实验一

1. 题目：

程序：

A=[-0.3950,0.01145;-0.011,0];

B=[0.03362,1.038;0.000966,0];

C=[1,1];

D=[0,0];

[num1,den1]=ss2tf(A,B,C,D,1);

[num2,den2]=ss2tf(A,B,C,D,2);

disp('system transfer function of the first input is:');

num1

den1

disp('system transfer function of the second input is:');

num2

den2

disp('u(1) to y:');

tf1=tf(num1(1,:),den1)

zp1=zpk(tf1)

disp('u(2) to y:');

tf2=tf(num2(1,:),den1)

zp2=zpk(tf2)

运行结果：

system transfer function of the first input is:

num1 =

0 0.0346 0.0000

den1 =

1.0000 0.3950 0.0001

system transfer function of the second input is:

num2 =

0 1.0380 -0.0114

den2 =

1.0000 0.3950 0.0001

u(1) to y:

tf1 =

0.03459 s + 2.281e-05

-------------------------

s^2 + 0.395 s + 0.0001259

Continuous-time transfer function.

zp1 =

0.034586 (s+0.0006595)

------------------------

(s+0.3947) (s+0.0003191)

Continuous-time zero/pole/gain model.

u(2) to y:

tf2 =

1.038 s - 0.01142

-------------------------

s^2 + 0.395 s + 0.0001259

Continuous-time transfer function.

zp2 =

1.038 (s-0.011)

------------------------

(s+0.3947) (s+0.0003191)

Continuous-time zero/pole/gain model.

1. 题目：

程序：

num=[0,0,1];

den=[1,3,2];

disp('状态空间表达式:');

[A,B,C,D]=tf2ss(num,den)

运行结果：

状态空间表达式:

A =

-3 -2

1 0

B =

1

0

C =

0 1

D =

0

思考：由传递函数到状态空间表达式的形式不唯一，这里matlab默认采用的是x2的导数等于x1来选取的状态变量。

1. 题目：

程序：

num=[0,0,0,5,100];

den=conv([1,0],conv([1,4.6],[1,3.4,16.35]));

disp('开环传递函数:');

otf=tf(num,den)

cnum=[5,100];

cden=num+den;

disp('闭环传递函数:');

ctf=tf(cnum,cden)

[cz,cp]=tf2zp(cnum,cden)

n=length(find(real(cp)>0));

if(n>0)

disp('系统不稳定');

else

disp('系统稳定');

end

运行结果：

开环传递函数:

otf =

5 s + 100

---------------------------------

s^4 + 8 s^3 + 31.99 s^2 + 75.21 s

Continuous-time transfer function.

闭环传递函数:

ctf =

5 s + 100

---------------------------------------

s^4 + 8 s^3 + 31.99 s^2 + 80.21 s + 100

Continuous-time transfer function.

cz =

-20

cp =

-0.9987 + 3.0091i

-0.9987 - 3.0091i

-3.0013 + 0.9697i

-3.0013 - 0.9697i

系统稳定

1. 题目：

程序：

%sum of series

disp('级数求和结果:');

n=63;

a=2;

x=[0:n];

x=a.^x;

y=ones(n+1,1);

z=x\*y

运行结果：

级数求和结果:

z =

1.8447e+19

实验二

1. 题目：

程序：

A=[4,7,9;6,5,2;12,5,7];

B=[1,9,7;2,8,3;5,10,4];

%A+B

A+B

%A-B

A-B

%A\*B

A\*B

%A/B

A/B

%rank of A,B

rank(A),rank(B)

运行结果：

ans =

5 16 16

8 13 5

17 15 11

ans =

3 -2 2

4 -3 -1

7 -5 3

ans =

63 182 85

26 114 65

57 218 127

ans =

1.6400 -3.1200 1.7200

-0.0933 -1.5867 1.8533

1.0800 -6.6400 4.8400

ans =

3

ans =

3

ans =

-296

ans =

-75

ans =

4 6 12

7 5 5

9 2 7

ans =

1 2 5

9 8 10

7 3 4

ans =

-0.0845 0.0135 0.1047

0.0608 0.2703 -0.1554

0.1014 -0.2162 0.0743

ans =

-0.0267 -0.4533 0.3867

-0.0933 0.4133 -0.1467

0.2667 -0.4667 0.1333

1. 题目：

程序：

x=pi\*[-2:0.05:2];

y1=2\*sin(x);

y2=3\*cos(2\*x);

y3=x.^2;

plot(x,y1,'r-');

hold on

plot(x,y2,'b:');

hold on

plot(x,y3,'y-.');

hold on

legend('y1','y2','y3');

grid on

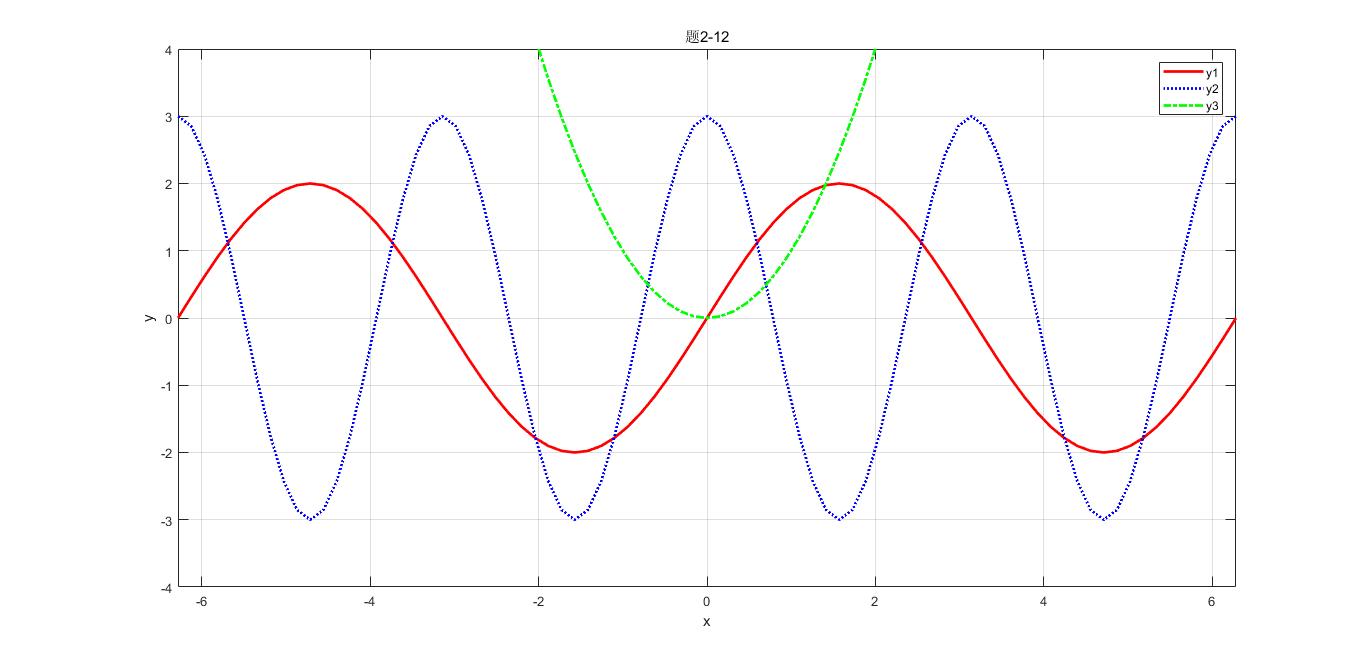
axis([-2\*pi,2\*pi,-4,4]);

title('题2-12');

xlabel('x');

ylabel('y');

运行结果：



1. 题目：

程序：

%欧拉法

h1=0.1;

N1=round(1/h1)+1;

x1=[0:h1:1];

y1=zeros(1,N1);

y1(1)=1;

for i=2:N1

y1(i)=y1(i-1)-h1\*y1(i-1);

end

%四阶龙格库塔

h2=0.1;

N2=round(1/h2)+1;

x2=[0:h2:1];

y2=zeros(1,N2);

y2(1)=1;

for i=2:N2

k1=-y2(i-1);

k2=-y2(i-1)-0.5\*h2\*k1;

k3=-y2(i-1)-0.5\*h2\*k2;

k4=-y2(i-1)-h2\*k3;

y2(i)=y2(i-1)+1/6\*h2\*(k1+2\*k2+2\*k3+k4);

end

%输出图像比较

x3=[0:0.05:1];

y3=exp(-x3);

plot(x1,y1,'r-');

hold on

plot(x2,y2,'b:');

hold on

plot(x3,y3,'g-.');

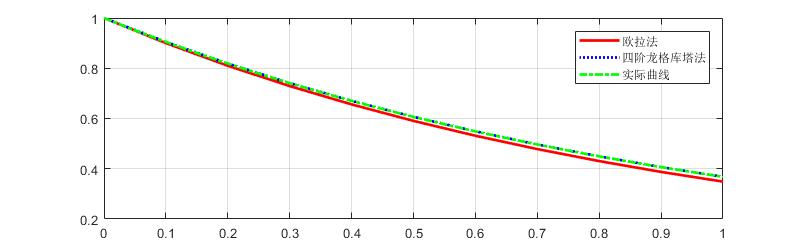
hold on

legend('欧拉法','四阶龙格库塔法','实际曲线');

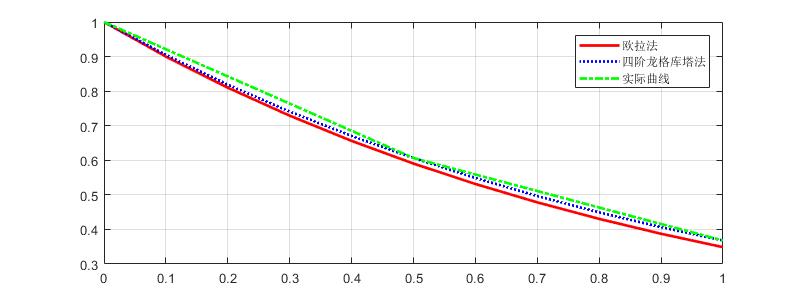
grid on

运行结果：

当绘图步长选的小一点（0.05）时，发现四阶龙格库塔法曲线和实际曲线基本重合。



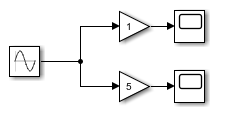
当绘图步长选的大一些（0.5）时，此时能够清楚地显现出三条曲线的误差差距。



实验三

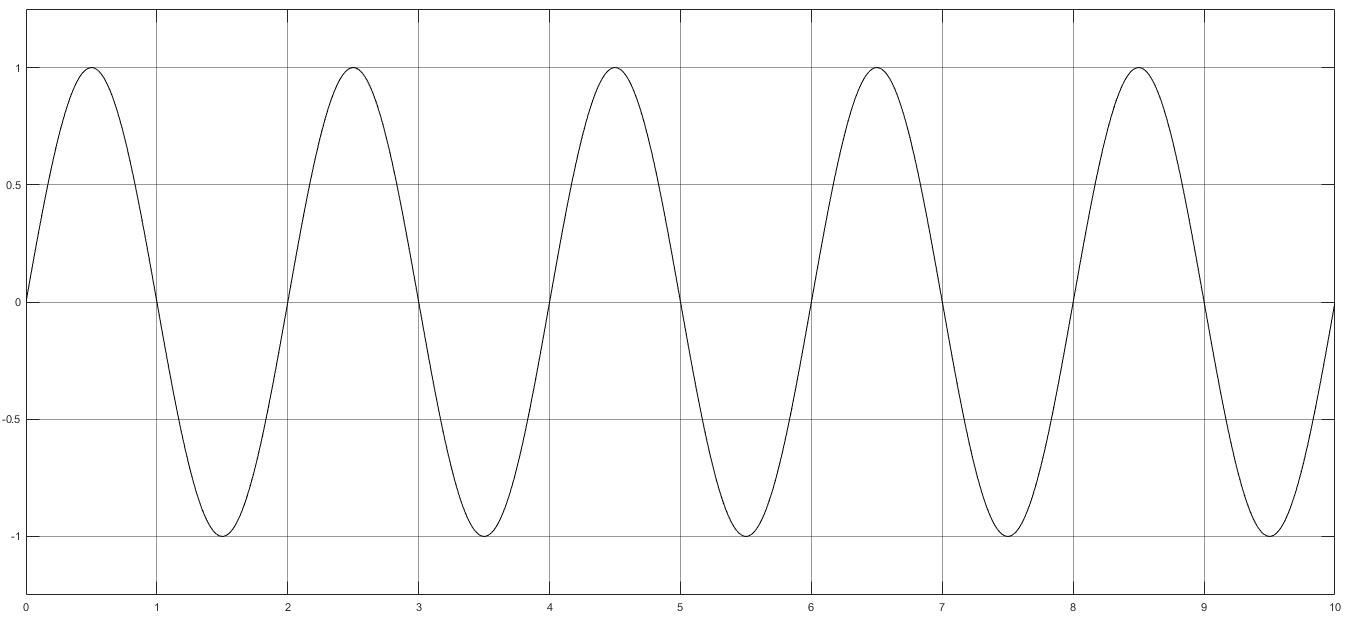
1. 题目：

Simulink框图：

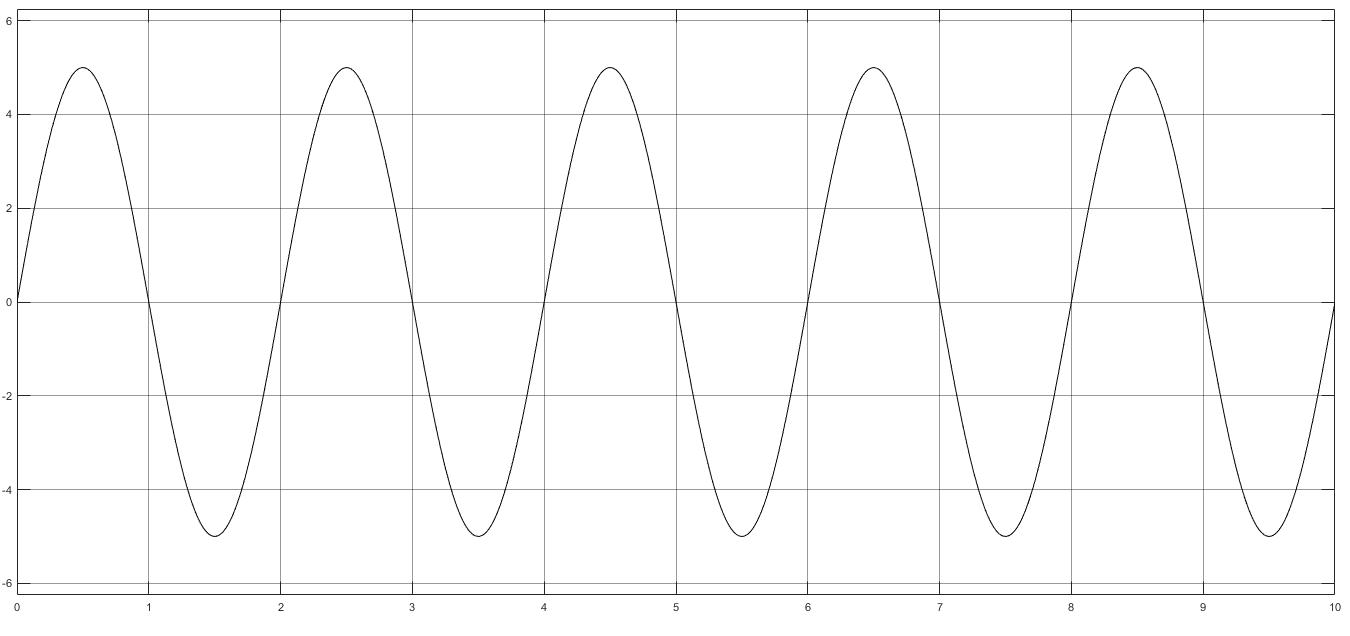


设置正弦波频率为0.5Hz：



运行结果：按一倍输入示波器：

按五倍输入示波器：



1. 题目：

分析：

由于初始状态[x1,x2]=[0,0]时，x1、x2恒为零，因此这里我们选初始状态[x1,x2]=[1,1]。最终结果状态响应曲线呈周期性变化，相平面曲线呈现为向外发散的极限环，存在不稳定焦点，此时系统阻尼比介于-1和0之间。

M文件求解程序：

fun\_u=@(t,u) [u(2);-((u(1)^2-1)\*u(2)+u(1))];

[t,u]=ode45(fun\_u, 0:0.05:20, [1 1]);

subplot(2,2,1);

plot(t,u(:,1));

xlabel('t');

ylabel('x1');

grid on

subplot(2,2,2);

plot(t,u(:,2));

xlabel('t');

ylabel('x2');

grid on

subplot(2,2,3);

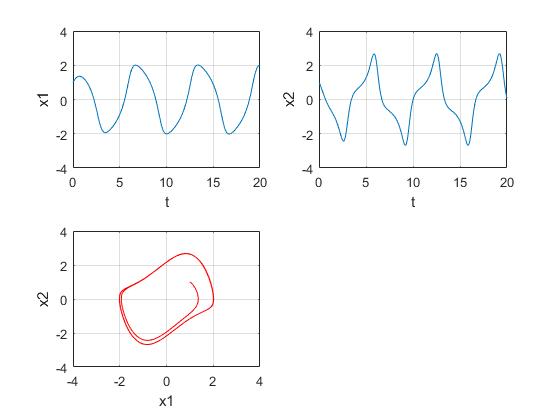
plot(u(:,1),u(:,2),'r');%极限环向外发散，不稳定焦点，阻尼比介于-1和0之间

xlabel('x1');

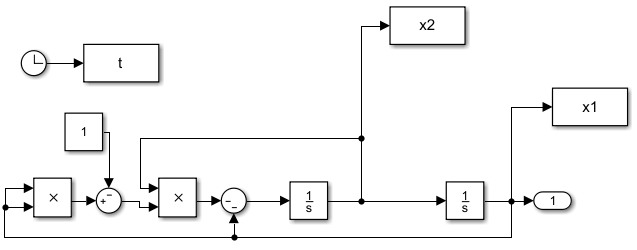
ylabel('x2');

grid on

运行结果：



Simulink求解框图：



其中x1和x2初值分别由两个积分器的初值给定，将t、x1、x2保存到工作空间打印出曲线来，程序如下：

subplot(2,2,1);

plot(t,x1);

xlabel('t');

ylabel('x1');

grid on

axis([0,10,-4,4]);

subplot(2,2,2);

plot(t,x2);

xlabel('t');

ylabel('x2');

grid on

axis([0,10,-4,4]);

subplot(2,2,3);

plot(x1,x2,'r');%，极限环向外发散，不稳定焦点，阻尼比介于-1和0之间

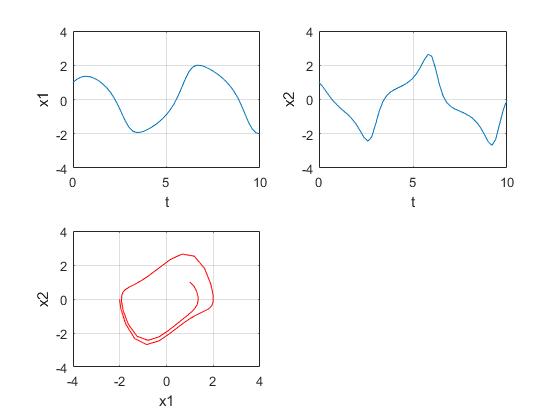
xlabel('x1');

ylabel('x2');

grid on

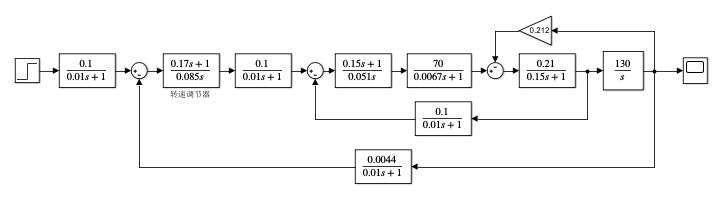
axis([-4,4,-4,4]);

运行结果：

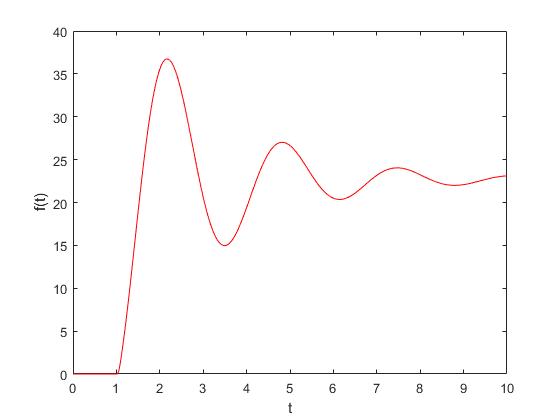


1. 题目：

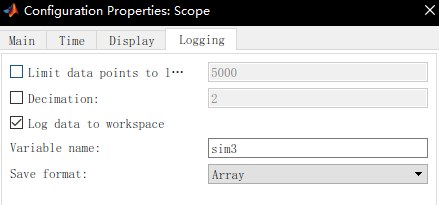
Simulink框图：



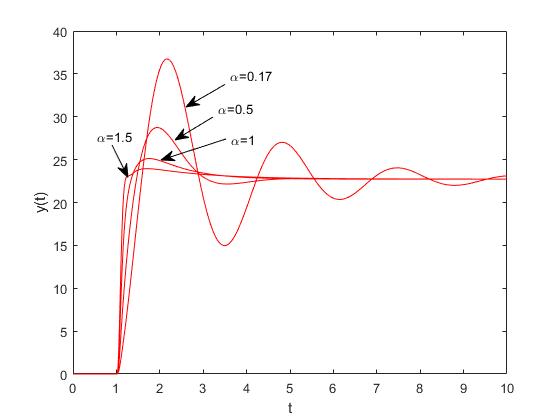
直接控制效果：



调整转速调节器参数，通过示波器的设置将每次不同参数调节的效果输出到工作空间，然后用plot函数画出曲线：



调整参数控制效果：



实验四

1. 题目：

程序：

A=[1,2,-1;0,1,0;0,-4,3];

B=[0;1;1];

C=[1,-1,1];

Qc=ctrb(A,B);%能控性矩阵

n1=rank(Qc)

if n1<3

[AB1,BB1,CB1,T1,K1]=ctrbf(A,B,C)

end

Qo=obsv(A,C);%能观测性矩阵

n2=rank(Qo)

if n2<3

[AB2,BB2,CB2,T2,K2]=obsvf(A,B,C)

end

运行结果：

n1 =

2

AB1 =

1.0000 -0.0000 0.0000

2.1213 4.0000 1.2247

-1.7321 -2.4495 -0.0000

BB1 =

0.0000

-0.0000

1.4142

CB1 =

-1.6330 -0.5774 0

T1 =

-0.8165 0.4082 -0.4082

0.5774 0.5774 -0.5774

0 0.7071 0.7071

K1 =

1 1 0

n2 =

2

AB2 =

1.0000 2.8868 3.5355

0.0000 2.0000 1.2247

0.0000 0.8165 2.0000

BB2 =

1.2247

-0.7071

0

CB2 =

0 0.0000 1.7321

T2 =

-0.4082 0.4082 0.8165

-0.7071 -0.7071 -0.0000

0.5774 -0.5774 0.5774

K2 =

1 1 0

1. 题目：

程序：

den=[1,4,-4,4,-7,-8,10];

disp('特征根:');

cp=roots(den)

n=length(find(real(cp)>0));

if(n>0)

disp('系统不稳定');

else

disp('系统稳定');

end

运行结果：

特征根:

cp =

-5.0000 + 0.0000i

-0.0000 + 1.4142i

-0.0000 - 1.4142i

-1.0000 + 0.0000i

1.0000 + 0.0000i

1.0000 + 0.0000i

系统不稳定

1. 题目：

程序：

A=[1,2,3;4,5,6;7,8,0];

C=[1,5,4;5,6,7;4,7,9];

V=lyap(A,C)

num=length(V)

flag=0;

for i=1:num

detn(i)=det(V([1:i],[1:i]));

if detn(i)<=0

flag=flag+1;

end

end

detn

if flag>0

disp('系统不稳定');

elseif flag==0

disp('系统稳定');

end

运行结果：

V =

-1.5556 1.1111 -0.3889

1.1111 -1.2222 -0.2222

-0.3889 -0.2222 -0.3889

num =

3

detn =

-1.5556 0.6667 0.1944

系统不稳定

1. 题目：

程序：

num=[1];

den=conv([1,0],conv([1,4],[1,4,16]));

rlocus(num,den);%绘制根轨迹

%k为十字光标选中点的增益，poles存放了几个重要极点

[k,poles]=rlocfind(num,den)

运行结果：

selected\_point =

-1.9964 - 2.0433i

k =

63.9695（分离点的K值）

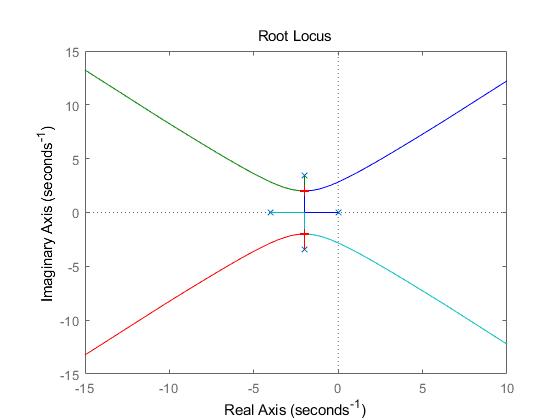
poles =

-2.0000 + 2.0432i

-2.0000 - 2.0432i

-2.0000 + 1.9559i

-2.0000 - 1.9559i



1. 题目：

程序：

num=[10];

den=conv([1,0],conv([1,1],[1,5]));

%绘制bode图，并求出幅值裕度Gm和相角裕度Pm

margin(num,den)

运行结果：

