulators or fingers, are required to directly contact the environment) "clean and soft", but also to exploit the extra control authority available due to the continuum contact conditions with a minimum number of actuators. For example, the Octarm VI continuum manipulator, discussed frequently in this paper, has nine independent actuated degrees-of-freedom with only three sections. Continuum manipulators differ fundamentally from rigid-link and hyper-redundant robots by having an unconventional structure that lacks links and joints. Hence, standard techniques like the Denavit-Hartenberg (D-H) algorithm cannot be directly applied for developing continuum arm kinematics. Moreover, the design of each continuum arm varies with respect to the flexible backbone present in the system, the positioning, type and number of actuators. The constraints imposed by these factors make the set of reachable configurations and nature of movements unique to every continuum robot. This makes it difficult to formulate generalized kinematic or dynamic models for continuum robot hardware. Chirikjian and Burdick were the first to introduce a method for modeling the kinematics of a continuum structure by representing the curve-shaping function using modal functions [6]. Mochiyama used the Serret- Frenet formulae to develop kinematics of hyper-degrees of freedom continuum manipulators [5]. For details on the previously developed and more manipulator-specific kinematics of the Rice/Clemson "Elephant trunk" manipulator, see [1-5]. For the Air Octor and Octarm continuum robots, more general forward and inverse kinematics have been developed by incorporating the transformations of each section of the manipulator (using D-H parameters of an equivalent virtual rigid link robot) and expressing those in terms of the continuum manipulator section parameters [4]. The net result of the work in [3-6] is the establishment of a general set of kinematic algorithms for continuum robots. Thus, the kinematics (*i.e.*, geometry based modeling) of a quite general set of prototypes of continuum manipulators has been developed and basic control strategies now exist based on these. The development of analytical models to analyze continuum arm dynamics (*i.e.*, physicsbased models involving forces in addition to geometry) is an active, ongoing research topic in this field. From a practical perspective, the modeling approaches currently available in the literature prove to be very complicated and a dynamic model which could be conveniently implemented in an actual device's real-time controller has not been developed yet. The absence of a computationally tractable dynamic model for these robots also prevents the study of interaction of external forces and the impact of collisions on these continuum structures. This impedes the study and ultimate usage of continuum robots in various practical applications like grasping and manipulation, where impulsive dynamics [1-4] are important factors. Although continuum robotics is an interesting subclass of robotics with promising applications for the future, from the current state of the literature, this field is still in its stages of inception.

In modern usage, the word of control has many meanings, this word is usually taken to mean regulate, direct or command. The word feedback plays a vital role in the advance engineering and science. The conceptual frame work in Feed-back theory has developed only since world war II. In the twentieth century, there was a rapid growth in the application of feedback controllers in process industries. According to Ogata, to do the first significant work in three-term or PID controllers which Nicholas Minorsky worked on it by automatic controllers in 1922. In 1934, Stefen Black was invention of the feedback amplifiers to develop the negative feedback amplifier [3]. Negative feedback invited communications engineer Harold Black in 1928 and it occurs when the output is subtracted from the input. Automatic control has played an important role in advance science and engineering and its extreme importance in many industrial applications, i.e., aerospace, mechanical engineering and robotic systems. The first significant work in automatic control was James Watt's centrifugal governor for the speed control in motor engine in eighteenth century [7-14]. There are several methods for controlling a robot manipulator, which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes [13-23]. 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 [24-30]. 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 [24-36]. One of the simplest ways to analysis control of multiple DOF robot manipulators are analyzed each joint separately such as SISO systems and design an independent joint controller for each joint. In this methodology, the coupling effects between the joints are modeled as disturbance inputs. To make this controller, the inputs are modeled as: total velocity/displacement and disturbance. Design a controller with the same formulation and different coefficient, low cost hardware and simple structure controller are some of most important independent-joint space controller advantages. Nonlinear controllers divided into six groups, namely, feedback linearization (computed-torque control), passivity-based control, sliding mode control (variable structure control), artificial intelligence control, Lyapunov-based control and adaptive control [1, 6].

Sliding mode controller (SMC) is a powerful nonlinear controller which has been analyzed by many researchers especially in recent years. This theory was first proposed in the early 1950 by Emelyanov and several co-workers and has been extensively developed since then with the invention of high speed control devices [2]. The main reason to opt for this controller is its acceptable control performance in wide range and solves two most important challenging topics in control which names, stability and robustness [7, 17-37]. 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 can causes some problems such as saturation and heats 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 recent years, artificial intelligence theory has been used in nonlinear control systems. Neural network, fuzzy logic and neuro-fuzzy are synergically combined with nonlinear classical controller and used in nonlinear, time variant and uncertain plant (*e.g.*, continuum robot manipulator).

Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control nonlinear, uncertain, and noisy systems. This method is free of some model techniques as in model-based controllers. As mentioned that fuzzy logic application is not only limited to the modelling of nonlinear systems [31-36] but also this method can help engineers to design a model-free controller. Control robot arm manipulators using model-based controllers are based on manipulator dynamic model. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of robot manipulator, but most of time these models are MIMO, nonlinear and partly uncertain therefore calculate accurate dynamic model is complicated [32]. The main reasons to use fuzzy logic methodology are able to give approximate recommended solution for uncertain and also certain complicated systems to easy understanding and flexible. Fuzzy logic provides a method to design a model-free controller for nonlinear plant with a set of IF-THEN rules [32-35].

This method is based on design modified sliding surface function switching sliding mode controller and resolves the uncertainty term and chattering phenomenon by fuzzy logic methodology in presence of modified switching sliding surface function. To have the best performance and reduce the chattering in presence of high speed modified PD method based on boundary derivative methodology is design. The first time coefficients are calculated by Gradient Descent Optimization. To compensate the continuum robot manipulator system's dynamic, 7 rules parallel Mamdani fuzzy inference system is design and applied to modified sliding mode methodology with switching highly stable function.

This paper is organized as follows; second part focuses on the modeling dynamic formulation based on Lagrange methodology, fuzzy logic methodology, Gradient Descent Optimization and sliding mode controller to have a robust control. Third part is focused on the methodology which can be used to reduce the error, increase the performance quality and increase the robustness and stability. Simulation result and discussion is illustrated in forth part which based on trajectory following and disturbance rejection. The last part focuses on the conclusion and compare between this method and the other ones.

2. Theory

The Continuum section analytical model developed here consists of three modules stacked together in series. In general, the model will be a more precise replication of the behavior of a continuum arm with a greater of modules included in series. However, we will show that three modules effectively represent the dynamic behavior of the hardware, so more complex models are not motivated. Thus, the constant curvature bend exhibited by the section is incorporated inherently within the model. The model resulting from the application of Lagrange's equations of motion obtained for this system can be represented in the form

$$\boldsymbol{F_{coeff}}\,\underline{\boldsymbol{\tau}} = \boldsymbol{D}\left(\underline{\boldsymbol{q}}\right)\underline{\ddot{\boldsymbol{q}}} + \boldsymbol{C}\left(\underline{\boldsymbol{q}}\right)\underline{\dot{\boldsymbol{q}}} + \boldsymbol{G}\left(\underline{\boldsymbol{q}}\right) \tag{1}$$

where τ is a vector of input forces and q is a vector of generalized co-ordinates. The force coefficient matrix F_{coeff} transforms the input forces to the generalized forces and torques in the system. The inertia matrix, D is composed of four block matrices. The block matrices that correspond to pure linear accelerations and pure angular accelerations in the system (on the top left and on the bottom right) are symmetric. The matrix C contains coefficients of the first order derivatives of the generalized co-ordinates. Since the system is nonlinear, many elements of C contain first order derivatives of the generalized co-ordinates. The remaining terms in the dynamic equations resulting from gravitational potential energies and spring energies are collected in the matrix G. The coefficient matrices of the dynamic equations are given below,

$$\boldsymbol{Fcoeff} = \begin{bmatrix} 1 & 1 & cos(\theta_1) & cos(\theta_1) & cos(\theta_1+\theta_2) & cos(\theta_1+\theta_2) \\ 0 & 0 & 1 & 1 & cos(\theta_2) & cos(\theta_2) \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1/2 & -1/2 & 1/2 & -1/2 & 1/2+s_2sin(\theta_2) & -1/2+s_2sin(\theta_2) \\ 0 & 0 & 1/2 & -1/2 & 1/2 & -1/2 \\ 0 & 0 & 0 & 0 & 1/2 & -1/2 \end{bmatrix} \quad (2)$$

$$D(\underline{q}) = \begin{bmatrix} m_1+m_2+m_3 & m_2cos(\theta_1)+m_3cos(\theta_1) & m_3cos(\theta_1+\theta_2) & -m_2s_2sin(\theta_1)-m_3s_2sin(\theta_1)-m_3s_3sin(\theta_1+\theta_2) & -m_3s_3sin(\theta_1+\theta_2) & 0 \\ m_2cos(\theta_1)+m_3cos(\theta_1) & m_2+m_3 & m_3cos(\theta_2) & -m_3s_3sin(\theta_2) & -m_3s_3sin(\theta_2) & 0 \\ m_3cos(\theta_1+\theta_2) & m_3cos(\theta_2) & m_3 & m_3s_3sin(\theta_2) & 0 & 0 \\ -m_2s_2sin(\theta_1)-m_3s_2sin(\theta_1)-m_3s_3sin(\theta_1+\theta_2) & -m_3s_3sin(\theta_2) & m_3s_2sin(\theta_2) & m_2s_2^2+I_1+I_2+I_3+m_3s_2^2+m_3s_3^2+2m_3s_3cos(\theta_2)s_2 & I_2+m_3s_3^2+I_3+m_3s_3cos(\theta_2)s_2 & I_3 \\ -m_3s_3sin(\theta_1+\theta_2) & -m_3s_3sin(\theta_2) & 0 & I_2+m_3s_3^2+I_3+m_3s_3cos(\theta_2)s_2I & I_2+m_3s_3^2+I_3 & I_3 \\ 0 & 0 & 0 & I_3 & I_3 & I_3 \end{bmatrix} \quad (3)$$

$$C(\underline{q}) = \begin{bmatrix} c_{11}+c_{21} & -2m_2sin(\theta_1)\dot{\theta}_1-2m_3sin(\theta_1)\dot{\theta}_1 & -2m_3sin(\theta_1+\theta_2)(\dot{\theta}_1+\dot{\theta}_2) & -m_2s_2cos(\theta_1)(\dot{\theta}_1)+(1/2)(c_{11}+c_{21})-m_3s_2cos(\theta_1)(\dot{\theta}_1)-m_3s_3cos(\theta_1+\theta_2)(\dot{\theta}_1) & -m_3s_3sin(\theta_1+\theta_2) & 0 \\ 0 & c_{12}+c_{22} & -2m_3sin(\theta_2)(\dot{\theta}_1+\dot{\theta}_2) & -m_3s_3(\dot{\theta}_1)+(1/2)(c_{12}+c_{22})-m_3s_2(\dot{\theta}_1)-m_3s_3cos(\theta_2)(\dot{\theta}_1) & -2m_3s_3cos(\theta_2)(\dot{\theta}_1)-m_3s_3cos(\theta_2)(\dot{\theta}_2) & 0 \\ 0 & 2m_3sin(\theta_2)(\dot{\theta}_1) & c_{13}+c_{23} & -m_3s_3s_2cos(\theta_2)(\dot{\theta}_1)-m_3s_3(\dot{\theta}_1) & -2m_3s_3(\dot{\theta}_1)-m_3s_3(\dot{\theta}_2) & (1/2)(c_{13}+c_{23}) \\ (1/2)(c_{11}+c_{21}) & 2m_3s_3cos(\theta_2)(\dot{\theta}_1)-2m_3s_2(\dot{\theta}_1)+2m_2s_2(\dot{\theta}_1) & 2m_3s_3(\dot{\theta}_1+\dot{\theta}_2)-2m_3s_2cos(\theta_2)(\dot{\theta}_1+\dot{\theta}_2) & 2m_3s_3s_2sin(\theta_2)(\dot{\theta}_2)+(1^2/4)(c_{11}+c_{21}) & m_3s_3s_2sin(\theta_2)(\dot{\theta}_2) & 0 \\ 0 & (1/2)(c_{12}+c_{22})+2m_3s_3cos(\theta_2)(\dot{\theta}_1) & 2m_3s_3(\dot{\theta}_1+\dot{\theta}_2) & m_3s_3s_2sin(\theta_2)(\dot{\theta}_1) & (1^2/4)(c_{12}+c_{22}) & 0 \\ 0 & 0 & (1/2)(c_{13}-c_{23}) & 0 & 0 & (1^2/4)(c_{13}+c_{23}) \end{bmatrix} \quad (4)$$

$$G\left(\underline{q}\right) = \begin{bmatrix} -m_1 g - m_2 g + k_{11}(s_1 + (1/2)\theta_1 - s_{01}) + k_{21}(s_1 - (1/2)\theta_1 - s_{01}) - m_3 g \\ -m_2 g cos(\theta_1) + k_{12}(s_2 + (1/2)\theta_2 - s_{02}) + k_{22}(s_2 - (1/2)\theta_2 - s_{02}) - m_3 g cos(\theta_1) \\ -m_3 g cos(\theta_1 + \theta_2) + k_{13}(s_3 + (1/2)\theta_3 - s_{03}) + k_{23}(s_3 - (1/2)\theta_3 - s_{03}) \\ m_2 s_2 g sin(\theta_1) + m_3 s_3 g sin(\theta_1 + \theta_2) + m_3 s_2 g sin(\theta_1) + k_{11}(s_1 + (1/2)\theta_1 - s_{01})(1/2) \\ +k_{21}(s_1 - (1/2)\theta_1 - s_{01})(-1/2) \\ m_3 s_3 g sin(\theta_1 + \theta_2) + k_{12}(s_2 + (1/2)\theta_2 - s_{02})(1/2) + k_{22}(s_2 - (1/2)\theta_2 - s_{02})(-1/2) \\ k_{13}(s_3 + (1/2)\theta_3 - s_{03})(1/2) + k_{23}(s_3 - (1/2)\theta_3 - s_{03})(-1/2) \end{bmatrix} \quad (5)$$

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

$$\boldsymbol{x^{(n)} = f(\vec{x}) + b(\vec{x})u} \quad (6)$$

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]:

$$\boldsymbol{\widetilde{x} = x - x_d = [\widetilde{x}, \dots, \widetilde{x}^{(n-1)}]^T} \quad (7)$$

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

$$\boldsymbol{s(x,t) = (\frac{d}{dt} + \lambda)^{n-1}\, \widetilde{x} = 0} \quad (8)$$

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

$$\boldsymbol{s(x,t)} = (\frac{d}{dt} + \boldsymbol{\lambda})^{\boldsymbol{n-1}} \left(\int_{\boldsymbol{0}}^{\boldsymbol{t}} \boldsymbol{\widetilde{x}\, dt}\right) = \boldsymbol{0} \quad (9)$$

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].

$$\boldsymbol{\frac{1}{2}\frac{d}{dt} s^2(x,t) \le -\zeta |s(x,t)|} \quad (10)$$

where ζ is positive constant.

$$\text{If } \mathbf{S(0)>0} \rightarrow \mathbf{\frac{d}{dt} S(t) \le -\boldsymbol{\zeta}} \quad (11)$$

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

$$\int_{t=0}^{t=t_{reach}} \frac{d}{dt} S(t) \le -\int_{t=0}^{t=t_{reach}} \eta \rightarrow S\,(t_{reach}) - S(0) \le -\zeta(t_{reach} - 0) \qquad \textbf{(12)}$$

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

$$0 - S(0) \le -\eta(t_{reach}) \rightarrow t_{reach} \le \frac{S(0)}{\zeta} \qquad \textbf{(13)}$$

and

$$if\ S(0) < 0 \rightarrow 0 - S(0) \le -\eta(t_{reach}) \rightarrow S(0) \le -\zeta(t_{reach}) \rightarrow t_{reach} \le \frac{|S(0)|}{\eta} \qquad \textbf{(14)}$$

Equation (14) guarantees time to reach the sliding surface is smaller than $\frac{|S(0)|}{\zeta}$ since the trajectories are outside of $S(t)$.

$$if\ S_{t_{reach}} = S(0) \rightarrow error(x - x_d) = 0 \qquad \textbf{(15)}$$

suppose S is defined as

$$s(x,t) = (\frac{d}{dt} + \lambda) \quad \widetilde{x} = (\dot{x} - \dot{x}_d) + \lambda(x - x_d) \qquad \textbf{(16)}$$

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

$$\dot{S} = (\ddot{x} - \ddot{x}_d) + \lambda(\dot{x} - \dot{x}_d) \qquad \textbf{(17)}$$

suppose the second order system is defined as;

$$\ddot{x} = f + u \ \rightarrow \dot{S} = f + U - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \qquad \textbf{(18)}$$

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

$$\widehat{U} = -\hat{f} + \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) \qquad \textbf{(19)}$$

A simple solution to get the sliding condition when the dynamic parameters have uncertainty is the switching control law [52-53]:

$$U_{dis} = \widehat{U} - K(\vec{x}, t) \cdot \mathbf{sgn}(s) \qquad \textbf{(20)}$$

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

$$sgn(s) = \begin{cases} 1 & s > 0 \\ -1 & s < 0 \\ 0 & s = 0 \end{cases} \qquad \textbf{(21)}$$

and the $\boldsymbol{K(\vec{x},t)}$ is the positive constant. Suppose by (22) the following equation can be written as,

$$\frac{1}{2}\frac{d}{dt}s^2(x,t) = \dot{S}\cdot S = \left[f - \hat{f} - K\mathrm{sgn}(s)\right]\cdot S = \left(f - \hat{f}\right)\cdot S - K|S| \quad (22)$$

and if the equation (14) instead of (13) the sliding surface can be calculated as

$$s(x,t) = \left(\frac{d}{dt} + \lambda\right)^2 \left(\int_0^t \tilde{x}\, dt\right) = (\dot{x} - \dot{x}_d) + 2\lambda(\dot{x} - \dot{x}_d) - \lambda^2(x - x_d) \quad (23)$$

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

$$\hat{U} = -\hat{f} + \ddot{x}_d - 2\lambda(\dot{x} - \dot{x}_d) + \lambda^2(x - x_d) \quad (24)$$

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

$$\tau = \tau_{eq} + \tau_{dis} \quad (25)$$

Where, the model-based component τ_{eq} is the nominal dynamics of systems calculated as follows [1]:

$$\tau_{eq} = \left[D^{-1}(f + C + G) + \dot{S}\right]D \quad (26)$$

and τ_{dis} is computed as [1];

$$\tau_{dis} = K\cdot \mathrm{sgn}(S) \quad (27)$$

By (27) and (26) the sliding mode control of robot manipulator is calculated as;

$$\tau = \left[D^{-1}(f + C + G) + \dot{S}\right]D + K\cdot \mathrm{sgn}(S) \quad (28)$$

where $S = \lambda e + \dot{e}$ in PD-SMC.

Linear Controller: In the absence of robot knowledge, proportional-integral-derivative (PID), proportional-integral (PI) and proportional -derivative (PD) may be the best controllers, because they are model-free, and they're parameters can be adjusted easily and separately [1] and it is the most used in continuum robot manipulators. In order to remove steady-state error caused by uncertainties and noise, the integrator gain has to be increased. This leads to worse transient performance, even destroys the stability. The integrator in a PID controller also reduces the bandwidth of the closed-loop system. PD control guarantees stability only when the PD gains tend to infinity, the tracking error does not tend to zero when friction and gravity forces are included in the continuum robot manipulator dynamics [2]. Model-based compensation for PD control is an alternative method to substitute PID control [1], such as adaptive gravity compensation [3], desired gravity compensation [2], and PD+ with position measurement [4]. They all needed structure information of the robot gravity. Some nonlinear PD controllers can also achieve asymptotic stability, for example PD control with time-varying gains [5], PD control with nonlinear gains [6], and PD control with feedback linearization compensation [8]. But these controllers are complex; many good properties of the linear PID control do not exist because these controllers do not have the same form as the industrial PID. Design of a linear methodology to control of continuum robot manipulator was very straight forward. Since there was an output from the torque model, this means that there would be two inputs into the PID controller. Similarly, the outputs of the controller result from the two control inputs of the torque signal. In a typical PID method, the controller corrects the error between the desired input value and the measured value. Since the actual position is the measured signal [9-16].

$$e(t) = \theta_a(t) - \theta_d(t) \quad \textbf{(29)}$$

$$U_{PD} = K_{p_a} e + K_{V_a} \dot{e} \quad \textbf{(30)}$$

The model-free control strategy is based on the assumption that the joints of the manipulators are all independent and the system can be decoupled into a group of single-axis control systems [14-16]. Therefore, the kinematic control method always results in a group of individual controllers, each for an active joint of the manipulator. With the independent joint assumption, no a priori knowledge of robot manipulator dynamics is needed in the kinematic controller design, so the complex computation of its dynamics can be avoided and the controller design can be greatly simplified. This is suitable for real-time control applications when powerful processors, which can execute complex algorithms rapidly, are not accessible. However, since joints coupling is neglected, control performance degrades as operating speed increases and a manipulator controlled in this way is only appropriate for relatively slow motion [13-16]. The fast motion requirement results in even higher dynamic coupling between the various robot joints, which cannot be compensated for by a standard robot controller such as PID [16], and hence model-based control becomes the alternative.

Fuzzy Inference Engine: Based on foundation of fuzzy logic methodology; fuzzy logic management has played important rule to design nonlinear management for nonlinear and uncertain systems [16-36]. However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of; Input fuzzification (binary-to-fuzzy [B/F] conversion), Fuzzy rule base (knowledge base), Inference engine and Output defuzzification (fuzzy-to-binary [F/B] conversion). Figure 1 shows the fuzzy controller part. The fuzzy inference engine offers a mechanism for transferring the rule base in fuzzy set which it is divided into two most important methods, namely, Mamdani method and Sugeno method. Mamdani method is one of the common fuzzy inference systems and he designed one of the first fuzzy managements to control of system engine. Mamdani's fuzzy inference system is divided into four major steps: fuzzification, rule evaluation, aggregation of the rule outputs and defuzzification. Michio Sugeno uses a singleton as a membership function of the rule consequent part. The following definition shows the Mamdani and Sugeno fuzzy rule base [22-33]

$$\boldsymbol{if\ x\ is\ A\ and\ y\ is\ B\ then\ z\ is\ C\ 'mar}$$
$$\boldsymbol{if\ x\ is\ A\ and\ y\ is\ B\ then\ z\ is\ f(x,y)}$$

When x and y have crisp values fuzzification calculates the membership degrees for antecedent part. Rule evaluation focuses on fuzzy operation (AND/OR) in the antecedent of the fuzzy rules. The aggregation is used to calculate the output fuzzy set and several methodologies can be used in fuzzy logic controller aggregation, namely, Max-Min aggregation, Sum-Min aggregation, Max-bounded product, Max-drastic product, Max-bounded sum, Max-algebraic sum and Min-max. Max-min aggregation defined as below

$$\boldsymbol{\mu_U(x_k,y_k,U) = \mu_{\bigcup_{i=1}^{r} FR^i}(x_k,y_k,U) = \max\left\{\min_{i=1}^{r}\left[\mu_{R_{pq}}(x_k,y_k),\mu_{p_m}(U)\right]\right\}} \tag{31}$$

The Sum-min aggregation defined as below

$$\boldsymbol{\mu_U(x_k,y_k,U) = \mu_{\bigcup_{i=1}^{r} FR^i}(x_k,y_k,U) = \sum \min_{i=1}^{r}\left[\mu_{R_{pq}}(x_k,y_k),\mu_{p_m}(U)\right]} \tag{32}$$

where r is the number of fuzzy rules activated by x_k and y_k and also $\mu_{\bigcup_{i=1}^{r} FR^i}(x_k,y_k,U)$ is a fuzzy interpretation of $i-th$ rule. Defuzzification is the last step in the fuzzy inference system which it is used to transform fuzzy set to crisp set. Consequently defuzzification's input is the aggregate output and the defuzzification's output is a crisp number. Centre of gravity method (COG) and Centre of area method (COA) are two most common defuzzification methods [34-36]. COG method used the following equation to calculate the defuzzification

$$\boldsymbol{COG(x_k,y_k) = \frac{\sum_i U_i \sum_{j=1}^{r} \cdot \mu_u(x_k,y_k,U_i)}{\sum_i \sum_{j=1}^{r} \cdot \mu_u(x_k,y_k,U_i)}} \tag{33}$$

and COA method used the following equation to calculate the defuzzification

$$\boldsymbol{COA(x_k,y_k) = \frac{\sum_i U_i \cdot \mu_u(x_k,y_k,U_i)}{\sum_i \mu_U \cdot (x_k,y_k,U_i)}} \tag{34}$$

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

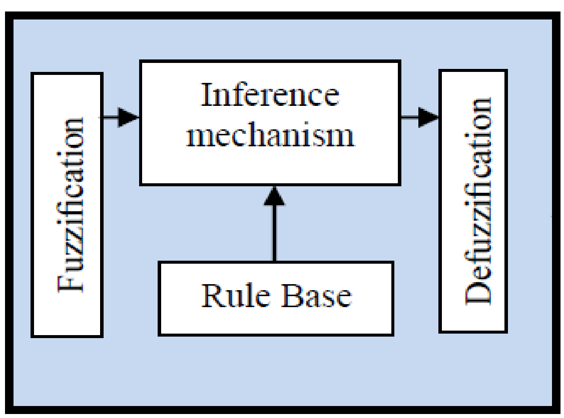


Figure 1. Fuzzy Controller Part

3. Methodology

Sliding mode controller (SMC) is a nonlinear robust controller. Conversely pure sliding mode controller is a high-quality nonlinear controller; it has two important problems; chattering phenomenon and nonlinear equivalent dynamic formulation in uncertain dynamic parameter. To reduce the chattering phenomenon and equivalent dynamic problems, three methods are used: 1. modified PD boundary derivative method; 2. Fuzzy logic methodology is used to design a nonlinear sliding surface function, 3. applied fuzzy logic theory to estimate a nonlinear dynamic part of continuum robot manipulator.

In a typical PD method, the controller corrects the error between the desired input value and the measured value. Since the actual position is the measured signal. The derivative part of PD methodology is worked based on change of error and the derivative coefficient. In this research the modified PD is used based on boundary derivative part.

$$\dot{e}(t) \triangleq \left(\frac{S}{\alpha S+\beta}\right) \times e(t) \tag{35}$$

$$U_{PD} = K_{p_a} e + K_{V_a} \dot{e} \tag{36}$$

This is suitable for real-time control applications when powerful processors, which can execute complex algorithms rapidly, are not accessible. The result of modified PD method shows the power of disturbance rejection in this methodology. Based on the modified formulation the partly linear sliding mode controller formulation is;

$$\tau_{M-dis} = K \cdot \mathbf{sgn}(\lambda e + \dot{e}) = K.sgn\left(\lambda e + \left(\frac{S}{\alpha S+\beta}\right) \times e(t)\right) \tag{37}$$

In the proposed method fuzzy rule base was designed to have a nonlinear sliding surface slope function. The fuzzy system can be defined as below

$$f(x) = \tau_{fuzzy} = \sum_{l=1}^{M} \theta^T \zeta(x) = \psi(e, \dot{e}) \tag{38}$$

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

$$\zeta^1(x) = \frac{\sum_i \mu_{(xi)} x_i}{\sum_i \mu_{(xi)}} \tag{39}$$

where $\theta = (\theta^1, \theta^2, \theta^3, \dots \dots, \theta^M)$ is adjustable parameter in (38) and $\mu_{(xi)}$ is membership function. Error base fuzzy controller can be defined as

$$\tau_{fuzzy} = \psi(e, \dot{e}) \tag{40}$$

To eliminate the chattering fuzzy inference system is used instead of saturation and/or switching function. Design a nonlinear sliding function has five steps:

1. **Determine inputs and outputs:** This controller has one input (S) and one output (α). The input is sliding function (S) and the output is coefficient which estimate the saturation function (α).

2. **Find membership function and linguistic variable:** The linguistic variables for sliding surface (S) are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and the linguistic variables to find the saturation coefficient (α) are; Large Left (LL), Medium Left (ML), Small Left (SL), Zero (Z), Small Right (SR), Medium Right (MR), Large Right (LR).

3. **Choice of shape of membership function:** In this work triangular membership function was selected.

4. **Design fuzzy rule table:** design the rule base of fuzzy logic controller can play important role to design best performance for proposed method, suppose that two fuzzy rules in this controller are

F.R[1]: ***IF S is Z, THEN*** α ***is Z.*** **(41)**
F.R[2]: ***IF S is (PB) THEN*** α ***is (LR).***

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

Table 1. Rule table for proposed method

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

The control strategy that deduced by Table1 are

- If sliding surface (S) is N.B, the control applied is N.B for moving S to S=0.
- If sliding surface (S) is Z, the control applied is Z for moving S to S=0.

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

$$COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^{r} \cdot \mu_u(x_k, y_k, U_i)} \quad (42)$$

$$if\ S = 0\ then - \ \dot{e} = \lambda e \quad (43)$$

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

$$\theta^* = \arg\min[Sup_{x\in U} \mid \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} \mid] \quad (44)$$

Where θ^* is the minimum error, $sup_{x\in U} | \sum_{l=1}^{M} \theta^T \zeta(x) - \tau_{equ} |$ is the minimum approximation error. Figure 2 is shown the fuzzy instead of saturation function.

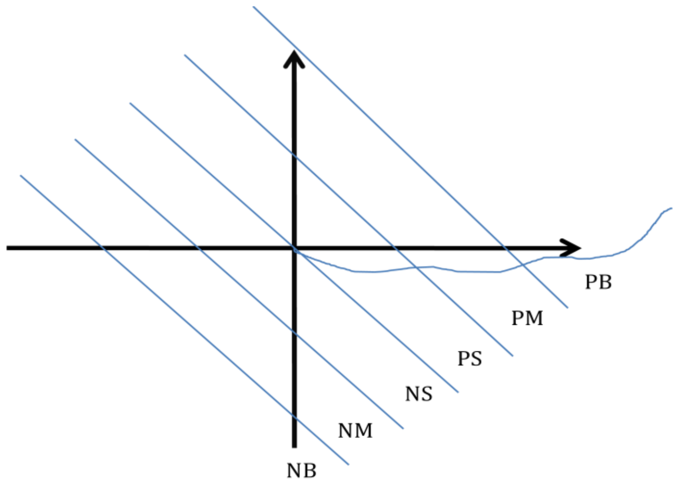


Figure 2. Nonlinear fuzzy inference system instead of saturation function

In proposed method fuzzy logic method is applied to equivalent part to estimate nonlinear dynamic formulation of continuum robot. In modified PD fuzzy error-based partly switching SMC; error based Mamdani's fuzzy inference system has considered with one input, one output and totally 7 rules to estimate the dynamic equivalent part. Based on above discussion, the control law for multi degrees of freedom continuum robot manipulator is written as:

$$\boldsymbol{\tau} = \boldsymbol{K}.\boldsymbol{sgn}\left(\lambda e + \left(\frac{\boldsymbol{S}}{\boldsymbol{0.1\,S+1}}\right) \times \boldsymbol{e}(\boldsymbol{t})\right) + \boldsymbol{\tau_{fuzzy}} \tag{45}$$

Based on fuzzy logic methodology

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{\tau_{fuzzy}} = \sum_{l=1}^{M} \boldsymbol{\theta^T}\,\boldsymbol{\zeta}(\boldsymbol{x}) \tag{46}$$

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

$$\boldsymbol{\zeta}(\boldsymbol{x}) = \frac{\sum_i \boldsymbol{\mu}(\boldsymbol{x_i})\boldsymbol{x_i}}{\sum_i \boldsymbol{\mu}(\boldsymbol{x_i})} \tag{47}$$

The design of error-based fuzzy estimator of 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. **Fuzzification**: the first step in fuzzification is determine inputs and outputs which, it has one input (τ_{Mdis}) and one output (τ_{fuzzy}). The second step is chosen an appropriate membership function for input and output which, to simplicity in implementation triangular membership function is selected in this research. The third step is chose the correct labels for each fuzzy set which, in this research namely as linguistic variable. Based on experience knowledge the linguistic variables for input τ_{MPID} are; Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), Positive Big (PB), and based on experience knowledge 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 and the linguistic variables to find the output are;

Large Left (LL), Medium Left (ML), Small Left (SL), Zero (Z), Small Right (SR), Medium Right (MR), Large Right (LR) and it is quantized in to thirteen levels represented by: -85, -70.8, -56.7, -42.5, -28.3, -14.2, 0, 14.2, 28.3, 42.5, 56.7, 70.8, 85. **Fuzzy rule base and rule evaluation**: the first step in rule base and evaluation is to provide a least structured method to derive the fuzzy rule base which, expert experience and control engineering knowledge is used because this method is the least structure of the other one and the researcher derivation the fuzzy rule base from the knowledge of system operate and/or the classical controller. Design the rule base of fuzzy inference system can play important role to design the best performance of fuzzy proposed controller, that to calculate the fuzzy rule base the researcher is used to heuristic method which, it is based on the behavior of the control of robot manipulator.

The complete rule base for this controller is shown in Table 2. **Aggregation of the rule output (Fuzzy inference):** Max-Min aggregation is used in this work. **Defuzzification:** The last step to design fuzzy inference in proposed controller is defuzzification. This part is used to transform fuzzy set to crisp set, therefore the input for defuzzification is the aggregate output and the output of it is a crisp number. Center of Gravity method (COG) is used in this research.

Table 2. Design Rule Base of Fuzzy Inference System

τ_{Mdis}	NB	NM	NS	Z	PS	PM	PB
τ_{Fuzzy}	LL	ML	SL	Z	SR	MR	LR

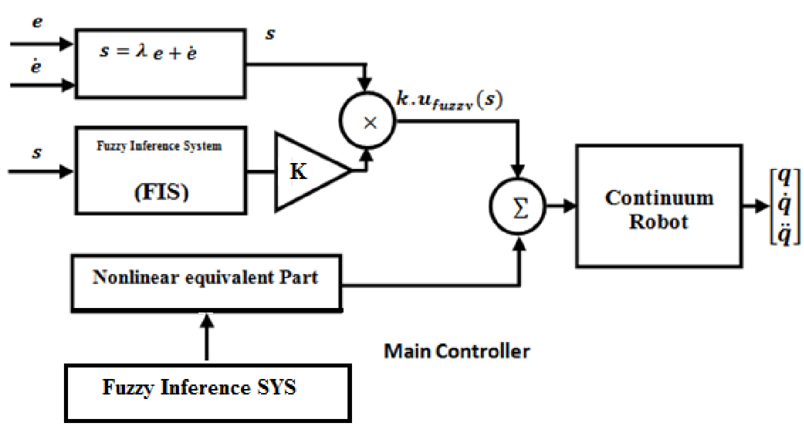


Figure 3: Block diagram of proposed artificial chattering free fuzzy sliding mode controller with minimum rule base in fuzzy equivalent part.

4. Results and Discussion

Classical sliding mode control (SMC) and proposed method are implemented in MATLAB/SIMULINK environment. Different sliding surface slope performance, tracking performance and robustness are compared.

- **Different sliding surface slope Vs different performance**

For various value of sliding surface slope (λ) in SMC and proposed method the trajectory performances shows in Figures 4 and 5.

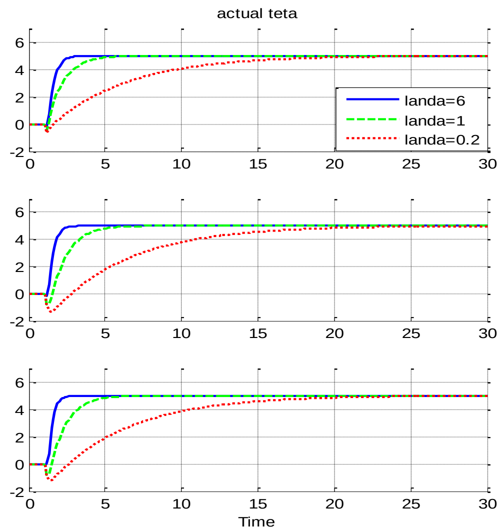


Figure 4. SMC trajectory performance, first; second and third link

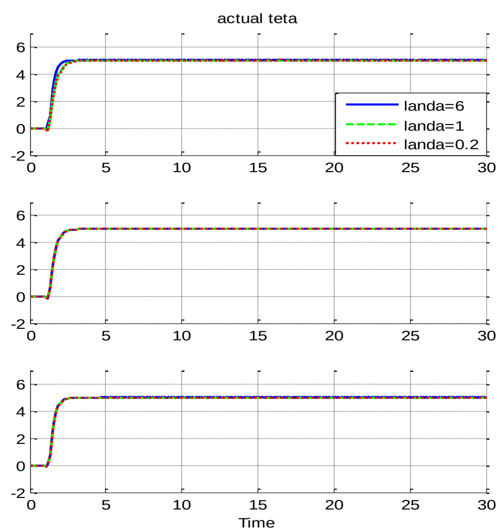


Figure 5. proposed method trajectory performance, first; second and third link

Figures 4 and 5 are shown trajectory performance with different sliding function for, sliding mode controller and proposed method. It is shown that the sensitivity in proposed method to sliding function is lower than SMC.

- **Tracking performances:**

From the simulation for first, second, and third trajectory without any disturbance, it can be seen that proposed method and classical SMC have same performance. This is primarily due to the constant parameters in simulation. Figure 6 shows tracking performance in certain system for SMC and proposed method.

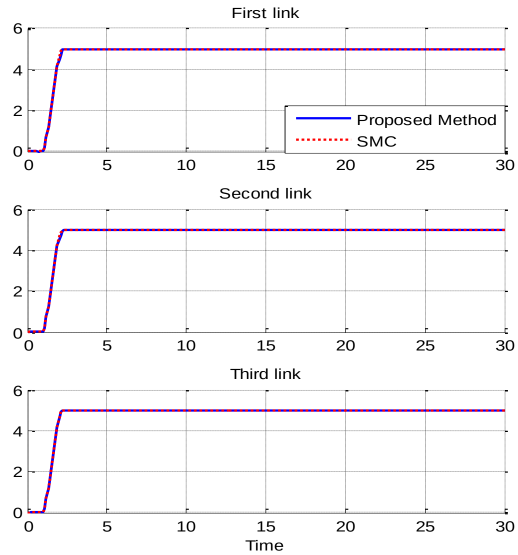


Figure 6. Trajectory performance: proposed method and SMC (first; second and third link)

- **Disturbance Rejection:**

A band limited white noise with predefined of 30% the power of input signal is applied to the response. Figure 7 shows disturbance rejection for proposed method and SMC.

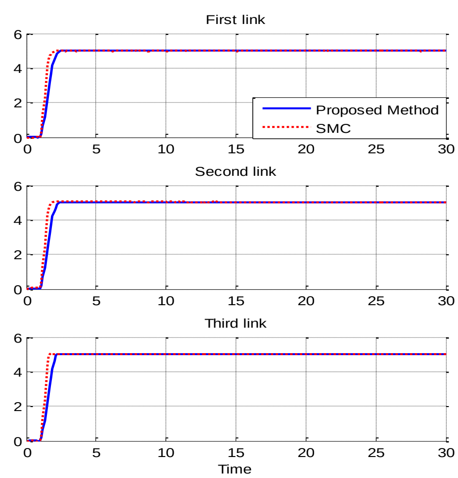


Figure 7. Disturbance rejection: proposed method and SMC (first; second and third link)

5. Conclusion

Refer to the research, a 7 rules Mamdani's new on-line chattering free fuzzy sliding mode controller and this suitability for use in the control of continuum robot manipulator has proposed in order to design high performance nonlinear controller in the presence of uncertainties and external disturbances. Sliding mode control methodology is selected as a frame work to construct the control law and address the stability and robustness of the close-loop system. The proposed approach effectively combines the design techniques from sliding mode control and fuzzy logic to improve the performance (*e.g.*, trajectory, disturbance rejection, error and chattering) and enhance the robustness property of the controller. Each method by adding to the previous controller has covered negative points. The system performance in sliding mode controller is sensitive to the sliding function. Therefore, compute the optimum value of sliding function for a system is the important which this problem has solved by adjusting surface slope of the sliding function continuously in real-time. The chattering phenomenon is estimated by fuzzy method when estimate the saturation/switching function with 7 rule base and modified PD sliding surface function. In this way, the overall system performance has improved with respect to the classical sliding mode controller. This controller solved chattering phenomenon as well as mathematical nonlinear equivalent part by applied modified fuzzy inference system method in new sliding mode controller.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their careful reading of this paper and for their helpful comments. This work was supported by the SSP Research and Development Corporation Program of Iran under grant no. 2012-Persian Gulf-4D.

References

[1] G. Robinson and J. Davies, "Continuum robots – a state of the art", Proc. IEEE International Conference on Robotics and Automation, Detroit, MI, vol. 4, **(1999)**, pp. 2849-2854.

[2] I. D. Walker, D. Dawson, T. Flash, F. Grasso, R. Hanlon, B. Hochner, W. M. Kier, C. Pagano, C. D. Rahn and Q. Zhang, "Continuum Robot Arms Inspired by Cephalopods", Proceedings SPIE Conference on Unmanned Ground Vehicle Technology VII, Orlando, FL, **(2005)**, pp. 303-314.

[3] K. Suzumori, S. Iikura and H. Tanaka, "Development of Flexible Microactuator and it's Applications to Robotic Mechanisms", Proceedings IEEE International Conference on Robotics and Automation, Sacramento, California, **(1991)**, pp. 1622-1627.

[4] D. Trivedi, C. D. Rahn, W. M. Kier and I. D. Walker, "Soft Robotics: Biological Inspiration, State of the Art, and Future Research", Applied Bionics and Biomechanics, vol. 5, no. 2, **(2008)**, pp. 99-117.

[5] W. McMahan, M. Pritts, V. Chitrakaran, D. Dienno, M. Grissom, B. Jones, M. Csencsits, C. D. Rahn, D. Dawson and I. D. Walker, "Field Trials and Testing of "OCTARM" Continuum Robots", Proc. IEEE International Conference on Robotics and Automation, **(2006)**, pp. 2336-2341.

[6] W. McMahan and I. D. Walker, "Octopus-Inspired Grasp Synergies for Continuum Manipulators", Proc. IEEE International Conference on Robotics and Biomimetics, **(2009)**, pp. 945- 950.

[7] F. Piltan and S. T. Haghighi, "Design Gradient Descent Optimal Sliding Mode Control of Continuum Robots", IAES International journal of Robotics and Automation (IAES-IJRA), vol. 1, no. 4, **(2012)**, pp. 175-189.

[8] M. Bazregar, F. Piltan, A. Nabaee and M. M. Ebrahimi, "Parallel Soft Computing Control Optimization Algorithm for Uncertainty Dynamic Systems", International Journal of Advanced Science and Technology, vol. 51, **(2013)**.

[9] F. Piltan, M. H. Yarmahmoudi, M. Mirzaei, S. Emamzadeh and Z. Hivand, "Design Novel Fuzzy Robust Feedback Linearization Control with Application to Robot Manipulator", International Journal of Intelligent Systems and Applications, vol. 5, no. 5, **(2013)**.

[10] S. T. Haghighi, S. Soltani, F. Piltan, M. kamgari and S. Zare, "Evaluation Performance of IC Engine: Linear Tunable Gain Computed Torque Controller Vs. Sliding Mode Controller", International Journal of Intelligent Systems and Applications, vol. 5, no. 6, **(2013)**.

[11] A. Jalali, F. Piltan, M. Keshtgar and M. Jalali, "Colonial Competitive Optimization Sliding Mode Controller with Application to Robot Manipulator", International Journal of Intelligent Systems and Applications, vol. 5, no. 7, **(2013)**.

[12] A. Salehi, F. Piltan, M. Mousavi, A. Khajeh and M. R. Rashidian, "Intelligent Robust Feed-forward Fuzzy Feedback Linearization Estimation of PID Control with Application to Continuum Robot", International Journal of Information Engineering and Electronic Business, vol. 5, no. 1, **(2013)**.

[13] F. Piltan, M. J. Rafaati, F. Khazaeni, A. Hosainpour and S. Soltani, "A Design High Impact Lyapunov Fuzzy PD-Plus-Gravity Controller with Application to Rigid Manipulator", International Journal of Information Engineering and Electronic Business, vol. 5, no. 1, **(2013)**.

[14] A. Jalali, F. Piltan, A. Gavahian, M. Jalali and M. Adibi, "Model-Free Adaptive Fuzzy Sliding Mode Controller Optimized by Particle Swarm for Robot manipulator", International Journal of Information Engineering and Electronic Business, vol. 5, no. 1, **(2013)**.

[15] F. Piltan, M. A. Bairami, F. Aghayari and M. R. Rashidian, "Stable Fuzzy PD Control with Parallel Sliding Mode Compensation with Application to Rigid Manipulator", International Journal of Information Technology and Computer Science, vol. 5, no. 7, **(2013)**.

[16] F. Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari and M. 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, vol. 3, no. 3, **(2012)**, pp. 106-150.

[17] F. Piltan, A. Hosainpour, E. Mazlomian, M. Shamsodini and M. H. Yarmahmoudi, "Online Tuning Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach", International Journal of Robotics and Automation, vol. 3, no. 3, **(2012)**, pp. 77-105.

[18] F. Piltan, R. Bayat, F. Aghayari and B. Boroomand, "Design Error-Based Linear Model-Free Evaluation Performance Computed Torque Controller", International Journal of Robotics and Automation, vol. 3, no. 3, **(2012)**, pp. 151-166.

[19] F. Piltan, J. Meigolinedjad, S. Mehrara and S. Rahmdel, "Evaluation Performance of 2nd Order Nonlinear System: Baseline Control Tunable Gain Sliding Mode Methodology", International Journal of Robotics and Automation, vol. 3, no. 3, **(2012)**, pp. 192-211.

[20] F. Piltan, M. Mirzaei, F. Shahriari, I. Nazari and S. Emamzadeh, "Design Baseline Computed Torque Controller", International Journal of Engineering, vol. 6, no. 3, **(2012)**, pp. 129-141.

[21] F. Piltan, S. Rahmdel, S. Mehrara and R. Bayat, "Sliding Mode Methodology Vs. Computed Torque Methodology Using MATLAB/SIMULINK and Their Integration into Graduate Nonlinear Control Courses", International Journal of Engineering, vol. 6, no. 3, **(2012)**, pp. 142-177.

[22] F. Piltan, M. H. Yarmahmoudi, M. Shamsodini, E. Mazlomian and 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, vol. 3, no. 3, **(2012)**, pp. 167-191.

[23] F. Piltan, H. Rezaie, B. Boroomand and A. Jahed, "Design Robust Backstepping on-line Tuning Feedback Linearization Control Applied to IC Engine", International Journal of Advance Science and Technology, vol. 11, **(2012)**, pp. 40-22.

[24] F. Piltan, S. Siamak, M. A. Bairami and I. Nazari, "Gradient Descent Optimal Chattering Free Sliding Mode Fuzzy Control Design: Lyapunov Approach", International Journal of Advanced Science and Technology, vol. 43, **(2012)**, pp. 73-90.

[25] F. Piltan, M. R. Rashidian, M. Shamsodini and S. Allahdadi, "Effect of Rule Base on the Fuzzy-Based Tuning Fuzzy Sliding Mode Controller: Applied to 2nd Order Nonlinear System", International Journal of Advanced Science and Technology, vol. 46, **(2012)**, pp. 39-70.

[26] F. Piltan, A. Jahed, H. Rezaie and B. Boroomand, "Methodology of Robust Linear On-line High Speed Tuning for Stable Sliding Mode Controller: Applied to Nonlinear System", International Journal of Control and Automation, vol. 5, no. 3, **(2012)**, pp. 217-236.

[27] F. Piltan, R. Bayat, S. Mehara and J. Meigolinedjad, "GDO Artificial Intelligence-Based Switching PID Baseline Feedback Linearization Method: Controlled PUMA Workspace", International Journal of Information Engineering and Electronic Business, vol. 5, **(2012)**, pp. 17-26.

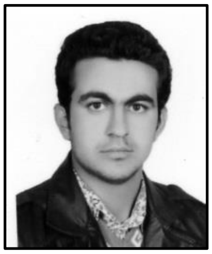
[28] F. Piltan, B. Boroomand, A. Jahed and H. Rezaie, "Performance-Based Adaptive Gradient Descent Optimal Coefficient Fuzzy Sliding Mode Methodology", International Journal of Intelligent Systems and Applications, vol. 11, **(2012)**, pp. 40-52.

[29] F. Piltan, S. Mehrara, R. Bayat and S. Rahmdel, "Design New Control Methodology of Industrial Robot Manipulator: Sliding Mode Baseline Methodology", International Journal of Hybrid Information Technology, vol. 5, no. 4, **(2012)**, pp. 41-54.
[30] F. Piltan and S. T. Haghighi, “Design Gradient Descent Optimal Sliding Mode Control of Continuum Robots”, International Journal of Robotics and Automation, vol. 1, no. 4, **(2012)**, pp. 175-189.
[31] F. Piltan, N. Sulaiman, A. Jalali and K. 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.
[32] F. Piltan, N. Sulaiman, A. Zare, M. Dialame and S. Allahdadi, “Design Adaptive Fuzzy Inference Sliding Mode Algorithm: Applied to Robot Arm”, International Journal of Robotics and Automation, vol. 3, no. 1, **(2011)**, pp. 283-297.
[33] F. Piltan, N. Sulaiman, S. Roosta, A. Gavahian and S. 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.
[34] F. Piltan, N. Sulaiman, A. Jalali, S. Siamak and I. 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.
[35] F. Piltan, N. Sulaiman, A. Gavahian, S. Roosta and S. 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.
[36] F. Piltan, N. Sulaiman, S. Soltani, S. Roosta and A. Gavahian, "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)**, pp. 360-379.
[37] F. Piltan, A. Nabaee, M. M. Ebrahimi and M. Bazregar, “Design robust fuzzy sliding mode control technique for robot manipulator systems with modeling uncertainties”, I. J. Information Technology and Computer Science, vol. 5, no. 8, **(2013)**, pp. 123-135.

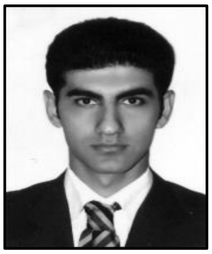
Authors

Sara Heidari is a mathematical researcher of research and development company SSP. Co. She is an expert artificial intelligence and expert systems in this company. Her research activities deal with the robotic control, artificial intelligence and expert system.

Farzin Piltan was born on 1975, Shiraz, Iran. In 2004 he is jointed the research and development company, SSP Co, Shiraz, Iran. In addition to 7 textbooks, Farzin Piltan is the main author of more than 79 scientific papers in refereed journals. He is editorial board of international journal of control and automation (IJCA), editorial board of International Journal of Intelligent System and Applications (IJISA), editorial board of IAES international journal of robotics and automation, editorial board of International Journal of Reconfigurable and Embedded Systems and reviewer of (CSC) international journal of robotics and automation. His main areas of research interests are nonlinear control, artificial control system and applied to FPGA, robotics and artificial nonlinear control and IC engine modelling and control.



Mohammad Shamsodini is an electrical Electronic researcher in research and development company SSP. Co. His main areas of research interests are nonlinear control, artificial control system and robotics.



Kamran Heidari is a researcher of research and development company SSP. Co. His research activities deal with the robotic control, artificial intelligence and expert system.

Samaneh Zahmatkesh is a Electrical engineer researcher of research and development company SSP. Co. She is an expert artificial intelligence and expert systems in this company. Her research activities deal with the robotic control, artificial intelligence and expert system.