EE594-INDUSTRIAL ROBOTICS AND AUTOMATION ASSIGNMENT 1

FERNANDO P.D.R.
E/16/103
SEMESTER 7
29/04/2022

RHINO XR3

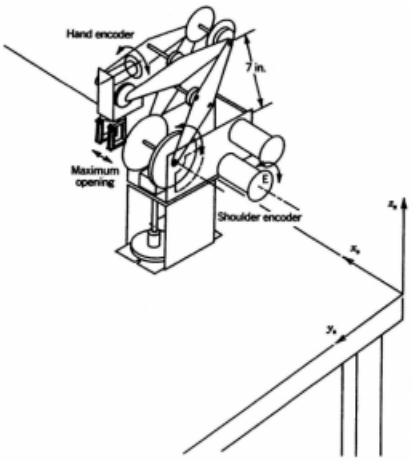
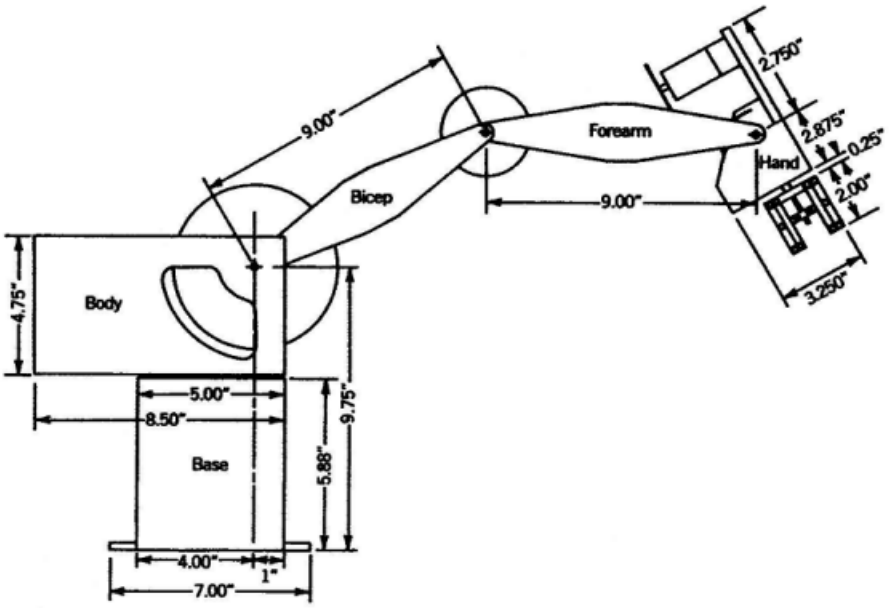


Fig. 01: RHINO Robot description



1. Obtain the DH parameters for the 5-link RHINO robot manipulator shown in the figures 01 and 02 (use SI units). Then create the Links and connect them serially to form the Robotic manipulator.

Link (i)	θ_i	d_i	a_i	α_i
1	Θ_1	0.24765	0	$\pi/2$
2	Θ_2	0	0.2286	0
3	Θ_3	0	0.2286	0
4	Θ_4	0	0	$-\pi/2$
5	Θ_5	0.079375	0	$\pi/2$

2. The home (idle) position (Fig. 03, H) and orientation of the manipulator end-effector is XYZ (- 0.146, 0, 0.409) and RPY (0, -90, -180) respectively. First, set the robotic manipulator to the home position

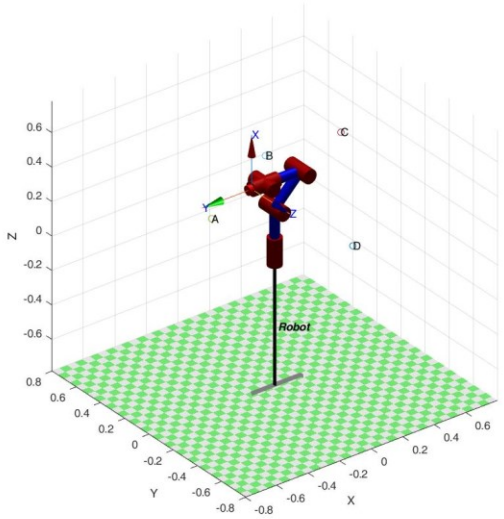


Figure 1 – Home position

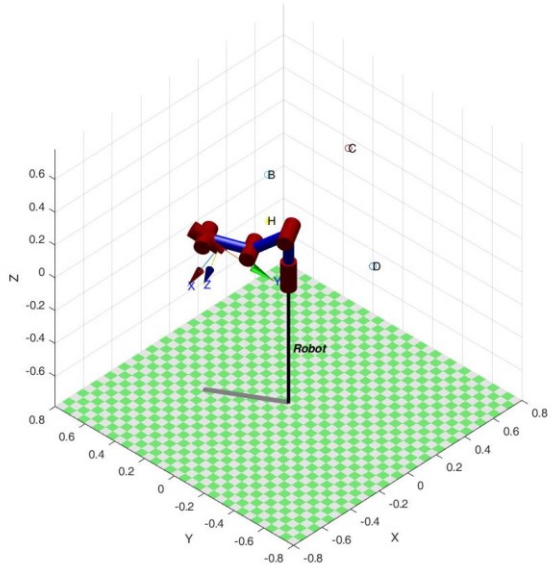


Figure 2 – A position

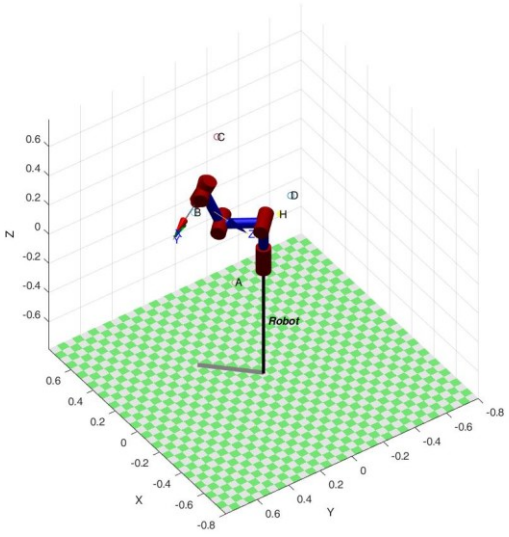


Figure 3 – B position

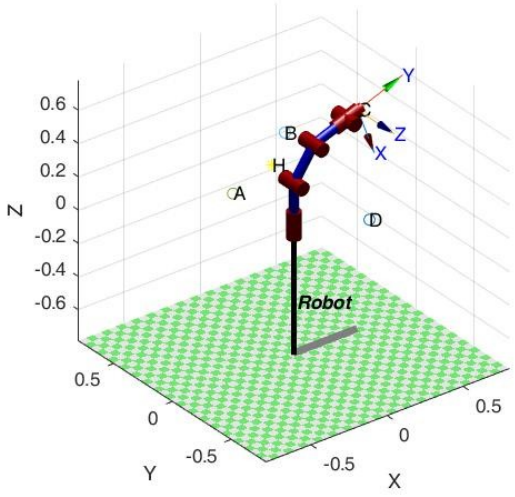


Figure 4 – C position

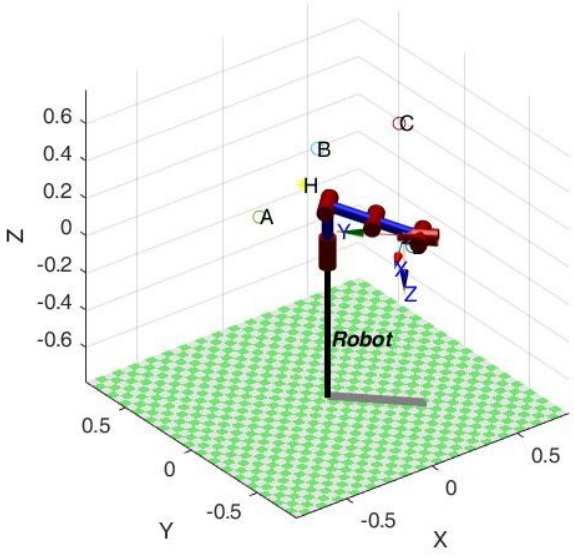


Figure 5 – D position

3. Generate trajectories (in joint space) to move the manipulator from H to A, then A to B, B to D via C and finally, back to H.

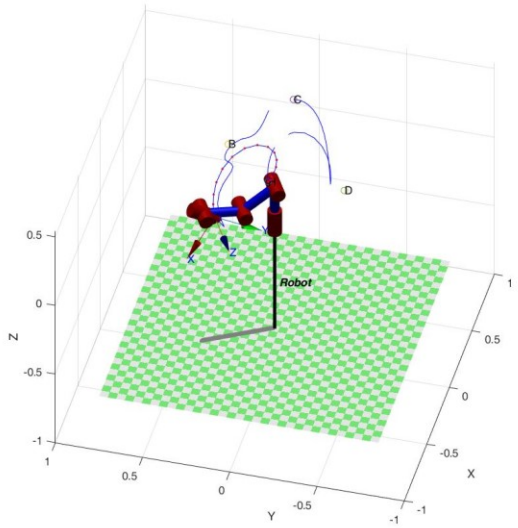


Figure 6 – H-A trajectory

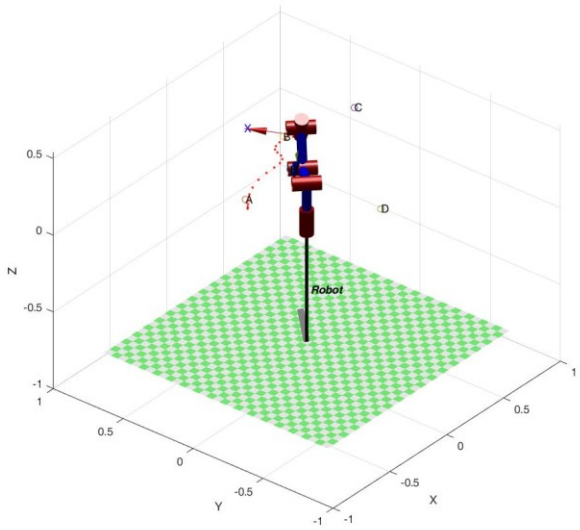


Figure 7 – A-B trajectory

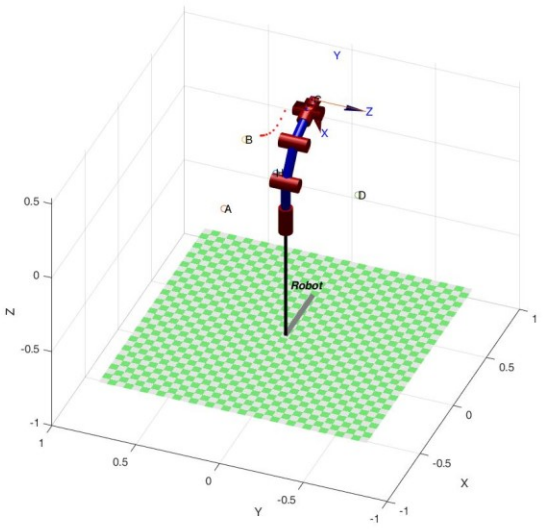


Figure 8 – B-C trajectory

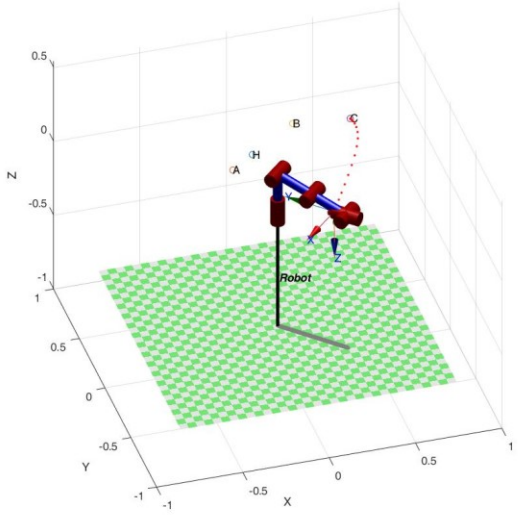


Figure 9 – C-D trajectory

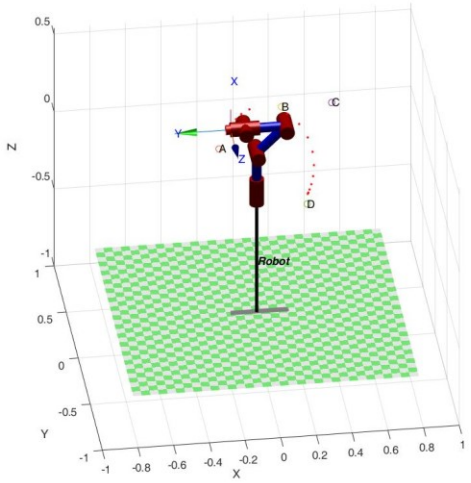


Figure 10 – D-H trajectory

4. Visualize the end-effector of the manipulator moving along the trajectories generated above. Save the figure frames when the end-effector at each of the following locations A, B, C, D and H. You can use the following Matlab code snippet for visualizing the Robot (stick diagram) movement.

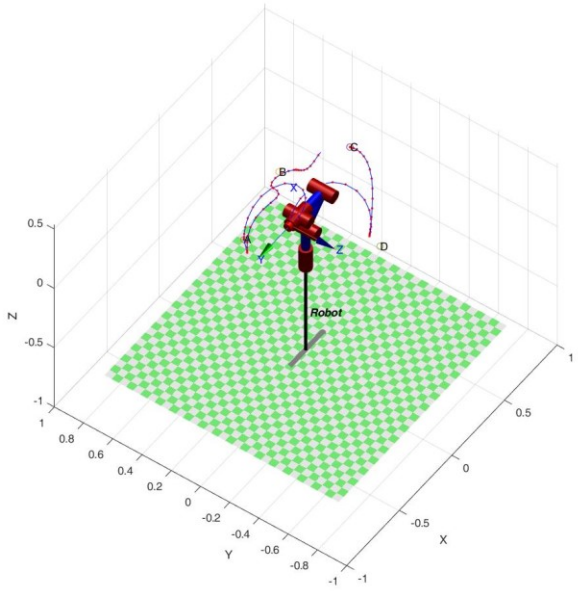


Figure 11 – H-A-B-C-D-H trajectory

MATLAB CODE

```
%link([theta     d        a      alpha])  connecting robot
arms
L(1)=Link([0      .24765    0        pi/2]);
L(2)=Link([0       0        .2286       0]);
L(3)=Link([0       0        .2286       0]);
L(4)=Link([0       0        0       -pi/2]);
L(5)=Link([0      .079375   0        pi/2]);
Robot=SerialLink(L);
Robot.name="Robot";
%F=Robot.fkine(qi)

%desired end effector ponts
H = SE3(-0.146, 0, 0.409) * SE3.rpy(0,-90,-180, 'deg');
%e number - 103 odd hence
Ay=0.3; Az=.01+103*0.4/420;
A = SE3(-0.17, Ay,Az) * SE3.rpy(-180,0,60, 'deg');
B = SE3(0.181, 0.313, 0.345) * SE3.rpy(-125, 26, 106,
'deg');
C = SE3(0.420, 0.000, 0.540) * SE3.rpy( 0, 70, 0, 'deg');
D = SE3(0.237, -0.338, 0.100) * SE3.rpy( 180, 0, -125,
'deg');

%time
t = [0:.1:2]';

%H - A joint angles configuration
qH=Robot.ikine(H,'mask',[1 1 1 0 1 1]);
qA=Robot.ikcon(A);
%trajectory H-A
q1 = jtraj(qH, qA, t);

%A-B straight line joint angles configuration
TAB = ctraj(A, B, length(t));
%trajectory A-B
q2 = Robot.ikunc(TAB);

%B-C-D joint angles configuration
qC=Robot.ikine(C,'mask',[1 1 1 0 1 1]);  %qC=Robot.ikunc(C);
qD=Robot.ikunc(D);
%trajectory B-C-D
q3 = jtraj(qC, qD, t);
q2_3 = jtraj(q2(end,:), qC, t);

%D-H trajectory
q4 = jtraj(qD, qH, t);
```

```
% TDH = ctraj(D, H, length(t));
% q4 = Robot.ikunc(TDH);
% q5 = jtraj(q4(end,:), qH, t);

%ploting points ABCDH
plot3(-0.146, 0, 0.409,'o');hold on;text(-0.146, 0,
0.409,'H');
plot3(-0.17, Ay,Az,'o');hold on;text(-0.17, Ay,Az,'A');
plot3(0.181, 0.313, 0.345,'o');hold on;text(0.181, 0.313,
0.345,'B');
plot3(0.420, 0.000, 0.540,'o');hold on;text(0.420, 0.000,
0.540,'C');
plot3(0.237, -0.338, 0.100,'o');hold on;text(0.237, -0.338,
0.100,'D');


q = [q1;q2;q2_3;q3;q4];

EET = zeros(4,4);

for i = 1:length(q)
 EET = Robot.fkine(q(i,:));
 EEp(i,:)=transl(EET);
 plot2(EEp(i,:),'r.');hold on;
 Robot.plot(q(i,:));hold on;
 plot2(EEp,'b');%pause(0.1);
 %saveas(gcf,['E:\sem 7\roboticcs\assignment\frame
2/filename' num2str(i) '.png'])
end


% Robot.plot(qC);hold on;plot3(-0.146, 0, 0.409,'y*');hold
on;text(-0.146, 0, 0.409,'H');
% plot3(-0.17, Ay,Az,'o');hold on;text(-0.17, Ay,Az,'A');
% plot3(0.181, 0.313, 0.345,'o');hold on;text(0.181, 0.313,
0.345,'B');
% plot3(0.420, 0.000, 0.540,'o');hold on;text(0.420, 0.000,
0.540,'C');
% plot3(0.237, -0.338, 0.100,'o');hold on;text(0.237, -
0.338, 0.100,'D')
```