```
% Optimization of two-link robot arm tracking
clear; clc;

% Define desired trajectory and Middle Points
qDes = [ -0.4986    2.5681;
          0.5371    1.5108 ];

qMid = [inverse_kinematics(0.4, 0.6, 1, 1), ...
        inverse_kinematics(0.4, 0.8, 1, 1), ...
        inverse_kinematics(0.4, 0.9, 1, 1), ...
        inverse_kinematics(0.4, 1.2, 1, 1)];

%  Parameters
time = [10 20];       % time
wn = 2;               % Prefilter Omega
kj = [40 25];         % Spring  [q1 q2]
bj = [10 30];         % Damping [q1 q2]
wt = [500, 0, 100];   % weights [qDes, Time, qMid]

% Optimization setup
initParams = [time  wn bj kj]; % Initial guess for [time, wn, bj, kj]

[init_T, init_Y] = ode45(@(t, x) myTwolinkwithprefilter(t, x, wn,
initParams(1:2), qDes, bj, kj),   [0 initParams(2)], zeros(8, 1));

% Lower and upper boundaries
lb = [0 0   1.5   10  10    2   2     ];   % Lower bounds
ub = [2 6   50    200 200   200 200 ];     % Upper bounds

% Objective Function
objectiveFunc = @(params) objectiveFunction(params, qDes, wt,qMid);

% Run optimization
options = optimset('Display', 'iter', 'TolFun', 1e-6, 'MaxIter', 200);
optimalParams = fmincon(objectiveFunc, initParams, [], [], [], [], lb, ub,
[], options);

% Simulate with optimal parameters and plot results
[t, y] = ode45(@(t, x) myTwolinkwithprefilter(t, x, optimalParams(3),
optimalParams(1:2), qDes, optimalParams(4:5), optimalParams(6:7)), [0
optimalParams(2)], zeros(8, 1));

% Output
xAct = forward_kinematics(y(:, 5), y(:, 6), 1, 1);
xDes = forward_kinematics(qDes(:, 1), qDes(:, 2), 1, 1);
xInit = forward_kinematics(init_Y(:, 5), init_Y(:, 6), 1, 1);

% Plotting
% Desired, Actual and Optimised Data
figure(1); hold on; grid on;
plot(xInit(:, 1), xInit(:, 2), '-');
plot(xAct(:, 1), xAct(:, 2), '-');
```

```
plot(xDes(:, 1), xDes(:, 2), 'o-');
plot(0.4,0.6, '*',0.4,0.8, '*',0.4,0.9, '*',0.4,1.2, '*');
xlabel('X axis'); ylabel('Y axis');
legend('Initial','Optimised', 'Desired');
title('Optimized Trajectory Tracking');
disp(['Optimized Parameters :', num2str(optimalParams)])

% Mid points in joint space
figure(2);plot(y(:,5),y(:,6),qMid(1,:),qMid(2,:),'o');
xlabel('Joint 1 position')
ylabel('Joint 2 position')

title('joint space of a (near) optimal staight line in cartesian space')

% joint space plot
figure(3); grid on; hold on;
plot(t,y(:,5:6));
xlabel('Time (s)')
ylabel('Position (rad)')
legend('Q1','Q2')
title('Joint position (rad)')

%  cartesian space plot
figure(4); hold on; grid on;
plot(xAct(:,1),xAct(:,2))
xlabel('X axis')
ylabel('Y axis')
legend('X','Y')
title('Cartesian Position (m)')

% x/y vs time
figure(5); grid on; hold on;
plot(t,xAct(:,1:2))
xlabel('Time (s)')
ylabel('Position')
legend('X','Y')
title('Cartesian Position vs Time')

% publish('simOpt.m','pdf');
% disp(sprintf('KY %s \t %s \t %s',mfilename,pwd,datetime("now")));

% Objective function
function error = objectiveFunction(params, qDes,wt,qMid)

    % Initial conditions
    x0 = zeros(8, 1);
    x0(1:2) = [qDes(1, 1); qDes(1, 2)];

    % Simulate the system
    [t, y] = ode45(@(t, x) myTwolinkwithprefilter(t, x, params(3),
params(1:2), qDes, params(4:5), params(6:7)), [0 params(2)], x0);

    % Calculate the error metric
    distto1 = min(sum((y(:, 5:6) - qDes(1,:)).^2,2) + sum((params(1) -
```

```
t).^2,2) );
    distto2 = min(sum((y(:, 5:6) - qDes(2,:)).^2,2) + sum((params(2) -
t).^2,2) );

    distMid1 = min(sum((y(:, 5:6) - qMid(:,1)').^2,2));
    distMid2 = min(sum((y(:, 5:6) - qMid(:,2)').^2,2));
    distMid3 = min(sum((y(:, 5:6) - qMid(:,3)').^2,2));
    distMid4 = min(sum((y(:, 5:6) - qMid(:,4)').^2,2));

    time1 = min(sum((params(1) - t).^2,2));
    time2 = min(sum((params(2) - t).^2,2));

    error   = wt(1) * distto1  + wt(1) * distto2  + ...  % Desired
              wt(2) * time1    + wt(2) * time2    + ...  % time
              wt(3) * distMid1 + wt(3) * distMid2 + ...  % Mid-point
              wt(3) * distMid3 + wt(3) * distMid4;       % Mid-point


    % distto5 = 5000 * sum((y(:, 5:6) - qMid3'),2) + w2 *
(sum(  (   (time(1) + (time(2) - time(1))/2 ) - t).^2   ,2));

end

% myTwolinkwithprefilter function
function dxdt = myTwolinkwithprefilter(t, x, wn, time, qDes, bj, kj)
    zeta = 1;
    A = [zeros([2 2]) eye(2); -eye(2)*wn^2 -eye(2)*2*zeta*wn];
    B = [0 0; 0 0; wn^2 0; 0 wn^2];

    % Actual position and velocity
    q = x(5:6);
    qd = x(7:8);
    q1p = x(7); q2p = x(8);
    q1 = x(5); q2 = x(6);

    % Robot constants
    L_1 = 1; L_2 = 1; m_1 = 1; m_2 = 1;
    ka = L_2^2 * m_2;
    kb = L_1 * L_2 * m_2;
    kc = L_1^2 * (m_1 + m_2);

    M = [ka + 2*kb*cos(q2) + kc, ka + kb*cos(q2);
         ka + kb*cos(q2), ka];
    V = ka*sin(q2)*([0 -1; 1 0] * [q1p^2; q2p^2] + [-2*q1p*q2p; 0]);

    Numerator = V + [-bj(1) 0; 0 -bj(2)]*qd + [-kj(1) 0; 0 -kj(2)]*(q -
x(1:2));
    qdd = M\Numerator;
    if t < time(1)
        dotx = A*x(1:4) + B*qDes(1, :)';
    else
        dotx = A*x(1:4) + B*qDes(2, :)';
    end
```

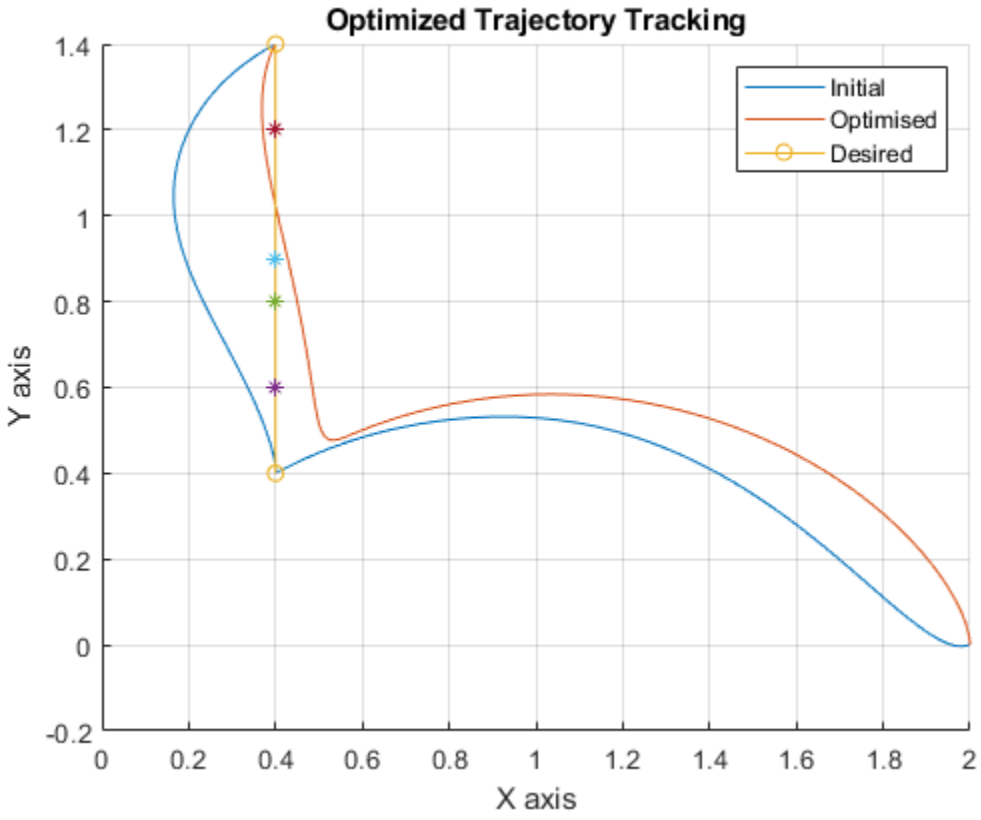
```
    dxdt = [dotx; qd; qdd];
end
```

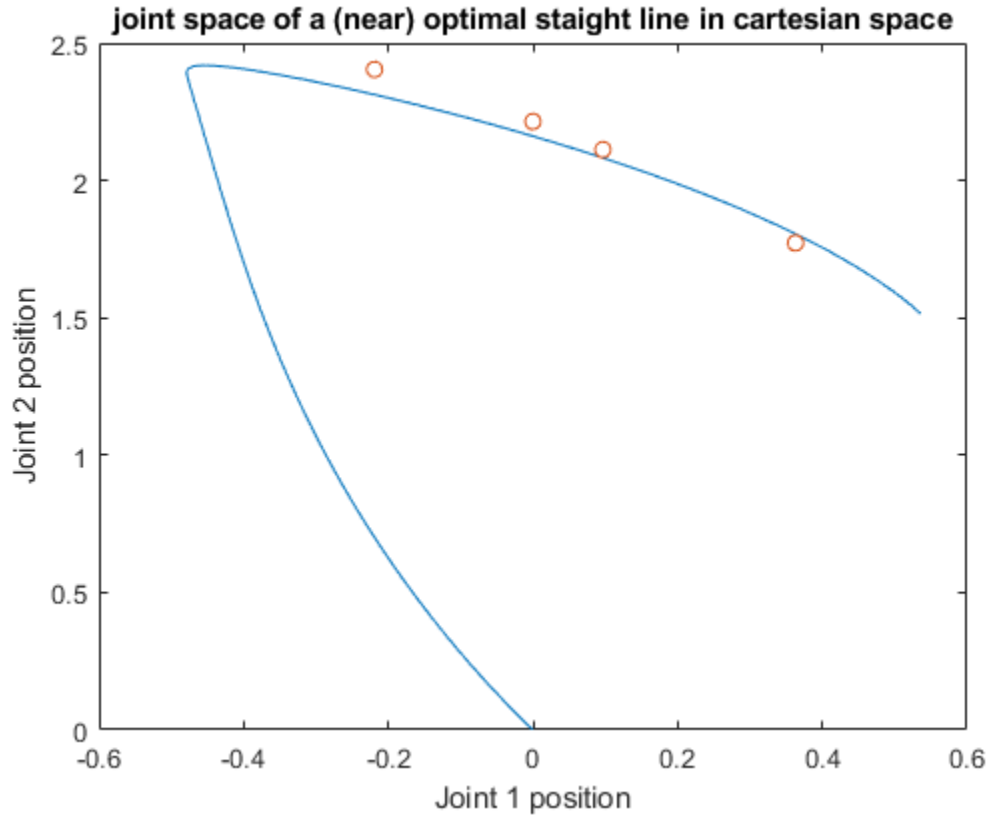
Initial point X0 is not between bounds LB and UB;
FMINCON shifted X0 to strictly satisfy the bounds.

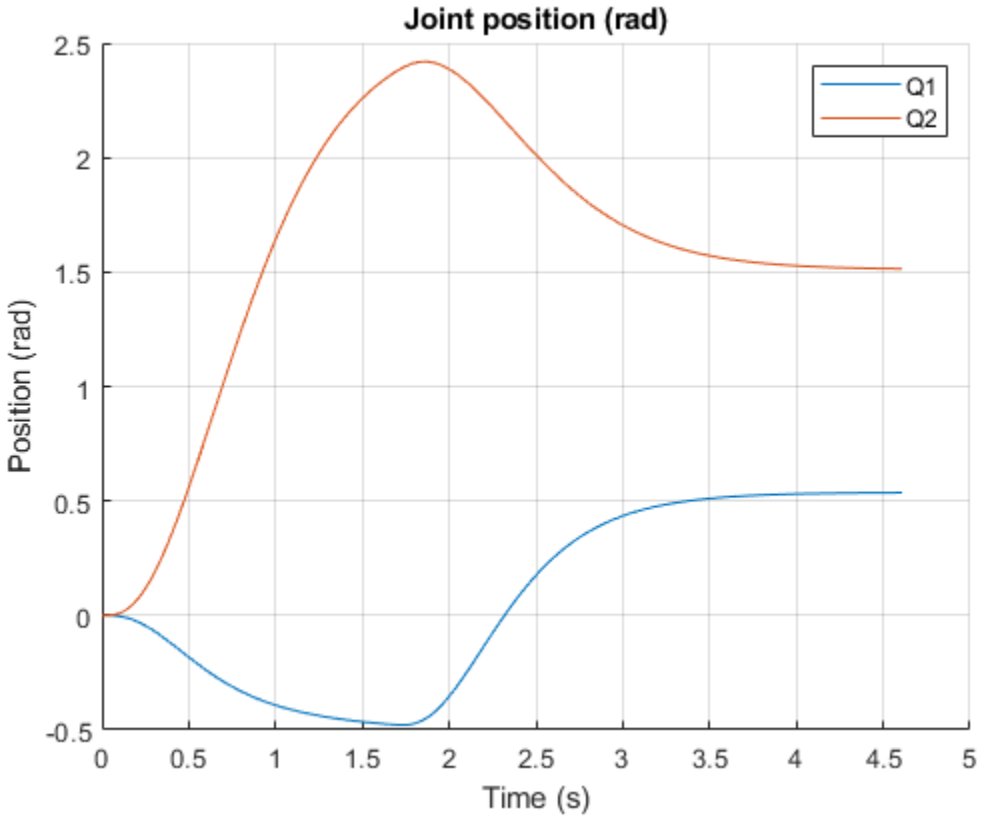
Iter	F-count	f(x)	Feasibility	First-order optimality	Norm of step
0	8	4.057938e+02	0.000e+00	2.252e+02	
1	16	3.720545e+00	0.000e+00	1.522e+02	2.605e+01
2	24	3.236958e+00	0.000e+00	2.004e+01	5.247e-01
3	32	7.564503e-01	0.000e+00	1.768e+01	4.969e+00
4	41	5.031059e-01	0.000e+00	8.680e+00	2.441e+00
5	52	4.512967e-01	0.000e+00	7.411e+00	1.573e+00
6	60	4.635546e-01	0.000e+00	7.381e-01	3.678e-01
7	69	6.760858e-01	0.000e+00	2.059e-01	2.580e+00
8	77	7.176896e-01	0.000e+00	2.407e-01	7.124e-01
9	86	7.702614e-01	0.000e+00	1.041e+00	1.069e+00
10	94	6.435921e-01	0.000e+00	7.390e-01	1.339e+00
11	105	6.097480e-01	0.000e+00	8.369e-01	4.097e-01
12	118	5.603979e-01	0.000e+00	6.456e-01	3.102e-01
13	128	5.173068e-01	0.000e+00	2.178e-01	2.507e-01
14	139	5.132995e-01	0.000e+00	3.291e-01	1.652e-02
15	147	5.179078e-01	0.000e+00	4.483e-01	5.092e-01
16	155	5.743428e-01	0.000e+00	1.408e-01	9.111e-01
17	163	6.034995e-01	0.000e+00	2.602e-01	7.010e-01
18	171	6.001572e-01	0.000e+00	5.865e-01	5.147e-01
19	179	5.769701e-01	0.000e+00	4.674e-01	1.725e-01
20	187	5.333313e-01	0.000e+00	4.631e-01	9.470e-01
21	196	5.134713e-01	0.000e+00	2.425e-01	3.419e+00
22	204	4.927055e-01	0.000e+00	2.364e-01	9.332e-01
23	213	5.129319e-01	0.000e+00	2.070e-01	2.332e+00
24	221	6.337309e-01	0.000e+00	7.097e-01	7.578e+00
25	229	5.672040e-01	0.000e+00	3.617e-01	1.910e+00
26	238	5.559809e-01	0.000e+00	3.531e-01	7.306e-02
27	256	5.531331e-01	0.000e+00	3.188e-01	2.025e-03
28	273	5.528946e-01	0.000e+00	3.170e-01	4.288e-04
29	281	5.104415e-01	0.000e+00	2.064e-01	2.114e+00
30	291	5.124037e-01	0.000e+00	4.056e-01	4.334e-01

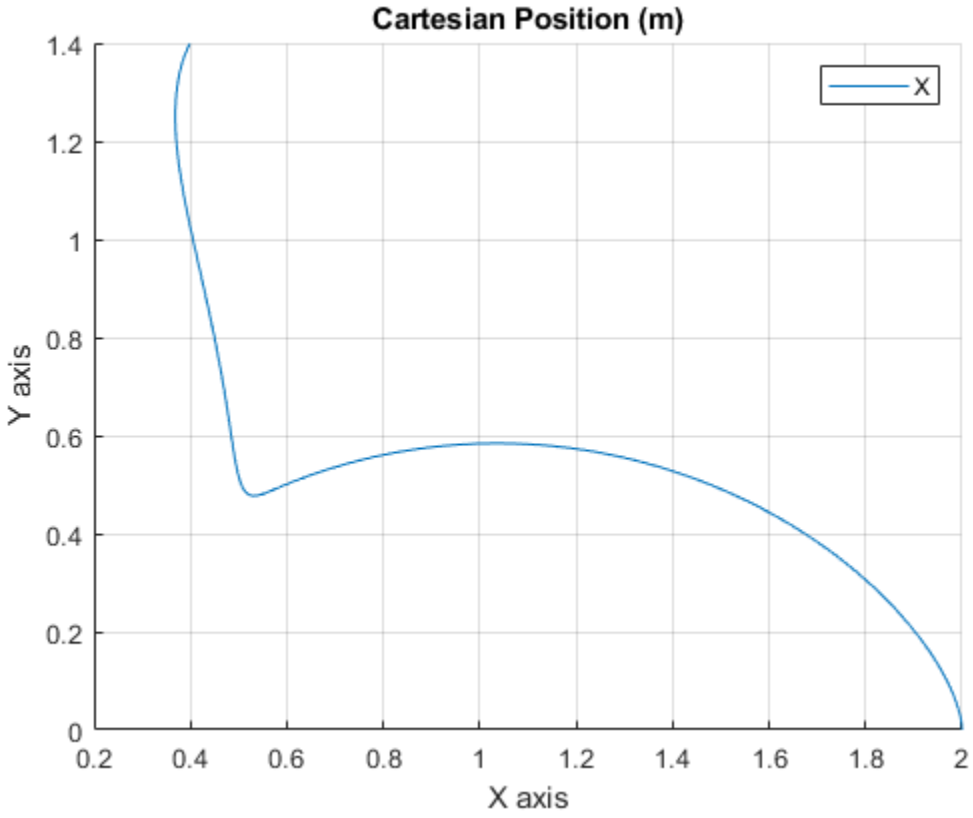
Iter	F-count	f(x)	Feasibility	First-order optimality	Norm of step
31	303	5.161816e-01	0.000e+00	4.740e-01	8.663e-01
32	311	4.508732e-01	0.000e+00	2.798e-01	3.562e+00
33	319	4.563707e-01	0.000e+00	4.922e-01	3.661e+00
34	335	4.455275e-01	0.000e+00	2.452e-01	1.463e-01
35	343	4.453473e-01	0.000e+00	5.491e-01	4.843e+00
36	351	4.654540e-01	0.000e+00	4.583e-01	2.485e+01
37	359	4.199834e-01	0.000e+00	5.555e-01	5.047e-01
38	368	2.921708e-01	0.000e+00	3.963e-01	3.051e+00
39	376	2.414759e-01	0.000e+00	2.711e-01	5.602e+00
40	384	2.219708e-01	0.000e+00	2.287e-01	1.699e+01
41	392	2.792205e-01	0.000e+00	4.897e-01	4.671e+01
42	400	2.535983e-01	0.000e+00	5.193e-01	2.831e+01

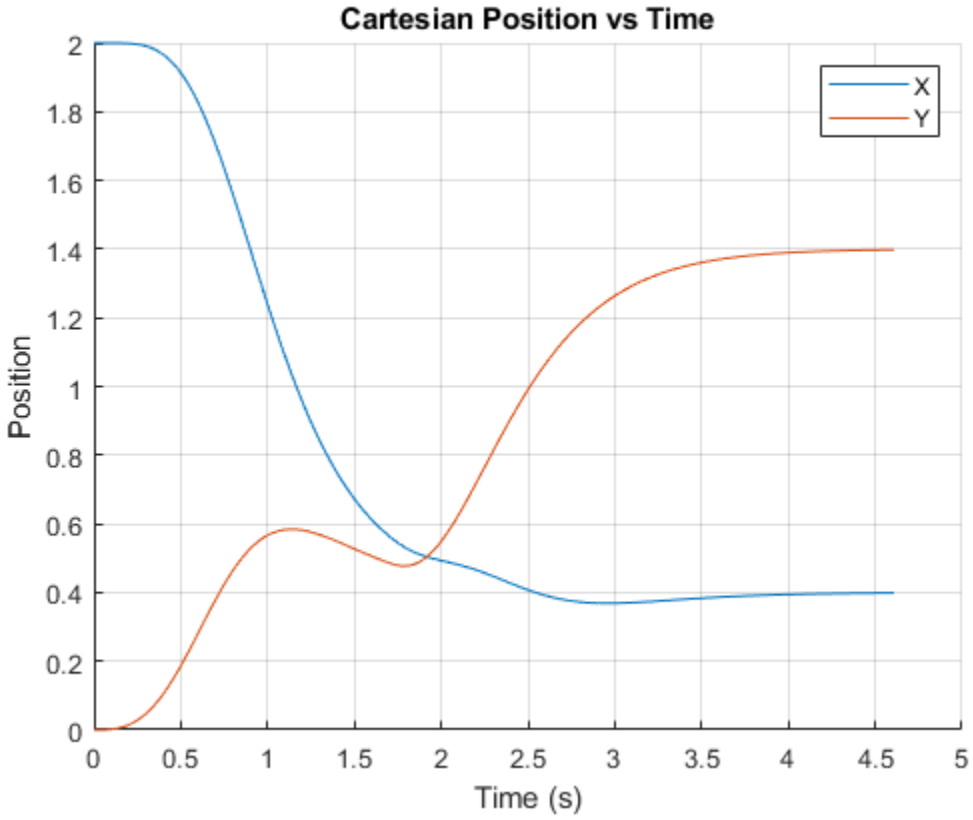
```
    43      409     2.253164e-01    0.000e+00    1.835e-01    4.222e+00
    44      417     1.788021e-01    0.000e+00    2.368e-01    6.117e+00
    45      427     1.669383e-01    0.000e+00    8.989e-02    8.850e+00
    46      440     1.627801e-01    0.000e+00    1.637e-01    1.690e-02
    47      467     1.627789e-01    0.000e+00    1.663e-01    6.751e-06
    48      485     1.627788e-01    0.000e+00    1.685e-01    7.383e-07
    49      503     1.627788e-01    0.000e+00    1.377e+05    8.082e-08
    50      515     1.627788e-01    0.000e+00    1.377e+05    1.206e-08

Local minimum possible. Constraints satisfied.

fmincon stopped because the size of the current step is less than
the value of the step size tolerance and constraints are
satisfied to within the value of the constraint tolerance.

Optimized Parameters :1.657855      4.608777       3.38994
28.27638      19.54304      151.5962      60.90622
Warning: Ignoring extra legend entries.
```




joint space of a (near) optimal staight line in cartesian space
Joint 2 position
Joint 1 position
2.5
2
1.5
1
0.5
0
-0.6
-0.4
-0.2
0
0.2
0.4
0.6


Joint position (rad)
Position (rad)
Time (s)
Q1
Q2
2.5
2
1.5
1
0.5
0
-0.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5


Cartesian Position (m)
X
Y axis
X axis
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2


Cartesian Position vs Time
X
Y
Position
Time (s)
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

Published with MATLAB® R2023b