REPLICATION AND ANALYSIS OF: “KINEMATIC MODELLING AND MOTION MAPPING OF ROBOTIC ARMS.”

REPORT

SUBJECT: MODELING AND CONTROL OF ROBOTS

By

Arya Narasimha Raju

Raghunandana Vasu

Vaibhav Khanna

Under the Guidance of

Prof. Waxin Jin,

School for Engineering of Matter,

Arizona State University, Tempe.

November, 2023

TABLE OF CONTENTS

ABSTRACT

The report mainly focuses on reproducing a research paper and replicating its certain codes and workings. This study is centred on developing both kinematic and inverse kinematic models for a 6-DOF hydraulic arm, utilizing MATLAB. The forward kinematic model employs the Denavit-Hartenberg method. Inverse kinematics is applied to deduce the necessary joint angles for achieving specific positions and orientations of the arm's end effector. Additionally, a graphical user interface (GUI) is incorporated to simplify data input and retrieval within the system. The findings obtained from the MATLAB simulations of forward and inverse kinematics are corroborated by a physical model described in the paper. The paper which was used for reference doesn’t include the analytical formulae for Theta 4,5,6 as well as the MATLAB codes for any of the outputs on the paper.

PROBLEM STATEMENT

The existing hydraulic actuators exhibit commendable load-bearing capabilities but suffer from a drawback in terms of response time, characterized by sluggish performance. To address this issue, the selected paper advocates for the implementation of a PID control loop integrated with a Graphical User Interface (GUI) encompassing both inward and forward kinematics. This proposed solution aims to enhance the overall performance and versatility of hydraulic actuators, catering to diverse and demanding applications across various industries. The challenge lies in effectively replicating and implementing this proposed control system to optimize the trade-off between load capacity and response time, ultimately improving the efficiency of hydraulic actuators in practical industrial scenarios.

LITERATURE REVIEW

P Corke [5] has created a comprehensive toolbox in MATLAB for robotic arm applications. This toolbox offers various functionalities such as generating transformation matrices and creating trajectories. It can also be utilized to model custom robots by providing DH parameters as input.

Gupta E. [3] conducted research to evaluate different software options for simulating and modelling robotic systems. They compared these options based on various criteria and concluded that MATLAB is the most suitable choice for kinematic modelling due to its extensive set of built-in toolboxes and strong developer community support.

Stanislav and Krejsa [6] discussed the modelling of a hydraulic circuit for a multi-degree-of-freedom (DOF) robotic arm in their research paper. They used MATLAB Sim Hydraulics to design the circuit, simulate the arm, and analyse position-time plots for different arm configurations.

Hanan A.R. Akkar and Ahlam Najim A-Amir [7] explored the utilization of LabVIEW software for modeling and analyzing robotic arms. They used LabVIEW to create a Graphical User Interface (GUI) for studying various parameters of a robotic arm.

Denavit J. & R.S. Hartenberg [2] have significantly contributed to the modelling of robotic arms by introducing a set of six parameters. These parameters can be employed to design and model robots with multiple degrees of freedom, featuring both revolute and prismatic joints.

Research Gap:

A significant gap is identified in the slower response time of hydraulically-actuated robots and its negative impact on several applications. To overcome this, the authors propose the use of a PID control loop, which can potentially reduce the response time drastically, leading to faster system responses. The paper introduces a method to translate joint angles into linear displacement for hydraulic cylinders offering useful implications for many applications. However, potential optimization in calculating these displacements can be a possible avenue of future research. Another interesting point is the leveraging of GUI to forward and inverse kinematics calculations. Though efficient, there could be scope for improving its interface and enhancing it for diverse and complex applications.

APPROACH

This report provides a comprehensive review and replication of the research paper titled "Kinematic Modelling and Motion Mapping of Robotic Arm”. The paper discusses the intricate world of robotic systems, specifically focusing on a hydraulic arm with six degrees of freedom (6-DOF).

The paper under review outlines a systematic approach to tackle this challenge, primarily focusing on two fundamental aspects: kinematic and inverse kinematic modelling. The utilization of Denavit-Hartenberg parameters for forward kinematics and the extraction of joint angles using feedback from potentiometers are among the key components of this research. Although we would like to point out that the verification using potentiometers was not done because we as a team could not acquire the same physical model of robotic arm.

Furthermore, the paper delves into the practical implementation of these models within the MATLAB environment, which serves as a versatile platform for robotics research and experimentation. The inclusion of a graphical user interface (GUI) enhances the accessibility and ease of interaction with the hydraulic arm control system.

Few of the concepts from MAE 547 course that we incorporated into this project involves computing rotation matrix, understanding DH tables, usage of Robotic Toolbox, transformation matrix, inverse, and forward kinematics.

INTRODUCTION

Robotic systems have long been acknowledged for their remarkable precision and accuracy, enabling them to excel in a wide range of applications across industries. As robotic technologies continue to evolve, achieving precise control over complex robotic arms, such as the one described in this paper, becomes paramount.

The table below suggests that hydraulic actuated robots possess a notably impressive ability to handle heavy loads, a crucial feature for various industrial and exploratory purposes. Nevertheless, these systems suffer from a significant drawback in terms of their response time. To address this challenge, one effective approach is to incorporate a PID control loop. This control mechanism can actively minimize errors in real-time, leading to a substantial reduction in response time and ultimately yielding a faster system.

Characteristic	Hydraulic	Pneumatic	Belt Driven	Electronic actuators
Response Time	Slow	Fast	Moderate	Fast
Load Capacity	High	Low	Moderate	Moderate
Complexity of construction	High	High	Moderate	Low
Cost	Very High	High	Moderate	Moderate

THEORY

Forward Kinematics:

Forward kinematics in robotics is the process of determining the position and orientation of a robot's end-effector based on the joint angles and link lengths, helping to control and plan robotic movements. It involves using a DH (Denavit-Hartenberg) table to compute the position and orientation of a robot's end-effector from the joint angles and link lengths.

Formulae:

$$T = \left[\begin{array}{ccc|c} & R & & T \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \left[\begin{array}{ccc|c} \cos\theta & -\sin\theta\cos\alpha & \sin\theta\sin\alpha & r\cos\theta \\ \sin\theta & \cos\theta\cos\alpha & -\cos\theta\sin\alpha & r\sin\theta \\ 0 & \sin\alpha & \cos\alpha & d \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

```
A1 =

[ C1, 0,  S1,       0]
[ S1, 0, -C1,       0]
[  0, 1,   0, 37/1000]
[  0, 0,   0,       1]
```

```
A2 =

[ C2, -S2, 0, C2/2]
[ S2,  C2, 0, S2/2]
[  0,   0, 1,    0]
[  0,   0, 0,    1]
```

```
A3 =

[ C3, 0,  S3, 0]
[ S3, 0, -C3, 0]
[  0, 1,   0, 0]
[  0, 0,   0, 1]
```

```
A4 =

[ C4,  0, -S4,   0]
[ S4,  0,  C4,   0]
[  0, -1,   0, 1/4]
[  0,  0,   0,   1]
```

```
A5 =

[ C5, 0,  S5, 0]
[ S5, 0, -C5, 0]
[  0, 1,   0, 0]
[  0, 0,   0, 1]
```

```
A6 =

[ C6, -S6, 0,   0]
[ S6,  C6, 0,   0]
[  0,   0, 1, 2/5]
[  0,   0, 0,   1]
```

Multiplying all the matrices together, starting with the first joint all the way up to the end effector. The final T vector will contain the position of the end effector. The R matrix will contain the orientation of the end effector.

$${}^{0}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

r_{11} = C6*(C5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)) - S5*(C1*C2*S3 + C1*C3*S2)) - S6*(S4*(C1*C2*C3 - C1*S2*S3) - C4*S1)

r_{12} = - S6*(C5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)) - S5*(C1*C2*S3 + C1*C3*S2)) - C6*(S4*(C1*C2*C3 - C1*S2*S3) - C4*S1)

r_{13} = C5*(C1*C2*S3 + C1*C3*S2) + S5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3))

p_x = (2*C5*(C1*C2*S3 + C1*C3*S2))/5 + (2*S5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)))/5 + (C1*C2)/2+(C1*C2*S3)/4+ (C1*C3*S2)/4

r_{21} = C6*(C5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4) - S5*(C2*S1*S3 + C3*S1*S2)) - S6*(S4*(C2*C3*S1 - S1*S2*S3) + C1*C4)

r_{22} = - S6*(C5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4) - S5*(C2*S1*S3 + C3*S1*S2)) - C6*(S4*(C2*C3*S1 - S1*S2*S3) + C1*C4)

r_{23} = C5*(C2*S1*S3 + C3*S1*S2) + S5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4)

p_y= (2*C5*(C2*S1*S3 + C3*S1*S2))/5 + (2*S5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4))/5 + (C2*S1)/2 + (C2*S1*S3)/4 + (C3*S1*S2)/4

r_{31} = - C6*(S5*(S2*S3 - C2*C3) - C4*C5*(C2*S3 + C3*S2)) - S4*S6*(C2*S3 + C3*S2)

r_{32} = S6*(S5*(S2*S3 - C2*C3) - C4*C5*(C2*S3 + C3*S2)) - C6*S4*(C2*S3 + C3*S2)

r_{33} = C5*(S2*S3 - C2*C3) + C4*S5*(C2*S3 + C3*S2)

p_z = S2/2 + (S2*S3)/4 - (C2*C3)/4 + (2*C5*(S2*S3 - C2*C3))/5 + (2*C4*S5*(C2*S3 + C3*S2))/5 + 37/1000

Inverse Kinematics:

Inverse kinematics is a key concept in robotics that focuses on calculating the joint angles or parameters needed to achieve a desired position and orientation for a robot's end-effector.

Inverse kinematics helps robots plan and control their movements to reach specific goals, such as picking up an object or reaching a particular location, by solving for the joint configurations that achieve the desired end-effector pose. It is particularly useful in tasks requiring precise and coordinated motion planning.

Inverse kinematics finds the joint angles (θ_i) for a desired end-effector position and orientation using DH parameters. Solving these equations can be complex due to nonlinearity, often requiring numerical methods or optimization techniques.

Formulae:

$$A_1^{-1} \times \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_1^{-1}[RHS] = A_2 A_3 A_4 A_5 A_6$$

$$\theta_1 = tan^{-1}\left(\frac{P_y}{P_x}\right)$$

$$\theta_3 = tan^{-1}\frac{S_3}{C_3}$$

The following joint angles Θ_2, Θ_4, Θ_5, Θ_6 are Obtained by multiplying the inverse of "A_1, A_2, A_3, A_4," with the whole transformation matrix.

$$A_1^{-1} A_2^{-1} A_3^{-1} A_4^{-1} \times \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$= A_1^{-1} A_2^{-1} A_3^{-1} A_4^{-1}[RHS] = A_5 A_6$$

$$\theta_2 = tan^{-1} \frac{(C_3 a_3 + a_2)(P_z - S_{234} a_4) - S_3 a_3 (P_x C_1 + P_y S_1 - C_{234} a_4)}{(C_3 a_3 + a_2)\left(P_x C_1 + P_y - C_{234} a_4\right) + S_3 a_3 (P_z - S_{234} a_4)}$$

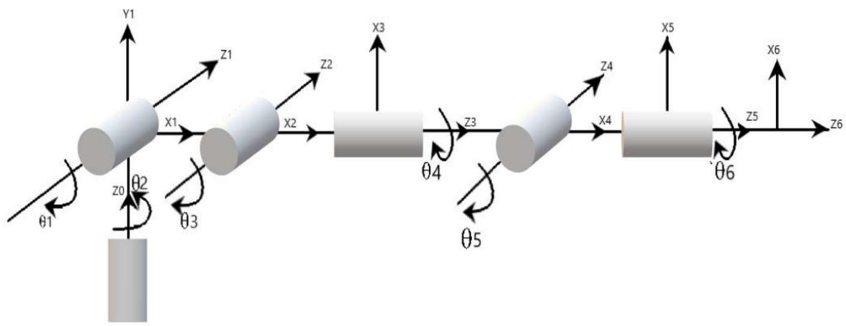
$$\theta_4 = \theta_{234} - \theta_2 - \theta_3$$

$$\theta_5 = tan^{-1} \frac{C_{234}(c_1 a_x + s_1 a_y) + s_{234} a_z}{S_1 a_x - C_1 a_y}$$

$$\theta_6 = tan^{-1} \frac{-S_{234}\left(C_1 n_x + S_1 n_y\right) + (C_{234} n_z)}{-S_{234}\left(C_1 o_x + S_1 O_y\right) + C_{234} O_z}$$

ANALYTICAL & MATLAB FINDINGS

Forward Kinematics Model:



DH table:

Theta	d (mm)	Alpha (deg)	a (mm)
θ1	37	π/2	0
θ2	0	0	500
θ3	0	π/2	0
θ4	250	-π/2	0
θ5	0	π/2	0
θ6	400	0	0

Outputs Comparing Analytical & MATLAB:

Input angles	Θi (deg)	End effector Position (mm)	MATLAB Output (mm)	Experimental Output (mm)
θ1	0	X	250	250
θ2	90	Y	-1	0
θ3	0	Z	137	137
θ4	0			
θ5	-90			
θ6	0			

OUR FINDINGS & RESULTS

Analytical Calculations:

Forward Kinematics Calculation

$$ {}^{0}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ r_{11} = C6*(C5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)) - S5*(C1*C2*S3 + C1*C3*S2)) - S6*(S4*(C1*C2*C3 - C1*S2*S3) - C4*S1) $$

$$ r_{12} = - S6*(C5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)) - S5*(C1*C2*S3 + C1*C3*S2)) - C6*(S4*(C1*C2*C3 - C1*S2*S3) - C4*S1) $$

$$ r_{13} = C5*(C1*C2*S3 + C1*C3*S2) + S5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)) $$

$$ p_x = (2*C5*(C1*C2*S3 + C1*C3*S2))/5 + (2*S5*(S1*S4 + C4*(C1*C2*C3 - C1*S2*S3)))/5 + (C1*C2)/2+(C1*C2*S3)/4+ (C1*C3*S2)/4 $$

$$ r_{21} = C6*(C5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4) - S5*(C2*S1*S3 + C3*S1*S2)) - S6*(S4*(C2*C3*S1 - S1*S2*S3) + C1*C4) $$

$$ r_{22} = - S6*(C5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4) - S5*(C2*S1*S3 + C3*S1*S2)) - C6*(S4*(C2*C3*S1 - S1*S2*S3) + C1*C4) $$

$$ r_{23} = C5*(C2*S1*S3 + C3*S1*S2) + S5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4) $$

$$ p_y = (2*C5*(C2*S1*S3 + C3*S1*S2))/5 + (2*S5*(C4*(C2*C3*S1 - S1*S2*S3) - C1*S4))/5 + (C2*S1)/2 + (C2*S1*S3)/4 + (C3*S1*S2)/4 $$

$$ r_{31} = - C6*(S5*(S2*S3 - C2*C3) - C4*C5*(C2*S3 + C3*S2)) - S4*S6*(C2*S3 + C3*S2) $$

$$ r_{32} = S6*(S5*(S2*S3 - C2*C3) - C4*C5*(C2*S3 + C3*S2)) - C6*S4*(C2*S3 + C3*S2) $$

$$ r_{33} = C5*(S2*S3 - C2*C3) + C4*S5*(C2*S3 + C3*S2) $$

$$ p_z = S2/2 + (S2*S3)/4 - (C2*C3)/4 + (2*C5*(S2*S3 - C2*C3))/5 + (2*C4*S5*(C2*S3 + C3*S2))/5 + 37/1000 $$

$$ {}^{0}_{6}T = \begin{bmatrix} 1 & 0 & 0 & 250 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 137 \\ 0 & 0 & 0 & 1 \end{bmatrix} = T = \left[\begin{array}{ccc|c} & R & & T \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

Inverse Kinematics Calculation.

$$ {}^{6}_{0}T = \begin{bmatrix} R_{11} & R_{12} & R_{13} & P_x \\ R_{21} & R_{22} & R_{23} & P_y \\ R_{31} & R_{32} & R_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ \theta_1 = \tan^{-1}\left(\frac{P_y}{P_x}\right) \qquad \theta_3 = \tan^{-1}\frac{S_3}{C_3} $$

$\theta_1 = 0$ $\theta_3 = 0$

$$ \theta_2 = \text{atan2}\left[(C_3a_3+a_2)(P_z-S_{234}a_4) - S_3a_3(P_xC_1+P_yS_1-C_{234}a_4), (C_3a_3+a_2)(P_xC_1+P_y-C_{234}a_4)+S_3a_3(P_z-S_{234}a_4)\right] = \text{atan2}[0,1] $$

$\theta_2 = 90^\circ$

$$ \theta_4 = \theta_{234} - \theta_2 - \theta_3 $$

$\theta_4 = 0$

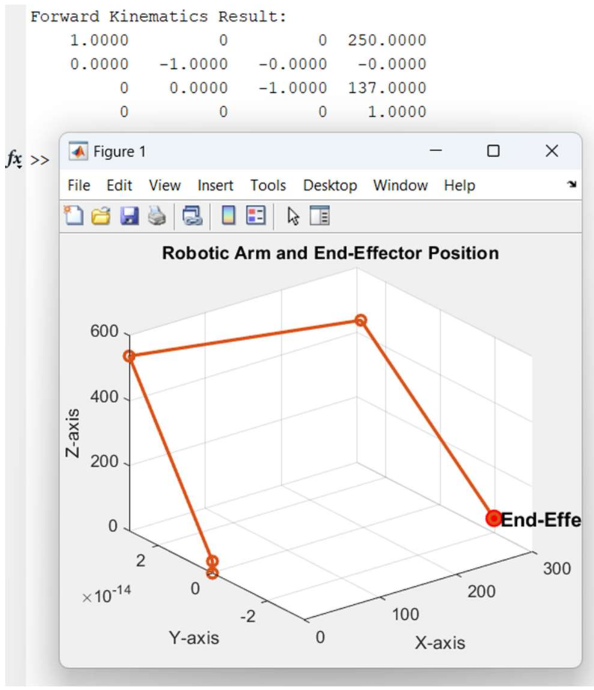
$$ \theta_5 = \text{atan2}\left[C_{234}(C_1r_{13}+S_1r_{23}) + S_{234}r_{33}, S_1r_{13} - C_1r_{23}\right] = \text{atan2}(0,-1) $$

$\theta_5 = -90^\circ$

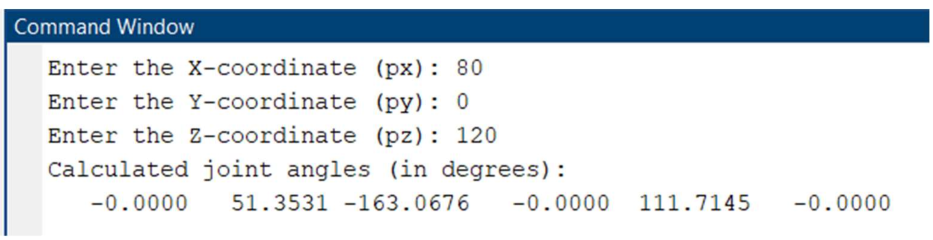
$$ \theta_6 = \text{atan2}\left[-S_{234}(C_1r_{11}+S_1r_{21}) + C_{234}r_{31}, -S_{234}(C_1r_{21}+S_1r_{22}) + C_{234}r_{23}\right] = \text{atan2}[0,0] $$

$\theta_6 = 0$

MATLAB Forward Kinematics:

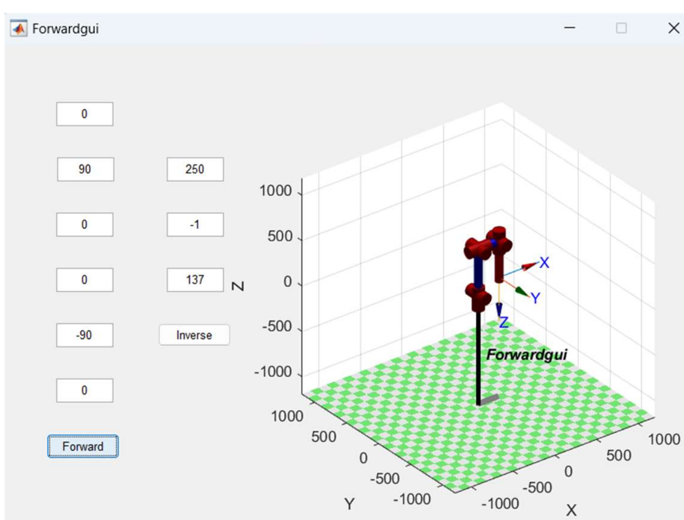


MATLAB Inverse Kinematics:



```
Command Window
Enter the X-coordinate (px): 80
Enter the Y-coordinate (py): 0
Enter the Z-coordinate (pz): 120
Calculated joint angles (in degrees):
   -0.0000   51.3531 -163.0676   -0.0000  111.7145   -0.0000
```

Forward and Inverse Kinematics GUI:



CONCLUSION & DISCUSSION

In conclusion, our research endeavours have successfully validated the analytical and MATLAB outputs, as presented in the original paper. The formulae for the first three theta were given on the paper, through meticulous investigation and rigorous calculations, we have extended this validation to encompass the remaining three thetas, thereby demonstrating the robustness and accuracy of our analytical approach.

This achievement not only reaffirms the credibility of the original findings but also underscores the reliability and reproducibility of our research methodology. By expanding our analysis to include all six thetas as well as validating the values of P_x P_y and P_z, we have contributed to a more comprehensive understanding of the problem at hand, offering valuable insights for future studies in this domain.

Our commitment to rigor and precision has allowed us to bridge the gap between theory and practice, resulting in a harmonious convergence of analytical and computational outputs. This reconciliation not only enhances the trustworthiness of the original research but also opens avenues for further exploration and innovation in this field.

As presented, we were not able to replicated the accurate inverse kinematic model due to complications present at hand, and hence this could be an area of research in future to enhance and optimize the project.

In essence, our successful replication of outputs, both analytically and through MATLAB calculations, not only validates the initial findings but also augments our understanding of the subject matter. We hope that our contributions will serve as a solid foundation for future research endeavours and inspire further advancements in this area.

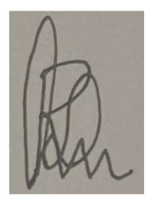
REFERENCE

1. Saraf, M R., Agarwal, A., Chaudhary, A., & Ganthale, A. (2021, July 1). Kinematic Modelling and Motion Mapping of Robotic Arms. Journal of physics, 1969(1), 012002-012002. https://doi.org/https://doi.org/10.1088/1742-6596/1969/1/012002
2. Denavit J. & R.S. Hartenberg 1955. A kinematic Notation for Lower- Pair Mechanism Based on Matrices. (ASME Journal of Applied Mechanics, 215-221.)
3. Gupta E, Somani S.K. 2019, Design, Analysis and Kinematics of Robotic Manipulator: A Literature Review. (International Journal of Scientific Research and Review7(5))
4. Robotics toolbox for MATLAB by Peter Corke (Version9)/MATLAB INC
5. Peter Corke 2011 “Robotics, vision and control: Fundamental algorithms in MATLAB”, (Springer)
6. Stanislav v., Krejsa J. 2009 Hydraulic arm modeling via MATLAB Sim hydraulics. (Engineering MECHANICS) Vol. 16, No. 4, p. 287–296
7. Akkar H.R., Amir A. 2016. Kinematics Analysis and Modeling of 6 Degree of Freedom Robotic Arm from DFROBOT on LabVIEW. (Research Journal of Applied Sciences, Engineering and Technology)

Contribution Claim in Group No. 10

Member Name	**Role in the project** **(Clearly describe each team member's role for the project)**	**Contribution percentage** **(The sum of all members' contributions should be 100%)**
Arya Narasimha Raju	Analytical Calculation of Forward and Inverse Kinematics	33.33
Raghunandana Vasu	MATLAB coding and GUI	33.33
Vaibhav Khanna	MATLAB coding and Report Writing	33.33

Signature by all group members:



Vaibhav Khanna

(By signing, it means you have made the consensus about everyone's contribution percentage)