

Trajectory tracking optimization and control of a three link robotic manipulator for application in casting

Mahendra Kumar Jangid^{1*}, Sunil Kumar² and Jagtar Singh³

Research Scholar, Department of Mechanical Engineering, SLIET Longowal Punjab India¹

Assistant Professor, Department of Mechanical Engineering, SLIET Longowal Punjab India²

Professor, Department of Mechanical Engineering, SLIET Longowal Punjab India³

Received: 21-July-2021; Revised: 17-October-2021; Accepted: 19-October-2021

©2021 Mahendra Kumar Jangid et al. This is an open access article distributed under the Creative Commons Attribution (CC BY) License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

In this study, forward kinematics, inverse kinematics, dynamic simulation and control of a three-link robotic manipulator for the pouring of molten metal using a small crucible are described. A 3D Computer Added Design (CAD) model of the robotic manipulator is designed in SolidWorks software. Simulation and control of this robotic manipulator were conducted in MATLAB 2016 b. The robotic manipulator SolidWorks CAD file is first converted into an Extensible Markup Language (XML) file after and then imported into MATLAB/Sim-Mechanics environment. A study on forward and inverse dynamics with a sinusoidal wave and cycloid trajectory as input signal was conducted and the joint actuator is controlled by Proportional-Integral-Derivative (PID) controller having a derivative filter. Trajectory tracking is optimized by Integral of Time multiplied by Absolute Error (ITAE) criteria using a pattern search algorithm. Variation in joint torque with a vertical load on the end-effector is described. It was found that the optimized controller reduced the ITAE error by 93.05%, 94.44%, and 97.6% in joint1, joint2, and joint3 respectively.

Keywords

Forward kinematics, Inverse kinematics, Robotic manipulator, PID, Trajectory tracking optimization.

1.Introduction

When it comes to working for repetitive tasks, in hazardous environments, continuous working for 24×7 hours, which is challenging for humans, where robots or robotic manipulators came in a picture to do similar tasks. In order to enhance the efficacy of the robots and robotic manipulators, robust exploration and advanced research are necessary these days [1]. Since motion control of a robotic manipulator has been a difficult task because of the non-linear dynamic model. Robotic manipulator is controlled by different control algorithms such as Proportional Integral Derivative (PID) control [2], Computed Torque Control (CTC) [3], Sliding Mode Control (SMC) [4], fuzzy logic [5], artificial neural network (ANN) [6]. A true linear model is difficult to obtain so different optimization methods are used to tune the controller parameters.

A dynamic model of a robotic manipulator can be obtained by using analytic methods such as Lagrangian Mechanics and Newton-Euler formulation. But with an increase in the number of degrees of freedom, the dynamic equation becomes complicated and difficult to calculate on pen and paper. So, the MATLAB SIM-mechanics toolbox is used for modeling the complicated dynamic equation of the robotic manipulator in recent years reported in [7–9]. Robotic manipulators are used in many areas, but in this study, the main focus has been done on robotic manipulators which can be used in casting applications. A study on the reuse of a robotic manipulator applied in die casting [10].

The main objective of this study is to minimize the trajectory tracking error during the pouring of liquid metal in casting to make the product accurate, efficient and increase in production rate. To achieve this goal, the main objectives of this study are as below-

1. To design a three-link robotic manipulator in solid works software according to Denavit–Hartenberg (D-H) parameters.

*Author for correspondence

2. To simulate and control this designed robotic manipulator in MATLAB/Sim-mechanics.
3. To optimize the trajectory tracking of this robotic manipulator during liquid metal pouring in casting by a pattern search algorithm using Integral of Time multiplied by Absolute Error (ITAE) criteria.

To achieve the above objectives, a three-link robotic manipulator is first designed in SolidWorks software with D-H parameters, then Extensible Markup Language (XML) and Standard Tessellation Language (STL) files are generated using SIM-mechanics Computer Added Design (CAD) translator add-ons. Then, an XML file is imported into the MATLAB software for dynamics control, simulation, and motion analysis of the robotic manipulator.

This article is organized as follows; section 2 discussed the literature review. Section 3 elaborates the methodology covering the modeling of a three-link robotic manipulator, dynamic modeling of the three-link robotic manipulator discussion, detailing of the dynamic simulations, and control of robotic manipulators in MATLAB SIM-mechanics and trajectory tracking optimization using the ITAE criterion. Results are given in section 4 along with the study limitations. In section 5 research work impacts have been discussed, and in the last, the research work is concluded in section 6.

2.Literature review

Different optimization techniques have been used by the researchers to tune the various control parameters. Computing technology, mostly used nowadays for the optimization and simulation of a robotic manipulator. Here are some literatures focusing mainly on the field of robotic manipulator controller parameters and trajectory optimization.

A Modified Neural Network Algorithm (MNNA) is defined by [11] and use as a novel adaptive tuning algorithm for optimization of controller parameters of two degrees of freedom robotic arm. In this study, a comparative study was conducted between Genetic Algorithm (GA)-PID and Cuckoo Search Algorithm (CSA)-PID. A cubic polynomial trajectory was used for analysis in MATLAB. It was found that MNNA is superior to other optimization techniques.

One another study on event-driven sliding mode control trajectory tracking for a nonlinear system with disturbance is found in [12]. In this study, the nonlinear system shows a better response for a given reference trajectory in terms of reduced control

updates. The control updates for the presence of constant found 44% and time-varying disturbance 61%.

An adaptive sliding mode tracking control for a six-degree of freedom industrial robot is found in [13]. This controller is a combination of sliding mode control, adaptive and robust controls. The unknown parameters were updated online based on discontinuous projection adaptation law. It was found that the proposed controller gives better dynamic trajectory tracking compare to the conventional PID controllers. A novel sliding mode controller was proposed by [14] applying on two degrees of freedom robotic manipulator and trajectory was optimized by the Extended Grey Wolf Optimizer (EGWO) that was obtained by adding emphasis coefficients in the Grey Wolf Optimizer (GWO). The optimized parameters were compared with two other optimization techniques Proportional Derivative Sliding Mode Control (PDSMC) and SMC. It was found that the effectiveness of EGWO is more compared to other PDSMC and SMC.

One another comparative study on the optimization of PID controller parameters on two degrees of freedom robotic manipulator is found in [15]. In this study, three optimization techniques simulated annealing (SA), particle swarm optimization (PSO), and genetic algorithm (GA) was studied. It was concluded that PSO shows more effectiveness in trajectory tracking as compared to SA and GA.

Recently a comparative study of SMC and PID on two links robotic manipulators using MSC Adams and MATLAB/Response optimization toolbox found in [16]. The parameters of SMC were optimized by Gradient Descent (GD), Pattern Search (PS), and Simplex Search (SS). SMC shows 36 %, 35%, and 15% more improvement in ITAE, IAE, and ISE respectively as compare to PID. SS gives better results in trajectory tracking as compare to others.

Optimization of PID controller parameters using a combination of GA and PSO called Genetic-Swarm Optimization (GSO) is found in [17]. In this article, the author also uses GSO for the solution of the inverse kinematics of a multi-joint serial robotic manipulator. The author concludes that GSO has a lower average error at each joint. GSO shows- Joint 4 average error of 19.33% and 22.7% less compares to PSO and GA respectively.

One article found on optimization of PID controller parameters for a nonlinear system using nature-inspired metaheuristic algorithms. Social Spider Algorithm (SSA) was used for tuning of PID parameters of two degrees of freedom non-linear system and compare with other optimization techniques GA and PSO. The author concludes that SSA shows minimum error as compared to other techniques [18]. In conclusion, after studying literature regarding this topic most of the literature found on two and six degrees of freedom robotic manipulator, but there is the very lack of research found on trajectory tracking of articulated three degrees of freedom robotic manipulator. Most of the articles reported on optimization of controller parameters for two-link robotic manipulators with very lack of real applications. There is a very confined study found on trajectory tracking optimization using ITAE criteria by the pattern search algorithm.

3.Methods

In this section methods that are applied for achieving the objectives are discussed.

3.1Modelling of three link robotic manipulators

In this section, kinematic, inverse kinematic, and dynamic mathematic modeling equations of a three-link robotic manipulator are derived. Link parameters are obtained by the D-H notation as shown in *Table 1*. Studies in the modeling of the robotic manipulator are found in [19–21].

3.1.1 Kinematic modelling of the robot

Kinematic modeling gives the orientation and position of a robotic manipulator end-effector. Kinematics is classified as Forward kinematics and Inverse kinematics.

3.1.2 Forward kinematics transformation matrix

Forward kinematics are used for the calculation of the position and orientation of the end-effector of a robotic manipulator. First transformation coordination frames are assigned to the robotic manipulator in *Figure 1 (a)* CAD model or *Figure 1 (b)* spherical coordinate model. This transformation coordination frames is shown in *Figure 2*.

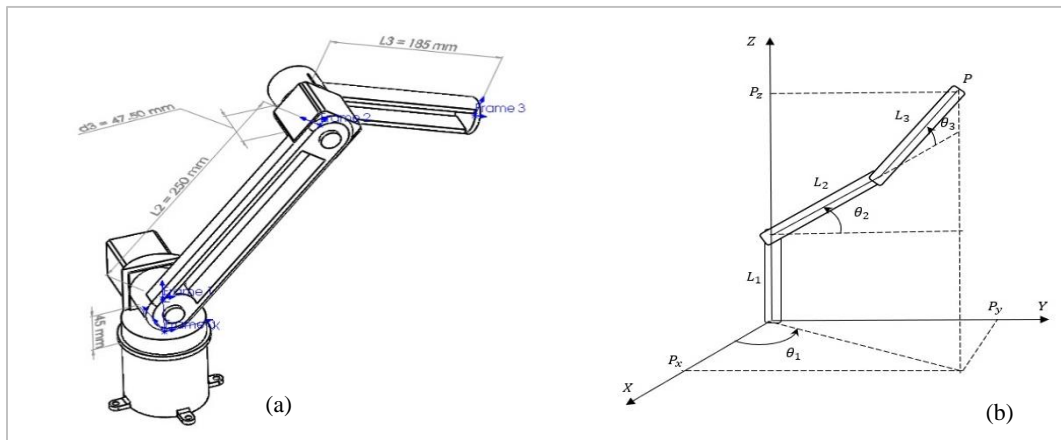


Figure 1 (a) A schematic designed 3D CAD model of a 3-DOF Robotic Manipulator (b) Spherical coordinates of this robotic manipulator

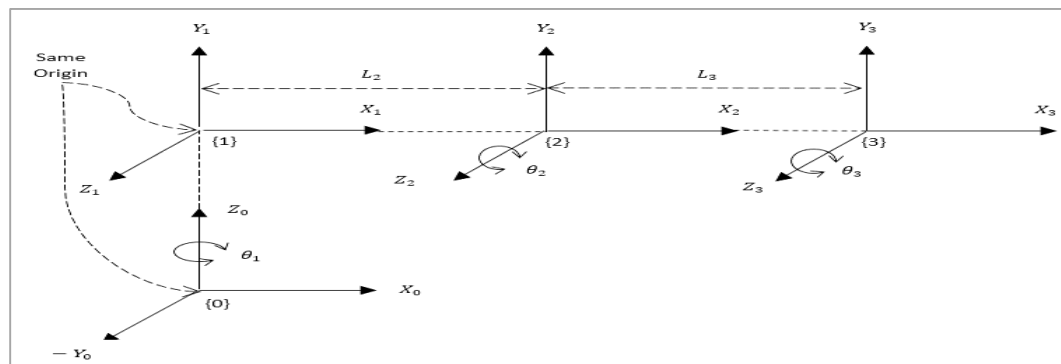


Figure 2 Frame assignment of the robotic manipulator each joint

$$T_1^0(\theta_1) = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$T_2^1(\theta_2) = \begin{bmatrix} C_2 & -S_2 & 0 & L_2 C_2 \\ S_2 & C_2 & 0 & L_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$T_3^2(\theta_3) = \begin{bmatrix} C_3 & -S_3 & 0 & L_3 C_3 \\ S_3 & C_3 & 0 & L_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$T_3^0 = T_1^0 T_2^1 T_3^2 = \begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 & C_1(L_3 C_{23} + L_2 C_2) \\ S_1 C_{23} & -S_1 S_{23} & -C_1 & S_1(L_3 C_{23} + L_2 C_2) \\ S_{23} & C_{23} & 0 & L_3 S_{23} + L_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

End- effector transformation matrix

$$T_E = \begin{bmatrix} n_x & o_x & a_x & d_x \\ n_y & o_y & a_y & d_y \\ n_z & o_z & a_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 & C_1(L_3 C_{23} + L_2 C_2) \\ S_1 C_{23} & -S_1 S_{23} & -C_1 & S_1(L_3 C_{23} + L_2 C_2) \\ S_{23} & C_{23} & 0 & L_3 S_{23} + L_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

After that D-H notation is used for the calculation of the transformation matrix. Forward kinematics according to D-H parameters (Table 1) are calculated in Equations 1 to 5. End effector position can also be calculated according to spherical coordinates as in Figure 1 (b).

Table 1 D-H parameters

Link i	a_i	α_i	d_i	θ_i	q_i	$C\alpha_i$	$S\alpha_i$
1	$L_1 = 0$	90°	0	θ_1	θ_1	0	1
2	$L_2 = 250$ mm	0	0	θ_2	θ_2	1	0
3	$L_3 = 185$ mm	0	0	θ_3	θ_3	1	0

3.1.3 Inverse kinematics

Inverse kinematics are used for findings of joint angles for giving the end effector transformation matrix as 1258

given in Equation 6 analytically joint angles are calculated according to closed-form solutions given in Equations 7 to 9 [22].

$$\begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 & C_1(L_3 C_{23} + L_2 C_2) \\ S_1 C_{23} & -S_1 S_{23} & -C_1 & S_1(L_3 C_{23} + L_2 C_2) \\ S_{23} & C_{23} & 0 & L_3 S_{23} + L_2 S_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

From closed form solutions

$$\theta_1 = \text{Atan2}(r_{24}, r_{14}) \quad (7)$$

$$\theta_2 = \text{Atan2}\left(-L_3 r_{31} + r_{34}, \pm\sqrt{(-L_3 r_{11} + r_{14})^2 + (-L_3 r_{21} + r_{24})^2}\right) \quad (8)$$

$$\theta_3 = \text{Atan2}(r_{31}, r_{32}) - \theta_2 \quad (9)$$

3.2 Dynamic modelling of the robot

A scalar function called *Lagrange function* or *Lagrangian* \mathcal{L} is evaluated as the difference between the total kinetic energy \mathcal{K} and the total potential energy \mathcal{P} of a mechanical system (Equation 10).

$$\mathcal{L} = \mathcal{K} - \mathcal{P} \quad (10)$$

The dynamic model based on Lagrange-Euler formulation is obtained from the *Lagrangian*, as a set of equations (Equation 11).

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i \quad (11)$$

The dynamic behaviour of joint i of the manipulator neglecting the joint friction effect is given in Equation 12 and 13.

$$\tau_i = \sum_{j=1}^n M_{ij}(q) \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk} \dot{q}_j \dot{q}_k + G_i \quad (12)$$

For this manipulator

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\theta} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} H_{11} \\ H_{21} \\ H_{31} \end{bmatrix} + \begin{bmatrix} G_{11} \\ G_{21} \\ G_{31} \end{bmatrix} \quad (13)$$

Where τ_i is the input torque of the i^{th} joint, $M_{ij}(q)$ is the inertia matrix, H_{ij} is the Coriolis/centripetal force matrix, G_{ij} is the gravity vector matrix.

The obtained dynamic equations for considering point mass are given in *Appendix I*.

3.3 Dynamic simulation and control of robot in MATLAB Sim-Mechanics

The robotic manipulator model is designed in SolidWorks software as shown in *Figure 3(a)*, length is selected according to D-H parameters (*Table 1*), mass and inertia are obtained (*Table 2*) according to material property that is selected in SolidWorks. This SolidWorks CAD file is exported into an XML and STL file using SIM-mechanics 1st generation add-ons. This XML file is imported in MATLAB software using command `mech_import` with the 1st generation Simulink model as shown in *Figure 3 (b)* (physical model). Mass and inertia tensor property was obtained in SIM-mechanics for each link block as shown in

Table 2. After running the simulation, a CAD model animation is obtained in SIM-mechanics as shown in *Figure 3(b)* showing the Centre of gravity (CG) of each link.

A solid works CAD model of the three-link robotic manipulator attached with stepper motor at each joint is shown in *Figure 4*. The casting process of grasping of small crucible filled with liquid metal is in *Figure 4 (a)* to the pouring of liquid metal in casting mould by turning the crucible is in *Figure 4 (b)*. The working procedure for achieving of all objectives in section 1 are described as a flow chart in *Figure 5*.

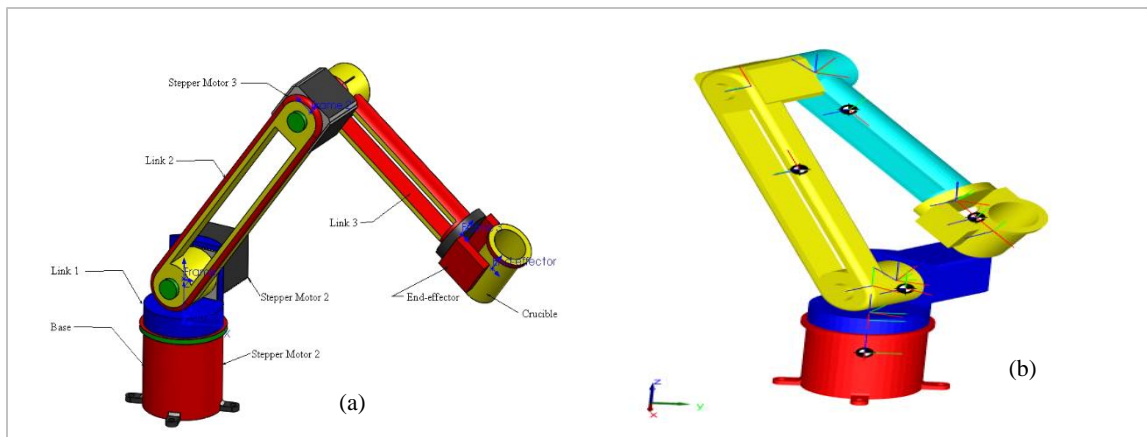


Figure 3 (a) 3D CAD model of the robotic manipulator with end-effector gripper grasping small crucible (b) CAD model view in MATLAB/Sim-mechanics showing CG of each link

Table 2 Mass and Inertia of robot different parts

Link	Mass (Kg)	Mass moment of inertia tensor(Kg-m ²)	Name
0	3.41047	$\begin{bmatrix} 0.00353824 & 0 & 0 \\ 0 & 0 & 0.00353824 \\ 0 & 0 & 0.00280315 \end{bmatrix}$	Base
1	2.03413	$\begin{bmatrix} 0.00297078 & -1.39932e-100 & 2.64538e-100 \\ -1.39932e-100 & 0.002648820 & 0.000877107 \\ 0.2.64538e-10 & 00.000877107 & 0.00147878 \end{bmatrix}$	Link 1
2	2.76911	$\begin{bmatrix} 0.00148764 & 00.000550148 & 00.00110764 \\ 0.0005501480 & 0.03401440 & -1.83105e-05 \\ 00.00110764 & -1.83105e-05 & 0.0339577 \end{bmatrix}$	Link 2
3	1.09587	$\begin{bmatrix} 0.000313499 & -2.50076e-090 & -0.0002151340 \\ -2.50076e-09 & 0.004499040 & -7.33306e-09 \\ -0.000215134 & -7.33306e-09 & 0.00459017 \end{bmatrix}$	Link 3
4	0.724496	$\begin{bmatrix} 0.000477451 & 05.37058e-06 & 0 \\ 5.37058e-060 & 0.000559974 & 0 \\ 0 & 0 & 0.00054326 \end{bmatrix}$	End-effector

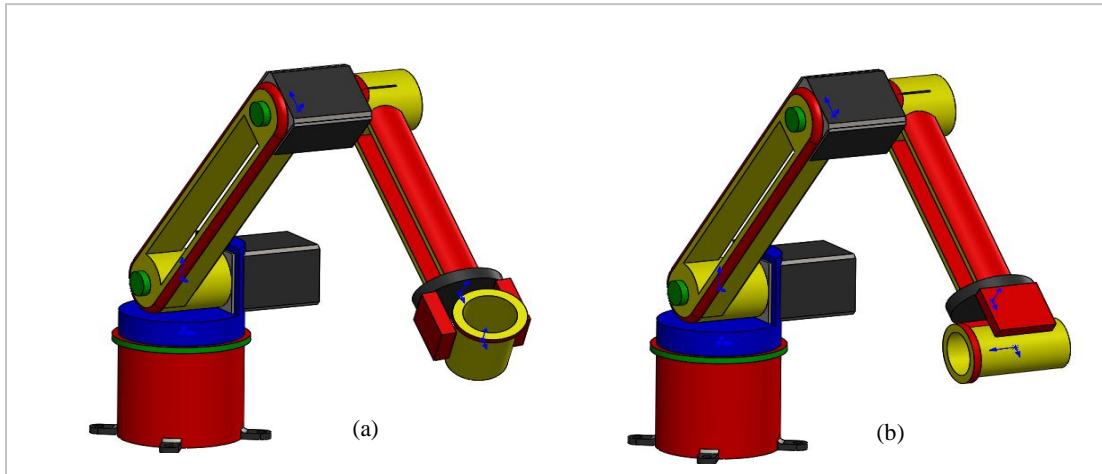


Figure 4 (a) End-effector grasping the crucible (b) End-effector pouring the liquid metal by turning the crucible

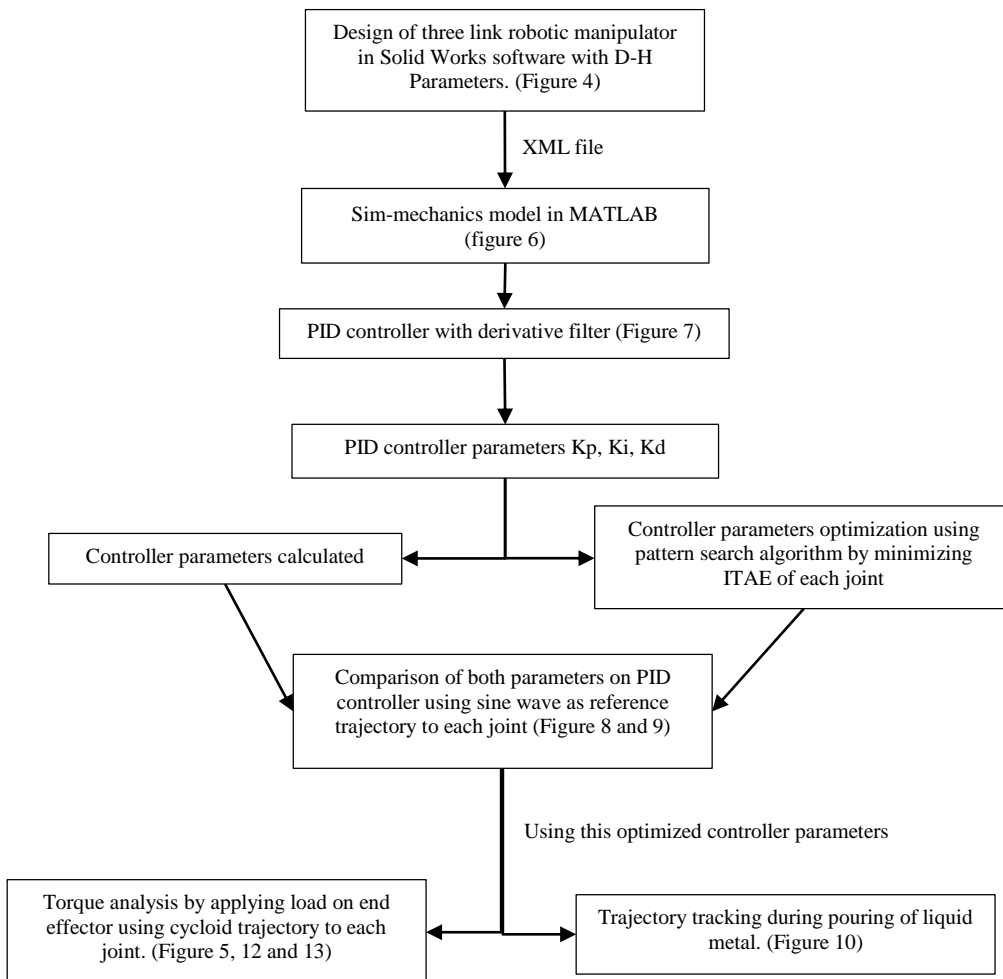


Figure 5 Working procedure

3.4 Trajectory tracking optimizations

A MATLAB, Simulink model of forward dynamics simulation of a three-link robotic manipulator is shown in *Figure 6*. In this robotic manipulator each joint is controlled by a PID controller with derivative filter. The Laplace transfer equation of this PID controller is given by Equation 14-18. A modulus of sinusoidal reference trajectory (Equation 19) was given to each joint actuator for performance evaluation of the controller parameters. The slider gains are used at each joint to control the joint torque range. The Control parameters of the PID controller are tuned according to ITAE criteria Equation 15. ITAE is minimized according to a pattern search algorithm with ITAE criteria [23–24]. The optimization of controller parameter by a pattern search algorithm is completed in 24 iterations. Calculated parameters of the characteristic equation and optimized parameters are given in *Table 3*. Studied related to optimization of PID controller parameters are found in [25–28].

$$C_{PID}(s) = K_p + K_i \frac{1}{s} + K_d \frac{N}{1 + N \frac{1}{s}} \quad (14)$$

Where: K_p = Proportional gain, K_i = Integral gain, K_d = Derivative gain, N = Filter coefficient

$$ITAE = \int_0^T t|e(t)| dt \quad (15)$$

Where: $ITAE$ = Integral of Time multiplied by Absolute Error, t = time, $e(t)$ = error

$$\theta_i(t) = \theta_i(0) + \frac{\theta_i(T) - \theta_i(0)}{T} \left[t - \frac{T}{2\pi} \sin\left(\frac{2\pi}{T} t\right) \right] \quad (16)$$

$$\dot{\theta}_i(t) = \frac{\theta_i(T) - \theta_i(0)}{T} \left[1 - \cos\left(\frac{2\pi}{T} t\right) \right] \quad (17)$$

$$\ddot{\theta}_i(t) = \frac{\theta_i(T) - \theta_i(0)}{T} \left[\frac{2\pi}{T} \sin\left(\frac{2\pi}{T} t\right) \right] \quad (18)$$

For, inverse dynamic simulation of the robotic manipulator, SIM-mechanics 1st generation Simulink model of the robotic manipulator is shown in *Figure 7*. In sub-system (*Figure 8*) joint actuators and joint sensors are connected to each joint of the robotic manipulator. A cycloid trajectory (Equation 16) is applied to each joint (*Figure 7*), velocity (Equation 17) and acceleration (Equation 18) are obtained by the derivative of the Equation 16.

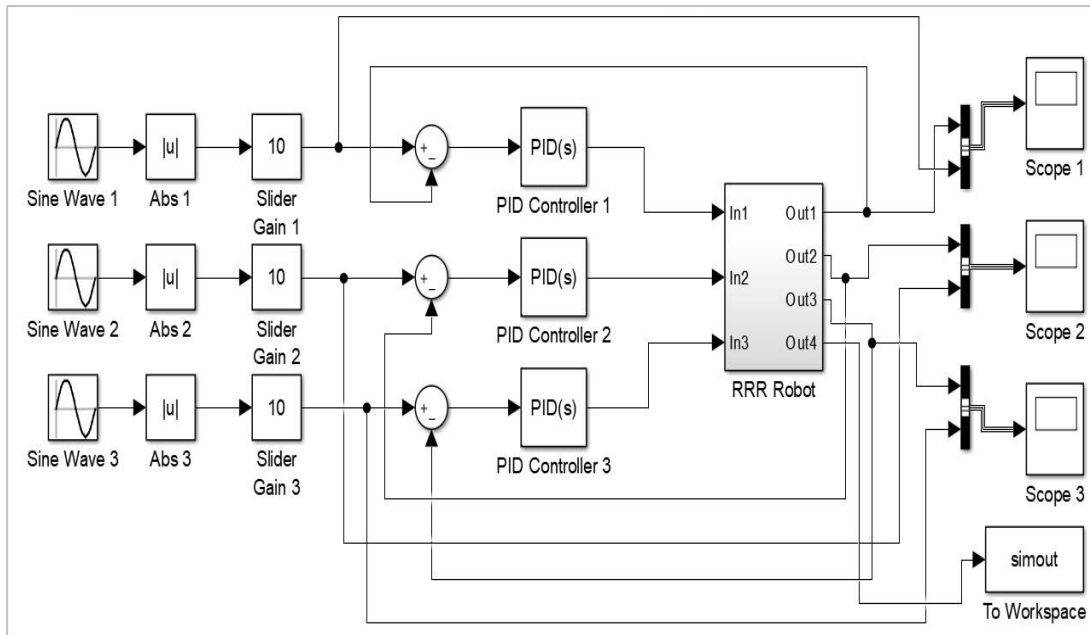


Figure 6 Forward dynamics Simulink model of the robotic manipulator controlled by PID controller

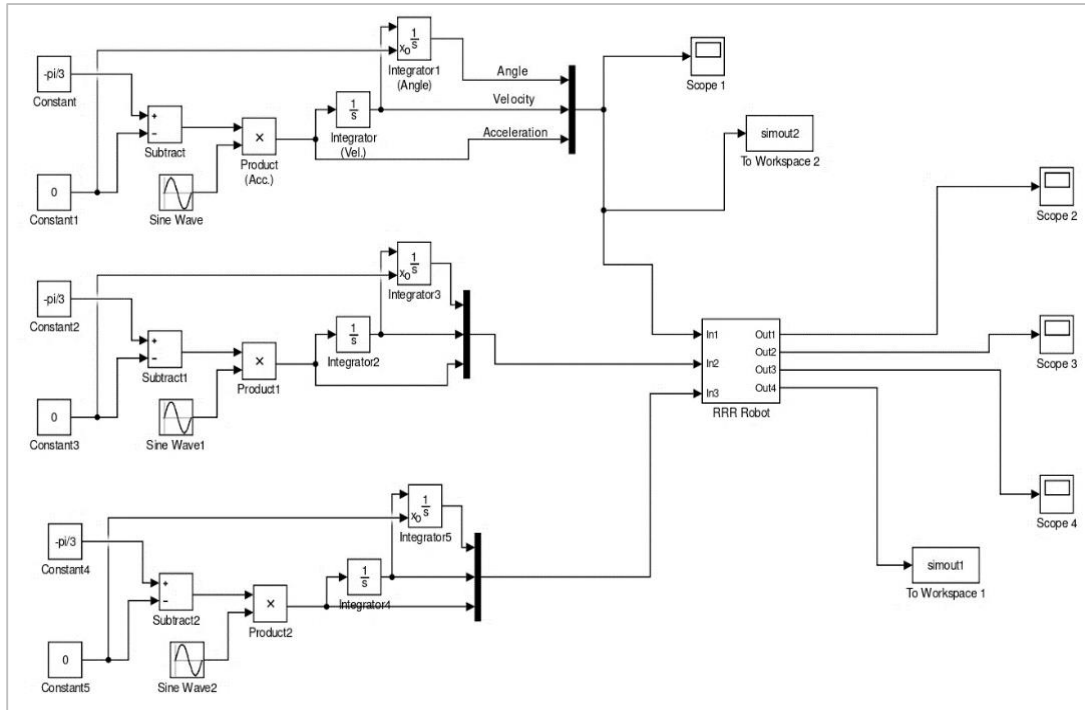


Figure 7 Inverse dynamic of a three-link robotic manipulator sim-mechanics model with cycloid input trajectory at each joint

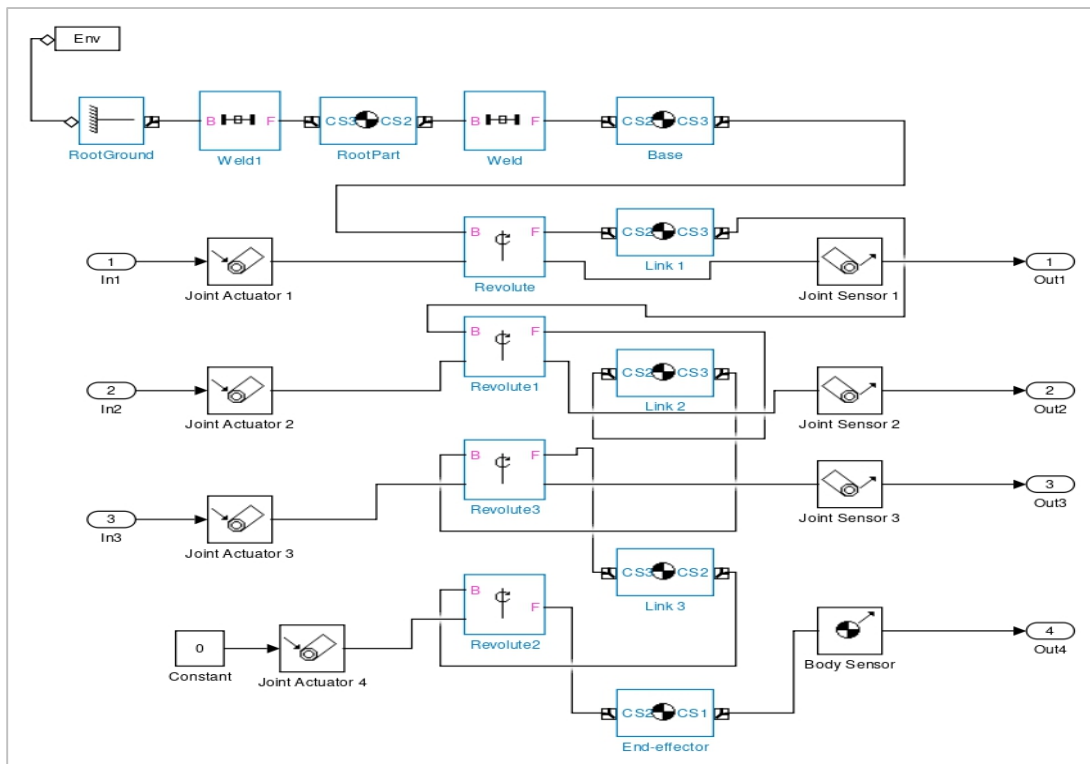


Figure 8 Sub-system of the RRR manipulator as shown in Figure 7

4.Results

Reference trajectory was given to each joint after putting calculated and optimized controller parameters in PID controller from Table 3. Trajectory tracking

obtained by calculated parameters is shown in Figure 9 and by optimized parameters in Figure 10. The results obtain from above studies are described here.

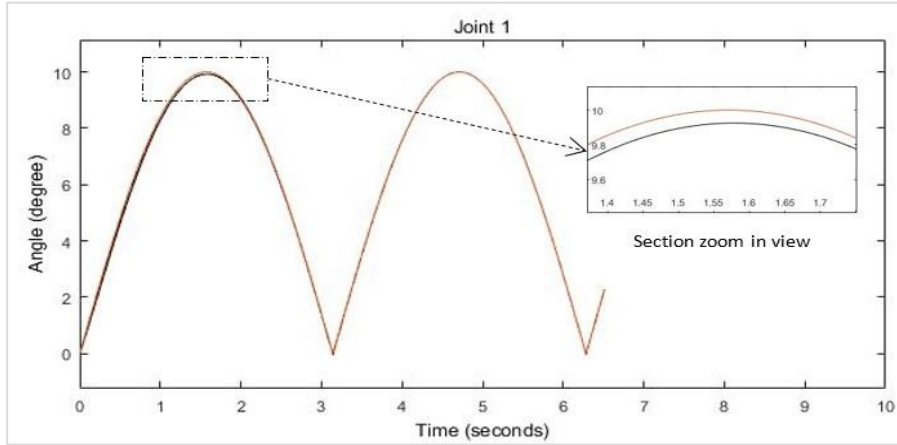


Figure 9 Trajectory tracking error of the robotic manipulator joint 1 with calculated controller parameters (Red line- desired trajectory and Black line- response trajectory)

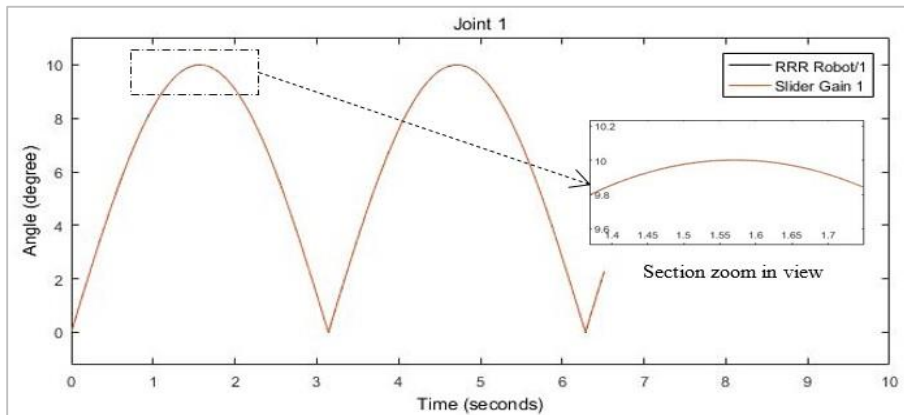


Figure 10 Trajectory tracking error of the robotic manipulator joint 1 with optimization controller parameters (Red line- desired trajectory and Black line- response trajectory)

Table 3 Calculated and optimized parameters of the PID controller. N = 100

Joint No.	Calculated				Optimized			
	Controller parameters				Controller parameters			
	K_p	K_i	K_d	ITAE	K_p	K_i	K_d	ITAE
1	47.3	1123.46	20.3	0.023461	63.8	1378.69	12.9	0.0016313
2	38.7	1454.33	44.67	0.045660	43.25	1945.44	32.33	0.0016257
3	43.56	979.23	38.78	0.067844	56.78	1156.43	31.44	0.0016266

Reference trajectory $\theta_{desired}(t) = 10|\sin(\omega t)|$ (19)

Response trajectory $\theta_{responded}(t) = \theta_{desired}(t) - e(t)$ (20)

Error $e(t) = \theta_{desired} - \theta_{responded}$ (21)

- Output difference in desired and response trajectory tracking by calculated parameters is shown in Figure 9.
- ITAE error $e(t) = \theta_{desired} - \theta_{responded}$ from calculated parameters obtained as 0.023461, 0.045660 and 0.067844 from joint 1, joint 2 and joint 3 respectively.

- Output difference in desired and response trajectory tracking by optimized parameters is shown in *Figure 10*.
- ITAE error $e(t) = \theta_{desired} - \theta_{responded}$ from optimized parameters obtained as 0.0016313, 0.0016257 and 0.0016266 from joint 1, joint 2 and joint 3 respectively.

A casting application for the pouring of liquid metal from the small crucible by the gripper can be seen in *Figure 4*. First a small crucible is grasped (*Figure 4.a*) after that liquid metal is poured into the pouring basin by turning the crucible (*Figure 4.b*). Now, during this task a 3D curve trajectory and scatter point to point trajectory of the robotic manipulator end-effector for

simulation running time of 10 seconds are obtained as shown in *Figure 11(a)* and *Figure 11(b)*. This trajectory is obtained with the optimizing PID controller parameters by attaching a body sensor as shown in *Figure 8* that gives point to point [X, Y, Z] position of end-effector.

A study on end-effector trajectory with simulation running time 200 seconds are shown in *Figure 12*. This trajectory is generated by giving cycloid trajectory angle, velocity, acceleration Equation 14 to 15 to each joint as shown in *Figure 7*. Further a study of variation in joint torque with adding of some value of load or force on end-effector in (-Z) direction are shown in *Figure 13* and *Figure 14*.

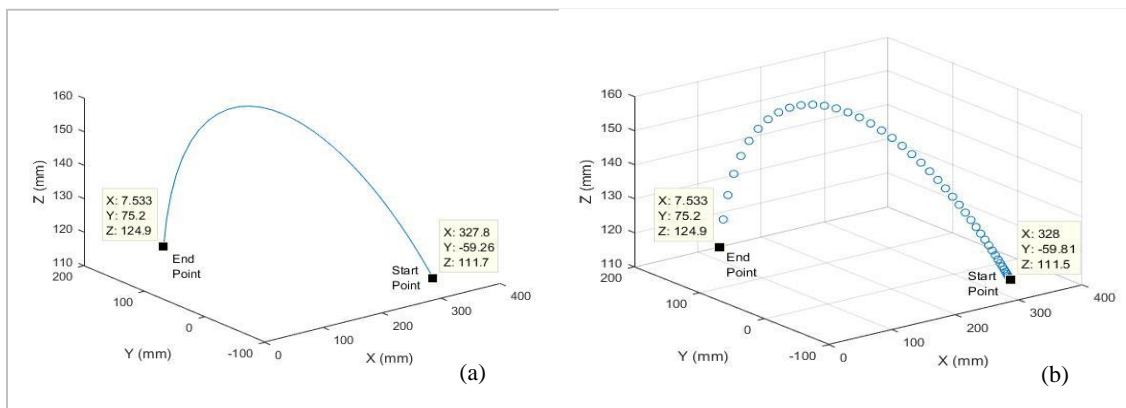


Figure 11 (a) 3D curve trajectory of robotic manipulator end-effector from start point [X Y Z] = [327.9 -59.26 111.6] mm to end point [X Y Z] = [7.533 75.2 124.9] mm (b) 3D Scatter point to point trajectory of robotic manipulator end-effector from start point [X Y Z] = [328 -59.26 111.5] mm to end point [X Y Z] = [7.533 75.2 124.9] mm for simulation running time 10 Sec

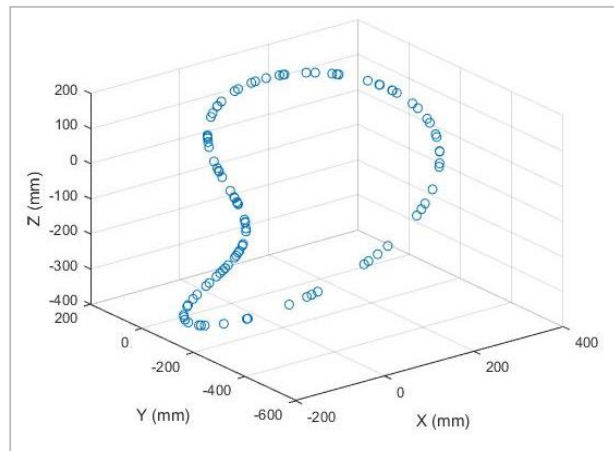


Figure 12 End-effector 3D scatter trajectory for simulation running time of 200 seconds

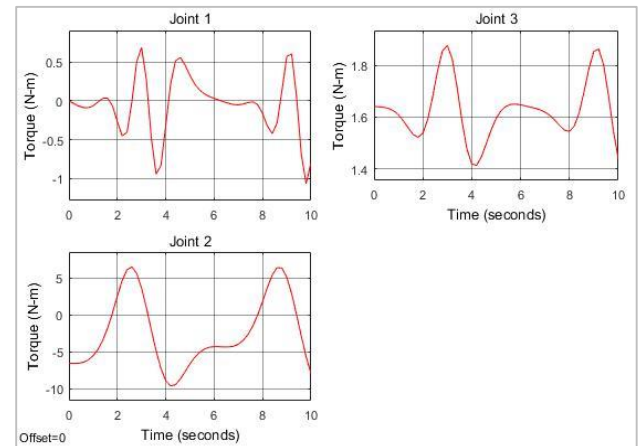


Figure 13 Torque variation of each joint with 0 N force in (-Z) direction on the end-effector for simulation running time of 10 seconds

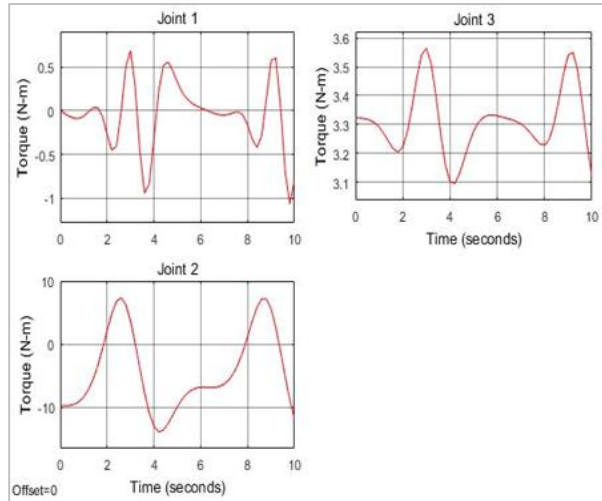


Figure 14 Torque variation of each joint with 10 N force in (-Z) direction on the end-effector for simulation running time of 10 sec

Torque equation of all three joints is described in Equation 13 and *Appendix II*, corresponding cycloid trajectory angle, angular velocity and angular acceleration Equations 14, 15 and 16 respectively were given to each joint (*Figure 7*).

- Variation of torque with time (*Figure 13*) for a simulation running period of 10 seconds were obtained according to the input cycloid trajectory by putting optimized controller parameters (*Table 3*) in PID controller with no load on end effector in (-Z) direction.
- When load of 10 N applied on end effector (-Z) direction, torque increases on each joint that variation with according to the input cycloid trajectory is shown in *Figure 14*.

4.1 Limitations

Limitations of this study are as below-

- Friction and other environmental effects are not considered during simulation so small variation in results is possible during the actual experiment.
- Same number of iterations in different interval of time give different controller parameters of PID controller with small variations.

5. Discussion

When a trajectory of $\theta_{desired}(t) = 10|\sin(\omega t)|$ is given to each joint (*Figure 6*) the output results are obtained according to calculated controller parameters (*Figure 9*) and optimized controller parameters (*Figure 10*). We can see optimized controller parameters give good trajectory tracking as compare to calculated by observing the difference in red line

curve (desired trajectory) and Blackline curve (respond trajectory). The optimized controller reduced the ITAE error by 93.05%, 94.44%, and 97.6% in joint1, joint2, and joint3 respectively. This technology, using SIM-mechanics and pattern search algorithm gives better trajectory tracking as comparable to other methods as SMC in [16] gives high ITAE and the IAE error in two degrees of freedom robot. This can be also compared for the inverse kinematics solution analysis reported in [17] using a GSO hybrid of GA and PSO. Joint torque variation with load (*Figure 14*) and without load (*Figure 13*) varies according to the cycloid joint trajectory using inverse dynamics when load increases on the end-effector this curve gives more amplitude of torque on each joint.

A complete list of abbreviations is shown in *Appendix II*.

6. Conclusion and future work

The aim of this study is to optimize the controller parameters of the three-link robotic manipulator during the pouring of liquid metal in the casting. Simulation and control of the robotic manipulator were conducted in MATLAB. Parameters of PID controller with derivative filter are optimized according to a pattern search algorithm by minimizing ITAE. Optimized controller parameters show good tracking as compare to calculated controller parameters by comparison of ITAE. A study of torque variation based on inverse dynamics was successfully carried out. In the future, other optimization techniques on the different controllers can be used for multi-degree of freedom robots.

Acknowledgment

The authors are grateful for the research support by NDF/ADF AICTE NEW DELHI.

Conflicts of interest

The authors have no conflicts of interest to declare.

References

- [1] Danthala SW, Rao SE, Mannepalli KA, Shilpa D. Robotic manipulator control by using machine learning algorithms: a review. *International Journal of Mechanical and Production Engineering Research and Development*. 2018; 8(5):305-10.
- [2] Annisa J, Darus IM, Tokhi MO, Mohamaddan S. Implementation of PID based controller tuned by evolutionary algorithm for double link flexible robotic manipulator. In *international conference on computational approach in smart systems design and applications 2018* (pp. 1-5). IEEE.
- [3] Chen C, Zhang C, Hu T, Ni H, Luo W. Model-assisted extended state observer-based computed torque control

- for trajectory tracking of uncertain robotic manipulator systems. *International Journal of Advanced Robotic Systems*. 2018; 15(5):1-12.
- [4] Ahmed S, Wang H, Tian Y. Adaptive fractional high-order terminal sliding mode control for nonlinear robotic manipulator under alternating loads. *Asian Journal of Control*. 2021; 23(4):1900-10.
- [5] Chang W, Li Y, Tong S. Adaptive fuzzy backstepping tracking control for flexible robotic manipulator. *IEEE/CAA Journal of Automatica Sinica*. 2018; 8(12):1923-30.
- [6] Guo Q, Zhang Y, Celler BG, Su SW. Neural adaptive backstepping control of a robotic manipulator with prescribed performance constraint. *IEEE Transactions on Neural Networks and Learning Systems*. 2018; 30(12):3572-83.
- [7] González-garcía S, Rodríguez-grce J, Loreto-gómez G, Montaña-serrano VM. Teaching forward kinematics in a robotics course using simulations: transfer to a real-world context using LEGO mindstorms™. *International Journal on Interactive Design and Manufacturing*. 2020; 14(3):773-87.
- [8] Concha SA, Figueroa-rodríguez JF, Fuentes-covarrubias AG, Fuentes-covarrubias R, Gadi SK. Recycling and updating an educational robot manipulator with open-hardware-architecture. *Sensors*. 2020; 20(6):1-22.
- [9] Tang Y, Li X. Simulation research of manipulator control system based on solidworks and simmechanics. *Academic Journal of Computing & Information Science*. 2018; 1(1):19-27.
- [10] Karastoyanov D, Karastanev S. Reuse of industrial robots. *IFAC-PapersOnLine*. 2018; 51(30):44-7.
- [11] Elsis M, Mahmoud K, Lehtonen M, Darwish MM. An improved neural network algorithm to efficiently track various trajectories of robot manipulator arms. *IEEE Access*. 2021; 9:11911-20.
- [12] Mustafa A, Dhar NK, Verma NK. Event-triggered sliding mode control for trajectory tracking of nonlinear systems. *IEEE/CAA Journal of Automatica Sinica*. 2019; 7(1):307-14.
- [13] Pan L, Bao G, Xu F, Zhang L. Adaptive robust sliding mode trajectory tracking control for 6 degree-of-freedom industrial assembly robot with disturbances. *Assembly Automation*. 2018.
- [14] Rahmani M, Komijani H, Rahman MH. New sliding mode control of 2-DOF robot manipulator based on extended grey wolf optimizer. *International Journal of Control, Automation and Systems*. 2020;18:1572-80.
- [15] Sharma R, Rana KP, Kumar V. Comparative study of controller optimization techniques for a robotic manipulator. In proceedings of the third international conference on soft computing for problem solving 2014 (pp. 379-93). Springer, New Delhi.
- [16] Ilgen S, Durdu A, Gulbahce E, Cakan A. Optimal tuning of the SMC parameters for a two-link manipulator co-simulation control. *Elektronika Ir Elektrotechnika*. 2021.
- [17] Soleimani AM, Ramli R. Intelligent trajectory tracking behavior of a multi-joint robotic arm via genetic-swarm optimization for the inverse kinematic solution. *Sensors*. 2021; 21(9):1-18.
- [18] Sherrin MR, Manoharan PS, Anand JV. Optimizing PID parameters for non-linear system using social spider optimization algorithm. In international conference on intelligent computing and control systems 2021 (pp. 706-12). IEEE.
- [19] Kebria PM, Al-wais S, Abdi H, Nahavandi S. Kinematic and dynamic modelling of UR5 manipulator. In international conference on systems, man, and cybernetics 2016 (pp. 4229-34). IEEE.
- [20] Xiao J, Han W, Wang A. Simulation research of a six degrees of freedom manipulator kinematics based on MATLAB toolbox. In international conference on advanced mechatronic systems 2017 (pp. 376-80). IEEE.
- [21] Misra A, Singh G. Kinematic and dynamic analysis of an industrial six-axis robotic manipulator. In international conference on robotics, automation and non-destructive evaluation 2019 (pp.1-14).
- [22] Zarrin A, Azizi S, Aliasghary M. A novel inverse kinematics scheme for the design and fabrication of a five degree of freedom arm robot. *International Journal of Dynamics and Control*. 2020; 8(2):604-14.
- [23] West C, Montazeri A, Monk SD, Taylor CJ. A genetic algorithm approach for parameter optimization of a 7DOF robotic manipulator. *IFAC-PapersOnLine*. 2016; 49(12):1261-6.
- [24] O'mahony T, Downing CJ. Klaudiusz Fatla (Wroclaw University of Technology, Poland). Genetic algorithms for PID parameter optimization, minimizing error criteria.
- [25] Özdemir MT, Öztürk D. Comparative performance analysis of optimal PID parameters tuning based on the optics inspired optimization methods for automatic generation control. *Energies*. 2017; 10(12):1-19.
- [26] Şen Ma, Kalyoncu M. Optimal tuning of PID controller using grey wolf optimizer algorithm for quadruped robot. *Balkan Journal of Electrical and Computer Engineering*. 2018; 6(1):29-35.
- [27] Kumar S, Rastogi V, Gupta P. Trajectory control of single arm underwater flexible welding robot using bond graphs. In the proceedings of the international conference on bond graph modelling and simulation, 2014 (pp. 79-84).
- [28] Kumar S, Kumar S. Trajectory control of underwater flexible robot manipulator using overwhelming controller using bond graph approach. *International Journal for Science and Advance Research in Technology*. 2015; 1(8):50-5.



Mahendra Kumar Jangid is currently a Ph.D. Research Scholar in Department of Mechanical Engineering SLIET Longowal, Punjab. He is doing Ph.D. under National Doctor Fellowship, AICTE, New Delhi selected through GATE. His research interests are Robotic Manipulator, Mobile Robots, Kinematics and Dynamics.

Email: mahendra.jangid0007@gmail.com



Dr. Sunil Kumar is an Assistant Professor at Department of Mechanical Engineering SLIET Longowal, Punjab. He received B.Tech. degree from REC Jalandhar, M.E. degree from TIET Patiala and Ph.D. from SLIET Lonowal. He has teaching experience of 17 years. His research interests are Robotics, Machine Design, Kinematics and Dynamics.

Email: sunil_thappa@yahoo.com



Dr. Jagtar Singh is currently Professor at Department of Mechanical Engineering SLIET Longowal, Punjab. He received the B.E. degree from Gulbarga University, M.E. degree from TIET Patiala and Ph.D. from NIT Kurukshetra. He is a life member of IWS (Indian Welding Society), ISTE (Indian Society for Technical Education) and SMES (SLIET Mechanical Engineering Society). His research interests are Wear, Additive Manufacturing, Farm Machinery, Cryogenic and Welding.

Email: jagtarliet@gmail.com

Appendix I

$$\begin{aligned}
 M_{11} &= \frac{1}{2}m_1(L_1)^2 + \frac{1}{3}m_2(L_2)^2(\cos\theta_2)^2 + \frac{1}{3}m_3(L_3)^2\cos^2(\theta_2 + \theta_3) + m_3(L_2)^2(\cos\theta_2)^2 + m_3L_2L_3\cos(\theta_2 + \theta_3)\cos\theta_3 \\
 M_{12} &= 0, M_{13} = 0, M_{21} = 0, M_{31} = 0, G_{11} = 0 \\
 M_{22} &= \frac{1}{3}m_2(L_2)^2 + \frac{1}{3}m_3(L_3)^2 + m_3(L_2)^2 + m_3L_2L_3\cos\theta_3 \\
 M_{32} &= \frac{1}{3}m_3(L_3)^2 + m_3(L_2)^2 + \frac{1}{3}m_2L_2L_3\cos\theta_3 \\
 M_{33} &= \frac{1}{3}m_3(L_3)^2 \\
 H_{11} &= \left[-\frac{4}{3}m_2(L_2)^2\sin 2\theta_2 - \frac{1}{3}m_3(L_3)^2\sin 2(\theta_2 + \theta_3) - m_3L_2L_3\sin 2(\theta_2 + \theta_3) \right] \dot{\theta}_2\dot{\theta}_1 + \left[-\frac{1}{3}m_3(L_3)^2\sin 2(\theta_2 + \theta_3) - m_3L_2L_3\cos\theta_2\sin(\theta_2 + \theta_3) \right] \dot{\theta}_3\dot{\theta}_1 \\
 H_{21} &= [-m_3L_2L_3\sin\theta_3]\dot{\theta}_2\dot{\theta}_3 + \left[-\frac{1}{2}m_2L_2L_3\sin\theta_3 \right] (\dot{\theta}_3)^2 + \left[\frac{1}{6}m_2(L_2)^2\sin 2\theta_2 + \frac{1}{6}m_3(L_3)^2\sin 2(\theta_2 + \theta_3) + \frac{1}{2}m_3(L_2)^2\sin 2\theta_2 + \frac{1}{2}m_3L_2L_3\sin 2(\theta_2 + \theta_3) \right] (\dot{\theta}_1)^2
 \end{aligned}$$

1267

$$\begin{aligned}
 H_{31} &= \left[\frac{1}{2}m_3L_2L_3\sin\theta_3 \right] (\dot{\theta}_2)^2 + \left[\frac{1}{6}m_3(L_3)^2\sin 2(\theta_2 + \theta_3) + \frac{1}{2}m_3L_2L_3\cos\theta_2\sin(\theta_2 + \theta_3) \right] (\dot{\theta}_1)^2 \\
 G_{21} &= \frac{1}{2}m_2gL_2\cos\theta_2 + \frac{1}{2}m_3gL_3\cos(\theta_2 + \theta_3) + m_3gL_2\cos\theta_2 \\
 G_{31} &= \frac{1}{2}m_3gL_3\cos(\theta_2 + \theta_3)
 \end{aligned}$$

Appendix II

S. No.	Abbreviation	Description
1	ANN	Artificial Neural Network
2	CAD	Computer Aided Design
3	CG	Centre of Gravity
4	CSA	Cuckoo Search Algorithm
5	CTC	Computed Torque Control
6	D-H	Denavit-Hartenberg parameters
7	EGWO	Extended Grey Wolf Optimizer
8	GA	Genetic Algorithm
9	GD	Gradient Descent
10	GSO	Genetic-Swarm Optimization
11	GWO	Grey Wolf Optimizer
12	IAE	Integral of Absolute Error
13	ISE	Integral of Square Error
14	ITAE	Integral of Time multiplied by Absolute Error
15	MNNA	Modified Neural Network Algorithm
16	MSC - ADAMS	Macneal Schwendler Corporation - Automated Dynamic Analysis of Mechanical Systems
17	PDSMC	Proportional Derivative Sliding Mode Control
18	PID	Proportional-Integral-Derivative
19	PS	Pattern Search
20	PSO	Particle Swarm Optimization
21	SA	Simulated Annealing
22	SMC	Sliding Mode Control
23	SS	Simplex Search
24	SSA	Social Spider Algorithm
25	STL	Standard Tessellation Language
26	XML	Extensible Markup Language
27	a_i	Link length
28	α_i	Link twist
29	d_i	Joint offset/distance
30	θ_i	Joint angle
31	$C\alpha_i$	$\cos\alpha_i$
32	$S\alpha_i$	$\sin\alpha_i$
33	m_i	Mass of the i^{th} link
34	L_i	Length of the i^{th} link
35	L	Lagrange function
36	τ_i	Torque of the i^{th} joint