

Intelligent Mechatronic Model Reference Theory for Robot End-effector Control

Mohammad sadegh Dahideh, Mohammad Najafi, AliReza Zarei, Yaser Barmayeh and Mehran Afshar

Department of Electrical Engineering, Delvar Branch, Islamic Azad University, Delvar, Iran
Yaser.barmayeh2014@gmail.com

Abstract

The control problem for manipulators is to determine the joint inputs required to case the end-effector execute the commanded motion. The nonminimum phase characteristic of a rigid manipulator makes the design of stable controller that ensures stringent tracking requirements a highly nontrivial and challenging problem. A useful controller in the computed torque family is the gravity controller. To compensate the dynamic parameters, fuzzy logic methodology is used and applied parallel to this method. When the arm is at rest, the only nonzero terms in the dynamic is the gravity. Proposed method can cancels the effects of the terms of gravity. In this case inorder to decrease the error and satteling time, higher gain controller is design and applied to nonlinear system.

Keywords: Gravity controller, fuzzy logic theory, fuzzy gravity method, robot manipulator

1. Introduction

Robot manipulators have many applications in aerospace, manufacturing, automotive, medicine and other industries. Robot manipulators consist of three main parts: mechanical, electrical, and control. In the mechanical point of view, robot manipulators are collection of serial or parallel links which have connected by revolute and/or prismatic joints between base and end-effector frame. The robot manipulators electrical parts are used to run the controllers, actuators for links motion and sensors, which including the following subparts: power supply to supply the electrical and control parts, power amplifier to amplify the signal and driving the actuators, DC/stepper/servo motors or hydraulic/pneumatic cylinders to move the links, and transmission part to transfer data between robot manipulator subparts [1]. Control part is used to adjust the timing between the subparts of robot manipulator to reach the best trajectory. It provides four main abilities in robot manipulators [2]: controlling the manipulators movement in correct workspace, sensing the information from the environment, being able to intelligent control behavior and processing the data and information between all subparts.

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[3]. 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 [4]. 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 [5]. 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 $X_d(t)$.

Some of robot manipulators are controlled by linear PID controllers, but the design of linear controller for robot manipulator is extremely difficult because this system is hardly nonlinear and uncertain [6]. A useful controller is the gravity controller. This method is much simpler to implement than the exact CTC. 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 [7]. 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 [8]. The applications of artificial intelligence such as neural networks and fuzzy logic in modelling and control are significantly growing especially in recent years.

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. To reduce the above challenges, the nonlinear robust controller is used to control of robot manipulator. PD plus gravity is a nonlinear controller [9-11]. Parallel fuzzy gravity controller is used to control of highly nonlinear systems especially for robot manipulators. Estimate nonlinear equivalent dynamic formulation in uncertain dynamic parameter is the main drawback in pure gravity controller. The nonlinear equivalent dynamic formulation problem in uncertain system is solved by using fuzzy logic theorem. Fuzzy logic theory is used to estimate the system's dynamics. This methodology is based on applied fuzzy logic in gravity to estimate the nonlinear term of robot manipulator [12-15].

This paper is organized as follows: In section 2, main subject of modeling robot manipulator formulation, detail of computed torque methodology, introduction to gravity methodology and fuzzy logic method are presented. Detail of proposed methodology is presented in section 3. In section 4, the simulation result is presented and finally in section 5, the conclusion is presented.

2. Dynamic Formulation of Robot Manipulator

Based on mechanical and control methodologies research in robotic system, mechanical design, type of actuators and type of systems drive play important roles to have the best performance controller. Types of kinematics chain, i.e., serial Vs. parallel manipulators, and types of connection between link and joint actuators, i.e., highly geared systems Vs. direct-drive systems are presented in the following sections because these topics played important roles to select and design the best acceptable performance controllers [1-4]. A serial link robot is a sequence of joints and links which begins with a base frame and ends with an end-effector. This type of robot manipulators, comparing with the load capacity is more weightily because each link must be supported the weights of all next links and actuators between the present link and end-effector [1, 6]. Serial robot manipulators have been used in automotive industry, medical application, and also in research laboratories [1, 6]. Study of robot manipulators is classified into two main groups: kinematics and dynamics. Dynamic equation is the study of motion with regard to forces.

Dynamic modeling is vital for control, mechanical design, and simulation. It is used to describe dynamic parameters and also to describe the relationship between displacement, velocity and acceleration to force acting on robot manipulator. The equation of an n -DOF robot manipulator governed by the following equation [1, 4]:

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

Where τ is actuation torque, $M(q)$ is a symmetric and positive definite 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]:

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

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

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

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

3. Control Theory: Computed Torque Controller

Computed torque controller (CTC) is a powerful nonlinear method, which it is widely used in control of robot manipulator. It is based on feedback linearization and computes the required results using the nonlinear feedback control law. This controller works very well when all dynamic and physical parameters are known. In practice, most of physical systems parameters are unknown or time variant, therefore, CTC must be mixed to the other methodology to compensate dynamic equation of robot manipulator. VIVAS and MOSQUERA have proposed a computed torque controller for tracking response in uncertain environment. They compared this method and predictive methodology, however both controllers have been used in feedback linearization, but predictive strategy gives better result as a performance in above research. If an alternative linear state-space equation in the form $\dot{x} = Ax + BU$ can be defined as

$$\dot{x} = \begin{bmatrix} \mathbf{0} & I \\ \mathbf{0} & \mathbf{0} \end{bmatrix} x + \begin{bmatrix} \mathbf{0} \\ I \end{bmatrix} N \quad (4)$$

With $N = B(q)[\dot{q} \dot{q}] + C(q)[\dot{q}]^2 + G(q)$ and this is known as the Brunousky canonical form. By equation (4) and (5) the Brunousky canonical form can be written in terms of the state $x = [e^T \dot{e}^T]^T$ as [1]:

$$\frac{d}{dt} \begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ I \end{bmatrix} N \quad (5)$$

This is a nonlinear feedback control law that guarantees tracking of robot manipulator trajectory. Selecting proportional-plus-derivative (PD) feedback for $N(t)$ results in the PD-CTC ;

$$\tau = M(q)(\ddot{q}_d + K_v\dot{e} + K_p e) + N(q, \dot{q}) \quad (6)$$

According to the linear system theory, convergence of the tracking error to zero is guaranteed. Where K_p and K_v are the controller gains.

Gravity Methodology: A useful family of computed torque controller (CTC) is PD plus Gravity which the results when $M = I, N = G(q)$. Based on CTC method and uncertainty in dynamic formulation, this method is defined by the following formulation;

$$\tau = M(q)(\ddot{q}_d + K_v\dot{e} + K_p e) + G(q) \quad (7)$$

If the Lyapunov function in this method defined as follows;

$$V = \left(\frac{1}{2}\right) (M(q) \times \dot{q}^T \times \dot{q}) + (e^T K_p e) \quad (8)$$

and the differentiate to obtain

$$\dot{V} = (\dot{q}^T)(M(q) \times \ddot{q} + \frac{1}{2}\dot{q}\dot{M} - K_p e) \quad (9)$$

Based on CTC formulation and Lyapunov formulation in CTC;

$$\dot{V} = (\dot{q}^T)\left(\frac{1}{2}\dot{M}(q) - (N(q, \dot{q}) + G(q))\right) \times \dot{q} = \dot{q}^T K_v \dot{q} \quad (10)$$

Therefore the skew symmetry of the first term is given by;

$$\dot{V} = -\dot{q}^T K_v \dot{q} \quad (11)$$

Based on above formulation this methodology has demonstrate stability in the sense of Lyapunov in bounded of error and joint velocity when \dot{V} is negative.

4. Methodology

Based on foundation of fuzzy logic methodology; fuzzy logic controller has played important rule to design nonlinear controller for nonlinear and uncertain systems. However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of;

Input fuzzification (binary-to-fuzzy [B/F] conversion)

Fuzzy rule base (knowledge base), Inference engine and Output defuzzification (fuzzy-to-binary [F/B] conversion). The fuzzy inference engine offers a mechanism for transferring the rule base in fuzzy set which it is divided into two most important methods, namely, Mamdani method and Sugeno method. Mamdani method is one of the common fuzzy inference systems and he designed one of the first fuzzy controllers to control of system engine. Mamdani's fuzzy inference system is divided into four major steps: fuzzification, rule evaluation, aggregation of the rule outputs and defuzzification. Michio Sugeno use a singleton as a membership function of the rule consequent part. The following definition shows the Mamdani and Sugeno fuzzy rule base

$$\begin{aligned} & \text{if } x \text{ is } A \text{ and } y \text{ is } B \text{ then } z \text{ is } C \text{ 'mamdani'} \\ & \text{if } x \text{ is } A \text{ and } y \text{ is } B \text{ then } z \text{ is } f(x, y) \text{ 'sugeno'} \end{aligned} \quad (12)$$

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

$$\mu_U(x_k, y_k, U) = \mu_{\cup_{i=1}^r FR^i}(x_k, y_k, U) = \max \left\{ \min_{i=1}^r \left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U) \right] \right\} \quad (13)$$

The Sum-min aggregation defined as below

$$\mu_U(x_k, y_k, U) = \mu_{\cup_{i=1}^r FR^i}(x_k, y_k, U) = \sum \min_{i=1}^r \left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U) \right] \quad (14)$$

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

$$COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^r \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^r \mu_u(x_k, y_k, U_i)} \quad (15)$$

and *COA* method used the following equation to calculate the defuzzification

$$COA(x_k, y_k) = \frac{\sum_i U_i \cdot \mu_u(x_k, y_k, U_i)}{\sum_i \mu_u(x_k, y_k, U_i)} \quad (16)$$

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

The method of computed torque control works quite well, and we can have better control than linear PD or PID control, but only if we have all necessary information about nonlinear dynamic formulation of system and the parameters of robot manipulator. These are very hard to have in practice. At the same time, the dynamics of the robot can change during the process, and that can affect the result of the control, too. In this case the result of CTC can decrease because the inquiry of dynamic model. To avoid of this situation fuzzy logic method can applied to PD plus gravity method to estimate uncertainty and nonlinear part which it is caused to reduce the performance quality. In this case we can achieve the desired settling time and we can achieve very small steady state tracking errors. Based on fuzzy logic methodology

$$f(x) = U_{fuzzy} = \sum_{l=1}^M \theta^l \zeta(x) \quad (17)$$

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

$$\zeta(x) = \frac{\sum_i \mu(x_i)x_i}{\sum_i \mu(x_i)} \quad (18)$$

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

$$\tau_{fuzzy} = \sum_{l=1}^M \theta^l \zeta(x) = B(q)[\dot{q} \dot{q}] + C(q)[\dot{q}]^2 \quad (19)$$

This methodology has three main parts; linear PD part based on PD linear formulation, nonlinear gravity part to eliminate the term of gravity and fuzzy like nonlinear equivalent part to eliminate the nonlinearity part. Based on this methodology;

$$\tau = M(q)(\ddot{q}_d + K_v \dot{e} + K_p e) + G(q) + \sum_{l=1}^M \theta^l \zeta(x) \quad (20)$$

5. Results and Discussion

Proposed fuzzy gravity (proposed) and gravity controller (PDG) was used to decrease the error. The simulation was implemented in MATLAB/SIMULINK environment.

Based on proposed formulation in PD plus gravity method and fuzzy PD plus gravity method; the performance of these controllers are depended on the PD (K_p and K_v) and gain updating factor coefficients. These factors are calculated by optimization method. In presence of uncertainty both methods have fluctuations

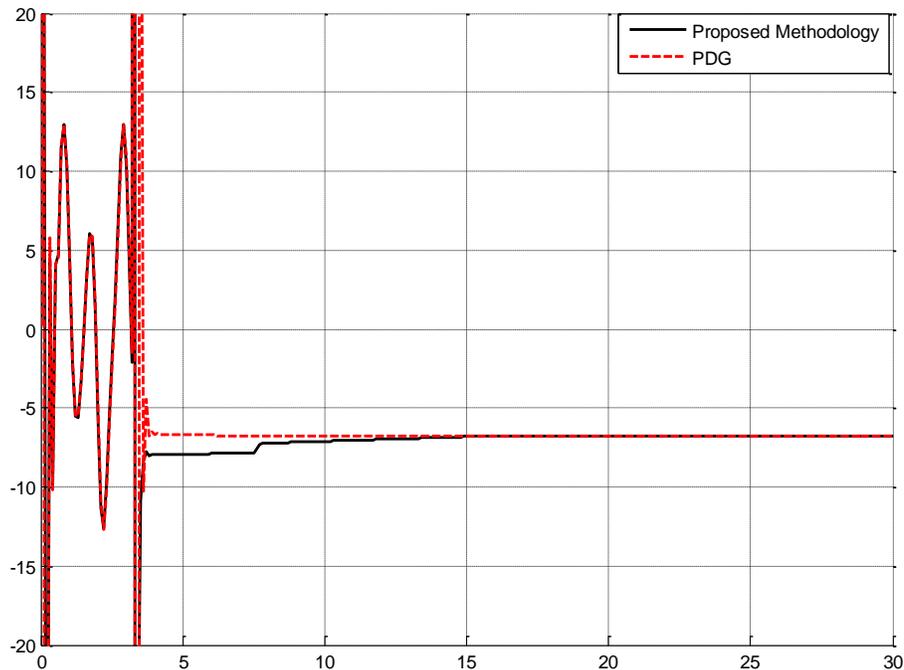


Figure 1. Tracking Performance; Fuzzy Gravity Methodology and Gravity Method

Figure 2 shows the proposed and pure PD plus gravity in presence of 15% disturbance. Figure 2 shows the power disturbance elimination in these two types of controller. The disturbance rejection is used to test the robustness comparisons of these two controllers for desired trajectory.

A band limited white noise with predefined of 15% the power of input signal is applied to the trajectory. It found fairly fluctuations in trajectory responses. Based on Figure 2; by comparing response trajectory with 15% disturbance of relative to the input signal amplitude in fuzzy PD gravity method and gravity method, fuzzy PD gravity method's overshoot about (**1%**) is lower than gravity method's (**9.1%**). fuzzy PD gravity method's rise time (**0.5**) is lower than PD gravity method's (**0.53**). Besides the Steady State and RMS error in fuzzy PD gravity method (**Steady State error =1.6e-6 and RMS error=1.9e-6**) are fairly lower than PD gravity method's (**Steady State error \cong 0.003 and RMS error=0.0048**). Based on Figure 5, PD gravity method has oscillation in trajectory response with regard to 15% of the input signal disturbance but fuzzy PD gravity method is more robust.

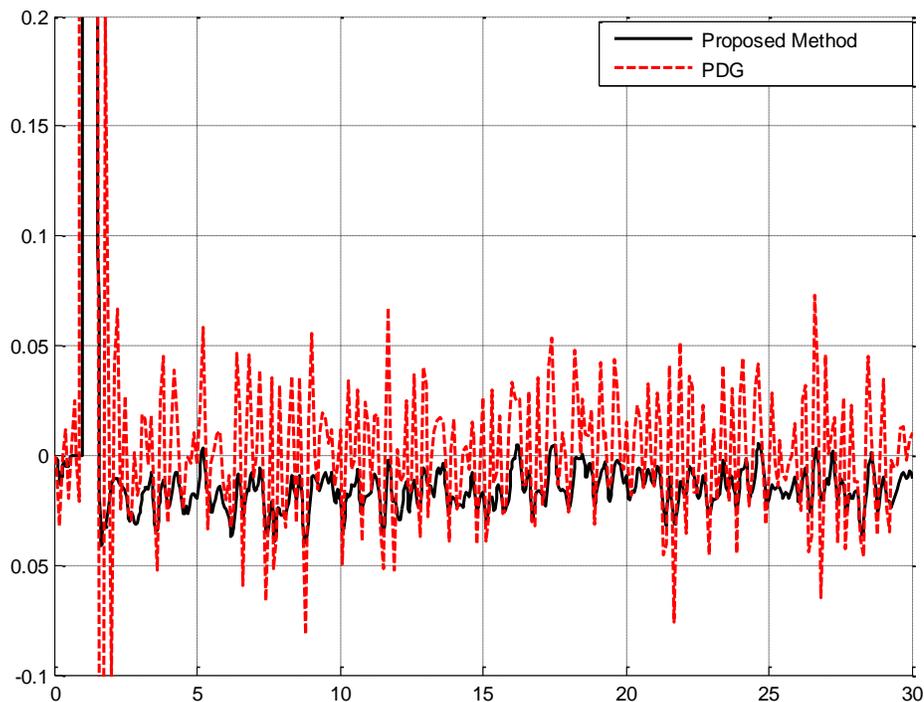


Figure 2. Disturbance Rejection; Fuzzy Gravity Methodology and Gravity Method

6. Conclusion

In this research, we develop a robust model free dynamical controller for solving the problem of tracking and regulation for both of certain and uncertain dynamic parameters. The design of the controller is based on eliminate the gravity of rigid link manipulator and estimate the centrifugal torques and coriolios torques parameters of robot manipulator by artificial intelligence method. Stability conditions imposed the output performance is insensitive to the disturbances. Furthermore, PD gravity method is investigated for comparative purpose. In this research the main control objective is to achieve sufficiently small tracking error. In order to increase the influence of disturbances on dynamic properties of the output trajectories, a high feedback gain, and a higher order output derivatives in the feedback loop are used.

References

- [1] G. I. Vachtsevanos, K. Davey and K. M. Lee, "Development of a Novel Intelligent Robotic Manipulator", *IEEE Control System Magazine*, (1987), pp. 9-15.
- [2] K. Davey, G. I. Vachtsevanos and R. Powers, "An analysis of Fields and Torques in Spherical Induction Motors", *IEE Transactions on Magnetics*, vol. MAG-23, (1987), pp. 273-282.
- [3] A. Foggia, E. Oliver and F. Chappuis, "New Three Degrees of Freedom Electromagnetic Actuator", *Conference Record -IAS Annual Meeting*, vol. 35, New York, (1988).
- [4] K. M. Lee, G. Vachtsevanos and C-K. Kwan, "Development of a Spherical Wrist Stepper Motor", *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, Philadelphia, PA, April 26-29.
- [5] K. M. Lee and I. Pei, "Kinematic Analysis of a Three Degree-of-Freedom Spherical Wrist Actuator", *The Fifth International Conference on Advanced Robotics*, Italy, (1991).
- [6] I. Wang, G. Jewel and D. Howe, "Modeling of a Novel Spherical Pennant Magnet Actuator", *Proceedings of IEEE International Conference on Robotics and Automation*, Albuquerque, New Mexico, (1997), pp. 1190-1195.
- [7] I. Wang, G. Jewel and D. Howe, "Analysis, Design and Control of a Novel Spherical Pennant Magnet Actuator", *IEE Proceedings on Electrical Power Applications*, vol. 154, no. 1, (1998).
- [8] G. S. Chirikjian and D. Stein, "Kinematic Design and Commutation of a Spherical Stepper Motor", *IEEE/ASME Transactions on Mechatronics*, Piscataway, New Jersey, vol. 4, no. 4, (1999) December, pp. 342-353.
- [9] K. Kahlen and R. W. De Doncker, "CW/Tent Regulators for Multi-phase Pennant Magnet Spherical Machines", *Industry Applications Conference Record of the 2000 IEEE*, vol. 3, (2000), pp. 2011-2016.
- [10] K. M. Lee, I. Pei and U. Gilboa, "On the Development of a Spherical Wrist Actuator", *Proceedings of the 16th NSF Conference on Manufacturing Systems Research*, Tempe AZ, (1990) January 8-12.
- [11] C. Yang and Y. S. Back, "Design and Control of the 3-dof freedom actuator by Controlling the Electromagnetic Force", *IEEE Transactions on Magnetics*, (1999) May, pp. 3607-3609.
- [12] F. Piltan, S. Emamzadeh, Z. Hivand, F. Shahriyari and M. Mirzaei, "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.
- [13] 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.
- [14] F. Piltan, M. Eram, M. Taghavi, O. R. Sadrnia and M. Jafari, "Nonlinear Fuzzy Model-base Technique to Compensate Highly Nonlinear Continuum Robot Manipulator", *IJISA*, vol. 5, no. 12, (2013), pp. 135-148. DOI: 10.5815/ijisa.2013.12.12.
- [15] F. Piltan, N. Sulaiman, A. Zargari, M. Keshavarz and A. 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.