ME-452 Robotics
Complex Engineering Problem
Final Report

Taha Mahmood	**2017472**
Sheikh Abdul Majid	**2017437**

Submitted to: Dr Abid Imran

Abstract

In this paper, a 3 DOF robotic arm with RPR joint configuration will be analyzed and simulated on the MATLAB software. The forward kinematic model will predicate the Denavit-Hartenburg (DH) parametric for the robotic arm position. The inverse kinematic will also be performed by analyzing the value of the joint parameter based on the desired position and orientation of the end-effector. The MATLAB software is utilized to define the transformation matrix, simulating the robotic manipulator for different parameters. Moreover, the dynamic analysis of the robot will also be performed to verify its structural strength and reliability.

CONTENTS

List Of Figures

List Of Tables

1 INTRODUCTION

A robotic manipulator is a mechanical arm, where each link engages with other links to create a motion. A joint is used to ensure a proper connection with one link to another. The links are then connected to form a kinematic chain. At the end of the kinematic chain, an end-effector consists of a tool performing a proper function.

1.1 HISTORY

After World War 2, the world shifted towards the automation of various industrial processes. As a result, many researchers came up with the idea of designing a mechanism that can perform monotonous tasks with better accuracy. Hence, scientists started researching different the concept of an automated robotic arm in the late 1960s. With the advancement in technology, the popularity of the robotic arm increased.

One of the primary reasons for the rise in popularity of robotic arm Is because of lower cost. The prices of the parts required to construct a manipulator have reduced significantly. Furthermore, the motors used to control the joints also become more efficient in due time since they could produce a faster motion with more efficiency. The robots also could perform multiple tasks and perform the task which was difficult for the average human being. [1]

Nowadays, robotic arms can be found in almost all industries ranging from manufacturing automation to warehouse automation. The tasks performed by a robotic arm can be performed with more accuracy and efficiency. This has resulted in increased productivity for the particular process and an increase in product quality. However, a significant amount of the human workforce has been replaced, resulting in the rise of unemployment.

1.2 DESIGN METHODOLOGY

In this paper, a 3 DOF RPR manipulator will be designed, operating in a planer space. First, to conduct the position analysis, a DH Table will be designed for the manipulator, and then forward kinematics will be conducted to derive the series of equations defining the position of the end-effector with respect to a coordinate axis. Later on, the inverse kinematics will be performed to define the relation of the joint parameter with the position of the end-effector. [2]

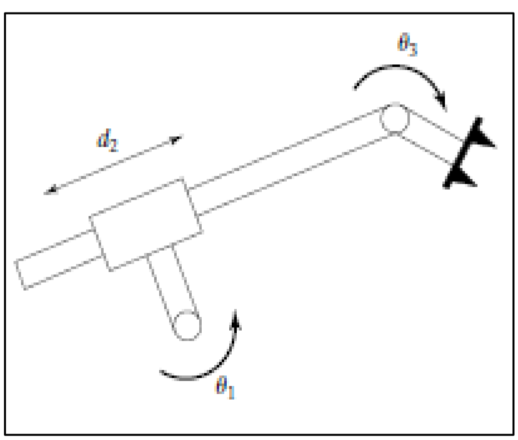

Figure 1: 3 DOF RPR manipulator

2 COMPLETE MATHEMATICAL MODEL

2.1 INTRODUCTION

In this paper, a 3 DOF RPR manipulator will be designed, operating in a planer space. First, to conduct the position analysis, a DH Table will be designed for the manipulator, and then forward kinematics will be conducted to derive the series of equations defining the position of the end-effector with respect to a coordinate axis. Later on, the inverse kinematics will be performed to define the relation of the joint parameter with the position of the end-effector.

First, the geometry of the robotic arm needs to be defined. To do so, we will construct the Denavit-Hartenberg (DH) table. With this, we will be able to define the joint parameters adapted in the manipulator. [3]

In mechanical engineering, the Denavit–Hartenberg parameters (also called DH parameters) are the parameters associated by attaching a reference frame to each link in the kinematic chain.

These parameters are:

a_i *is the distance from* Z_{i-1} *to* Z_i *measured along* X_i

α_i *is the angle from* Z_{i-1} *to* Z_i *measured about* X_i

d_i *is the distance from origin* X_{i-1} *to* X_i *measured along* Z_{i-1}

θ_i *is the distance from* X_{i-1} *to* X_i *measured about* Z_{i-1}

$\boldsymbol{i}$	$\boldsymbol{\alpha_{i-1}}$	$\boldsymbol{a_{i-1}}$	$\boldsymbol{d_i}$	$\boldsymbol{\theta_i}$
1	0	0	0	θ_1
2	$\pi/2$	l_1	d_2	0
3	$-\pi/2$	0	0	θ_3
4	0	l_3	0	0

Table 1: Denavit–Hartenberg parameters

It is evident from the figure above that the manipulator has 3 distinct links, and 4 coordinate frames, with the last one attached at the end-effector. Furthermore, there are two different joints. There is a revolute joint between the link 0 to 1, and between link 3 to 4, while there is one prismatic joint between link 2 to 3. Link 0 is associated with the reference frame, which is the base. While link 4 is associated with the end-effector. After defining the links, they have been modelled into a single manipulator.

After developing the DH table, the next step was to construct transformation matrices between each link. The transformation matrix ${}^{i-1}_{i}T$ represent the transformation from the (i-1) coordinate frame to the (i) coordinate frame. The formula for ${}^{i-1}_{i}T$ is defined as:

$$
{}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

The DH Table is then incorporated into the formula.

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

$$
{}^{1}_{2}T = \begin{bmatrix} 1 & 0 & 0 & l_1 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}
$$

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

$$
{}^{3}_{4}T = \begin{bmatrix} 1 & 0 & 0 & l_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}
$$

After multiplying each transformation matrix. The final transformation matrix is derived by multiplying each reference frame transformation matrix with each other.

$$
{}^{0}_{4}T = {}^{0}_{1}T \, {}^{1}_{2}T \, {}^{2}_{3}T \, {}^{3}_{4}T
$$

$$
{}^{0}_{4}T = \begin{bmatrix} c_1c_3 - s_1s_3 & -c_1s_3 - c_3s_1 & 0 & l_3(c_1c_3 - s_1s_3) + l_1c_1 + d_2s_1 \\ c_1s_3 + c_3s_1 & c_1c_3 - s_1s_3 & 0 & l_3(c_1s_3 + c_3s_1) + l_1s_1 - d_2c_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

where

$c_2 = \cos(\theta_2)$, $s_2 = \sin(\theta_2)$, $s_{23} = \sin(\theta_2 + \theta_3)$, $c_{23} = \cos(\theta_2 + \theta_3)$

2.2 FORWARD KINEMATICS

Forward Kinematic refers to computing the position of the end-effector using each joint's value. With this, we develop analytical equations which will relate the position of the end-effector with the joint parameters. To determine this equation, we will be using the DH table to correlate the x and y position of the end-effector.

$$ {}^{0}_{n}T = \begin{bmatrix} . & . & . & p_x \\ . & . & . & p_y \\ . & . & . & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Here, p_x, p_y, p_z define the position of the end-effector in the x, y, and z-axis respectively.

$$ p_x = l_3(c_1c_3 - s_1s_3) + l_1c_1 + d_2s_1 \tag{5} $$

$$ p_y = l_3(c_1s_3 + c_3s_1) + l_1s_1 - d_2c_1 \tag{6} $$

$$ p_z = 0 \tag{7} $$

These are the equations that define the position of the end-effector with respect to the joint's parameters. Here, d_2 is the position of the prismatic joint, while θ_1 and θ_3 are the values of the two revolute joints. Now, to find the translational vector of the transformation matrix, the joint parameters have been defined in terms of time. The equations of joint variables have been summarized below:

θ_1	d_2	θ_3
$(\pi/3)t$	$3t$	$(\pi/3)t$

Table 2: Joint Variable Equations

According to the equations shown in the table, the values of all the joint parameters are zero at the home position, as shown in Fig

2.3 INVERSE KINEMATICS

Inverse kinematics refers to finding the values of the joint parameters using the end-effector's position. It is the opposite of the forward kinematics. To calculate the joint parameters, we need to define a series of analytical equations.

Since the robotic manipulator is a planar robot, the manipulator's position will be defined in the x and y-axis. However, three joint parameters are to be evaluated. To account for the redundancies, another variable, phi φ, will be introduced. The equation that defines phi will be as follows: -

$$ \varphi = \theta_1 + \theta_3 $$

Initially, in the forward kinematics, the position of the end-effector with respect to the x and y-axis was defined as follows: -

$$ l_3 \cos(\theta_1 + \theta_3) + l_1 \cos\theta_1 + d_2 \sin\theta_1 = x $$

$$ l_3 \sin(\theta_1 + \theta_3) + l_1 \sin\theta_1 + d_2 \cos\theta_1 = y $$

Performing inverse kinematics for each joint

Joint 2: d_2

$$l_3 \cos(\theta_1 + \theta_3) + l_1 \cos\theta_1 + d_2 \sin\theta_1 = x$$

$$l_3 \cos(\varphi) + l_1 \cos\theta_1 + d_2 \sin\theta_1 = x$$

$$l_1 \cos\theta_1 + d_2 \sin\theta_1 = x - l_3 \cos(\varphi) = x'$$

$$l_3 \sin(\theta_1 + \theta_3) + l_1 \sin\theta_1 + d_2 \cos\theta_1 = y$$

$$l_3 \sin(\varphi) + l_1 \sin\theta_1 + d_2 \cos\theta_1 = y$$

$$l_1 \sin\theta_1 + d_2 \cos\theta_1 = y - l_3 \sin(\varphi) = y'$$

Squaring x' and y' and summing them together

$$x'^2 + y'^2 = (l_1 \cos\theta_1 + d_2 \sin\theta_1)^2 + (l_1 \sin\theta_1 + d_2 \cos\theta_1)^2$$

$$x'^2 + y'^2 = l_1^2 + d_2^2$$

$$d_2 = \sqrt{x'^2 + y'^2 - l_1^2} \tag{8}$$

Joint 1: θ_1

$$l_1 \cos\theta_1 + d_2 \sin\theta_1 = x - l_3 \cos(\varphi) = x'$$

$$\theta_1 = 2\tan^{-1}\left(\frac{d_2 \pm \sqrt{d_2^2 - l_1^2 - x'^2}}{l_1 + x'}\right) \tag{9}$$

Joint 3: θ_3

$$\theta_3 = \varphi - \theta_1 \tag{10}$$

3 KINEMATICS RESULTS

3.1 FORWARD KINEMATICS

This is the MATLAB code that has been compiled to perform the forward kinematics of our manipulator

```
clc
clear all
format short
L1=0.8;L3=0.8;
q1=0;d2=2;q3=0;

q=q1;d=0;a=0;alpha=0; %for i=1
T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;d=d2;a=L1;alpha=pi/2; %for i=2
T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=q3;d=0;a=0;alpha=-pi/2; %for i=3
T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;a=L3;d=0;alpha=0; %for i=4
T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];

T02=T01*T12;
T03=T01*T12*T23;
T04=T01*T12*T23*T34

axis([-5 5 -5 5]);

Ax1 = [T01(1,4),T02(1,4)];
Ay1 = [T01(2,4),T02(2,4)];
Ax2 = [T02(1,4),T03(1,4)];
Ay2 = [T02(2,4),T03(2,4)];
Ax3 = [T03(1,4),T04(1,4)];
Ay3 = [T03(2,4),T04(2,4)];
Ax4 = [-.1,.1];
Ay4 =  [0,0];
p1  = line(Ax1,Ay1,'LineWidth',[3],'Color','k');
p2  = line(Ax2,Ay2,'LineWidth',[3],'Color','M');
p3  = line(Ax3,Ay3,'LineWidth',[3],'Color','R');
p4  = line(Ax4,Ay4,'LineWidth',[12],'Color','B');

drawnow
xlabel('x (m)')
ylabel('y (m)')

pause()
```

```
for t=0:.01:1
    q1=pi/3*t;
    d2=2+3*t;
    q3=pi/3*t;

    q=q1;d=0;a=0;alpha=0; %for i=1
    T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) 
-sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0 
0 1];
    q=0;d=d2;a=L1;alpha=pi/2; %for i=2
    T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) 
-sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0 
0 1];
    q=q3;d=0;a=0;alpha=- pi/2; %for i=3
    T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) 
-sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0 
0 1];
    q=0;a=L3;d=0;alpha=0; %for i=4
    T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) 
-sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0 
0 1];

    T02=T01*T12;
    T03=T01*T12*T23;
    T04=T01*T12*T23*T34

    Ax1 = [T01(1,4),T02(1,4)];
    Ay1 = [T01(2,4),T02(2,4)];
    Ax2 = [T02(1,4),T03(1,4)];
    Ay2 = [T02(2,4),T03(2,4)];
    Ax3 = [T03(1,4),T04(1,4)];
    Ay3 = [T03(2,4),T04(2,4)];
    Ax4 = [-.1,.1];
    Ay4 =  [0,0];

    set(p1,'X', Ax1, 'Y',Ay1)
    set(p2,'X', Ax2, 'Y',Ay2)
    set(p3,'X', Ax3, 'Y',Ay3)
    set(p4,'X', Ax4, 'Y',Ay4)
    set(p4,'X', Ax4, 'Y',Ay4)

    drawnow

    pause(.01)
end
```

Initially, the joint parameters were defined as

$$q_1 = 0\,rad$$

$$d_2 = 2m$$

$$q_3 = 0\,rad$$

After executing the code, the following transformation matrix was found

$$ {}^{0}_{4}T = \begin{bmatrix} 1 & 0 & 0 & 1.6 \\ 0 & 1 & 0 & -2.0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Here the position of the end-effector has been identified, where

$$ x = 1.6\,m $$

$$ y = -2.0\,m $$

And this is the plot that has been obtained

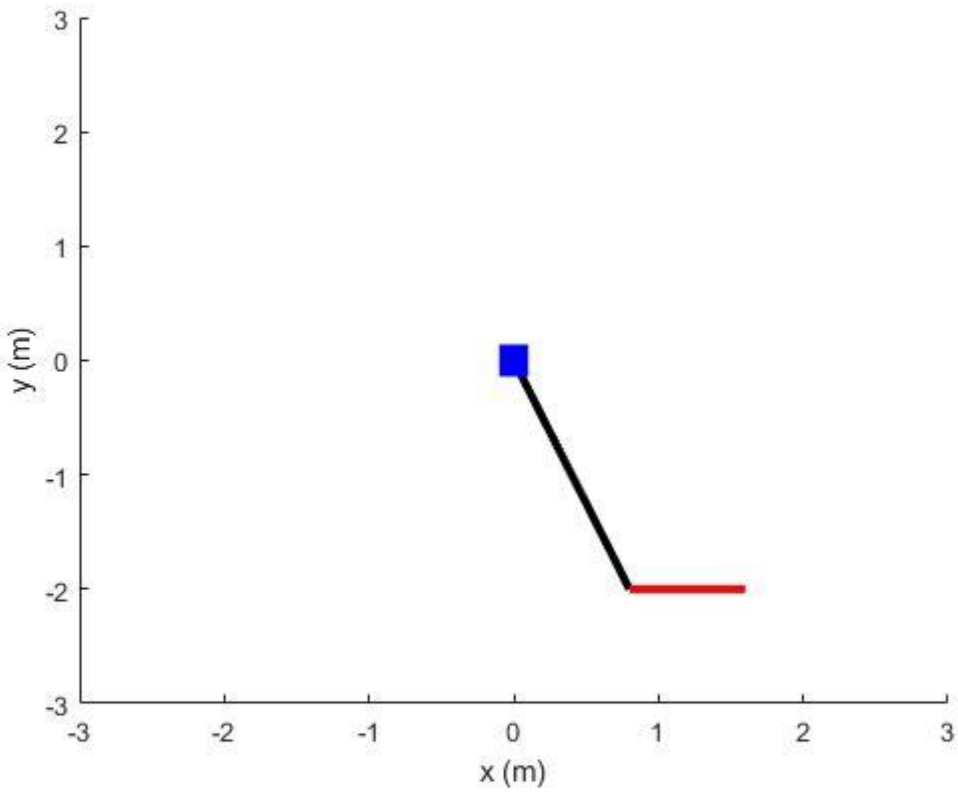

Figure 2: Forward Kinematic plot

Later the joint parameters were defined as a function of time

$$ q_1 = \frac{\pi}{3} t $$

$$ d_2 = 3t $$

$$ q_3 = \frac{\pi}{3} t $$

Where,

$$t \rightarrow 0 \; to \; 1 \; sec$$

This was the simulation result that was derived after executing the code.

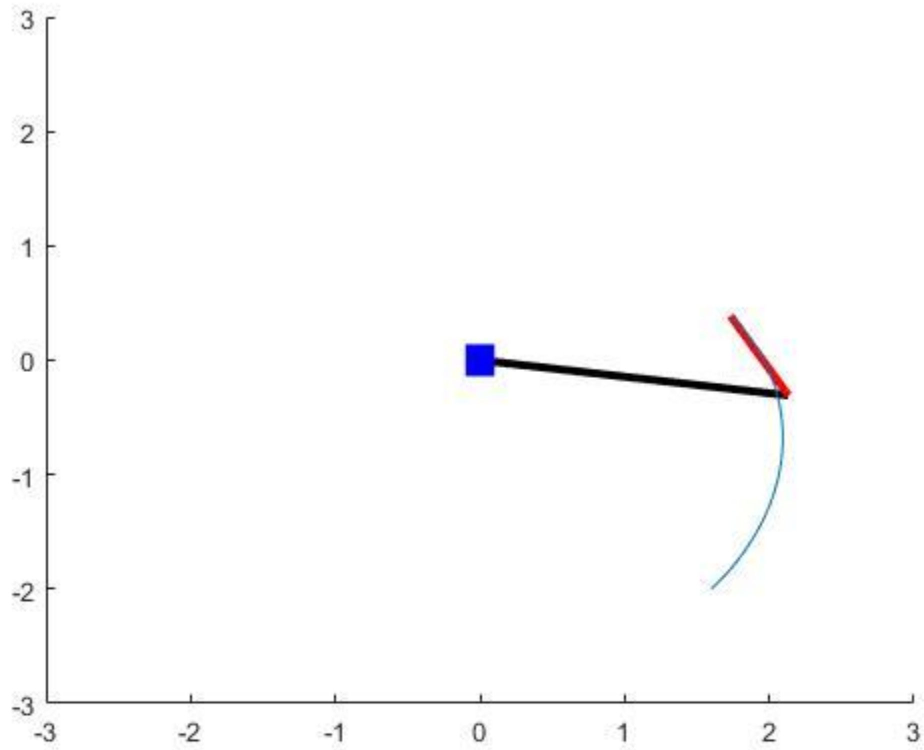

Figure 3: Forward Kinematics Simulation plot

3.2 Inverse Kinematics

After analyzing the inverse kinematics mathematical model, the following MATLAB code was executed.

```
clc;clear all;
format short

L1=0.8;L3=0.8;


x=1.5;
y=2.0;
PHIH=pi/2;
xi = x-L3*cos(PHIH);
yi = y-L3*sin(PHIH);
d2=sqrt(xi^2+yi^2-L1^2);
q1=2*atan2((d2+sqrt((d2^2)-(L1^2)-(xi^2))),(L1+xi));
q3=PHIH-q1;

q=q1;d=0;a=0;alpha=0; %for i=1
```

```
T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;d=d2;a=L1;alpha=pi/2; %for i=2
T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=q3;d=0;a=0;alpha=-pi/2; %for i=3
T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;a=L3;d=0;alpha=0; %for i=4
T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];

T02=T01*T12;
T03=T01*T12*T23;
T04=T01*T12*T23*T34
axis([-2 3 -2 3]);

  Ax1 = [T01(1,4),T02(1,4)];
  Ay1 = [T01(2,4),T02(2,4)];
  Ax2 = [T02(1,4),T03(1,4)];
  Ay2 = [T02(2,4),T03(2,4)];
  Ax3 = [T03(1,4),T04(1,4)];
  Ay3 = [T03(2,4),T04(2,4)];
  Ax4 = [-.1,.1];
  Ay4=  [0,0];
  p1  = line(Ax1,Ay1,'LineWidth',[3],'Color','k');
  p2  = line(Ax2,Ay2,'LineWidth',[3],'Color','M');
  p3  = line(Ax3,Ay3,'LineWidth',[3],'Color','R');
  p4  = line(Ax4,Ay4,'LineWidth',[12],'Color','B');

  drawnow

n=1;
r=.3;
  for t=0:.01:1

phi=t*2*pi;
x=1.5+r*cos(phi);
y=2.5+r*sin(phi);

PHIH=pi/2;
xi = x-L3*cos(PHIH);
yi = y-L3*sin(PHIH);
d2=sqrt(xi^2+yi^2-L1^2);
q1=2*atan2((d2+sqrt((d2^2)-(L1^2)-(xi^2))),(L1+xi));
q3=PHIH-q1;

Xp = L3*(cos(q1+q3))+L1*cos(q1)+d2*sin(q1);
Yp = L3*(sin(q1+q3))+L1*sin(q1)-d2*cos(q1);
q=q1;d=0;a=0;alpha=0; %for i=1
```

```
T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;d=d2;a=L1;alpha=pi/2; %for i=2
T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=q3;d=0;a=0;alpha=-pi/2; %for i=3
T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;a=L3;d=0;alpha=0; %for i=4
T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
T02=T01*T12;
T03=T01*T12*T23;
T04=T01*T12*T23*T34;

  Ax1 = [T01(1,4),T02(1,4)];
  Ay1 = [T01(2,4),T02(2,4)];
  Ax2 = [T02(1,4),T03(1,4)];
  Ay2 = [T02(2,4),T03(2,4)];
  Ax3 = [T03(1,4),T04(1,4)];
  Ay3 = [T03(2,4),T04(2,4)];
  Ax4 = [-.1,.1];
  Ay4=  [0,0];

  set(p1,'X', Ax1, 'Y',Ay1)
  set(p2,'X', Ax2, 'Y',Ay2)
  set(p3,'X', Ax3, 'Y',Ay3)
  set(p4,'X', Ax4, 'Y',Ay4)
  o1(n,1)=Xp;
  o2(n,1)=Yp;


  drawnow
  pause(.01)
  n=n+1;
  end
hold on
plot(o1(:,1),o2(:,1))
```

Initially, the end-effector position was defined as

$$x = 1.5\ m$$

$$y = 2.0\ m$$

After executing the following code, the joints parameters were evaluated as follows:

θ_1	d_2	θ_3
1.5017	1.7464	0.0691

Table 3: Joint Parameters Values

Similarly, the transformation matrix was also evaluated as following:

$$ {}_4^0T = \begin{bmatrix} 0 & -1 & 0 & 1.7975 \\ 1 & 0 & 0 & 1.4776 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

The plot of the entire manipulator is shown in the figure below:

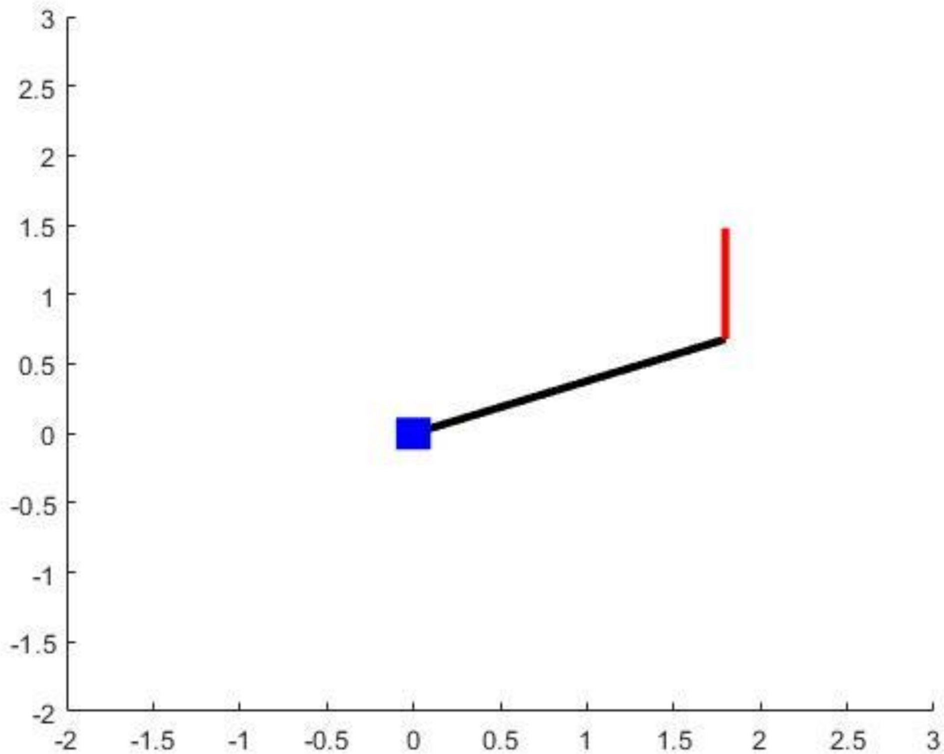

Figure 4: Inverse Kinematics Plot

Later, the trajectory of the end-effector position was defined as follows:

$$\varphi = 2\pi t$$

$$x = 1.5 + r\cos\varphi$$

$$y = 2.5 + r\sin\varphi$$

Where,

$$r = 0.3\ m$$

$$t \to 0\ to\ 1\ sec$$

This was the simulation result that was derived after executing the code.

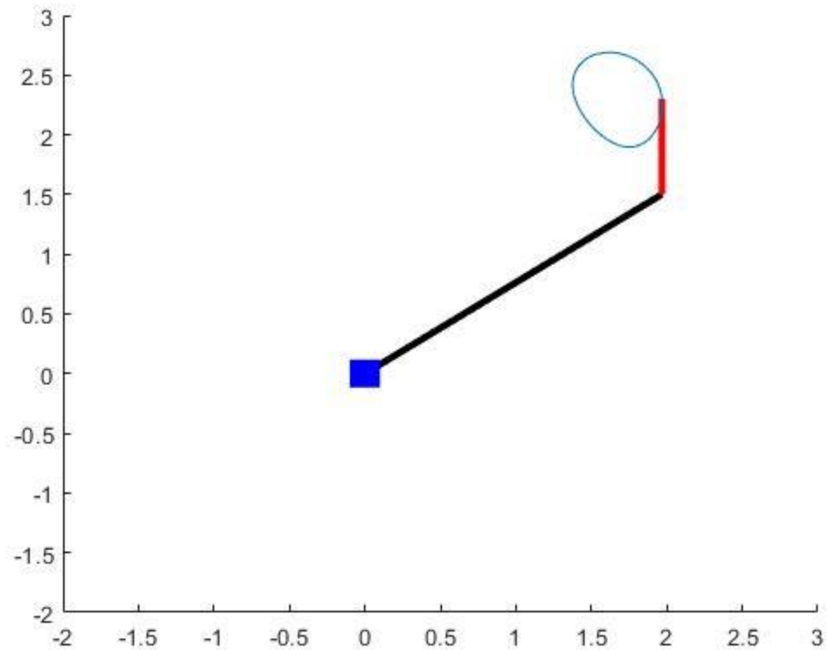

Figure 5: Inverse Kinematics Simulation plot

4 CLOSED-FORM DYNAMIC SOLUTION

For the dynamic analysis, the Lagrangian approach has been utilized. The method is based on energy conservation, and generation due to external torques applied. For this purpose, first the center of masses of the links have been defined, as shown in Fig 6. The center of mass of link 1 is at its center, while for the second link, it is assumed to be coinciding with the prismatic joint. Finally, for the third link, it is again assumed to be at the center of link 3.

After determining the center of masses, their coordinates with respect to the inertial frame, have been determined, which have then been further manipulated to derive the equations of kinetic energies, and potential energies for each of the links of the manipulator using the formula shown below,

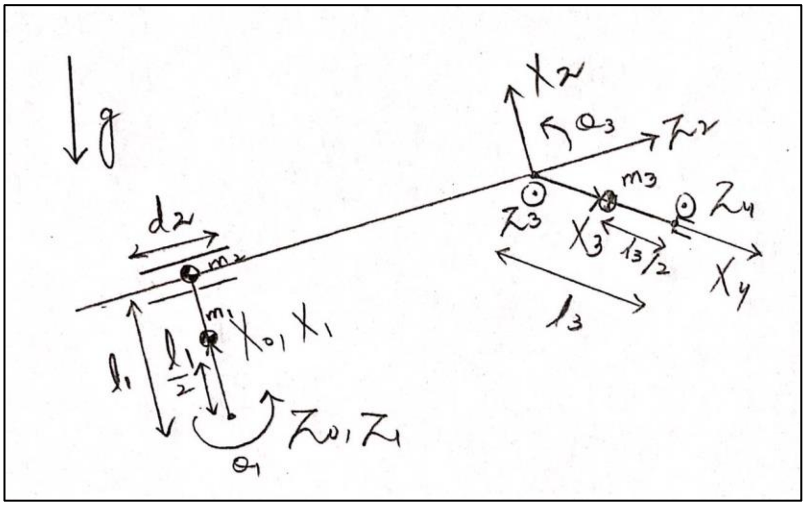

Figure 6: Coordinate Frames and Center of Masses

For the i^{th} link;

$$k_i = \frac{1}{2} m_i v_{C_i}^T v_{C_i} + \frac{1}{2}\ {}^{i}\omega_i^{T}\ {}^{C_i}I_i\ {}^{i}\omega_i \tag{11}$$

The total kinetic energy of the system is,

$$k = \sum_{i=1}^{n} k_i$$

Total kinetic energy for the manipulator is given by,

$$k(\Theta, \dot{\Theta}) = \frac{1}{2} \dot{\Theta}^T M(\Theta) \dot{\Theta} \tag{12}$$

The potential energy for the i^{th} link;

$$u_i = -m_i\ {}^{0}g^{T}\ {}^{0}P_{C_i} \tag{13}$$

Total potential energy of the system,

$$u = \sum_{i=1}^{n} u_i = u(\Theta)$$

The motion equation of the manipulator is given below, with L representing the Lagrange operator,

$$L(\Theta,\dot{\Theta}) = k(\Theta,\dot{\Theta}) - u(\Theta) \tag{14}$$

where the torque, τ, is given by,

$$\tau = \frac{d}{dt}\frac{\partial L}{\partial \dot{\Theta}} - \frac{\partial L}{\partial \Theta} = \frac{d}{dt}\frac{\partial k}{\partial \dot{\Theta}} - \frac{\partial k}{\partial \Theta} + \frac{\partial u}{\partial \Theta} \tag{15}$$

Now, calculating the position of center of masses for each link, using the transformation matrices developed in the forward kinematics section.

For Link 1,

The center of mass is given by,

$$x_{c1} = \frac{l_1 \cos\theta_1}{2}$$

$$y_{c1} = \frac{l_1 \sin\theta_1}{2}$$

The mass center velocities are worked out to be,

$$x_{c1} = \frac{l_1 c_1}{2} \rightarrow v_{xc1} = -\frac{l_1 s_1 \dot{\theta}_1}{2}$$

$$y_{c1} = \frac{l_1 s_1}{2} \rightarrow v_{yc1} = \frac{l_1 c_1 \dot{\theta}_1}{2}$$

$$v_{c1} = \begin{bmatrix} -\dfrac{l_1 s_1 \dot{\theta}_1}{2} \\ \dfrac{l_1 c_1 \dot{\theta}_1}{2} \end{bmatrix}$$

Now, calculating the dot product of velocity vector,

$$v_{c1}^T v_{c1} = \begin{bmatrix} -\dfrac{l_1 s_1 \dot{\theta}_1}{2} & \dfrac{l_1 c_1 \dot{\theta}_1}{2} \end{bmatrix} \begin{bmatrix} -\dfrac{l_1 s_1 \dot{\theta}_1}{2} \\ \dfrac{l_1 c_1 \dot{\theta}_1}{2} \end{bmatrix} = \left(\frac{l_1 \dot{\theta}_1}{2}\right)^2$$

The kinetic energy due to angular motion is given by,

$$\frac{1}{2}\ {}^{1}\omega_1\ {}^{c_1}I_1\ {}^{1}\omega_1 = \frac{1}{2}\begin{bmatrix} 0 & 0 & \dot{\theta}_1 \end{bmatrix} \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} = \frac{1}{2} I_{zz1} \dot{\theta}_1^{\ 2} = \frac{m_1 l_1^2}{24} \dot{\theta}_1^{\ 2}$$

$$\frac{1}{2}\ {}^{1}\omega_1{}^{c_1}I_1\ {}^{1}\omega_1 = \frac{m_1 l_1^2}{24}\dot{\theta}_1{}^2$$

The total kinetic energy of the first link is found by summing the translational and rotational kinetic energy components,

$$k_1 = \frac{1}{8}m_1 l_1^2\dot{\theta}_1{}^2 + \frac{1}{24}m_1 l_1^2\dot{\theta}_1{}^2 = \frac{1}{6}m_1 l_1^2\dot{\theta}_1{}^2 \qquad (16)$$

Finally, the potential energy of link 1 is found to be,

$$u_1 = \frac{m_1 g l_1 s_1}{2} \qquad (17)$$

For Link 2,

The center of mass is given by,

$$x_{c2} = l_1 \cos\theta_1 + d_2 \sin\theta_1$$

$$y_{c2} = l_1 \sin\theta_1 - d_2 \cos\theta_1$$

The mass center velocities are worked out to be,

$$v_{xc2} = -l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + d_2 s_1$$

$$v_{yc2} = l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - d_2 c_1$$

$$v_{c2} = \begin{bmatrix} -l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + d_2 s_1 \\ l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - d_2 c_1 \end{bmatrix}$$

Now, calculating the dot product of velocity vector,

$$v_{c2}^T v_{c2} = [-l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + d_2 s_1 \quad l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - d_2 c_1]\begin{bmatrix} -l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + d_2 s_1 \\ l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - d_2 c_1 \end{bmatrix}$$

$$v_{c2}^T v_{c2} = \left(l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + \dot{d}_2 s_1\right)^2 + \left(l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - \dot{d}_2 c_1\right)^2$$

$$v_{c2}^T v_{c2} = \left(l_1\dot{\theta}_1\right)^2 + \left(d_2\dot{\theta}_1\right)^2 + \left(\dot{d}_2\right)^2 + 2l_1 s_1^2\dot{\theta}_1\dot{d}_2 - 2l_1 c_1^2\dot{\theta}_1\dot{d}_2 + 4l_1 s_1 c_1 d_2\dot{\theta}_1{}^2$$

The kinetic energy due to angular motion is given by,

$$\frac{1}{2}\ {}^{2}\omega_2{}^{c_2}I_2\ {}^{2}\omega_2 = \frac{1}{2}[0 \quad 0 \quad \dot{0}]\begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$K.E = \frac{1}{2}m_2 v_{c2}^T v_{c2} = \frac{1}{2}m_2(\left(l_1 s_1\dot{\theta}_1 + d_2 c_1\dot{\theta}_1 + d_2 s_1\right)^2 + \left(l_1 c_1\dot{\theta}_1 + d_2 s_1\dot{\theta}_1 - d_2 c_1\right)^2)$$

The total kinetic energy of the second link is found by summing the translational and rotational kinetic energy components,

$$K.E = \frac{1}{2} m_2 \left\{ \left(l_1 \dot{\theta}_1\right)^2 + \left(d_2 \dot{\theta}_1\right)^2 + \left(\dot{d}_2\right)^2 + 2 l_1 s_1^2 \dot{\theta}_1 \dot{d}_2 - 2 l_1 c_1^2 \dot{\theta}_1 \dot{d}_2 + 4 l_1 s_1 c_1 d_2 \dot{\theta}_1{}^2 \right\} \quad (18)$$

Finally, the potential energy of link 2 is found to be,

$$u_2 = m_2 \begin{bmatrix} 0 \\ -y \\ 0 \end{bmatrix}^T \begin{bmatrix} l_1 c_1 + d_2 s_1 \\ l_1 s_1 - d_2 c_1 \\ 0 \end{bmatrix} = m_2 g (l_1 s_1 - d_2 c_1) \quad (19)$$

For Link 3,

The center of mass is given by,

$$x_{c3} = l_1 c_1 + d_2 s_1 + \frac{l_3 c_{13}}{2}$$

$$y_{c3} = l_1 s_1 - d_2 c_1 + \frac{l_3 s_{13}}{2}$$

The mass center velocities are worked out to be,

$$v_{cx3} = -l_1 s_1 \dot{\theta}_1 + \dot{d}_2 s_1 + d_2 c_1 \dot{\theta}_1 - \frac{l_3}{2} s_{13} \left(\dot{\theta}_1 + \dot{\theta}_3\right)$$

$$v_{cy3} = l_1 c_1 \dot{\theta}_1 - \dot{d}_2 c_1 + d_2 s_1 \dot{\theta}_1 + \frac{l_3 c_{13} \left(\dot{\theta}_1 + \dot{\theta}_3\right)}{2}$$

Now, calculating the dot product of velocity vector,

$$v_{c3}^T v_{c3} = \left(-l_1 s_1 \dot{\theta}_1 + \dot{d}_2 s_1 + d_2 c_1 \dot{\theta}_1 - \frac{l_3}{2} s_{13} \left(\dot{\theta}_1 + \dot{\theta}_3\right) \right)^2 + \left(l_1 c_1 \dot{\theta}_1 - \dot{d}_2 c_1 + d_2 s_1 \dot{\theta}_1 + \frac{l_3 c_{13} \left(\dot{\theta}_1 + \dot{\theta}_3\right)}{2} \right)^2$$

$$v_{c3}^T v_{c3} = \dot{d}_2{}^2 + \left(l_1 \dot{\theta}_1\right)^2 - 2 d_2 \dot{} l_1 \dot{\theta}_1 + \left(d_2 \dot{\theta}_1\right)^2 + \frac{l_3^2 \left(\dot{\theta}_1 + \dot{\theta}_3\right)^2}{4} - l_3 s_3 d_2 \dot{\theta}_1{}^2 - l_3 d_2 \dot{\theta}_1 \dot{\theta}_3 s_3$$

The kinetic energy due to angular motion is given by,

$$\frac{1}{2}\, {}^3\omega_3\, {}^{c_3}I_3\, {}^3\omega_3 = \frac{1}{2} \begin{bmatrix} 0 & 0 & \dot{0} \end{bmatrix} \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \left(\dot{\theta}_1 + \dot{\theta}_3\right) \end{bmatrix} = \frac{1}{24} m_3 l_3^2 \left(\dot{\theta}_1 + \dot{\theta}_3\right)^2$$

The total kinetic energy of the third link is found by summing the translational and rotational kinetic energy components,

$$\frac{1}{2} m_3 \left\{ \dot{d}_2{}^2 + \left(l_1 \dot{\theta}_1\right)^2 - 2 d_2 \dot{} l_1 \dot{\theta}_1 + \left(d_2 \dot{\theta}_1\right)^2 + \frac{l_3^2 \left(\dot{\theta}_1 + \dot{\theta}_3\right)^2}{3} - l_3 s_3 d_2 \dot{\theta}_1{}^2 - l_3 d_2 \dot{\theta}_1 \dot{\theta}_3 s_3 \right\} \quad (20)$$

Finally, the potential energy of link 3 is found to be,

$$u_3 = m_3 g(l_1 s_1 - d_2 c_1 + l_3 s_{13}) \tag{21}$$

Now, finding the derivative of total potential energy with respect to the joint variables of the manipulator,

$$u = u_1 + u_2 + u_3$$

$$u = \frac{m_1 g l_1 s_1}{2} + m_2 g l_1 s_1 - m_2 g d_2 c_1 + m_3 g l_1 s_1 - m_3 g d_2 c_1 + \frac{m_3 g l_3}{2}\sin(\theta_1 + \theta_3)$$

$$\frac{\partial u}{\partial \theta_1} = \frac{m_1 g l_1 c_1}{2} + m_2 g l_1 c_1 + m_2 g d_2 s_1 + m_3 g l_1 c_1 + m_3 g d_2 s_1 + \frac{m_3 g l_3}{2}\cos(\theta_1 + \theta_3) \tag{22}$$

$$\frac{\partial u}{\partial d_2} = m_2 g c_1 - m_3 g c_1 \tag{23}$$

$$\frac{\partial u}{\partial \theta_3} = \frac{m_3 g l_3}{2}\cos(\theta_1 + \theta_3) \tag{24}$$

Now, finding the derivative of total kinetic energy with respect to the joint variables of the manipulator,

$$k = k_1 + k_2 + k_3$$

$$k = \frac{1}{6} m_1 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_2 \left\{ l_1^2 \dot{\theta}_1^2 + d_2^2 \dot{\theta}_1^2 + \dot{d}_2^2 + 2 l_1 s_1^2 \dot{\theta}_1 d_2 + l_1 c_1^2 \dot{\theta}_1 d_2 + 4 l_1 s_1 c_1 d_2 \dot{\theta}_1^2 \right\} + \frac{1}{2} m_3 \left\{ \dot{d}_2^2 + l_1^2 \dot{\theta}_1^2 - 2 \dot{d}_2 l_1 \dot{\theta}_1 + d_2^2 \dot{\theta}_1^2 + \frac{l_3^2}{4}\left(\dot{\theta}_1 + \dot{\theta}_3\right)^2 - l_3 s_3 d_2 \dot{\theta}_1^2 - l_3 s_3 d_2 \dot{\theta}_1 \dot{\theta}_3 + \frac{1}{12} l_3^2 \left(\dot{\theta}_1 + \dot{\theta}_3\right)^2 \right\}$$

$$\frac{\partial k}{\partial \theta_1} = \frac{1}{2} m_2 \left\{ 4 l_1 s_1 \dot{\theta}_1 \dot{d}_2 c_1 + 4 l_1 c_1 \dot{\theta}_1 \dot{d}_2 s_1 + 4 l_1 c_1^2 \dot{\theta}_1^2 d_2 - 4 l_1 s_1^2 d_2 \dot{\theta}_1^2 \right\} \tag{25}$$

$$\frac{\partial k}{\partial d_2} = \frac{1}{2} m_2 \left\{ 2 d_2 \dot{\theta}_1 + 4 l_1 s_1 c_1 \dot{\theta}_1^2 \right\} + \frac{1}{2} m_3 \left\{ 2 d_2 \dot{\theta}_1^2 + l_3 s_3 \dot{\theta}_1^2 - l_3 s_3 \dot{\theta}_1 \dot{\theta}_3 \right\} \tag{26}$$

$$\frac{\partial k}{\partial \theta_3} = \frac{1}{2} m_3 \left\{ l_3 d_2 \dot{\theta}_1^2 c_3 + l_3 d_2 c_3 \dot{\theta}_1 \dot{\theta}_3 \right\} \tag{27}$$

Finally, finding the derivative of total kinetic energy with respect to the time rate of joint variables of the manipulator,

$$\frac{\partial k}{\partial \dot{\theta}_1} = \frac{1}{3} m_1 l_1^2 \dot{\theta}_1 + \frac{1}{2} m_2 \{2 l_1^2 \dot{\theta}_1 + 2 d_2^2 \dot{\theta}_1 + 2 l_1 s_1^2 \dot{d}_2 - 2 l_1 c_1^2 \dot{d}_2 + 8 l_1 s_1 c_1 d_2 \dot{\theta}_1\} + \frac{1}{2} m_3 \left\{2 l_1^2 \dot{\theta}_1 - 2 \dot{d}_2 l_1 + 2 d_2^2 \dot{\theta}_1 + \frac{2 l_3^2}{4} (\dot{\theta}_1 + \dot{\theta}_3) - 3 l_3 s_3 d_2 \dot{\theta}_1 + \frac{1}{6} l_3^2 (\dot{\theta}_1 + \dot{\theta}_3)\right\} \tag{28}$$

$$\frac{\partial k}{\partial \dot{d}_2} = \frac{1}{2} m_2 \{2 \dot{d}_2 + 2 l_1 s_1^2 \dot{\theta}_1 - 2 l_1 c_1^2 \dot{\theta}_1\} + \frac{1}{2} m_3 \{2 \dot{d}_2 - 2 l_1 \dot{\theta}_1\} \tag{29}$$

$$\frac{\partial k}{\partial \dot{\theta}_3} = \frac{1}{2} m_3 \left\{\frac{2 l_3^2}{4} (\dot{\theta}_1 + \dot{\theta}_3) - l_3 s_3 d_2 \dot{\theta}_1 + \frac{1}{6} l_3^2 (\dot{\theta}_1 + \dot{\theta}_3)\right\} \tag{30}$$

Taking the time derivatives of equation 28, 29, and 30,

$$\frac{d}{dt} \frac{\partial k}{\partial \dot{\theta}_1} = \frac{1}{3} m_1 l_1^2 \ddot{\theta}_1 + \frac{1}{2} m_2 \left\{2 l_1^2 \ddot{\theta}_1 + 2 d_2^2 \ddot{\theta}_1 + 4 d_2 \dot{\theta}_1 \dot{d}_2 + 8 l_1 s_1 c_1 \dot{d}_2 \dot{\theta}_1 + 2 l_1 s_1^2 \ddot{d}_2 - 2 l_1 c_1^2 \ddot{d}_2 + 8 l_1 c_1^2 d_2 \dot{\theta}_1^{\,2} - 8 l_1 s_1^2 d_2 \dot{\theta}_1^{\,2} + 8 l_1 s_1 c_1 \dot{d}_2 \dot{\theta}_1 + 8 l_1 s_1 c_1 d_2 \ddot{\theta}_1\right\} + \frac{1}{2} m_3 \left\{2 l_1^2 \ddot{\theta}_1 - 2 \ddot{d}_2 l_1 + 2 d_2^2 \ddot{\theta}_1 + 4 d_2 \dot{d}_2 \dot{\theta}_1 + \frac{2 l_3^2}{4} (\ddot{\theta}_1 + \ddot{\theta}_3) - 2 l_3 c_3 d_2 \dot{\theta}_1 \dot{\theta}_3 - 2 l_3 s_3 \dot{d}_2 \dot{\theta}_1 - 2 l_3 s_3 d_2 \ddot{\theta}_1 - l_3 c_3 d_2 \dot{\theta}_3^{\,2} - l_3 s_3 \dot{d}_2 \dot{\theta}_3 - l_3 s_3 d_2 \ddot{\theta}_3 + \frac{1}{6} l_3^2 (\ddot{\theta}_1 + \ddot{\theta}_3)\right\} \tag{31}$$

$$\frac{d}{dt} \frac{\partial k}{\partial \dot{d}_2} = \frac{1}{2} m_2 \left\{2 \ddot{d}_2 + 8 l_1 s_1 c_1 \dot{\theta}_1^{\,2} + 2 l_1 s_1^2 \ddot{\theta}_1 - 2 l_1 c_1^2 \ddot{\theta}_1\right\} + \frac{1}{2} m_3 \{2 \ddot{d}_2 - 2 l_1 \ddot{\theta}_1\} \tag{32}$$

$$\frac{d}{dt} \frac{\partial k}{\partial \dot{\theta}_3} = \frac{1}{2} m_3 \left\{\frac{2 l_3^2}{3} (\ddot{\theta}_1 + \ddot{\theta}_3) - l_3 c_3 d_2 \dot{\theta}_1 \dot{\theta}_3 - l_3 s_3 \dot{d}_2 \dot{\theta}_1 - l_3 s_3 d_2 \ddot{\theta}_1\right\} \tag{33}$$

4.1 State-Space Equations

After determining the required equations, necessary for torque calculations, the mass/inertia matrix, $M(\Theta)\ddot{\Theta}$, the centrifugal and Coriolis vector, $V(\Theta, \dot{\Theta})$, and the gravity vector, $G(\Theta)$, were determined in order to represent the torques in state-space form, as shown below,

$$\tau = M(\Theta)\ddot{\Theta} + V(\Theta, \dot{\Theta}) + G(\Theta) \tag{34}$$

Finding the mass/inertia matrix, $M(\Theta)\ddot{\Theta}$,

$$M(\Theta) = \begin{bmatrix} \frac{1}{3} m_1 l_1^2 + m_2 l_1^2 + m_2 d_2^2 + 4 m_2 l_1 s_1 c_1 d_2 + m_3 l_1^2 + m_3 d_2^2 + \frac{l_3^2}{3} m_3 - m_3 l_3 s_3 d_2 & m_2 l_1 s_1^2 - m_2 l_1 c_1^2 - m_3 l_1 & \frac{l_3^2}{3} m_3 - \frac{1}{2} m_3 l_3 s_3 d_2 \\ m_2 l_1 s_1^2 - m_2 l_1 c_1^2 - m_3 l_1 & m_2 + m_3 & 0 \\ \frac{l_3^2}{3} m_3 - \frac{1}{2} m_3 l_3 s_3 d_2 & 0 & \frac{l_3^2}{3} m_3 \end{bmatrix}$$

It is to be noted that the mass/inertia matrix is symmetric, hence, it proves that the dynamics analysis has been correctly performed.

Finding the centrifugal and Coriolis vector, $V(\Theta, \dot{\Theta})$,

$$V(\Theta,\dot{\Theta}) = \begin{bmatrix} 2m_2\dot{\theta}_1\dot{d}_2 + 4m_2l_1s_1c_1\dot{d}_2\dot{\theta}_1 + 4m_2l_1c_1^2d_2\dot{\theta}_1{}^2 - 4m_2l_1s_1^2d_2\dot{\theta}_1{}^2 + 4m_2l_1s_1c_1\dot{d}_2\dot{\theta}_1 + 2m_3d_2\dot{d}_2\dot{\theta}_1 - m_3l_3c_3d_2\dot{\theta}_1\dot{\theta}_3 - m_3l_3s_3\dot{d}_2(\dot{\theta}_1 + \frac{1}{2}\dot{\theta_3}) - \frac{1}{2}m_3l_3c_3d_2\dot{\theta}_3{}^2 \\ 4m_2l_1s_1c_1\dot{\theta}_1{}^2 \\ -\frac{1}{2}m_3l_3c_3d_2\dot{\theta}_1\dot{\theta}_3 - \frac{1}{2}m_3l_3s_3\dot{d}_2\dot{\theta}_1 \end{bmatrix}$$

Finally, finding the gravity vector, $G(\Theta)$,

$$G(\Theta) = \begin{bmatrix} \frac{m_1gl_1c_1}{2} + m_2gl_1c_1 + m_2gd_2s_1 + m_3gl_1c_1 + m_3gd_2s_1 + \frac{m_3gl_3}{2}\cos(\theta_1+\theta_3) \\ m_2gc_1 - m_3gc_1 \\ \frac{m_3gl_3}{2}\cos(\theta_1+\theta_3) \end{bmatrix}$$

Now, that all the matrices have been determined, the torques acting at each of the joints can be found by simply adding the rows, associated with each of the joints, as shown below,

Torque acting at the first joint (revolute),

$$\tau_1 = \frac{1}{3}m_1l_1^2\ddot{\theta}_1 + m_2l_1^2\ddot{\theta}_1 + m_2d_2^2\ddot{\theta}_1 + 4m_2l_1s_1c_1d_2\ddot{\theta}_1 + m_3l_1^2\ddot{\theta}_1 + m_3d_2^2\ddot{\theta}_1 + \frac{l_3^2}{3}m_3\ddot{\theta}_1 - m_3l_3s_3d_2\ddot{\theta}_1 + m_2l_1s_1^2\ddot{d}_2 - m_2l_1c_1^2\ddot{d}_2 - m_3l_1\ddot{d}_2 + \frac{l_3^2}{3}m_3\ddot{\theta}_3 - \frac{1}{2}m_3l_3s_3d_2\ddot{\theta}_3 + 2m_2\dot{\theta}_1\dot{d}_2 + 4m_2l_1s_1c_1\dot{d}_2\dot{\theta}_1 + 4m_2l_1c_1^2d_2\dot{\theta}_1{}^2 - 4m_2l_1s_1^2d_2\dot{\theta}_1{}^2 + 4m_2l_1s_1c_1\dot{d}_2\dot{\theta}_1 + 2m_3d_2\dot{d}_2\dot{\theta}_1 - m_3l_3c_3d_2\dot{\theta}_1\dot{\theta}_3 - m_3l_3s_3\dot{d}_2(\dot{\theta}_1 + \frac{1}{2}\dot{\theta_3}) - \frac{1}{2}m_3l_3c_3d_2\dot{\theta}_3{}^2 + \frac{m_1gl_1c_1}{2} + m_2gl_1c_1 + m_2gd_2s_1 + m_3gl_1c_1 + m_3gd_2s_1 + \frac{m_3gl_3}{2}\cos(\theta_1+\theta_3) \quad (35)$$

Torque acting at the second joint (prismatic),

$$\tau_2 = m_2l_1s_1^2\ddot{\theta}_1 - m_2l_1c_1^2\ddot{\theta}_1 - m_3l_1\ddot{\theta}_1 + m_2\ddot{d}_2 + m_3\ddot{d}_2 + 4m_2l_1s_1c_1\dot{\theta}_1{}^2 + m_2gc_1 - m_3gc_1 \quad (36)$$

Torque acting at the third joint (revolute),

$$\tau_3 = \frac{l_3^2}{3}m_3\ddot{\theta}_1 - \frac{1}{2}m_3l_3s_3d_2\ddot{\theta}_1 + \frac{l_3^2}{3}m_3\ddot{\theta}_3 - \frac{1}{2}m_3l_3c_3d_2\dot{\theta}_1\dot{\theta}_3 - \frac{1}{2}m_3l_3s_3\dot{d}_2\dot{\theta}_1 + \frac{m_3gl_3}{2}\cos(\theta_1 + \theta_3) \quad (37)$$

5 TRAJECTORY PLANNING

Trajectories are an integral part when it comes to every robotic manipulator analysis. The basic idea behind defining the trajectory is to move the manipulator from the initial position to the desired final position. Here, both the distance and the orientation of the end-effector changes accordingly. This is done by defining a function for each joint parameter with respect to time. However, the motion is defined is much more detail that just defining the end-effector final position and orientation. This is done by defining a sequence of points.

Another constraint to work on is for the motion of the manipulator to be smooth and not rigid. To do so, the equation defining the jerk should be of at least first order and hence, the equation of the distance for either and joint is defined using a cubic equation. In order to ensure a smooth path, a series of constraints needs to be defined for the path via the points.

5.1 TASK-SPACE TRAJECTORY

In task space trajectory, each point on the path is defined in terms of the position and the orientation of the end-effector with respect to the global coordinate. Therefore, the procedure is initated by defining the end-effector's position and the orientation for the path via the series of points. Once the series of coordinates are defined via path, an equation is derived between each path to form a trajectory as a function of time. These trajectories can be derived by the user based on the requirement of the robotic manipulator. Then using the inverse kinematics, the value of the joint parameter is calculated for single point along the path. This ensures smooth movement of the end-effector with respect to the global coordinate with no jerks.

In our problem statement, the trajectory of the end-effector is defined as the following:

Trajectory of end-effector from initial position to the point: - 1m to the x axis

Trajectory of end-effector from point to the final position: - 0.5m to the y axis

$$(x_o, y_o) = (1.5,2.0)$$

$$(x_m, y_m) = (2.5,2.0)$$

$$(x_f, y_f) = (2.5,2.5)$$

5.2 CALCULATIONS

From initial point to middle point,

From $(x_o, y_o) = (1.5,2.0)$, To $(x_m, y_m) = (2.5,2.0)$

For x-axis

$$x(0) = x_0 = 1.5\ m$$

$$x(t_{f1}) = x_m = 2.5\ m$$

$$\dot{x}(0) = 0\ m/s$$

$$\dot{x}(t_{f1}) = 0\ m/s$$

Where $t_f = 1\ sec$

$$x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$\dot{x}(t) = a_1 + 2a_2 t + 3a_3 t^2$$

If $t = 0\ sec$

$$x_0 = x(0) = a_0 + a_1(0) + a_2(0)^2 + a_3(0)^3 = 1.5$$

$$x(0) = a_0 = 1.5$$

$$\dot{x}(0) = a_1 + 2a_2(0) + 3a_3(0)^2$$

$$\dot{x}(0) = a_1 = 0$$

If $t = t_{f1} = 1\ sec$

$$x(t_{f1}) = a_0 + a_1(1) + a_2(1)^2 + a_3(1)^3$$

$$x(1) = x_m = a_0 + a_1 + a_2 + a_3 = 2.5$$

$$\dot{x}(1) = a_1 + 2a_2 + 3a_3 = 0$$

Combing into matrix form

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_{f1} & t_{f1}^2 & t_{f1}^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_{f1} & 3t_{f1}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} x(0) \\ x(t_{f1}) \\ 0 \\ 0 \end{bmatrix}$$

Substituting the values in the matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

Solving the matrix equation

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix}^{-1} \begin{bmatrix} 1.5 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 0 \\ 3 \\ -2 \end{bmatrix}$$

$$x(t) = 1.5 + 3t^2 - 2t^3$$

For y-axis

$$y(0) = y_0 = 2.0\ m$$

$$y(t_{f1}) = y_m = 2.0\ m$$

$$\dot{y}(0) = 0\ m/s$$

$$\dot{y}(t_{f1}) = 0\ m/s$$

Where $t_{f1} = 1\ sec$

$$y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$\dot{y}(t) = a_1 + 2a_2 t + 3a_3 t^2$$

If $t = 0\ sec$

$$y_0 = y(0) = a_0 + a_1(0) + a_2(0)^2 + a_3(0)^3 = 2.0$$

$$y(0) = a_0 = 2.0$$

$$\dot{y}(0) = a_1 + 2a_2(0) + 3a_3(0)^2$$

$$\dot{y}(0) = a_1 = 0$$

If $t = t_f = 1\ sec$

$$y(t_{f1}) = a_0 + a_1(1) + a_2(1)^2 + a_3(1)^3$$

$$y(1) = y_m = a_0 + a_1 + a_2 + a_3 = 2.0$$

$$\dot{y}(1) = a_1 + 2a_2 + 3a_3 = 0$$

Combing into matrix form

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_{f1} & t_{f1}^2 & t_{f1}^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_{f1} & 3t_{f1}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} y(0) \\ y(t_{f1}) \\ 0 \\ 0 \end{bmatrix}$$

Substituting the values in the matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 2.0 \\ 0 \\ 0 \end{bmatrix}$$

Solving the matrix equation

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix}^{-1} \begin{bmatrix} 2.0 \\ 2.0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$y(t) = 2.0$$

From middle point to final point,

From middle point $(x_m, y_m) = (2.5,2.0)$, To $\left(x_f, y_f\right) = (2.5,2.5)$

For x-axis

$$x\left(t_{f1}\right) = x_m = 2.5\ m$$

$$x\left(t_{f2}\right) = x_f = 2.5\ m$$

$$\dot{x}\left(t_{f1}\right) = 0\ m/s$$

$$\dot{x}\left(t_{f2}\right) = 0\ m/s$$

Where $t_{f1} = 1\ sec$, $t_{f2} = 2\ sec$

$$x(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$\dot{x}(t) = a_1 + 2a_2 t + 3a_3 t^2$$

If $t_{f1} = 1\ sec$

$$x\left(t_{f1}\right) = x_m = a_0 + a_1 t_{f1} + a_2 t_{f1}^2 + a_3 t_{f1}^3$$

$$x_m = x(1) = a_0 + a_1(1) + a_2(1)^2 + a_3(1)^3 = 2.5$$

$$x(1) = a_0 + a_1 + a_2 + a_3 = 2.5$$

$$\dot{x}(1) = a_1 + 2a_2(1) + 3a_3(1)^2$$

$$\dot{x}(1) = a_1 + 2a_2 + 3a_3 = 0$$

If $t = t_{f2} = 2\ sec$

$$x\left(t_{f2}\right) = x_f = a_0 + a_1 t_{f2} + a_2 t_{f2}^2 + a_3 t_{f2}^3$$

$$x_f = x(2) = a_0 + a_1(2) + a_2(2)^2 + a_3(2)^3 = 2.5$$

$$\dot{x}(t_{f1}) = a_1 + 2a_2 t_{f1} + 3a_3 t_{f1}^2 = 0$$

Combing into matrix form

$$\begin{bmatrix} 1 & t_{f1} & t_{f1}^2 & t_{f1}^3 \\ 1 & t_{f2} & t_{f2}^2 & t_{f2}^3 \\ 0 & 1 & 2t_{f1} & 3t_{f1}^2 \\ 0 & 1 & 2t_{f2} & 3t_{f2}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} x(t_{f1}) \\ x(t_{f2}) \\ 0 \\ 0 \end{bmatrix}$$

Substituting the values in the matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2.5 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

Solving the matrix equation

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1^2 & 1^3 \\ 1 & 2 & 2^2 & 2^3 \\ 0 & 1 & 2(1) & 3(1)^2 \\ 0 & 1 & 2(2) & 3(2)^2 \end{bmatrix}^{-1} \begin{bmatrix} 2.5 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2.5 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$x(t) = 2.5$$

For y-axis

$$y\left(t_{f1}\right) = y_m = 2.0\ m$$

$$y\left(t_{f2}\right) = y_f = 2.5\ m$$

$$\dot{y}\left(t_{f1}\right) = 0\ m/s$$

$$\dot{y}\left(t_{f2}\right) = 0\ m/s$$

Where $t_{f1} = 1\ sec$, $t_{f2} = 2\ sec$

$$y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$\dot{y}(t) = a_1 + 2a_2t + 3a_3t^2$$

If $t_{f1} = 1\ sec$

$$y(t_{f1}) = y_m = a_0 + a_1t_{f1} + a_2t_{f1}^2 + a_3t_{f1}^3$$

$$y_m = y(1) = a_0 + a_1(1) + a_2(1)^2 + a_3(1)^3 = 2.0$$

$$y(1) = a_0 + a_1 + a_2 + a_3 = 2.0$$

$$\dot{y}(1) = a_1 + 2a_2(1) + 3a_3(1)^2$$

$$\dot{y}(1) = a_1 + 2a_2 + 3a_3 = 0$$

If $t = t_{f2} = 2\ sec$

$$y(t_{f2}) = y_f = a_0 + a_1t_{f2} + a_2t_{f2}^2 + a_3t_{f2}^3$$

$$y_f = y(2) = a_0 + a_1(2) + a_2(2)^2 + a_3(2)^3 = 2.5$$

$$\dot{y}(t_{f1}) = a_1 + 2a_2t_{f1} + 3a_3t_{f1}^2 = 0$$

Combing into matrix form

$$\begin{bmatrix} 1 & t_{f1} & t_{f1}^2 & t_{f1}^3 \\ 1 & t_{f2} & t_{f2}^2 & t_{f2}^3 \\ 0 & 1 & 2t_{f1} & 3t_{f1}^2 \\ 0 & 1 & 2t_{f2} & 3t_{f2}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} y(t_{f1}) \\ y(t_{f2}) \\ 0 \\ 0 \end{bmatrix}$$

Substituting the values in the matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1^2 & 1^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2(1) & 3(1)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

Solving the matrix equation

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1^2 & 1^3 \\ 1 & 2 & 2^2 & 2^3 \\ 0 & 1 & 2(1) & 3(1)^2 \\ 0 & 1 & 2(2) & 3(2)^2 \end{bmatrix}^{-1} \begin{bmatrix} 2.5 \\ 2.5 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 4.5 \\ -6 \\ 4.5 \\ -1 \end{bmatrix}$$

$$y(t) = 4.5 - 6t + 4.5t^2 - t^3$$

Summary,

From initial point to middle point

$$x(t) = 1.5 + 3t^2 - 2t^3$$

$$y(t) = 2.0$$

Where t is from 0 to 1 sec.

From middle point to final point

$$x(t) = 2.5$$

$$y(t) = 4.5 - 6t + 4.5t^2 - t^3$$

Where t is from 1 to 2 sec.

This is the code which is used to define the simulation of the trajectory as well as describe the relationship of torque with respect to time.

```
clc;clear all;
format short
m1 = 1;
m2 = 1;
m3 =1;
g = 9.81;
L1=0.8;L3=0.8;
PHIH = pi/2;
%start point
xo=1.5;
yo=2.0;
vxo=0;vyo=0;

   %via point
xm=2.5;
ym=2.0;
vxm=0;
vym=0;

   %goal pint
xf=2.5;
yf=2.5;
 vfx=0;
 vfy=0;

xi = xo-L3*cos(PHIH);
yi = yo-L3*sin(PHIH);
d2=sqrt(xi^2+yi^2-L1^2);
q1=2*atan2((d2+sqrt((d2^2)-(L1^2)-(xi^2))),(L1+xi));
q3=PHIH-q1;

q=q1;d=0;a=0;alpha=0; %for i=1
T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;d=d2;a=L1;alpha=pi/2; %for i=2
T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=q3;d=0;a=0;alpha=-pi/2; %for i=3
T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;a=L3;d=0;alpha=0; %for i=4
T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
T02=T01*T12;
T03=T01*T12*T23;
T04=T01*T12*T23*T34;

axis([-2 3 -2 3]);
```

```
  Ax1 = [T01(1,4),T02(1,4)];
  Ay1 = [T01(2,4),T02(2,4)];
  Ax2 = [T02(1,4),T03(1,4)];
  Ay2 = [T02(2,4),T03(2,4)];
  Ax3 = [T03(1,4),T04(1,4)];
  Ay3 = [T03(2,4),T04(2,4)];
  Ax4 = [-.1,.1];
  Ay4=  [0,0];

  p1  = line(Ax1,Ay1,'LineWidth',[3],'Color','k');
  p2  = line(Ax2,Ay2,'LineWidth',[3],'Color','M');
  p3  = line(Ax3,Ay3,'LineWidth',[3],'Color','R');
  p4  = line(Ax4,Ay4,'LineWidth',[3],'Color','B');

drawnow
tf1=1;
tf2=2;
% for x

A=[1 0 0 0;1 tf1 tf1^2 tf1^3;0 1 0 0;0 1 2*tf1 3*tf1^2];
B=[xo xm vxm 0]';
cof=inv(A)*B;
a1o=cof(1,1);a11=cof(2,1);a12=cof(3,1);a13=cof(4,1);
%for y
A=[1 0 0 0;1 tf1 tf1^2 tf1^3;0 1 0 0;0 1 2*tf1 3*tf1^2];
B=[yo ym 0 0]';
cof=inv(A)*B;
b1o=cof(1,1);b11=cof(2,1);b12=cof(3,1);b13=cof(4,1);
%for x
A=[1 tf1 tf1^2 tf1^3;1 tf2 tf2^2 tf2^3;0 1 2*tf1 3*tf1^2;0 1 2*tf2 3*tf2^2];
B=[xm xf 0 0]';
cof=inv(A)*B;
a2o=cof(1,1);a21=cof(2,1);a22=cof(3,1);a23=cof(4,1);
%for y
A=[1 tf1 tf1^2 tf1^3;1 tf2 tf2^2 tf2^3;0 1 2*tf1 3*tf1^2;0 1 2*tf2 3*tf2^2];
B=[ym yf vym 0]';
cof=inv(A)*B;
b2o=cof(1,1);b21=cof(2,1);b22=cof(3,1);b23=cof(4,1);

pause()
n=1;
for t=0:.01:1.99
    time(n) = t;
    if t<=1
x=a1o+a11*t+a12*t^2+a13*t^3;
y=b1o+b11*t+b12*t^2+b13*t^3;
    else
x=a2o+a21*t+a22*t^2+a23*t^3;
y=b2o+b21*t+b22*t^2+b23*t^3;
    end
xi = x-L3*cos(PHIH);
yi = y-L3*sin(PHIH);
d2=sqrt(xi^2+yi^2-L1^2);
q1=2*atan2((d2+sqrt((d2^2)-(L1^2)-(xi^2))),(L1+xi));
q3=PHIH-q1;
```

```
q1a(n) = q1;
d2a(n) = d2;
q3a(n) = q3;
   if t<=1
vq1(n) = 0.0057*t.^2+0.0354*t-0.028;
aq1(n) = 0.0114*t+0.0354;
vd2(n) = -5.5023*t.^2+5.5280*t-0.0539;
ad2(n) = -11.0046*t+5.5280;
vq3(n) = -0.0057*t.^2+0.0354*t+0.0678
aq3(n) = -0.0114*t+0.0354;

    else
vq1(n) = -1.5159*t.^2+4.2712*t-2.5556;
aq1(n) = -3.0318*t+4.2712;
vd2(n) = -1.5624*t.^2+4.7596*t-3.2332;
ad2(n) = -3.1248*t+4.7596;
vq3(n) = 1.4586*t.^2-4.0676*t+2.3841;
aq3(n) = 2.9172*t-4.0676;
    end

M =
[1/3*m1*L1^2+m2*L1^2+m2*d2a(n)^2+4*m2*L1*sin(q1a(n))*cos(q1a(n))*d2a(n)+m3*(L
1^2)+m3*d2a(n)^2+((L3^3)*m3/3)-m3*L3*sin(q1a(n))*d2a(n)
(m2*L1*(sin(q1a(n))^2)-(m2*L1*(cos(q1a(n))^2)-m3*L1)) m3/3*(L3)^3;
m2*L1*sin(q1a(n))^2-(m2*L1*cos(q1a(n))^2)-m3*L1 (m2+m3) 0;(m3*(L3^3)/3)-
0.5*m3*L3*d2*sin(aq3(n)) 0 (m3*(L3^3)/3)-0.5*m3*L3*sin(q1a(n))*d2a(n)];
V =
[(2*m2*vq1(n)*vd2(n))+(8*m2*L1*sin(q1a(n))*cos(q1a(n))*vd2(n)*vq1(n))+4*m2*L1
*(cos(q1a(n))^2)*d2a(n)*(vq1(n)^2)-
4*m2*L1*(sin(q1a(n))^2)*d2a(n)*(vq1(n)^2)+(2*m3*d2a(n)*vd2(n)*vq1(n))-
(m3*L3*cos(q1a(n))*d2a(n)*vq1(n)*vq3(n))-
(m3*L3*sin(q1a(n))*vd2(n)*(vq1(n)+0.5*vq3(n))-
0.5*m3*L3*cos(q3a(n))*d2a(n)*vq3(n)^2);
4*m2*L1*sin(q1a(n))*cos(q1a(n))*vq1(n)^2; -
0.5*m3*L3*cos(q1a(n))*d2a(n)*vq1(n)*vq3(n)-
0.5*m3*L3*sin(q1a(n))*d2a(n)*vd2(n)*vq1(n)];
G =
[0.5*m1*L1*g*cos(q1a(n))+m2*g*L1*cos(q1a(n))+m2*g*d2a(n)*sin(q1a(n))+m3*g*L1*
cos(q1a(n))+m3*g*d2a(n)*sin(q1a(n))+0.5*m3*g*L3*cos(q1a(n)+q3a(n));
m2*g*cos(q1a(n))-m3*g*cos(q1a(n)); 0.5*m3*g*L3*cos(q1a(n)+q3a(n))]

T=M*[aq1(n);ad2(n);aq3(n)]+V+G;
T1(n)=T(1,1);
T2(n)=T(2,1);
T3(n)=T(3,1);

Xp = L3*(cos(q1+q3))+L1*cos(q1)+d2*sin(q1);
Yp = L3*(sin(q1+q3))+L1*sin(q1)-d2*cos(q1);
q=q1;d=0;a=0;alpha=0; %for i=1
T01=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;d=d2;a=L1;alpha=pi/2; %for i=2
T12=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
```

```
q=q3;d=0;a=0;alpha=-pi/2; %for i=3
T23=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
q=0;a=L3;d=0;alpha=0; %for i=4
T34=[cos(q) -sin(q) 0 a;sin(q)*cos(alpha) cos(q)*cos(alpha) -sin(alpha) -
sin(alpha)*d;sin(q)*sin(alpha) cos(q)*sin(alpha) cos(alpha) cos(alpha)*d;0 0
0 1];
T02=T01*T12;
T03=T01*T12*T23;
T04=T01*T12*T23*T34;

  Ax1 = [T01(1,4),T02(1,4)];
  Ay1 = [T01(2,4),T02(2,4)];
  Ax2 = [T02(1,4),T03(1,4)];
  Ay2 = [T02(2,4),T03(2,4)];
  Ax3 = [T03(1,4),T04(1,4)];
  Ay3 = [T03(2,4),T04(2,4)];
  Ax4 = [-.1,.1];
  Ay4=  [0,0];

  set(p1,'X', Ax1, 'Y',Ay1)
  set(p2,'X', Ax2, 'Y',Ay2)
  set(p3,'X', Ax3, 'Y',Ay3)
  set(p4,'X', Ax4, 'Y',Ay4)

  o1(n,1)=Xp;
  o2(n,1)=Yp;


  drawnow
  pause(.01)
  n=n+1;
  end
hold on
plot(o1(:,1),o2(:,1))
xx = time';
yy = q1a';
zz = d2a';
aa = q3a';
```

6 FINAL RESULTS

This is the initial plot of the robotic manipulator at (x_o, y_o) is shown in

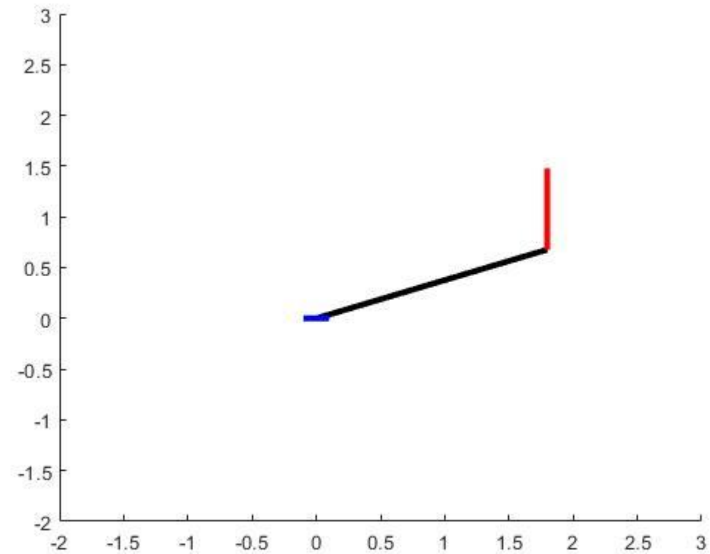

Figure 7: Initial plot

Furthermore, the plot of the trajectory is given in the graph below.

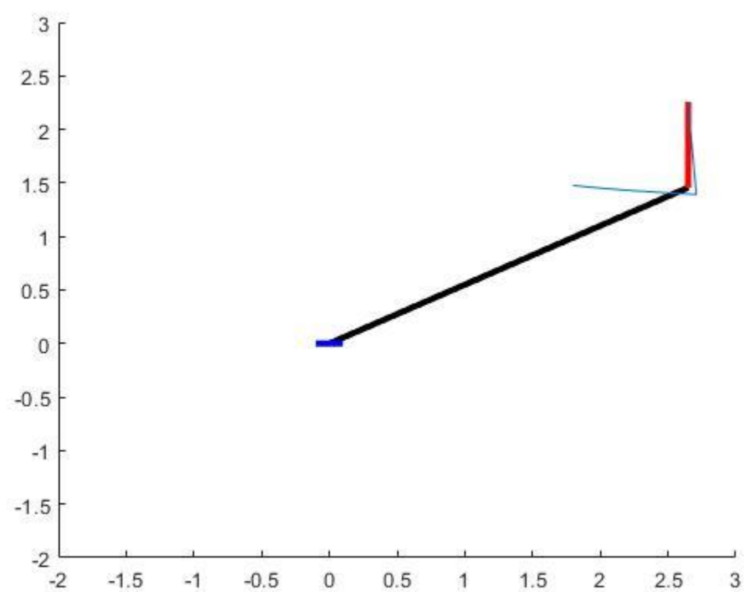

Figure 8: Trajectory

6.1 JOINT TORQUES IN TRAJECTORY PATH

Our next objective is to find the torque exerted on each joint through the motion. This allows us to check that the torque does not exceed the limits. To start off, the displacement, velocity and the acceleration of the joints needs to be defined through out the motion. However, since the task trajectory was performed, there is no equation which defines the joint parameter's velocity and acceleration as a function of time. To cater this, polynomial fit of the graph is performed in order to define the equation of joint displacement as a function of time. Later on, the first and second order derivative allows the equation of the velocity and acceleration be defined respectively.

The graph of second joint displacement against time is shown in Figure 7.

Finally, for the third joint, the displacement graph is shown in Figure 8.

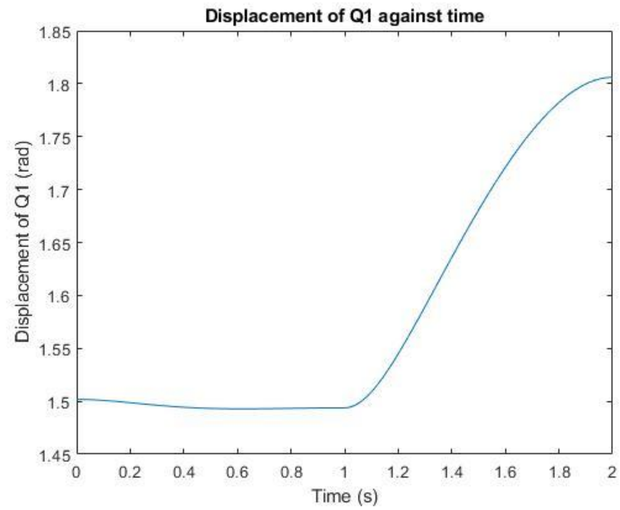

Figure 9:The graph of first joint displacement against time

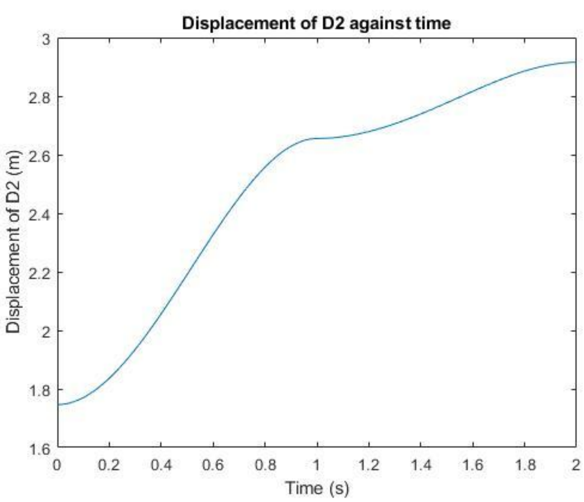

Figure 10: Displacement of second joint

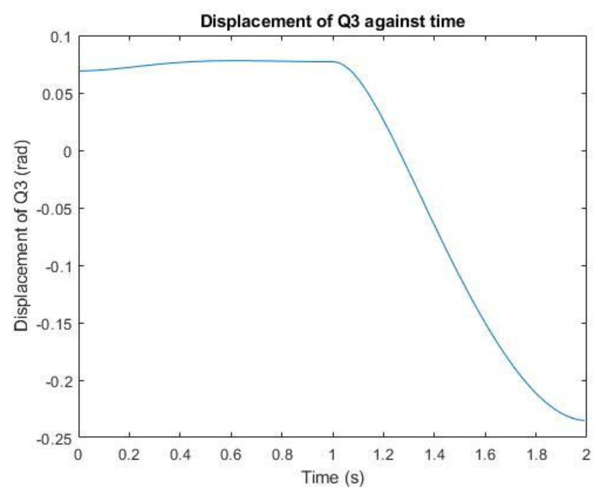

Figure 11: Displacement of third joint

Since the motion is defined in two points, hence the analysis of the motion will be divided into 2 parts.

To start off the analysis of motion for the first part is performed.

From initial point $(x_o, y_o) = (1.5,2.0)$, To middle point $(x_m, y_m) = (2.5,2.0)$

After performing the polynomial fit of the curve of joint 1,

.

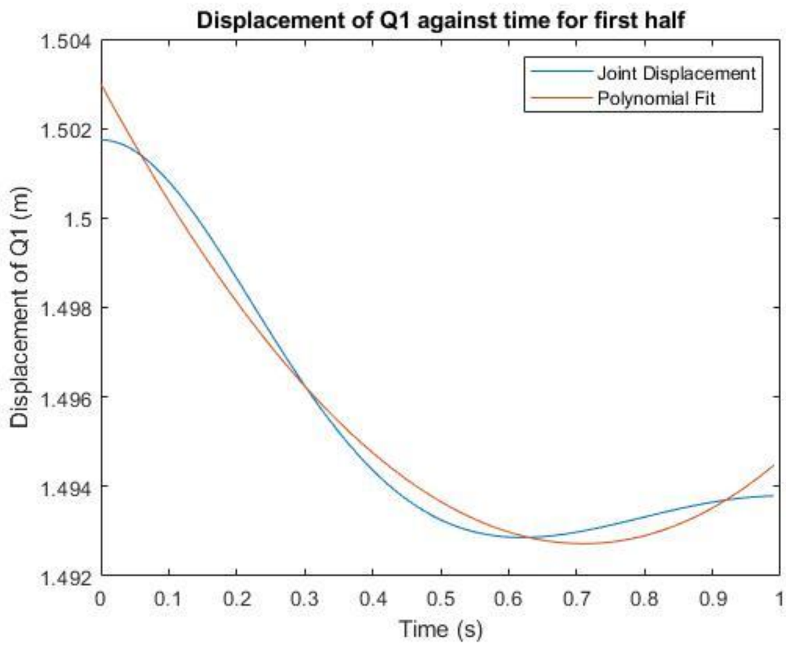

Figure 12: Joint 1 first half displacement

$$q_1 = 0.0019t^3 + 0.0177t^2 - 0.028t + 1.503 \tag{38}$$

After performing the polynomial fit of the curve of joint 2,

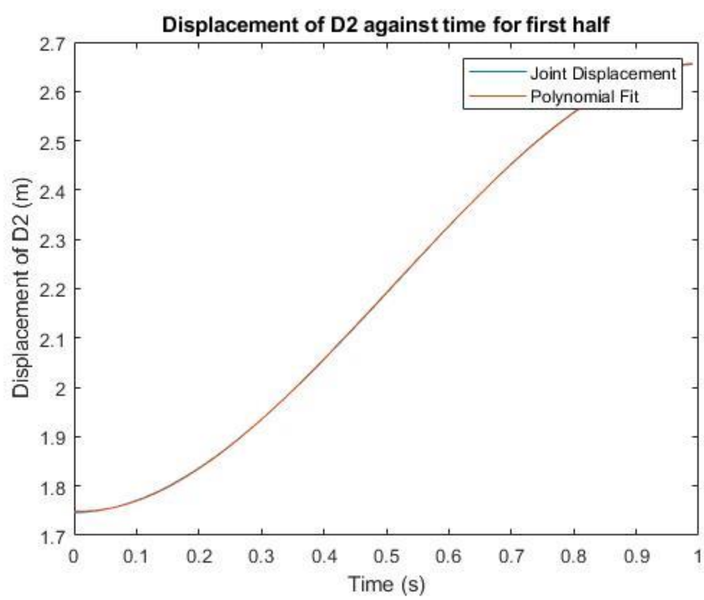

Figure 13:Joint 2 first half displacement

$$d_2 = -1.8341t^3 + 2.764t^2 - 0.0539t + 1.79 \tag{39}$$

After performing the polynomial fit of the curve of joint 3,

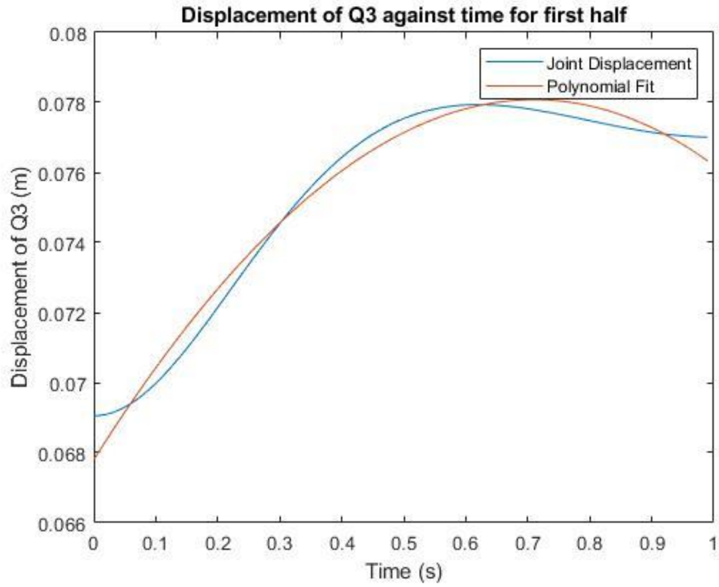

Figure 14:Joint 3 first half displacement

$$q_3 = -0.0019t^3 - 0.0177t^2 + 0.028t + 0.0678 \quad (40)$$

The analysis of motion for the second part is performed as follows,

From middle point $(x_m, y_m) = (2.5,2.0)$, To $(x_f, y_f) = (2.5,2.5)$

After performing the polynomial fit of the curve of joint 1.

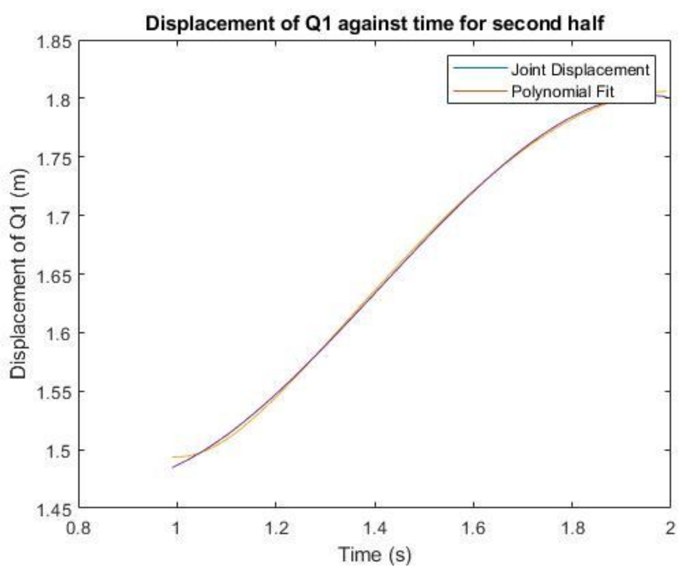

Figure 15: Joint 1 second half displacement

$$q_1 = -0.5053t^3 + 2.1356t^2 - 2.5556t + 2.412 \tag{41}$$

After performing the polynomial fit of the curve of joint 2.

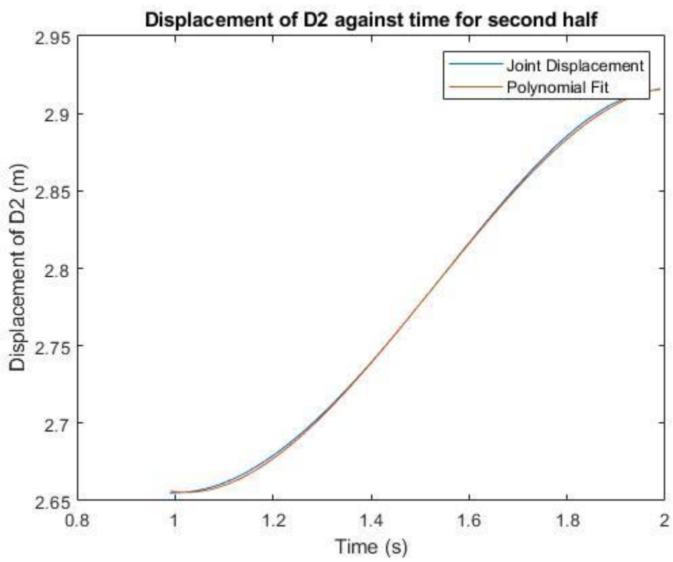

Figure 16:Joint 2 second half displacement

$$d_2 = -0.5208t^3 + 2.3798t^2 - 3.2332t + 4.0309 \tag{42}$$

After performing the polynomial fit of the curve of joint 3.

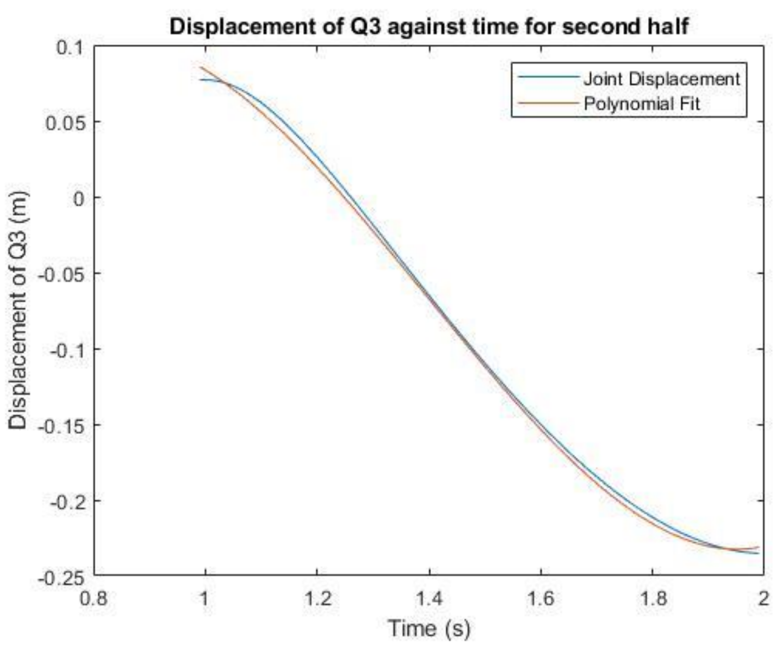

Figure 17: Joint 3 second half displacement

$$q_3 = 0.4862t^3 - 2.0338t^2 + 2.3841t - 0.7531 \quad (43)$$

After defining each joint parameter equation in terms of time, the first and second derivative of the equation allow us to find the velocity, and accleration of the joint parameter.

For the first joint,

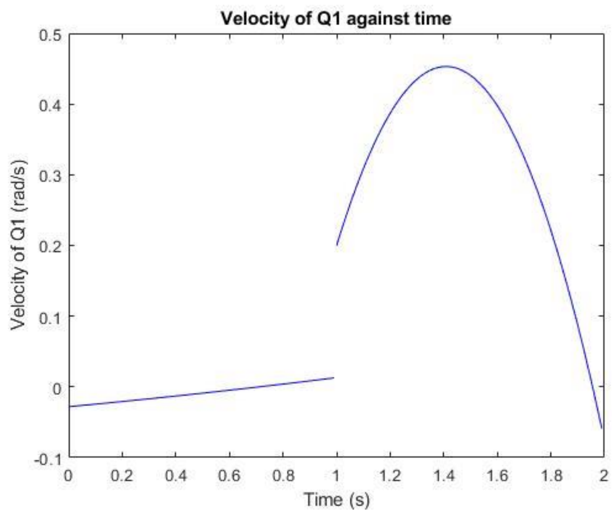

Figure 19: Velocity against time

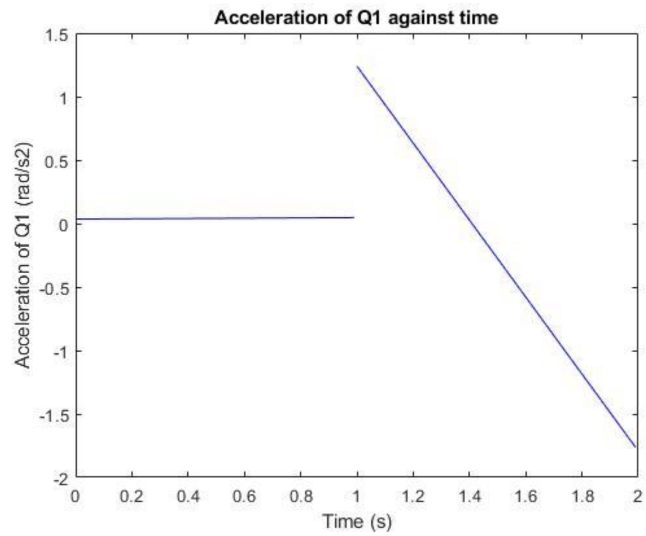

Figure 18: Acceleration against time

For the second joint,

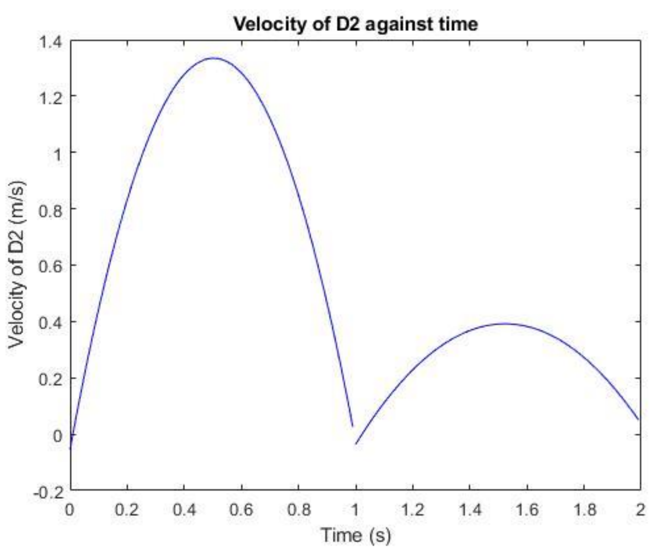

Figure 21:Velocity against time

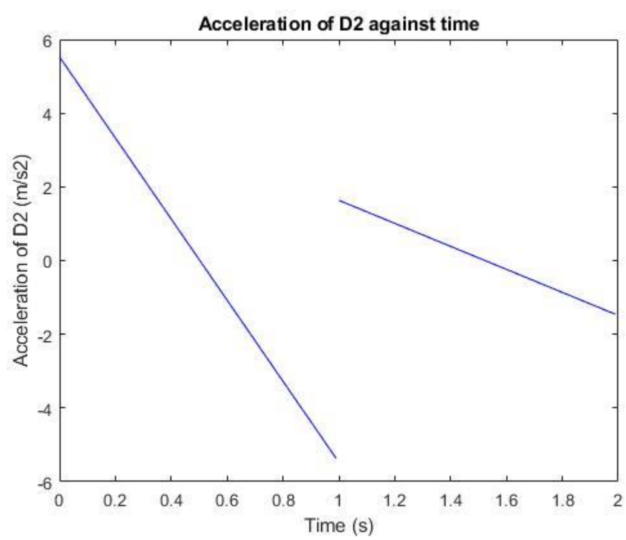

Figure 20:Acceleration against time

For the third joint,

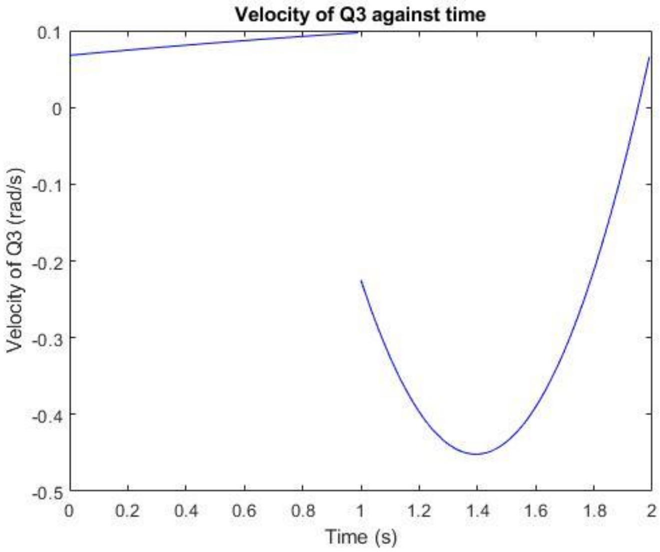

Figure 22:Velocity against time

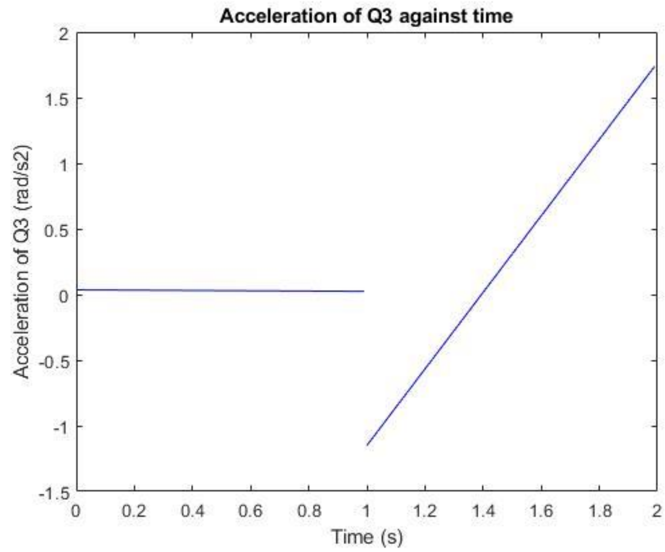

Figure 23: Acceleration against time

With this, the velocity and acceleration of each joint parameter is defined as a function of time

6.2 TORQUE CALCULATION

Once the displacement, velocity, and acceleration of each joint is defined. Then the torque can be calculated. This can be done by using the Lagrange's model as stated in the previous section. The equation of the model is given below

Since the equation of the displacement, velocity, and acceleration of each joint is defined, the following parameters is then substituted in the above equation for each specific time. The graph of torques is given below:

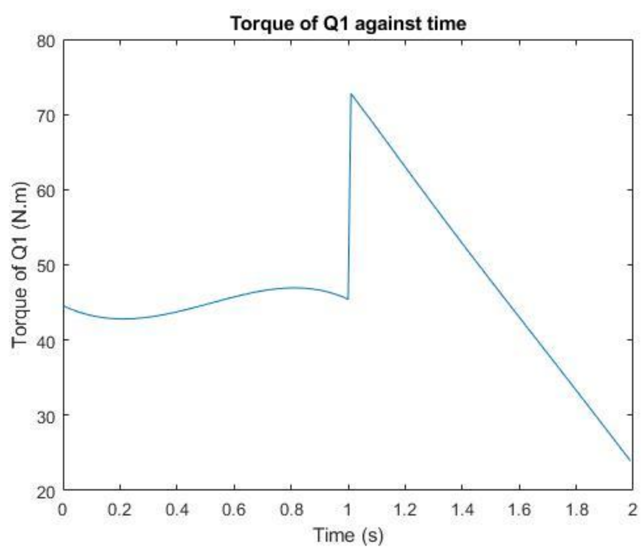

Figure 24: Torque at first joint

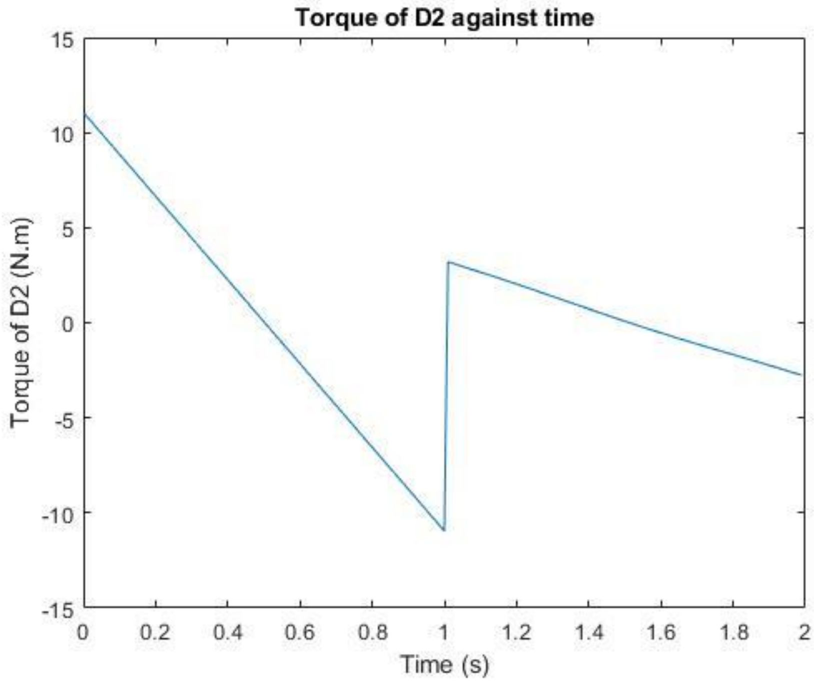

Figure 25: Torque at second joint

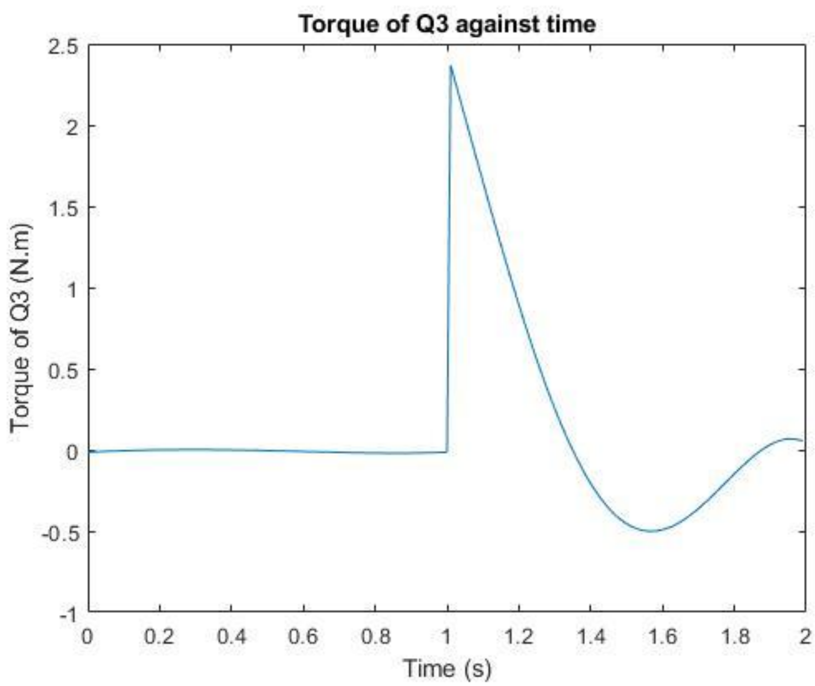

Figure 26: Torque at third joint

7 CONCLUSION

In this report, an RPR manipulator has been designed based on the requirements of dexterous and reachable workspace. First, the coordinate frames were attached to each of the joints of the links and to the end-effector. Based on the coordinate frames, the DH table was constructed which was further analyzed to build the transformation matrix, defining the task space coordinates in the inertial frame. The transformation matrices were then used to construct a forward kinematics code on MATLAB, where the joint variables were defined in terms of time, and using those joint parameters, the task space coordinates were calculated. Moreover, the home position and final position of the manipulator were also determined through the joint parameter equations.

After performing the forward kinematics, the manipulator was analyzed through inverse kinematics approach. In this case, first the joint parameters were defined in terms of the cartesian coordinates determined from the final transformation matrix. Then, the cartesian coordinates were defined as functions of φ and radius of the circle. Finally, the joint parameters' values were determined, and the desired trajectory was drawn on MATLAB. After performing a detailed kinematic analysis, the values of the parameters will be further utilized in performing the dynamic analysis, and in determining the closed form solution of the manipulator.

Furthermore, using Langrage's analysis, the coefficient matrix of each joint was then defined. Later on, task space analysis allowed us to define the trajectory of the end-effector and the joint parameters. Lastly, using the previous model generated, the torque on each joint was defined as a function of time. The analysis and simulation were conducted on the MATLAB software.

8 REFERENCES

1. https://www.osha.gov/dts/osta/otm/otm_iv/otm_iv_4.html
2. *"Robot Arm and Computer Vision". Retrieved 29 July 2016*
3. *Craig, John J. Introduction to robotics mechanics and control*