Mehmet KILIÇ - 498
Term Project - Part1

Table of Contents

System Specs.

```
T=1e-1;
M=0.5; %kg
m=0.2; %kg
b=0.1; %N/m/sec
I=0.006; %kg.^2
g=9.8;
l=0.3; %m
```

From Linearized CTSS to DTSS Model

```
denum=(I*(M+m)+M*m*(l^2));
A22=-((I+m*(l^2))*b)/denum;
A23=(m^2*g*l^2)/denum;
A42=-m*l*b/denum;
B21=(I+m*(l^2))/denum;
B41=m*l/denum;
A43=m*l*b/denum;
Ac=[0 1 0 0;
    0 A22 A23 0;
    0 0 0 1;
    0 A42 A43 0];
Bc=[0;B21;0;B41];
Cc=[1 0 0 0;
    0 0 1 0];
Dc=[0;0];
state={'x' 'x_dot' 'theta' 'theta_dot'};
input={'u'};
output={'x';'theta'};
ctss=ss(Ac,Bc,Cc,Dc,'inputname',input,'statename',state,'outputname',output);
dtss=c2d(ctss,T,'foh');

A=dtss.A;
B=dtss.B;
C=dtss.C;
```

Plant Properties

```
stability=eig(A);
controllability=rank(ctrb(A,B));
observability=rank(obsv(A,C));
% Plant is stable, controllable and observable!

[num,den]=ss2tf(Ac,Bc,Cc,Dc);
tf1=tf(num(1,:),den);
tf2=tf(num(2,:),den);
```
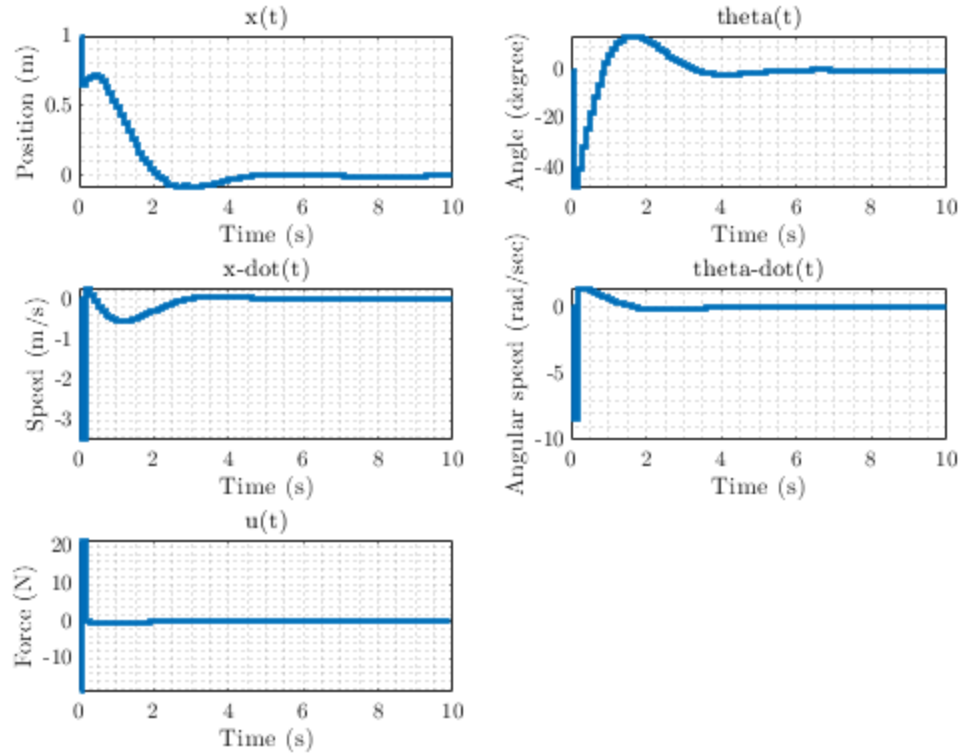
LQR Design - Finite Horizon

```
N=100;
Q=1000*(C'*C);
Qf=Q;
R=0.01;
x0=[1;0;0;0];
x=[x0;zeros((4*N-4),1)];
x1=[x0(1,1);zeros(N,1)];
x2=[x0(2,1);zeros(N,1)];
x3=[x0(3,1);zeros(N,1)];
x4=[x0(4,1);zeros(N,1)];
u=zeros((N-1),1);
P=zeros(4*(N+1),4);
P(4*N+1:4*N+4,1:4)=Qf;
K=zeros(N,4);
for i=N:-1:1
    Pp=P((4*i+1):(4*(i)+4),1:4);
    P((4*i-3):(4*i),1:4)=A'*Pp*A+Q-A'*Pp*B*((B'*Pp*B)^-1)*B'*Pp*A;
end

for i=1:1:N
    Pp=P(4*i+1:4*i+4,:);
    K(i,:)=((R+B'*Pp*B)^-1)*B'*Pp*A;
    u(i,1)=-K(i,:)*x(4*i-3:4*i,:);
    x(4*i+1:4*i+4,:)=A*x(4*i-3:4*i,:)+B*u(i,1);
    x1(i+1,1)=x(4*i+1,:);
    x2(i+1,1)=x(4*i+2,:);
    x3(i+1,1)=x(4*i+3,:);
    x4(i+1,1)=x(4*i+4,:);

end
time=linspace(0,N*T,N+1);
figure;
subplot(3,2,5)
stairs(time(1,1:N),u,'LineWidth',2);
title("u(t)");
xlim([0 N*T]);
xlabel("Time (s)");
ylabel("Force (N)");
grid minor
```

```matlab
subplot(3,2,1)
stairs(time,x1,'LineWidth',2);
title("x(t)");
xlabel("Time (s)");
xlim([0 N*T]);
ylabel("Position (m)");
grid minor

subplot(3,2,3)
stairs(time,x2,'LineWidth',2);
title('x-dot(t)');
xlabel("Time (s)");
ylabel("Speed (m/s)");
xlim([0 N*T]);
grid minor

subplot(3,2,2)
stairs(time,rad2deg(x3),'LineWidth',2);
title('theta(t)');
xlim([0 N*T]);
xlabel("Time (s)");
ylabel("Angle (degree)");
grid minor

subplot(3,2,4)
stairs(time,x4,'LineWidth',2);
title("theta-dot(t)");
xlabel("Time (s)");
xlim([0 N*T]);
ylabel("Angular speed (rad/sec)");
grid minor
```

LQR Design - Infinite Horizon

```
N=100;
Q=10*(C'*C);
Qf=Q;
R=0.1;
x0=[1;0;0;0];
x=[x0;zeros((4*N-4),1)];
x1=[x0(1,1);zeros(N,1)];
x2=[x0(2,1);zeros(N,1)];
x3=[x0(3,1);zeros(N,1)];
x4=[x0(4,1);zeros(N,1)];
u=zeros((N-1),1);
[Pinf]=idare(A,B,Q,R);
Kinf=((R+(B')*Pinf*B)^-1)*B'*Pinf*A;
for i=1:1:N
    u(i,1)=-Kinf*x(4*i-3:4*i,:);
    x(4*i+1:4*i+4,:)=A*x(4*i-3:4*i,:)+B*u(i,1);
    x1(i+1,1)=x(4*i+1,:);
    x2(i+1,1)=x(4*i+2,:);
    x3(i+1,1)=x(4*i+3,:);
    x4(i+1,1)=x(4*i+4,:);
end
time=linspace(0,N*T,N+1);
figure;
```
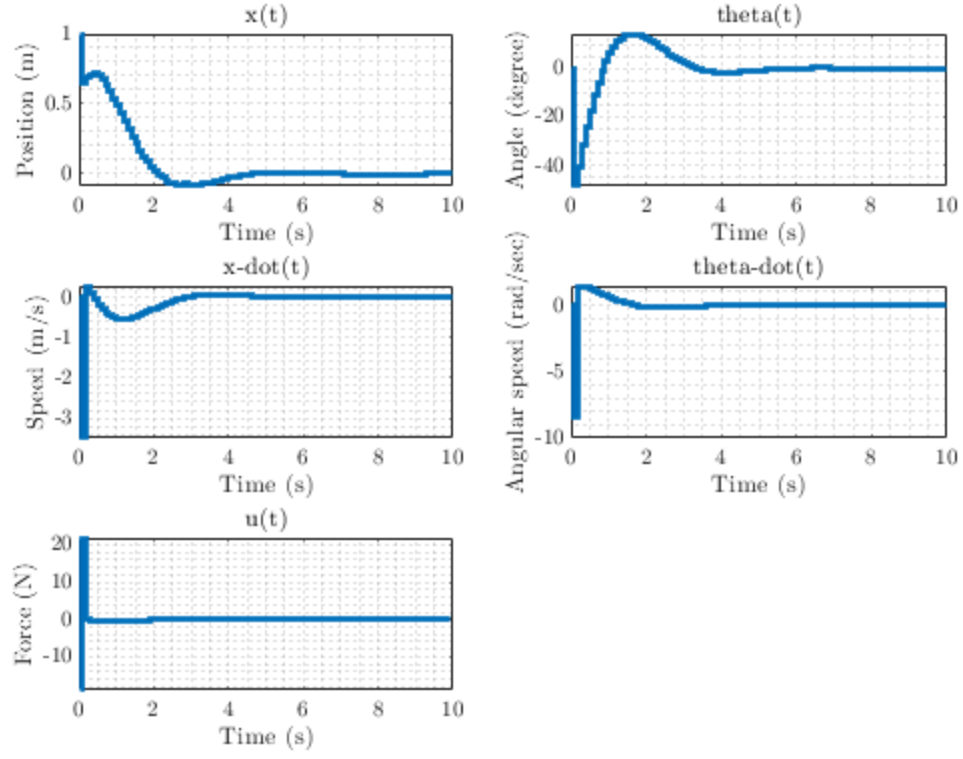
```matlab
sgtitle('Infinite Horizon LQR Solution')
subplot(3,2,5)
stairs(time(1,1:N),u,'LineWidth',2);
title("u(t)");
xlim([0 N*T]);
xlabel("Time (s)");
ylabel("Force (N)");
grid minor

subplot(3,2,1)
stairs(time,x1,'LineWidth',2);
title("x(t)");
xlabel("Time (s)");
xlim([0 N*T]);
ylabel("Position (m)");
grid minor

subplot(3,2,3)
stairs(time,x2,'LineWidth',2);
title('x-dot(t)');
xlabel("Time (s)");
ylabel("Speed (m/s)");
xlim([0 N*T]);
grid minor

subplot(3,2,2)
stairs(time,rad2deg(x3),'LineWidth',2);
title('theta(t)');
xlim([0 N*T]);
xlabel("Time (s)");
ylabel("Angle (degree)");
grid minor

subplot(3,2,4)
stairs(time,x4,'LineWidth',2);
title("theta-dot(t)");
xlabel("Time (s)");
xlim([0 N*T]);
ylabel("Angular speed (rad/sec)");
grid minor
```
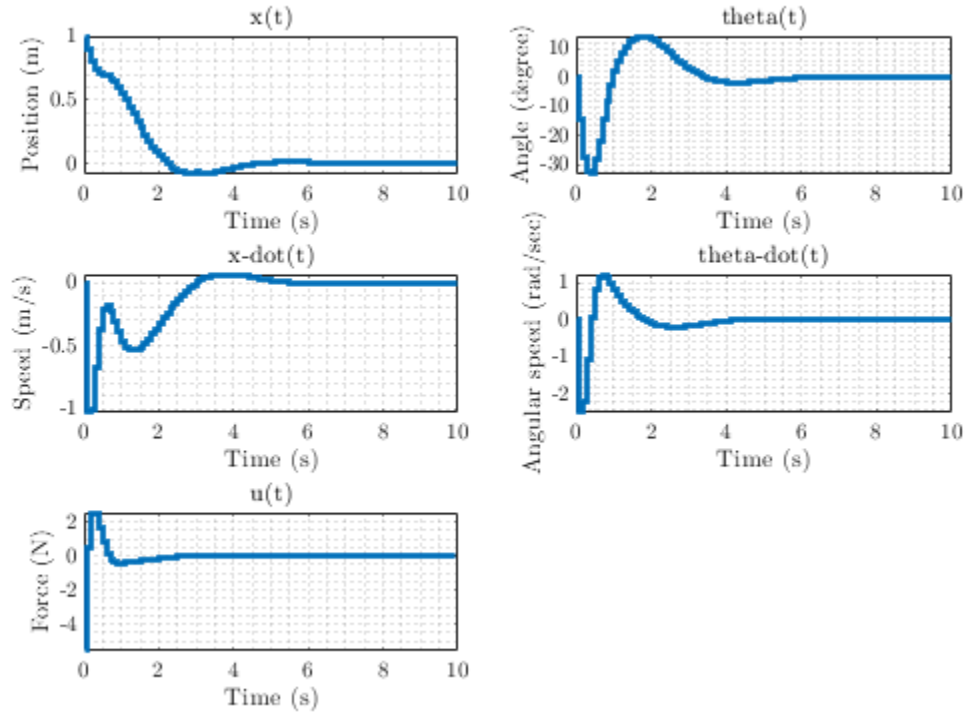
Infinite Horizon LQR Solution

Published with MATLAB® R2020b