現代控制理論 HW5

104303206 黃筱晴

a.

x_1_dot=x_2

x_2_dot=$x_1^3x_2+9x_1x_2^3+(x_1^2+x_2^2)u$

令f=$x_1+\lambda\ x_2$

則f_dot

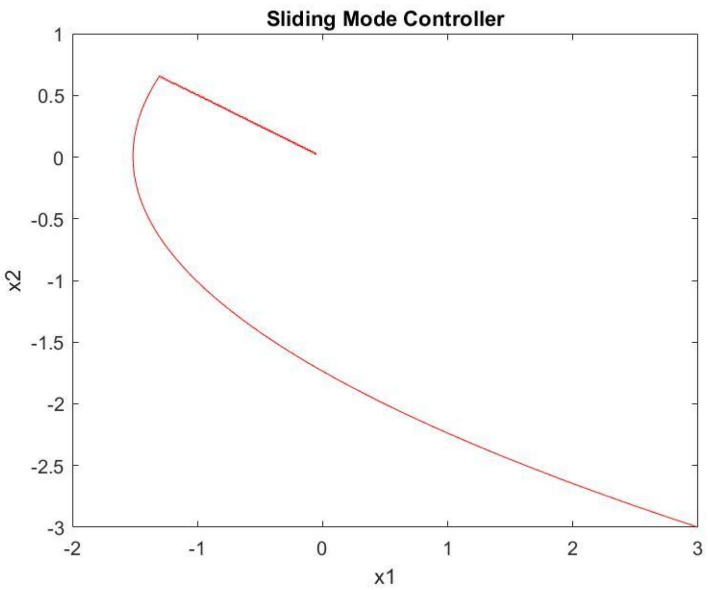
=x_2_dot+λ x_1_dot

= $x_1^3x_2+9x_1x_2^3+(x_1^2+x_2^2)u+\lambda\ x_2$

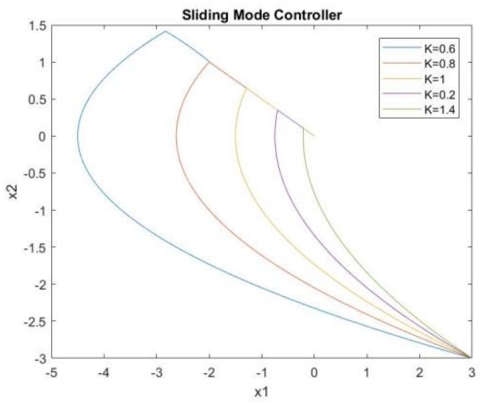
=-sign(f)*K

設計u=(-sign(f)*K- $x_1^3x_2-9x_1x_2^3$)/ $(x_1^2+x_2^2)$

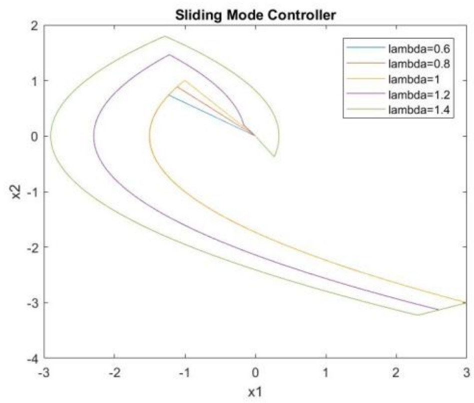
模擬結果1(K=1,λ =0.5,初值 x_1=3,x_2=-3)



模擬結果2(比較K和λ 對收斂情形的影響)



λ =0.5時不同K對收斂情形影響



K=1時不同λ 對收斂情形影響

b.

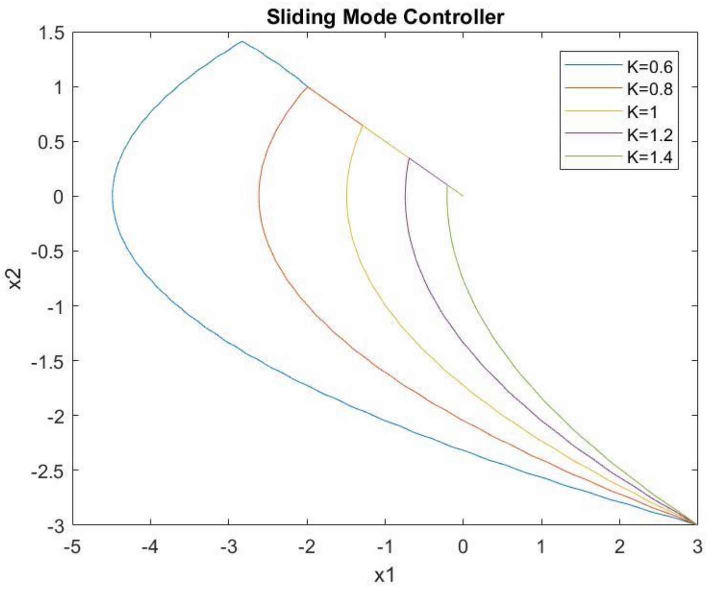
x_1_dot=x_2

x_2_dot=$x_1^3x_2+9x_1x_2^3+(x_1^2+x_2^2)u+d(t)$

d(t)=0.1sin(10π t)

控制器設計同上題

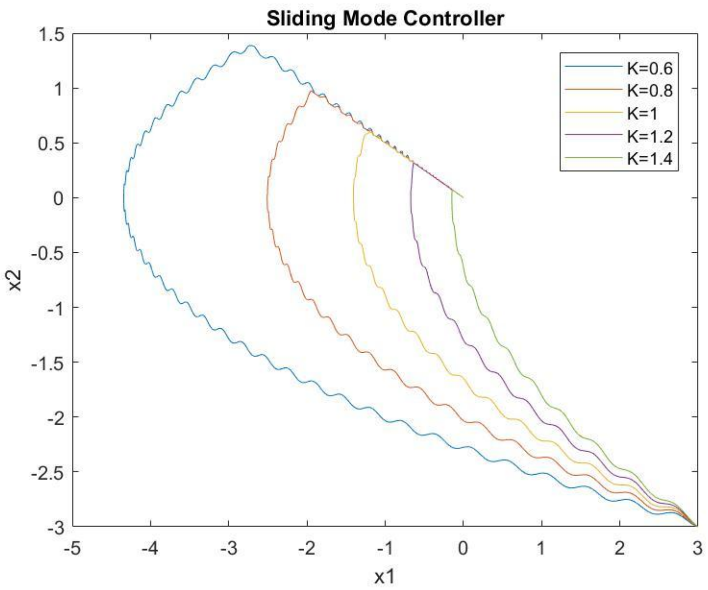
模擬結果1(λ =0.5,初值 $x_1=3,x_2=-3$)



模擬結果(2)

將干擾放大為d(t)=sin(10π t)效果比較明顯

K值越大，對抗干擾能力較佳



c.

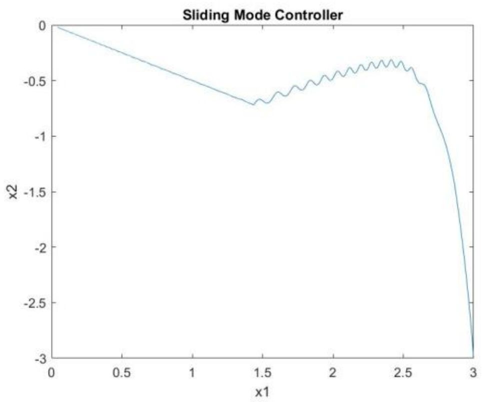
1.Sliding Mode Control

x_1_dot=x_2

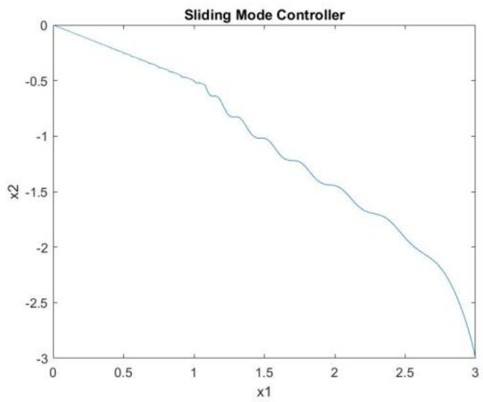
x_2_dot=$(1\pm0.2)x_1^3x_2+(9\pm0.4)x_1x_2^3+(x_1^2+x_2^2)u+d(t)$

$u=(-sign(f)*K-x_1^3x_2-9x_1x_2^3)/(x_1^2+x_2^2)$

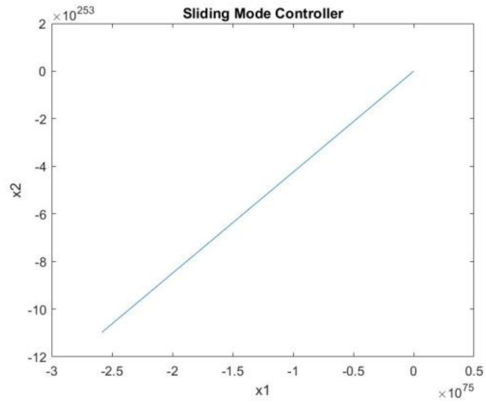
模擬結果(K=1,λ =0.5, 初值x_1=3,x_2=-3)



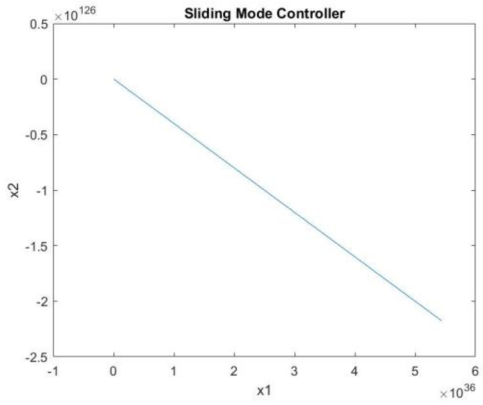
a=1-0.2,b=9-0.4(仍然收斂)



a=1+0.2,b=9-0.4(仍然收斂)



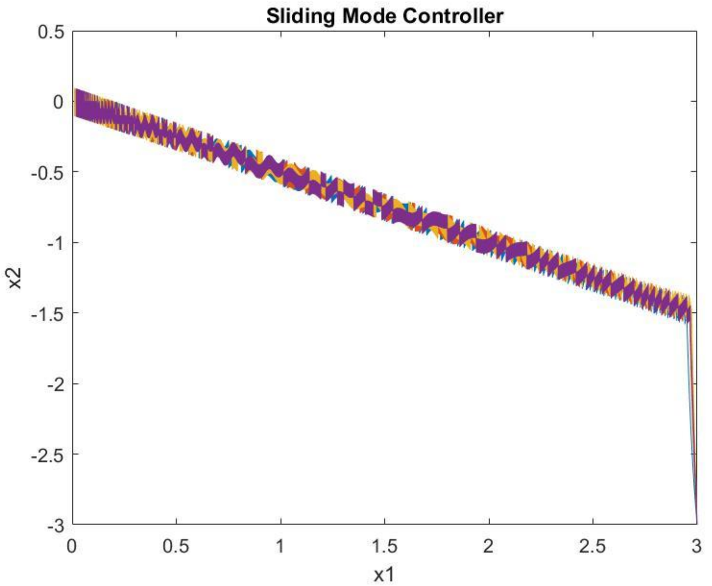
a=1-0.2,b=9+0.4(發散)



a=1+0.2,b=9+0.4(發散)

解決方法：將K設計大一點

當K=100時，剛剛四種誤差情形皆收斂



2.Feedback Linearization

x_1_dot=x_2

x_2_dot=$(1\pm0.2)x_1^3x_2+(9\pm0.4)x_1x_2^3+(x_1^2+x_2^2)u+d(t)$

取$z_1=x_1$

$z_2=z_1$_dot= x_1_dot=x_2

z_2_dot= x_2_dot=$x_1^3x_2+9x_1x_2^3+(x_1^2+x_2^2)u\equiv\alpha(x)+\beta(x)u$

$\alpha(x)= x_1^3x_2+9x_1x_2^3$

$\beta(x)= (x_1^2+x_2^2)$

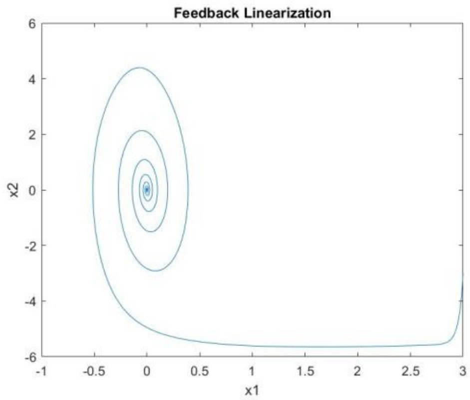
zdot=$\begin{bmatrix} 0 & 1 \\ k1 & k2 \end{bmatrix}$z，目標極點位置s=-1+9i, -1-9i

det($\begin{bmatrix} 0-s & 1 \\ k1 & k2-s \end{bmatrix}$)=$s^2-k_2s-k_1=s^2+2s+82$，經比較係數得 $k_1=-82, k_2=-2$

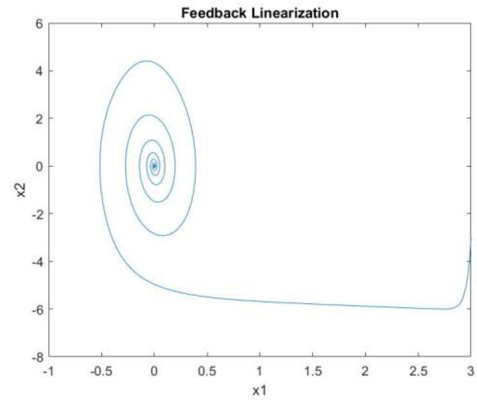
$u=(-\alpha(x)+(k_1z_1+k_2z_2))/\beta(x)$

$=(-(x_1^3x_2+9x_1x_2^3)+(-82x_1-2x_2))/(x_1^2+x_2^2)$

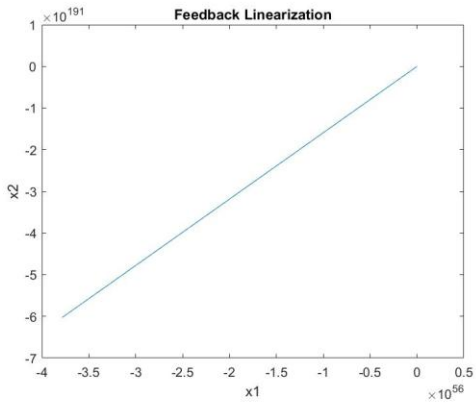
模擬結果(初值x_1=3,x_2=-3)



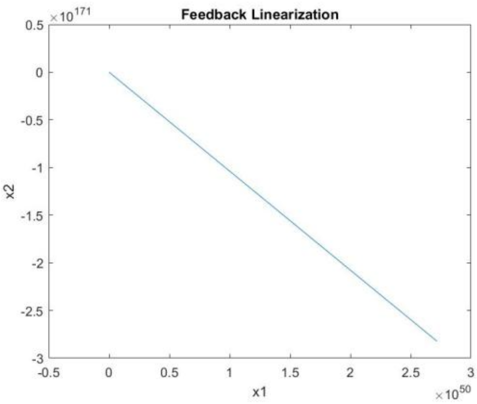
a=1-0.2,b=9-0.4(仍然收斂)



a=1+0.2,b=9-0.4(仍然收斂)



a=1-0.2,b=9+0.4(發散)



a=1+0.2,b=9+0.4(發散)

解決方法：Feedback Linearization無法解決誤差

附錄(Matlab Code)

1.Sliding Mode Control

(此為(b)小題程式碼，(a)和(c)也都差不多只調整一些參數)

```
clc;clear;

lambda=0.5;
K=1;
delta=0.001;
totalTime=10;
totalStep=totalTime/delta;
for K=0.6:0.2:1.4%compare with different K
    x1array=[1:totalStep]*0;x2array=x1array;
    x1array(1)=3;x2array(1)=-3;%init condition

    for i=1:totalStep
        x1=x1array(i);x2=x2array(i);

        f=x2+lambda*x1;
        u=(-sign(f)*K-x1^3*x2-9*x1*x2^3)/((x1^2+x2^2));

        d=0.1*sin(10*pi*(i*delta));
        x1_dot=x2;
        x2_dot=(1)*x1^3*x2+(9)*x1*x2^3+(x1^2+x2^2)*u+d;

        x1array(i+1)=x1+x1_dot*delta;
        x2array(i+1)=x2+x2_dot*delta;
    end

    plot(x1array,x2array);
    hold on;
end

xlabel('x1');
ylabel('x2');
title('Sliding Mode Controller');
legend({'K=0.6','K=0.8','K=1','K=1.2','K=1.4'});
```

2. Feedback Linearization

```
clear;clc;

delta=0.001;
totalTime=10;
totalStep=totalTime/delta;
x1array=[1:totalStep]*0;x2array=x1array;

x1array(1)=3;x2array(1)=-3;%init condition
for i=1:totalStep
    x1=x1array(i);x2=x2array(i);
    u=(-(x1^3*x2+9*x1*x2^3)+(-82*x1-2*x2))/(x1^2+x2^2);
    
    d=0.1*sin(10*pi*(i*delta));
    x1_dot=x2;
    x2_dot=(1-0.2)*x1^3*x2+(9-0.4)*x1*x2^3+(x1^2+x2^2)*u+d;

    x1array(i+1)=x1+x1_dot*delta;
    x2array(i+1)=x2+x2_dot*delta;
end
plot(x1array,x2array);
xlabel('x1');
ylabel('x2');
title('Feedback Linearization');
```