第2章 基于MATLAB/Simulink建立控制系统数学模型

【2-1】 建立系统传递函数的多项式模型

Matlab代码：

num1=5\*conv(conv([1,2],[1,2]),[1,6,7]);

den1=conv(conv(conv(conv([1,0],[1,1]),[1,1]),[1,1]),[1,0,2,1]);

den2=conv(conv([1,0],[1,1]),[1,4,4]);

g1=tf(num1,den1)

g2=tf(5,den2)

输出结果：

g1 =

5 s^4 + 50 s^3 + 175 s^2 + 260 s + 140

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

s^7 + 3 s^6 + 5 s^5 + 8 s^4 + 9 s^3 + 5 s^2 + s

Continuous-time transfer function.

g2 =

5

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

s^4 + 5 s^3 + 8 s^2 + 4 s

Continuous-time transfer function.

【2-2】 建立控制系统的零极点模型：

Matlab代码：

k=8;

z=[1j-1,-1j-1];

p=[0,0,-5,-6,-1j,1j];

G=zpk(z,p,k)

输出结果：

G =

8 (s^2 + 2s + 2)

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

s^2 (s+5) (s+6) (s^2 + 1)

Continuous-time zero/pole/gain model.

【2-3】建立控制系统的多项式模型。

Matlab代码：

k=8;

z=[-1,-2];

p=[0 -5 -6 -3];

[num,den]=zp2tf(z(:),p(:),k);

g=tf(num,den)

输出结果：

g =

8 s^2 + 24 s + 16

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

s^4 + 14 s^3 + 63 s^2 + 90 s

Continuous-time transfer function.

【2-4】已知系统前向通道的传递函数，求它的单位负反馈传递函数。

Matlab代码：

num0=[2,1];

den0=[1,2,3];

[num,den]=feedback(num0,den0,1,1);

g=tf(num,den)

输出结果：

g =

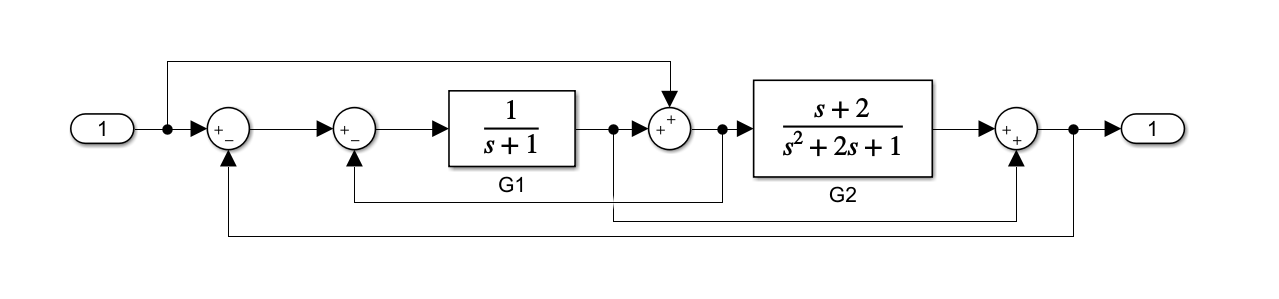
2 s + 1

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

s^2 + 4 s + 4

Continuous-time transfer function.

【2-5】已知系统结构图：



（1）使用Simulink建立系统模型

Simulink建立模型结果截图如上图

（2）在matlab中用梅森公式求系统的传递函数。提示使用factor()函数

分析：

factor()函数可对括号内公式因式分解。图中系统用梅森公式进行分析：共有5个回路，Δ=1+2\*g1+g1\*g2，两条前向通路，p1=g2,Δ1=1,p2=g1\*g2,Δ2=1，故根据梅森增益公式，g=(g1g2+g2)/(1+2\*g1+g1\*g2)，利用factor()函数进行化简。

Matlab代码：

syms s

g1=1/s+1;

g2=(s+2)/(s^2+2\*s+1);

g=factor(((g1+1)\*g2)/(1+2\*g1+g1\*g2))

输出结果：

g =

[ 2\*s + 1, s + 2, 1/(s + 1), 1/(3\*s^2 + 6\*s + 4)]

（3）在matlab中用结构图化简法求系统的传递函数。提示使用linmod()函数

分析：

linmod()函数用于将simulink中模型进行线性化。

Matlab代码：

[num,den]=linmod('t2\_5\_1')

printsys(num,den,'s')

输出结果：

num =

0 1 4 4

den =

1.0000 5.0000 8.0000 5.0000

num/den =

s^2 + 4 s + 4

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

s^3 + 5 s^2 + 8 s + 5

第3章 基于MATLAB控制系统的时域分析法

【3-1】试做出以下系统的阶跃响应，并于原系统的阶跃响应曲线进行比较，做出实验结果分析。

（1）系统有零点情况：z=-5

Matlab代码：

G=tf(10,[1,2,10]);

G1=tf([2,10],[1,2,10]);

G2=tf([1,0.5,10],[1,2,10]);

G3=tf([1,0.5,0],[1,2,10]);

G4=tf(1,[1,2,10]);

step(G)

hold on

step(G1)

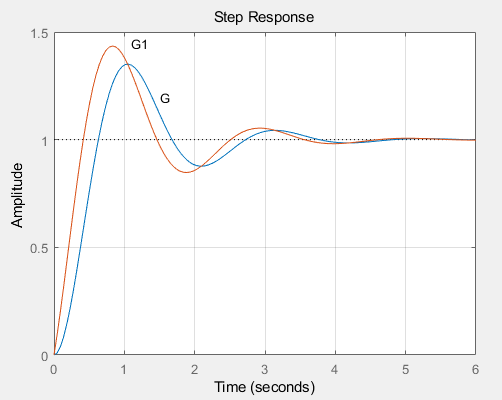
hold on

text(1.5,1.2,'G')

text(1.1,1.45,'G1')

grid on

输出结果：



（2）分子与分母多项式阶数相同：n=m=2

Matlab代码：

G=tf(10,[1,2,10]);

G1=tf([2,10],[1,2,10]);

G2=tf([1,0.5,10],[1,2,10]);

G3=tf([1,0.5,0],[1,2,10]);

G4=tf(1,[1,2,10]);

step(G)

hold on

step(G2)

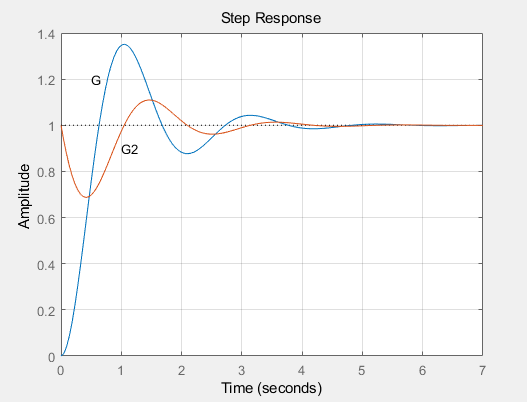
hold on

text(0.5,1.2,'G')

text(1,0.9,'G2')

grid on

输出结果：



（3）分子多项式零次项系数为0

Matlab代码：

G=tf(10,[1,2,10]);

G1=tf([2,10],[1,2,10]);

G2=tf([1,0.5,10],[1,2,10]);

G3=tf([1,0.5,0],[1,2,10]);

G4=tf(1,[1,2,10]);

step(G)

hold on

step(G3)

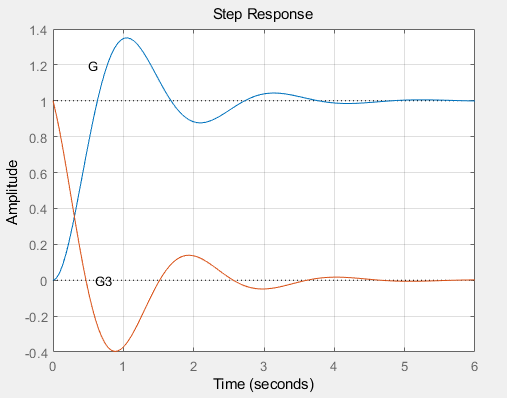
hold on

text(0.5,1.2,'G')

text(0.6,0,'G3')

grid on

输出结果：



（4）原系统的微分相应，微分系数为1/10

Matlab代码：

G=tf(10,[1,2,10]);

G1=tf([2,10],[1,2,10]);

G2=tf([1,0.5,10],[1,2,10]);

G3=tf([1,0.5,0],[1,2,10]);

G4=tf(1,[1,2,10]);

step(G)

hold on

step(G4)

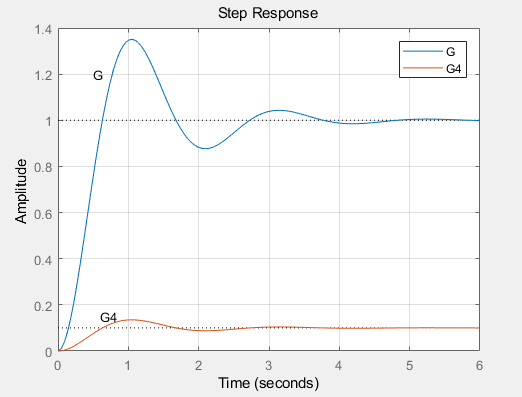
hold on

text(0.5,1.2,'G')

text(0.6,0.15,'G4')

grid on

输出结果：



分析

【3-2】已知控制系统的闭环传递函数

（1）用MATLAB软件分析该系统的单位阶跃响应及其动态性能指标

Matlab代码：

num=1.05\*[0.4762 1];

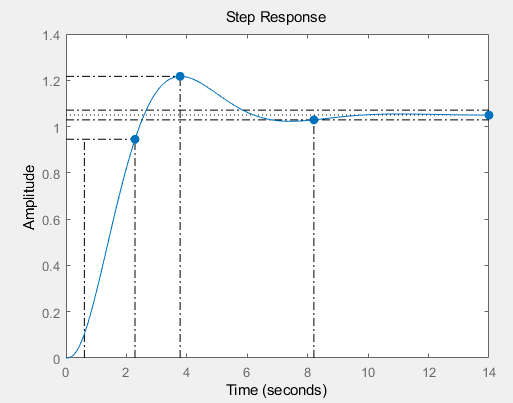
den=conv([0.125 1],conv([0.5 1],[1,1,1]));

g=tf(num,den);

step(g)

hold on

输出结果：



分析

tr=1.68s ts=8.2s tp=3.78s σ%=15.9%

（2）将该系统的阶跃响应与二阶系统的单位阶跃响应比较分析闭环系统主导极点的特点和作用

Matlab代码：

num=1.05\*[0.4762 1];

den=conv([0.125 1],conv([0.5 1],[1,1,1]));

g=tf(num,den);

g2=tf(1.05,[1,1,1]);

step(g)

hold on

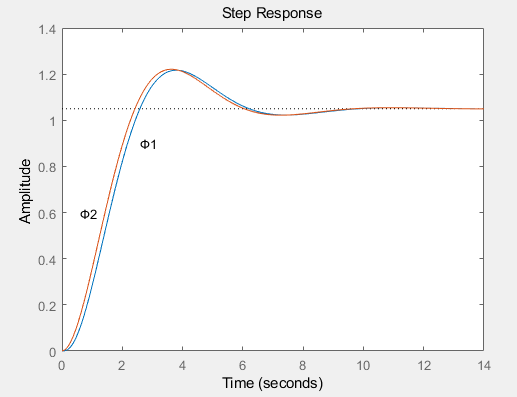
step(g2)

text(2.6,0.9,'Ф1')

text(0.6,0.6,'Ф2')

hold on

输出结果：



分析

（3）比较系统和的单位阶跃响应及其动态性能指标，观察闭环零点对系统动态性能产生的影响有哪些？

Matlab代码：

den1=conv(conv([0.125,1],[0.5,1]),[1,1,1]);

g1=tf(1.05,den1);

g2=tf(1.05\*[1,1],den1);

step(g1)

hold on

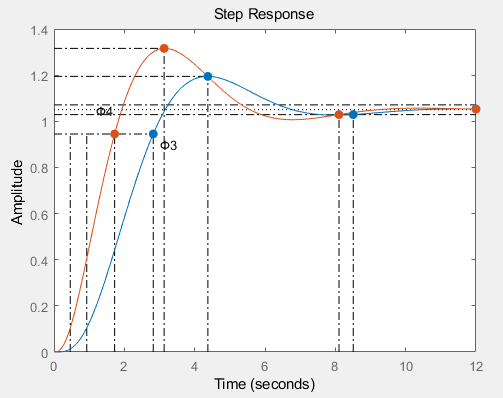
step(g2)

hold on

text(1.2,1.05,'Φ4')

text(3,0.9,'Φ3')

输出结果：



分析

Φ3：tr=1.89s tp=4.37s ts=8.51s σ%=13.8%

Φ4：tr=1.26s tp=3.13s ts=8.1s σ%=25.3%

（4）比较系统 和 的单位阶跃响应及其动态性能指标，分析主导极点对系统动态性能的影响及作用

Matlab代码：

num=1.05\*[0.4762,1];

den2=conv([1,1,1],[0.5,1]);

den1=conv(den2,[0.125,1]);

g1=tf(num,den1);

g2=tf(num,den2);

step(g1)

hold on

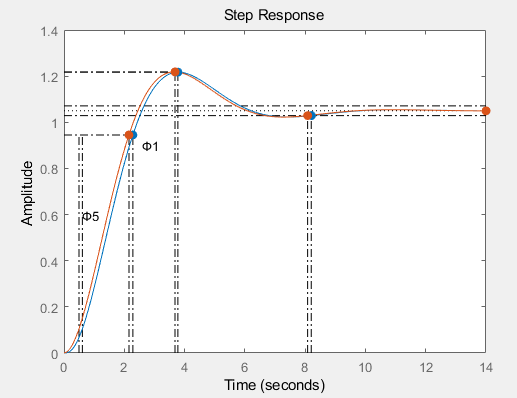
step(g2)

hold on

text(2.6,0.9,'Φ1')

text(0.6,0.6,'Φ5')

输出结果：



分析

Φ1：tr=1.68s tp=3.78s ts=8.2s σ%=15.9%

Φ5：tr=1.66s tp=3.68s ts=8.08s σ%=16%

（5）比较系统和的动态性能指标，分析偶极子对系统动态性能的影响及作用

Matlab代码：

den1=conv([1,1,1],[0.5,1]);

g1=tf(1.05\*[0.4762,1],den1);

g2=tf(1.05,[1,1,1]);

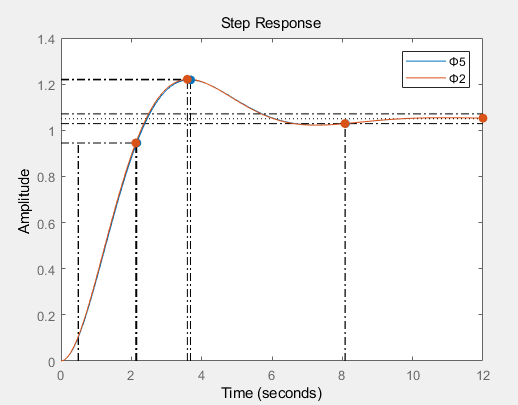
step(g1)

hold on

step(g2)

hold on

输出结果：



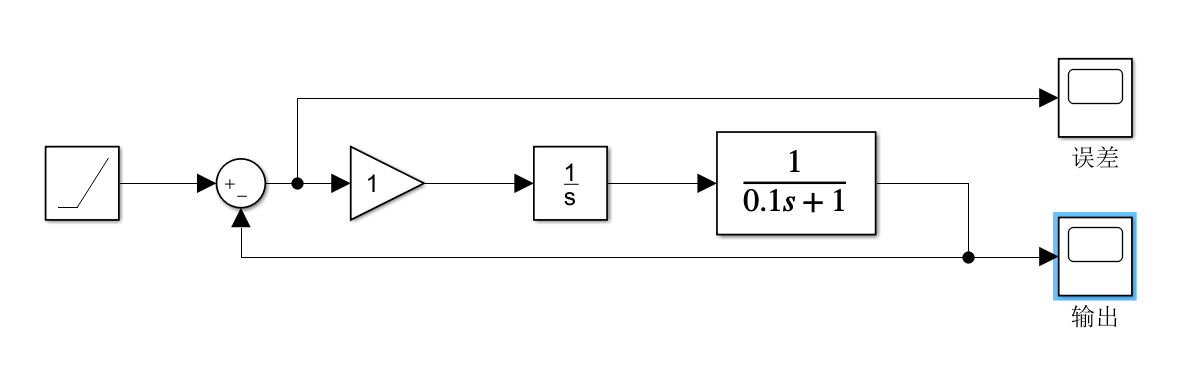
分析

Φ2：tr=1.64s tp=3.59s ts=8.08s σ%=16.3%

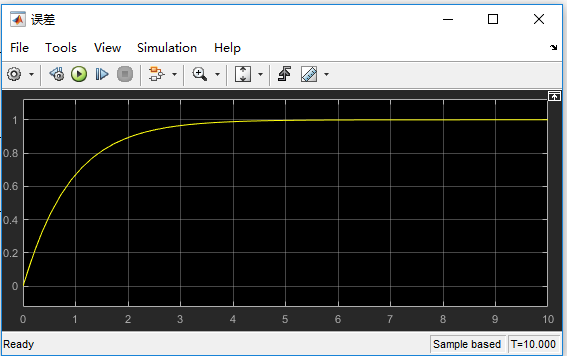
Φ5：tr=1.66s tp=3.68s ts=8.08s σ%=16%

【3-3】将范例3-5系统的单位阶跃响应输入信号改换成单位斜坡输入信号，重新仿真运行，分别观察K=0.1和K=1时，系统单位斜坡响应曲线并求单位斜坡响应稳态误差。并对实验结果曲线进行分析。

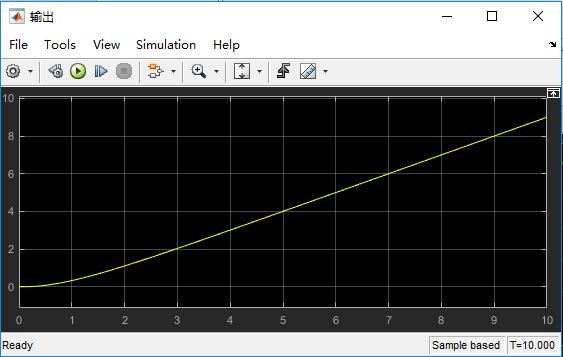
当K=0.1时，simulink模型如下图所示：



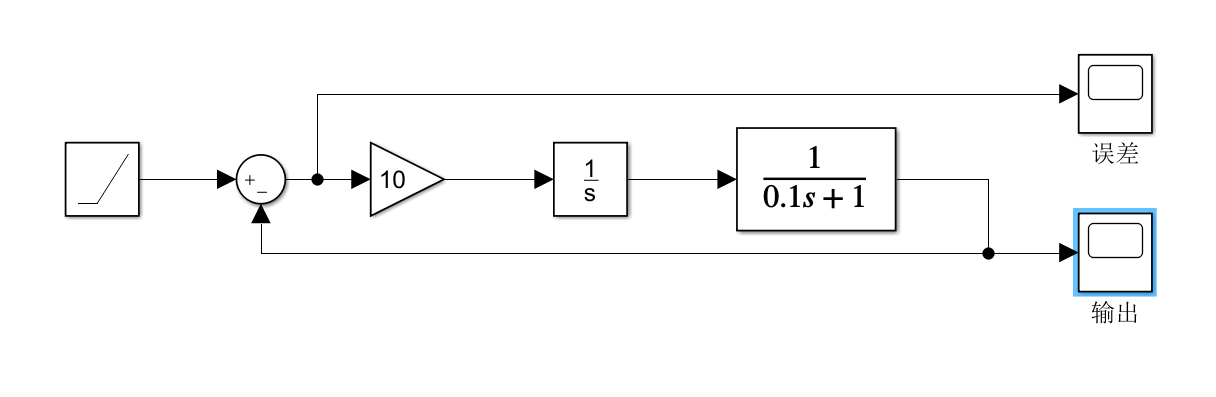
误差信号曲线如下图所示：



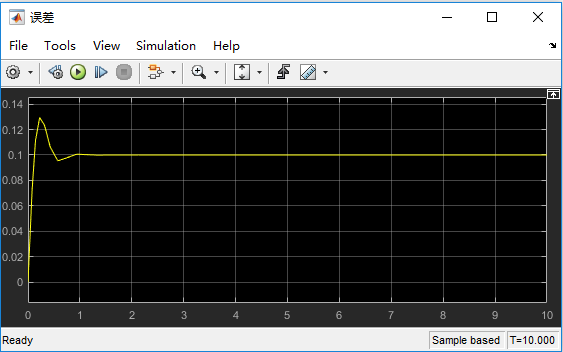
输出信号曲线如下图所示：



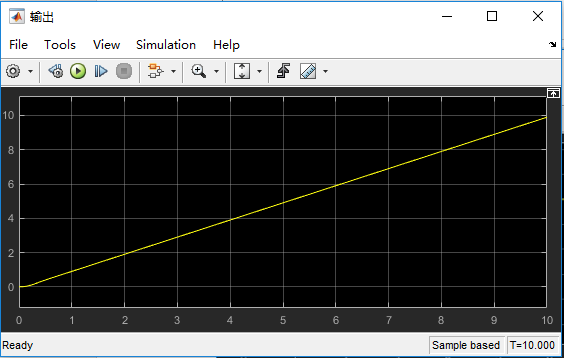
当K=1时，simulink模型如下图所示：



误差信号曲线如下图所示：



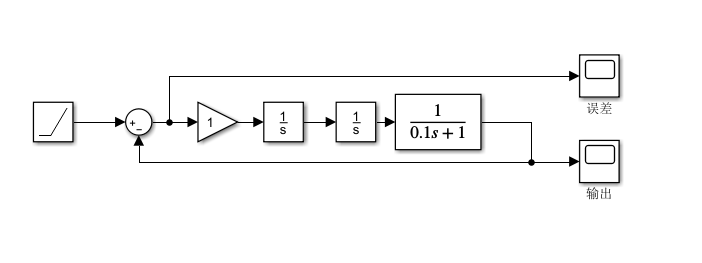
输出信号曲线如下图所示：



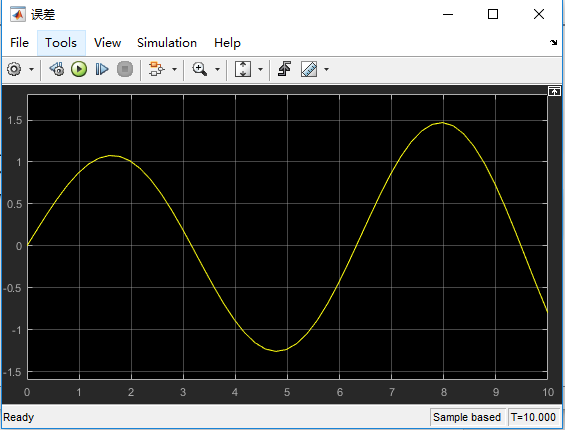
分析：

【3-4】将范例3-5中系统的开环增益改为1，在其前向通道中再增加一个积分环节，系统变成Ⅱ型系统。在输入端给定单位斜坡信号，重新仿真运行，在示波器scope中观察系统响应曲线。并对曲线结果进行分析。

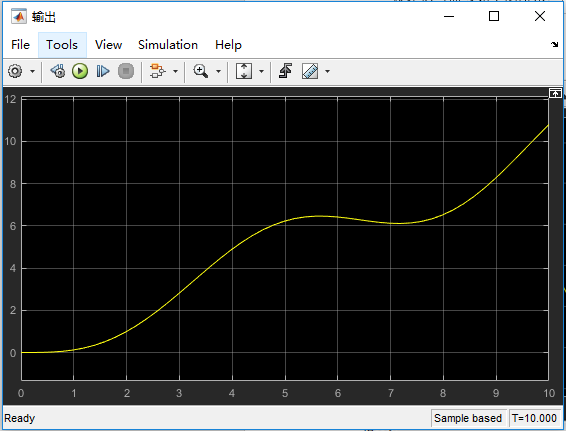
Simulink建立模型如下图所示：



误差信号曲线如下图所示：



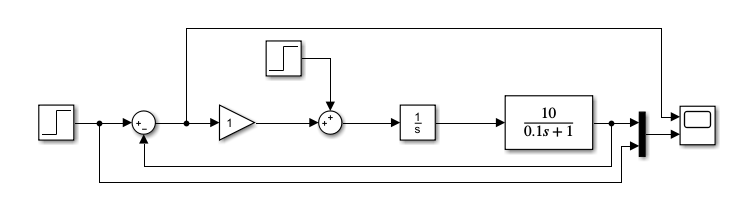
输出波形曲线如下图所示：



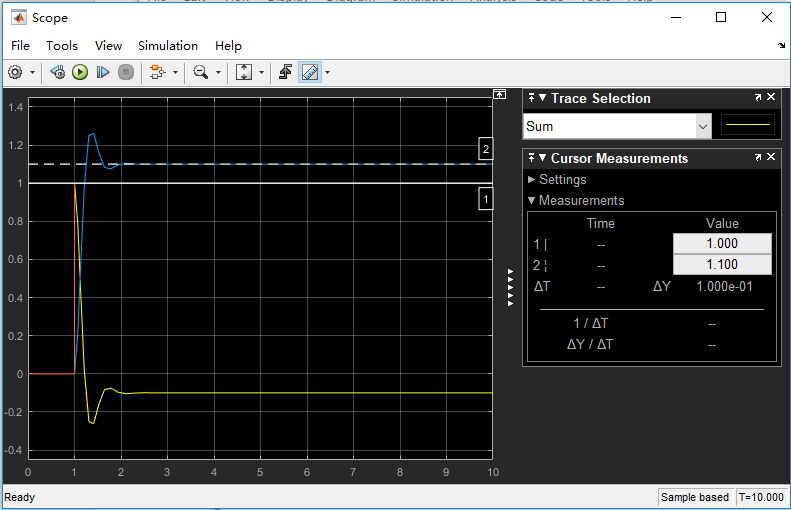
分析：

【3-5】在simulink中建立如下系统，若输入信号，扰动信号，令，求系统的总的稳态误差。

Simulink建立系统模型如下图所示：



系统的输入误差输出对比波形图如下图所示：



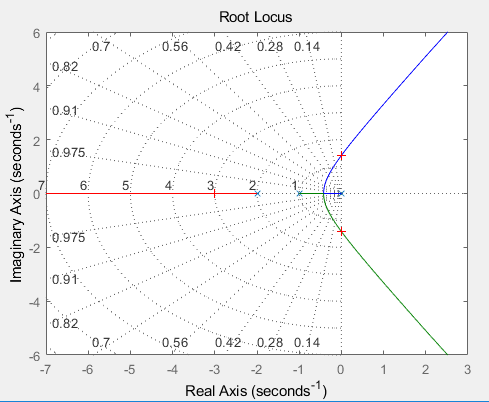
由图可知，系统总的稳态误差为0.1

第4章 基于MATLAB控制系统的根轨迹法

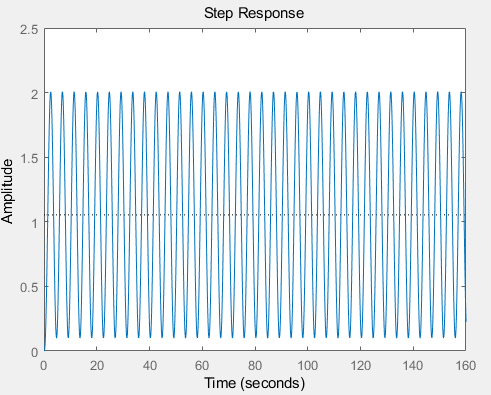
【4-1】在例4-2中控制系统的根轨迹上分区段取点，构造闭环系统传递函数，分别绘制其对应系统的阶跃响应曲线，并比较分析；及将数据填入实验数据记录表中。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 阻尼比 | 闭环极点p | 开环增益k | 自然频率wn | 超调量σ% | 调节时间ts |
| ξ=0 | -2.9990+0.0000i  0.0000+1.4131i  0.0000-1.4131i | 6.3095 | 1.5 |  |  |
| ξ=0.25 | -2.5263+0.0000i  -0.2368+0.8645i  -0.2368-0.8645i | 2.0297 | 0.939 | 39.5% | 16 |
| ξ=0.7 | -2.3010+0.0000i  -0.3495+0.5192i  -0.3495-0.5192i | 0.9012 | 0.666 | 11.5% | 9.82 |
| ξ=1 | -2.1547  -0.4242  -0.4211 | 0.3849 | 0.424 | 0% | 14.3 |
|  |  | 0.2 |  |  |  |

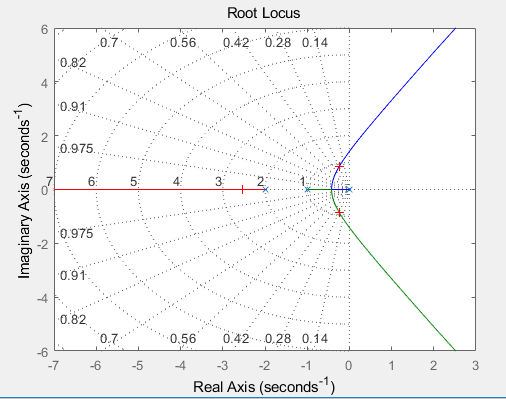
ξ=0时，根轨迹图如下图：



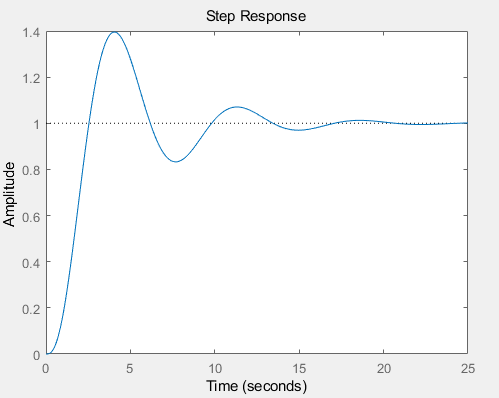
对应的阶跃响应曲线如下图：



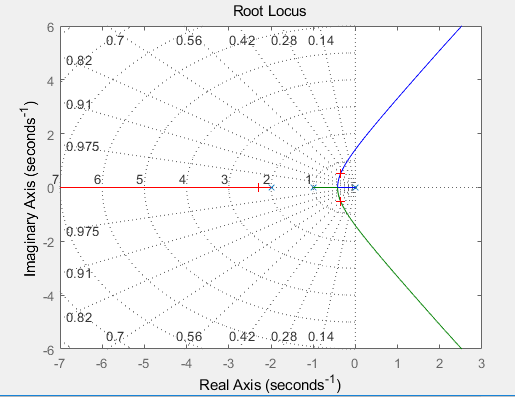
ξ=0.25时，根轨迹图如下图：



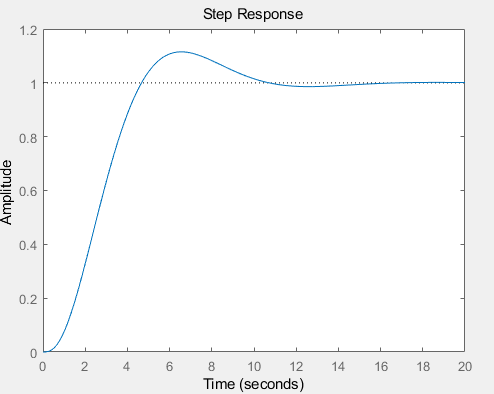
对应系统的阶跃响应曲线如下图：



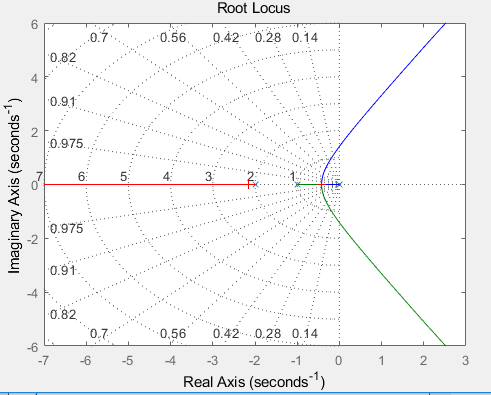
ξ=0.7时，根轨迹图如下图：



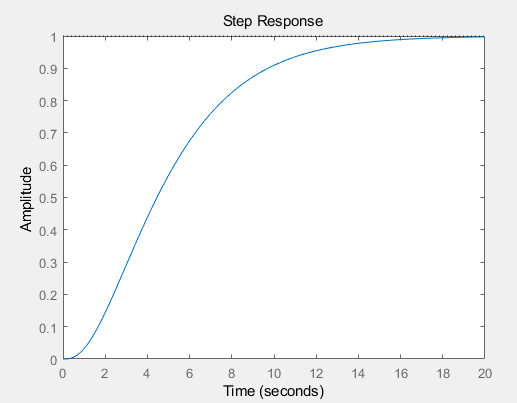
对应系统的阶跃响应曲线如下图：



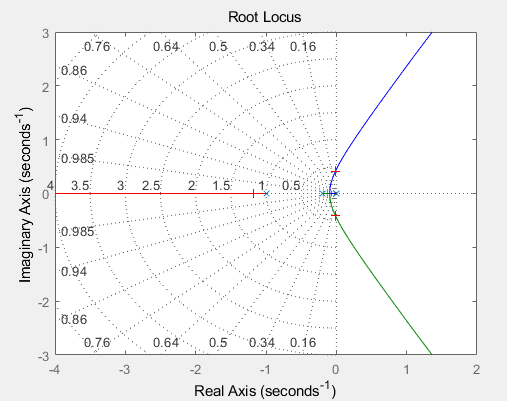
ξ=1时，根轨迹图如下图：



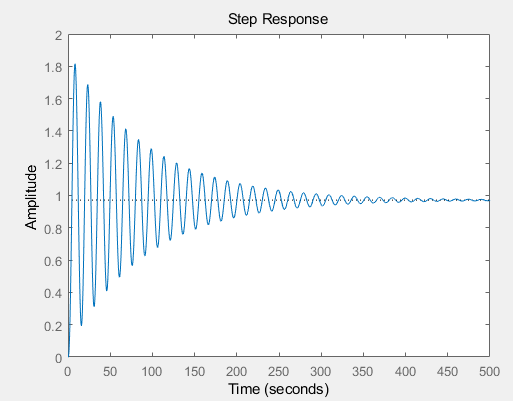
对应系统的阶跃响应曲线如下图：



K=0.2时，根轨迹图如下图所示：



对应系统的阶跃响应曲线如下图：



分析：

【4-2】已知开环系统传递函数比较增加一个开环几点s=-2后，，观察根轨迹及其闭环单位阶跃响应的变化。

Matlab代码：

k=1;

z=[];

p=[0 -1];

G=zpk(z,p,k);

rlocus(G);

hold on

k=1;

z=[-2];

p=[0 -1];

G1=zpk(z,p,k);

figure(3);

rlocus(G1);

hold off

figure(2);

sys=feedback(G,1);

step(sys);

hold on

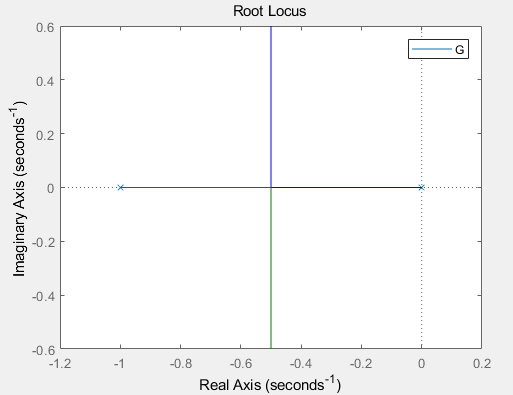
sys1=feedback(G1,1);

step(sys1);

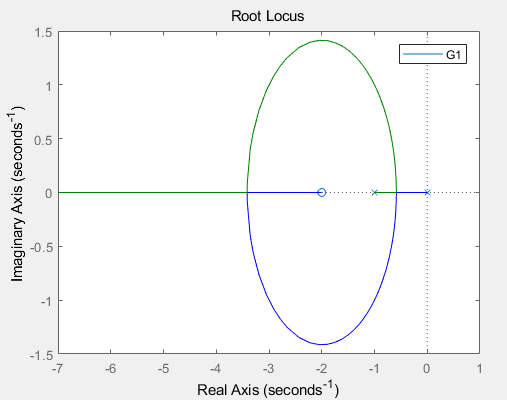
hold off

输出结果：

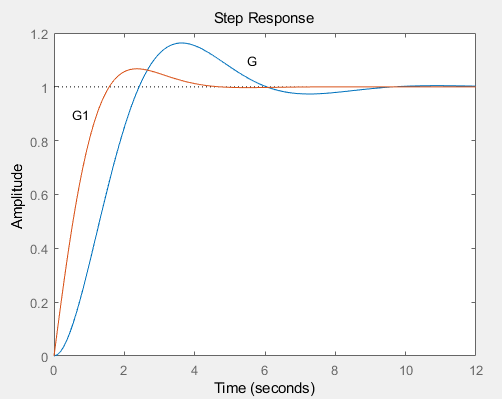
G(s)根轨迹曲线如下图：



G1(s)根轨迹曲线如下图：



阶跃响应曲线如下图：



分析：

【4-3】已知系统开环传递函数

要求用根轨迹法设计超前校正装置GC1，要求KV>20，希望该单位负反馈系统的时域性能指标σ%<15%，ts<1.5s。

Matlab代码：

s=tf('s');

G=10/(0.5\*s^2+s);

Gc=feedback(G,1);

roots(Gc.den{1});

figure

step(Gc)

rltool(Gc)

调用运行rltool（）。