







**Code:**

function LinHamSolv(N,q,r,p,h,Initial,Target)

%Check stage conditions

if length(h) == 1

h = h\*ones(N,1);

else

h = h(:);

end

%Create constraint and cost matrices

xu = [[-.5\*eye(N),zeros(N,1)] + [zeros(N,1),eye(N)],...

-r\*eye(N),...

zeros(N,N+1)];

xlam = [[-2\*q\*eye(N),zeros(N,1)]...

zeros(N),...

[eye(N),zeros(N,1)] + [zeros(N,1),-.5\*eye(N)]];

ulam = [zeros(N,N+1),...

r\*eye(N),...

[zeros(N,1),eye(N)]];

bound = [[1,zeros(1,N\*3+1)];...

[zeros(1,N),-2\*p, zeros(1,2\*N), 1]];

%Paste together

A = [xu ; xlam ; ulam ; bound];

%Define b vector

B = [zeros(N,1); -2\*h\*q ; zeros(N,1) ; [Initial ; -2\*p\*Target]];

%Solve

output = A^-1\*B;

%Plot

color = {'-b+','-ro','--b\*','--rx',':bd',':rs','-.bv','-.r^'};

figure;

title('\fontsize{16} Optimal Control and Trajectory, Case (i)');

xlabel('\fontsize{13} Timestep');

ylabel('\fontsize{13} Value of Position, Control');

hold on

plot([0:8], output(1:9), color{1},...

[0:7], output(10:17), color{2},...

8,output(9), color{3}, 'MarkerSize',10);

legend('\fontsize{13} Trajectory',...

'\fontsize{13} Control',...

['\fontsize{13} Final Pos: ' num2str((round(output(9)\*100))/100)],...

'Location','Best');

hold off