MEAM 520 LAB 1 REPORT

Name: Vanshil Atul Shah, Divyanshu Rajendra Sahu

5.1

Forward Kinematics Methodology:

The following steps were taken to get the full forward kinematics of the Panda Robot:

We used the brute force method to find out the forward kinematics of the robot. Each joint's axis of rotation was the z axis of the assigned coordinate frame. The x axis of the frames was assigned by referencing the coordinate frames in Rviz and the y axis was calculated accordingly using the right-hand rule. The positive direction of rotation for the joint's is considered in the counterclockwise direction.

The relative transformation between each consecutive pair of coordinate frames was calculated by using the brute force method.

For each consecutive pair of frames, say f+1 and f, the rotation is seen by fixing the f frame and viewing both the frames w.r.t the z axis of the f+1 frame. Projecting f+1 frame vectors on the f frame gives us the rotation matrix. For the translation, we look at the amount of displacement along each vector of the f frame. Combining the rotation and translation gives us the homogenous matrix for consecutive frames.

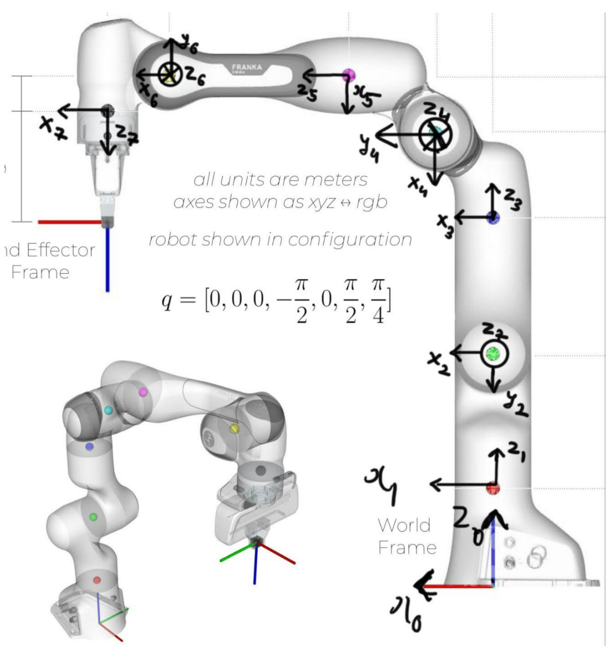
We post multiply the pose of the end effector w.r.t the base frame we post multiply consecutive pair homogeneous matrices. We also account for the initial configuration of the robot by premultiplying the homogenous matrix of the consecutive pair of frames with the inverted angle of the initial configuration. For example: The initial configuration of the joint 4 is given as -pi/2. To account for this, when we calculate the transformation we premultiply a rotation of pi/2 about the z axis of the 4th joint.

$$T_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0.141 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_2^1 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_2 & -c\theta_2 & 0 & 0.192 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ 0 & 0 & 1 & -0.195 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_4^3 = \begin{bmatrix} s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -c\theta_4 & s\theta_4 & 0 & 0.121 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & -0.083 \\ 0 & 0 & 1 & 0.125 \\ -s\theta_5 & -c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_6^5 = \begin{bmatrix} -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & -1 & 0.015 \\ c\theta_6 & -s\theta_6 & 0 & 0.259 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_6^7 = \begin{bmatrix} c\theta_7 & -s\theta_7 & 0 & 0.088 \\ 0 & 0 & -1 & -0.051 \\ s\theta_7 & c\theta_7 & 0 & 0.15 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_e^7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.159 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Inverse kinematics Methodology:

For solving inverse kinematics, we are given the position of the end effector and rotation in the world frame. We need to calculate the joint positions of the robot that will result in that particular pose of the end effector. Here we calculate the overall transformation matrix of the end effector with respect to the base frame from given targets. This is given by the following matrix:

$$T_e^0 = \begin{bmatrix} c\alpha & -s\alpha & 0 & x \\ s\alpha & c\alpha & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now we also know the forward kinematics transformation matrix front the robot calculated in prelab. That is a function of the joint angles. We equate that forward kinematics transformation matrix with the above matrix to find the joint angles.

$$T_e^0 = \begin{bmatrix} c(\theta_1+\theta_2+\theta_3) & -s(\theta_1+\theta_2+\theta_3) & 0 & a_3c(\theta_1+\theta_2+\theta_3)+a_2c(\theta_1+\theta_2)+a_1c(\theta_1) \\ s(\theta_1+\theta_2+\theta_3) & c(\theta_1+\theta_2+\theta_3) & 0 & a_3s(\theta_1+\theta_2+\theta_3)+a_2s(\theta_1+\theta_2)+a_1s(\theta_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This will result in solving the elements through a series of algebraic and trigonometric transformations resulting in the following set of equations.

We assign dx and dy the following values as they will come in handy for solving the equations later.

$$dx = x - a3 * cos(\alpha)$$
$$dy = y - a3 * sin(\alpha)$$
$$\text{sq\_sum} = dx^2 + dy^2$$

We get the following equation by comparing the terms of the rotation part of both the matrices:

$$\alpha = \theta_1 + \theta_2 + \theta_3$$

We calculated theta_2 by squaring and adding the translation terms of the matrix. There would be two possibilities of theta_2; The negative one would correspond to elbow up and the positive one would correspond to elbow down. After this, We take arctan to figure out both possible values of theta_2.

$$c\theta_2 = (\text{sq\_sum} - a_1^2 - a_2^2)/2a_1a_2$$
$$s\theta_2\_a = -\sqrt{1 - c\theta_2^2}$$
$$s\theta_2\_b = \sqrt{1 - c\theta_2^2}$$
$$\theta_2\_a = tan^{-1}(s\theta_2\_a/c\theta_2)$$
$$\theta_2\_b = tan^{-1}(s\theta_2\_b/c\theta_2)$$

According to the sign of sin(theta_2), there are two possibilities of theta_1 which can be seen from the equation below. We now take arctan to figure out both the possible values of theta_1.

$$s\theta_1 = \frac{((a_1 + a_2c\theta_2) * dy - a_2s\theta_2 * dx))}{(a_1 + a_2c\theta_2)^2 + (a_2s\theta_2)^2}$$
$$c\theta_1 = \frac{(a_1 + a_2c\theta_2) * dx + a_2s\theta_2 * dy))}{(a_1 + a_2c\theta_2)^2 + (a_2s\theta_2)^2}$$
$$\theta_1\_a = tan^{-1}(s\theta_1\_a/c\theta_1\_a)$$
$$\theta_1\_b = tan^{-1}(s\theta_2\_b/c\theta_1\_b)$$

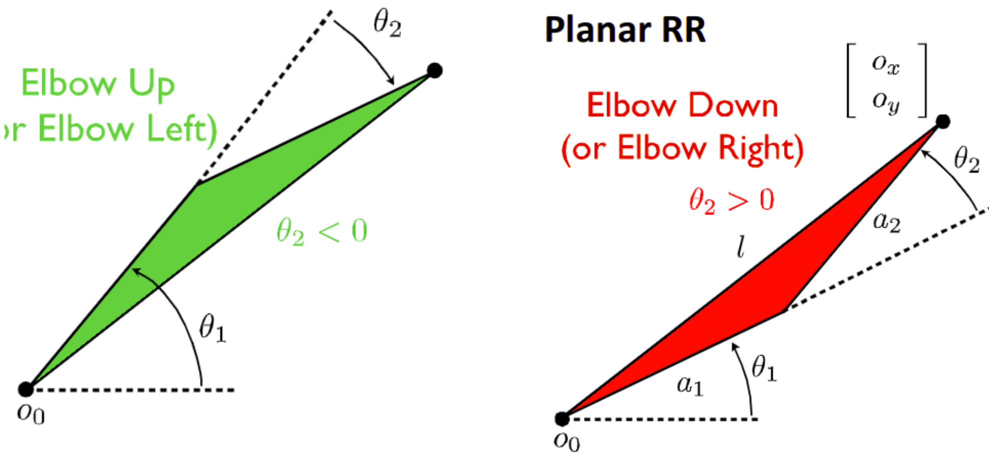
According to the calculated pair of theta_1 and theta_2 for both the elbow up and elbow down configuration of the robot, we get two corresponding values of theta_3 using this relationship.

$$\alpha = \theta_1 + \theta_2 + \theta_3$$

5.2 Evaluation

We tested the code of forward kinematics by taking configurations that are easy to calculate manually. For this, we considered the default configuration of the robot given in the handout. As the code also returned the position of all the joints, we were able to verify the forward kinematics code through geometric inspection as well.

For inverse kinematics, we checked the equations derived above analytically by seeing the effect of different values of theta_2 on theta_1. When the robot is in the elbow down configuration (theta_2>0), the value of theta_1 would be greater as compared to when the robot is in the elbow up configuration (theta_2<0). This can also be verified geometrically from the figure below.

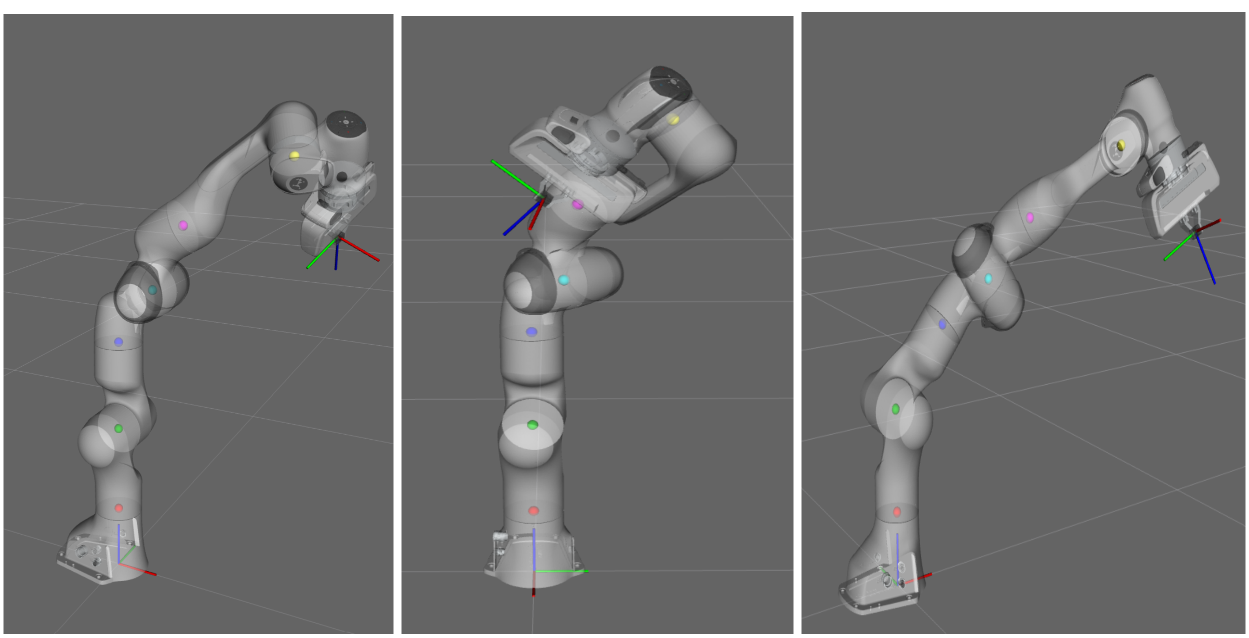


Three new configurations for verifying forward kinematics:

[0, 0, 0, -pi/3, pi/4, pi/3, pi/2]

[pi/2, 0, -pi/4, -pi/4, -pi/2, 0, 0]

[0,   pi/4, -pi/2, -pi/4, pi/3, pi/2,   pi/2]

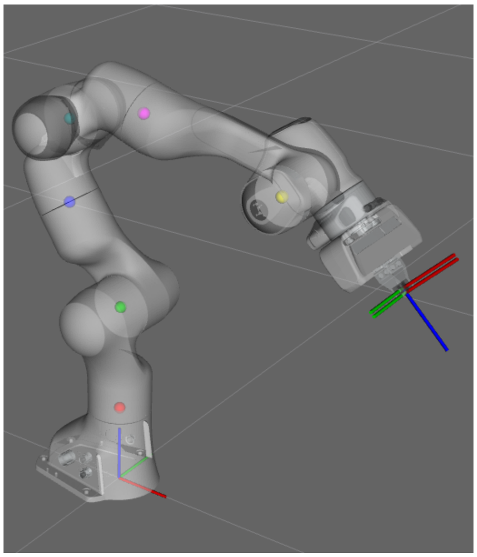
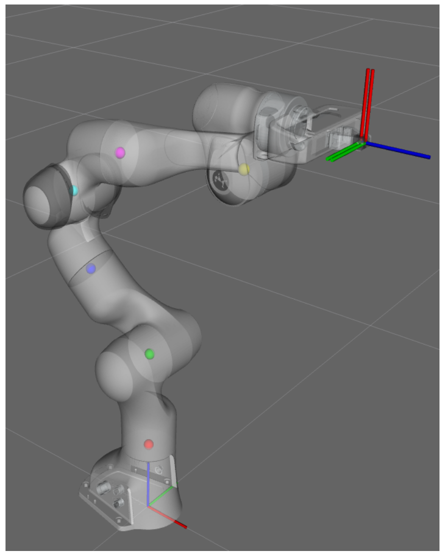
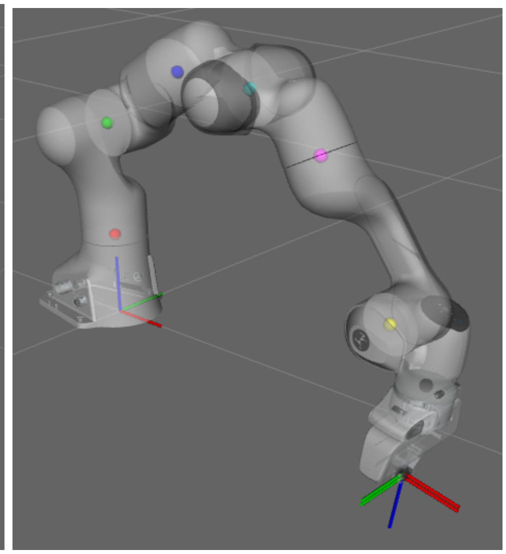


Three new targets for verifying the Planar inverse kinematic of the robot:

x = 0.5, y= 0.5, theta = pi/4

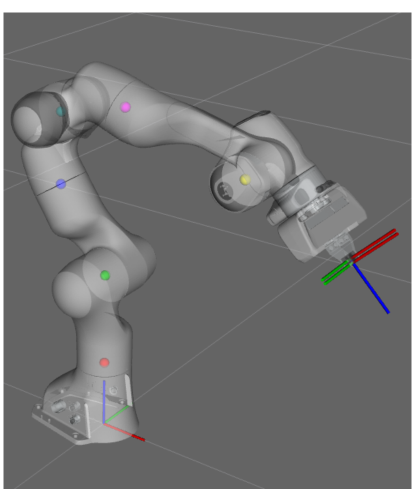
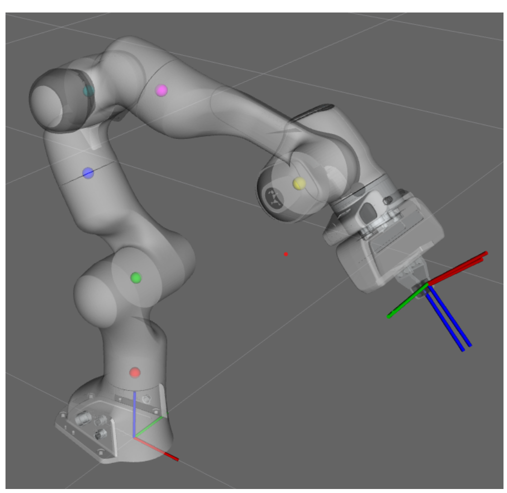
x = 0.4, y= 0.8, theta = 0

x = 0.6, y= -0.1, theta = pi/2+0.2

5.3.1 Effect of gravity

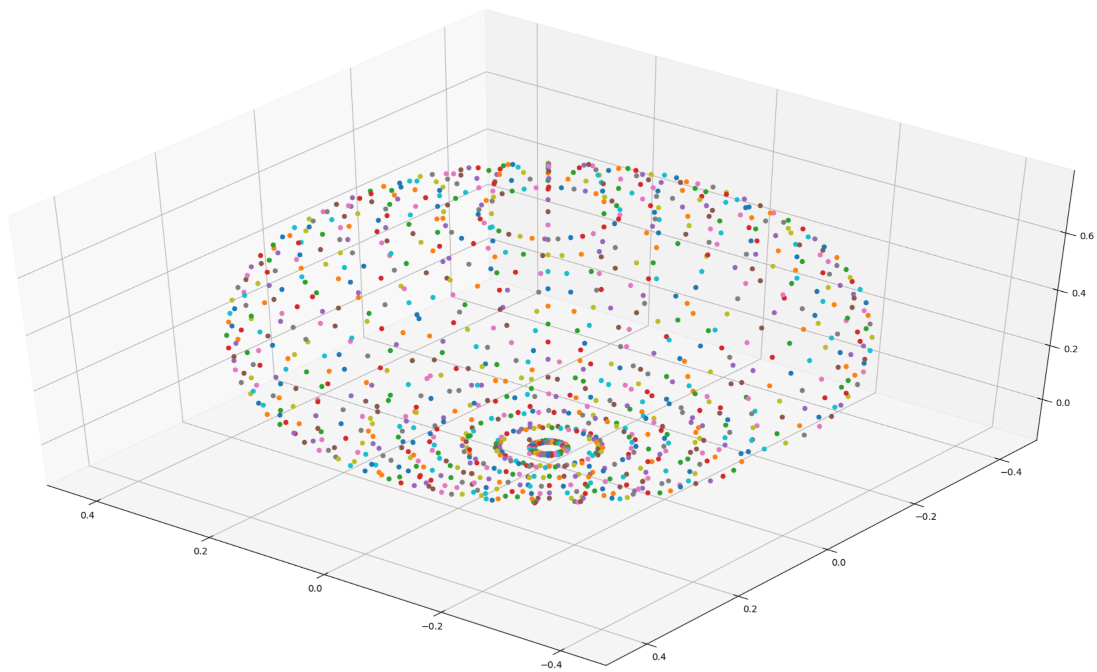
When the gravity of the gazebo is set to 9.8 m/s^2, the effect of mass of the link and the torque generated by it about its joint position also have to be taken into consideration. This results in an offset due to gravity and the robot undershoots the given goal position by a small amount. This is seen in both the inverse and forward kinematics.

5.3.2 Reachable Workspace

Using the forward kinematics matrix, we calculate the end effector position by varying the joint limits to get the reachable workspace of the robot. We calculate the outer region of the reachable workspace limits by stretching all the joints to its upper limit and varying the 1st and 2nd joint from its lower to its upper limit. For the inner limit of the reachable workspace we used the same technique but set all the joints except the 1st and 2nd joint to its lower limit. The resulting reachable workspace can be seen below.

Assumptions: We assume a continuous space for the joint limits of each of the joints in the robot. However, in reality, the joint positions are discretized. This means that, as per the resolution of the joints, the robot would not be able to reach every point of the continuous surface.



5.3.3 Extending inverse kinematics to 3D

1) No, the Panda arm does not have a spherical wrist. Even when the Panda arm end effector has roll pitch and yaw joints, they don't intersect at a single point.
2) Kinematic decoupling helps to solve the forward kinematics problem in subsets. However, this won’t work on a full Panda Robot as it does not have a spherical joint.
3) For any 7 DOF arm, calculation of inverse kinematics is more computationally expensive as compared to a 6 DOF arm. Also, for the Panda arm, as it does not have a spherical joint, the computation will be even more complicated.