



Comparative Study of Control Algorithms for 2-Degree of Freedom RR Planar Robot Manipulator Using Partical Swarm Optimization

Umut MAYETİN^{1*} (Orcid ID: 0000-0002-9114-088X)

Serdar KÜÇÜK¹, (Orcid ID: 0000-0002-5543-7539)

¹Kocaeli University, Faculty of Technology, Department of Biomedical Engineering, Kocaeli

*Sorumlu yazar (Corresponding author): umayetin41@gmail.com

Geliş Tarihi (Received): 15.10.2022

Kabul Tarihi (Accepted): 18.11.2022

Abstract

In this paper; Proportional-Integral-Derivative Control, Fuzzy Logic and Integral Control, Sliding Mode Control algorithms are examined for trajectory control of 2 Degrees-of-Freedom (DOF) Planar Robot Manipulator (PRM). Firstly, conceptual model of the 2-DOF PRM is designed by using 3-dimensional design software. Subsequently, the actual mechanical architecture of 2-DOF PRM is constructed with imperfect transmission system. Kinematics/dynamics equations and control algorithms of the planar robot manipulator are embedded to the digital signal processor by using Matlab/Simulink toolbox. Afterwards, the parameters of control methods are tuned by using Partical Swarm Optimization (PSO). Four different experiments are carried out by using the same circular Cartesian trajectory in order to test the robustness of the control algorithms. Finally, the comparison results of the control algorithms obtained from the actual mechanism of the 2-DOF PRM are presented and discussed.

Keywords: Planar robot manipulator, proportional-integral-derivative control, fuzzy logic and integral control, sliding mode control, particle swarm optimization, digital signal processor

INTRODUCTION

The major task of industrial robot manipulators is to operate its end-effector with high precision. In general, robot manipulators are forced to track the desired trajectory for performing required task. However, a position error exists between the desired and actual trajectories while the manipulator performs this task. There are several types of position errors for industrial robot manipulators which result from improper production of the manipulator's mechanical components (geometric errors), gravity, joint compliances, gear transmission (non-geometric errors) (Gong et al., 2000) and controllers (Mavroidis et al., 1997). Geometric and non-geometric errors can be eliminated by manufacturing the manipulator parts precisely and using the error calibration models (Jang et al., 2001). The controller errors might be minimized by selecting the proper control algorithm for the specified robotic manipulator.

Several scientific studies have been focused on control algorithms for 2-Degree of Freedom (DOF) Planar Robot Manipulator (PRM) during the last decades. Most of these control algorithms have been tested by using computer simulations only. It is well known that there can be differences between simulation and experimental results in general. In simulation studies, the practical aspects of physical mechanisms such as friction, noise and actuator dynamics are not mostly considered. Therefore testing robustness of the control algorithms only in a computer environment yields incomplete results (Reyes and Kelly, 2001). On the other hand, experimentally validated control algorithms are of great importance in terms of their potential implementation to industrial robotic manipulators. Despite a great amount of simulation work in control algorithms for 2-DOF PRM (Wang et al., 2010; Bingul and Karahan, 2011, Nagesh et al., 2012; Queen et al., 2012; Acob et al., 2013; Elkhateeb et al., 2017; Ozkan, 2018; Sharkawy and Koustoumpardis, 2019; Zakia et al., 2019; Baccouch and Dodds, 2020; Ilgen et al., 2021), limited number of experimental studies have been completed for 2-DOF PRM (Reyes and Kelly, 2001; Parra-Vega et al., 2003; Reyes and Rosado, 2005; Osypiuk et al., 2006; Sharkawy et al., 2011; Garcia-Rodriguez and Parra-Vega, 2012; D'ippolito et al., 2014; Kormushev et al., 2015). Most of the experimental studies include performance analysis of solely one control method or comparison of two different control algorithms only (Reyes and Kelly, 2001; Queen et al., 2012; Acob et al., 2013; Zakia et al., 2019; Ilgen et al., 2021). There are very few studies comparing more than two algorithms in the same paper like Sharkawy et al. (2011). Experimental control algorithms for 2-DOF PRM are mostly carried out in horizontal plane (Parra-Vega et al., 2003; Sharkawy et al., 2011; Garcia-Rodriguez and Parra-Vega, 2012; D'ippolito et al., 2014; Kormushev et al., 2015). Real performance of control algorithms may not be measured properly in the horizontal plane since gravitational force is not any effect on

the control algorithms in horizontal plane. Only a few experimental studies are carried out in vertical plane (Reyes and Kelly, 2001; Reyes and Rosado, 2005; Osypiuk et al., 2006). Almost all of the experimental studies about 2-DOF PRM are related to Proportional-Integral-Derivative (PID) control and its derivatives. Only Sharkawy et al. (2011) used four algorithms Proportional Derivative (PD), Computed Torque Control (CTC), Sliding Mode Control (SMC) and Fuzzy Logic Control (FLC) for tuning the control parameters of 2-DOF PRM. However, this study suffers from two conditions under consideration, i) 2-DOF PRM manipulator is in the horizontal plane and ii) any optimization algorithm is not used for tuning control parameters. Contributions of this paper can be stated as follows: i) On the contrary of experimental studies mentioned above, this paper is the first study that compares the trajectory tracking performances of three control methods (PID, FLIC-Fuzzy Logic and Integral Control and SMC) at the same study considering 2-DOF PRM in the vertical plane where gravitational force have full effect, ii) this paper is one of the few experimental studies that uses PSO algorithm for tuning control parameters of three control methods at the same time. PSO algorithm is especially selected in this study as an optimization algorithm since it has been proven an effective technique for tuning control parameters (Poli, 2008; Wang et al., 2010; Bingul and Karahan, 2011; Nagaraj and Vijayakumar, 2011; Zhang et al., 2015; Sungthonga and Assawinchaichoteb, 2016). Advanced optimization algorithms like PSO, Artificial Bee Colony (ABC) and Genetic Algorithm (GA) generate better trajectory tracking results than traditional analytical methods. iii) Since 2-DOF PRM presented in this study is manufactured in the laboratory, it has imperfect axial design, higher backlashes in the gearboxes and higher friction than the mass production industrial robot manipulators. These imperfect conditions make 2-DOF PRM an appropriate robot control test platform for comparing the robustness of the control algorithms. Therefore comparative results obtained from PID, FLIC and SMC in this study are especially valuable since they are attained from an imperfect robot mechanism. iv) Achieving close trajectory tracking performance from imperfect mechanism of 2-DOF PRM to the advanced robotic systems which have expensive drivers like in Garcia-Rodriguez and Parra-Vega (2012) and expensive control board like in D'ippolito et al. (2014) is the another contribution of this study.

MATERIAL AND METHODS

Dynamic model of the 2-DOF PRM

The rigid body model of 2-DOF PRM illustrated in Figure 1 composes of two revolute joints. Two direct current (DC) motors are used for actuating these revolute joints. The first DC motor

is directly coupled to the first link with the maximum torque of about 5Nm while the second DC motor is indirectly coupled to the second link through a gearbox as shown in Figure 1.

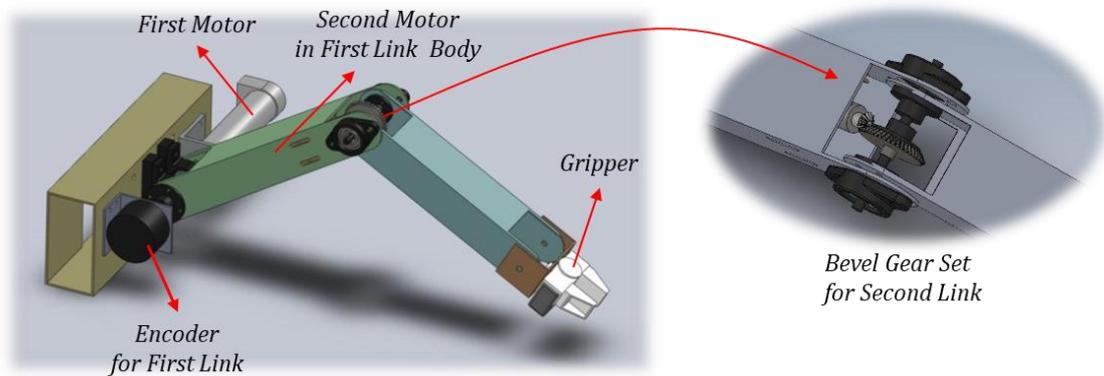


Figure 1. The rigid body of the two-link planar robot manipulator

Figure 2 illustrates the dynamics model of 2-DOF PRM given in Figure 1. The m_1 and m_2 illustrate the total masses of links l_1 and l_2 , respectively. The $c_{m_1} = [c_{m_1,x} \ c_{m_1,y} \ c_{m_1,z}]^T$ and $c_{m_2} = [c_{m_2,x} \ c_{m_2,y} \ c_{m_2,z}]^T$ are the mass centers of the links l_1 and l_2 , respectively. Letter r_i illustrates the distance between point A_i and the center of mass of link l_i , where $i = 1$ and 2. Considering the above explanations and Figure 2, the torque equations (τ_1 and τ_2) are obtained as follows where $\theta_i, \dot{\theta}_i$ and $\ddot{\theta}_i$ are the positions, velocity and accelerations, respectively.

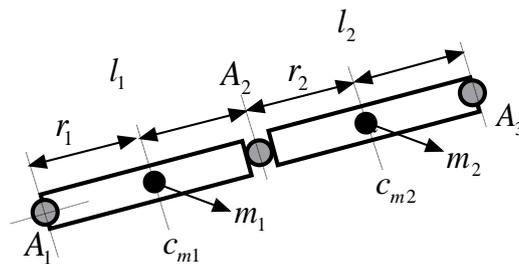


Figure 2. The dynamics model of 2-DOF PRM

$$\begin{aligned}
 \tau_1 = & [m_2 l_1^2 + 2m_2 l_1 c_{m2x} \cos(\theta_2) - 2m_2 l_1 c_{m2y} \sin(\theta_2) + m_2 c_{m2x}^2 + m_2 c_{m2y}^2 + I_{zz2} + \\
 & m_1 (c_{m1x}^2 + c_{m1y}^2) + I_{zz1}] \ddot{\theta}_1 + [m_2 l_1 c_{m2x} \cos(\theta_2) - m_2 l_1 c_{m2y} \sin(\theta_2) + m_2 c_{m2x}^2 + \\
 & m_2 c_{m2y}^2 + I_{zz2}] \ddot{\theta}_2 + [-2m_2 l_1 c_{m2x} \sin(\theta_2) - 2m_2 l_1 c_{m2y} \cos(\theta_2)] \dot{\theta}_1 \dot{\theta}_2 + \\
 & [-m_2 l_1 c_{m2x} \sin(\theta_2) - m_2 l_1 c_{m2y} \cos(\theta_2)] \dot{\theta}_2^2 + gm_1 (c_{m1y} \sin(\theta_1) + c_{m1x} \cos(\theta_1)) + \\
 & gm_2 (c_{m2y} \sin(\theta_1 + \theta_2) + c_{m2x} \cos(\theta_1 + \theta_2) - l_1 \cos(\theta_1))
 \end{aligned} \tag{1a}$$

$$\begin{aligned}
 \tau_2 = & [m_2 l_1 c_{m2x} \cos(\theta_2) - m_2 l_1 c_{m2y} \sin(\theta_2) + m_2 c_{m2x}^2 + m_2 c_{m2y}^2 + I_{zz2}] \ddot{\theta}_1 + \\
 & [m_2 c_{m2x}^2 + m_2 c_{m2y}^2 + I_{zz2}] \ddot{\theta}_2 [m_2 l_1 c_{m2y} \cos(\theta_2) + m_2 l_1 c_{m2x} \sin(\theta_2)] \dot{\theta}_1^2 + \\
 & gm_2 (c_{m2y} \sin(\theta_1 + \theta_2) - c_{m2x} \cos(\theta_1 + \theta_2))
 \end{aligned} \tag{1b}$$

Discrete time mathematical model of DC motor

The discrete-time control scheme illustrated in Figure 3 includes in blocks of desired Cartesian trajectory, inverse kinematics, dynamic model, controller and DC motor. The controller block includes PID, FLIC and SMC.

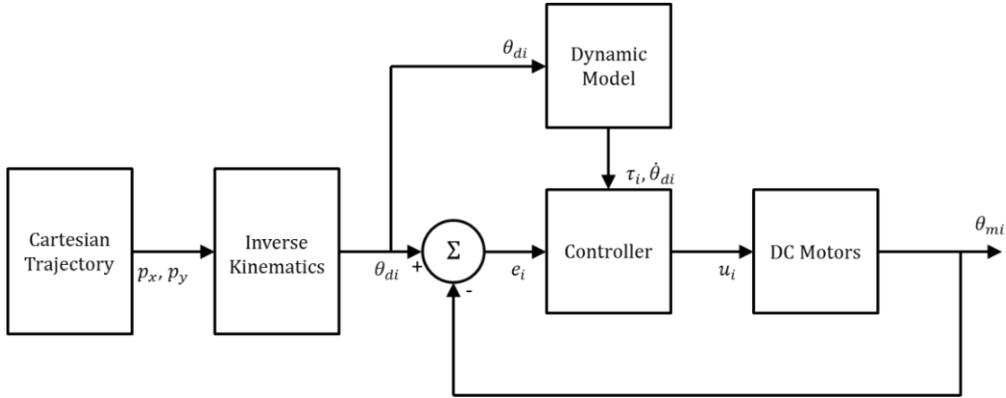


Figure 3. The discrete-time control scheme for trajectory control of 2-DOF PRM

The symbols used in Figure 3 are as follows: θ_{di} is the desired or reference position, θ_{mi} is the measured position, e_i is the error between desired and measured position, τ_i is the computed torque, V_i is the motor armature voltage, where $i = 1$ and 2 . The desired joint angles are obtained by performing inverse kinematics. The equations related to the electrical part of DC motor are given by

$$L_a \frac{di_a}{dt} + R_a i_a = V_a - E_m \quad \text{and} \quad E_m = K_b \frac{d\theta_m}{dt} \quad (2)$$

where L_a , R_a , V_a , E_m and K_b are the armature inductance, armature resistance, armature voltage, back electromotor-force and back electromotor-force constant, respectively. The equations related to the mechanical part of DC motor are stated as

$$J_m \frac{d^2\theta_m}{dt^2} + B_m \frac{d\theta_m}{dt} = \tau_m - \frac{1}{n} \tau_l \quad \text{and} \quad \tau_m = K_m i_a \quad (3)$$

where J_m , θ_m , B_m , τ_m , τ_l , n , K_m , and i_a illustrate the rotor inertia, rotor position, damping constant, generated torque, load torque, gearbox ratio, torque constant and armature current, respectively. DC motor model is obtained as in Figure 4 when Equation 2 and Equation 3 are combined in Laplace domain. In Figure 4, Ω_m denotes rotor velocity, $G_1(s) = \left(\frac{1}{L_a s + R_a}\right)$ and $G_2(s) = \left(\frac{1}{J_m s + B_m}\right)$.

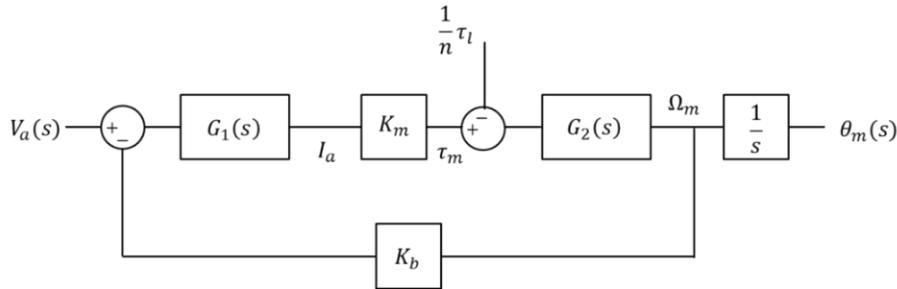


Figure 4. Block diagram of DC motor model

The following transformations can be performed to derive the Discrete-time Motor Model (DMM).

$$G_i(z) = (1 - z^{-1}) \mathcal{Z} \left(\frac{G_i(s)}{s} \right) \quad (4a)$$

where $i = 1$ and 2 . Let's substitute $G_i(s)$ into the Equation 4a

$$G_i(z) = \beta_i (1 - z^{-1}) \mathcal{Z} \left(\frac{1}{s(s + \alpha_i)} \right) \quad (4b)$$

where $\beta_1 = \frac{1}{L_a}$, $\beta_2 = \frac{1}{J_m}$, $\alpha_1 = \frac{R_a}{L_a}$ and $\alpha_2 = \frac{B_m}{J_m}$. The following expression is derived by using partial fractions

$$G_i(z) = \beta_i(1 - z^{-1})\mathcal{Z}\left(\frac{A}{s} + \frac{B}{s + \alpha_i}\right) = \beta_i(1 - z^{-1})\mathcal{Z}\frac{1}{\alpha_i}\left(\frac{1}{s} - \frac{1}{s + \alpha_i}\right) \quad (4c)$$

where $A = -B = \frac{1}{\alpha_i}$. Equation 4d is obtained by transforming $\left(\frac{1}{s} - \frac{1}{s + \alpha_i}\right)$ from s domain to z domain.

$$G_i(z) = \frac{\beta_i}{\alpha_i}(1 - z^{-1})\left(\frac{z}{z - 1} - \frac{z}{z - e^{-\alpha_i T}}\right) \quad (4d)$$

where $\frac{\beta_1}{\alpha_1} = \frac{1}{R_a}$ and $\frac{\beta_2}{\alpha_2} = \frac{1}{B_m}$ and T illustrates the sampling time. $G_1(z)$ and $G_2(z)$ are obtained as follows by substituting $\frac{1}{R_a}$ and $\frac{1}{B_m}$ instead of $\frac{\beta_1}{\alpha_1}$ and $\frac{\beta_2}{\alpha_2}$ into Equation 4d, respectively.

$$G_1(z) = \frac{1}{R_a}\left(\frac{z - 1}{z}\right)\left(\frac{z}{z - 1} - \frac{z}{z - e^{-\alpha_1 T}}\right) = \frac{1}{R_a}\left(1 - \frac{z - 1}{z - e^{-\alpha_1 T}}\right) = \frac{1}{R_a}\left(\frac{z - e^{-\alpha_1 T} - z + 1}{z - e^{-\alpha_1 T}}\right) \quad (4e)$$

$$G_2(z) = \frac{1}{B_m}\left(\frac{z - 1}{z}\right)\left(\frac{z}{z - 1} - \frac{z}{z - e^{-\alpha_2 T}}\right) = \frac{1}{B_m}\left(1 - \frac{z - 1}{z - e^{-\alpha_2 T}}\right) = \frac{1}{B_m}\left(\frac{z - e^{-\alpha_2 T} - z + 1}{z - e^{-\alpha_2 T}}\right) \quad (4f)$$

After simplification $G_1(z)$ and $G_2(z)$ are obtained as

$$G_1(z) = \frac{1}{R_a} \frac{1 - e^{-\alpha_1 T}}{z - e^{-\alpha_1 T}} \quad \text{and} \quad G_2(z) = \frac{1}{B_m} \frac{1 - e^{-\alpha_2 T}}{z - e^{-\alpha_2 T}} \quad (4g)$$

Finally, block diagram of discrete-time motor model is obtained as in Figure 5.

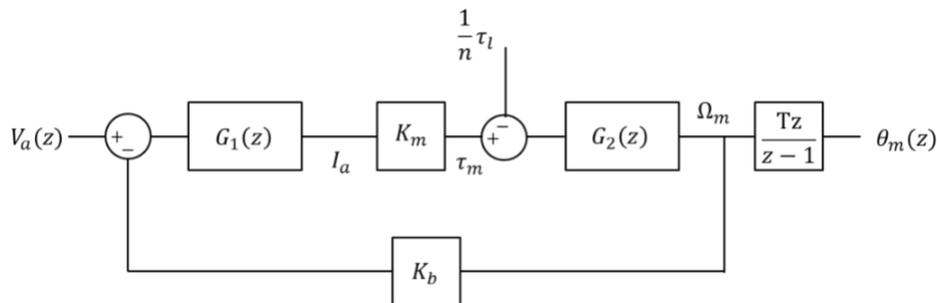


Figure 5. Discrete-time motor model

PID control algorithm

Although several advanced controllers have been developed for robot manipulators, PID controllers are still one of the extensively preferred algorithms in real-time industrial control applications (Xue et al., 2002; Kuo, 2009). The simple structure, easily tuned control parameters and providing effective and reliable results in practical applications are the basic reasons why the PID controllers are still preferred. Adjusting the proportional (k_p), derivative (k_d) and integral (k_i) gains are the main issue of PID controller. In this method, initially an error “ $e(t)$ ” (difference between actual and desired trajectory) is computed. Afterwards, this error is tried to be minimized by adjusting the control signal below.

$$u(t) = k_p e(t) + k_i \int_0^t e(t) d\tau + k_d \frac{de(t)}{dt} \tag{5}$$

The proportional gain k_p adjusts speed of system response. If the k_p is set to a large value, system response can become unstable. Therefore, a suitable k_p value is required to be tuned for operating the system properly. Derivative gain k_d is used for decreasing magnitude of overshoot caused by integral components and improving transient response of the system. In general, very low derivative time constant is used in practical control systems since derivative response is very sensitive to noise signals. Finally, integral gain k_i is used for eliminating the steady-state error. The discrete-time PID controller embedded to the controller block in Figure 3 is given in Figure 6.

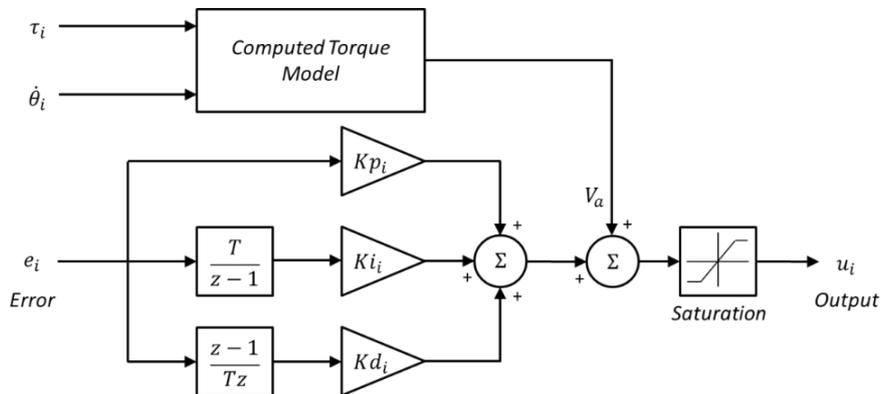


Figure 6. Discrete-time PID controller

Computed torque model illustrated in Figure 6 provides an additional control signal that eliminates disturbances caused by the joint masses, velocities and accelerations. The following current equation can be obtained by using Equation 3 for finding additional control signal.

$$i_a = \frac{J_m}{K_t} \ddot{\theta} + \frac{B_m}{K_t} \dot{\theta} + \frac{1}{nK_t} \tau_L \quad (6)$$

The additional control signal taken from output of computed torque model can be obtained as follows if the Equation 6 is substituted in Equation 2.

$$V_a = \frac{R_a J_m}{K_t} \ddot{\theta} + \frac{R_a B_m}{K_t} \dot{\theta} + \frac{R_a}{nK_t} \tau_L + \frac{L_a J_m}{K_t} \ddot{\ddot{\theta}} + \frac{L_a B_m}{K_t} \ddot{\dot{\theta}} + \frac{L_a}{nK_t} \dot{\tau}_L + K_b \dot{\theta} \quad (7)$$

Fuzzy logic and integral controller

The main concept of fuzzy logic was first proposed by Zadeh (1972, 1973). After its suggestion, FLC has been increasingly used in robotic manipulators which do not have precise dynamic model. Robustness, easy modification, fast and cheaper implementation are the some major advantageous of FLC that comprises of three main stages namely, fuzzification, inference mechanism and defuzzification. The fuzzification stage converts input data into a fuzzy set using membership functions. There are several different types of membership functions (triangular, trape-zoidal, piecewise linear, Gaussian, or singleton) used in fuzzy control system. The triangular membership function used also in this study is chosen in general due to its simple, familiar structure and easy computation. Inference mechanism criticizes the conveniences of the control rules for current step and then decides which input is applied to system. There are two types of inference mechanism namely, Mamdani and Sugeni. Mamdani type is the most preferred one (Panjehfouladgaran et al., 2010) and used also in this study. Defuzzification stage converts membership degrees of fuzzy set into a real value. There are several defuzzification methods (Souise and Bose, 1994; Driankov et al., 1996; Rao and Saraf, 1996) proposed for FLC namely, Center of Area (COA), Center of Sums (COS), Height Method (HM), Mean of Maxima (MOM), Center of Largest Area (COLA), and First of Maxima (FOM). COA is used as defuzzification method in this study. Control performance of system is highly dependent on the membership functions and fuzzy control rules (Simon, 2002). The general architecture of FLC is shown in Figure 7.

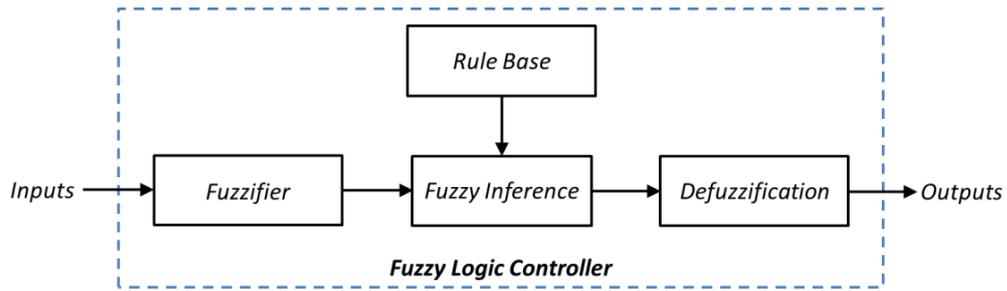


Figure 7. The general architecture of FLC

Discrete-time FLIC embedded to the controller block as shown in Figure 4 is given in Figure 8 where fKi_i illustrates the integral coefficient of the FLIC, $i=1$ and 2.

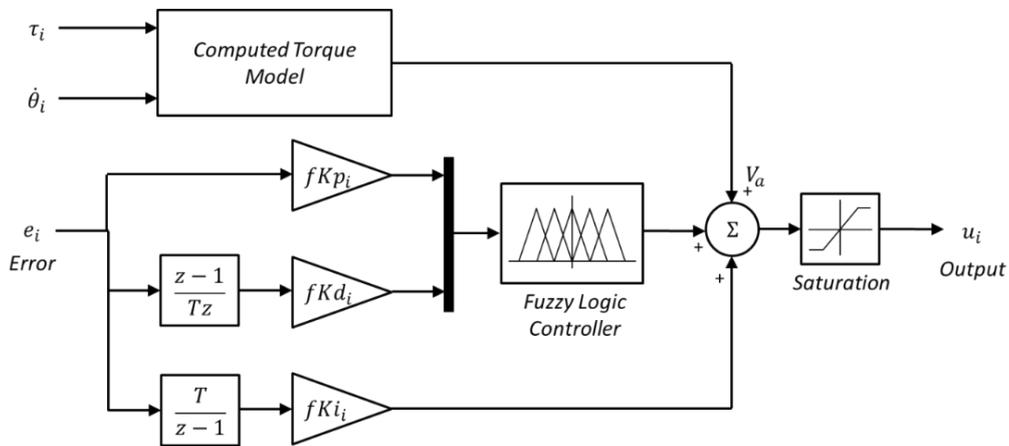


Figure 8. Discrete-time FLIC controller block

Input variables are position errors “ e ” ($radian - rad$) and derivative of position errors “ de ” ($radian/second - rad/s$) while output variable is the motor voltage ($Volt - V$) of the first and second joints. Values of position error inputs and derivative of position error inputs are scaled from $-1 rad$ to $+1 rad$ and from $-10 rad/s$ to $+10 rad/s$, respectively, while motor voltage varies between $-24V$ and $+24V$ for the first and second joints. Fuzzification operator maps input variables to five linguistic rules. Gaussian membership functions are preferred for input and output variables. Figure 9 illustrates initial fuzzy membership functions.

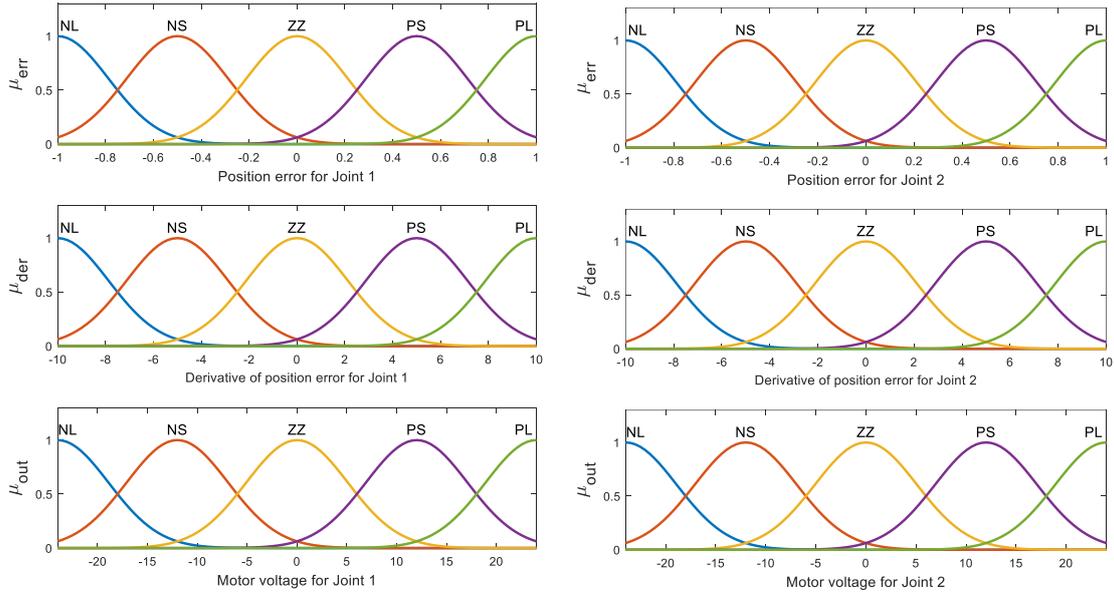


Figure 9. The initial fuzzy membership functions

Fuzzy logic linguistic rules are summarized in Table 1 where NL, NS, ZZ, PS, PL are negative large, negative small, zero-zero, positive small and positive large, respectively. Centroid area is used for defuzzification process that converts fuzzy sets to the control signal u_i .

Table 1. The fuzzy logic linguistic rules

$de \setminus e$	NL	NS	ZZ	PS	PL
NL	NL	NL	NL	NS	ZZ
NS	NL	NL	NS	ZZ	PS
ZZ	NL	NS	ZZ	PS	PL
PS	NS	ZZ	PS	PL	PL
PL	ZZ	PS	PL	PL	PL

Sliding mode control

The sliding mode control has been received more attention from the robotic community especially last decades due to its successfully tracking performance of the desired trajectory under improper conditions such as modeling errors, parameter uncertainties and disturbances (Utkin, 1977; Young, 1978; Slotine, 1985). SMC design is composed of two different stages. The first stage involves designing the switching function selected from the system trajectories. These trajectories are called sliding modes. The sliding modes that do not depend on both system dynamics and control law can be obtained by changing the parameters of switching function. The second stage composes of designing sliding mode control law that involves both

a corrective control law and a suitable sliding surface. Corrective control law forces system trajectories to reach the sliding surface and remains there at all times. The system dynamics and switching surface are given in state space as follows.

$$\dot{x}(t) = Ax(t) + Bu(t) + f(t, x) \quad (8)$$

where $x \in R^n$ represents the states, $A \in R^{n \times n}$, $B \in R^{n \times m}$ $u \in R^{m \times 1}$ denotes control input and $f(t, x) \in R^{n \times 1}$ illustrates disturbance input or unmodeled dynamic term. The discretized counterpart of Equation 8 is obtained by performing “ $z = Tx$ ” coordinate transformation and taking time derivative of the resultant transformation.

$$\dot{z}(t) = T\dot{x} \quad (9)$$

One can obtain the following equation after substituting the $\dot{x}(t)$ into Equation 9 and substituting “ $x = T^{-1}z$ ” into the resultant equation.

$$\dot{z} = TAT^{-1}z + TBu + Tf = A^*z + B^*u + f^* \quad (10)$$

where $TAT^{-1} = A^*$, $TB = B^*$ and $Tf = f^*$. The main purpose of the sliding mode control law (Utkin, 1977) is to force the variable error and its derivative to zero. In this study, the position error $e(k)$ and its derivative $\dot{e}(k)$ are chosen as coordinate variables $z_1(k)$ and $z_2(k)$, respectively.

$$z_1(k) = e(k) = \theta_r(k) - \theta_m(k) \quad (11a)$$

$$z_2(k) = \dot{z}_1 = \dot{e}(k) = \dot{\theta}_r(k) - \dot{\theta}_m(k) \quad (11b)$$

where $\theta_r(k)$ and $\theta_m(k)$ are the desired reference position and actual position (rotor position) at the k^{th} sampling interval, respectively. The $\dot{z}_2(k)$ can be stated by using time derivative of \dot{z}_1 as

$$\dot{z}_2(k) = \ddot{e}(k) = \ddot{\theta}_r(k) - \ddot{\theta}_m(k) \quad (12)$$

Equation 8 can also be written as follows taking Equations 10, 11a, 11b and 12 into account.

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ A_{21}^* & A_{22}^* \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ b_2^* \end{bmatrix} u + \begin{bmatrix} 0 \\ f_2^* \end{bmatrix} \quad (13)$$

where A_{21}^* , A_{22}^* , b_2^* , z_1 , z_2 , f_2^* and $u \in R$. The system dynamic given in Equation 13 is transformed into two sub-systems in order to derive the slope matrix of the sliding surface. The first sub-system given by Equation 14 does not include the control input while the second sub-system stated in Equation 15 contains the control input.

$$\dot{z}_1 = z_2 \quad (14)$$

$$\dot{z}_2 = A_{21}^* z_1 + A_{22}^* z_2 + b_2^* u + f_2^* \quad (15)$$

where u is the control input. In order to find A_{21}^* , A_{22}^* , b_2^* and f_2^* in terms of system parameters the term $\ddot{\theta}_r(k) - \ddot{\theta}_m(k)$ can be written instead of \dot{z}_2 on the left side of Equation 15. The following equation can be obtained after substituting the $\ddot{\theta}_m$ in the resultant equation where $\ddot{\theta}_m$ can be determined by using the electrical and mechanical equations given by Equation 2 and Equation 3, respectively.

$$A_{21}^* z_1 + A_{22}^* z_2 + b_2^* u + f_2^* = \ddot{\theta}_r - \mu_1 \dot{\theta}_m - \mu_2 V_a + \frac{1}{nJ_m} \tau_L \quad (16)$$

where $\mu_1 = \left(-\frac{K_b}{J_m R_a} - \frac{B_m}{J_m} \right)$ and $\mu_2 = \frac{K_m}{J_m R_a}$. Equality does not change when the term $\mu_1 \dot{\theta}_r$ is added to the both sides of Equation 16. The following equation is obtained after transferring the term $\mu_1 \dot{\theta}_r$ to right-hand side of the equation.

$$A_{21}^* z_1 + A_{22}^* z_2 + b_2^* u + f_2^* = \ddot{\theta}_r - \mu_1 (\dot{\theta}_r - \dot{\theta}_m) + \mu_1 \dot{\theta}_m - \mu_2 V_a + \frac{1}{nJ_m} \tau_L \quad (17)$$

The above equation can be rewritten as follows since $\dot{\theta}_r - \dot{\theta}_m = z_2$.

$$A_{21}^* z_1 + A_{22}^* z_2 + b_2^* u + f_2^* = -\mu_1 z_2 - \mu_2 V_a + \left[\ddot{\theta}_r + \mu_1 \dot{\theta}_m + \frac{1}{nJ_m} \tau_L \right] \quad (18)$$

The parameters A_{21}^* , A_{22}^* , b_2^* and f_2^* can be extracted from Equation 18 as follows.

$$\begin{aligned}
 A_{21}^* &= 0 & A_{22}^* &= -\mu_1 = -\left(-\frac{K_b}{J_m R_a} - \frac{B_m}{J_m}\right) \\
 b_2^* &= -\mu_2 = -\frac{K_m}{J_m R_a} & f_2^* &= \ddot{\theta}_r + \mu_1 \dot{\theta}_m + \frac{1}{n J_m} \tau_L
 \end{aligned} \tag{19}$$

In the new situation, the state-space equations can be written as

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & A_{22}^* \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ b_2^* \end{bmatrix} u + \begin{bmatrix} 0 \\ f_2^* \end{bmatrix} \tag{20}$$

where $\dot{z}_1 = z_2$ and $\dot{z}_2 = A_{22}^* z_2 + b_2^* u + f_2^*$. The following sliding surface can be introduced in order to design the controller.

$$\sigma(k) = S^T z(k) \tag{21}$$

where S is a vector and its elements S_1 and $S_2 \in R$ are selected such that they drive the state trajectories to the following sliding surface where sliding dynamics is globally asymptotically stable.

$$\sigma(k) = S_1 z_1(k) + S_2 z_2(k) = 0 \tag{22}$$

where $S_1 \neq 0$ and $S_2 \neq 0$. The controller can be designed considering corrective control (u_c) and equivalent control (u_{eq}). The corrective control forces the system trajectories to reach sliding surface while the equivalent control gets the derivative of the sliding surface equal to zero and remains on the sliding surface for all the time. The following equation is obtained after performing derivative of the sliding surface given by Equation 22 and selecting $S_2 = 1$.

$$\dot{\sigma} = S_1 \dot{z}_1 + \dot{z}_2 = 0 \tag{23}$$

Let's substitute \dot{z}_1 and \dot{z}_2 in Equation 23

$$\dot{\sigma} = (S_1 + A_{22}^*) z_2 + b_2^* u_{eq} + f_2^* = 0 \tag{24}$$

The equivalent control that composes of low frequency part of controller is defined as

$$u_{eq} = -\frac{1}{b_2} [(S_1 + A_{22}^*)z_2 + f_2^*] \quad (25)$$

The corrective control (u_c) comprises high frequency part of controller and is stated as

$$u_c = -k \cdot \text{sign}(\sigma(k)) \quad (26)$$

As parameter k gets larger values, system reaches to sliding surface faster. When k gets much larger, amplitude of chattering gets higher. Total control signal is finally obtained as

$$u = u_c + u_{eq} \quad (27)$$

The block diagram of the control signal of SMC method is shown in Figure 10.

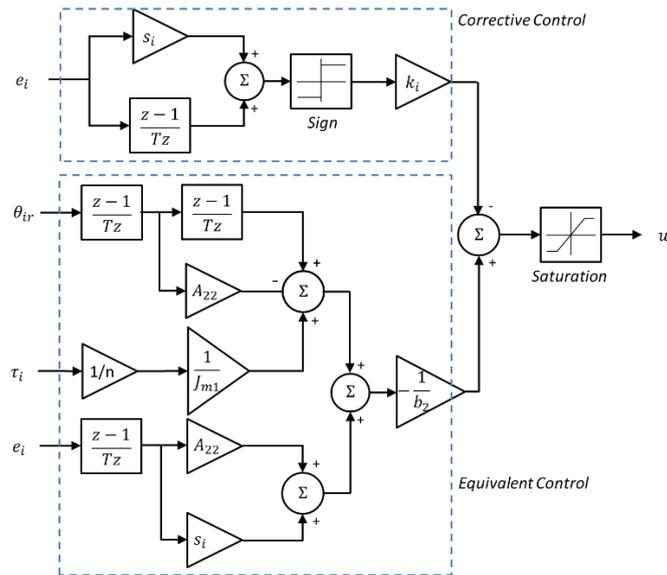


Figure 10. Sliding Mode Controller

Particle swarm optimization

Activities of an individual in a swarm are considered trivial in general, however, their communal actions present great importance for scientists to develop novel computational methods. Behaviors of swarms such as birds, ants and bees can be used for finding new techniques to handle complex problems more efficiently. The particle swarm optimization which has been enormously successful (Poli, 2008) is one of the best techniques developed recently. PSO algorithm optimizes the problem iteratively in order to have a population or a swarm of candidate solutions. Every individual in the swarm is accepted as a particle in the

search space which has own position and velocity. The swarm of particles travels through the search space in accordance with a few simple formulas looking for the best solution (Kennedy and Eberhart, 1995). Each particle has position and velocity based on its own performance as well as the performance of another selected guide particle. In general, the guide particle is selected as the best current performing particle in the swarm.

PSO algorithm based on the movement and intelligence of swarms (Zhang et al., 2015) produces generally better results in control problems than several other optimization algorithms (Nagaraj and Vijayakumar, 2011; Zhang et al., 2015; Sungthonga and Assawinchaichoteb, 2016). Recently, Bingul and Karahan (2011) and Elkhateeb and Badr (2017) studied trajectory control of 2-DOF PRM which is the same configuration under consideration here. Bingul and Karahan (2011) used PSO while Elkhateeb and Badr (2017) used ABC algorithm for tuning controller parameters. Comparisons of two studies illustrate that PSO-PID controller produces smaller trajectory tracking error for 2-DOF PRM than ABC–PID controller. Sungthonga and Assawinchaichoteb (2016) used PSO, GA and traditional Ziegler-Nichols (ZN) method for optimal PID parameters of heater temperature control system. Comparison results illustrate that PSO-PID controller generates better step response than GA and ZN for optimizing parameters of PID controller. Nagaraj and Vijayakumar (2011) used GA, Evolutionary Programming (EP), Ant Colony Optimization (ACO) and PSO algorithms for tuning PID parameters. Comparisons showed that rise time, settling time and integral square error obtained from PSO algorithm are obtained better than GA, EP and ACO algorithms.

EXPERIMENTS WITH 2-DOF PRM

Experimental setup

2-DOF PRM shown in Figure 11 is produced by using homogenous aluminum beams. It consists of two DC motors, motor drivers, links, position sensors, gripper and controller unit.

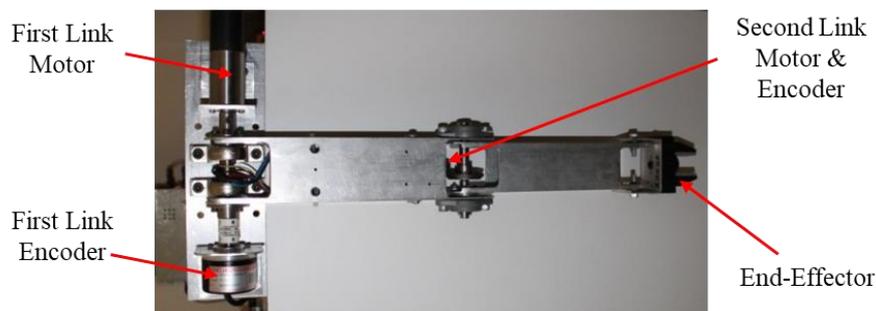


Figure 11. The 2-DOF PRM

Faulhaber DC motor with a gear ratio of 86: 1 is directly coupled to the first link while the Maxon DC motor with the same gear ratio is indirectly coupled to the second link by a bevel gear with a ratio of 44: 13. Both DC motors are supplied by a 24V. The incremental encoders are used for measuring the actual joint positions. The 3600 *ppr* (pulses per revolution) encoder is directly coupled to the first motor shaft while 500 *ppr* encoder directly coupled to the back of the second motor shaft. In order to perform real-time implementation, the kinematics and control algorithms of 2-DOF PRM are embedded to TMS320F28335 DSP using MATLAB Simulink toolbox. The block diagram of the control system is illustrated in Figure 12.

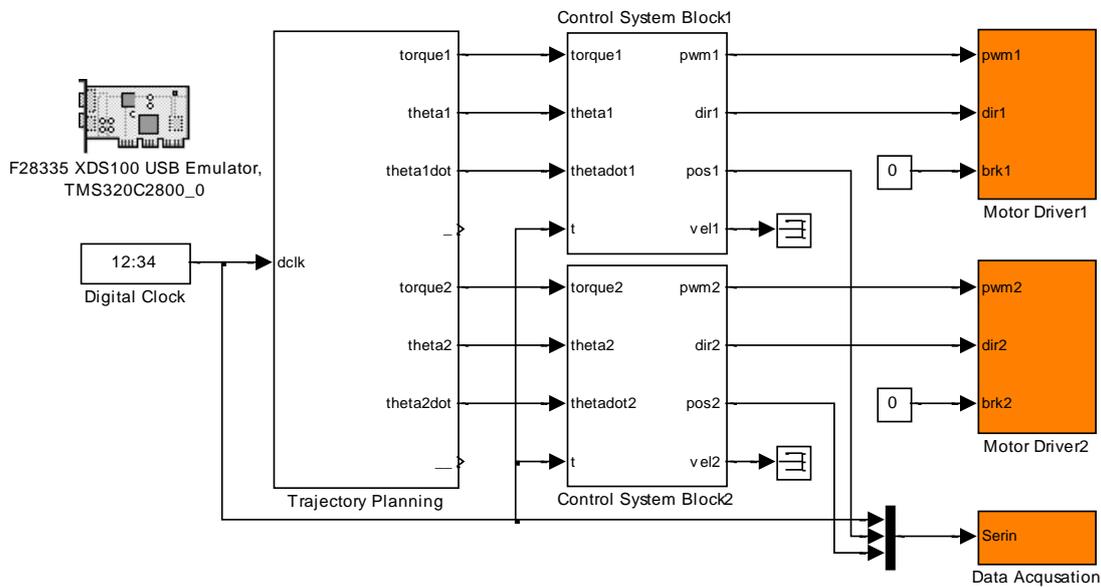


Figure 12. The block diagram of the control system

In the trajectory planning block in Figure 12, end-effector motion is converted from Cartesian space to the joint space. The terms θ_1 & θ_2 and $\dot{\theta}_1$ & $\dot{\theta}_2$ denote desired positions and velocities of the first and second joints, respectively. The control system block in Figure 12 includes controller, signal conversion and encoder units illustrated in Figure 13.

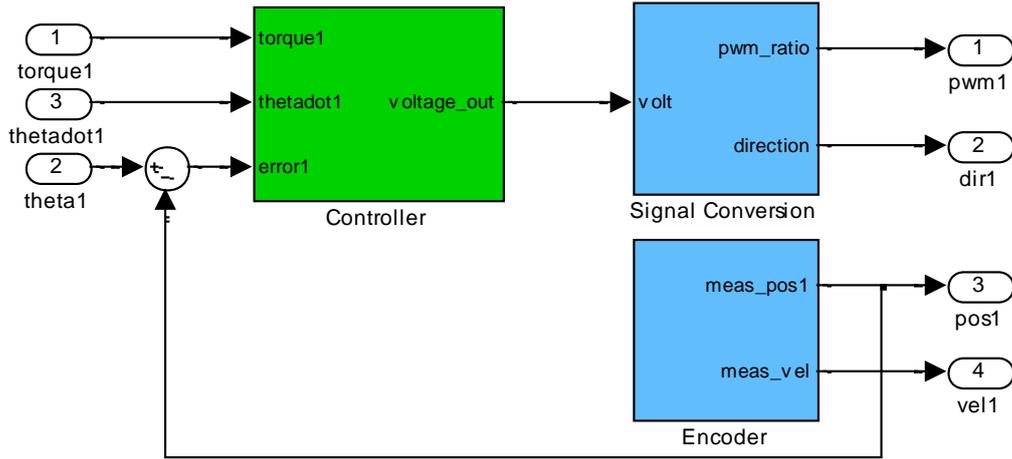


Figure 13. The control system block including controller, signal conversion and encoder.

Controller unit includes in PID, FLIC and SMC algorithms. Signal conversion unit converts the voltage (u_i) supplied by controller unit into the Pulse Width Modulation (PWM) signal and direction signal for motor drivers. Encoder unit measures actual position in radians. Error is the difference between measured actual position and desired position.

Experimental results

Kinematics and dynamic parameters of 2-DOF PRM are illustrated in Table 2 while DC motor parameters used in this study are shown in Table 3. It should be noted that since viscous frictions of DC motors are not illustrated in datasheets, they are obtained from experimental tests.

Table 2. The kinematics and dynamic parameters of 2-DOF PRM

	Notation	Value	Unit
Link1 length	L_1	0,23	m
Link2 length	L_2	0,25	m
Link1 mass	m_1	0,91335	kg
Link2 mass	m_2	0,36375	kg
Link1 center of mass	$c_{m_1,x}$	0,11950	m
	$c_{m_1,y}$	0	
Link2 center of mass	$c_{m_2,x}$	0,11945	m
	$c_{m_2,y}$	-0,00110	
Link1 inertia	I_{xx_1}	$0,547 \cdot 10^{-3}$	kg m ²
	I_{yy_1}	$5,727 \cdot 10^{-3}$	
Link2 inertia	I_{xx_2}	$0,177 \cdot 10^{-3}$	kg m ²
	I_{yy_2}	$3,634 \cdot 10^{-3}$	
Gravity	g	9,81	m/sn ²

Table 3. Numerical values for the DC motors

	Notation	Value	Unit
Armature voltage	V_{a1}	24	V
	V_{a2}	24	V
Rotor inertia	J_{m1}	$4,1.10^{-6}$	kg m ²
	J_{m2}	$1,05.10^{-6}$	kg m ²
Armature resistance	R_{a1}	1,63	Ω
	R_{a2}	7,31	Ω
Armature inductance	L_{a1}	0,270	mH
	L_{a2}	0,832	mH
Torque constant	K_{m1}	0,0377	Nm/A
	K_{m2}	0,044	Nm/A
Back emf constant	K_{b1}	0,0377	V/(rad/sn)
	K_{b2}	0,044	V/(rad/sn)
Viscous friction	B_{m1}	$0,756.10^{-6}$	Nms
	B_{m2}	$1,2.10^{-6}$	Nms
Gearbox ratio	n_1	86	-
	n_2	86	-

K_p , K_d and K_i parameters in PID, fK_p , fK_d , fK_i in FLIC and s and k in SMC algorithms are tuned by using PSO algorithm. The most important step to evaluate the performance of the controllers is to select proper cost function. In this study Mean of Root squared Error (MRSE) and Mean of Absolute Magnitude of the Error (MAE) are used as cost functions.

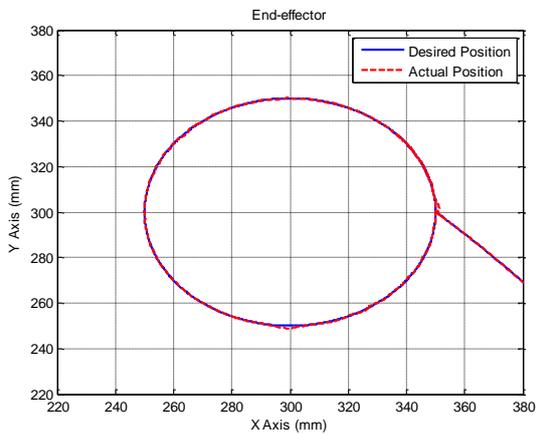
$$\text{MRSE} \rightarrow E(k) = \frac{1}{N} \sum_{i=1}^N \sqrt{e_1^2(i) + e_2^2(i)} \quad (28a)$$

$$\text{MAE} \rightarrow E(k) = \frac{1}{N} \sum_{i=1}^N |e_1(i)| + |e_2(i)| \quad (28b)$$

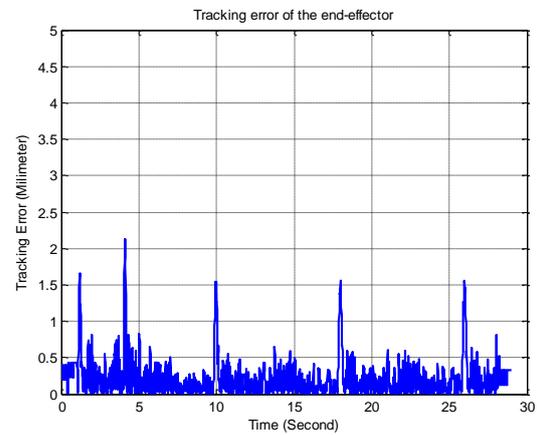
The same circular Cartesian trajectory is used for testing robustness of the control algorithms. Radius and central coordinates of the trajectory are chosen as $50mm$ and $(p_x = 300mm, p_y = 300mm)$, respectively. Duration of the experiments is chosen as 28 seconds. When the experiments are initialized, manipulator joints keep their horizontal line positions and reach the start points of their trajectories in the first four seconds. Afterward, end-effector follows the trajectory three times during the succeeding 24 seconds. Four different experiments are carried

out using the same trajectory. Experiment-I: The 2-DOF PRM is forced to follow the trajectory without carrying any load; Experiment-II, III and IV: 2-DOF PRM is forced to follow the trajectory with loads of 100gr, 200gr and 300gr, respectively. Only trajectory tracking results of PID, FLIC and SMC controllers for Experiment-III are given in this section due to the page limit. However, all results obtained from Experiment-I, II, III and IV are given in Table 4 and Table 5 for easy comparison.

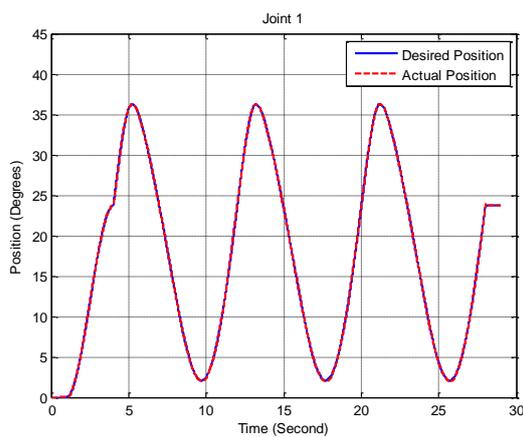
The experimental results from PID controller are illustrated in Figure 14 where average tracking errors are obtained less than 0.5mm for end-effector, 0.1 degrees for first joint and 0.05 degrees for second joint. Backlashes in transmission system cause overshoots at 2nd, 4th, 10th, 18th and 26th seconds as seen in Figure 14b and Figure 14d. PID controller cannot sufficiently minimize these overshoots.



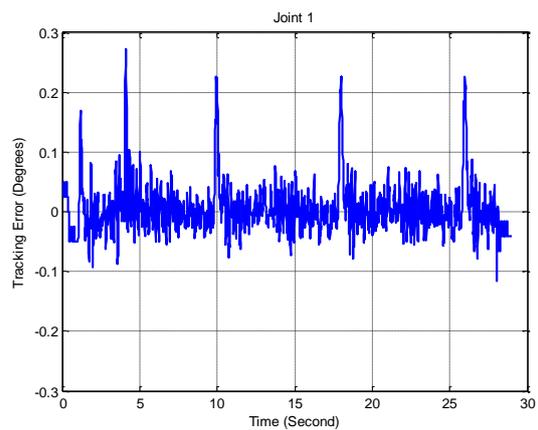
(a)



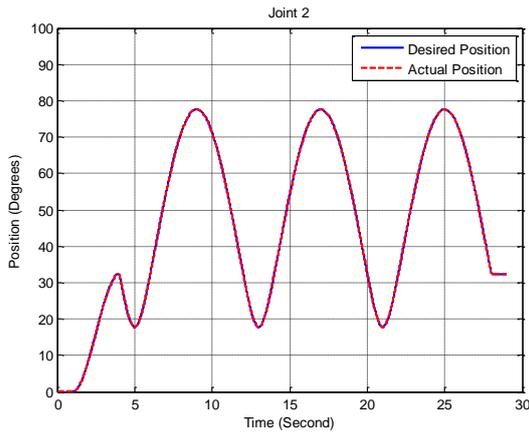
(b)



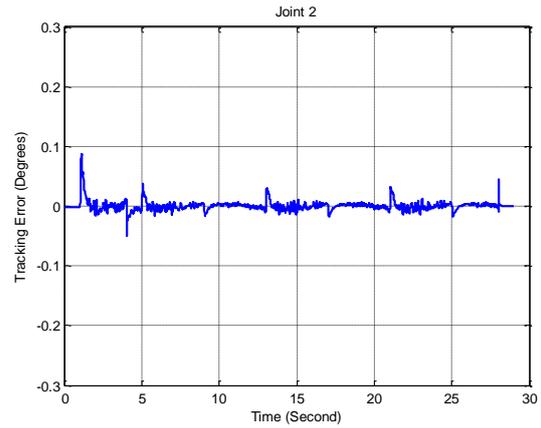
(c)



(d)



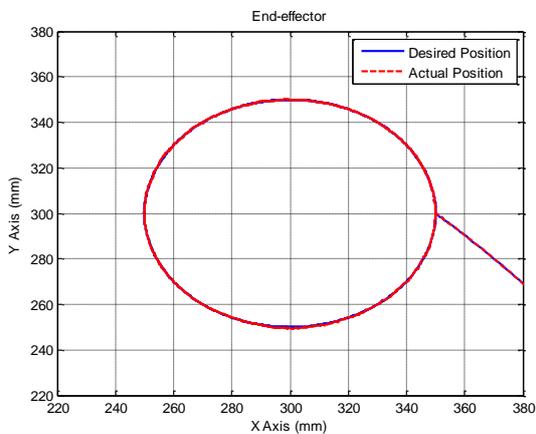
(e)



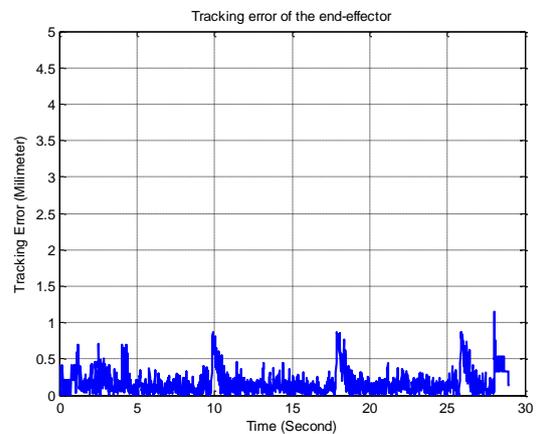
(f)

Figure 14. Desired and actual positions of a) end effector, c) first joint, e) second joints and trajectory tracking errors b) end-effector, d) first joint, f) second joints for PID controller

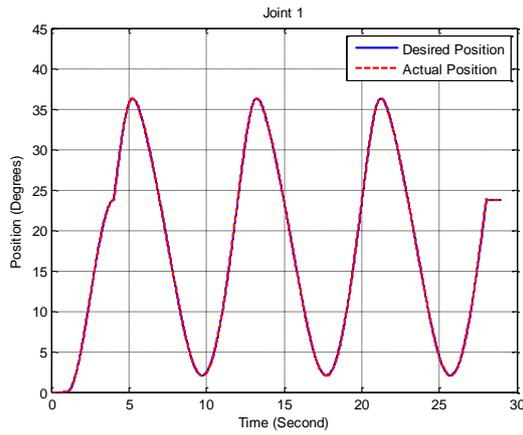
The experimental results obtained from the FLIC are illustrated in Figure 15 where average tracking errors are obtained less than $0.3mm$ for end-effector, 0.06 degrees for first joint and 0.01 degrees for second joint. As seen in Figure 15b and Figure 15d, FLIC controller demonstrates better performance than PID controller at minimizing the overshoots at 2nd, 4th, 10th, 18th and 26th seconds.



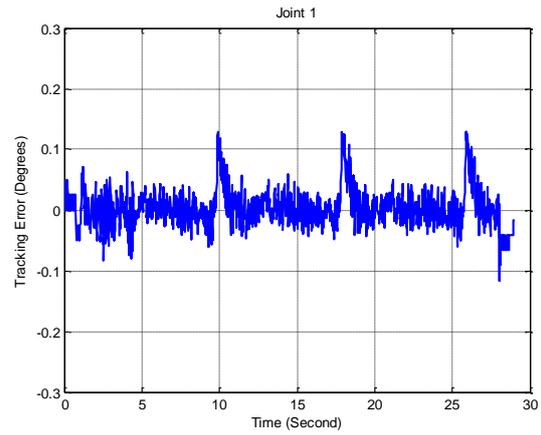
(a)



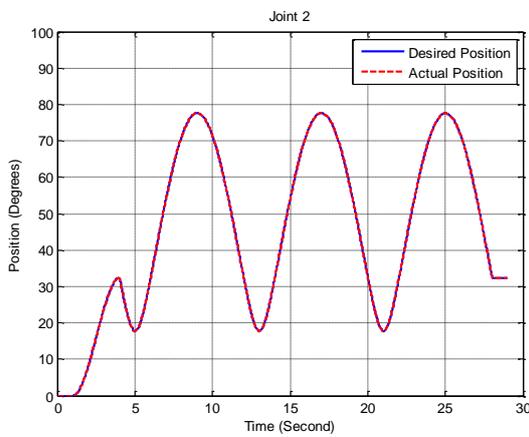
(b)



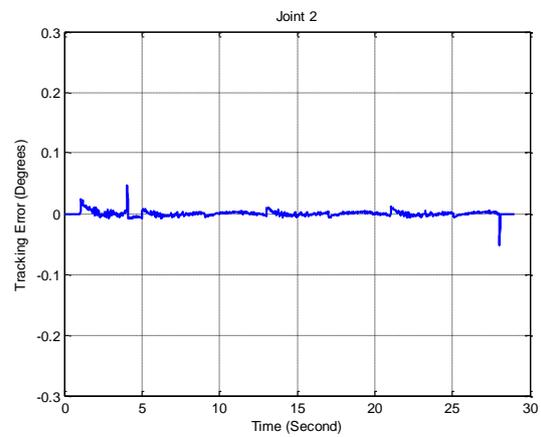
(c)



(d)



(e)



(f)

Figure 15. Desired and actual positions of a) end effector, c) first joint, e) second joints and trajectory tracking errors b) end-effector, d) first joint, f) second joints for FLIC.

The experimental results from the SMC are illustrated in Figure 16 where average tracking errors are obtained less than 0.7mm for end-effector, 0.1 degrees for first joint and 0.05 degrees for second joint. As seen in Figure 16b and Figure 16d, although overshoots do not take place, SMC controller generates chattering along the trajectory. In addition, experiments show that the second joint generates less tracking error compared to the first joint for PID, FLIC and SMC controllers.

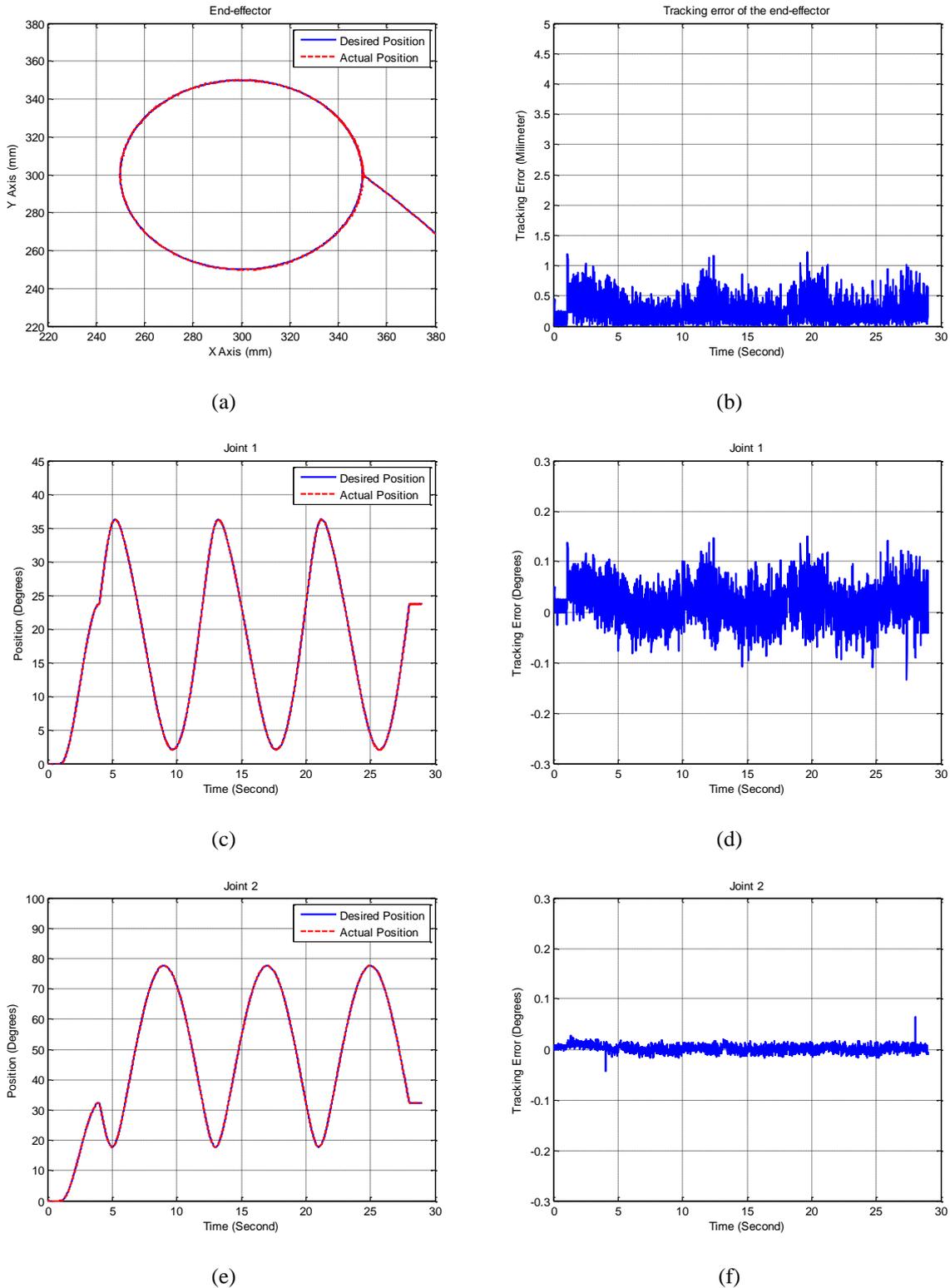


Figure 16. Desired and actual positions of a) end effector, c) first joint, e) second joints and trajectory tracking errors b) end-effector, d) first joint, f) second joints for SMC

MRSE and MAE cost function values obtained from PID, FLIC and SMC for Experiment-I, I, III and IV are given in Table 4 and Table 5, respectively, in degrees. According to Table 4, when end-effector tracks the trajectory without load, minimum tracking errors (in degrees) take

place for all of the controllers such as 0.0271 for PID Control, 0.0187 for FLIC and 0.0216 for SMC. If the end-effector load is increased, tracking errors will also increase for all of the controllers. If the end-effector load is less than 200gr, SMC generates less tracking error than PID controller. If the end-effector load is greater than 200gr, PID controller produces less tracking error than SMC. It can be concluded that FLIC is the best control algorithm compared to PID and SMC since the end effector with FLIC follows the circular Cartesian trajectory with minimum tracking error for all of the experiments (I, I, III and IV).

Table 4. MRSE cost function values obtained from PID, FLIC and SMC in degrees

Experiments	Load Value	PID	FLIC	SMC
Experiments – I	<i>No Load</i>	0.0271	0.0187	0.0216
Experiments – II	100 gr	0.0289	0.0217	0.0274
Experiments – III	200 gr	0.0300	0.0234	0.0348
Experiments – IV	300 gr	0.0342	0.0276	0.0481

Table 5. MAE cost function values obtained from PID, FLIC and SMC in degrees

Experiments	Load Value	PID	FLIC	SMC
Experiments – I	<i>No Load</i>	0.0296	0.0201	0.0238
Experiments – II	100 gr	0.0320	0.0235	0.0301
Experiments – III	200 gr	0.0336	0.0255	0.0395
Experiments – IV	300 gr	0.0385	0.0302	0.0566

CONCLUSIONS

Initially, trajectory tracking performances of classical FLC and PID algorithms are compared for 2-DOF PRM in vertical plane. This comparison showed that PID algorithm generates better performance than classical FLC especially when direction of the manipulator links is changed from reverse to the same direction with gravitational force. That’s why the integral component is added to the classical FLC. In the new condition, PID, FLIC and SMC control algorithms are compared for trajectory control of 2-DOF PRM in vertical plane. The actual mechanical architecture of 2-DOF PRM has imperfect transmission system that generates a challenge for robot control test platform in comparing the robustness of the control algorithms. PSO

algorithm, which can provide better trajectory tracking performance compared to analytical methods, has successfully optimized the parameters of PID, FLIC and SMC control algorithms. Although 2-DOF PRM has imperfect mechanism its trajectory tracking performance is obtained so close to advance robotic systems with expensive drivers and control cards.

In this study, four different experiments are carried out using the same circular Cartesian trajectory to test the robustness of the control algorithms that consider trajectory tracking errors as performance criteria. MRSE and MAE are selected as cost functions since they have been extensively used in control problems. Experiments produced the following results: i) if the end-effector load is increased, tracking errors will also gradually increase for all of the controllers, ii) the second joint of 2-DOF PRM produces less tracking error than the first joint for all of the controllers, and finally, iii) FLIC is the best control algorithm compared with PID and SMC since the end effector with FLIC controller follows the circular cartesian trajectory minimum tracking error for all of the experiments.

REFERENCES

- Acob, JM., Pano, V., Ouyang, PR. 2013. Hybrid PD sliding mode control of a two degree-of-freedom parallel manipulator. International Conference on Control and Automation, 12 – 14 June, HANGZHOU p.1760-1765.
- Baccouch, M., Dodds, S. 2020. Two-Link robot manipulator: simulation and control design. International Journal of Robotic Engineering, 5(2): 1-17.
- Bingul, Z., Karahan, O. 2011. A Fuzzy Logic Controller tuned with PSO for 2 DOF robot trajectory control. Expert Systems with Applications, 38(1): 1017–1031.
- D'ippolito, F., Alonge, F, Cucco, E. 2014. Contact estimation in robot interaction. International Journal of Advanced Robotic Systems, 11(96): 1-10.
- Driankov, V., Hellendoorn, H., Reinfrank, M. 1996. An Introduction to Fuzzy Control. 2nd ed. Springer Publication Berlin, Germany, Pg: 115-126.
- Elkhateeb, NA., Badr, RI. 2017. Novel PID tracking controller for 2DOF robotic manipulator system based on artificial bee colony algorithm. The Journal of Riga Technical University, 13(1): 55-62.
- Garcia-Rodriguez, R., Parra-Vega, V. 2012. Cartesian sliding PID control schemes for tracking robots with uncertain Jacobian. Transactions of the Institute of Measurement and Control, 34(4): 1-15.

Gong, C., Yuan, J., Ni, J. 2000. Non-geometric error identification and compensation for robotic system by inverse calibration. *International Journal of Machine Tools & Manufacture*, 40: 2119–2137.

Ilgen, S., Durdu, A., Gulbahce, E., Cakan, A. 2021. Optimal tuning of the SMC parameters for a two-link manipulator co-simulation control. *Elektronika Ir Elektrotechnika*, 27(5): 4-10.

Jang, H., Kim, SH., Kwak, YK. 2001. Calibration of geometric and non-geometric errors of an industrial robot. *Robotica*, 19(3): 311–321.

Kennedy, J., Eberhart, R. 1995. ‘Particle swarm optimization. *International Conference on Neural Networks*, 27 November – 1 December, WASHINGTON p.1942-1948.

Kormushev, P., Demiris, Y., Caldwell, DG. 2015. Kinematic-free position control of a 2-DOF planar robot arm. *International Conference on Intelligent Robots and Systems*, 17 December, HAMBURG p. 5518-5525.

Kuo, BC. 2009. *Automatic Control Systems*, 9th ed. Prentice-Hall A Simon & Schuster Company Englewood Cliffs New Jersey, USA.

Mavroidis, C., Dubowsky, S., Drouet, P., Hintersteiner, J., Flanz, J. 1997. A systematic error analysis of robotic manipulators: Application to a high performance medical robot. *International Conference on Robotics and Automation*, 25 April, NEW JERSEY p.980-985.

Nagaraj, B. Vijayakumar, P. 2011. A comparative study of PID controller tuning using GA, EP, PS and ACO. *Journal of Automation Mobile Robotics and Intelligent Systems*, 5(2): 42-48.

Nagesh, SB., Lendek, ZS., Khalate, AA., Babuska, R. 2012. Adaptive fuzzy observer and robust controller for a 2DOF robot arm. *International Conference on Fuzzy Systems*, 10 – 15 June, BRISBANE p.1-7.

Osypiuk, R, Finkemeyer, B., Skoczowski, S. 2006. Simple two degree of freedom structures and their properties. *Robotica*, 24(3): 365-372.

Ozkan, B. 2018. Guidance and control of a planar robot manipulator used in an assembly line. *Transactions of the Institute of Measurement and Control*, 40(2): 389-399.

Panjehfouladgaran, H., Yusuff R., Hong TS., Homayouni, SM. 2010. Qualitative performance measurement of supply chain management using fuzzy logic controller. *The 11th Asia Pacific Industrial Engineering and Management Systems Conference*, 7-10 December, MELAKA p.1-6.

Parra-Vega, V., Arimotor, S., Liu, Y., Hrizinger, G., Akella, P. 2003. Dynamics sliding PID control for tracking of robot manipulators theory and experiments. *IEEE Transactions on Robotics and Automation*, 19(6): 967-976.

Poli, R. 2008. Analysis of the publications on the applications of particle swarm optimization. *Journal of Artificial Evolution and Applications*, 2008: 1-10.

Queen, MPF., Kumar, MS., Aurtherson, PB. 2012. An effective torque minimization and precise position tracking control of robot manipulators using genetic algorithms. *European Journal of Scientific Research*, 72(2): 263-272.

Rao, DH., Saraf, SS. 1996. Study of defuzzification methods of fuzzy logic controller for speed control of a DC motor. *International Conference on Power Electronics, Drives and Energy Systems for Industrial Growth*, 2: 782-787.

Reyes, F., Kelly, R. 2001. Experimental evaluation of model-based controllers on direct-drive robot arm. *Mechatronics*, 11(3): 267-282.

Reyes, F., Rosado, A. 2005. Polynomial family of PD-type controllers for robot manipulators. *Control Engineering Practice*, 13(4): 441-450.

Sharkawy, AB., Othman, MM., Khalil, AMA. 2011. A robust fuzzy tracking control scheme for robotic manipulators with experimental verification. *Intelligent Control and Automation*, 2(2): 100-111.

Sharkawy, AB., Koustoumpardis, PN. 2019. Dynamics and computed-torque control of a 2-DOF manipulator: Mathematical analysis. *International Journal of Advanced Science and Technology*, 28(12): 201-212.

Simon, D. 2002. Training fuzzy systems with the extended Kalman filter. *Fuzzy Sets and Systems*, 132(2002): 189-199.

Slotine, JJ. 1985. The robust control of robot manipulators. *The International Journal of Robotics Research*, 4(2): 49-64.

Sousa, GCD., Bose, BK. 1994. A fuzzy set theory based control of a phase-controlled converter DC machine drive. *IEEE Transactions on Industry Applications*, 30(1): 34-44.

Sunthonga, A., Assawinchaichoteb, W. 2016. Particle swam optimization based optimal PID parameters for air heater temperature control system. *Procedia Computer Science*, 86: 108-111.

Utkin, VI. 1977. Variable Structure Systems with Sliding Modes, *IEEE Transactions on Automatic Control*, AC-22(2): 212-222.

Wang, H., Zhang, J., Dai, Y., Qu, J. 2010. Optimal design for 2-DOF PID regulator based on PSO algorithm. *Advances in Swarm Intelligence - Lecture Notes in Computer Science*, 6146: 515-522.

Wang, X., Hou, B. 2018. Trajectory tracking control of a 2-DOF manipulator using computed torque control combined with an implicit lyapunov function method. *Journal of Mechanical Science and Technology*, 32(6): 2803-2816.

Xue, D., Chen, Y., Atherton, DP. 2002. *Linear Feedback Control: Analysis and Design with Matlab*. Society for Industrial and Applied Mathematics, Philadelphia, USA, Pg. 183-196.

Young, KKD. 1978. A new controller design for a manipulator using theory of variable structure systems. *IEEE Transactions on Systems, Man and Cybernetics*, 8(2): 101-109.

Zadeh, LA. 1972. A rationale for fuzzy control. *Journal of Dynamic Systems, Measurements and Control*, 94(1): 3-4.

Zadeh, LA. 1973. Outline of a new approach to the analysis of complex systems and decision processes. *Transactions on Systems, Man, and Cybernetics*, SMC-3(1): 28-44.

Zakia, U., Moallem, M., Menon, C. 2019. PID-SMC controller for a 2-DOF planar robot. *International Conference on Electrical, Computer and Communication Engineering*, 7 – 9 February, BANGLADESH p.1-5.

Zhang, Y., Wang, S., Ji, G. 2015. A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, 2015: 1-38.