Homework-5

ENPM662: Introduction to Robot Modelling

SAI DINESH GELAM

UID: 120167140

1.1) D-H Table without end effector pen

links	a_i	α_i	d_i	θ_i
0-1	0	90	128	θ_0+180
1-2	-612.7	180	0	θ_1-90
2-3	-571.6	180	0	θ_2
3-4	0	-90	163.9	θ_3+90
4-5	0	90	115.7	θ_4
5-E	0	0	92.2	θ_5

1.2) D-H Table with end effector pen

links	a_i	α_i	d_i	θ_i
0-1	0	90	128	θ_0+180
1-2	-612.7	180	0	θ_1-90
2-3	-571.6	180	0	θ_2
3-4	0	-90	163.9	θ_3+90
4-5	0	90	115.7	θ_4
5-E	0	0	192.2	θ_5

⊙ → Out of the plane
⊗ → into the plane

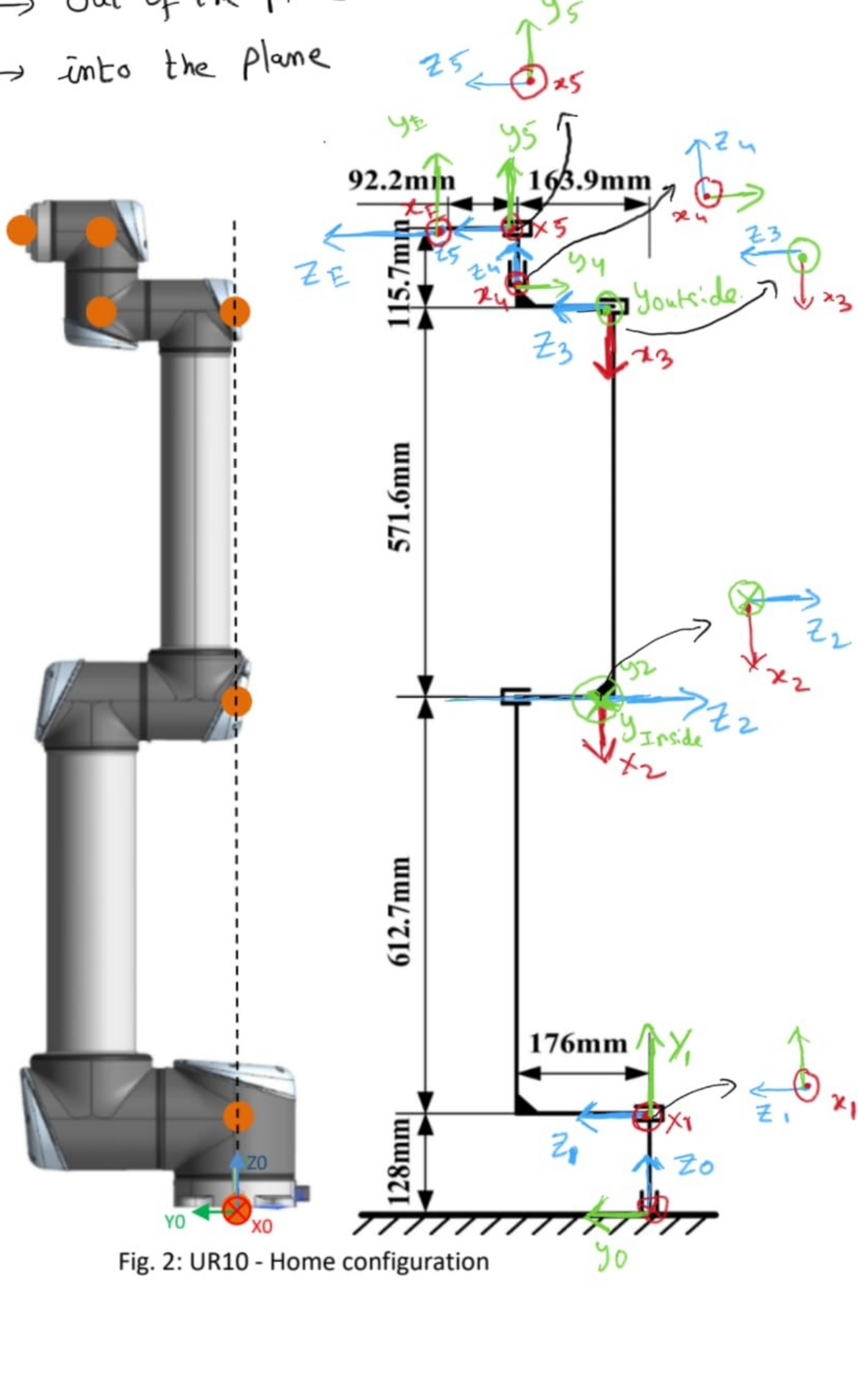


Fig. 2: UR10 - Home configuration

Part 1: To find Jacobian matrix I followed method one :

$$J_v = [J_{v_1} .....J_{v_n}]$$
$$J_{v_i} = R_{i-1} \times (o_n - o_{i-1})$$
$$J_w = [J_{w_1} ....J_{w_n}]$$
$$J_{w_i} = R_{i-1}$$

$$J_i \quad = \quad \begin{bmatrix} z_{i-1} \times (o_n - o_{i-1}) \\ z_{i-1} \end{bmatrix}$$

$$J = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix}$$

So the following will be the J matrix

$$J = \begin{bmatrix} R_0^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_0^0) & R_1^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_1^0) & R_2^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_2^0) & R_3^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_3^0) & R_4^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_4^0) & R_5^0 \begin{bmatrix}0\\0\\1\end{bmatrix} (o_6^0 - o_5^0) \\ R_0^0 \begin{bmatrix}0\\0\\1\end{bmatrix} & R_1^0 \begin{bmatrix}0\\0\\1\end{bmatrix} & R_2^0 \begin{bmatrix}0\\0\\1\end{bmatrix} & R_3^0 \begin{bmatrix}0\\0\\1\end{bmatrix} & R_4^0 \begin{bmatrix}0\\0\\1\end{bmatrix} & R_5^0 \begin{bmatrix}0\\0\\1\end{bmatrix} \end{bmatrix}$$

Part 2 : Defining velocity matrix

$$\xi = \begin{bmatrix} v_n^0 \\ \omega_n^0 \end{bmatrix}$$

Where in our case Vn will be vector of Vx, Vy, Vz and Wn will be Wx, Wy, Wz

Since our end effector is moving in a single plane of z-x the remaining all the terms of Vn and Wn will be zero except Vx and Vz.

So in our case we need to plot circle, by the parametric equations of circle we get the following equation

$$x = r.cos\theta, z = r.sin\theta$$
$$\dot{x} = r.cos\theta.\dot{\theta}, \dot{z} = r.sin\theta.\dot{\theta}$$

Where$d\theta$/dt = angular velocity which is equal to 2*π / 200 since it is required to complete the circle in 200 seconds

So using this method we get velocity matrix as

$$E = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

Now using inverse kinematics method

$$E = J \cdot \dot{q}$$

$$\dot{q} = J^{-1} \cdot E$$

2) Dynamics

In order to find the dynamics (torques required) for each and every joint we need to find the term lagrangian for each and every link, which is given by

L = K – P

Where,

K is kinetic energy of the link

P is the potential energy of the link

In our case since the robot is said to be quasi static acceleration, Velocity and angular velocities of all the links are taken to be zero

So L becomes

L= -P

Now after obtaining L we substitute L value in the following equation

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i$$ ----------- (1)

Where Qi is the general torque components on every joint

In our case, since we have a Reaction Force of 5N from the wall Qi can be divided as individual joint torques and Resisting torques produced Forces.

$$\tau + J^T(q)\,F$$

These resistance forces can be converted to torques by multiplying with the transpose of Jacobian matrix calculated in the part-1

Since the robot is quasi static in the robot dynamic equation the terms of angular acceleration and angular velocity becomes zero

So Robot dynamic equation is given by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau + J^T(q)\,F$$

Here Mass and Coriolis terms will be multiplied by Zeros and gravity matrix will only be remained, So hence the final equation becomes

$$g(q) = \tau + J^T(q)\,F$$

So we will define G matrix by using equation (1) using Lagrangian term

```
PE = g*((m1*z1/2)+(m2*(z1+((z2-z1)/2)))+(m3*(z2+((z3-z2)/2)))+(m4*(z3+((z4-z3)/2)))+(m5*(z4+((z5-z4)/2)))+(m6*(z5+((z6-z5)/2))))
G =  Matrix([[diff(PE, th0)],
      [diff(PE, th1)],
      [diff(PE, th2)],
      [diff(PE, th3)],
      [diff(PE, th4)],
      [diff(PE, th5)]
      ])
```

(G matrix will be printed in the code)

After computing G matrix we get torque matrix by substituting the F matrix

```
F = Matrix([[0],
            [-5],
            [0],
            [0],
            [0],
            [0]])
```

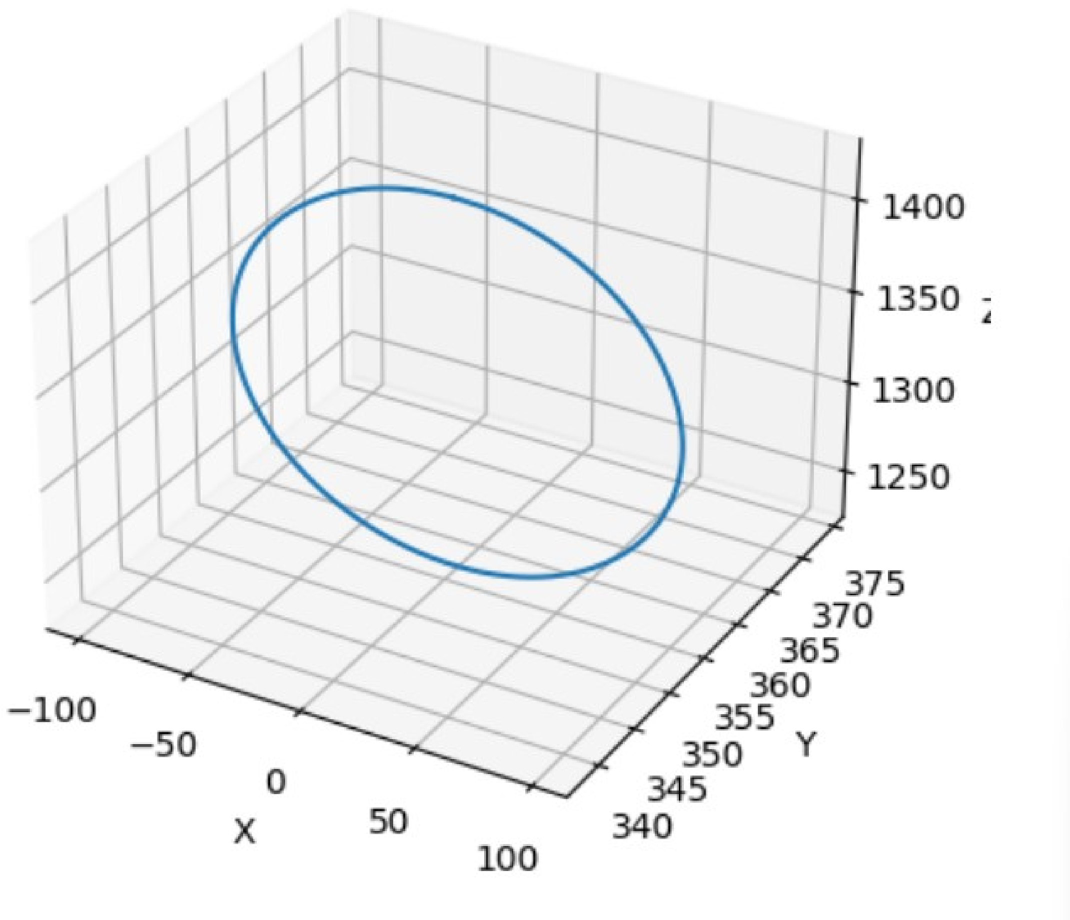
(since the reaction force is in only negative Y direction)

Torques at various joints is given by.

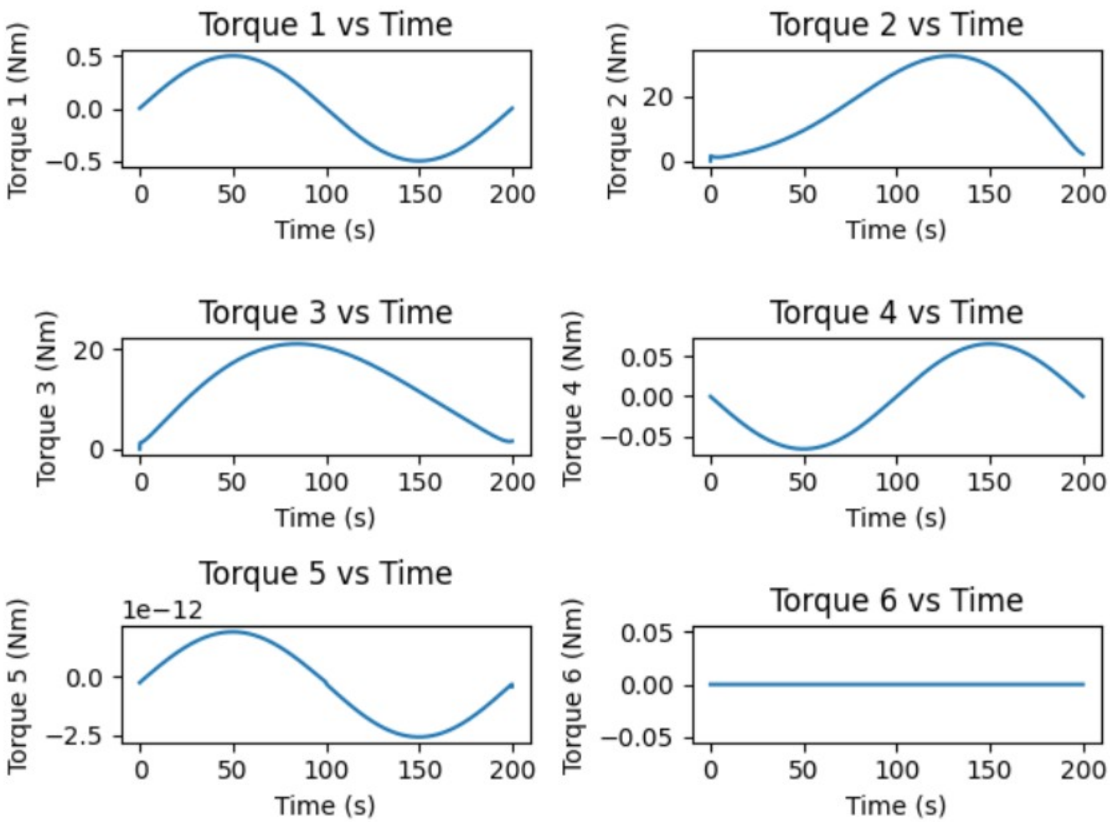
```
tou = G1 - (J_transpose*F)
```

Hence all the torques values are appended in the graphs

Results: Implementing the above two methods in the code we get the following final plot of circle and plots of torques at different joint required to apply 5N at the end effector



3-D trajectory



Here is the code file in google collab

https://colab.research.google.com/drive/1wsFLVDxksSShVl3Hd-T0rLgHqegqtwNo?usp=sharing

Reference: For masses the following documentation is refered

https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/