1. 实验笔记

1.1 常用的数学模型

(1)传递函数模型（系统的外部模型）；

对线性定常系统，式中s的系数均为常数，且a1不等于零，这时系统在MATLAB中可以方便地由分子和分母系数构成的两个向量唯一地确定出来，这两个向量分别用num和den表示。

分子：num=[b1,b2,...,bm,bm+1] 分母：den=[a1,a2,...,an,an+1] 注意：它们都是按s的降幂进行排列的。

(2)零极点增益模型等

K为系统增益，zi为零点，pj为极点,在MATLAB中零极点增益模型用[z,p,K]矢量组表示。即： z=[z1,z2,...,zm] p=[p1,p2,...,pn] K=[k]

函数tf2zp()可以用来求传递函数的零极点和增益。 传递函数的形式 sys=tf(num,den) 其中tf()代表传递函数的形式描述系统，

零极点形式 sys1=zpk(sys) 传递函数形式和零极点形式之间可以相互转化，语句为 [z,p,k] = tf2zp(num,den) [num,den] = zp2tf(z,p,k) 当传递函数复杂时，应用多项式乘法函数conv()等实现。

(3)部分分式展开

[r,p,k]=residue(num,den)对两个多项式的比进行部分展开，以及把传函分解为微分单元的形式。

[num, den]=residue(r,p,k)可以将部分分式转化为多项式比

1.2 结构图模型

(1) 串联

一个开环控制系统可以通过G1(s)与G2(s)两个环节的串联而得到，利用series()函数可以求串联连接的传递函数，函数的具体形式为：

[num,den]= series(num1,den1, num2,den2)

(2) 并联

当系统是以并联的形式连接时，利用parallel()函数可得到系统的传递函数。指令的具体形式为：%将并联连接的传递函数进行相加。

[num,den]= parallel (num1,den1, num2,den2)

(3) 反馈

系统以反馈方式构成闭环,求闭环传递函数的MATLAB函数有两个：cloop()和feedback()其中cloop()函数只能用于H (s)=1（即单位反馈）的情况。

cloop()函数的具体用法为： [num,den]= cloop (numg,deng, sign)

feedback()函数的用法为：[num,den]= feedback (numg,deng,numh,denh, sign)

1.3 线性系统的时间响应分析

函数名	函数功能	调用格式	参数说明	备注
step()	求取系统单位阶跃响应	y=step(num,den,t)	num和den含义与上文相同	如果对具体的响应数值不感兴趣，而只想绘制出系统的阶跃响应曲线，则调用step(sys,t)或step(sys)
impulse()	求取系统的冲激响应	y=impulse(num,den,t)	与上文相同	如果对具体的响应数值不感兴趣，而只想绘制出系统的阶跃响应曲线，则调用impulse(sys,t)或impulse(sys)
lsim()	求取系统任意激励响应	[y,x]=lsim(sys,u,t)	与上文相同	如果对具体的响应数值不感兴趣，而只想绘制出系统的阶跃响应曲线，则调用lsim(sys,u,t)或impulse(num,den,,u,t)
gensig	对LSIM产生输入信号	[u,t]=gensig(type,tau)	按指定类型type和周期tau生成特定类型的激励信号。Type可取为：‘sin’、‘square’、‘pulse’	例如[u,t] = gensig('square',4,10,0.1);

1.4 控制系统的频域响应分析

Gw=polyval(num, sqrt(-1)*w)./polyval(den,sqrt(-1)*w); 其中num与den分别为系统的分子分母多项式系数向量

频域分析法是应用频率特性研究控制系统的一种典型方法。通常将频率特性用曲线的形式进行表示，包括对数频率特性曲线(波特图/bode图)和幅相频率特性曲线简称幅相曲线(奈奎斯特图/nyquist图)，MATLAB提供了绘制这两种曲线的函数。

函数名	函数功能	调用格式	参数说明	备注
bode()	对数频率特性图（波特图）绘制	[m,p,w]=bode(num,den,w)	m,p,w分别代表Bode响应的幅值向量、相位向量和角频率。w为频率点构成的向量，该向量最好由logspace()函数构成	bode(num,den)或者bode(num,den,w)格式调用也是可行的

函数名	函数功能	调用格式	参数说明	备注
logspace()	产生n个对数行向量	logspace(a,b,n)	在（10a~ 10b）之间产生n个对数行向量；当没有n时，产生50个对数行向量	常用于生成w
nyquist()	Nyquist曲线绘制	[re,im,w]= nyquist(num,den,w)	返回实部re和虚部im及角频率点w矢量（为正的部分）。然后用plot(re,im)绘制出对应w从负无穷到零变化的部分。	nyquist(num,den)或者nyquist(num,den,w)格式调用也是可行的
freqs()	计算由矢量a和b构成的模拟滤波器H(s)=B(s)/A(s)的幅频响应(模拟滤波器特性)	h=freqs(b,a,w)	其中实矢量w用于指定频率值，返回值h为一个复数行向量，要得到幅值必须对它取绝对值，即求模。	[h,w]=freqs(b,a)自动设定200个频率点来计算频率响应，这200个频率值记录在w中。

1.5. 控制系统的根轨迹图

1. rlocus(a,b,c,d)或者rlocus(num,den)：根据SISO开环系统的状态空间描述模型和传递函数模型，直接在屏幕上绘制出系统的根轨迹图。开环增益的值从零到无穷大变化。

2. rlocus(a,b,c,d,k)或rlocus(num,den,k)： 通过指定开环增益k的变化范围来绘制系统的根轨迹图。

1.6. 系统稳定性分析

函数名	函数功能	调用格式	参数说明	备注
pzmap()	零极点图绘制	[p,z]=pzmap(num,den)	num,den与前面意思一样	以pzmap(a,b,c,d)或pzmap(num,den)格式调用也可
rlocfind()	找出给定的一组根（闭环极点）对应的根轨迹增益	[k,p]=rlocfind(num,den)	num,den与前面意思一样	不带输出参数项[k,p]时，同样可以执行，只是此时只将k的值返回到缺省变量ans中

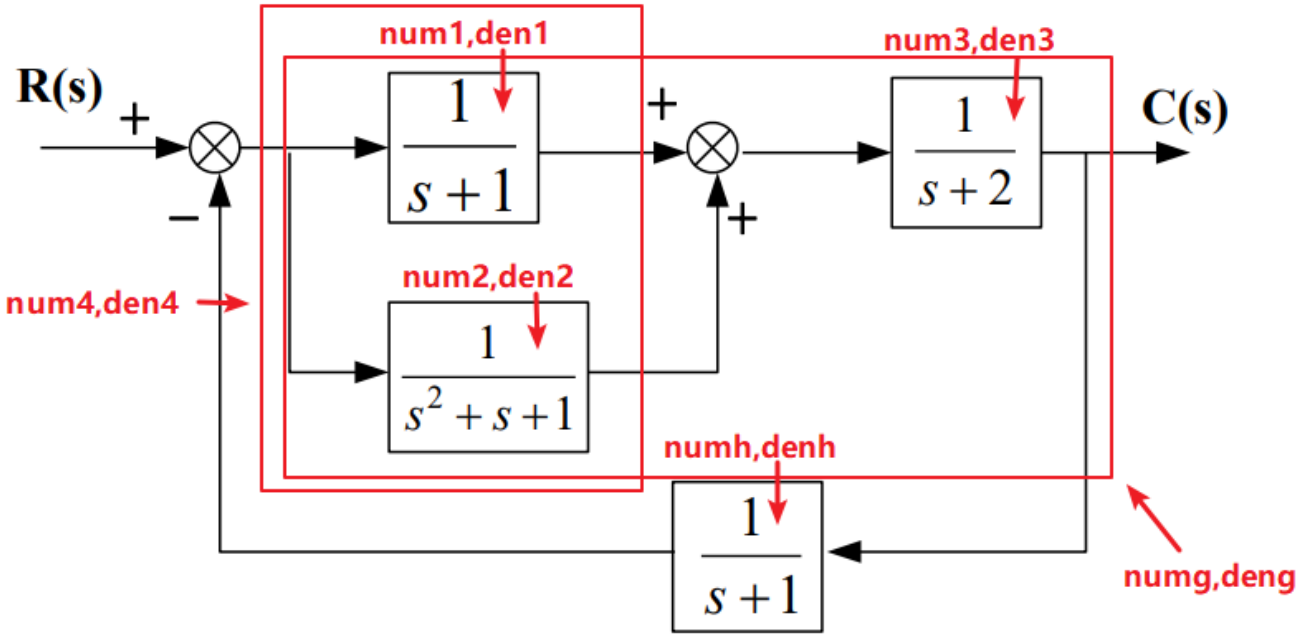
函数名	函数功能	调用格式	参数说明	备注
sgrid()	在现存的屏幕根轨迹或零极点图上绘制出自然振荡频率wn、阻尼比矢量z对应的格线。	sgrid('new')与sgrid(z,wn)	sgrid('new')：是先清屏，再画格线。sgrid(z,wn)：则绘制由用户指定的阻尼比矢量z、自然振荡频率wn的格线	无
ii=find(条件式)	用来求取满足条件的向量的下标向量，以列向量表示	ii=find(条件式)	将满足条件的结果返回到ii向量中	例如ii=find(real(p)>0):如果最终的结果里ii的元素个数大于0，则认为找到了不稳定极点，因而给出系统不稳定的提示。

1.7. Simulink仿真工具

连续模块库(Continuous)

模块名	模块功能
积分模块(Integrator)	对输入变量进行积分。说明：模块的输入可以是标量，也可以是矢量；输入信号的维数必须与输入信号保持一致。
微分模块(Derivative)	通过计算差分Δu/ Δt近似计算输入变量的微分。
线性状态空间模块(State-Space)	用于实现以下数学方程描述的系统：X=Ax + bu and Y=Cx + Du
传递函数模块(Transfer Fcn)	用执行一个线性传递函数。
零极点传递函数模块(Zero-Pole)	用于建立一个预先指定的零点、极点，并用延迟算子s表示的连续。
存储器模块(Memory)	保持输出前一步的输入值。
传输延迟模块(Transport Delay)	用于将输入端的信号延迟指定的时间后再传输给输出信号。
可变传输延迟模块(Variable Transport Delay)	用于将输入端的信号进行可变时间的延迟。

2. 题目1



2.1 三种形式的模型

2.1.1 传递函数模型

下述代码说明： 首先对图中的每个

```
clc,clear;
num1=[1];
den1=[1 1];

num2=[1];
den2=[1 1 1];

num3=[1];
den3=[1 2];

[num4,den4]=parallel(num1,den1,num2,den2);
[numg,deng]=series(num4,den4,num3,den3);

numh=[1];
denh=[1 1];

[num,den]=feedback(numg,deng,numh,denh,-1)
```

```
num =

     0     0     1     3     4     2
```

```
den =

     1     5    10    12     9     4

sys =

           s^3 + 3 s^2 + 4 s + 2
  ---------------------------------------
   s^5 + 5 s^4 + 10 s^3 + 12 s^2 + 9 s + 4
```

2.1.2 零极点模型

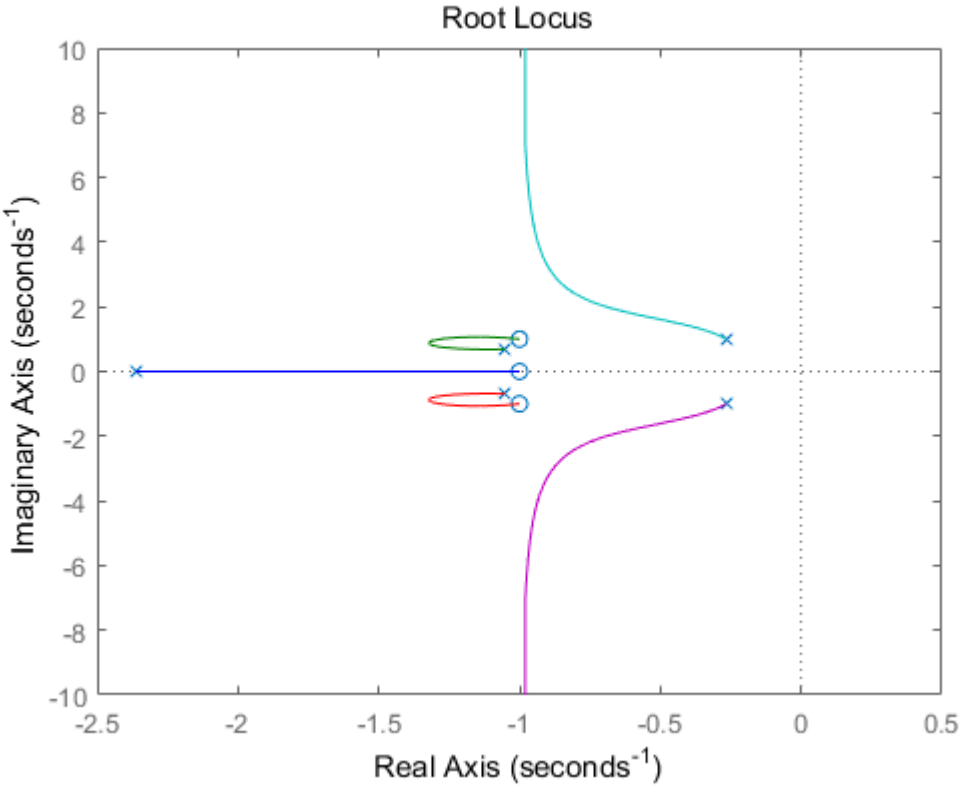
```
[z,p,k]=tf2zp(num,den)

s=zpk(z,p,k,-1)
```

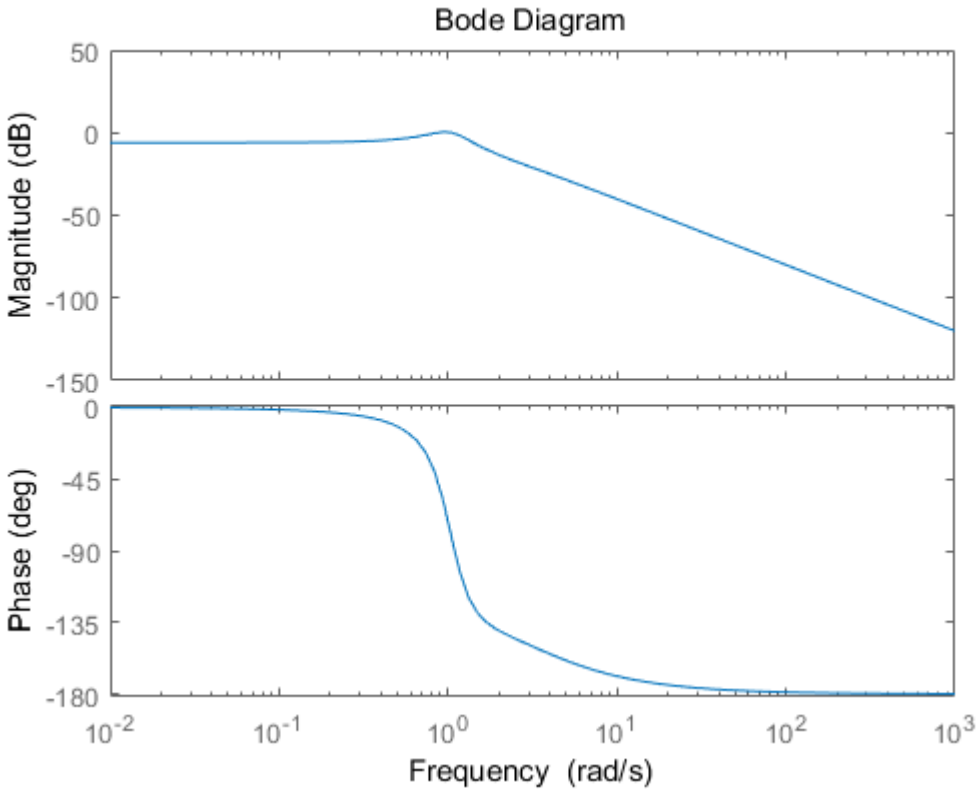
```
z =

  -1.0000 + 1.0000i
  -1.0000 - 1.0000i
  -1.0000 + 0.0000i

p =

  -2.3640 + 0.0000i
  -1.0528 + 0.6881i
  -1.0528 - 0.6881i
  -0.2652 + 0.9997i
  -0.2652 - 0.9997i

k =

     1

s =

                  (z+1) (z^2 + 2z + 2)
  ---------------------------------------------------------
    (z+2.364) (z^2 + 2.106z + 1.582) (z^2 + 0.5305z + 1.07)
```

2.1.3 部分分式模型

```
[r,p1,k1]=residue(num,den)
```

```
r =

  -0.3293 + 0.0000i
   0.1121 + 0.0244i
   0.1121 - 0.0244i
   0.0526 - 0.2596i
   0.0526 + 0.2596i


p1 =

  -2.3640 + 0.0000i
  -1.0528 + 0.6881i
  -1.0528 - 0.6881i
  -0.2652 + 0.9997i
  -0.2652 - 0.9997i


k1 =

     []

s =

F(s)
     =
       (-0.3293 + 0.0000i) / s-(-2.3640 + 0.0000i)
     + (0.1121 + 0.0244i) / s-(-1.0528 + 0.6881i)
     + (0.1121 - 0.0244i) / s-(-1.0528 - 0.6881i)
     + (0.0526 - 0.2596i) / s-(-0.2652 + 0.9997i)
     + (0.0526 + 0.2596i) / s-(-0.2652 - 0.9997i)
```





2.2 线性系统的时间响应分析

2.2.1 求取系统单位阶跃响应：**step()**

(1) y=step(sys,t)：其中sys可以由tf()或zpk()函数得到， t为选定的仿真时间向量，一般可以由t=0:step:end等步长地产生出来。此函数只返回仿真数据而不在屏幕上画仿真图形,返回值y为系统在仿真时刻各个输出所组成的

矩阵。

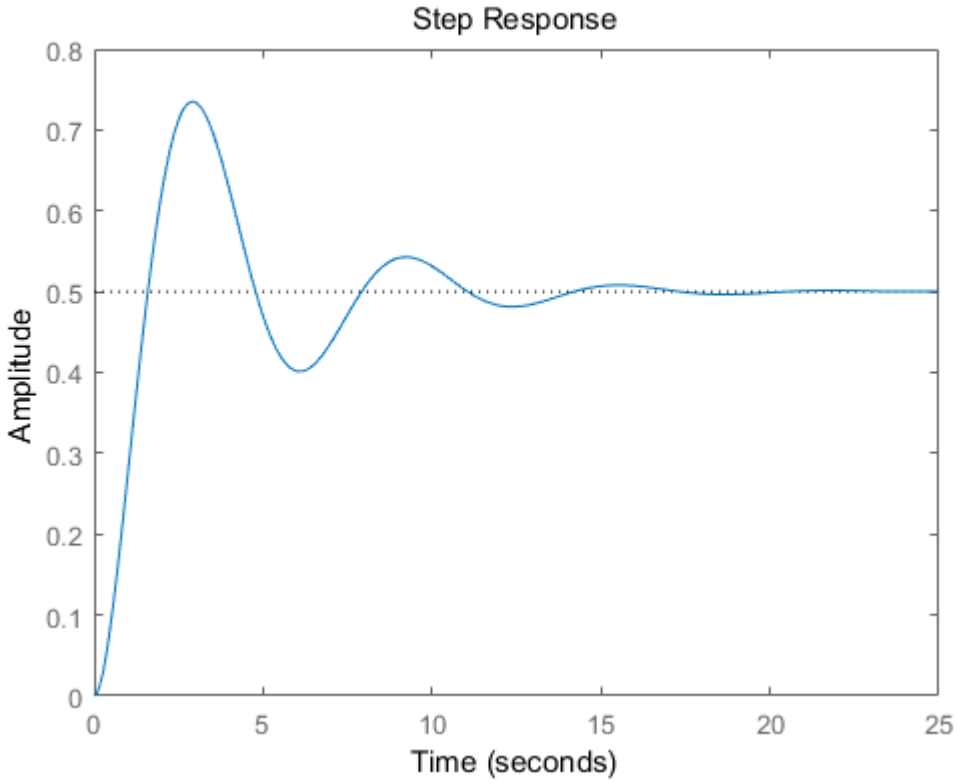
(2) [y,x,t]=step(sys)：此时时间向量t由系统模型的特性自动生成, 状态变量x返回为空矩阵。

(3) step(sys)或者step(num,den):绘制出系统的阶跃响应曲线:

输入代码

```
step(num,den);
```

结果图如下



2.2.2 求取系统的冲激响应：impulse()

与step()函数类似的，调用mpulse()函数也有四种格式

(1) y=impulse(num,den,t)；

(2) y=impulse(sys,t) 或 impulse(num,den,t)

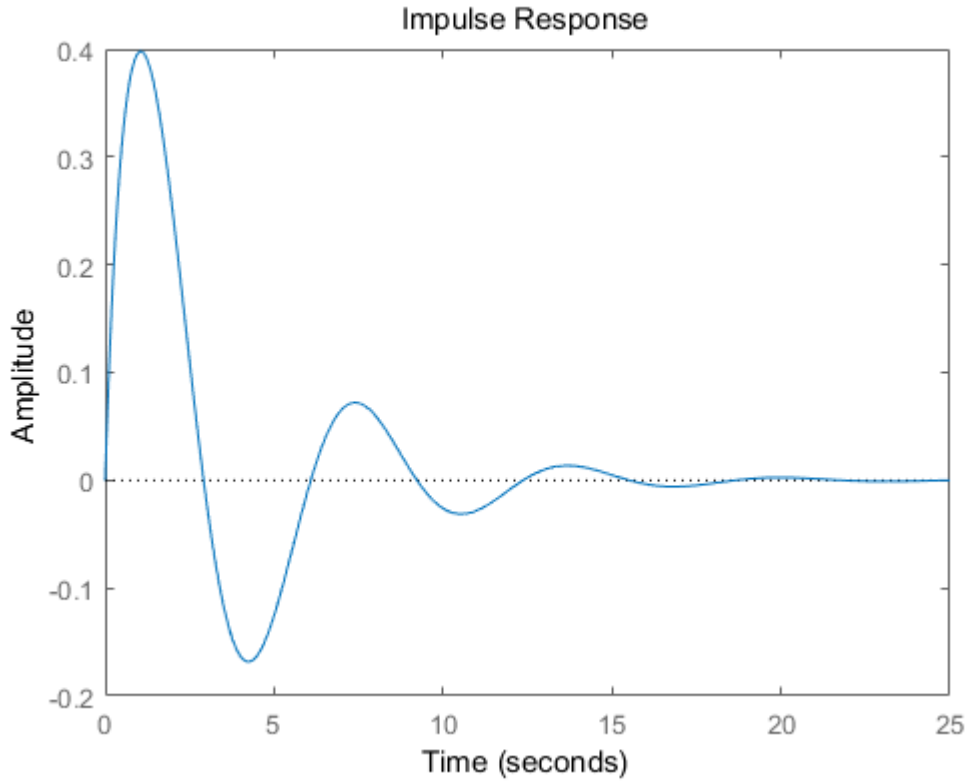
(3) [y,x,t]=impulse(num,den)；

(4) impulse(num,den)；

输入代码

```
impulse(num,den);
```

结果图如下



2.2.3 求取系统任意激励响应：lsim()

lsim()函数调用格式如下

(1) [y,x]=lsim(sys,u,t)

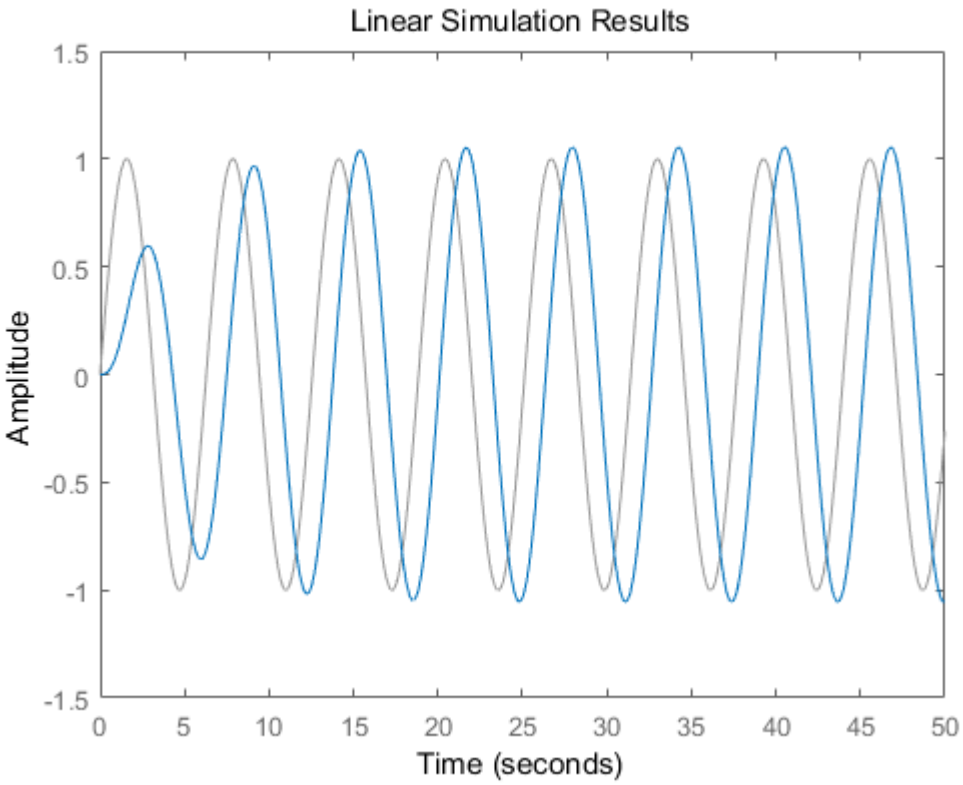
(2) lsim(sys,u,t)

式中，u为给定输入构成的列向量，它的元素个数应该和t的个数是一致的。当然该函数若调用时不返回参数，也可以直接绘制出响应曲线图形。

这次我尝试了三种激励的响应 (1) 正弦激励

```
t = 0:0.01:50;
u = sin(t);
lsim(num,den,u,t)
```

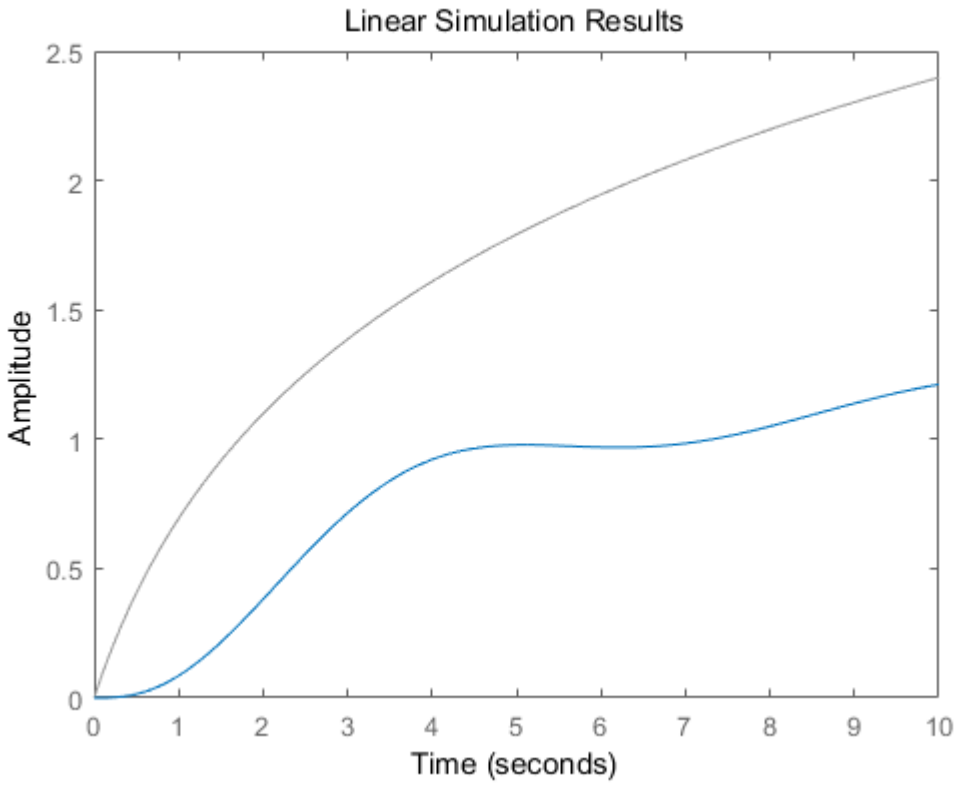
结果图如下



(2) 对数激励

```
t = 0:0.01:10;
u = log(t+1);
lsim(num,den,u,t)
```

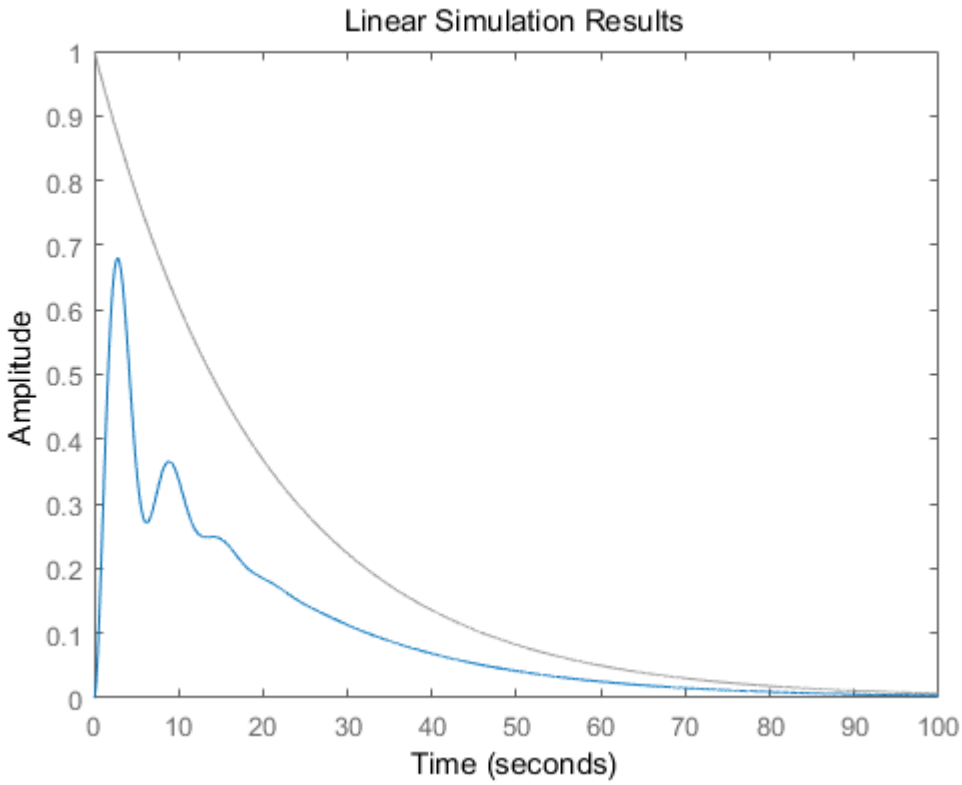
结果图如下



(3) e次幂激励

```
t = 0:0.01:100;
u = exp(-0.05*t);
lsim(num,den,u,t)
```

结果图如下



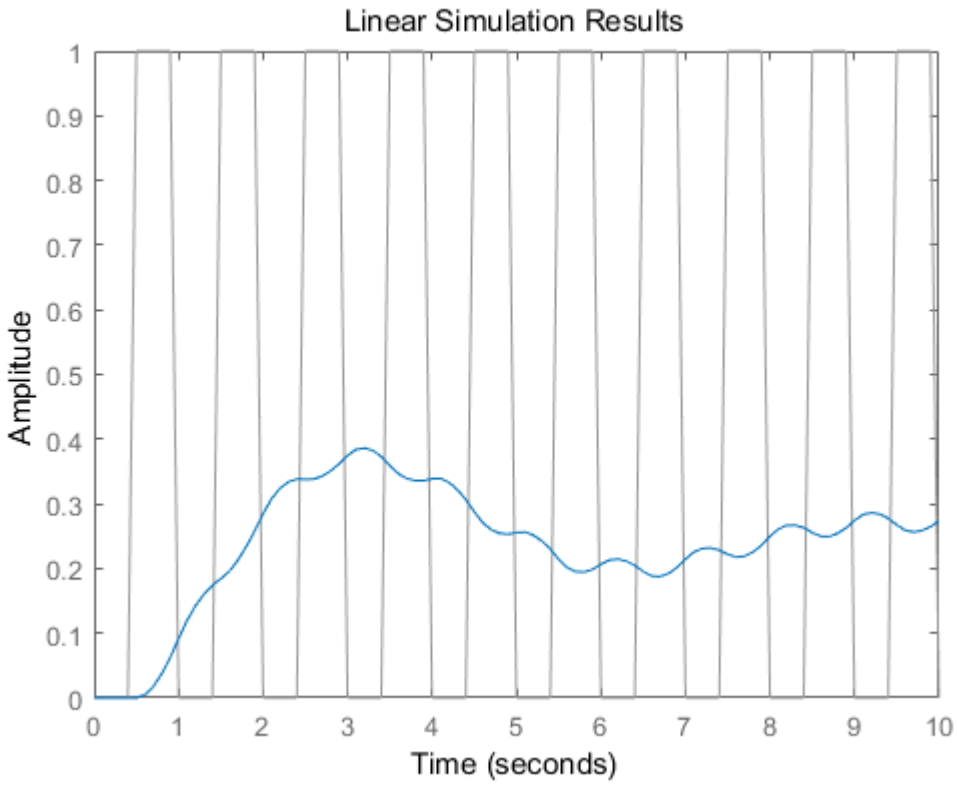
2.2.4 gensig - 对LSIM产生输入信号

调用格式：[u,t]=gensig(type,tau)

函数功能：按指定类型type和周期tau生成特定类型的激励信号。Type可取为：‘sin’、‘square’、‘pulse

```
[u,t] = gensig('square',4,10,0.1);
lsim(num,den,u,t)
```

结果图如下



2.2.5 stepfun - 产生单位阶跃输入

调用格式：y=stepfun(t,to)

函数功能：可用于产生单位阶跃输入

输入以下代码

```
t=0:1:10;
to=1;
y=stepfun(t,to)
```

结果如下

```
y =     0     1     1     1     1     1     1     1     1     1     1
```

2.3 控制系统的频域响应分析

2.3.1 logspace()函数

函数调用格式：logspace(a,b,n)

函数功能：在（10a~ 10b）之间产生n个对数行向量；当没有n时，产生50个对数行向量。

输入以下代码

```
w=logspace(0,2,5)
```

结果如下

```
w =    1.0000    3.1623   10.0000   31.6228  100.0000
```

2.3.2 幅相频率曲线图(极坐标图)：nyquist()

函数调用格式如下

(1) nyquist(num,den)：可绘制出以连续时间多项式传递函数表示的系统的极坐标图。

(2) nyquist(num,den,w)：可利用指定的角频率矢量绘制出系统的极坐标图。

(3) 当不带返回参数时，直接在屏幕上绘制出系统的极坐标图(图上用箭头表示w的变化方向，负无穷到正无穷)。

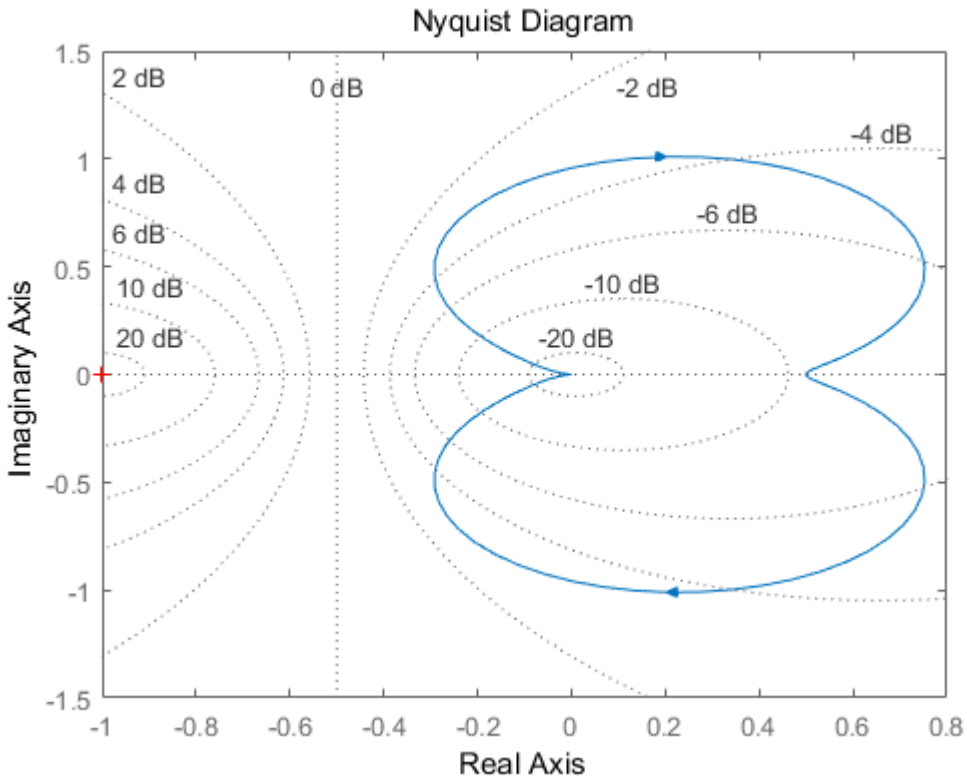
(4) [re,im,w]= nyquist(num,den,w)返回实部re和虚部im及角频率点w矢量（为正的部分）。

然后用plot(re,im)绘制出对应w从负无穷到零变化的部分。

实验代码

```
nyquist(num,den);
grid;
```

结果如下



2.3.3 对数频率曲线图**(**波特图**)**绘制：**bode()**

bode()函数的调用格式为： (1) [m,p,w]=bode(num,den,w)

(2) bode(num,den,w)

(3) bode(num,den)

其中bode(num,den)不需要输入频率，这时该函数会自动地根据模型的变化情况选择一个比较合适的频率范围。

num,den和前面的叙述一样，

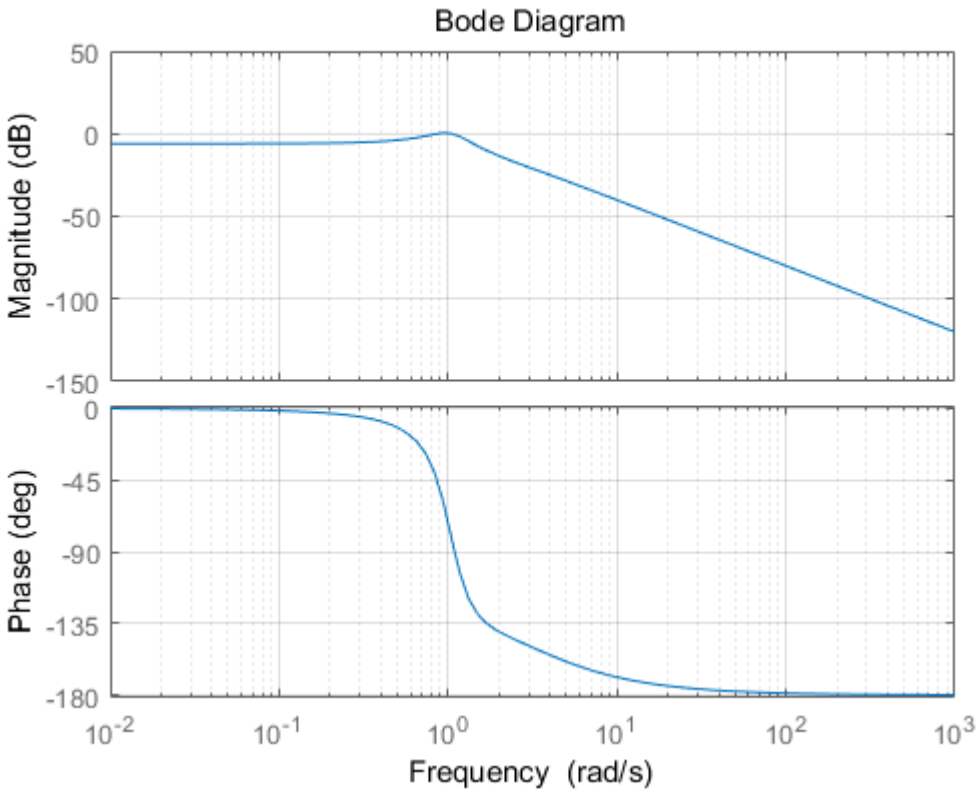
w为频率点构成的向量，该向量最好由logspace()函数构成。

m,p,w分别代表Bode响应的幅值向量、相位向量和角频率。相角以度为单位，幅值可转换为分贝单位，格式为：magdb=20*log (mag)

输入以下代码

```
bode(num,den);
grid;
```

结果如下



2.4 控制系统的根轨迹图

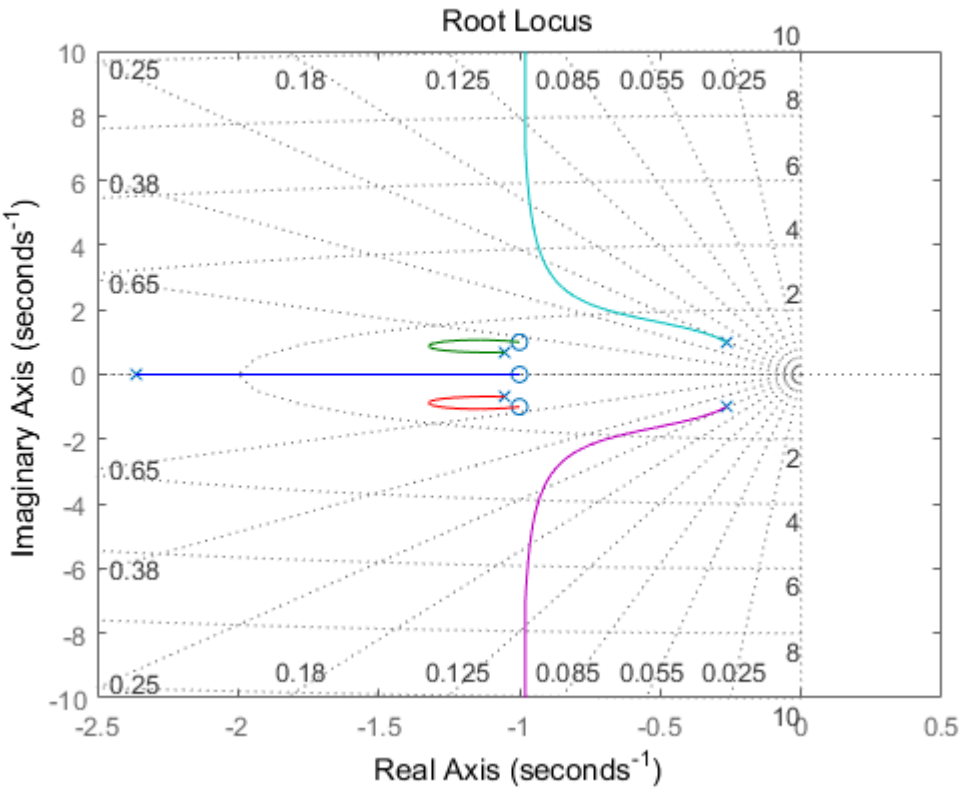
函数调用格式如下：

(1) rlocus（num，den） (2) rlocus(num,den,k) (3) r=rlocus(num,den,k) 或者[r,k]=rlocus(num,den)

函数功能:利用该命令，可以在屏幕上得到画出的根轨迹图。增益向量K可以手动输入也可以自动被确定。

```
rlocus(num,den);
grid;
```

结果如下



2.5 系统稳定性分析

(1) ii=find(条件式):用来求取满足条件的向量的下标向量，以列向量表示

(2) real(p>0):其含义就是找出极点向量p中满足实部的值大于0的所有元素下标.

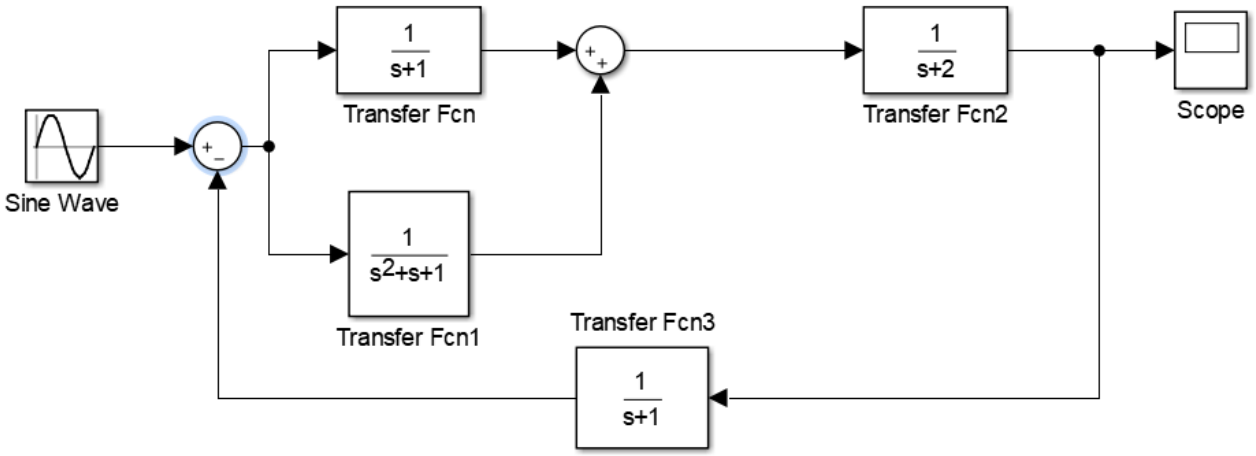
输入以下代码

```
ii=find(real(p)>0)
n1=length(ii)
if(n1>0)
    disp(['System is unstable, with ' int2str(n1) 'unstable poles']);
else
    disp('Syatem is stable');
end
```

实验结果如下

```
n1 =  0
Syatem is stable
```

2.6 simulink仿真



在搭建好仿真模型后提示有错误，不知道如何修改...所以这次仿真中途夭折。

3. 题目二

单位反馈系统的开环传函如下图所示：

$$G(s)=\frac{K}{s(s+3)(s^2+2s+2)}$$

3.1 编写代码

由于题目一都将各函数的原理叙述了一遍，本题就只展示代码及分析结果，后再附上Simulink仿真环节

```
clc,clear;
num1=[1];
den1=[1 0];

num2=[1];
den2=[1 3];

num3=[1];
den3=[1 2 2];

[num4,den4]=series(num1,den1,num2,den2);
[numg,deng]=series(num3,den3,num4,den4);
[num,den]=cloop(numg,deng,-1);
sys1=tf(num,den)

[z,p,k]=tf2zp(num,den);
```

```
sys2=zpk(z,p,k,-1);

[r,p1,k2]=residue(num,den);

figure(1)
subplot(2,3,1);
step(num,den);
grid

subplot(2,3,2);
impulse(num,den)
grid

subplot(2,3,3);
t = 0:0.01:10;
u = sin(t);
lsim(num,den,u,t)
grid

subplot(2,3,4);
t = 0:0.01:10;
u = log(t+1);
lsim(num,den,u,t)
grid

subplot(2,3,5);
t = 0:0.01:10;
u = exp(-0.05*t);
lsim(num,den,u,t)
grid

subplot(2,3,6);
[u,t] = gensig('square',1,10,0.1);
lsim(num,den,u,t)
grid


figure(2)
subplot(2,2,1);
pzmap(num,den)
grid

subplot(2,2,2);
rlocus(num,den);
grid

subplot(2,2,3);
nyquist(num,den)
grid

subplot(2,2,4);
bode(num,den)
grid
```

```
ii=find(real(p)>0);
n1=length(ii)
if(n1>0)
    disp(['System is unstable, with ' int2str(n1) 'unstable poles']);
else
    disp('Syatem is stable');
end
```

3.2 结果展示

3.2.1 结果1

(1) 1图展示了该系统在单位阶跃信号输入下的响应

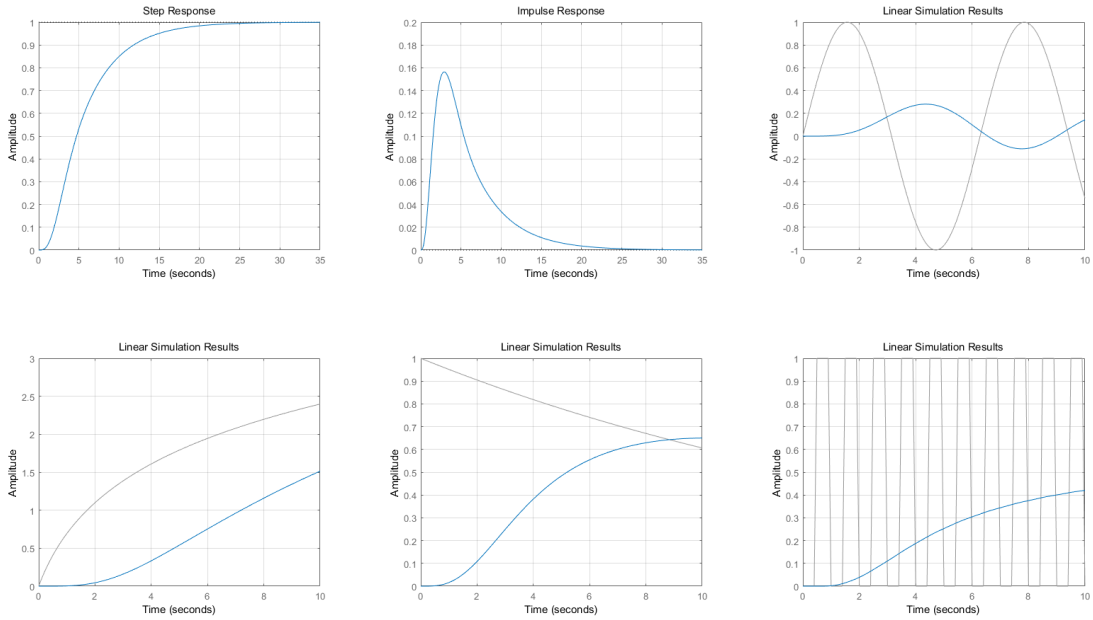
(2) 2图展示了该系统在脉冲信号输入下的响应

(3) 3图展示了该系统在正弦信号sin(t)输入下的响应

(4) 4图展示了该系统在对数信号log(t+1)信号输入下的响应

(5) 5图展示了该系统在指数信号exp(-0.05*t)信号输入下的响应

(6) 6图展示了该系统在方波信号gensig('square',1,10,0.1)脉冲信号输入下的响应



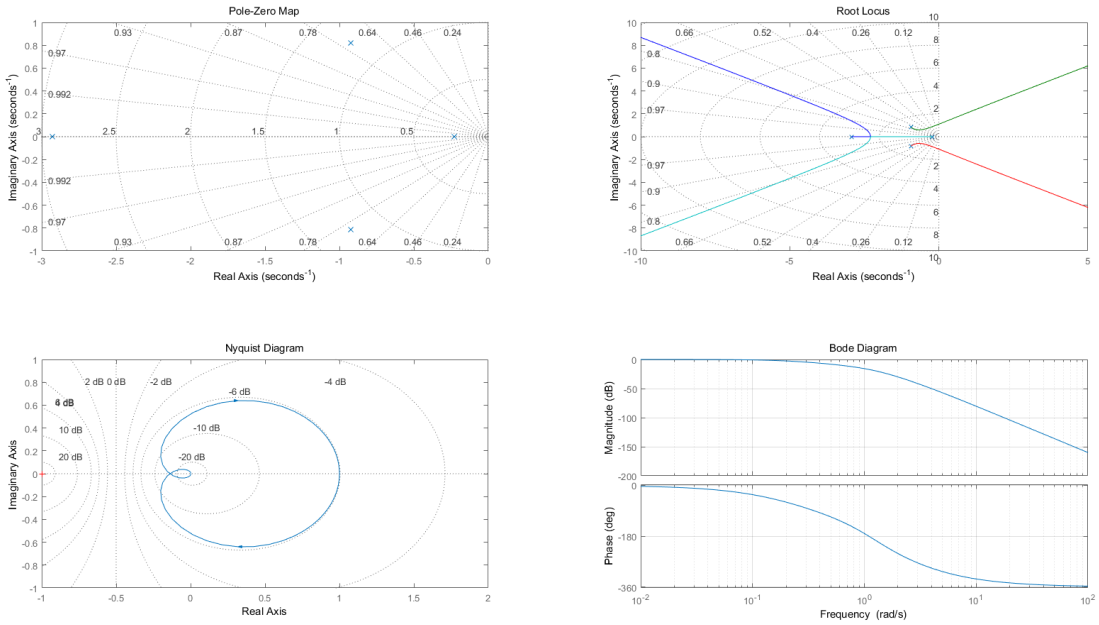
3.2.2 结果2

(1) 1图展示了该系统的零极点

(2) 2图展示了该系统的根轨迹

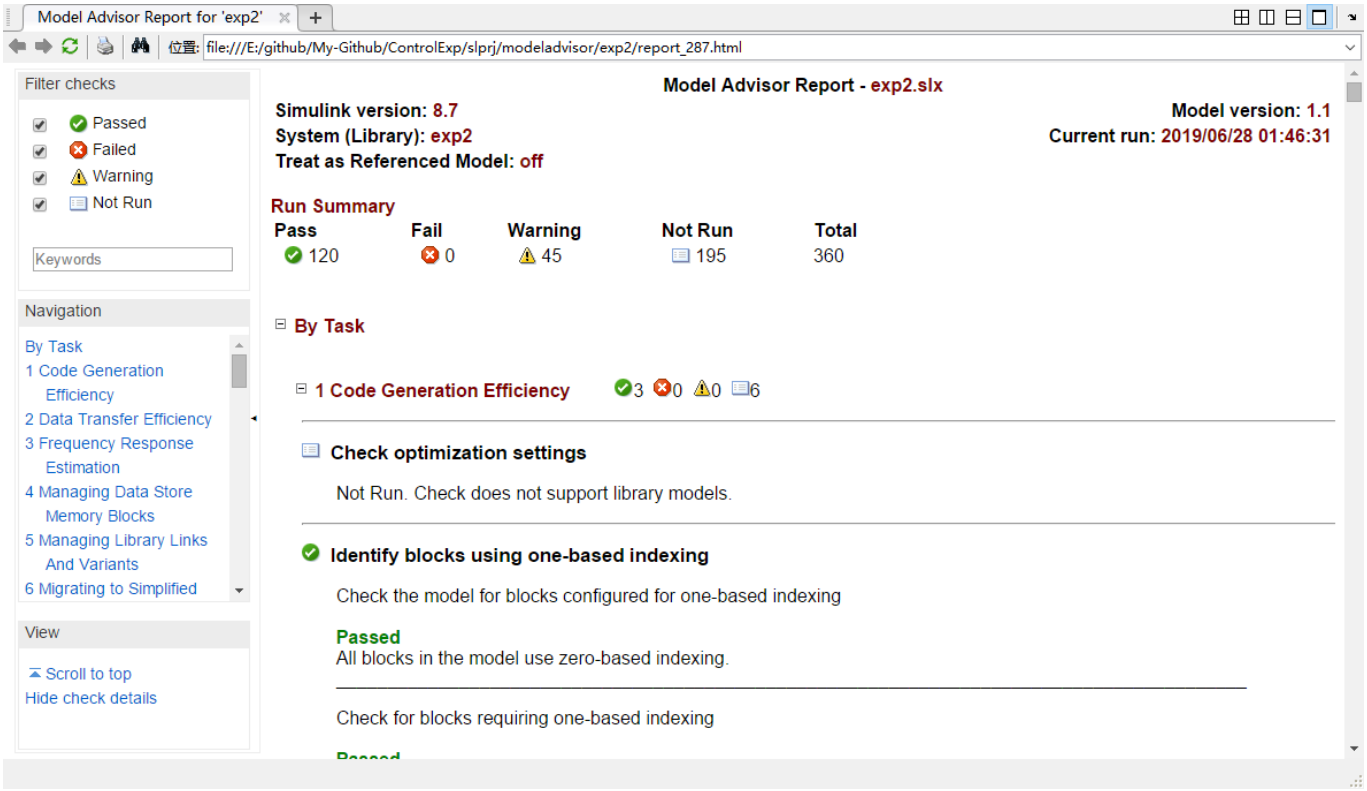
(3) 3图展示了该系统的幅相频率曲线图(极坐标图)

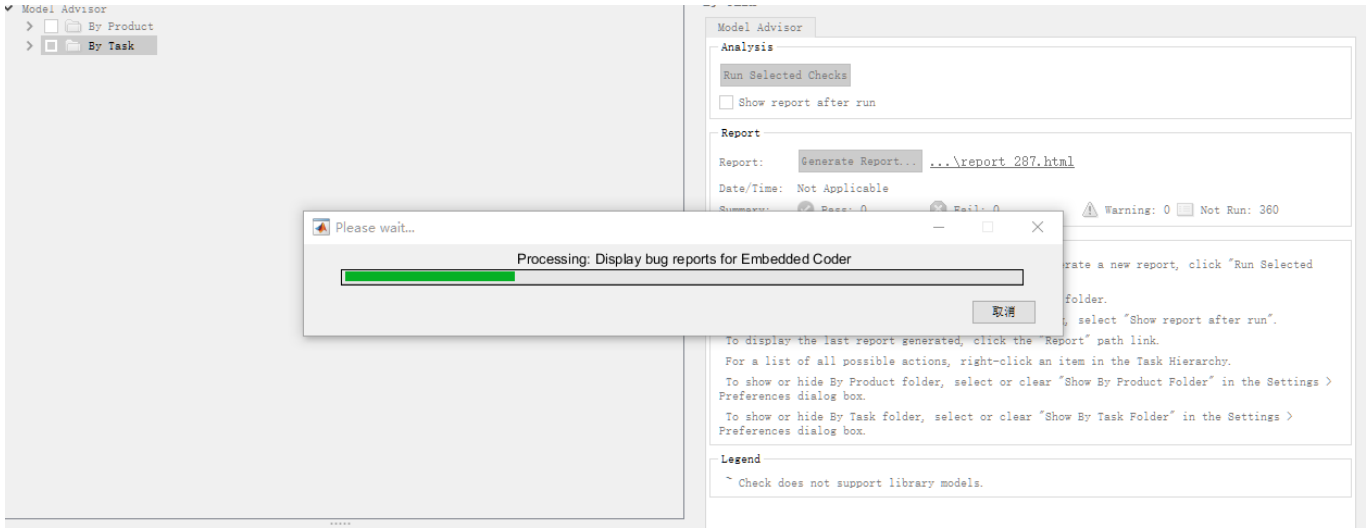
(4) 4图展示了该系统的对数频率曲线图(波特图)

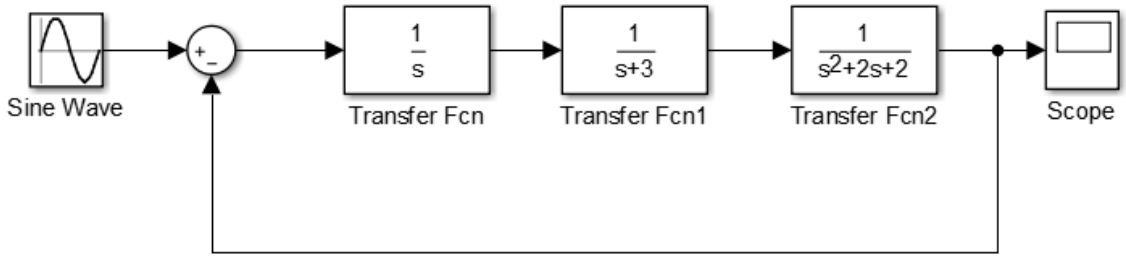


3.3 Simulink仿真

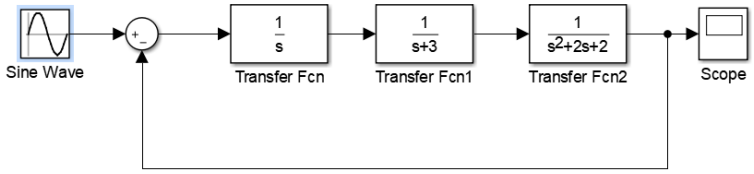
3.3.1 仿真连接

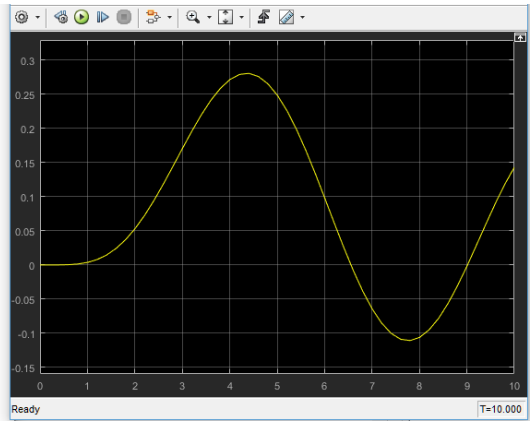



Model Advisor
By Product
By Task
Please wait...
Processing: Display bug reports for Embedded Coder
取消
Model Advisor
Analysis
Run Selected Checks
Show report after run
Report
Report:
Generate Report...
...\report_287.html
Date/Time: Not Applicable
Warning: 0
Not Run: 360
To display the last report generated, click the "Report" path link.
For a list of all possible actions, right-click an item in the Task Hierarchy.
To show or hide By Product folder, select or clear "Show By Product Folder" in the Settings > Preferences dialog box.
To show or hide By Task folder, select or clear "Show By Task Folder" in the Settings > Preferences dialog box.
Legend
Check does not support library models.


Sine Wave
1
s
Transfer Fcn
1
s+3
Transfer Fcn1
1
s2+2s+2
Transfer Fcn2
Scope

3.3.2 仿真结果


Sine Wave
1
s
Transfer Fcn
1
s+3
Transfer Fcn1
1
s2+2s+2
Transfer Fcn2
Scope


0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
1
2
3
4
5
6
7
8
9
10
Ready
T=10.000