İSTANBUL TEKNİK ÜNİVERSİTESİ

KONTROL VE OTOMASYON MÜHENDİSLİĞİ BÖLÜMÜ

KON318E – INTRODUCTION TO ROBOTICS

ÖDEV #2

Öğretim Üyesi : Prof. Dr. Hakan TEMELTAŞ

Öğrencinin;

Adı Soyadı : Ecem Işıldar

Numarası : 040170450

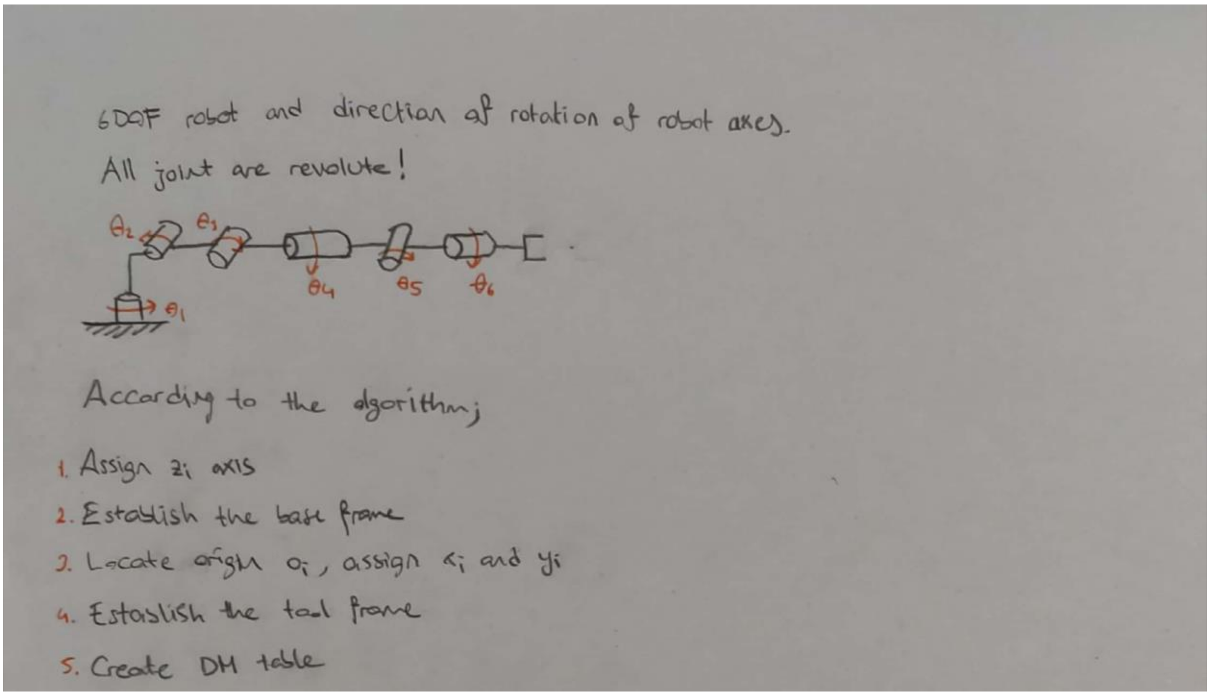
Bu raporda yer alan tüm içeriğin tamamen şahsıma ait olduğunu beyan ederim.

Tarih: 22/11/2021

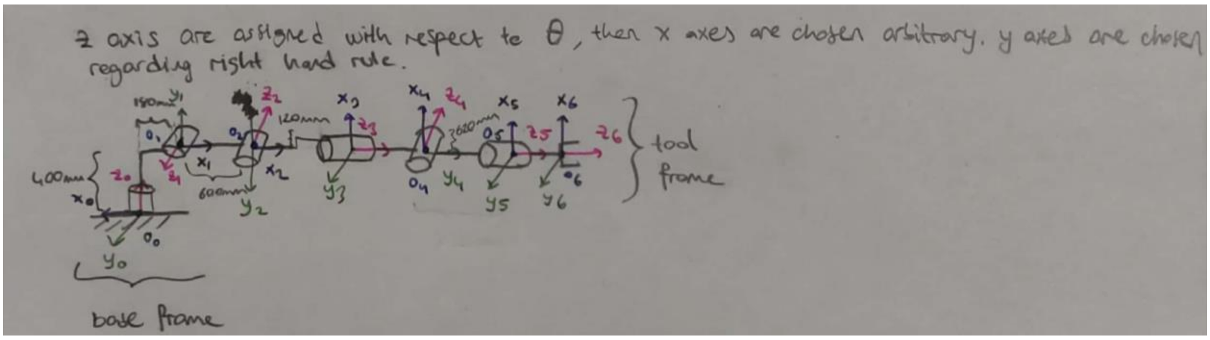
İmza:

Question 1: Kuka KR 5 ARC, a 6 DOF industrial robot, is given in Fig. 1. For this robot manipulator system:

a) Determine the joint types and the dimensions of the robot using the datasheet attached.



b) Assign the coordinate frames of the robot manipulator.



c) Define DH-table.

Using the figure below, twist angle, link length, joint offset and joint angle are chosen:

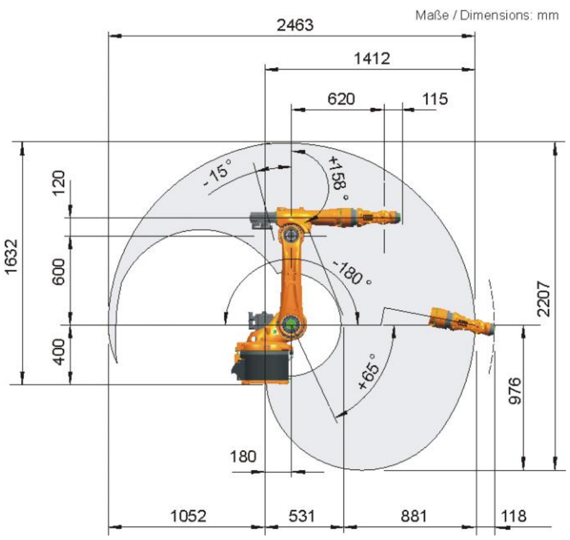


Figure 1: dimensions

To find DH table, $\alpha_i, d_i, \theta_i, a_i$ definitions are used.

Link	a_i	α_i	d_i	θ_i
1	180mm	+90°	400mm	θ_1^*
2	600mm	180°	0	θ_2^*
3	120mm	-90°	0	θ_3^*
4	0	+90°	620mm	θ_4^*
5	0	-90°	0	θ_5^*
6	0	0	115mm	θ_6^*

180mm is the difference between first two joints
600mm is the difference between 2nd and 3rd joints
120mm is the difference between 3rd and 4th joints on x_3 axis.
After wrist point there is no link length a.
400mm distance between first 2 joints in z axis.
620 mm distance between 4. and 5. joints in z axis.
115 mm distance between last 2 joints.

α_i: Angle from z_{i-1} to z_i around x_i
θ_i: Depends on the rotation angles

d) Write the forward kinematic model of the robot manipulator by means of the DH table.

$$T_6^1 = A_1A_2A_3A_4A_5A_6 \Rightarrow A_i = \begin{bmatrix} \cos\theta & -\sin\theta\cos\alpha & \sin\theta\sin\alpha & a.\cos\theta \\ \sin\theta & \cos\theta\cos\alpha & -\cos\theta\sin\alpha & a.\sin\theta \\ 0 & \sin\alpha & \cos\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0.18c\theta_1 \\ s\theta_1 & 0 & -c\theta_1 & 0.18s\theta_1 \\ 0 & 1 & 0 & 0.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} c\theta_2 & s\theta_2 & 0 & 0.6c\theta_2 \\ s\theta_2 & -c\theta_2 & 0 & 0.6s\theta_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} c\theta_3 & 0 & -s\theta_3 & 0.12c\theta_3 \\ s\theta_3 & 0 & c\theta_3 & 0.12s\theta_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_4 = \begin{bmatrix} c\theta_4 & 0 & s\theta_4 & 0 \\ s\theta_4 & 0 & -c\theta_4 & 0 \\ 0 & 1 & 0 & 0.62 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} c\theta_5 & 0 & -s\theta_5 & 0 \\ s\theta_5 & 0 & c\theta_5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_6 = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ s\theta_6 & c\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0.115 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The forward kinematic model is found by matlab!

Code:

```
syms th1 th2 th3 th4 th5 th6;

theta=[th1 th2 th3 th4 th5 th6];
a=[180 600 120 0 0 0]*0.001;%meter
alpha=[90 180 -90 90 -90 0];%degree
d=[400 0 0 620 0 115]*0.001;%meter
%theta=[0 0 0 0 0 0];%degree
t=1;mul=1;
for i=1:6%6 DOF
A=[cosd(theta(i)*t), -sind(theta(i)*t)*cosd(alpha(i)),
sind(theta(i)*t)*sind(alpha(i)), a(i)*cosd(theta(i)*t);
    sind(theta(i)*t), cosd(theta(i)*t)*cosd(alpha(i)), -
cosd(theta(i)*t)*sind(alpha(i)), a(i)*sind(theta(i)*t);
    0, sind(alpha(i)), cosd(alpha(i)), d(i);
    0, 0, 0, 1];
mul=mul*A;

end

forward_kinematic=mul
```

Matrix of the first joint:

A1=

[cos((pi*th1)/180), 0, sin((pi*th1)/180), (9*cos((pi*th1)/180))/50]

[sin((pi*th1)/180), 0, -cos((pi*th1)/180), (9*sin((pi*th1)/180))/50]

[0, 1, 0, 2/5]

[0, 0, 0, 1]

Matrix of the second joint:

T2=A1*A2=

[cos((pi*th1)/180)*cos((pi*th2)/180), cos((pi*th1)/180)*sin((pi*th2)/180), -sin((pi*th1)/180), (9*cos((pi*th1)/180))/50 + (3*cos((pi*th1)/180)*cos((pi*th2)/180))/5]

[cos((pi*th2)/180)*sin((pi*th1)/180), sin((pi*th1)/180)*sin((pi*th2)/180), cos((pi*th1)/180), (9*sin((pi*th1)/180))/50 + (3*cos((pi*th2)/180)*sin((pi*th1)/180))/5]

[sin((pi*th2)/180), -cos((pi*th2)/180), 0, (3*sin((pi*th2)/180))/5 + 2/5]

[0, 0, 0, 1]

After second matrix, it is really hard to show a kinematic model. For that reason, I calculated the initial kinematic model using Matlab. In this situation, all theta angles are equal to zero degree.

forward_kinematic =

1.0000	0	0	0.9000
0	-1.0000	0	0
0	0	-1.0000	-0.3350
0	0	0	1.0000

e) Prove that your forward kinematic model is correct.

i. Choose two separate suitable operating points for all joints considering the range of motion of the axes listed in the datasheet of the robot manipulator.

ii. Calculate the position and orientation of the tip point with respect to base frame for each case using MATLAB symbolic toolbox and comment on your results.

The joint angles are chosen using with this table

Axis	Range of motion, software-limited	Speed with rated payload
1	+/-155°	154°/s
2	+65° to -180°	154°/s
3	+158° to -15°	228°/s
4	+/-350°	343°/s
5	+/-130°	384°/s
6	+/-350°	721°/s

Two seperate points for all joints considering the range of motion of axes.

1. $Joints_{(1 to 6)} = [-90° \; 70° \; 160° \; 300° \; 100° \; -120°]$

2. $Joints_{(1 to 6)} = [-50° \; 100° \; 0° \; 120° \; 150° \; 20°]$

For the first point the kinematic model is

$$\left[\begin{array}{ccc|c} -0.5082 & 0.1198 & -0.8529 & -0.0981 \\ -0.4924 & 0.8529 & -0.1736 & 0.2148 \\ 0.7066 & 0.5082 & 0.4924 & 0.9004 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

3x3 part is orientation of the tool frame with respect to base frame and 3x1 part is the position of the tool frame with respect to the base frame.

For the second point the kinematic model is

$$\left[\begin{array}{ccc|c} 0.9560 & 0.1625 & -0.2444 & 0.3997 \\ 0.2232 & 0.1182 & 0.9649 & -0.3989 \\ 0.1906 & -0.9770 & 0.0958 & 1.2277 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Code:

```
theta=[-90 70 160 300 100 -120];%degree
theta=[-50 100 0 120 150 20];%degree
```

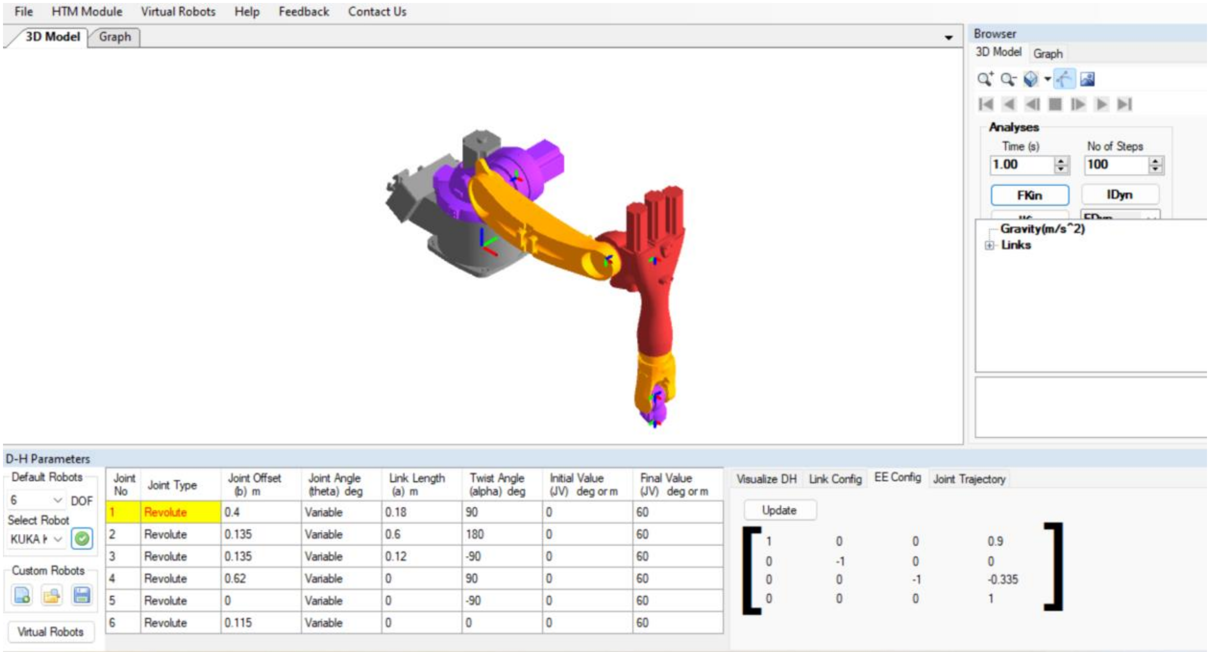
Comment:

3x1 translational part of the martix determines the position of the tool frame with respect to base frame.

Question 2: For Kuka KR 5 ARC robot manipulator:

a) Compare the DH table you find in Question 1 and DH table defined in RoboAnalyzer®

program. Explain, if exists, the reasons of differences.



The only difference in DH table, there is 135mm joint offset in 2. and 3. joints. However, it is not affect the table at all. And this is the same result in 1d.

b) Define an initial configuration for the joint parameters and calculate the position and

orientation of tip point with respect to base frame

i. in MATLAB using your forward kinematic model.

ii. in RoboAnalyzer® program.

Initial configuration of the robot is chosen as theta: [0 0 0 0 0 0].

Matlab result:

forward_kinematic =

```
    1.0000         0         0    0.9000
         0   -1.0000         0         0
         0         0   -1.0000   -0.3350
         0         0         0    1.0000
```

Analyzer result:

$$\begin{bmatrix} 1 & 0 & 0 & 0.9 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & -0.335 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

As seen above, result are the same for the initial configuration of the robot.

c) Assign two separate suitable trajectories starting from initial configuration for all joints considering the range of motion of the axes listed in the datasheet of the robot manipulator. And draw/plot tip point trajectories with respect to base frame for each case

i. in MATLAB using your forward kinematic model.

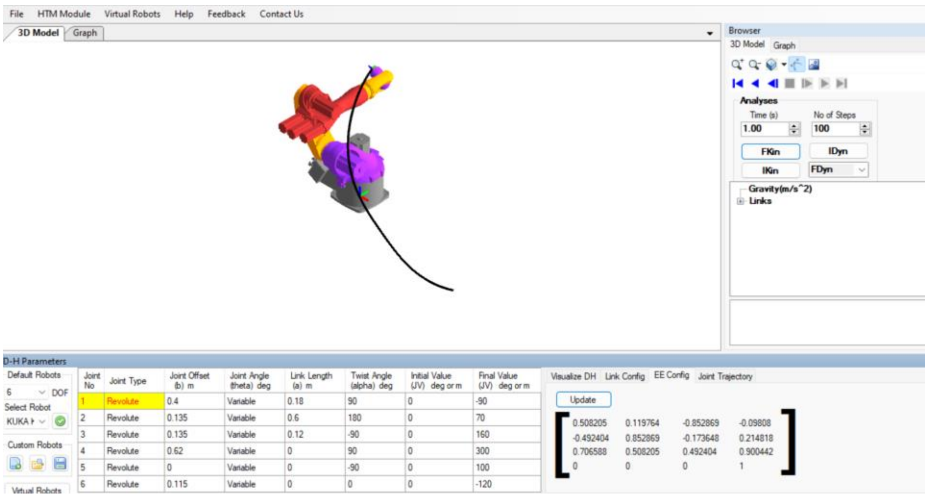
ii. in RoboAnalyzer® program via graph mode.

iii. Compare the graphs achieved in MATLAB and RoboAnalyzer® program.

The same trajectories are selected as in the first question.

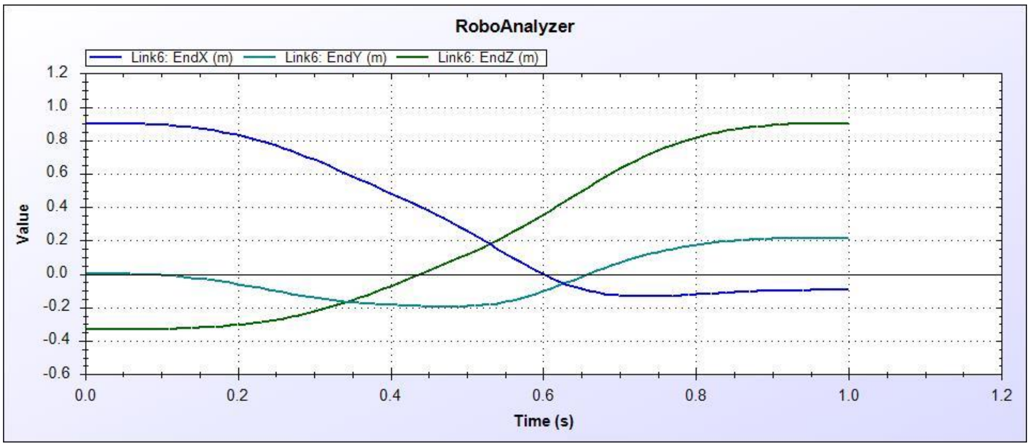
First : `theta=[-90 70 160 300 100 -120];%degree`
Second: `theta=[-50 100 0 120 150 20];%degree`

First, the trajectory of robot is obtained via Roboanalyzer

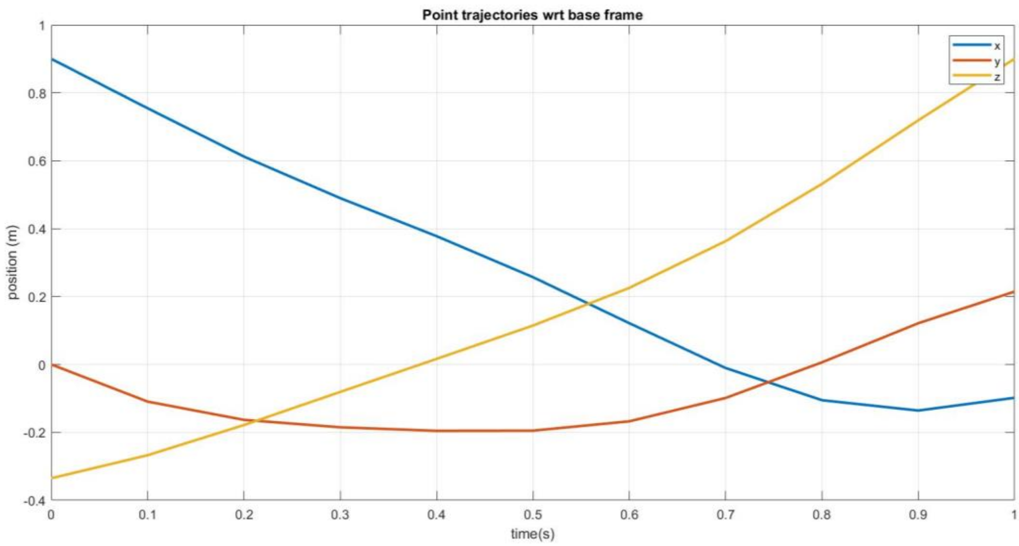


After that, the trajectory of the robot is plotted as x, y, and z axis.

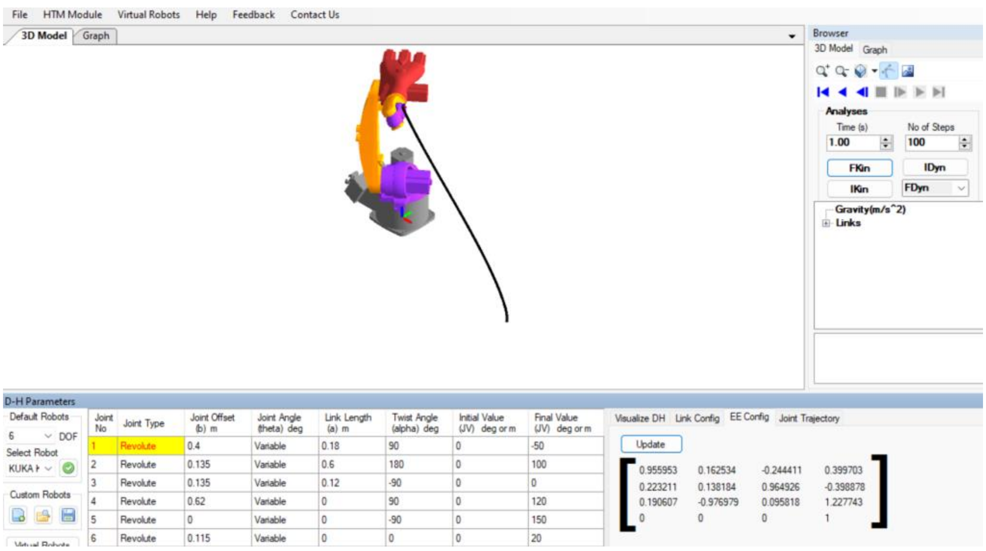
Roboanalyzer result:



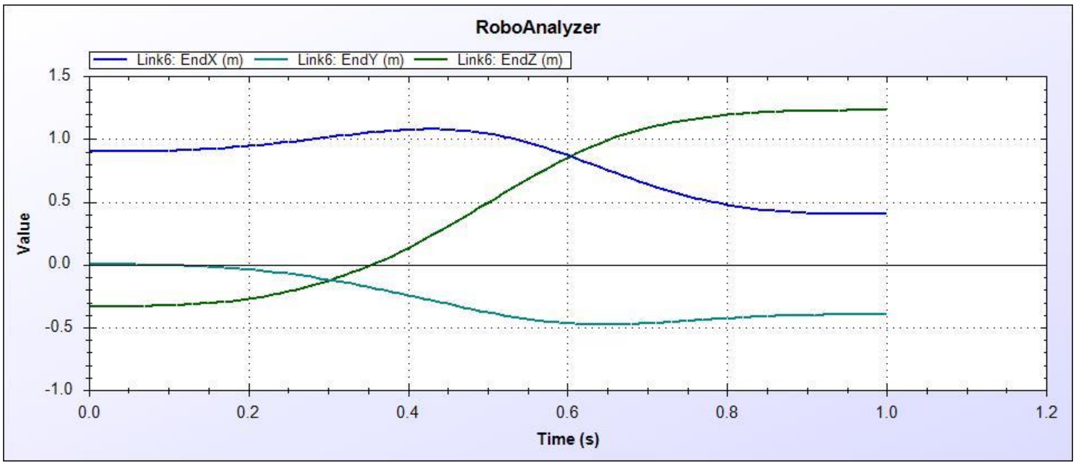
Matlab result:



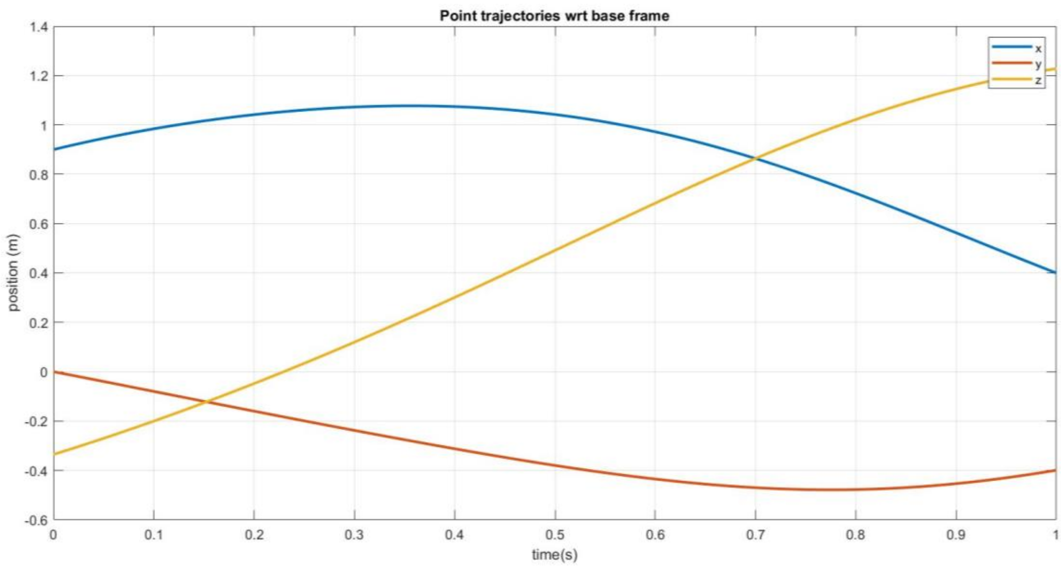
Second trajectory:



Roboanalyzer result:



Matlab result:



Although the initial and final values are correct, trajectories are seemed a bit different from each other, it is probably because of the velocity differences of the joint. I had not used the speed parameter in my code, so Matlab plots seem to reach faster than the analyzer.

Because DH tables are the same for both situation.

Full Code:

```
% syms th1 th2 th3 th4 th5 th6;
% theta=[th1 th2 th3 th4 th5 th6];
syms t;
a=[180 600 120 0 0 0]*0.001;%meter
alpha=[90 180 -90 90 -90 0];%degree
d=[400 0 0 620 0 115]*0.001;%meter
% theta=[0 0 0 0 0 0];%degree
% theta=[-90 70 160 300 100 -120];%degree
theta=[-50 100 0 120 150 20];%degree
% t=1;
mul=1;

for i=1:6%6 DOF
A=[cosd(theta(i)*t), -sind(theta(i)*t)*cosd(alpha(i)),
sind(theta(i)*t)*sind(alpha(i)), a(i)*cosd(theta(i)*t);
    sind(theta(i)*t), cosd(theta(i)*t)*cosd(alpha(i)), -
cosd(theta(i)*t)*sind(alpha(i)), a(i)*sind(theta(i)*t);
    0, sind(alpha(i)), cosd(alpha(i)), d(i);
    0, 0, 0, 1];
mul=mul*A;
end

forward_kinematic=mul;

x=forward_kinematic(1,4);
y=forward_kinematic(2,4);
z=forward_kinematic(3,4);
t1 = 0:0.01:1;
Tx = subs(x,t,t1);
plot(t1,Tx,'Linewidth',2);hold on
Ty = subs(y,t,t1);
plot(t1,Ty,'Linewidth',2);hold on
Tz = subs(z,t,t1);
plot(t1,Tz,'Linewidth',2);hold on;grid on
ylabel('position (m)');
xlabel('time(s)');
title('Point trajectories wrt base frame')
legend('x','y','z')
```