Cleveland State University

Optimal Control Homework 11

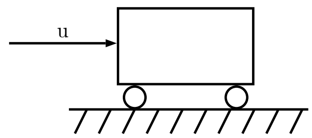
Student Name:
Poya Khalaf

Student ID:
2625396

November 23, 2015

We found $E(J)$ for a stochastic LQR problem with cost function J. Define an LQR problem (maybe an RC circuit, or a two-state Newtonian system, or a linearized version of your project problem). Implement stochastic LQR control. Run your system many times (maybe 100 times or so) and verify that the numerical value for $E(J)$ matches the analytical expression for $E(J)$.

A simple cart system is chosen for this problem. the dynamic equations for this system are :



$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\dot{x} = Ax + Bu + w$$

where $x = [x_1, x_2]^T$ is the state vector. x_1 and x_2 are the position and velocity respectively. The control u is the acceleration. The noise vector $w = [w_1, w_2]^T$ is chosen to be white noise with mean of zero and covariance matrix v

$$w \sim (0, v)$$

Also the initial condition $x(0)$ is chosen to be white noise with the covariance matrix p_0:

$$x(0) \sim (0, p_0)$$

where v and p_0 are chosen to be:

$$v = \begin{bmatrix} 10^{-5} & 0 \\ 0 & 10^{-5} \end{bmatrix}$$

$$p_0 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$

in order to simulate the system the covariance matrix v is scaled by the time step Δt:

$$\frac{v}{\Delta t}$$

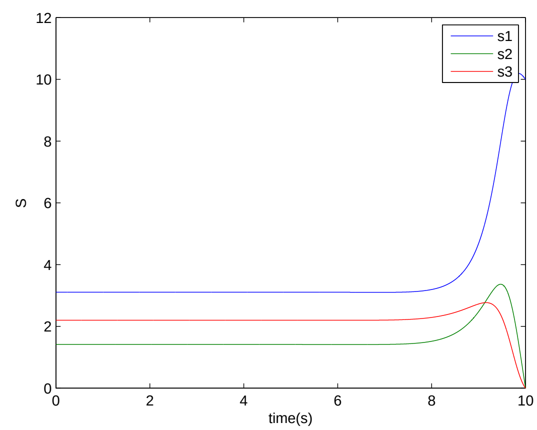
The following code has been written for this problem:

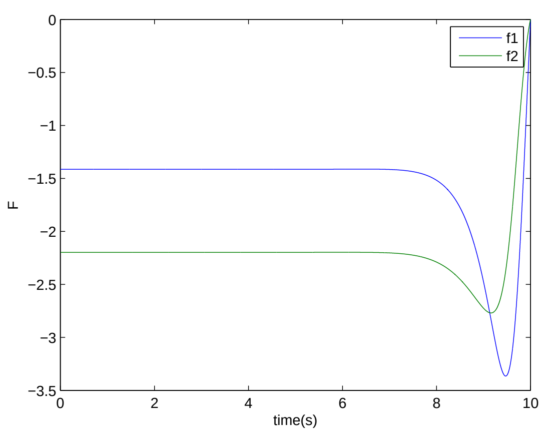
```
clc
clear
close all


%state equations
A=[0 1;0 0];
B=[0;1];

%Cost function
Q=[2 0;0 2];
R=1;
invR=1; %R^-1
H=[10 0;0 0];


dt = 0.001; % Integration step size
tf = 10; % Simulation length
t=0:dt:tf;
x0 = [0;0]; % Initial state
N = round(tf/dt) + 1; %Number of steps

```

```
S=zeros(2,2,N);
S(:,:,N)=H; %final conditions for S
F(N,:)=-invR*B'*S(:,:,N); % F=-R^-1*B'*S

%Backwards integration in time
for i = 1 : N-1
Sdot=-Q+S(:,:,N-i+1)*B*invR*B'*S(:,:,N-i+1)-S(:,:,N-i+1)*A-A'*S(:,:,N-i+1); % Riccati ...
    equation
S(:,:,N-i)=S(:,:,N-i+1)-Sdot*dt;
F(N-i,:)=-invR*B'*S(:,:,N-i);
end

%plot
plot(t,permute(S(1,1,:),[3,2,1]),t,permute(S(2,1,:),[3,2,1]),t,permute(S(2,2,:),[3,2,1]))
xlabel('time(s)')
ylabel('S')
legend('s1','s2','s3')

figure
plot(t,F(:,1),t,F(:,2))
xlabel('time(s)')
ylabel('F')
legend('f1','f2')


runs=100; %number of runs
J=zeros(runs,1);

for k=1:runs;
disp(['#',num2str(k)]) %diplay run number
v=[1e-5,0;0,1e-5]; %covariance matrix
%v=zeros(2);
p0=[0.1,0;0,0.1];  %covariance matrix for initial conditions
x=x0+(p0).^0.5*randn(2,1);
%x=x0;
xArray = zeros(2, N);
uArray = zeros(1, N);
wArray = zeros(2, N);
%System simulation
for i=1:N
w=(v/dt).^0.5*randn(2,1);
u=F(i,:)*x;
xdot=A*x+B*u+w;
xArray(:,i)=x;
uArray(i)=u;
wArray(:,i)=w;
x=x+xdot*dt;
end

% compute cost
integrand=zeros(1,N-1);
for j=1:N-1
integrand(j)=xArray(:,j)'*Q*xArray(:,j)+uArray(j)*R*uArray(j);
end

J(k)=xArray(:,end)'*H*xArray(:,end)+trapz(t(1:end-1),integrand);

end


disp(['Average of numerical cost over ',num2str(k),' runs'])
mean(J)

%compute analytical cost
integrand2=zeros(1,N);
for j=1:N
integrand2(j)=trace(v*S(:,:,j));
end
```

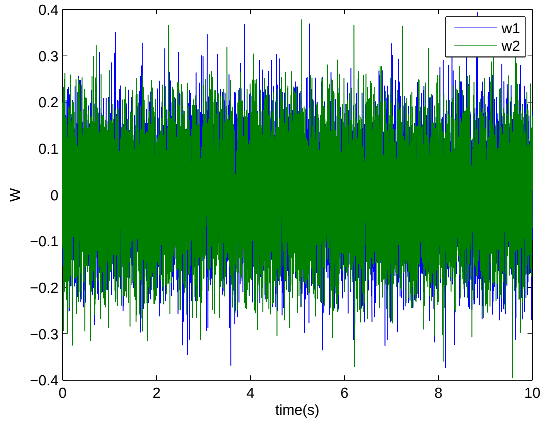
```
disp('Analytical cost')
J2=trace(S(:,:,1)*p0)+trapz(t(2:end),integrand2(2:end))


%plot
figure
plot(t,xArray(1,:),t,xArray(2,:))
xlabel('time(s)')
ylabel('x')
legend('x1','x2')


figure
plot(t,uArray)
xlabel('time(s)')
ylabel('u')

figure
plot(t,wArray(1,:),t,wArray(2,:))
xlabel('time(s)')
ylabel('W')
legend('w1','w2')
```

The following figures show the the components of the S matrix and the F vector:

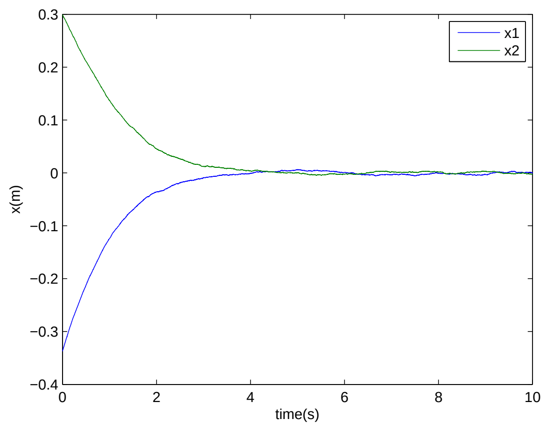


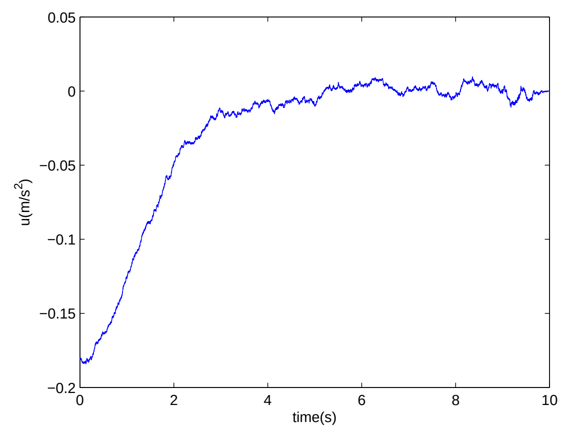


the following figure shows the components of the noise vector w:



the flowing figures show the optimal state trajectory and control for a sample run:





The final results are as follows:

```
Average of numerical cost over 100 runs
ans =
       0.54327
Analytical cost
J2 =
       0.53108
```