Robotics for IEM
(Preparatory Tasks)

Alok Ranjan s3816494
Xavier Cremades s3649512

October 10, 2019

1 Forward Kinematics

1.1 a

Given for the configuration:

$\theta1 = 0°$,$\theta_2 = -90°$, $\theta_3 = -90°$, $\theta_4 = 0°$, $\theta_5 = 0°$
L1 = 17 cm, L2 = 17 cm, L3 = 7 cm, L4 = 4 cm, L5 = 4 cm, L6 = 9 cm

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	L_1	θ_1
2	90	0	0	$\theta_2 + 90$
3	0	L_2	0	$\theta_3 + 90$
4	90	L_4	$L_3 + L_5$	$\theta_4 + 90$
5	90	0	0	θ_5
6	-90	0	L_6	90

Table 1: DH Paramaters

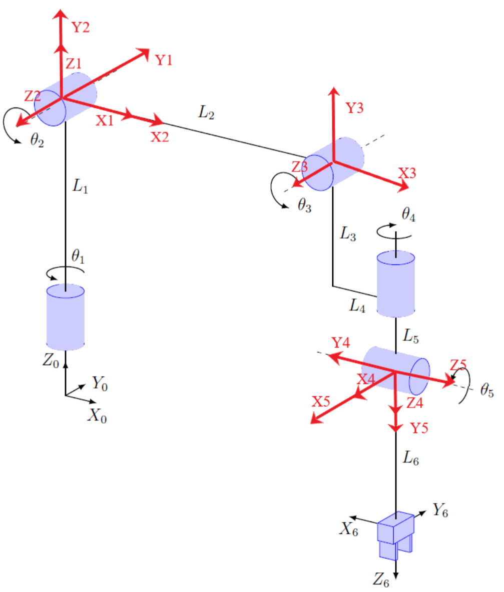

Figure 1: Axis Assignment where $\theta1 = 0°$,$\theta_2 = -90°$, $\theta_3 = -90°$, $\theta_4 = 0°$, $\theta_5 = 0°$

1.2 b

$$ {}^{0}_{1}T = \begin{pmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 17 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{1}_{2}T = \begin{pmatrix} -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ c_2 & -s_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{2}_{3}T = \begin{pmatrix} -s_3 & -c_3 & 0 & 17 \\ c_3 & -s_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{3}_{4}T = \begin{pmatrix} -s_4 & -c_4 & 0 & 4 \\ 0 & 0 & -1 & -11 \\ c_4 & -s_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{4}_{5}T = \begin{pmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{5}_{6}T = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 9 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$ {}^{0}_{6}T = {}^{0}_{1}T \; {}^{1}_{2}T \; {}^{2}_{3}T \; {}^{3}_{4}T \; {}^{4}_{5}T \; {}^{5}_{6}T $

${}^{0}P_6$ is the 4^{th} column of ${}^{0}_{6}T$, can be given by,

${}^{0}P_6 =$

$$ \begin{pmatrix} 4\,c_1\,s_2\,s_3 - 9\,c_5\,\left(c_1\,c_2\,s_3 + c_1\,c_3\,s_2\right) - 9\,s_5\,\left(c_4\,s_1 + s_4\,\left(c_1\,c_2\,c_3 - c_1\,s_2\,s_3\right)\right) - 4\,c_1\,c_2\,c_3 - 11\,c_1\,c_2\,s_3 - 11\,c_1\,c_3\,s_2 - 17\,c_1\,s_2 \\ 9\,s_5\,\left(c_1\,c_4 - s_4\,\left(c_2\,c_3\,s_1 - s_1\,s_2\,s_3\right)\right) - 9\,c_5\,\left(c_2\,s_1\,s_3 + c_3\,s_1\,s_2\right) - 17\,s_1\,s_2 - 4\,c_2\,c_3\,s_1 - 11\,c_2\,s_1\,s_3 - 11\,c_3\,s_1\,s_2 + 4\,s_1\,s_2\,s_3 \\ 17\,c_2 + 11\,c_2\,c_3 - 4\,c_2\,s_3 - 4\,c_3\,s_2 - 11\,s_2\,s_3 + 9\,c_5\,\left(c_2\,c_3 - s_2\,s_3\right) - 9\,s_4\,s_5\,\left(c_2\,s_3 + c_3\,s_2\right) + 17 \\ 1 \end{pmatrix} $$

2 Inverse Kinematics

In 1.2 (b) we have calculated ${}^{0}p_6$. To work on the inverse kinematics problem, it is assumed that $\theta_4 \equiv 0$ and $\theta_5 \equiv 0$. We have input these values in vector ${}^{0}p_6$. The reduced vector is shown in equation 1.

$$ {}^{0}p_6 = \begin{bmatrix} -9c_1 s_{2+3} - c_1(4c_{2+3} + 11s_{2+3} + 17s_2) \\ -9s_1 s_{2+3} - s_1(4c_{2+3} + 11s_{2+3} + 17s_2) \\ 9c_{2+3} + 17c_2 + 11c_{2+3} - 4s_{2+3} + 17 \end{bmatrix} $$

Further, simplifying equation 1 we have obtained equation 2.

$$ {}^{0}p_6 = \begin{bmatrix} -c_1(4c_{2+3} + 20s_{2+3} + 17s_2) \\ -s_1(4c_{2+3} + 20s_{2+3} + 17s_2) \\ 17c_2 + 20c_{2+3} - 4s_{2+3} + 17 \end{bmatrix} \tag{1} $$

Further, equating equation 2 to corresponding x, y, z

$$ \begin{bmatrix} -c_1(4c_{2+3} + 20s_{2+3} + 17s_2) \\ -s_1(4c_{2+3} + 20s_{2+3} + 17s_2) \\ 17c_2 + 20c_{2+3} - 4s_{2+3} + 17 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2} $$

Solve for; y/x of equation 2 will give us θ_1 as shown in equation 3.

$$x = -c_1(4c_{2+3} + 20s_{2+3} + 17s_2)$$

$$y = -s_1(4c_{2+3} + 20s_{2+3} + 17s_2)$$

$$tan(\theta_1) = y/x$$

Thus,

$$\theta_1 = Atan2(y, x) \tag{3}$$

As we are using Atan2 to calculate the inverse, it returns values in the closed interval of $[-\pi, \pi]$. So, to get the next solution for θ_1 we have to add π to right hand side of equation 3.

Now in order to obtain θ_3 we have used the "law of cosines". It states that if a, b and c are three sides of a triangle and θ is the angle contained between b and c, then

$$cos(\theta) = (b^2 + c^2 - a^2)/2bc$$

Here,

$$\theta = (\theta_3 + \alpha) - 180$$

∘

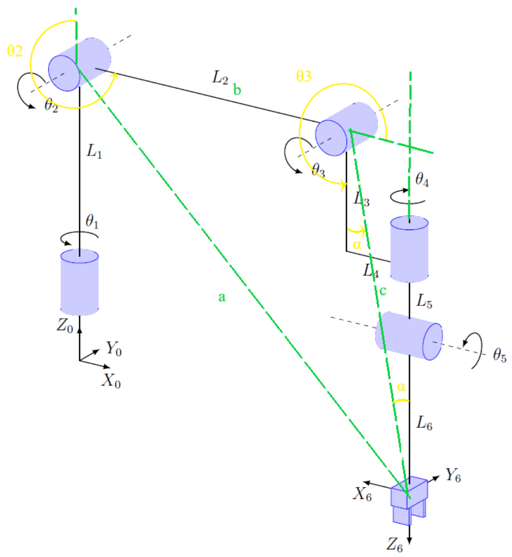

Figure 2: Angle representation in robotic arm

$$a^2 = x^2 + y^2 + (z - 17)^2$$

$$b = L_2$$

$$c^2 = L_4^2 + (L_3 + L_5 + L_6)^2$$

Substituting all the values of a, b, c and θ we get the following expression

$$cos(\theta_3 + \alpha) = \frac{[x^2 + y^2 + (z - 17)^2 - (L_2^2 + L_4^2 + (L_3 + L_5 + L_6)^2)]}{2L_2\sqrt{L_4^2 + (L_3 + L_5 + L_6)^2}} = U \tag{4}$$

The angle $(\theta_3 + \alpha)$ is contained between sides b and c is . We can evaluate α as follows:

$$tan(\alpha) = \frac{L_4}{L_3 + L_5 + L_6}$$

$$\alpha = Atan2(L_4, (L_3 + L_5 + L_6)) \tag{5}$$

Using equation 4 and 5, we derive the value of θ_3 which is given as follows:

$$\theta_3 = Atan2(\pm\sqrt{(1-U^2)}, U) - \alpha \tag{6}$$

We have two solutions here. One solution is for elbow up position and other is for elbow down position.

Further, we can obtain the following equality:

$$4c_2c_3 - 4s_2s_3 + 20s_2c_3 + 20c_2s_3 + 17s_2 = (4c_3 + 20s_3)c_2 + (20c_3 - 4s_3 + 17)s_2 = \pm\sqrt{x^2 + y^2} \tag{7}$$

We also know that the second joint rotates “counter-clockwise” and as we will be working with the negative angles so we will be choosing the “negative solution” here. From the third equation (equation of z) we can obtain

$$20c_2c_3 - 20s_2s_3 - 4s_2c_3 - 4c_2s_3 + 17c_2 = (20c_3 - 4s_3 + 17)c_2 - (4c_3 + 20s_3)s_2 = z - 17 \tag{8}$$

We can further simplify these equations as follows:

$$b_1 = k_1c_2 + k_2s_2$$

$$b2 = k_2c_2 - k_1s_2$$

Where we can define k_1, k_2, b_1 and b_2 as follows:

$$k_1 = 4c_3 + 20s_3$$

$$k_2 = 20c_3 - 4s_3 + 17$$

$$b_1 = -\sqrt{x^2 + y^2}$$

$$b_2 = z - 17$$

We represent these equations in matrix form:

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} k_1 & k_2 \\ k_2 & -k_1 \end{bmatrix} \begin{bmatrix} c_2 \\ s_2 \end{bmatrix}$$

$$\begin{bmatrix} c_2 \\ s_2 \end{bmatrix} = \begin{bmatrix} k_1 & k_2 \\ k_2 & -k_1 \end{bmatrix}^{-1} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{9}$$

From equation 9, we derive the expression for θ_2:

$$\theta_2 = Atan2(\frac{s_2}{c_2}) \tag{10}$$

From equations 3, 6 and 10, we have derived values for θ_1, θ_3 and θ_2, respectively.

3 Jacobians

3.1 a

To derive the basic Jacobians we have computed the derivative of the last column of ${}^0_6T$

For linear velocities following matlab function was used, where the X stands for ${}^0_6T$:

$$\begin{bmatrix} J1 = [simplify(diff(X(1,4),t1)); simplify(diff(X(2,4),t1)); simplify(diff(X(3,4),t1))]; \\ J2 = [simplify(diff(X(1,4),t2)); simplify(diff(X(2,4),t2)); simplify(diff(X(3,4),t2))]; \\ J3 = [simplify(diff(X(1,4),t3)); simplify(diff(X(2,4),t3)); simplify(diff(X(3,4),t3))]; \\ J4 = [simplify(diff(X(1,4),t4)); simplify(diff(X(2,4),t4)); simplify(diff(X(3,4),t4))]; \\ J5 = [simplify(diff(X(1,4),t5)); simplify(diff(X(2,4),t5)); simplify(diff(X(3,4),t5))]; \end{bmatrix}$$

Here,

$$J_1 = \begin{bmatrix} -9c_1c_4s_5 + 9s_1c_{2+3}s_4s_5 + 9s_1s_{2+3}c_5 + s1(4c_{2+3} + 11s_{2+3} + 17s_2) \\ -9s_1c_4s_5 - 9c_1c_{2+3}s_4s_5 - 9c_1s_{2+3}c_5 - c1(4c_{2+3} + 11s_{2+3} + 17s_2) \\ 0 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} 9c_1s_{2+3}s_4s_5 - 9c_1c_{2+3}c_5 - c_1(-4s_{2+3} + 11c_{2+3} + 17c_2) \\ 9s_1s_{2+3}s_4s_5 - 9s_1c_{2+3}c_5 - s_1(-4s_{2+3} + 11c_{2+3} + 17c_2) \\ -9c_{2+3}s_4s_5 - 9s_{2+3}c_5 - 17s_2 - 11s_{2+3} - 4c_{2+3} \end{bmatrix}$$

$$J_3 = \begin{bmatrix} 9c_1s_{2+3}s_4s_5 - 9c_1c_{2+3}c_5 - c_1(-4s_{2+3} + 11c_{2+3}) \\ 9s_1s_{2+3}s_4s_5 - 9s_1c_{2+3}c_5 - s_1(-4s_{2+3} + 11c_{2+3}) \\ -9c_{2+3}s_4s_5 - 9s_{2+3}c_5 - 11s_{2+3} - 4c_{2+3} \end{bmatrix}$$

$$\mathrm{J}_4 = \begin{bmatrix} 9s_1s_4c_5 - 9c_1c_{2+3}c_4s_5) \\ -9c_1c_4c_5 - 9s_1c_{2+3}c_4s_5) \\ -9c_{2+3}s_5) \end{bmatrix}$$

$$\mathrm{J}_5 = \begin{bmatrix} -9s_1c_4c_5 - 9c_1c_{2+3}s_4c_5 + 9c_1s_{2+3}s_5) \\ 9c_1c_4c_5 - 9s_1c_{2+3}s_4c_5 + 9s_1s_{2+3}s_5) \\ -9s_{2+3}s_4c_5 - 9c_{2+3}s_5) \end{bmatrix}$$

For the angular velocities following matlab script was used:

$$\begin{bmatrix} Z01 = [T01(1,3); T01(2,3); T01(3,3)]; \\ Z02 = [T02(1,3); T02(2,3); T02(3,3)]; \\ Z03 = [T03(1,3); T03(2,3); T03(3,3)]; \\ Z04 = [T04(1,3); T04(2,3); T04(3,3)]; \\ Z05 = [T05(1,3); T05(2,3); T05(3,3)]; \\ Z = [Z01, Z02, Z03, Z04, Z05]; \\ \omega\text{=simplify(Z);} \end{bmatrix}$$

Writing ω of end effector in angular velocity form:

$$\omega_x = s_1\dot{\theta_2} + s_1\dot{\theta_3} - c_1s_{2+3}\dot{\theta_4} - (c_1c_{2+3}c_4 - s_1s_4)\dot{\theta_5}$$

$$\omega_y = -c_1\dot{\theta_2} - c_1\dot{\theta_3} - s_1s_{2+3}\dot{\theta_4} - (s_1c_{2+3}c_4 - c_1s_4)\dot{\theta_5}$$

$$\omega_z = \dot{\theta_1} + c_{2+3}\dot{\theta_4} - s_{2+3}c_4\dot{\theta_5}$$

3.2 b

There are two types of singularities:

1) Workspace-boundary singularities occur when the manipulator is fully stretched out or folded back on itself in such a way that the end-effector is at or very near the boundary of the workspace.

With our robotic arm, there are two workspace-boundary singularities:

• Firstly, When $\theta_1 = \theta_2 = \theta_3 = 0°$, the robotic arm is fully stretched out (figure1), the singularity is present at the outer boundary of its workspace because we can not reach any point after that extent. If we rotate θ_1 and θ_2 full circle, The outer boundary can be viewed as a sphere of radius (L2+L3+L5+L5) = 38 cm around joint 2.

• Additionally, When $\theta_1 = 0°$, $\theta_2 = \pm 180°$, $\theta_3 = 0°$ and $\theta_5 = \pm 180°$ the robotic arm is folded back and the singularity is present at the inner boundary of its workspace because we can not reach any point near to the base of that extent. The inner boundary can be viewed as a sphere around joint 1 with a radius of $2\sqrt{5}$ cm (figure2).

2) Workspace-interior singularities occur away from the workspace boundary; they generally are caused by a lining up of two or more joint axes. With our robotic arm we do not have any case where two or more axes line up, therefore there is no workspace-interior singularity.

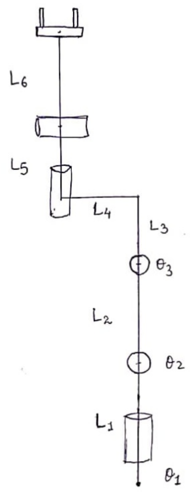

Figure 3: Zero configuration of the arm, singularity at outer boundary

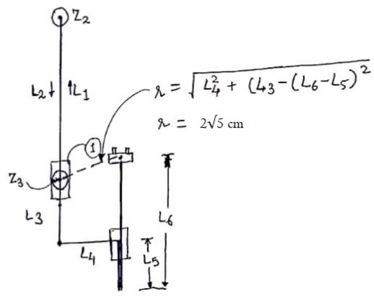

Figure 4: Singularity at inner boundary

4 Coding Tasks

Questions 1 b) and 3 a) are solved in the script called Script_ 1b_and_3a. Numerical values of translational vector ($^0P_6$) and Jacobians were calculated in this script and were matched with the values of the script arm_creation_val. Forward kinematics and inverse kinematics are implemented as functions [p]=fk(q) and [q]=ik(p), respectively.

The answers of 4 d) and 4 e) are in the script called traj_planning. In 4 e) while planning the trajectory for Cartesian space using 'mask' we have to figure out and define the mask vector. As our robotic arm has only 5 degrees of freedoms and in the 3D space there are 6 degree of freedoms possible, we will always get a sub optimal solution. To get that, in 'mask' we have to figure out which values of that vector should be 0 or 1.

5 Lab-session related Assignment

The Lab-session related Assignment is done in lab_assigments.m script.