Novel m-PSO Optimized LQR Control Design for Flexible Link Manipulator: An Experimental Validation

Naveen Kumar[1*], Jyuti Ohri[1]
1- Department of Electrical Engineering, NIT Kurukshetra, India.
Email: naveenvermaindia@gmail.com(Corresponding author)

Received: August 2019 Revised: November 2019 Accepted: January 2020

ABSTRACT:
Recently, robotic manipulators are the key industry requirement. These have find the importance to enhance the productivity as well as accuracy. Furthermore, industries are also moving towards the use of Flexible Link Manipulator (FLM) owing to their unique characteristics i.e. light weight, high speed operations, and the larger workspace. The FLM system has flexibility of link that causes vibrations and oscillations which affect adversary to the performance of robotic arm. The performance of FLM system is measured w.r.t. minimum error and oscillations in trajectory tracking. In this research paper, an attempt has been made to overcome the complications of FLM system. A full state feedback Linear Quadratic Regulator (LQR), is designed for FLM. It is observed that the designed controller can enhance the accuracy of the robotic arm, while reducing oscillations and vibrations. In addition, to enhance the performance of controller and to reduce the hassle in terms of selecting the parameter of Q matrix in LQR, modified particle swarm optimization (m-PSO) is used. The effectiveness of designed controller is simulated in MATLAB. Further, the validation of designed controller is tested on hardware FLM device. The results obtained from the simulation and hardware are compared.
.

KEYWORDS: PSO, FLM, LQR, Vibration, Tracking Error.

1. INTRODUCTION

Robotics field involves the application of diverse disciplines such as physical, static and dynamic properties of materials, control theory, electronics, vision and signal processing, computer science. A robotic manipulator is basically a mechanical arm designed to work the similar task as of human arm. These are used for various industrial applications to perform repeated task precisely. These are consist of number of links and joints, rigid as well as flexible [1]. The Flexible Link Manipulator (FLM), received great attention in the past few decade among the researchers. For industrial application, it has shown various advantages over rigid manipulator in terms of light weight, high speed, low inertia, lower energy consumption and large work space [2]. The FLM has complex dynamic structure compared to rigid link and hence it becomes a difficult task for control engineers to design a control law. In case of industrial applications, preciseness about the given tasks, are always desirable. Therefore, control of position and oscillation becomes an important performance aspect, whereas in case of FLM, flexibility and the vibration presented in the link itself, leads to oscillations in the output. Hence, a control law is necessary to design with the objective to track the desired position with minimum or zero oscillations. Various authors have proposed different control strategies to follow the desired position or trajectories precisely. In [3], LMI and SMC based control law is design for FLM. PI and Fuzzy logic based controller for flexible joint are designed in [4]. Various other control strategies given in literature are variable structure control [5], optimal control [6], adaptive control [7], robust control [8], and intelligent based neural control [9], [10] etc. Among them, optimal control method is chosen in this work. In this, a full state feedback system is designed to find the gain of state feedback control. Simulation based LQR method has been designed for FLM in [11]. Furthermore, in LQR, Q matrix parameter selection is always a hectic task. In literature, it has been chosen based on research experience available and fined tuned. Therefore, for better tracking and stability, optimization algorithms may be employed such as Genetic algorithm (GA) [12], [13], Particle Swarm Optimization (PSO) [14], Ant Colony Optimization (ACO) [15] etc. In these, PSO has proved its capability over wide range of applications, in case of simulation as well as experimental work [16], [17]. Further, this PSO tuned LQR optimal control law is rarely used for flexible link manipulator simulation which can help largely to find the suitable parameter of Q matrix in LQR. It can enhance the tracking ability along with stability of FLM.

This paper presents the modelling and optimal control law (LQR) for Flexible Link Manipulator. Moreover, the regress process of selection of parameters of Q matrix is obliterated using optimization algorithm (PSO). Also, results have verified using experimental lab setup.

2. FLEXIBLE LINK MANIPULATOR

The single link Flexible Link (FLM) is major attraction of research in the field of robotics nowadays, due to its various advantages. The Lagrange Method has been used to develop the dynamics modelling FLM. It basically consist of flexible link module part and base part. The strain gauge, optimal encoder and potentiometer are fixed on clamped end of FLM module to measure tip deflection, shaft position and vibrations. The base part consists of DC motor and pinion gear system [18]. The schematic diagram is shown in Fig. 1.

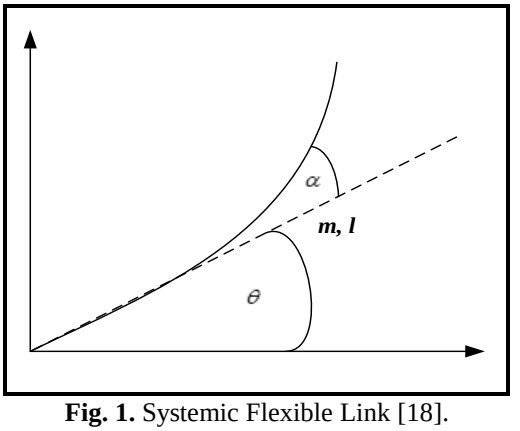

Fig. 1. Systemic Flexible Link [18].

The physical parameters considered while developing the mathematical dynamics of FLM are given in Table 1.

There are some nonlinearity presents in the system which make it complex, so some assumptions have been taken while developing the model as below:

- The link is long and slim hence rotary inertial and shear forces are nullified.
- The link progresses only in the horizontal plane. So, there is no effect of gravity on the link.
- The hub is rigidly attached by link.

The system model describing the motion is sufficient for tip point control of the link.

Let the total output angle:

$$y(t) = \theta(t) + \alpha(t) \tag{1}$$

Where, $\theta(t)$ and $\alpha(t)$ are position load angle and tip deflection

The potential energy (PE) of the FLM is because of its arm and can be calculated as:

$$V = \frac{1}{2} K_s \alpha^2(t) \tag{2}$$

Where, K_s represent the spring stiffness.

The combined kinetic energy (KE) is because of the rotational base and the link of flexible link manipulator. It is expressed as the following equation:

$$T = \frac{1}{2} J_{eq} \left(\frac{d\theta}{dt}\right)^2 + \frac{1}{2} J_{link} \left(\frac{d\theta}{dt} + \frac{d\alpha}{dt}\right)^2 \tag{3}$$

Where, J_{link} is a moment of inertia of the flexible link.

The damping forces are the corresponding τ_L non-conservative forces:

$$Q_\theta = \tau_L - B_{eq} \frac{d\theta}{dt} \tag{4}$$

$$Q_\alpha = B_{link} \frac{d\alpha}{dt} \tag{5}$$

Where, B_{link} is the viscous force of flexible link and can be neglected, which means $B_{link} = 0$, and τ_L is derived in equation (4).

Euler-Lagrange's equation can be expressed as:

$$\frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i \tag{6}$$

Where, $L = T - V$ and $q_i = [\theta \ \alpha]^T$ are the generalized coordinates; and Q_i = non-conservative force with $i = 1, 2$;

Model obtained by substituting equation (2) to (5) in Euler-Lagrange's equation (6) is linearized using Jacobian method further and the following linearized model obtained can be represented as the following linear state space form of equation (7) and (8).

$$\dot{x} = Ax + Bu \tag{7}$$

$$y = Cx + Du \tag{8}$$

Where, chosen state vector is $x = [\theta \ \alpha \ \dot{\theta} \ \dot{\alpha}]^T$ and

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{K_s}{J_{eq}} & -\frac{\eta_g K_g^2 \eta_m K_t K_m + B_{eq} R_m}{R_m J_{eq}} & 0 \\ 0 & -\frac{K_s (J_{eq} + J_{link})}{J_{eq} J_{link}} & \frac{\eta_g K_g^2 \eta_m K_t K_m + B_{eq} R_m}{R_m J_{eq}} & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{\eta_g K_g \eta_m K_t}{R_m J_{eq}} \\ -\frac{\eta_g K_g \eta_m K_t}{R_m J_{eq}} \end{bmatrix} \quad ; C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table 1. Description of various parameters.

l : link length (m)	Lm : Motor armature inductance (H)
m : Mass of link (Kg)	Rm : Motor armature resistance (Ohm)
fc : Natural frequency (Hz)	Beq : Viscous damping coefficient
Jl : Moment of inertia of link (Kg m2)	Kt : Motor torque constant (Nm/A)
Kg : Total gear ratio	Ks : Total stiffness of model (Nm/deg)
θ : Position of load angle (deg)	Eemf : Equivalent back emf (V)
η_m : Motor efficiency	Jeq : Equivalent moment of inertia of the hub (Kg m2)
η_g : Gearbox efficiency	θm : Motor shaft position (deg)
Tl : Load Torque (Nm)	Km : Motor back emf constant (V s/deg)
Tm : Motor Torque (Nm)	Vm : DC input voltage (V)
Jm : Motor Inertia (kg m2)	Im : Input current (A)
α : Tip deflection (deg)	

Table 2. System parameter nominal value.

Symbol	Description	Value
B_{eq}	Viscous damping coefficient of high-gear	0.004 N.m/(rad/s)
J_{eq}	Equivalent moment of inertia of high-gear	0.00208 kg.m^2
η_m	Motor efficiency	0.69
K_m	Back − emf constant	0.00768 V/(rad/s)
η_g	Gearbox efficiency	0.90
K_g	High gear total gear	70
R_m	Motor armature res	2.6 Ω
K_t	Stiffness constant	1.4
J_{link}	Moment of inertia c	0.004 kg.m^2

By substituting, the system parameters and numerical values from Table 2 in equation (7) & (8) and assuming all initial conditions zero, we get the state-space model as equation (9):

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 673.07 & -35.1667 & 0\\ 0 & -1023.07 & 35.1667 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 61.7325\\ -61.7325\end{bmatrix}[u]$$

$$\begin{bmatrix}y_1\\ y_2\\ y_3\\ y_4\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} \quad (9)$$

3. CONTROL LAW DESIGN

3.1. Stability Testing of FLM Model

The obtained linearized model is analysed for stability both in the time domain and frequency domain. To stabilize the system, the condition of controllability must be met which can be determined using Kalman controllability test. The controllability matrix for the flexible link model is given by:

$$[B\ \ AB\ \ A^2B\ \ A^3B] = \begin{bmatrix}0 & 61.73 & -2171 & 34794\\ 0 & -61.73 & 2171 & -13190\\ 61.73 & -2171 & 34794 & 237590\\ -61.73 & 2171 & -13190 & -997420\end{bmatrix} \quad (10)$$

The obtained controllability matrix is of full rank which ensures the system is controllable, hence the controller for stabilization and control of the FLM system at the desired point can be designed.

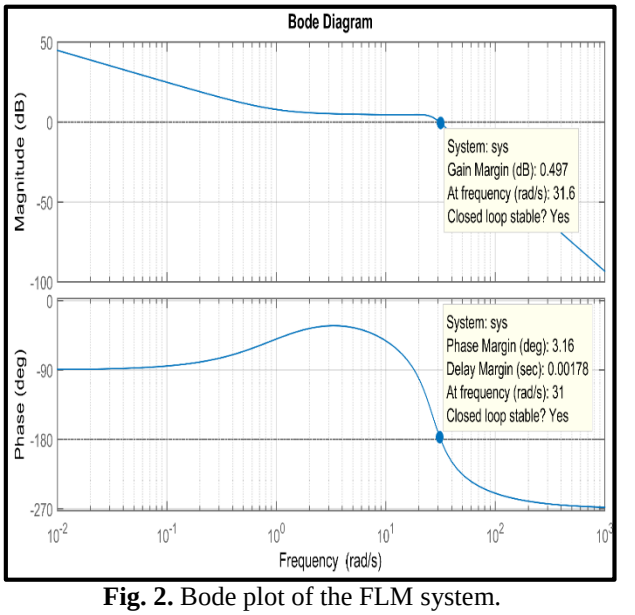

Fig. 2. Bode plot of the FLM system.

The Bode plot of the FLM is shown in Fig. 2. This plot displays the frequency response of FLM as magnitude (in dB) and phase (in degrees) plot. The Gain Margin of the system is obtained as 0.497 dB at its corresponding Phase cross over frequency of 31.6 rad/sec. The phase Margins is obtained as 3.16 degrees at its Gain cross over frequency of 31 rad/sec. Bode diagram shows that closed loop system is stable.

3.2. Linear Quadratic Regulator (LQR)

It is optimal control strategy, which provides a stable and robust state feedback controller. The basic requirement of LQR control is that the system must be controllable and observable. The working of LQR is based on selection of controller gains (Q and R) so as to minimize the performance index or cost function [11]:

For the state space model in equation (11)

$$\dot{x} = Ax + Bu \tag{11}$$

We require optimal control which is given by control law

$$u = -Kx \tag{12}$$

Where, K is Kalman gain.

Here, it has been assumed that all the states of FLM are measurable and attempt to achieve state variable feedback control. The block diagram of LQR is shown in Fig. 3.

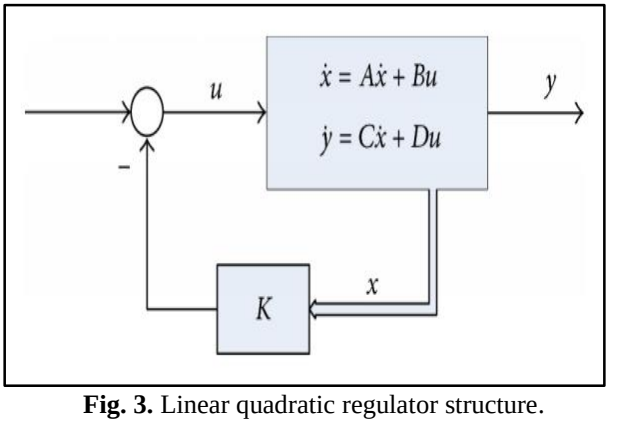

Fig. 3. Linear quadratic regulator structure.

This gives desirable closed-loop properties. The quadratic performance index (PI) is defined as (13)

$$J = \frac{1}{2}\int_0^\infty (x^T QX + u^T Ru)dt \tag{13}$$

Where, Q is diagonal weighing matrix and R is the input matrix. The requirement of Q and R matrices are positive semi-definite and positive definite. Substituting the state variable feedback control (12) in (13), we get

$$J = \frac{1}{2}\int_0^\infty x^T (Q + K^T RK)xdt \tag{14}$$

To find the optimal feedback K we proceed as follows. Suppose there exists a constant square matrix P such that:

$$\frac{d}{dt}(x^T Px) = -x^T (Q + K^T RK)x \tag{15}$$

Substituting in (14) yields:

$$J = -\frac{1}{2}\int_0^\infty \frac{d}{dt}(x^T Px)dt = \frac{1}{2}x^T(0)Px(0) \tag{16}$$

The performance index (J) can be minimized when gain K is chosen as (17)

$$K = R^{-1}B^T P \tag{17}$$

The matrix P is determined from the solution of Algebraic Riccati Equation (18), given as:

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{18}$$

The LQR problem to find the state variable feedback gain such that quadratic performance index is minimized and states go towards origin with minimum control efforts is called linear quadratic regulator problem. The design process to obtain the K is as follows:

- Firstly, the selection of matrices Q *and* R by trial and error methods, so that our design requirements are met.
- Then obtain the solution of the P matrix by algebraic Ricatti equation.
- Then find the state feedback control K.

1) LQR design for FLM

The objective of control scheme design specifications of the FLM is that it should provide stability and also ensure desired trajectory tracking and link vibration must be minimum.

Following Q and R weighing matrices are chosen initially by trial and error procedure as following with Q and R chosen as equation (19)

$$Q = \begin{bmatrix} 112 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and R=1} \tag{19}$$

Using the LQR method, performance index (J) is minimized and state feedback gain K is calculated as equation (20):

$$K = [10.5830 \quad -10.9735 \quad 1.1003 \quad -0.0139] \tag{20}$$

3.3. Particle Swarm Optimization (PSO)

1) Basic Concept of PSO

The concept of PSO was introduced by J. Kennedy

et. al. in [19]. It is inspired by social behaviour of birds in a specific spectrum during their flying in the sky. Their way of flying together in a spectrum shows there is some communication between them by which they can interact with each other and follow the position of a member who is very close to destination [20], [21].

This algorithm is a population based search algorithm in which the particle tends to converge towards best optimal solution in a defined search space. Each particle moves with a velocity which dynamically changes according to particle's previous best experience and along with its neighborhood best position in the previously visited search space [22], [23]. Velocity and position of each particle is updated according to the current position and velocity locally in *pbest* and globally in *gbest*. Movement of particles in two dimensional problem hyperspace and update in velocity and position is shown in Fig. 4.

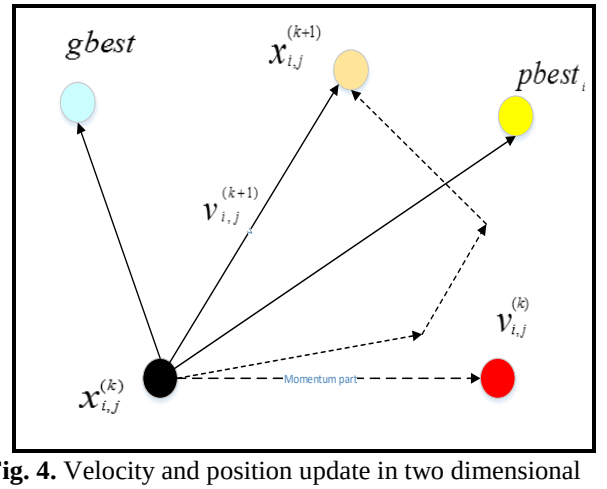

Fig. 4. Velocity and position update in two dimensional problem space.

Velocity and position update formulas are given as (21) and (22) respectively [24].

$$v_{i,j}^{(k+1)} = w * v_{i,j}^{(k)} + c1 * rand(1) * \left(pbest_{i,j} - x_{i,j}^{(k)}\right) + c2 * rand(2) * \left(gbest_j - x_{i,j}^{(k)}\right) \quad (21)$$

$$x_{i,j}^{(k+1)} = x_{i,j}^{(k)} + v_{i,j}^{(k+1)} \quad (22)$$

Where, i=1, 2, …….., n; j=1, 2, ……., m and
k : 1, 2, ……., t
n : Number of particles in swarm
m : Dimension of the search space
t : Maximum iterations
$v_{i,j}^{(k)}$: The j^{th} component of the velocity of particle i at k^{th} iteration k
if $v_{i,j}^{(k)} > v_{max}$ then $v_{i,j}^{(k)} = v_{max}$
else if $v_{i,j}^{(k)} < -v_{max}$ *then* $v_{i,j}^{(k)} = -v_{max}$
$x_{i,j}^{(k)}$ j^{th} component of the position of particle i at iteration k
w : Inertia weight factor
$c1, c2$: Acceleration factors
rand(1) : Random number in between 0 to 1
rand(2) : Random number in between 0 to 1
$pbest_{i,j}$: Best position of i^{th} particle at iteration k
$gbest_j$: Best position of the swarm until iteration k

2) Modifications in Particle Swarm Optimization

The significance of the various component of velocity update equation given in equation (21) is as following:

- The first component of this equation is known as "inertia", which is scaled by weighting factor 'w'. The tendency of a particle to move in the same direction is moving.
- Second component of this equation is a linear attraction towards the best position of the i^{th} particle which is scaled by acceleration constant $c1$. This component is known as "memory".
- Third component is linear attraction towards the best position found by any particle in the swarm which is scaled by $c2$ and is referred as "cooperation" or "group knowledge".

Based on the above it can be influenced that the selection of the parameters 'w', $c1$ and $c2$ greatly affects the performance of the PSO. Hence to improve the performance of PSO optimizer, researchers have proposed various modification over the time. These modification have shown significant effect on optimizing performance. These modifications are mainly based on various components in velocity update equation explained as below [22], [24].

Modification-I: PSO with Time Varying Acceleration Constant (PSO-TVAC)

In the PSO, the search toward the optimum solution is guided by the two stochastic acceleration components (the cognitive component c_1 and the social componentc_2). Therefore, proper selection of these two components is very important to find the optimum solution accurately and efficiently. Improvement in the PSO is proposed in which cognitive component is reduced with time and the social component is increased with time, by changing the acceleration coefficients. With a large cognitive component and small social component at the beginning, particles are allowed to move around the search space, instead of moving toward the population best. On the other hand, a small cognitive component and a large social component allow the particles to converge to the global optima in the latter

part of the optimization. This method is known as PSO-TVAC method and its mathematical formulation is given in (23) for selection of $c1$ and $c2$ [25]

$$c_1 = \left(c_{1f} - c_{1i}\right)\frac{iter}{iter_{max}} + c_{1i} \quad (23)$$

$$c_2 = \left(c_{2f} - c_{2i}\right)\frac{iter}{iter_{max}} + c_{2i} \quad (24)$$

Where, $c_{1i,}\, c_{1f},\ c_{2i}\ and\ c_{2f}$ are constants, $iter$ is the current iteration number and $iter_{max}$ is the maximum number of allowable iterations. An improved optimum solution for most of the benchmarks was observed when changing c_1 from 2.5 to 0.5 and changing c_2 from 0.5 to 2.5, over the full range of the search [25], [26].

Modification II: Defining the Range of Velocity

In addition to above, [27] has proposed the range of velocity v_{max} and v_{min}, as given in equation (25) and (26)

$$v_{max} = 0.1 * (k_{max} - k_{min}) \quad (25)$$

$$v_{min} = -0.1 * (k_{max} - k_{min}) \quad (26)$$

Where, k_{min} and k_{max} are lower and the upper limits respectively for the parameter to be optimized. Selection of initial values for these parameters are considered as given in (27) and (28).

$$v_{inital} = v_{min} + (v_{max} - v_{min}) * \epsilon \quad (27)$$

$$p_{inital} = v_{min} + (v_{max} - v_{min}) * \epsilon \quad (28)$$

Where, $v_{initial}$ and $p_{initial}$ are initial velocity and the position, ϵ is random number matrix.

Modification III: Inertia weight coefficient

The selection of weight 'w' would help in quick finding of optimal result among the population. In [24], [28] Shi and Eberhart, came up with concept of variable weight 'w' and in [27] and [28] applied for various applications successfully. This modification also reduces the local search time and total time of convergence. Here w is introduced as inertia weight nonlinearity decreasing with the iteration defined as (29).

$$w = (w_{max} - w_{min}) - \left(\frac{iter_{max} - iter}{iter_{max}}\right) * w_{min} \quad (29)$$

Where,

- w : The inertia weight nonlinearly decreasing with the iterations
- w_{max}: The upper limit of inertia weight selected as *0.9*
- w_{min}: The lower limit of inertia weight selected as *0.4*
- $iter_{max}$: The current iteration and $iter_{max}$ is the maximum number of iteration

3) Optimal parameter of LQR design

To minimize the error signal *e(t),* possible formulation of fitness are tabulated in the Table 3 [14], [29].

Table 3. Performance Estimation of PID Controller [29].

Name of Criterion	Formula
Integral of the Absolute Error (IAE)	$IAE = \int_0^\infty \lvert e(t)\rvert dt$
Integral of the square Error (ISE)	$ISE = \int_0^\infty e^2(t)dt$
Integral of the Time-weighted Square of Error (ITSE)	$ITSE = \int_0^\infty t * e^2(t)dt$
Integral of the Time-weighted Absolute Error (ITAE)	$ITAE = \int_0^\infty t * \lvert e(t)\rvert dt$

The ITAE performance index has advantages of producing smaller overshoots and oscillation than the IAE index or the ISE Overall, ITAE is the most sensitive among the three criteria. ITSE is also sensitive but it is not best suitable for computation [30], [31]. So ITAE is chosen as fitness function in this work.

The PSO is employed for the optimal selection of Q and R design parameters of LQR, to enhance its performance for the control of flexible link manipulator arm.

Lower bound for Q= [100 0 0 1] Upper bound for Q= [150 5 1 8].

The following steps for selecting the optimal parameters:

Step I: Firstly, population size, position and velocity of particles, number of iteration and size of search space is to be initialized.

Step II: Each particle's best position *p* is denoted as local best *pbest*. The best value among all the local best *pbest* is termed as global best *gbest*.

Step III: Then fitness *fp* of each particle is evaluated.

Step IV: This evaluated fitness is to be compared with its *pbest* value. If this *pbest>fp* then set *pbest = fp*. The current position coordinates of particle *xp* is to be updated to particles best coordinates for best fitness and $best_{xp}$ is the coordinates corresponding to particle *p's* best fitness so far.

Step V: Value of objective function is evaluated for each particle position. The *pbest* is to update with better position which is encountered during the iteration. Further, as defined in step I, best value among *pbest* is termed as *gbest*. If during the iteration, a better value is encountered then replace the previous *gbest* value to newly encountered value.

If *gbest >fp,* then set *gbest = fp,* where *gbest* denotes the best fitness value among all particles in swarm.

Step VI: Next, update the inertia weight, location and velocity of the particles according to equations (14) and (15), respectively [22], [23].

Step VII: Step II to VI be repeated until stopping criterion is met.

Step VIII: Use the obtained value in simulation model of FLM and check whether the desired responses are achieved.

Step IX: Step II to XI until desired output is archived.

Employing the said algorithm, obtained Q matrix is,

$$Q = \begin{bmatrix} 145 & 0 & 0 & 0 \\ 0 & 4.1 & 0 & 0 \\ 0 & 0 & 1.05 & 0 \\ 0 & 0 & 0 & 7.1 \end{bmatrix} \quad \text{and} \quad R=1 \quad (30)$$

And using this with performance index (J) along with *Q* and *R* matrices, and using equation 19, state feedback gain K could be calculated as:

$$K = [12.0416 \quad -37.3209 \quad 1.6291 \quad -0.9073] \quad (31)$$

4. EXPERIMENTAL SET-UP

4.1. FLM Complete System Set-up

The FLM system consists of a computer system, rotatory link system, data acquisition and voltage amplifier interconnections. The Simulation model, has been prepared in computer system with following configuration as: Dell i7processor, 8GB RAM. MATLAB version 2017a and QUARC software, has been used for designing the model and control law. First, the designed control law, is tested using simulation model in QUARC ® and MATLAB, followed by experimental verification using the hardware. The hardware FLM module is connected with the system via Data Acquisition and Voltage amplifier, to drive the device smoothly as shown in Fig. 5.

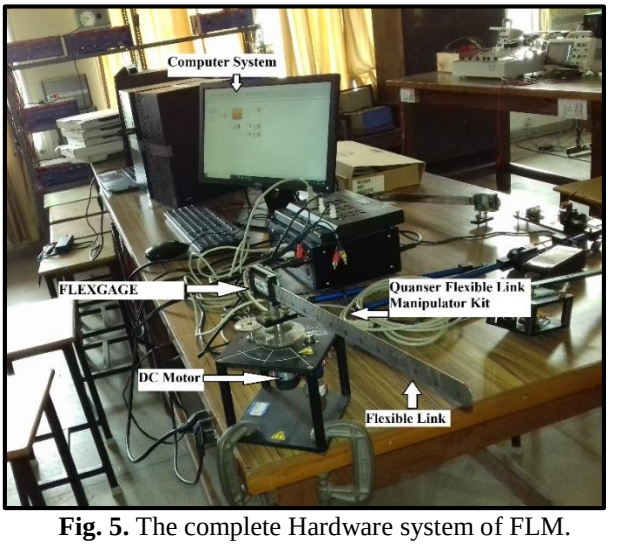

Fig. 5. The complete Hardware system of FLM.

4.2. Flexible Link Module

The flexible link module is made-up of base, DC motor, pinion gear system. It has on-board mounted optical encoder and potentiometer to measure the shaft position, a tachometer to measure the speed of the motor. QUANSER's Rotary Servo Base Unit (SRV02) and stainless steel link has been used for present experiment shown in Fig. 6.

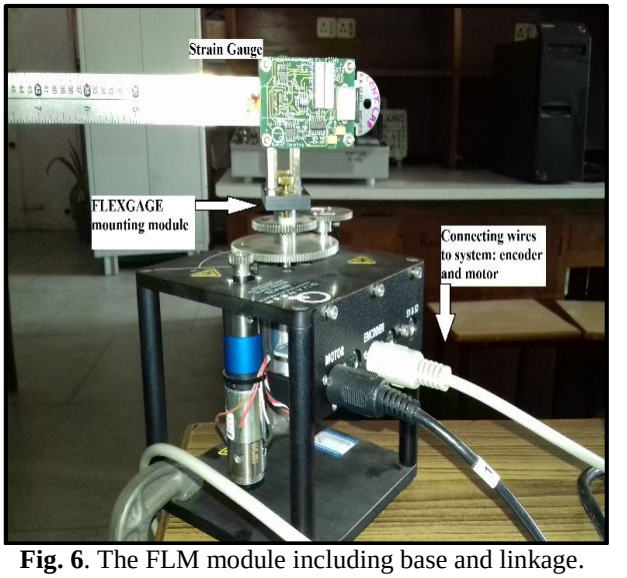

Fig. 6. The FLM module including base and linkage.

4.3. Data Acquisition & Voltage Amplifier

In this experiment, a QUANSER Q8-USB device has been used for easy and smooth data acquisition. It has low I/O conversion time compared to other devices and can achieve up to 2kHz close loop control rate. For regulated and amplified power supply, QUANSER VoltPAQ X1 amplifier has been used. It regulates ±12V, 1A DC supply as shown in Fig. 7.

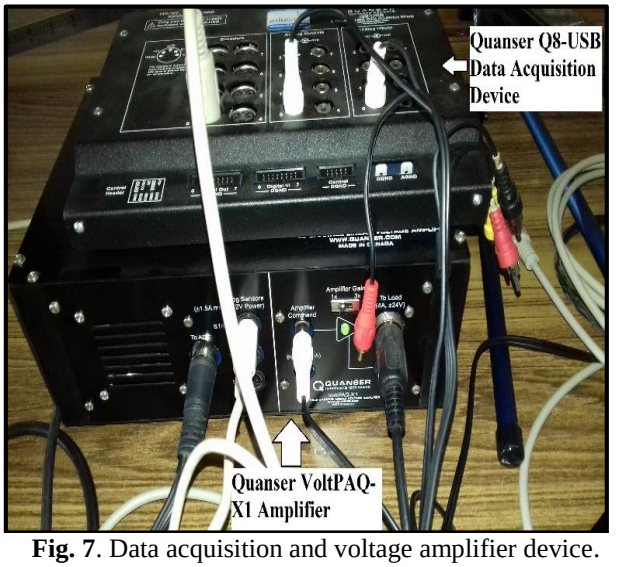

Fig. 7. Data acquisition and voltage amplifier device.

5. SIMULATION RESULT AND EXPERIMENTAL VALIDATION

5.1. Simulation Result of Control of FLM

The LQR is applied for the portion and tip deflection control of FLM where state space model is given in (9). The state feedback gain is calculated in (17) using the solution of Algebraic Riccati Equation (ARE) of (18) and control input (u) is obtained from (12). Reference input is chosen as square wave signal.

Following cases has been investigated to study the performance of the controller.

- Manual tuning of LQR
- PSO tuned LQR

5.1.1 Simulation result for manual tuning of LQR

The LQR has been applied for position and tip deflection control of FLM. The state feedback gain K is calculated using ARE of (18) and manual fine-tuned given in equation (19) and (20) as given below:

$$Q = \begin{bmatrix} 112 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad R=1$$

$$K = [10.5830 \quad -10.9735 \quad 1.1003 \quad -0.0139]$$

The response obtained of angular position θ vs time (t), tip deflection angle $\alpha(t)$ are shown in figure below:

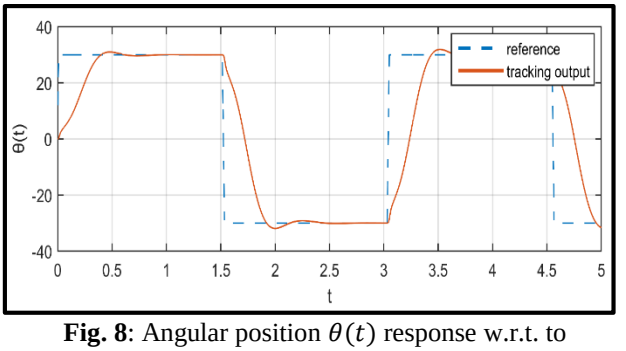

Fig. 8: Angular position $\theta(t)$ response w.r.t. to reference input.

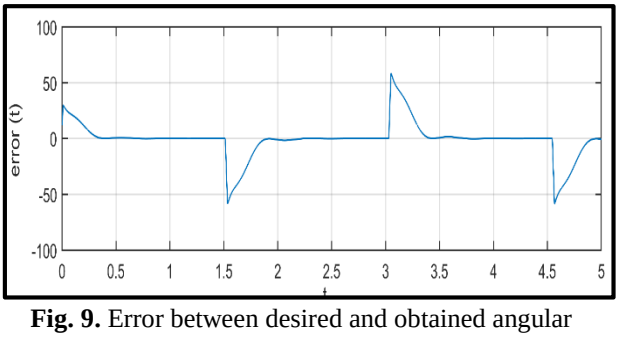

Fig. 9. Error between desired and obtained angular position $\theta(t)$.

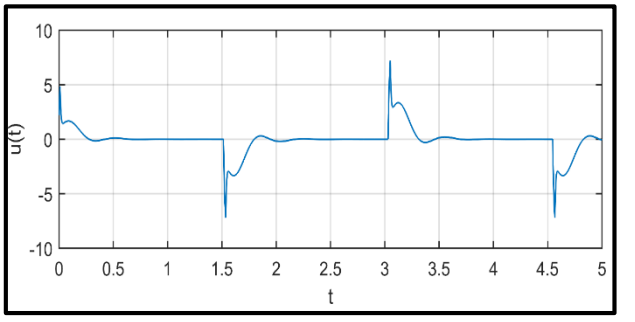

Fig. 10. Controlled voltage $u(t)$ to the position.

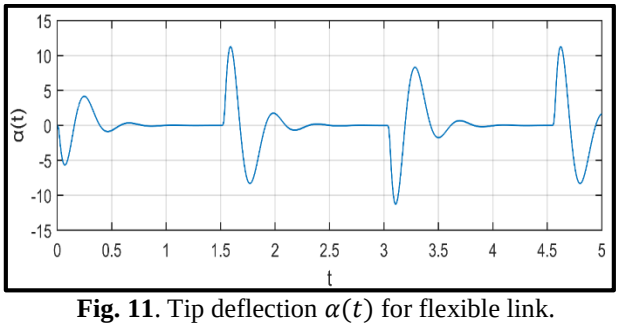

Fig. 11. Tip deflection $\alpha(t)$ for flexible link.

The results in Fig. 8 and the angular position $\theta(t)$ illustrate that the desired path is approximately tracked by the FLM arm after every deflection. The error between desired trajectory and actual trajectory is shown in Fig. 9. After every deflection, the manipulator tracks the new desired position within certain period and hence the tracking error approaches to zero as shown in Fig. 10. Next, the actuator (motor) are the key part to provide the motion in FLM. These actuator are controlled by controller by means of control voltage. Therefore, the control voltage has an important role to play. After every deflection, controller has to provide the control efforts based on error value, in terms of control voltage.

Actuator work is provided by motor in the system. So, the control voltage $u(t)$ should be minimum. Here, Figs. 10 and 11 shows stable control action voltage and tip deflection for flexible link. Table 4 shows various time response parameters of FLM.

Table 4. Time domain parameter (TDP) of FLM using LQR.

Sr. No	Peak Overshoot (in sec)	Rise Time (in sec)	Settling Time (in sec)
θ	31	0.3	0.37
u	7.2	0.01	0.25
α	11.3	0.05	0.6
Error	58.5	0.01	0.35

5.1.2 Simulation result using m-PSO tuned LQR

In this section, modified-PSO has been used to find

the optimal parameter of matrix Q and to eliminate the regress process of choosing Q & K matrix parameter. The parameters are obtained using the process given in section III (C) and corresponding parameters are given in equation 30 & 31. LQR controller has been simulated with these parameters and corresponding results are displayed in following figures.

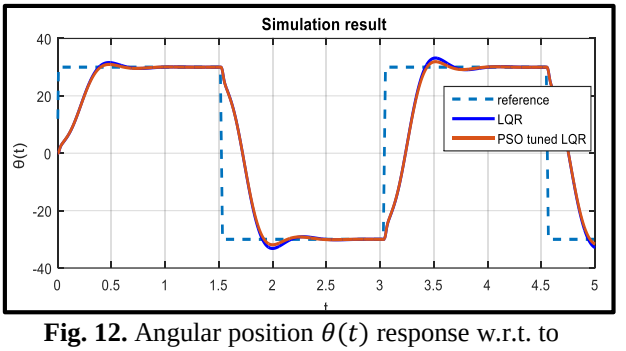

Fig. 12. Angular position $\theta(t)$ response w.r.t. to reference input.

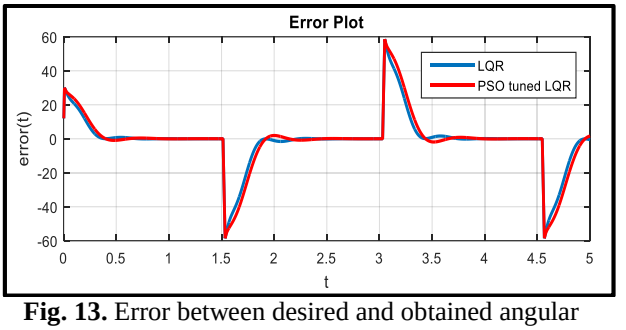

Fig. 13. Error between desired and obtained angular position $\theta(t)$.

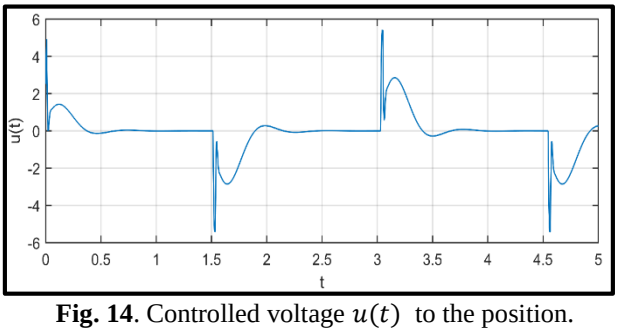

Fig. 14. Controlled voltage $u(t)$ to the position.

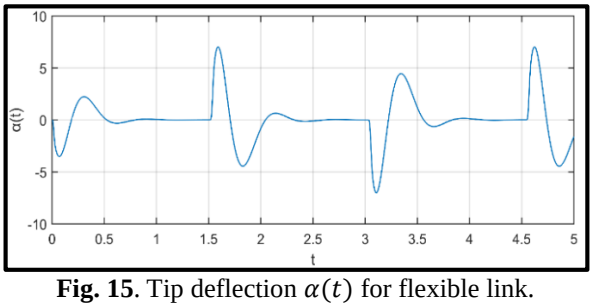

Fig. 15. Tip deflection $\alpha(t)$ for flexible link.

Using optimal selection of parameters of LQR, it has been observed from Fig. 12 that, FLM is tracking the desired input trajectory more accurately as compared to Fig. 11. The tracking error has been compared in Fig. 13, with LQR design error. Figs. 12 and 13 also show stable control action voltage and minimal tip deflection respectively. These results are shown in Table 5.

Table 5. TDP of FLM using PSO tuned LQR.

	Peak Overshoot (in sec)	Rise Time (in sec)	Settling Time (in sec)
θ	30	0.27	0.35
u	5.4	0.01	0.35
α	7	0.09	0.6
Error	59	0.01	0.5

By comparing the parameters of Table 4 and 5, it has been observed that in case of settling time and rise time, much difference is not identified. But for the case of peak overshot, all three measures i.e. θ, u, and α have been reduced in case of PSO tuned LQR as compared to LQR.

5.2 Experimental Validation of result for FLM

5.2.1 Experiment result using LQR

In this section, control law designed for simulation model, is validated using laboratory hardware set-up of FLM model. The control action voltage is measured using data acquisition setup. The angular position and tip deflection are measured using strain gauge sensor installed on FLM kit as shown in Fig. 6. First, model is tested with initial Q=diag ([112 1 1 1]) and corresponding result are displayed in following figures for *theta*, *alpha* and *u*.

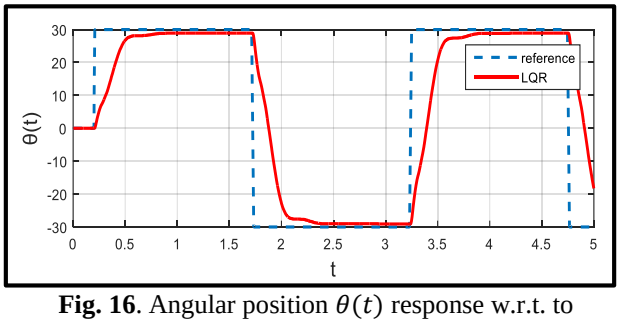

Fig. 16. Angular position $\theta(t)$ response w.r.t. to reference input.

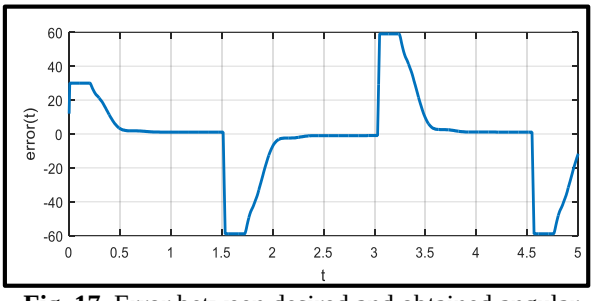

Fig. 17. Error between desired and obtained angular position $\theta(t)$.

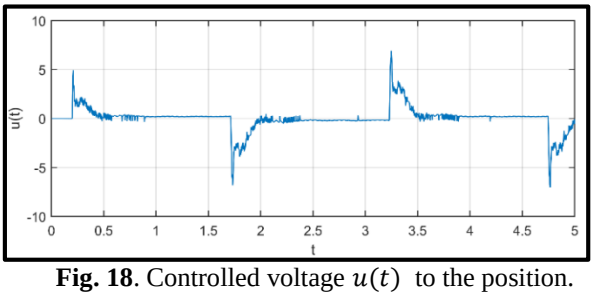

Fig. 18. Controlled voltage $u(t)$ to the position.

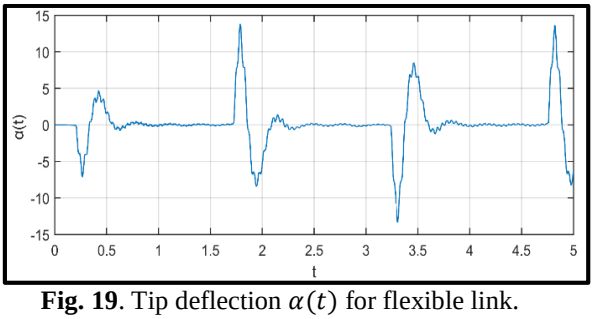

Fig. 19. Tip deflection $\alpha(t)$ for flexible link.

Although, there is slightly error in tracking as shown in Figs. 8 & 16 between simulated model and experimental model respectively, but overall the system tracks the desired trajectory with designed control law successfully. This validates the simulated model of LQR control law working in practical environment.

Table 6. TDP of FLM hardware kit using LQR.

Sr. No	Peak Overshoot (in sec)	Rise Time (in sec)	Settling Time (in sec)
θ	29	0.5	0.6
u	7	0.21	0.9
α	13.5	0.26	0.6
Error	59	0.01	0.8

The quantized results obtained in Table 6 show the experimental.

5.2.2 Experiment result using m-PSO optimized LQR

Further, as given in section III (C), m-PSO has been employed to optimize the LQR design for FLM and corresponding parameter are given in equation 30 & 31. The effectiveness of simulated model is implemented on FLM hardware laboratory kit and corresponding result are displayed as following.

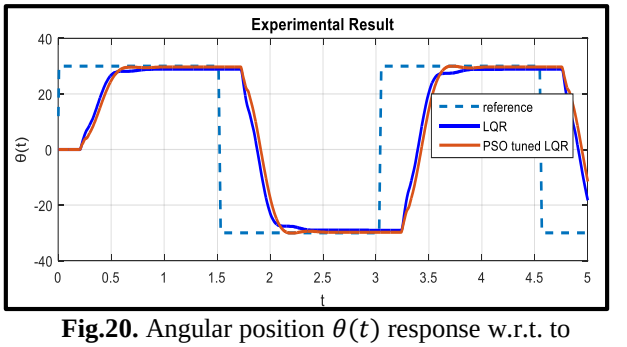

Fig.20. Angular position $\theta(t)$ response w.r.t. to reference input.

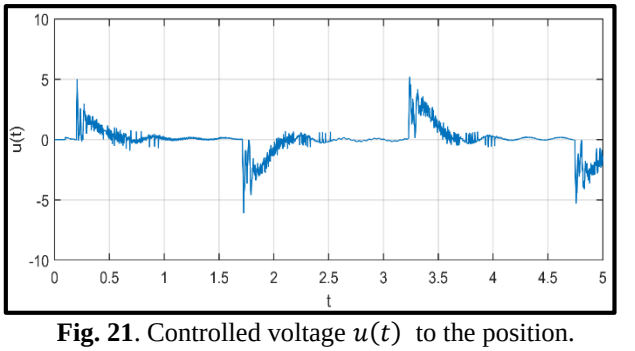

Fig. 21. Controlled voltage $u(t)$ to the position.

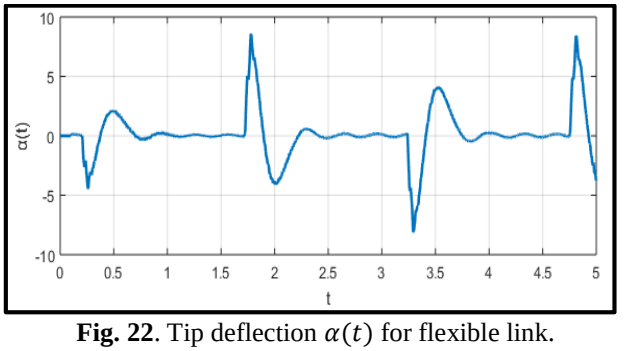

Fig. 22. Tip deflection $\alpha(t)$ for flexible link.

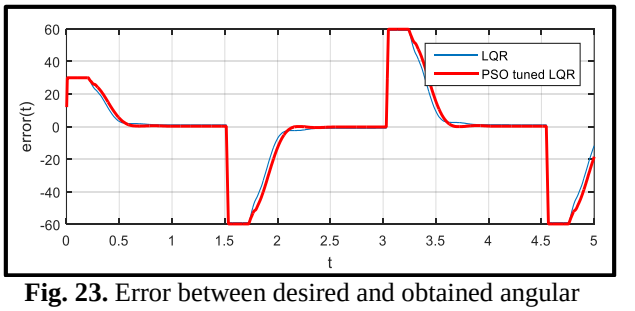

Fig. 23. Error between desired and obtained angular position $\theta(t)$ for LQR and PSO tuned LQR.

Figs. 20 and 23 show that m-PSO based LQR designed model is tracking the desired trajectory more precisely and faster than manually selected LQR based control law. The tip deflection and control voltage show stable result. So a better control strategy designed in Simulink has been validated experimentally.

Table 7. TDP of FLM kit using PSO tuned LQR.

Sr. No	Peak Overshoot (in sec)	Rise Time (in sec)	Settling Time (in sec)
θ	30	0.55	0.61
u	5.2	0.2	1
α	8.5	0.24	1
Error	60	0.01	0.6

6. CONCLUSION

Owing the various advantages in industries, flexible link manipulator is a complex in nature such as flexible and light weight of link. The vibration & oscillations occur in system due to said complexity, while tracking the trajectory. In this paper, LQR controller has been designed to minimize the tracking error, vibrations and oscillations. The LQR parameter are primarily tuned based on literature and manually fine-tuned to obtain the good result. In addition, problem arising in literature for selection of LQR parameters, has been eliminated by using modified PSO. Here first, a simulated model has been designed using m-PSO optimized LQR so as to minimize the vibrations and oscillation at the change in every deflection angle. The simulated designed model is further implemented in FLM laboratory experimental set-up to test the ability to work in real time. The working ability of designed simulated control law model is further tested and validated in laboratory experimental set-up. Further, M-PSO based LQR gives better result while following the desired trajectory in simulation as well as experiment.

7. ACKNOWLEDGMENT

The first author gratefully acknowledges the UGC New Delhi India for the financial support of this work.

REFERENCES

[1] M. Vidyasagar and M. W. Spong, **"Robot Dynamics and Control".** *Willey*, 2008.

[2] B. Subudhi and A. S. Morris, **"Dynamic Modelling, Simulation and Control of a Manipulator with Flexible Links and Joints B.,"** *Robot. Auton. Syst. 41*, Vol. 41, pp. 257–270 Dynamic, 2002.

[3] N. Singh and S. Rajendran, **"Integral Fast Output Sampling Control for Flexible Link Manipulators with LMI Approach,"** in *1st IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems, ICPEICES 2016*, No. 1, pp. 1–6, 2017.

[4] A. Dehghani and H. Khodadadi, **"Fuzzy Logic Self-Tuning PID Control for a Single-Link Flexible Joint Robot Manipulator in the Presence Of Uncertainty,"** in *Proceeding of 15th International Conference on Control, Automation and Systems*, No. Iccas, pp. 186–191, 2015.

[5] Ingole, Bandyopadhyay, and Gorez, **"Variable Structure Control Application for Flexible Manipulators,"** in *Proceedings of the IEEE Conference on Control Applications, New York, USA*, Vol. 2, pp. 1311–1316, 1994.

[6] M. Baroudi, M. Saad, and W. Ghie, **"State-feedback and Linear Quadratic Regulator Applied to a Single-Link Flexible Manipulator,"** in *IEEE International Conference on Robotics and Biomimetics, ROBIO 2009*, No. 2, pp. 1381–1386, 2009.

[7] J. H. Yang, F. L. Lian, L. C. Fu, and Abstract—As, **"Nonlinear Adaptive Control for Flexible-Link Manipulators,"** *IRE Trans. Circuit Theory*, vol. 13, No. 1, pp. 140–148, 1997.

[8] G. Song and L. Cai, "**A New Approach to Robust Position/Force Control of Flexible-Joint Robot Manipulators,"** *J. Robot. Syst.*, Vol. 13, No. 7, pp. 429–444, 1996.

[9] L. B. Gutierrez and F. L. Lewis, **"Implementation of a Neural Net Tracking Controller for a Single Flexible Link: Comparison with PD and PID Controllers,"** *IEEE Trans. Ind. Electron.*, Vol. 45, No. 2, pp. 307–318, 1998.

[10] N. Kumar and J. Ohri, **"SVM and Neural Network based Optimal Controller Design for Haptic Interface,"** in *1st International Conference on New Frontiers in Engineering, Science and Technology, NFEST 2018, New Delhi*, pp. 1–8, 2018.

[11] Y. Sharma and J. Ohri, **"LabVIEW based Linear Quadratic Regulator and Model Predictive Controller for DC Motor and Flexible Link Manipulator,"** in *Proceedings of 4th International Conference on Power, Control and Embedded Systems, ICPCES 2017*, Vol. 2017-Janua, pp. 1–7, 2017.

[12] J. Ohri, N. Kumar, and M. Chinda, **"an Improved Genetic Algorithm for PID Parameter Tuning,"** *Proc.of 2014 Int. Conf. Circuits, Syst. Signal Process.*, pp. 191–198, 2014.

[13] M. Patrascu and A. Ion, "**Nature-Inspired Computing for Control Systems",** Vol. 40. 2016.

[14] X. Wang, Y. Wang, H. Zhou, and X. Huai, "**PSO-PID: a Novel Controller for AQM Routers,"** in *Proceedings of IFIP International Conference on Wireless and Optical Communications Networks*, 2006.

[15] V. K. Singh and J. Ohri, **"Simultaneous Control of Position and Vibration of Flexible Link Manipulator by Nature-Inspired Algorithms,"** in *Proceedings of IEEE 8th Power India International Conference (PIICON)*, pp. 1–6, 2018.

[16] N. Kassarwani, J. Ohri, and A. Singh, **"Performance Analysis of Dynamic Voltage Restorer using Improved PSO Technique,"** *Int. J. Electron.*, Vol. 106, No. 2, pp. 212–236, 2019.

[17] A. Al-Mahturi and H. Wahid, **"Optimal Tuning of Linear Quadratic Regulator Controller Using a Particle Swarm Optimization for Two-Rotor Aerodynamical System,"** *Int. J. Electron. Commun. Eng.*, Vol. 11, No. 2, pp. 196–202, 2017.

[18] Quanser, **"Quanser Manuals, Rotary Flexible 2011."** [Online]. Available: http://www.quanser.com/.

[19] J. Kennedy and R. Eberhart, **"Particle Swarm Optimization,"** in *Proceedings of IEEE int. conf. on Neural Network*, 1995, Vol. IV, No. 1, pp. 1942–1948.

[20] J. Kennedy, **"The Particle Swarm: Social Adaptation of Knowledge,"** pp. 303–308, 2002.

[21] S. Panda and J. Yadav, **"Evolutionary echniques for model Order Reduction of Large Scale Linear Systems,"** *Int. J. Electr. Comput. Eng.*, Vol. 6, No. 9, pp. 1105–1111, 2012.

[22] G. K. Venayagamoorthy, J.-C. Hernandez, Y. del Valle, R. G. Harley, and S. Mohagheghi, **"Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems,"** *IEEE Trans. Evol. Comput.*, Vol. 12, No. 2, pp. 171–195, 2008.

[23] G. C. Goodwin, S. F. Graebe, and M. E. Salgado, **"Control System Design".** *University of Michigan: Prentice Hall*, 2001.

[24] Y. Shi and R. Eberhart, **"A Modified Particle Swarm Optimizer,"** pp. 69–73, 1998.