



Research Article

Investigation of feasible controller for position control of flexible joint manipulator using multiple control techniques

Subodh Kumar¹ · Kuldeep Jayaswal² · D. P. Kothari³

Received: 13 September 2019 / Accepted: 11 November 2019 / Published online: 19 November 2019
© Springer Nature Switzerland AG 2019

Abstract

With the advancement in technology, industries are moving from automation to robotics in the era of robotization. Different control techniques are being used in a different sector of production to control the whole machinery with special attention towards control the position and vibrations. In this research article, the feasible controller has been investigated for position control of flexible joint manipulator. For this, an effective version of a single linked flexible joint robotic manipulator has been used as a platform to control the position of manipulator using different control approaches. Euler's–Lagrange equations have been used to obtain the effective version of the system and the position control has been performed by proportional-integral-derivative (PID) controller, pole –placement and linear quadratic regulator (LQR) methods. The main target of this research work is to maintain the rotational angle of the joint at an appropriate position and to remove the fluctuations at a specific robotic tool which is known as end effectors. The gains of PID controller have been delineated with the help of genetic algorithm and tuned for obtaining the most suitable gains to control the position of flexible joint manipulator and vibrations. As the settling time of the system was high so genetic algorithm has been used to optimize it. The Matlab[®]/simulation results depict that genetic algorithm tuned PID controller performed better correlated to both pole placement and LQR method of control.

Keywords Flexible joint manipulator (FJM) · PID · Pole-placement · Linear quadratic regulator (LQR)

List of symbols

R_m	Motor resistance
i	Armature current
K_S	Flexible constant of joint
K_g	Gear ratio of redactor
K_m	Motor constant
R_a	Motor resistance
G	Gravitational acceleration
K_p	Proportional gain
K_i	Integral gain
K_d	Derivative gain
K	Kalman gain
$u(t)$	Control input
α	Oscillation angle

Greek symbols

τ	Motor generated torque
θ	Rotational angle
α	Oscillation angle

Abbreviations

FJM	Flexible joint manipulator
SLFJRM	Flexible joint robotic manipulator
PID	Proportional–integral–derivative controller
GA	Genetic algorithm
LQR	Linear quadratic regulator
ARE	Algebraic Riccati equation

✉ Subodh Kumar, subodh01dec@gmail.com; Kuldeep Jayaswal, Kuldeep12555@gmail.com; D. P. Kothari, dkothari@ces.iitd.ernet.in
| ¹Department of Electrical Engineering, National Institute of Technology, Kurukshetra, India. ²Department of Electrical Engineering, Rajasthan Technical University, Kota, India. ³Centre for Energy Studies, Indian Institute of Technology, New Delhi, Delhi, India.



1 Introduction

The robot technology is advancing day by day in the era of robotization. Researchers are focusing on fabrication of modern robotic manipulators to satisfy the special needs of industrial applications by decreasing the weight of the joint manipulators and utilize flexible materials as it is an important process to satisfy the special needs of industries [1]. Victor [2] proposed the manipulator in which they reported about a machine which works similar to upper limbs of humans, and moves the objects spatially. There are certain limitations in term of mobility, speed, consumption of energy and heavyweight.

Development of flexible manipulator can overcome above-said issues. These supple manipulators have many advantages over conventional rigid link manipulators in term of energy consumption, weight, and high-speed response [3]. In the 1970s [4–6], flexible joint manipulators received high effort research in terms of modeling, analysis and, control of the flexible mechanism. The possibility of modeling and controlling the vibration phenomena of flexible joint manipulator allowed engineers and researchers to design and manufacture lighter robot manipulators which would satisfy the demand for advantages [7]. A flexible joint robot arm has distributed parameters that are characterized by an infinite order system and utilizes flexible materials. Due to the nimble virtues of flexible manipulators, it's very complicated to do mathematical modeling and subsequent model-based control of the system [8].

A variety of controlling techniques are employed for the creation of a controller which can control the position and vibration of flexible joint manipulator accurately within less time. There are several control schemes such as PID, LQR, self-tuning, optimal control, state feedback control and model reference adaptive control used to control the manipulator's motion [9].

This research is aimed to design and tune a PID controller which has been compared with pole-placement and LQR controller. For full-filling the target of single-link flexible joint manipulator's position and controlling the vibrations, the proposed controllers have been tuned in order to identify the best performance. This paper has been accomplished by Matlab®/Simulink environment.

2 Dynamic modeling of flexible joint manipulator

In modern robotic systems, flexibility is playing a very crucial role as it full-filling the need of automation industries [10]. Flexible mechanism and flexible joint



Fig. 1 Structure of rotary manipulator with flexible joint

manipulators are being used in various applications. Flexibility is an unenviable feature as it is responsible for causing control issues likewise vibration and static deflections [11]. External effects, the non-linear dynamic behaviour of flexible manipulators are the various causes of different control issues [12]. Because of such issues like nonlinear vibrations reduces the end point accuracy, affects the settling time and make the controller design scheme so much convoluted [10]. Flexible joint based robotic manipulators provides many advantages likewise; less energy demand and consumption, decreased the capacity of payload, inexpensive construction, faster movements, longer reach and safer operation [11].

As per the survey on flexible structures, various control algorithms have been developed which have various restrictions for accuracy and precision. These features of controllers can further be enhanced with the help of various control structures [12].

In this article, a linear spring has been utilized to constitute the robot's joint flexibility as shown in Fig. 1. Mainly it contains a servomotor and a flexible attachment to the homogeneous springs. This servomotor handles the springs which are situated on an aluminium platform. Due to joint flexibility, arm base faces the oscillations due to the rotation that is shown by the springs. For measurement of angular position of shaft $v(t)$, optical encoder has been connected to the shaft of dc motor. Dynamic characteristics are maintained by manipulators control that is directly dependent on the accuracy of dynamic model and efficiency of control algorithms [13]. Figure 2, shows the rotation angle (θ) and the oscillation angle (α) of the end effectors. The total energy for the potential and the kinetic energy is

$$L = KE - PE \quad (1)$$

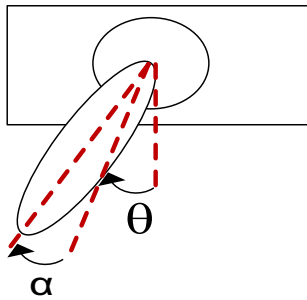


Fig. 2 Structure of rotation angle (θ) and oscillation angle (α) of the end effectors for rotary manipulator with flexible joint

where KE is kinetic energy and PE is potential energies. Therefore, the Lagrange equations [3] of the motion can be described as

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\alpha}} \right) - \frac{\partial L}{\partial \alpha} = 0 \tag{2}$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \tau \tag{3}$$

After achieving the Lagrange for the manipulator,

$$nJ_1 \ddot{\theta} + J_2 \ddot{\alpha} + K_s \alpha - mgh \sin(\theta + \alpha) = 0 \tag{4}$$

$$(J_1 + J_h) \ddot{\theta} + J_l \ddot{\alpha} - mgh \sin(\theta + \alpha) = 0 \tag{5}$$

Here, τ defines the motor generated torque. The torque has been obtained by the applied voltage v to the armature and illustrated the control input to the system. Here the relation between applied voltage and torque has been shown as

$$v = iR_m + K_m K_g \omega \tag{6}$$

$$i = \frac{v}{R_m} - \frac{K_m K_g \omega}{R_m} \tag{7}$$

where ω is motor's angular velocity and R_m is the motor resistance and i defines the armature current. Also K_m and K_g are the constant parameters respectively. Therefore

$$i = \frac{\tau}{K_m K_g} \dot{\theta} = \omega \tag{8}$$

And the desired relationship the equation as

$$\tau = \frac{K_m K_g}{R_m} v - \frac{K_m^2 K_g^2}{R_m} \dot{\theta} \tag{9}$$

By determining the state variable as

$$x_1 = \theta \tag{10}$$

$$x_2 = \alpha \tag{11}$$

$$x_3 = \dot{\theta} \tag{12}$$

$$x_4 = \dot{\alpha} \tag{13}$$

The system will be transformed in the form

$$\dot{x}_1 = x_3 \tag{14}$$

$$\dot{x}_2 = x_4 \tag{15}$$

$$\dot{x}_3 = \frac{K_s}{J_h} x_2 - \frac{K_m^2 K_g^2}{R_m J_h} x_3 + \frac{K_m K_g}{R_m} v \tag{16}$$

$$\dot{x}_4 = -\frac{K_s}{J_h} x_2 - \frac{K_m^2 K_g^2}{R_m J_h} x_3 - \frac{K_m K_g}{R_m} v - \frac{K_s}{J_1} x_2 - \frac{mgh}{J_1} \tag{17}$$

By considering v as the system input and $y = x_1$ as the output, the final form will be

$$\begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \\ \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{K_s}{J_{eq}} & -\frac{K_m^2 K_g^2}{R_m J_{eq}} & 0 \\ 0 & -\frac{K_s (J_{eq} + J_{link})}{J_{eq} J_{link}} & \frac{K_m^2 K_g^2}{R_m J_{eq}} & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \alpha \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \tag{18}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ \frac{K_m K_g}{R_m J_{eq}} \\ -\frac{K_m K_g}{R_m J_{eq}} \end{bmatrix} v_i$$

$$Y(t) = [1000] \begin{bmatrix} \theta \\ \alpha \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \tag{19}$$

The value of parameters of flexible joint considered for this work which is given in Table 1. These values of parameters have been used to calculate the system matrices for the flexible link. Below given table shows the numerical value for system parameters. Substituting the system

Table 1 Parameters of robot manipulator having flexible joint

No	Parameters	Symbols	Value
1	Inertia of flexible link manipulator	J_{link}	0.003882 kg m ²
2	Mass of the flexible joint	M	0.03235 kg
3	Inertia of Rotational platform	J_h	0.00035 kg m ²
4	Height	H	0.06 m
5	Flexibility constant of Joint	K_s	5.468 N/m
6	Gear ratio of reductor	K_g	1.36
7	Motor constant	K_m	0.0089 N/(rad/s)
8	Armature resistance	R_a	15.5 Ω
9	Gravitational acceleration	G	9.81 m/s ²

parameters of A and B matrices in Eq. (19) with numerical values from Table 1.

$$\begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \\ \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 767.05 & -52.795 & 0 \\ 0 & -1040.1 & 52.795 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \alpha \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \quad (20)$$

$$\begin{bmatrix} 0 \\ 0 \\ 98.333 \\ -98.333 \end{bmatrix} v_i$$

$$Y(t) = [1000] \begin{bmatrix} \theta \\ \alpha \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \quad (21)$$

3 Control techniques for proposed system

The controller is required to maintain the measured output to the set point value so as to minimize the error between the reference input and output [14]. In this

work, an effort has been made to control the angular position flexible joint manipulators using different control techniques like PID, state—feedback, pole placement, linear quadratic regulator (LQR), Genetic algorithm, etc.

3.1 PID control

PID is the simplest controller used in various industrial applications. They are simple in design, robust and easy to implement that’s why they are widely used in industries [15]. PID controllers calculate the gap between measured output and reference setting and further try to minimize the gap or error. PID controllers enhance the transient behavior of the system by minimizing the peak overshoot and settling time that have the proficiency to remove the steady-state error with the help of integral action [10]. The output of the controller adjusted by the proportional action as per the error size and further steady state offset can be removed by integral action mode. Here Derivative action (mode) plays a significant role [16]. The figure of the PID controller has been shown in Fig. 3. In which the set point $u(t)$ is the input signal and the controller output is the $f(s)$.

$$F(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{d}{dt} e(t) \quad (22)$$

where K_p , K_i and K_d are proportional, integral and derivative gain respectively. The effects of the above terms on closed-loop performance have been summarized in Table 2.

Fig. 3 Parallel PID Controller

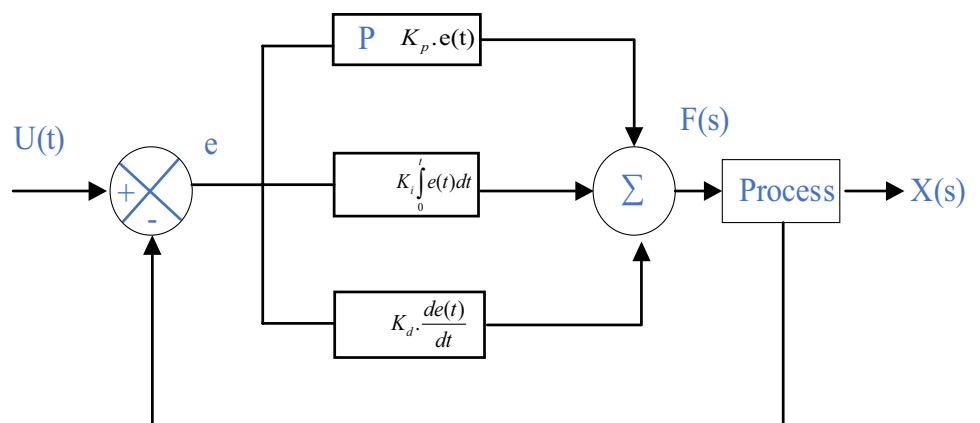


Table 2 Effect of increasing the PID parameters ('+': Increase, '-': decrease)

Parameters	Rise time	Steady-state error	Settling time	Overshoot
K_p	-	-	Small change	+
K_i	-	-	+	-
K_d	Small change	Small change	-	-

3.2 PID tuning method

For getting the optimum performance of the system of determination of corresponding PID parameter values, there is a process called Tuning. It is an important part of a closed loop control system. There are varieties of methods available for tuning the parameter of the PID controller. They are classified into two main parts which are open and closed loop methods respectively.

3.3 State feedback methodology

State-space feedback is paramount section of the modern control system, with the help of proper state feedback; the system can be designed with reduced damping and great stabilizing property. It is the method for the flexible link which allows placement of poles according to the specifications, provided the system to be stable.

Let us consider a system

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{23}$$

$$y(t) = Cx(t) + Du(t) \tag{24}$$

A is $n \times n$ system matrix, B is $n \times 1$ input matrix, C is $1 \times n$ output matrix, D is constant, The control signal is given by

$$u(t) = x(t) [-K] \tag{25}$$

Here, K is the gain matrix. Below given Fig. 4, shows the diagram of the state feedback controller. The main aim is to calculate the matrix of gains. Substituting Eq. (25) in

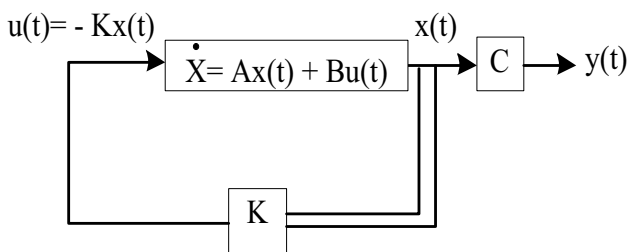


Fig. 4 Block diagram of state feedback control

Eq. (23), state variable representation of the closed-loop system becomes

$$\dot{x}(t) = [A - BK]x(t) = A_f x(t) \tag{26}$$

In this work, a state feedback controller has been designed using the pole placement method [17].

3.3.1 Pole placement method

Pole Placement method is used in feedback control theory for the placement of the closed loop poles of a plant at the desired location in s-plane. To implement this method the system must be controllable. The basic aim of the pole placement control is to place not only dominant poles but all poles at specific desired locations. Consider the state space model given in Eqs. (23) and (24) and the control signal is given in Eq. (25) then according to the pole placement design approach, the gain matrix K is in such a way that

$$|sI - (A - BK)| = (s - a_1)(s - a_2) \dots (s - a_n) \tag{27}$$

where $a_1 a_2 \dots a_n$ are the desired pole locations. Design method for calculating the variable K matrix given as follows-

- Obtain the state space model of the system
- Check the controllability of the system
- Determine the desired characteristic equation
- Solve for matrix K by comparing like powers of both sides in Eq. (27)

3.4 Linear quadratic regulator (LQR)

This Regulator is a paramount part of the controller in control system engineering. For this regulator, the system can be represented in the pattern of state variables. Here, all available states can be measured. Below given Fig. 5 shows the diagram of the quadratic regulator.

The control law for this regulator is given as-

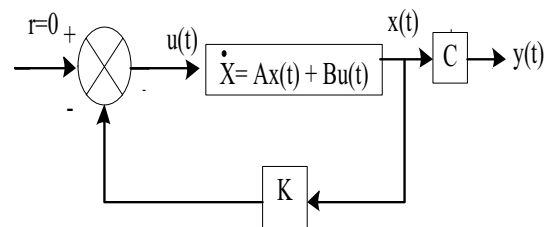


Fig. 5 Linear quadratic structure

$$u = -Kx \tag{28}$$

where K is a Kalman gain. Here, the PI is given as

$$J = \frac{1}{2} \int_0^{\infty} (x^T Qx + u^T Ru) dt \tag{29}$$

where Q denotes semi-definite and R as definite matrix. From Eq. (28) in Eq. (29) we obtained,

$$J = \frac{1}{2} \int_0^{\infty} x^T (Q + K^T RK) x dt \tag{30}$$

The 'J' can be served by energy function. Here minimization means making control loop system's energy very small. Here state $x(t)$ and control input $u(t)$ are weighted in 'J' that's why both can't be so large. If the performance index is minimized, then it will be finite which reveals about the closed-loop system in which it has to be balanced or stable. To find the optimal feedback 'K', let us consider a constant square matrix 'P' as-

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

Substituting in Eq. (29) yield: (31)

$$J = -\frac{1}{2} \int_0^{\infty} \frac{d}{dt}(x^T Px) dt = \frac{1}{2} x^T(0)Px(0) \tag{31}$$

The 'J' can be reduced when the gain performance index is selected as Eq. (32)

$$K = R^{-1}B^T P \tag{32}$$

The matrix P ($n \times n$) is calculated from Eq. (33), termed as "ARE-algebraic Riccati equation".

$$Q + PA + A^T P - PBR^{-1}B^T P = 0 \tag{33}$$

The 'K' is quadratic in nature, that's why it is called a linear Quadratic regulator problem in which 'K' is determined for minimizing the 'J'. For searching out the state feedback variables 'K', 'K' initiates with finding the state space model of the system than with the help of trial and error method, design parameter for matrix Q and R selected. Further for finding out the matrix P, Eq. (33) has been solved and finally the feedback control matrix K in state variable by Eq. (32).

3.5 Genetic algorithm

Genetic algorithm (GA) is a methodology that is used to solve optimization problems which is similar to the evaluation process in the biological world where the fittest

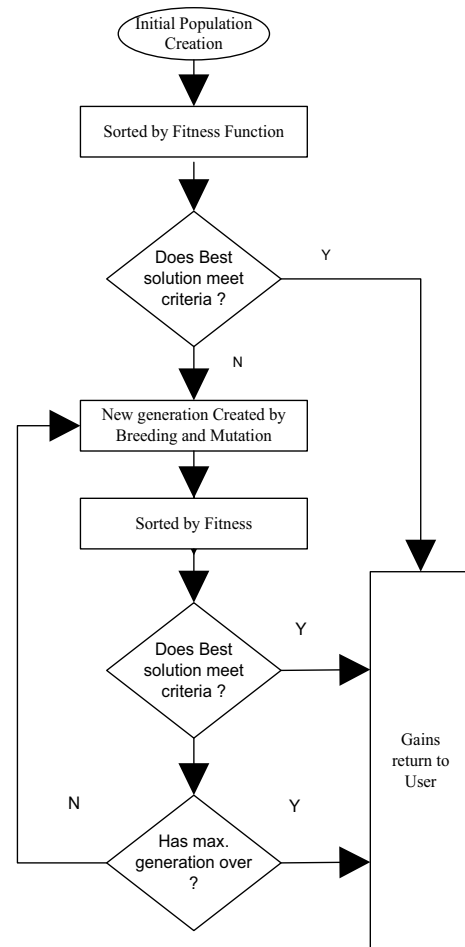
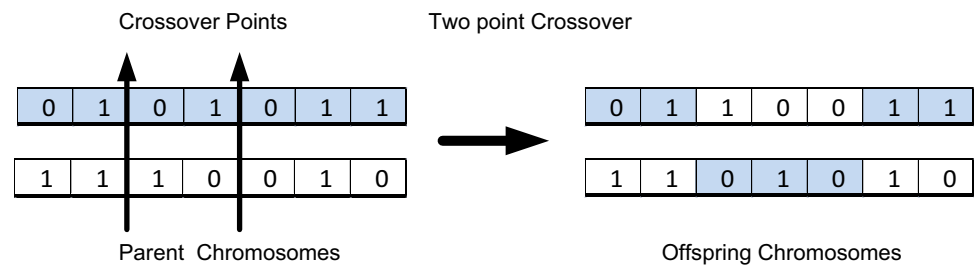


Fig. 6 Genetic algorithm flow chart

survive to reproduce [18]. Genetic Algorithm is a subset of a comparatively larger branch of computation termed as Evolutionary Computation. American Professor John Henry Holland [18, 19] proposed at first the basic ideas of GA. The method was inspired by the mechanism of selection from natural things. This method is used for optimization that initiates with no knowledge of the true solution and depends entirely upon the outputs provided by its operators of evolution like crossover, reproduction, and mutation to attain the result which has the best possibility. Flow chart for the genetic algorithm has been shown in Fig. 6.

Genetic representation is a way for representing the solutions in evolutionary computation methods. GAs use linear binary representations as an array of bits. Common genetic representation is binary array, natural language, binary tree, etc. The objective functions are the types of fitness function used to check whether a design solution meets the aims or not. As the fitness function and genetics representation defined, genetic algorithm initialize the population of the solutions and then crossover,

Fig. 7 Crossover operation



mutation, inversion are applied repeatedly to obtain the best solution.

3.5.1 Initialization

The starting population is generated in an unsymmetrical way, which permits the complete range of all feasible solutions.

3.5.2 Selection

From the existing population, a portion is selected to produce the next generation. An individual solution is selected through a fitness test as measured according to the fitness function. Fitness function mainly measures the quality of the solutions and dependent on the problem.

3.5.3 Genetic operation

The second generation is developed through the combination of a genetic operator's line crossover and mutation. Crossover is a process of producing child solution from more than one parent solutions [20, 21]. For the generation of new chromosomes, these are taken out which have good features from the previous generation. There are single and two-point crossover techniques. Single point crossover is the simplest operator in which an asymmetrically selected bit positions the genetic material of two-parent chromosome string crossovers. Similarly, in the two-point crossover, two chromosomes strings are interchanged to generate children. This operation is shown in Fig. 7.

3.5.4 Termination

The process is repeated until the termination condition is reached. Figure 8, shows some common terminating conditions [22]

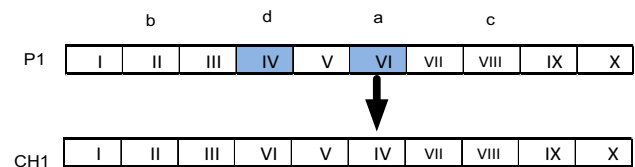


Fig. 8 Mutation operation

- The solution obtained that satisfies performance criteria
- Fixed number of generation is reached
- Manual inspection etc.

4 MATLAB®/simulation results and discussions

4.1 Open loop response

Open loop response of flexible joint manipulator using MATLAB/SIMULINK with step input reference. Figure 9, shows that the open loop response of flexible joint manipulator is unstable.

4.2 Genetic algorithm tuned PID controller

A proportional–integral–derivative controller is a classical control technique that is frequently used now a day's especially in the industry because of simple, reliable of demonstration, and broad applicability. The error signal is calculated by the PID controller and tries to minimize it. PID controller has been designed using the GA method. The aim of the design is to tune the PID gains-such that it minimizes the error between set point and output. Genetic algorithms are global, optimization and parallel search methods based on the concept of survival of fittest. The proposed PID control has been simulated with the gains obtained by GA method depicted in Table 3. For the tuning procedure first chromosome representations has been defined by three values that related to the gains which

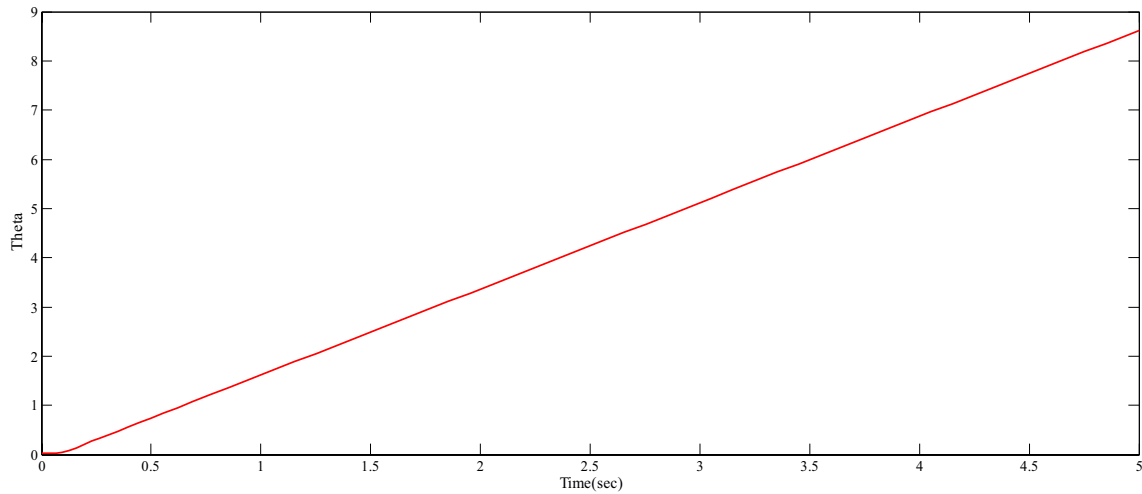


Fig. 9 Output of the open loop system

Table 3 PID controller gain using genetic algorithm

Parameters	K_p	K_i	K_d
PID gain	3.5668	0.0896	0.0922

have to be adjusted in order to obtain proper behavior. The three gains are K_p , K_i and K_d .

As the main aim is to reduce the error between outputs and input that is set point, fitness function has been given as

$$\text{Fitness} = \sum^n [\text{desired output} - \text{actual input}]^2$$

Here the intervals are divided into different 'n' points. The minimization of fitness function performed by the genetic algorithm using programming in MATLAB with the help of GA optimization block which ensured the output near to desired value. The internal process of the genetic algorithm has been explained in Sect. 3.5. Figure 10 represents the step response of the PID controller design and Fig. 11, shows tracking error, the system is stable and the position reaches zero steady-state error in a second. The rise time is satisfied, but the settling time and overshoot need to be reduced.

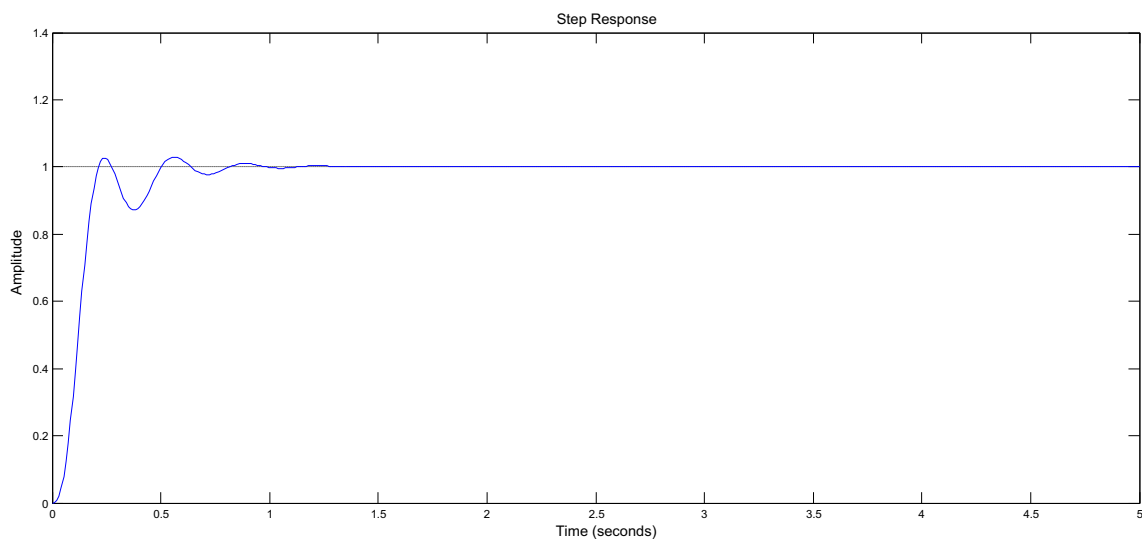


Fig. 10 Step response of PID controller

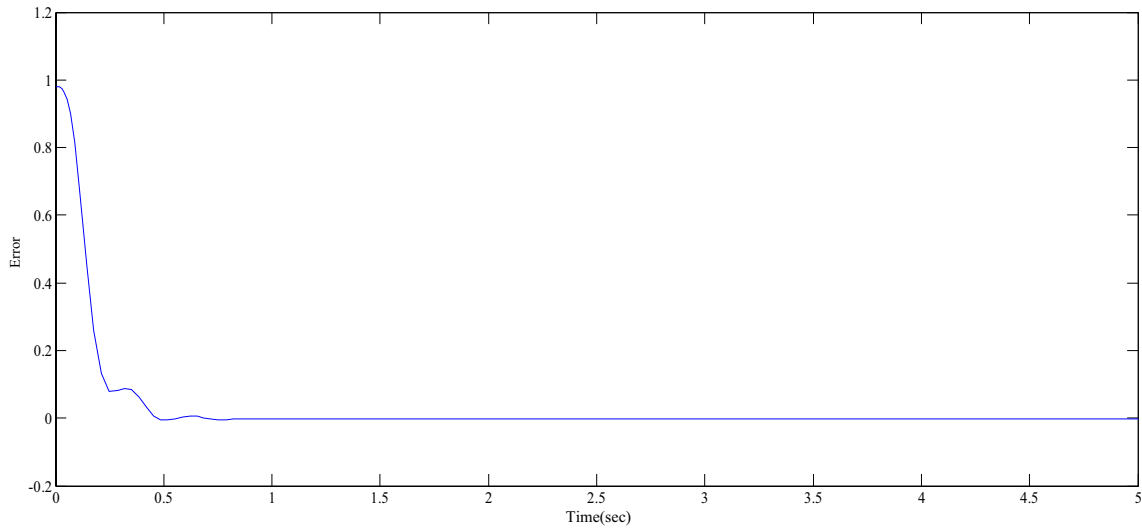


Fig. 11 Tracking error using PID

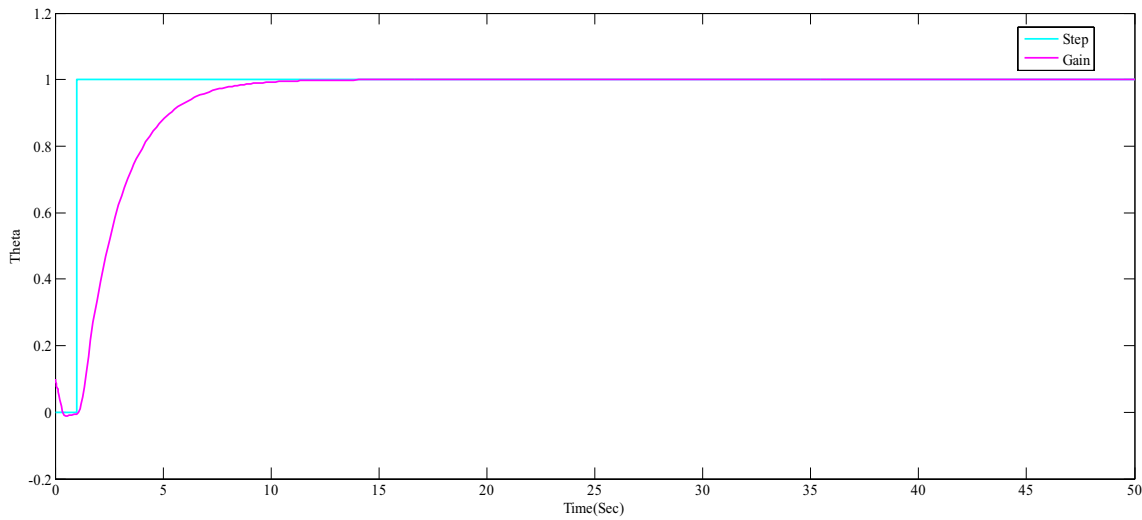


Fig. 12 Step response of theta under reference input

4.3 Pole placement method

Here, in this case, new poles have been added at required places such that many of the poles should be in the left half of S-plane to more stabilize the system. Here the desired location to place the poles of flexible link manipulator have been given as:

$$P = [-15, -13, -10, -14, -16]$$

K= by placing (A B, p) using running this command window the matrix 'K' values are as:

$$K = [6.1418, -1.9829, 0.3239, 0.1693, 16.2688]$$

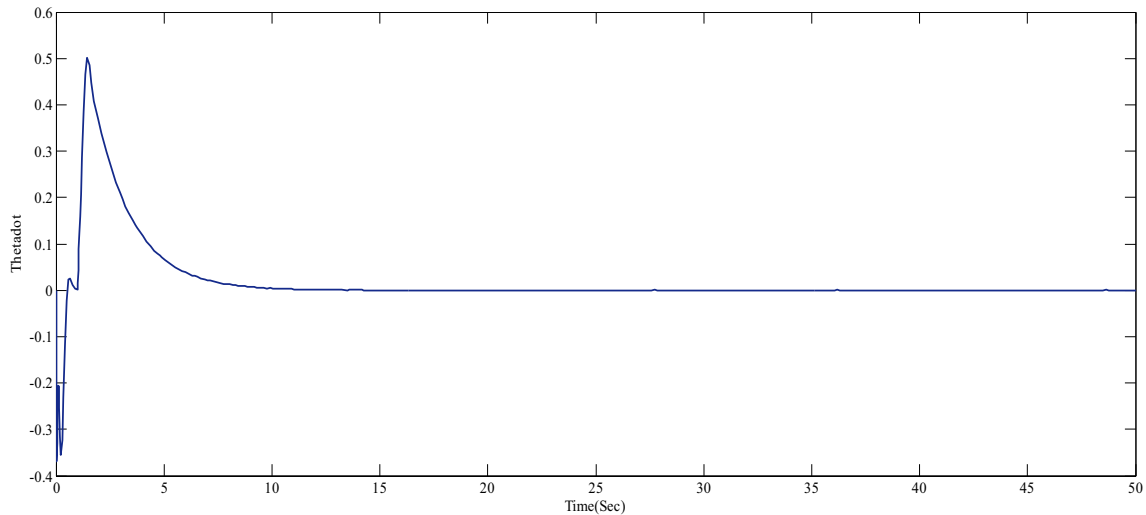


Fig. 13 Step response of theta dot using pole placement

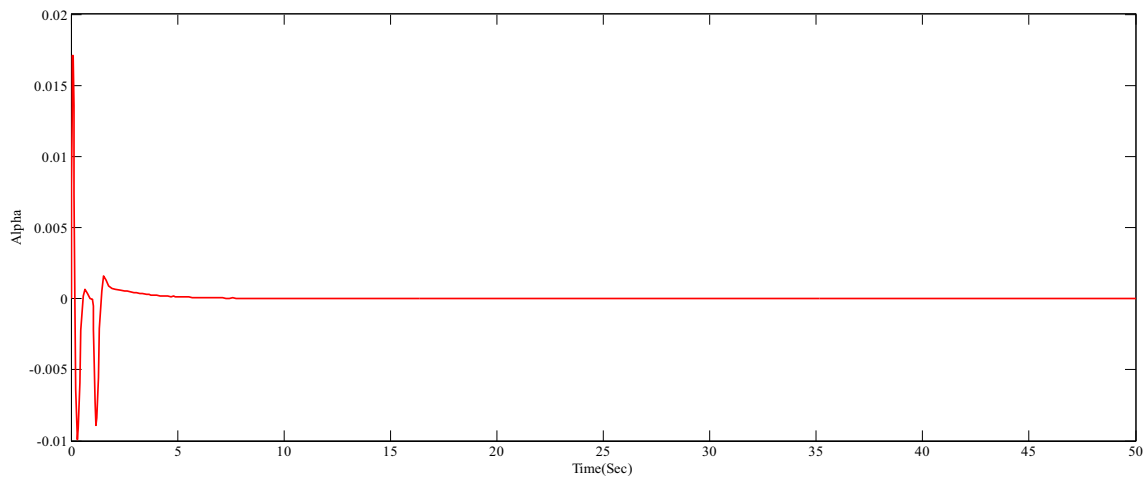


Fig. 14 Step response of alpha using pole placement

Figure 12 shows the simulation result of theta under step input and Fig. 13 shows the step response of theta dot. It shows a faster response with no oscillation than PID. Figure 14 shows simulation results for deflection angle and Figs. 15, 16, shows simulation result for tracking error.

4.4 LQR method

By tuning the weighting matrix using GA method

$$Q = \text{diag}(0.584, 0.3891, 0, 0.0397)$$

Obtained controller feedback gain

$$K = [0.9241 \quad -0.6898 \quad 0.0669 \quad 0.0031]$$

Figures 17, 18, 19 and 20, shows simulation result of LQR for flexible joint manipulator using controller feedback gain K. It shows a faster response than the PID controller and pole placement. Figure 21 shows the analysis of tracking error. Further Fig. 22, shows a comparison graph of theta and Fig. 23, shows 2norm error for Fig. 9. Output of the open loop system the flexible joint manipulator. From the results it has been evident that

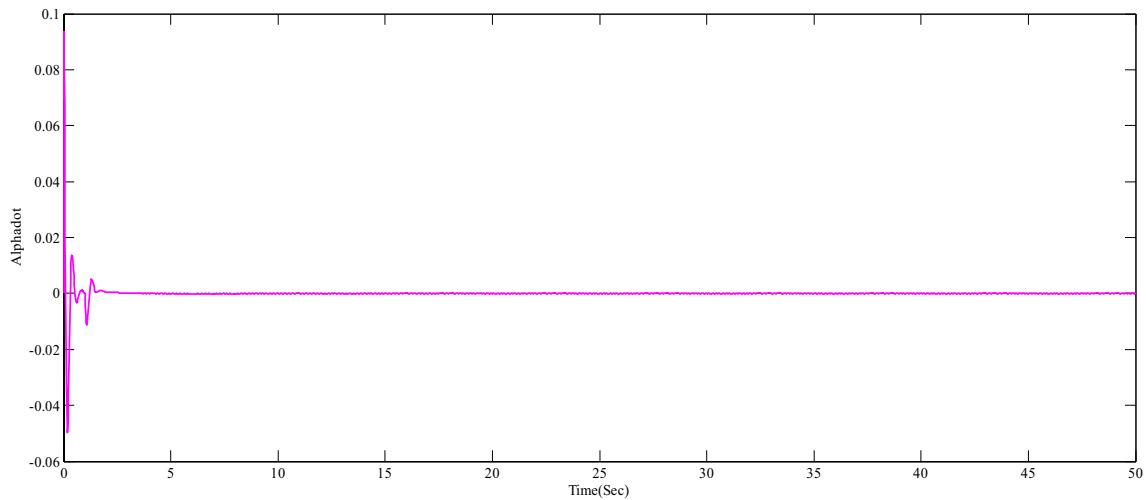


Fig. 15 Step response of alphasdot using pole placement

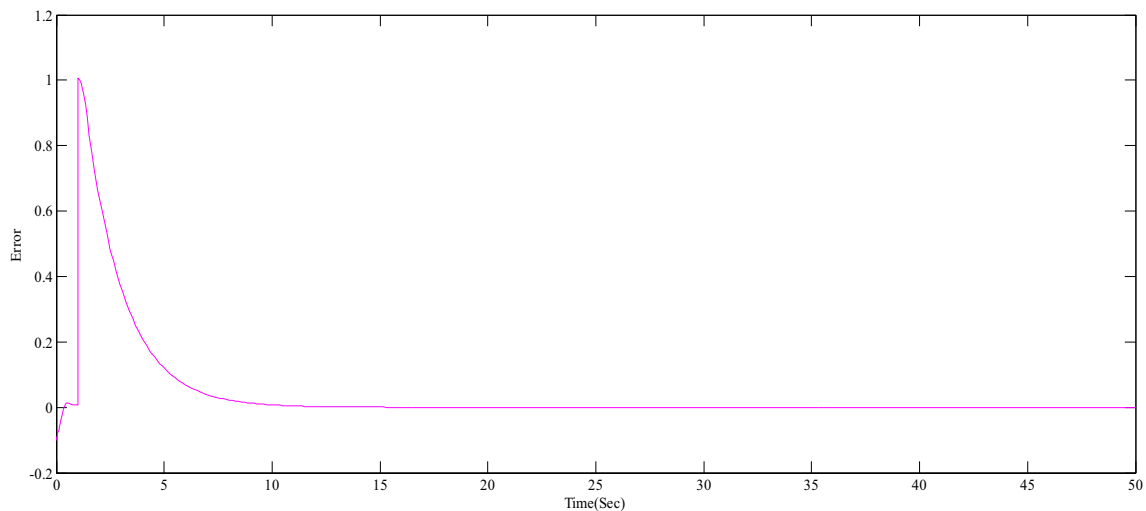


Fig. 16 Tracking error using pole-placement

the pole-placement method performs better in comparison of PID and, LQR perform better than both PID and pole-placement method.

5 Conclusions

In this article, flexible Joint manipulator's dynamic model has been modeled with the help of Euler–Lagrange's theory into a state space form. The open loop response

depicts that the system is not stable, to stabilize the system three control schemes have been proposed. Simulation has been performed for pole placement method by which the system is stabilized and the position reaches to zero steady-state error but the settling time and the overshoot have been found to be optimized. To achieve it, the linear quadratic regulator (LQR) technique has been used. From which, it has been evident that the step responses are better than Pole placement method. But obtained responses are not satisfactory so

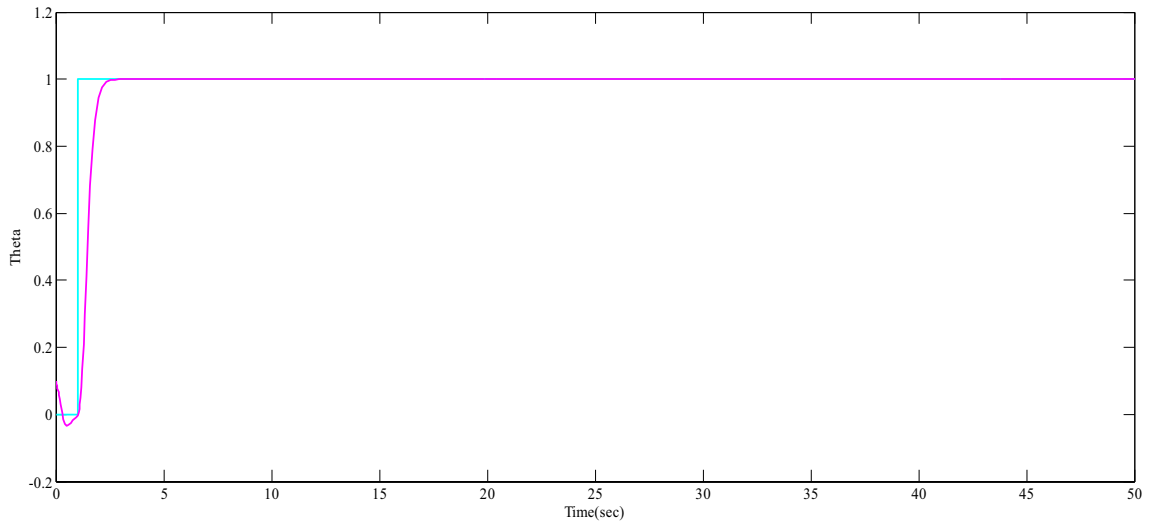


Fig. 17 Step response of theta under reference input

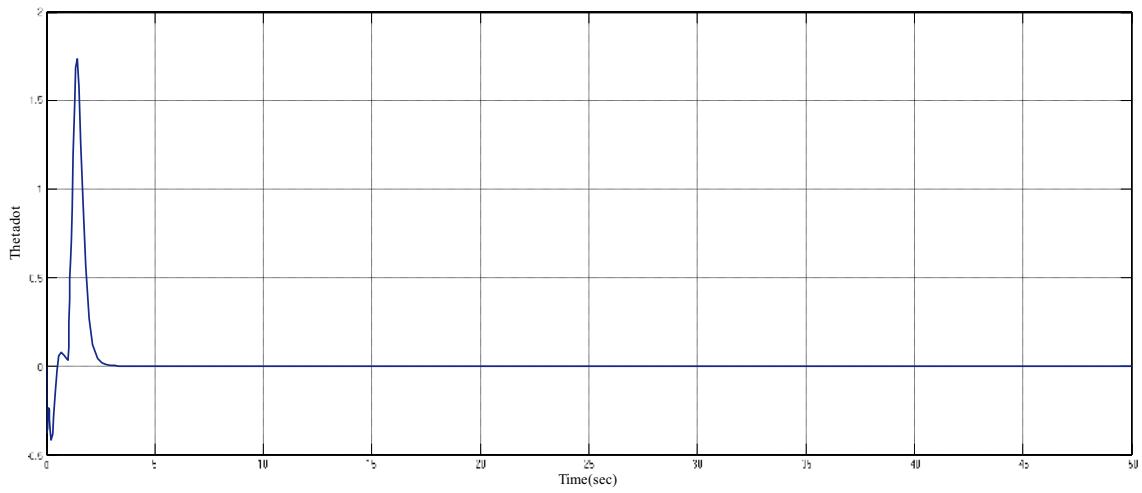


Fig. 18 Step response of theta dot

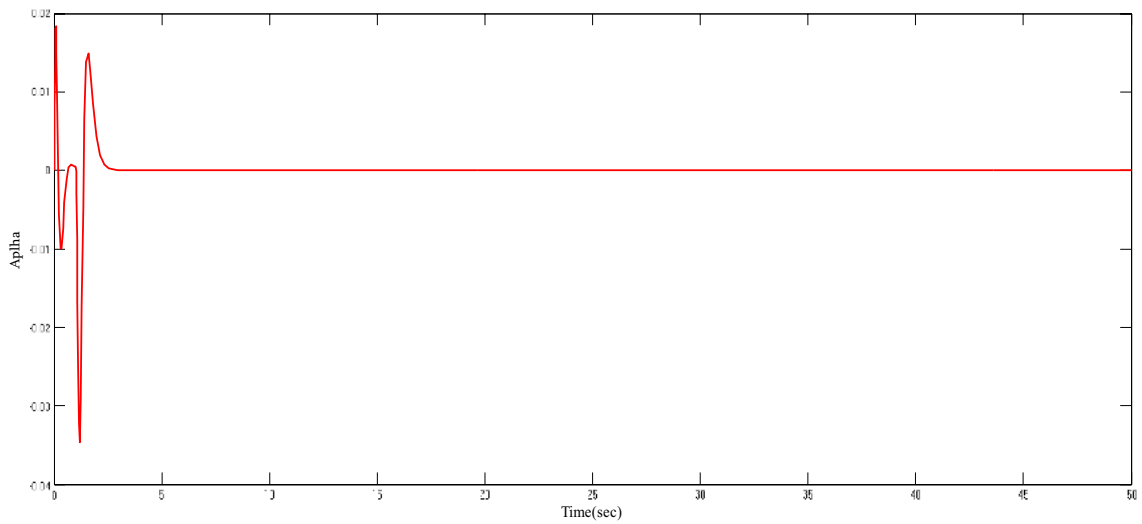


Fig. 19 Step response of alpha

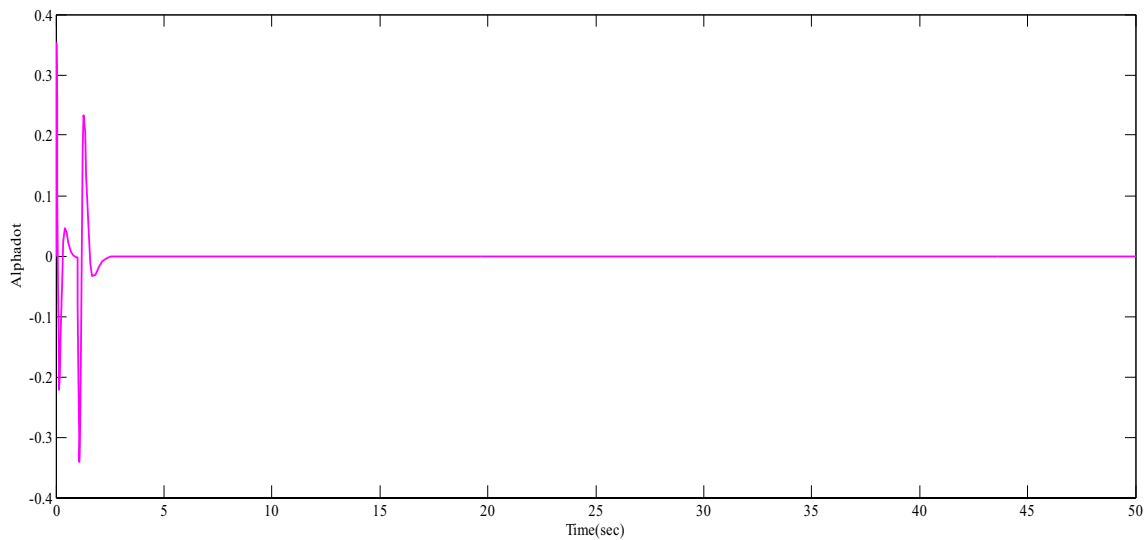


Fig. 20 Step response of alphasdot

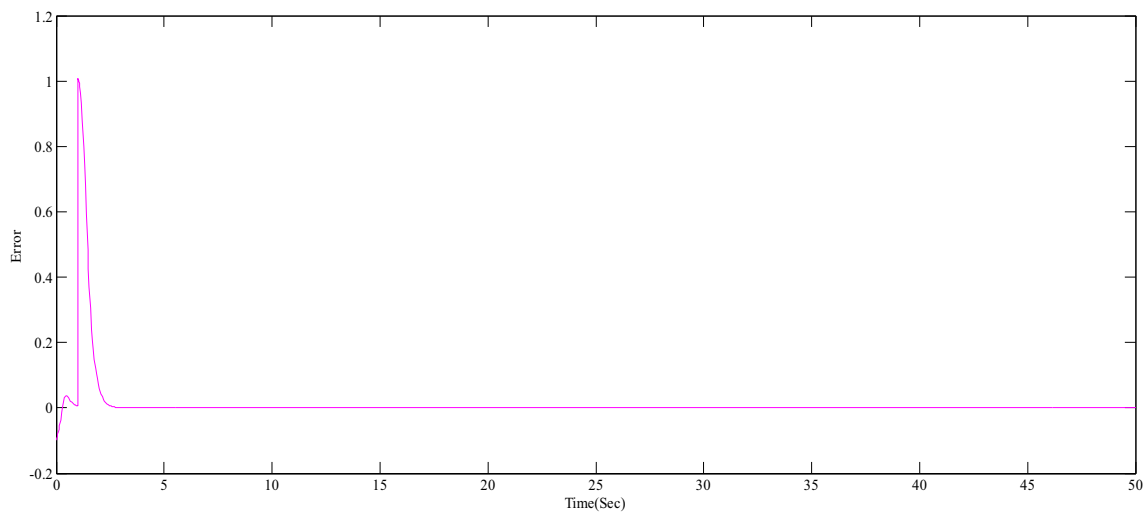


Fig. 21 Tracking error using LQR

for better optimization, Genetic Algorithm (GA) tuned PID controller has been used. From which it is clear that Step responses of the manipulator using GA tuned PID

controller gains yield faster response in comparison the Pole Placement and LQR Techniques which ultimately provides better control to a flexible joint manipulator.

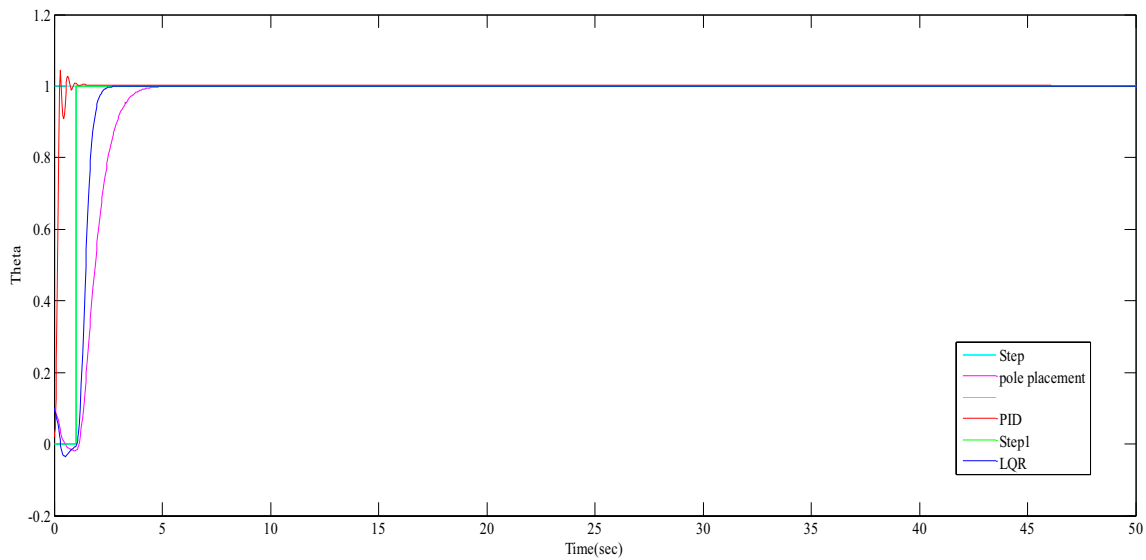


Fig. 22 Comparison of theta

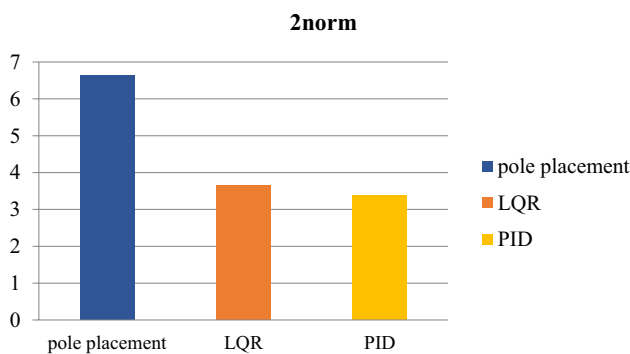


Fig. 23 Comparison graph for 2norm error

Compliance with ethical standards

Conflict of interest The authors declare no conflict of interest.

References

1. He W, Yan Z, Sun Y, Yongsheng O, Sun C (2018) Neural-learning-based control for a constrained robotic manipulator with flexible joints. *IEEE Trans Neural Netw Learn Syst* 29:5993–6003. <https://doi.org/10.1109/tnnls.2018.2803167>
2. Etxebarria V, Sanz A, Lizarraga I (2005) Control of a lightweight flexible robotic arm using sliding modes. *Int J Adv Robot Syst* 2:103–110. <https://doi.org/10.5772/5798>
3. Ahmad S (1993) Constrained motion (force/position) control of flexible joint robots. *IEEE Trans Syst Man Cybern* 23:374–381. <https://doi.org/10.1109/21.229451>
4. Gevarter WB (1970) Basic relations for control of flexible vehicles. *AIAA J* 8:666–672
5. Kirk D (1970) *Optimal control theory*. Prentice-Hall Inc., Upper Saddle River
6. Book WJ, Maizza-Neto O, Whitney DE (1975) Feedback control of two beam, two joint systems with distributed flexibility. *J Dyn Syst Meas Control* 97:424–431
7. Shawky A, Zydek D, Elhalwagy YZ, Ordys A (2013) Modeling and nonlinear control of a flexible-link manipulator. *Appl Math Model* 37:9591–9602. <https://doi.org/10.1016/j.apm.2013.05.003>
8. Siddique MNH, Tokhi MO (2006) GA-based neural fuzzy control of flexible-link manipulators. *Eng Lett* 13:148–157
9. Sk D, Eberhard P (2006) Dynamic analysis of flexible manipulators, a literature review. *Mech Mach Theory* 41:749–777. <https://doi.org/10.1016/j.mechmachtheory.2006.01.014>
10. Akyuz IH, Yolacan E, Ertunc HM, Bingul Z (2011) PID and state feedback control of a single-link flexible joint robot manipulator. In: 2011 IEEE international conference on mechatronics (ICM 011), pp 409–414. <https://doi.org/10.1109/icmech.2011.5971320>
11. Roshin R, Shihabudheen KV (2013) Mathematical modeling of flexible beam—a comparative study. In: 2013 International conference on control communication and computing (ICCC), pp 325–330. <https://doi.org/10.1109/iccc.2013.6731673>
12. Sun L, Yin W, Wang M, Liu J (2018) Position control for flexible joint robot based on online gravity compensation with vibration suppression. *IEEE Trans Ind Electron* 65:4840–4848. <https://doi.org/10.1109/TIE.2017.2772157>
13. Rana D Singh, Deepika M (2013) Modelling, stability analysis and control of flexible single-link robotic manipulator. *Int J Adv Res Electric Electron Instrum Eng* 3:7390–7401
14. Subudhi B, Morris AS (2002) Dynamic modeling, simulation and control of a manipulator with flexible links and joints. *Robot Auton Syst* 41:257–270. [https://doi.org/10.1016/s0921-8890\(02\)00295-6](https://doi.org/10.1016/s0921-8890(02)00295-6)
15. Alandoli E, Rashid MZA, Sulaiman M (2017) A comparison of PID and LQR controllers for position tracking and vibration suppression of flexible link manipulator. *J Theor Appl Inf Technol* 95:2949–2955
16. Pan Y, Li X, Haoyong Yu (2018) Efficient PID tracking control of robotic manipulators driven by compliant actuators. *IEEE Trans*

- Control Syst Technol 27:915–922. <https://doi.org/10.1109/TCST.2017.2783339>
17. Rahman TAZ, Darus IZM (2012) Active vibration control using pole placement method of a flexible plate structure optimized by genetic algorithm. In 2012 IEEE conference on control, systems and industrial informatics, pp 92–97. <https://doi.org/10.1109/ccsii.2012.6470480>
 18. Holland JH (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT Press, Cambridge
 19. Holland JH (1992) Adaptation in natural and artificial systems. MIT Press, Cambridge
 20. Mahony TO, Downing CJ, Fatla K (2000) Genetic algorithm for PID parameter optimization: minimizing error criteria. In: 2000 Proceedings process control and instrumentation, pp 148–153
 21. Andri M, Yoshii S, Furukawa M (2006) PID parameters optimization by using genetic algorithm. ISTECS J 8:34–43
 22. Sharma N, Anupama KR (2011) A novel genetic algorithm for adaptive resource allocation in MIMO-OFDM systems with proportional rate constraint. Wireless Pers Commun 61:113–128. <https://doi.org/10.1007/s11277-010-0013-9>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.