Experiment - 8: Kinematics of spatial manipulators - PUMA 560 and Svaya SR-L6

Aim: Forward and Inverse kinematics of PUMA 560 and Svaya SR-L6.

In this experiment, the operations of PUMA 560 manipulator and Svaya SR-L6 robot are demonstrated. The forward and inverse kinematics of the spatial manipulator are performed and executed on the robot. The python library of MuJoCo is also demonstrated for trajectory planning in joint space and trajectory tracking of the end-effector.

Software setup:

Software: PyCharm Community Edition

Python packages: NumPy, MuJoCo, Matplotlib

Tasks:

1. Learn the basic commands of PUMA 560 and Svaya SR-L6 to operate in Joint and Cartesian coordinates.
2. Write DH parameters and forward kinematics for PUMA 560 and Svaya SR-L6.
3. Learn the inverse kinematics of Svaya SR-L6.
4. Write the code for end effector trajectory of circle/ellipse and track the trajectory using Svaya SR-L6.

Sr. no.	Topic	Task	Report
1	PUMA 560 and Svaya SR-L6	Learn the basic commands of PUMA 560 and Svaya SR-L6 to operate in Joint and Cartesian Co-ordinates.	a. What is the programming command to save or teach the current position of PUMA 560 manipulator? b. What are the joint limits in the teach pendent of Svaya SR-L6? c. Name the python function of Svaya API which commands the robot to follow a trajectory.
2	Forward kinematics	Write the DH parameters and transformation matrices for PUMA 560 and Svaya SR-L6. Run the given code 'run_fwdkin.py' and find the position of the end effector given the joint angles of Svaya SR-L6. Modify the code to change the joint trajectory to cubic polynomial with zero initial and final joint velocities.	a. DH parameters table and kinematics of robot's end effector. b. Report the positions of end effector of Svaya SR-L6 robot for the given joint angles. c. Plot the joint angles with time for cubic polynomial trajectory.

3	Inverse kinematics	Run the code 'run_invkin.py' and find the joint angles for the given position of the end effector.	a. Report the joint angles for the given position of the end-effector.
4	Trajectory tracking	Run the code 'run_traj-tracking.py'. Modify the code to change the radius of the circle/ellipse.	a. Change the shape to ellipse and report the trajectory tracking. b. For 300mm radius of circle, report the tracking of the trajectory.

Task manual:

1. **PUMA 560 and Svaya SR-L6**

Figure 1: PUMA 560

Figure 2: Svaya SR-L6

2. **Forward kinematics**

Write DH parameters of the PUMA 560 robot shown in Figure 3 and 4 in the following table:

	θ_i	d_i	a_{i-1}	α_{i-1}
1.				
2.				
3.				
4.				
5.				
6.				

Similarly, assign the frames and write DH parameters table for the Svaya SR-L6 robot shown in Figure 5.

Report the expression for position and orientation of the end-effector of PUMA 560 and Svaya SR-L6 in terms of the joint angles.

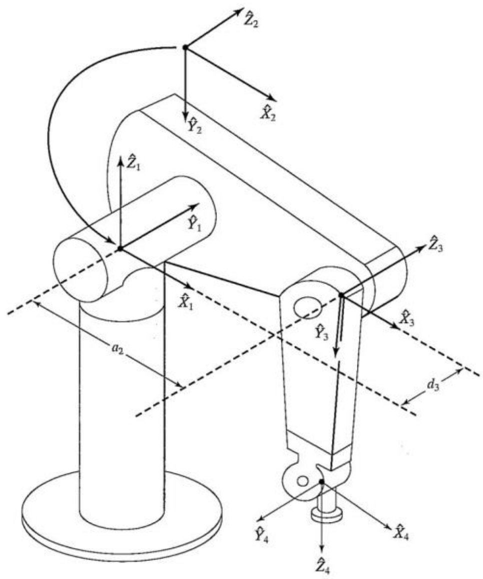


Figure 3: Frame assignments for PUMA 560 (source: Intro to Robotics: John Craig, 2006)

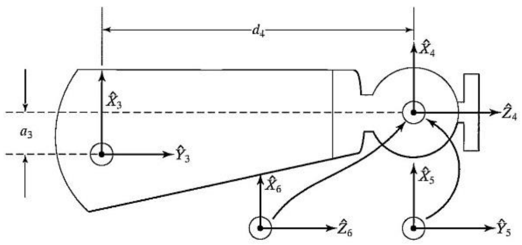


Figure 4: Frame assignments for forearm of PUMA 560 (source: Intro to Robotics: John Craig, 2006)

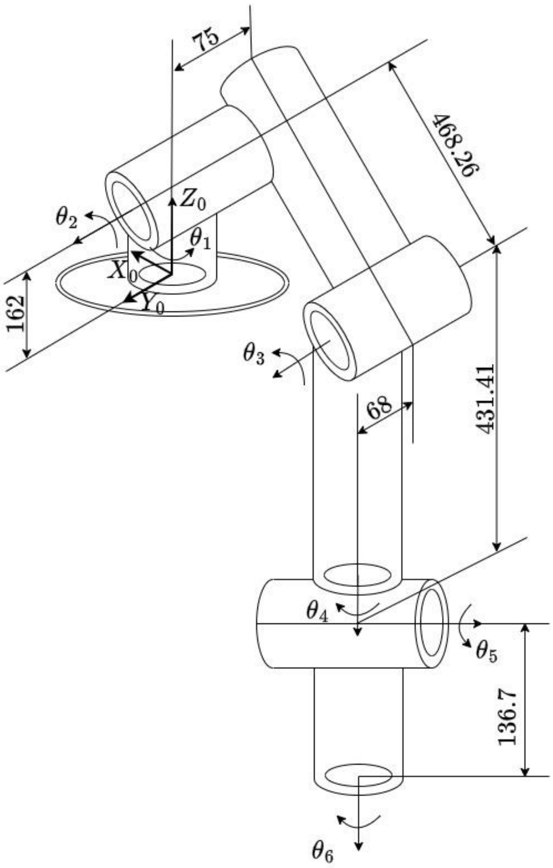
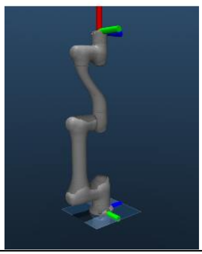


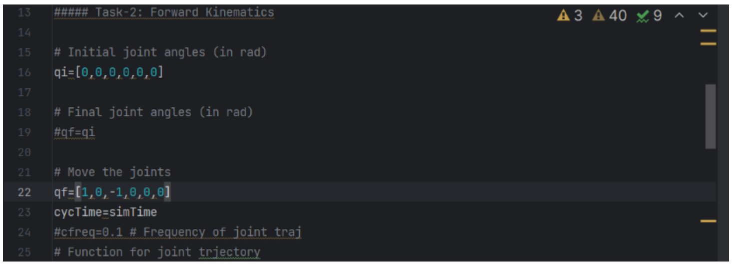
Figure 5: Dimensions of Svaya SR-L6

Follow the below steps for forward kinematics of Svaya SR-L6:

i. Run the python code 'run_fwdkin.py'.
ii. Change the values in variable 'qi' and fill the following table:

S. No.	Joint angles (in rad)	Position of End effector (in mm)	Photos
1.	[0,0,0,0,0,0]		
2.			
3.			
4.			
5.			

iii. In python code 'run_fwdkin.py', do the following change:

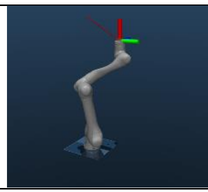


Modify the linear joint trajectory to cubic polynomial with zero initial and final velocities and report the plot of joint angles with time.

3. Inverse kinematics

Follow the below steps for inverse kinematics of Svaya SR-L6:

i. Run the python code 'run_invkin.py'.
ii. Change the position of end effector in the code and fill the following table:

S. No.	Position of End effector (in m)	Joint angles (in rad)	Photos
1.	[0.2, 0.2, 1]		
2.			
3.			
4.			
5.			

4. Trajectory tracking

For tracking a circular/elliptical trajectory of the end effector, follow the below steps:

i. Run the python code 'run_traj-tracking.py'.
ii. Modify the code to track an ellipse, take a screenshot and report the tracked trajectory.
iii. Change the radius of the circular trajectory to 300 mm and report the tracked trajectory.

REPORT FORMAT

(Submit a single pdf file of your report to Hello IITK after renaming the file in the format: ME381_E8_Name_RollNo_GroupNumber)

Experiment No. 8

Experiment Title: Kinematics of spatial manipulators - PUMA 560 and Svaya SR-L6

Group No.:

Name:

Roll no.

Results for the task:

Task no.	**Topic**	**Task**	**Report**
1			
2			
3			
4			