

# Inverse Kinematics and Trajectory Planning of 5 DoF Scorbot-ER Vplus

<sup>1</sup>Rajsinh Jhala  
<sup>1</sup>Research Scholar  
<sup>1</sup>Birla Vishvakarma Mahavidhyalaya

**Abstract** - This paper presents the inverse kinematics and trajectory planning of 5 DoF Scorbot-ER Vplus. Inverse kinematics of the manipulator is computed by using Iterative method. Damped least square (DLS) is an iterative method used to decrease the instability of manipulator near singularity condition. The formulation of inverse kinematics and trajectory planning is programmed using MATLAB/Simulink. Simulation of the manipulator is done by applying MATLAB Simscape Multibody module and importing the 3D model, designed in PTC CREO.

**keywords** - Inverse kinematics, Scorbot-ER Vplus, Jacobian matrix, Damped least square method, Trajectory Planning

## I. INTRODUCTION

Automation has transformed the industrial world, with its quality and sustainability. Over the past decades, industries are moving towards automation. The reason behind this is the need for accuracy, efficiency in repetitive tasks, high productivity and budgeting for industrial processes. Robots are basically highly automated manipulator which are controlled by electronic and computer system. This manipulator has evolved the production and manufacturing unit of the world. Its versatility is a major factor that robotics is a new evolving industrial sector.

For any robotic manipulator, kinematics is the key component to control the robot and perform a task. The robot which is considered here is serial manipulator. To begin with, forward kinematics of the manipulator is evaluated, using Denavit-Hartenberg (D-H) parameter. To obtain the joint variable, inverse kinematics of the manipulator is evaluated for the defined path or position of the robot. For robotic manipulation, paper [1] discussed forward, inverse kinematics and their major categories. To find the solutions for the inverse kinematics problem Algebraic, Geometric and Iterative methods are used. Algebraic solution for a 5-DOF manipulator suggested by [2] follow a given trajectory while keeping the orientation of one axis in the end-effector frame. [3], [4] and [5] discusses the geometric solution to calculate the motion path for robotic arm. Kinematics analysis of robotic arm and the application of cubic polynomial is suggested in [5]. An arm-wrist separateness method is adopted to solve the inverse problem of the robot system in [6].

Three methodologies used to solve the inverse kinematics of a 3 DOF robotic manipulator are highlighted in [7]. These approaches are the algebraic method through MATLAB solve function, Genetic Algorithms (GAs), Artificial Neural Networks (ANNs). Another aspect considered is the trajectory planning of the manipulator, which allows the user to control the desired movement in the joint space. Jacobian inverse method and MATLAB program are used to find joint angle and simulate in Kuka KR5 Arc Robot software model suggested in [8]. Kinematic and dynamic modelling of two link planar robot manipulator having two revolute joints are derived and equations are analysed by obtaining MATLAB code in [9]. Paper [10] has discussed the Arduino based communication for 3R planner manipulator. Inverse kinematics computation task by using four Jacobian-based methods are represented by [11].

There are many iterative method available but from the comparison of different iterative kinematics method the damped least square (DLS) method is superior to all other methods that are reviewed in terms of convergence rate and robustness against computational singularity [12].

## II. SCORBOT-ER VPLUS

Scorbot-ER Vplus is a stand up articulated robot arm, with 5 revolute joints and gripper attached as shown in Figure 1 [13]. Frame Assignment for 5 DoF Scorbot-ER Vplus is shown in Figure 2

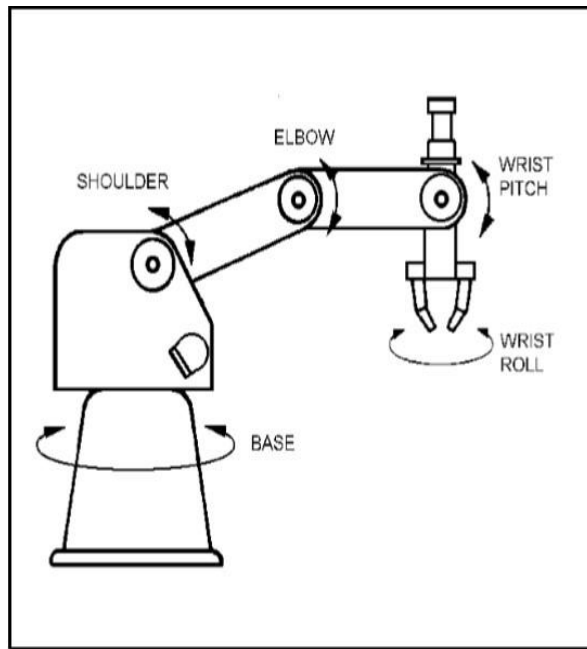


Figure 1 Scorbot-ER Vplus

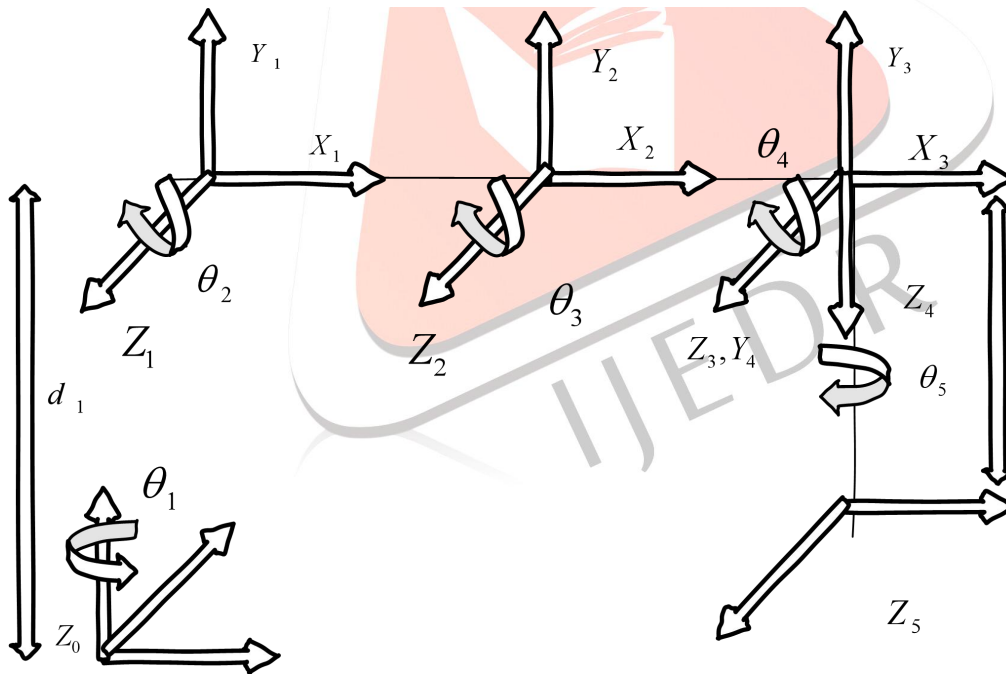


Figure 2 Frame Assignment for 5 DoF Scorbot-ER Vplus

Based on frame assignment D-H parameter are as shown in Table 1

Table 1 D-H parameter for Scorbot-ER Vplus

Joint (i)	Link Twist ( $\alpha_i$ )	Link Length ( $a_i$ )	Joint distance ( $D_i$ )	Joint angle ( $\theta_i$ )
1	$\pi/2$	101.25	334.25	$\theta_1$
2	0	220	0	$\theta_2$
3	0	220	0	$\theta_3$
4	$\pi/2$	0	0	$\theta_4$

5	0	0	137.35	$\theta_5$
---	---	---	--------	------------

Here,  $C_1 = \cos \theta_1, S_1 = \sin \theta_1, C_{23} = \cos(\theta_2 + \theta_3), S_{23} = \sin(\theta_2 + \theta_3), S_{234} = \sin(\theta_2 + \theta_3 + \theta_4), C_{234} = \cos(\theta_2 + \theta_3 + \theta_4)$

$${}^0T_1 = \begin{bmatrix} C_1 & 0 & S_1 & a_1.C_1 \\ S_1 & 0 & -C_1 & a_1.S_1 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) \quad {}^1T_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2.C_2 \\ S_2 & C_2 & 0 & a_2.S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$${}^2T_3 = \begin{bmatrix} C_3 & -S_3 & 0 & a_3.C_3 \\ S_3 & C_3 & 0 & a_3.S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) \quad {}^3T_4 = \begin{bmatrix} C_4 & 0 & S_4 & 0 \\ S_4 & 0 & -C_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$${}^4T_5 = \begin{bmatrix} C_5 & -S_5 & 0 & 0 \\ S_5 & C_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The overall transformation matrix can be obtained using equation 6 and 7.

$${}^0T_5 = {}^0T_1 \times {}^1T_2 \times {}^2T_3 \times {}^3T_4 \times {}^4T_5 \quad (6)$$

$${}^0T_5 = \begin{bmatrix} C_1C_{234}C_5 + S_1S_5 & -C_1C_{234}S_5 + S_1C_5 & C_1S_{234} & C_1(d_5S_{234} + a_3C_{23} + a_2C_2 + a_1) \\ S_1C_{234}C_5 - C_1S_5 & -S_1C_{234}S_5 - C_1C_5 & S_1S_{234} & S_1(d_5S_{234} + a_3C_{23} + a_2C_2 + a_1) \\ S_{234}C_5 & -S_{234}S_5 & C_{234} & (d_5C_{234} + a_3S_{23} + a_2S_2 + d_1) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

### III. JACOBIAN MATRIX

For the solution of inverse kinematics of 5-DOF by iterative method, Jacobian matrix is computed in equation. Jacobian matrix is of size  $(6 \times n)$  which is non square or not of full rank, so its direct matrix inversion is not possible. Here n stands for number of degree of freedom, so for 5-DOF the matrix size is  $(6 \times 5)$ . Jacobian Matrix is computed using equation 8 and Table 2.

$$J_i = \begin{Bmatrix} \text{Linear} \\ \text{Rotational} \end{Bmatrix}_{6 \times n} \quad (8)$$

Table 2 Jacobian Matrix formation table

	Prismatic	Revolute
<b>Linear</b>	$R_{i-1}^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$	$R_{i-1}^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times (d_n^0 - d_{i-1}^0)$

<b>Rotational</b>	$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$	$R_{i-1}^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
-------------------	---	---

Where R = Rotational matrix, d = Translational Matrix

Jacobian matrix for Scorbot-ER Vplus basic formula is given by equation 9.

$$J_i = \begin{bmatrix} R_{i-1}^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times (d_n^0 - d_{i-1}^0) \\ R_{i-1}^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{9}$$

Where i = number of rotational matrix, n = degree of freedom

The overall Jacobian Matrix is given by equation 10,

$$J = [J_1 \quad J_2 \quad J_3 \quad J_4 \quad J_5]_{6 \times 5} \tag{10}$$

$$J_1 = \begin{bmatrix} -S_1[d_5 C_{234} + a_3 S_{23} + a_2 C_2 + a_1] \\ C_1[d_5 + S_{234} + a_3 C_{23} + a_2 C_2 + a_1] \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{11}$$

$$J_2 = \begin{bmatrix} -C_1[d_5 C_{234} + a_3 S_{23} + a_2 S_2] \\ -S_1[d_5 C_{234} + a_3 S_{23} + a_2 S_2] \\ S_1^2[d_5 S_{234} + a_3 C_{23} + a_2 C_2] + C_1^2[d_5 S_{234} + a_3 C_{23} + a_2 C_2] \\ S_1 \\ -C_1 \\ 0 \end{bmatrix} \tag{12}$$

$$J_3 = \begin{bmatrix} -C_1[d_5 C_{234} + a_3 S_{23}] \\ -S_1[d_5 C_{234} + a_3 S_{23}] \\ S_1^2[d_5 S_{234} + a_3 C_{23}] + C_1^2[d_5 S_{234} + a_3 C_{23}] \\ S_1 \\ -C_1 \\ 0 \end{bmatrix} \tag{13}$$

$$J_4 = \begin{bmatrix} -C_1[d_5 C_{234}] \\ -S_1[d_5 C_{234}] \\ S_1^2[d_5 S_{234}] + C_1^2[d_5 S_{234}] \\ S_1 \\ -C_1 \\ 0 \end{bmatrix} \tag{14}$$

$$J_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ C_1 S_{234} \\ S_1 S_{234} \\ C_{234} \end{bmatrix} \tag{15}$$

**IV. INVERSE KINEMATICS USING DLS METHOD**

Damped least square method is used over other method because method uses damping factor to solve the instability problem near singularity. The damping constant depends on the details of the multibody and the target positions and must be chosen carefully to make equation numerically stable. Damping factor selection can increase and decrease convergence rate according to factor chosen [1].

$$\Delta q = J^T (JJ^T + \lambda^2 I)^{-1} .e \tag{16}$$

Where  $\Delta q$  = change in velocity

$J$  = Jacobian matrix

$\lambda$  = Non zero damping constant

$e$  = error

**V. TRAJECTORY PLANNING**

It is a planning of the desired movements of the manipulator. Usually, the user is requested to define the point and general features of the trajectory (e.g. initial/final points, duration, maximum velocity, etc.), and the real computation of the trajectory is demanded to the control system. For trajectory planning points are assigned to follow by robot, this point are given according to path required for operation.

Hermite cubic spline is selected for trajectory planning, it is a method of interpolating data points as a polynomial function.

Hermite Cubic Spline is a method of interpolating data points as a polynomial function.

Basic Hermite Function :-

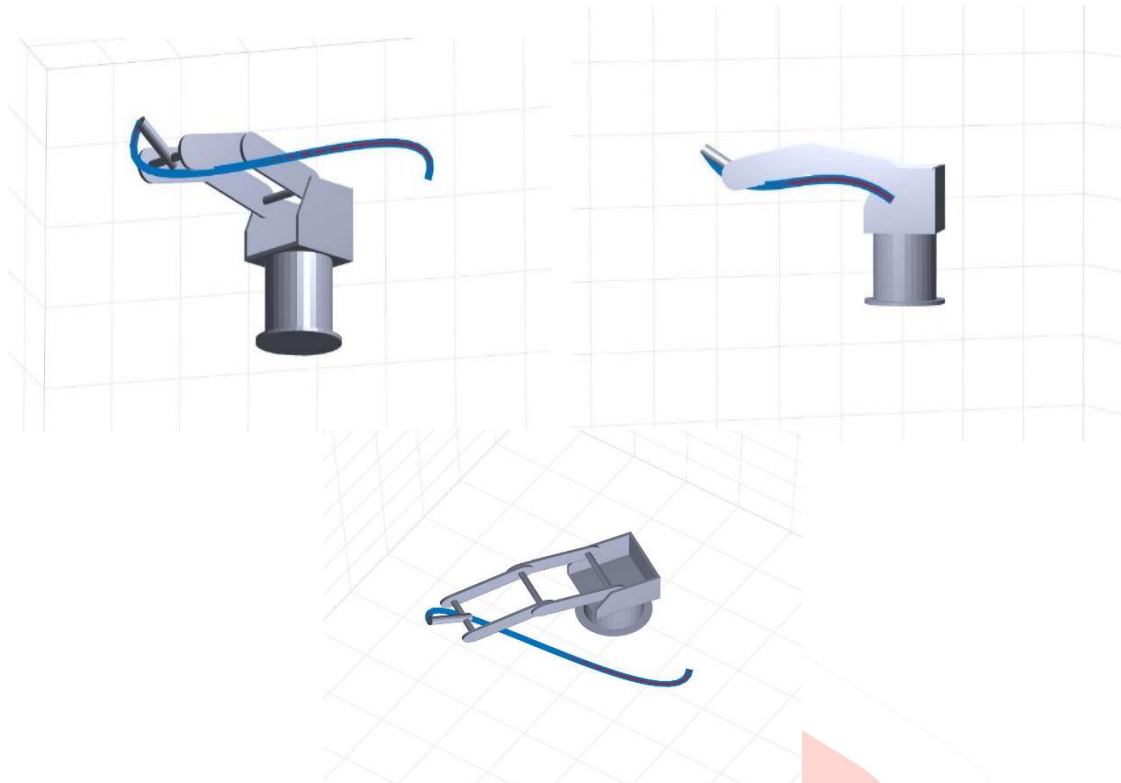
$$\text{Vector :-} \quad [u^3 \quad u^2 \quad u \quad 1] \quad MH = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

$$\text{Equation :-} \quad P(u) = [u^3 \quad u^2 \quad u \quad 1] \begin{bmatrix} 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_0' \\ p_1' \end{bmatrix} \tag{19}$$

Where  $p_0$  = initial position or point,  $p_1$  = final position or point,  $p_0'$  = tangent vector of initial position,  $p_1'$  = tangent vector of final position

Figure 3 shows the Scorbot-ER Vplus robot in MATLAB platform following the trajectory for the hermite cubic spline. The spline is in three dimensional form to show the reach and path planning.

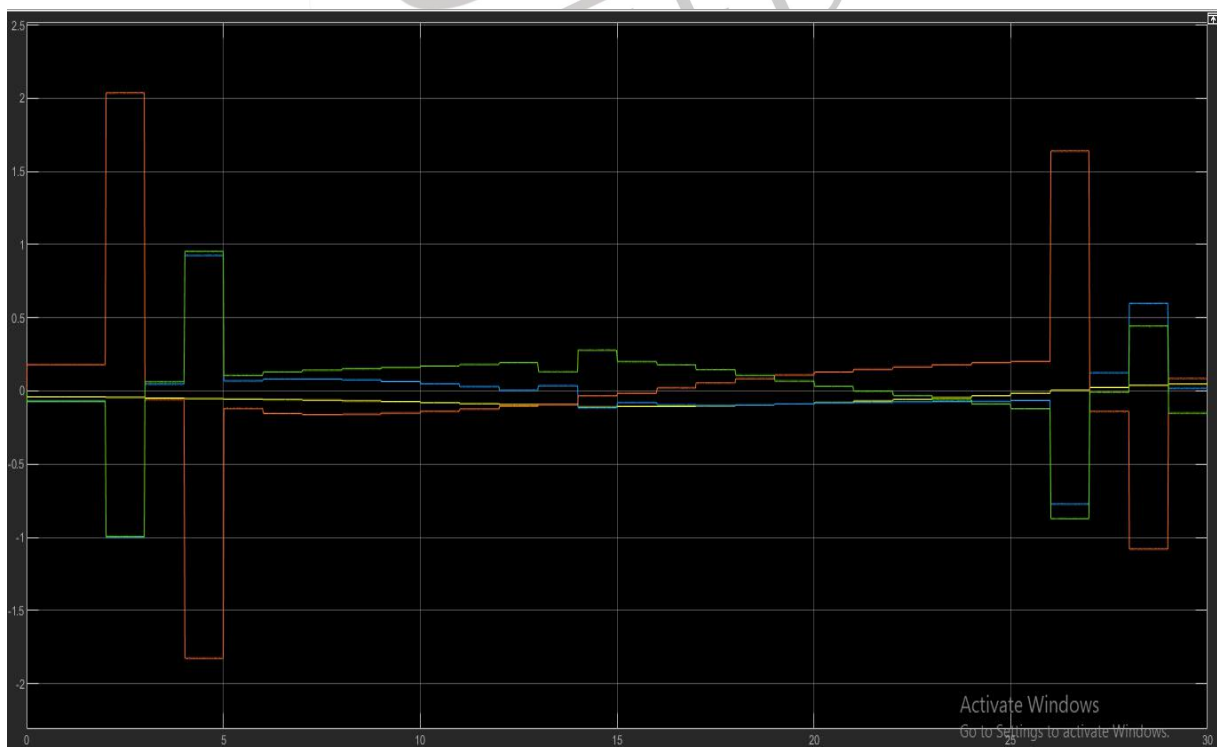
**VI. RESULT**



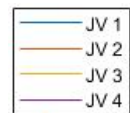
**Figure 3 Trajectory planning for hermite cubic spline**

After the trajectory planning the joint variable need to be calculated and controlled. The variable are calculated according to curves or spline for which trajectory planning is done. The acceleration, velocity and torque of the system is evaluated using Matlab program. Hermite cubic spline point are calculated and trajectory planning is done. Joint variable for individual joint are compared in the graph.

Figure 4 show the comparison of velocity at different joint, where JV1, JV2, JV3 and JV4 are joint velocity at joint 1, 2, 3 and 4 respectively.



**Figure 4 Velocity comparison graph**



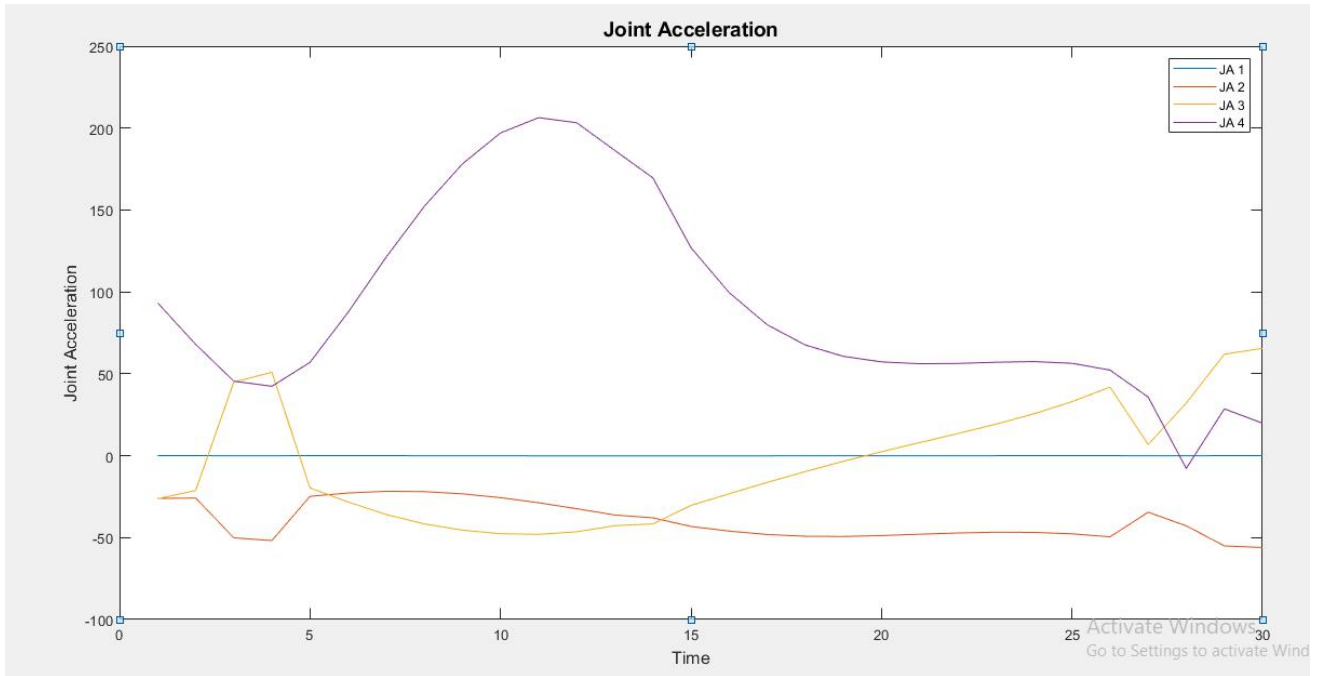
The result show a comparison between different joint with each other to get better idea of to how to control the variable according to required trajectory. The joint variable which required more torque or the value of velocity which is giving more fluctuation.

According to the comparison the joint variable of shoulder joint has largest fluctuation compared to other joints. As it has to overcome the self-weight of all the link which are moving away from the center of gravity and load applied to it.

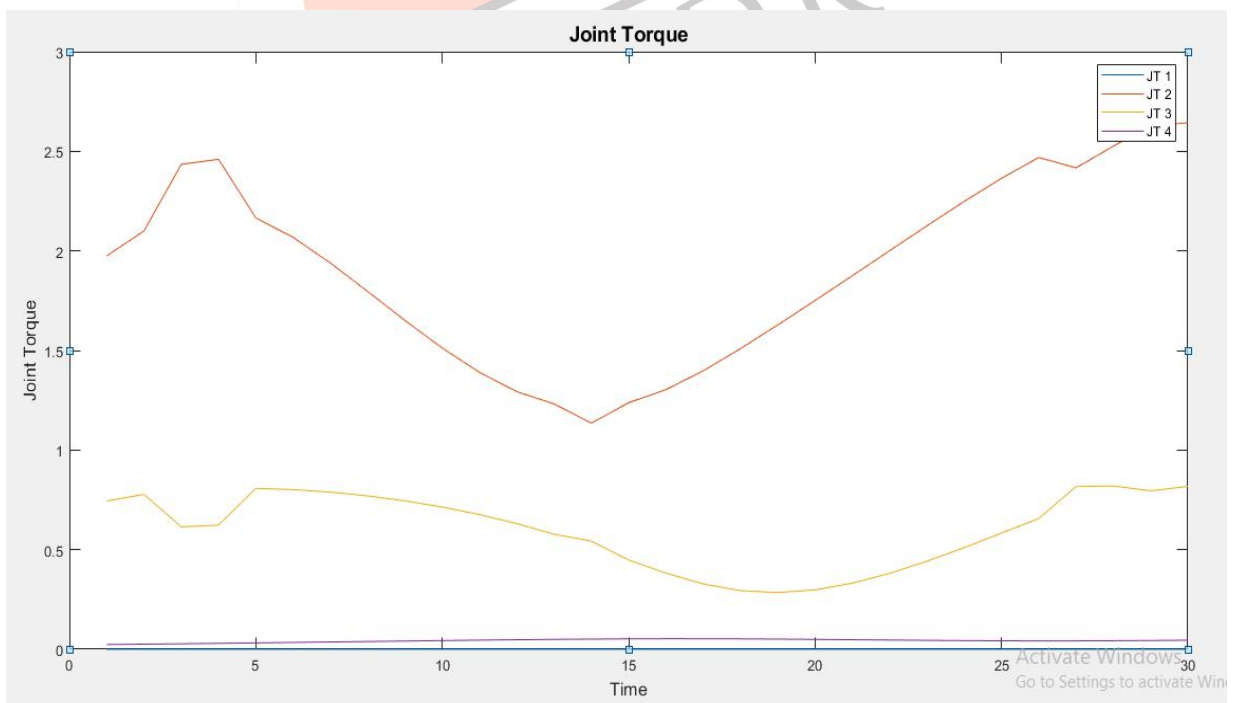
Figure 5 show the comparison of acceleration at different joint, where JA1, JA2, JA3 and JA4 are joint acceleration at joint 1, 2, 3 and 4 respectively.

And comparison of torque at different joint is shown in figure 6, where JT1, JT2, JT3 and JT4 are joint torque at joint 1, 2, 3 and 4 respectively.

Velocity (m/s)



**Figure 5 Acceleration comparison graph**



**Figure 6 Torque comparison graph**

## VII. CONCLUSION

In this paper inverse kinematics of 5 DoF robot can be calculated using Matlab program. Damped least square method is used to evaluate the inverse kinematics to minimized instability near singularity. Trajectory planning is done using the MATLAB program and simulation in obtained in Simscape Multibody (MATLAB). Joint variables are compared and obtained to optimize the overall performance of the system. According to the result, the program for trajectory planning gives reliable performance, when MATLAB simulation is used to check the path following of robot.

## REFERENCES

- [1] J. L. Y. C. a. A. S. A. Aristidou, "Inverse Kinematics Techniques in Computer Graphics: A Survey," *Computer graphics forum*, p. 1–24, 2017.
- [2] C. A. A. C. J. Q. G. H. H. De Xu, "An Analysis of the Inverse Kinematics for a 5-DOF Manipulator," *International Journal of Automation and Computing*, pp. 114-124, 2005.
- [3] K. M. D. B. A. A. V.N. Iliukhina, "The Modelling of Inverse Kinematics for 5 DOF manipulator," *Procedia Engineering* 176, p. 498 – 505, 2017 .
- [4] D. L. A. M. Alaa Hassan Shabeeb, "Inverse Kinematics Analysis using Close Form solution method for 5 DOF Robot Manipulate," *Engineering & Technical Journal*, 2015.
- [5] M. W. N. L. a. Y. T. Xiaojie Zhao, "Trajectory Planning for 6-DOF Robotic Arm Based on Quintic Polynormial," *Advances in Intelligent Systems Research, 2nd International Conference on Control, Automation, and Artificial Intelligence (CAAI)*, 2017.
- [6] C. X. Q. P. X. Z. a. X. L. Zongxing Lu, "Inverse Kinematic Analysis and Evaluation of a Robot for Nondestructive Testing Application," *Hindawi Publishing Corporation, Journal of Robotics Volume*, pp. 1-7, 2015.
- [7] J. F. D. H. S. K. M. M. M. A. F. M. K. P. Lucas B. de Souza, "Inverse Kinematics and Trajectory Planning Analysis of a Robotic Manipulator," *International Journal of Advanced Engineering Research and Science (IJAERS)*, 2018.
- [8] G. Y. Avantsa V.S.S.Somasundara, "Robotic path planning and simulation by jacobian inverse for industrial applications," *Procedia Computer Science* 133, p. 338–347, 2018 .
- [9] J. G. Pushkal Badoniya, "Two Link Planar Robot Manipulator Mechanism Analysis with MATLAB," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, 2018.
- [10] W. P. E. Moe Thu Zar, "Point to Point Trajectory Control of 3R Planar Robot Arm," *International Journal of Science, Engineering and Technology Research (IJSETR)*, 2017.
- [11] M. O. IGNACY DULE, BA, "A COMPARISON OF JACOBIAN-BASED METHODS OF INVERSE," *Int. J. Appl. Math. Comput. Sci.*, p. 373–382, 2013.
- [12] D. V. P. T. Manan Kalasariya, "Comparative Study of Iterative Inverse," *International Journal of Engineering Research & Technology (IJERT)*, 2018.
- [13] *Scorbot ER Vplus user's manual*, Eshed robotec, Feb. 1998.
- [14] *MATLAB/SIMULINK documentation*.