

第九章 控制系统的分析

控制系统的分析是系统设计的重要步骤之一，主要表现在以下两个方面：

1. 在设计控制器前要分析系统的不可变部分，确定原系统在哪些方面的性能指标不满足设计要求，有针对性的去设计控制器；
2. 控制器设计完成后要验证整个闭环系统的性能指标是否满足设计要求。

9.1 线性系统的时域响应

控制系统最常用的分析方法是，当输入信号为单位阶跃和单位冲激函数时，求出系统的响应函数。系统的响应函数是时间 t 的函数，我们称为时域响应。从时域响应可以获得系统从初始时刻开始的任意时刻的响应值，因此时域分析法最能直观地体现系统的各个方面的性能。本节详细介绍与系统时域分析相关函数的用法。

9.1.1 单位阶跃响应

单位阶跃信号是控制系统最常用的典型输入信号。根据系统单位阶跃响应的特征量可以定量的描述系统的快速性、阻尼性和稳定性等性能指标。Matlab 中求取连续系统的单位阶跃响应的函数为 `step`，其使用方法如下。

【调用格式】

<code>step(sys)</code>	%绘制系统 <code>sys</code> 的单位阶跃响应曲线，自动选取时间 <code>t</code>
<code>step(sys,t)</code>	%绘制系统 <code>sys</code> 在时间 <code>t</code> 范围内的单位阶跃响应曲线
<code>step(sys1,sys2,...,sysN)</code>	%绘制多个系统 <code>sys1, ..., sysN</code> 的单位阶跃响应曲线
<code>step(sys1,sys2,...,sysN,t)</code>	
<code>step(sys1,'PlotStyle1')</code>	%可以指定绘制单位阶跃响应曲线的绘图格式
<code>[y,t] = step(sys)</code>	% 带有输出变量的调用格式
<code>[y,t,x] = step(sys)</code>	% 仅适用于状态空间模型
<code>y = step(sys,t)</code>	
<code>step(A,B,C,D)</code>	% 输入变量为状态空间模型
<code>step(A,B,C,D,iu)</code>	% 输出对第 <code>iu</code> 个输入的响应
<code>step(A,B,C,D,iu,t)</code>	
<code>step(num,den)</code>	% 输入变量为传递函数模型
<code>step(num,den,t)</code>	

【说明】

1. 输入可以是 LTI 数学模型，也可以是 LTI 数学模型的属性值。输入变量 `sys` 为任意 LTI 数学模型，`a`, `b`, `c`, `d` 和 `num`, `den` 为模型的属性值。
2. `t` 为一维向量，其元素是单调递增的离散时间；也可以不指定绘图时间区间，此时 Matlab 系统自动选取绘图时间自变量 `t` 的区间。
3. 可以指定绘图的格式字符串，格式串的定义同 `plot` 函数。
4. 当采用无输出变量的调用方式时，将绘制函数的响应曲线；当采用带输出变量的调用方式时，不绘制响应曲线，只将响应数据放入输出变量。
5. 当 `sys` 是单输入单输出 (SISO) 系统时，`y` 是一个列向量；当 `sys` 是多输入多输出 (MIMO) 系统

时, y 是一个三维数组, $y(:,i,j)$ 表示的是第 i 个输出分量对第 j 个输入分量的单位阶跃响应。

6. x 为系统的状态值, 当 sys 是 SISO 系统时, x 是一个列向量, 当 sys 是 MIMO 系统时, x 是一个三维数组, $x(:,i,j)$ 表示的是第 i 个状态分量对第 j 个输入分量的单位阶跃响应过程中的状态值。

例 9.1.1 典型二阶系统传递函数为

$$G_c(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

试分析不同参数下的系统单位阶跃响应。

解: 1、假设将自然频率固定为 $\omega_n = 1$, $\zeta = 0, 0.1, 0.2, 0.3, \dots, 1, 2, 3, 5$ 。运行下面的程序, 得到闭环系统的单位阶跃响应曲线如图 9.1.1 所示。

```
wn=1;zetas=[0:0.1:1,2,3,5];t=0:0.1:12;
hold on
for i=1:length(zetas)
    Gc=tf(wn^2,[1,2*zetas(i)*wn,wn^2]);
    step(Gc,t)
end
grid on, hold off
```

由图9.1.1可知当阻尼比增加时, 系统的振荡程度会减弱; 当阻尼比大于等于1时, 系统响应曲线为单调曲线, 已经没有了振荡。

2、将阻尼比 ζ 的值固定在 $\zeta = 0.55$, 则可以通过下面程序绘制出在各个自然频率 ω_n 下的单位阶跃响应曲线, 如图9.1.2所示。

```
wn=[0.1:0.1:1]; z=0.55; t=0:0.1:12;
hold on
for i=1:length(wn)
    Gc=tf(wn(i)^2,[1,2*z*wn(i),wn(i)^2]);
    step(Gc,t)
end
grid on, hold off
```

由图9.1.2可知当无阻尼自由振荡频率增加时, 系统的响应速度将加快, 而响应曲线的峰值将保持不变, 对其他的阻尼比我们也可以得出相同的结论。

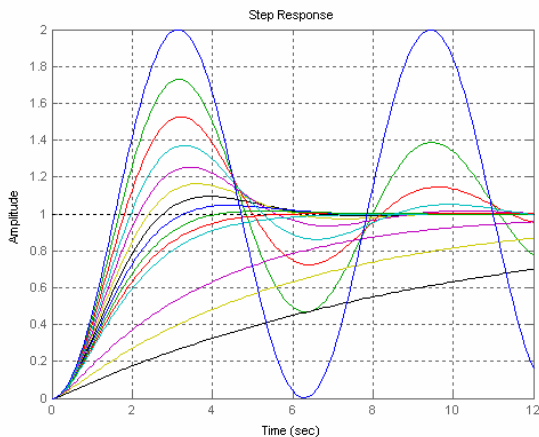


图9.1.1 ζ 变化时系统的单位阶跃响应曲线

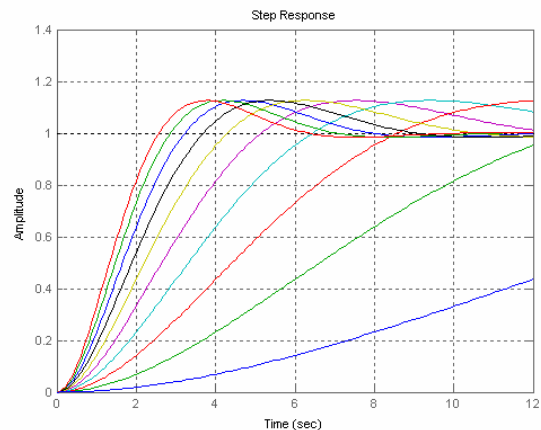


图9.1.2 ω_n 变化时系统的单位阶跃响应曲线

例 9.1.2 控制系统的状态空间模型为

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 6 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

试采用不同的 step 调用方法，绘制系统的单位阶跃响应曲线。

解：运行下列程序语句，仿真得到的响应曲线如图 9.1.3~9.1.5 所示。

a = [-1, -1; 6, 0]; b = [1, 1; 1, 0]; c = [1, 0; 0, 1]; d=[0,0;0,0];

sys = ss(a,b,c,d);

%在一个图像窗口绘制的 4 个子图分别绘制每个输出分量对每个输入分量的响应曲线

step(sys),grid on; %等效于调用 step(a, b, c, d)，图 9.1.3

%绘制输出对第 1 个输入分量的响应曲线

figure, step(sys(:, 1)),grid on; %等效于调用 step(a, b, c, d, 1)，图 9.1.4

%绘制第 1 个输出分量对第 2 个输入分量的响应曲线

figure, step(sys(1, 2)),grid on; %图 9.1.5

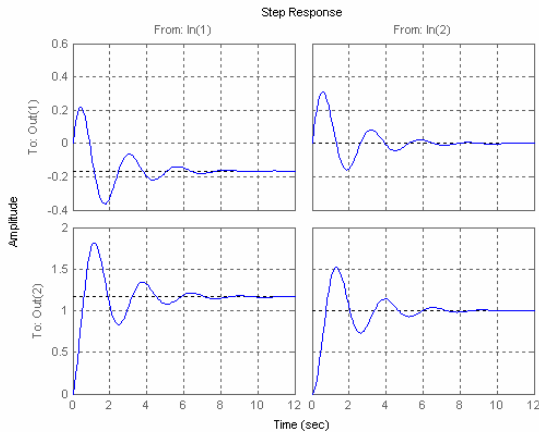


图 9.1.3 MIMO 系统的单位阶跃响应曲线

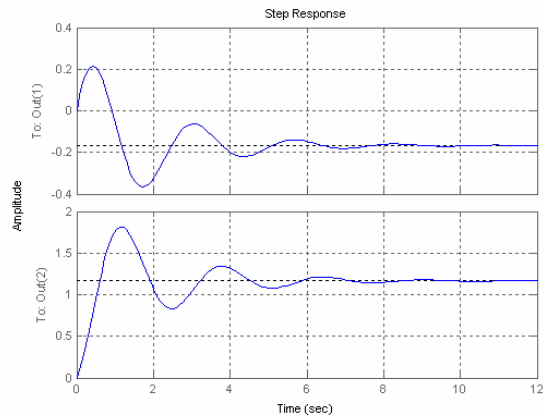


图 9.1.4 MIMO 系统的单位阶跃响应曲线

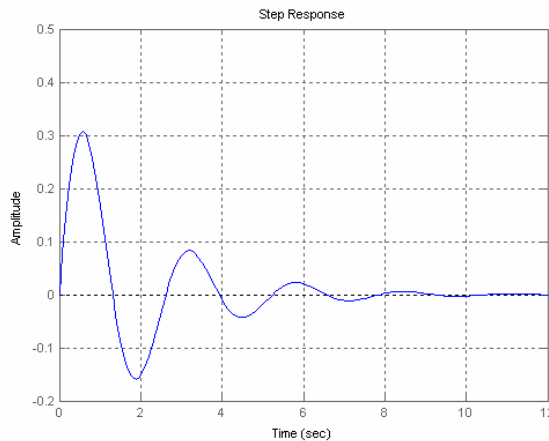


图 9.1.5 MIMO 系统的单位阶跃响应曲线

例9.1.3 系统的传递函数为

$$G(s) = \frac{1}{s^2 + 0.6s + 1}$$

试求取其单位阶跃响应的特征量：超调量，上升时间，调整时间，峰值时间。

解：（1）编写函数求取SISO系统的单位阶跃响应特征量。

【调用格式】

[pos,tr,ts,tp]=stepchar(g0,delta)

【说明】g0 为 SISO 系统的数学模型，delta 为稳态范围宽度，pos 为超调量，tr 为上升时间，ts 为调整时间，tp 为峰值时间。stepchar 函数参见附录 D：D-9-1。

(2) 在 Matlab 窗口键入下面程序，求取系统的单位阶跃响应特征量。

```
sys=tf([1],[1,0.6,1]);  
[pos,tr,ts,tp]=stepchar(sys,0.02)
```

运行结果：

```
pos =  
    37.5896  
tr =  
    2.0245  
ts =  
   11.2270  
tp =  
    3.3129
```

还可以根据单位阶跃响应的特征量的定义，在响应曲线上用鼠标拾取关键点的方法来确定它们的值。可以运行如下程序，其仿真结果如图 9.1.6 所示。

```
sys=tf([1],[1,0.6,1]);  
[y,t]=step(sys); %计算响应数据  
plot(t,y); %绘制响应曲线  
hold on  
range=[0.98*ones(length(t),1),1.02*ones(length(t),1),ones(length(t),1)]; %准备稳态区间直线的数据  
plot(t,range),grid on; %绘制计算响应时间的稳态区间的直线
```

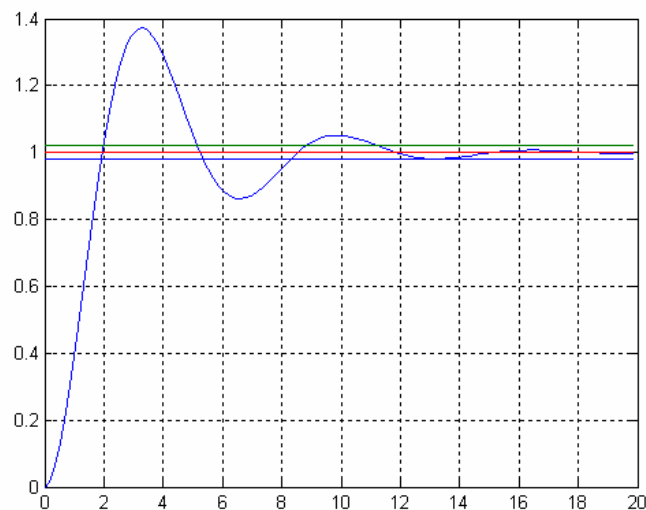


图 9.1.6 在单位阶跃响应曲线上求取性能指标

9.1.2 理想单位脉冲响应

impulse 函数用于求取连续系统理想单位脉冲响应，其使用方法和 step 函数相同。

【调用格式】

```
impulse (sys)  
impulse (sys,t)
```

```

impulse (sys1,sys2,...,sysN)
impulse (sys1,sys2,...,sysN,t)
impulse (sys1,'PlotStyle1)
[y,t] = impulse (sys)
[y,t,x] =impulse (sys)
y = impulse (sys,t)
impulse (A,B,C,D)
impulse (A,B,C,D,iu)
impulse (A,B,C,D,iu,t)
impulse (num,den)
impulse (num,den,t)

```

9.1.3 零输入响应

initial 函数用于求取连续 LTI 系统对初始条件的响应(零输入响应),其使用方法和 step 函数基本相同。

【调用格式】

```

initial(sys,x0)
initial(sys,x0,t)
initial(sys1,sys2,...,sysN,x0)
initial(sys1,sys2,...,sysN,x0,t)
initial(sys1,'PlotStyle1',...,sysN,'PlotStyleN',x0)
[y,t,x] = initial(sys,x0)

```

【说明】输入变量只能是状态空间模型, x0 为初始状态列向量。

例 9.1.4 二阶系统的状态空间描述为

$$\dot{x} = \begin{bmatrix} -0.5572 & -0.7814 \\ 0.7814 & 0 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = [1.9691 \quad 6.4493]x$$

当初始状态为 $x_0=[1;0]$ 时,求系统的零输入响应。

解:运行下面的程序,可以得到如图 9.1.7 所示的响应曲线。

```

a=[-0.5572,-0.7814;0.7814,0]; b=[1;0]; c=[1.9691,6.4493]; d=[0];
x0=[1;0];t=0:0.1:20;
initial(a,b,c,d,x0,t),grid on

```

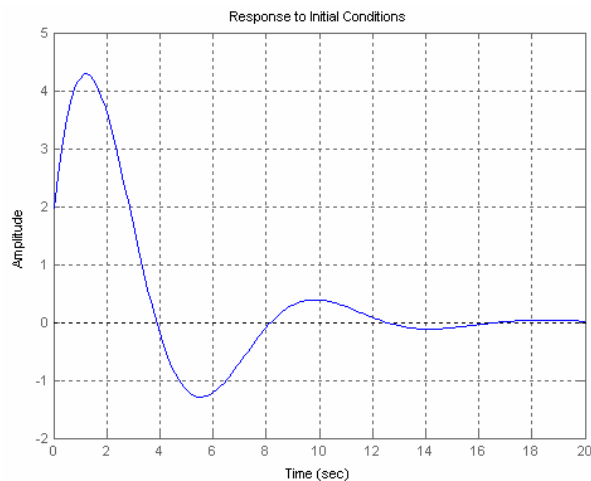


图 9.1.7 系统的零输入响应曲线

9.1.4 任意输入响应

当系统的输入信号不是典型信号时，可以使用 `lsim` 函数求取系统对任意输入信号的响应，其使用方法和参数定义与 `step` 函数基本相同。

【调用格式】

```
lsim(sys,u,t)
lsim(sys,u,t,x0) % 带有初始条件 x0
lsim(sys1,sys2,...,sysN,u,t)
lsim(sys1,sys2,...,sysN,u,t,x0)
lsim(sys1,'PlotStyle1',...,sysN,'PlotStyleN',u,t)
[y,t] = lsim(sys,u,t)
[y,t,x] = lsim(sys,u,t) % 仅适用于状态空间模型
[y,t,x] = lsim(sys,u,t,x0) % 仅适用于状态空间模型
```

【说明】

1. 输入可以是 LTI 数学模型，也可以是 LTI 数学模型的属性值，但是当带有初始条件的时候，只能为状态空间模型
2. 要构造输入信号的离散值，其中 `t` 为离散的时间值，`u` 为和 `t` 一一对应的输入信号幅值。

例 9.1.5 二阶系统的传递函数为

$$G(s) = \frac{16}{s^2 + 3s + 16}$$

当系统的输入信号是幅值为 1，周期为 8 秒的方波时，绘制系统的输出响应曲线。

解：运行下面的程序，可以得到如图 9.1.8 所示的响应曲线。

```
sys=tf([16],[1,3,16]);
[u,t] = gensig('square',8,32,0.1);
lsim(sys,u,t),grid on
```

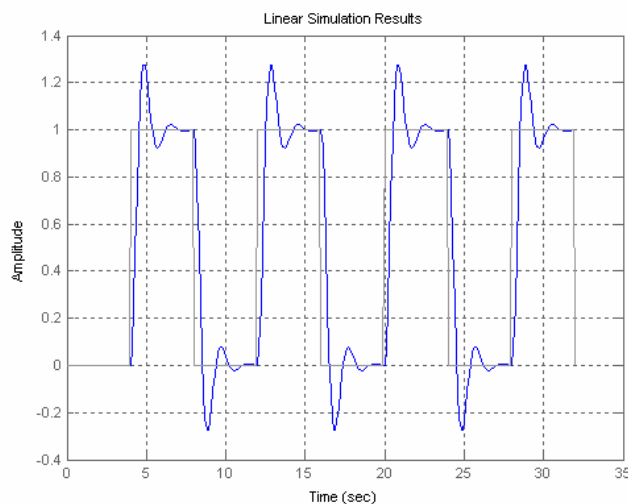


图 9.1.8 二阶系统的方波响应曲线

9.2 线性系统的根轨迹

根轨迹是以 K ($0 \sim +\infty$) 为可变参数，根据开环系统的零极点绘制出来的，因而它能反映出开环系统零极点与闭环系统极点（特征根）之间的关系。利用根轨迹可以分析系统参数和结构已定的系统的时域响应

特性，以及参数变化对时域响应特性的影响，而且还可以根据对时域响应特性的要求确定可变参数及调整开环系统零极点的位置，并改变它们的个数，也就是说根轨迹法可用于解决线性系统的分析与综合问题。

MATLAB 专门提供了绘制根轨迹的函数：`rlocfind`、`pzmap` 等，使绘制根轨迹变得轻松自如。

9.2.1 根轨迹绘制

【调用格式】

`rlocus(sys)` %绘制系统的根轨迹
`rlocus(sys,k)` %使用用户指定的根轨迹增益 K 值来绘制系统的根轨迹
`rlocus(sys1,sys2,...)` %绘制多个系统的根轨迹
`[R,K]=rlocus(sys)` %计算根轨迹增益值和闭环极点值
`R=rlocus(sys,k)` %计算对应于根轨迹增益值 k 的闭环极点值

【说明】

1. `rlocus` 函数绘制以 `k` 为参数的 SISO 系统的轨迹图。
2. 不带输出变量的调用方式将绘制系统的根轨迹。
3. 带有输出变量的调用方法将不绘制根轨迹，只计算根轨迹上各个点的值。`K` 中存放的是根轨迹增益向量；矩阵 `R` 的列数和增益 `K` 的长度相同，它的第 `m` 列元素是对于增益 `K(m)` 的闭环极点。

9.2.2 计算根轨迹增益

【调用格式】

`[K,poles]=rlocfind(sys)` %计算鼠标拾取点处的根轨迹增益和闭环极点
`[K,poles]=rlocfind(sys,P)` %计算最靠近给定闭环极点 P 处的根轨迹增益

【说明】

1. 函数 `rlocfind` 可计算出与根轨迹上极点相对应的根轨迹增益。`rlocfind` 既适用于连续系统，也适用于离散时间系统。
2. `P` 为给定的闭环极点，可以给定多个闭环极点，此时 `P` 为列向量。向量 `K` 的第 `m` 项是根据极点位置 `P(m)` 计算的增益，矩阵 `poles` 的第 `m` 列 `poles(m)` 是相应的闭环极点。

例 9.2.1 负反馈系统的开环传递函数为

$$G(s) = \frac{K}{s(s^2 + 4s + 5)}$$

绘制系统的根轨迹图，并求使系统稳定的 `K` 值范围和使系统无超调的 `K` 值范围。

解：下面的程序可以计算并绘制根轨迹图，如图 9.2.1 所示，然后用 `ginput(3)` 函数在图中选择 3 个极点位置：2 个实轴上的交点，一个虚轴上的交点，再调用 `rlocfind` 函数求这 3 个闭环极点处的增益值 `K`。

```
sys=tf(1,[1,4,5,0])
rlocus(sys)
[x,y]=ginput(3);
p= x+i*y
K=rlocfind(s,p)
```

运行结果：

`K =`

1.8518 2.0000 19.9899

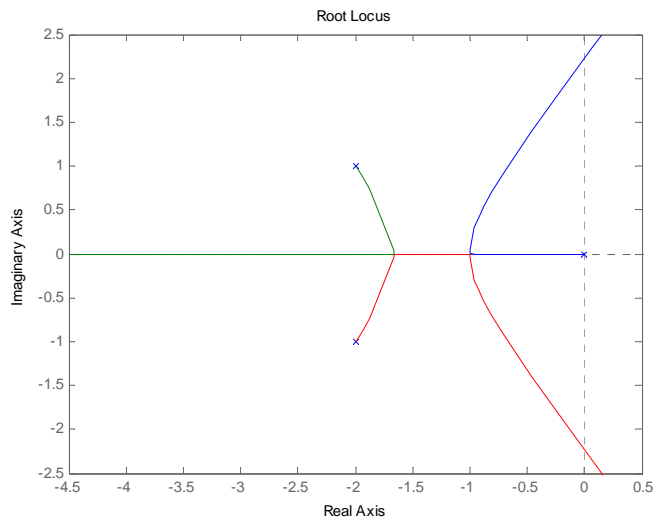


图9.2.1 系统的根轨迹图形

从图9.2.1可知当 $0 < K < 20$ 时系统稳定， $1.85 < K < 2$ 时，系统的根轨迹都位于实轴上，即闭环系统无超调。

9.3 线性系统的频域响应

频域分析法是利用频率特性研究线性系统的一种经典方法，频域分析法是一种图解方法，用开环系统的奈奎斯特图、伯德图、尼克尔斯图分析闭环系统的性能，如系统的稳态性能、动态性能。

9.3.1 奈奎斯特图

nyquist 函数可计算连续时间 LTI 系统的 Nyquist 频率曲线。

【调用格式】

`[re,im,w]=nyquist(sys)` %自动确定频率区间 w，计算频率响应
`[re,im]=nyquist(sys,w)` %求取指定频率区间 w 内的频率响应
`[re,im]=nyquist(A,B,C,D,iu,w)` %系统对第 iu 个输入分量的频率响应

【说明】

1. 当不带输出变量时，nyquist 函数会在当前图形窗口中直接绘制出 Nyquist 曲线。
2. 如果输入变量中没有指定频率范围，其频率范围由函数自动选取并用 w 变量返回，而且在响应快速变化的位置自动选取更多的取样点。
3. 输出变量 re，im 分别为系统 Nyquist 阵列的实部和虚部。如果只给出一个返回变量，则返回的变量为复数阵列，其实部和虚部可以用来绘制系统的 Nyquist 图。
4. re 和 im 是三维矩阵。如果系统有 NU 个输入量，NY 个输出量， $LW=length(w)$ ，re 和 im 的维数为 $(NY \times NU \times LW)$ 。对于 MIMO 系统， $re(i,j,:)$ 和 $im(i,j,:)$ 表示第 i 个输出变量针对第 j 个输入变量的频率响应实部和虚部。对于 SISO 系统，实部和虚部分别由 $re(:)$ 和 $im(:)$ 给出。

例 9.3.1 系统开环传递函数为

$$G(s) = \frac{1000}{(s^2 + 3s + 2)(s + 5)}$$

绘制系统的 Nyquist 图，并讨论其稳定性。

解：用下面的语句直接绘制出系统的 Nyquist 图，如图 9.3.1 所示。

```
G=tf(1000,conv([1,3,2],[1,5]));
nyquist(G);axis('square')
```

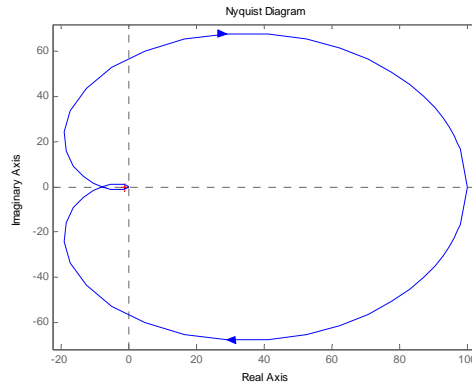



图 9.3.1 系统的 Nyquist 图

该图中 $(-1, j0)$ 点附近 Nyquist 图的情况不是很清楚, (选择图形窗口中的放大按钮), 从局部放大的图形可以看出, Nyquist 图逆时针包围 $(-1, j0)$ 点 2 次, 而原开环系统中没有不稳定极点, 从而可以得出结论, 闭环系统有 2 个不稳定极点。这也可以由下面的 MATLAB 语句来进一步验证。

```
G_close=feedback(G,1);
roots(G_close.den{1})
ans =
    -12.8196
     2.4098 + 8.5427i
     2.4098 - 8.5427i
```

由运行结果可知, 系统有三个根, 其中有两个根位于右半 s 平面, 由此可见该系统是不稳定的。

9.3.2 伯德图

bode 函数可计算连续时间 LTI 系统的幅频和相频曲线(即 Bode 图), 其使用方法与 nyquist 函数相同。

【调用格式】

```
[mag, phase,w]= bode (sys)
[mag, phase]= bode (sys,w)
[mag, phase]= bode (A,B,C,D,iu,w)
```

【说明】bode 函数返回的输出变量 mag, phase 分别为系统 Bode 图数据阵列的幅值 (dB) 和相角 (degrees)。

例 9.3.2 二阶系统传递函数为

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

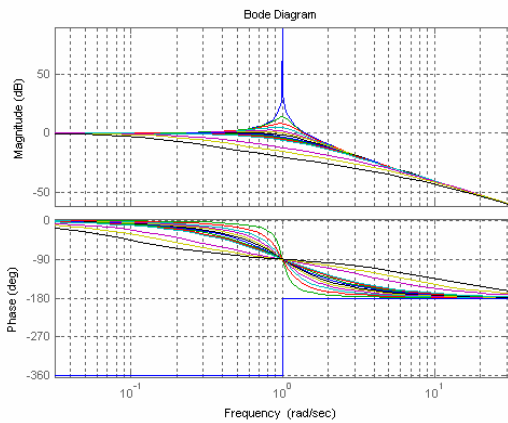
试用 MATLAB 绘制出不同 ζ 和 ω_n 的伯德图。

解: 1、 ω_n 为固定值, ζ 变化时, 运行下面的程序, 得到如图 9.3.2(a) 所示的伯德图

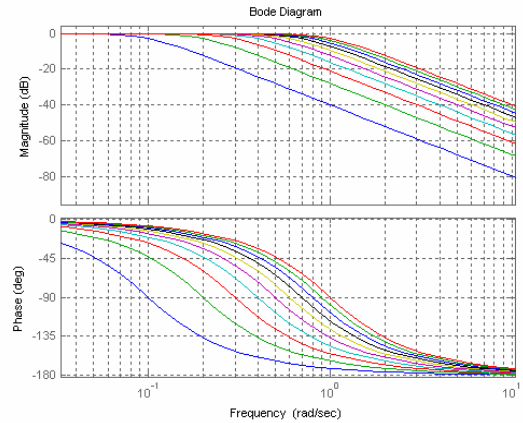
```
wn=1;zet=[0:0.1:1,2,3,5];
hold on
for i=1:length(zet)
    num=wn^2;den=[1,2*zet(i)*wn,wn^2];
    bode(num,den);
end
grid on, hold off
```

2、 ζ 为固定值， ω_n 变化时，运行下面的程序得到如图9.3.2 (b)所示的伯德图

```
wn=[0.1:0.1:1]; zet=0.707;
hold on
for i=1:length(wn)
    num=wn(i)^2; den=[1,2*zet*wn(i),wn(i)^2];
    bode(num,den);
end
grid on, hold off
```



(a) ζ 变化时的伯德图



(b) ω_n 变化时的伯德图

图 9.3.2 不同参数下的二阶系统的伯德图

从图 9.3.2 (a) 可以看出，当阻尼比 ζ 比较小时，则系统的频域响应在自然频率 ω_n 附近将表现出较强的振荡，该现象称为谐振。在图 9.3.2 (b) 中，当自然频率 ω_n 的值增加时，伯德图的带宽将增加，该现象使得系统的时域响应速度变快。

9.3.3 幅值和相角裕量计算

margin 函数可从频率响应数据中计算出幅值裕度、相角裕度和剪切频率。

【调用格式】

```
[Gm,Pm,Wcg,Wcp]=margin (mag,phase,w)
```

```
[Gm,Pm,Wcg,Wcp]=margin (sys)
```

【说明】

1. 函数的输出变量 Gm 为幅值裕度，Wcg 为幅值裕度处的频率值，Pm 为相角裕度，Wcp 为剪切频率。当不带输出参数时，margin 可在当前图形窗口中绘制出 Bode 图，并在 Bode 上标出幅值裕度和相角裕度的值。

2. mag, phase 和 w 为由 bode 函数得到的频率响应的幅值、相角及频率采样值。

例 9.3.3 系统的开环传递函数为

$$G(s) = \frac{3.5}{s^3 + 2s^2 + 3s + 2}$$

求系统的幅值裕度和相角裕度，并求其闭环阶跃响应。

解：运行下面的程序，并得到系统的单位阶跃响应曲线如图 9.3.3 所示

```
G=tf(3.5,[1,2,3,2]); G_close=feedback(G,1);
```

```
[Gm,Pm,Wcg,Wcp]=margin(G)
```

```
step(G_close),grid on
```

运行结果:

Gm =
1.1433
Pm =
7.1688
Wcg =
1.7323
Wcp =
1.6541

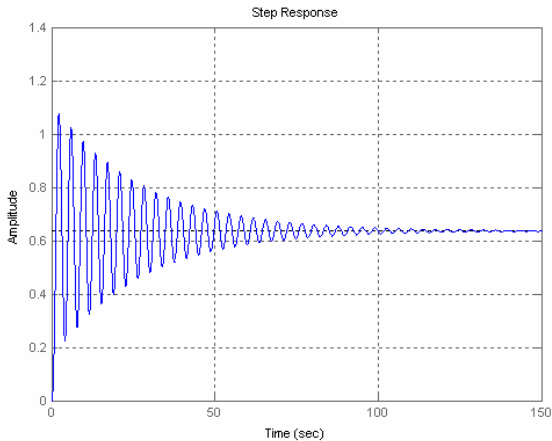


图 9.3.3 系统的单位阶跃响应曲线

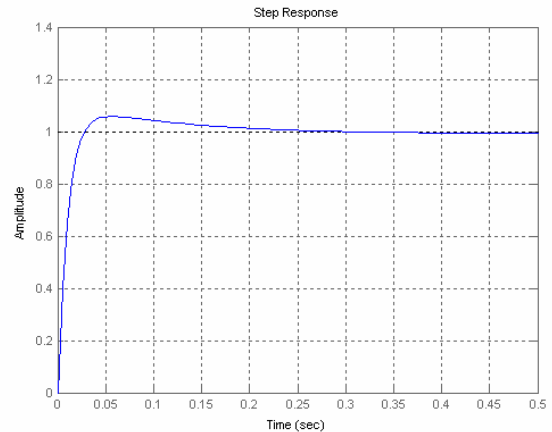


图 9.3.4 系统的单位阶跃响应曲线

从运行结果可知，系统的幅值裕度很接近稳定的边界点 1，且相角裕度只有 7.1578° ，所以尽管闭环系统稳定，但其性能不会太好。同时从图 9.3.3 中可以看出在闭环系统的响应中有较强的振荡。

例 9.3.4 系统的开环传递函数为

$$G(s) = \frac{100(s+5)^2}{(s+1)(s^2+s+9)}$$

求系统的幅值裕度与相角裕度。

解：运行下面的程序，并得到系统的单位阶跃响应曲线如图 9.3.4 所示

```
G=tf(100*conv([1,5],[1,5]),conv([1,1],[1,1,9]));  
[Gm,Pm,Wcg,Wcp]=margin(G)  
G_close=feedback(G,1);  
step(G_close),grid on
```

运行结果:

Gm =
Inf
Pm =
85.4365
Wcg =
NaN
Wcp =
100.3285

从运行结果可以看出，该系统有无穷大的幅值裕度，且相位裕度高达 85.4365° ，所以从图 9.3.4 所示系统的闭环响应也是较理想的。

9.3.4 尼克尔斯图

Nichols 曲线用于分析开环和闭环系统的特性，nichols 函数用于求连续系统的频率响应曲线，其使用方法同 nyquist 函数。

【调用格式】

```
[mag,phase,w]=nichols(sys)
[mag,phase]=nichols(sys,w)
[mag,phase,w]=nichols(A,B,C,D,iu)
[mag,phase]=nichols(A,B,C,D,iu,w)
```

9.4 线性系统的稳定性分析

9.4.1 直接判定方法

求解线性系统稳定性最简单的方法是求出该系统的所有极点，并观察是否含有实部大于零的极点（不稳定极点）。如果有这样的极点，则系统称为不稳定系统，否则称为稳定系统。若稳定系统中存在实部等于 0 的极点，则系统称为临界稳定系统。

要得出传递函数（num，den）描述的系统 and 状态方程（A,B,C,D）描述的系统的所有极点，只需调用 roots（den）函数或 eig（A）函数，这样就可以由得出的极点位置直接判定系统的稳定性了。

例 9.4.1 系统的闭环传递函数为

$$\Phi(s) = \frac{s^3 + 7s^2 + 24s + 24}{s^4 + 10s^3 + 35s^2 + 50s + 24}$$

判断系统的稳定性。

解：运行下面的程序，分析稳定性判别方法

```
G=tf([1 7 24 24],[1 10 35 50 24]);
roots(G.den{1})           %第一种方法，求分母多项式的根，根是一维的
运行结果：
ans =
    -4.0000
    -3.0000
    -2.0000
    -1.0000
G1=zpk(G);                %第二种方法，求极点
G1.p{1};
G2=ss(G);                 %第三种方法，求矩阵 a 的特征值
eig(G2.a);
```

闭环系统的特征根分别为-4，-3，-2，-1，系统的特征根都具有负实部，因此闭环系统是稳定的。

9.4.2 间接判定方法

本节介绍三种判定给定系统稳定性的间接方法，即 Routh 判据、Hurwitz 判据和 Lyapunov 判定法，它们分别适用于传递函数模型和状态空间模型。

1. hurwitz 函数

hurwitz 函数用于构造 hurwitz 矩阵。

【调用格式】

H=hurwitz (den)

【说明】H 为构造的 Hurwitz 矩阵，den 为系统的分母多项式，hurwitz 函数参见附录 D: D-9-2。

2. routh 函数

routh 函数用于构造 Routh 阵列表。

【调用格式】

[rtab,msg]=routh(den)

【说明】其中 den 是系统的分母多项式向量，rtab 是构造的 Routh 表矩阵，msg 变量为字符串型变量，返回有关信息，routh 函数参见附录 D: D-9-3。

3. Lyap 函数

Lyap 函数用于求解李亚普诺夫方程。

【调用格式】

X=lyap (A,B,C)

【说明】lyap(A,B,C)求解方程 $AX+XB=-C$ 的解 X；lyap(A,C)求解方程 $AX + XA' = -C$ 的解 X。

4. posdef 函数

posdef 函数判定矩阵的正定性

【调用格式】

[key,sdet]=posdef(A)

【说明】其中 key 返回矩阵 A 正定性的标记，若 key=1 则表示该矩阵 A 为正定矩阵，否则矩阵 A 为非正定矩阵。sdet 返回各个左上角子矩阵的行列式值。posdef 函数参见附录 D: D-9-4。

例 9.4.2 考虑例 9.4.1 中的系统模型使用 Hurwitz 判据判定系统的稳定性。

解：运行下面的程序，应用 Hurwitz 判据判定系统的稳定性。

```
den=[1,10,35,50,24];
```

```
H=hurwitz(den)
```

```
[key,sdet]=posdef(H)
```

运行结果：

```
H =
```

10	50	0	0
1	35	24	0
0	10	50	0
0	1	35	24

```
key =
```

```
1
```

```
sdet =
```

10	300	12600	302400
----	-----	-------	--------

由运行结果可知 Hurwitz 阵是正定的，因此系统稳定。

例 9.4.3 同例 9.4.1 中给出系统，应用 Routh 阵列表判断系统的稳定性。

解：运行下面的程序，应用 Routh 阵列表判定系统的稳定性

```
den=[1,10,35,50,24];
```

```
[rtab,msg]=routh(den)
```

运行结果：

```
rtab =
```

1	35	24
10	50	0

```

30    24    0
42    0    0
24    0    0

```

```
msg = []
```

从运行结果可知 Routh 表第一列没有符号的变化，所以系统是稳定的。

例 9.4.4 系统的状态方程为

$$\dot{x} = \begin{bmatrix} -4.1053 & 6.6842 & -2.3158 & 20 \\ -0.5789 & -0.7368 & 0.4632 & 4.2 \\ -0.0526 & -1.1579 & 0.6421 & 1.8 \\ -1.2632 & -1.1228 & 1.0772 & 7.2 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0.2 \\ 0.1 \\ 0.4 \end{bmatrix} u$$

$$y = [0 \ 0 \ 8 \ -2]x$$

利用 Lyapunov 判据判定出系统的稳定性。

解：选定一个正定的对角矩阵 $W=\text{diag}(1,2,3,4)$ ，并运行下面的程序

```

A=[-4.1053,6.6842,-2.3158,20;
   -0.5789,-0.7368,0.4632,4.2;
   -0.0526,-1.1579,0.6421,1.8;
   -1.2632,-1.1228,1.0772,7.2];
B=[1;0.2;0.1;0.4]; C=[0,0,8,-2];D=0;
G=ss(A,B,C,D);
W=diag([1,2,3,4]); V=lyap(G.a,-W)
[key,sdet]=posdef(V)

```

运行结果：

```

key=
   -1
sdet=1.0e+006*
   -0.0026   -0.2570   -1.7222   -1.6593

```

由运行结果可以看出，Lyapunov方程的解V不是正定矩阵，系统不稳定。

9.5 离散系统的分析

Matlab 中离散系统的分析方法和连续系统的分析方法十分类似，需要注意的是：

1. Matlab 的控制工具箱提供了分析离散系统的相关函数，这些函数的函数名是在具有同样功能的分析连续系统的函数名前加一个字母“d”，其使用方法和输入输出变量的定义和分析连续系统的函数基本相同。但是调用这些函数的时候，其输入的数学模型变量均为离散系统的数学模型。常用的离散系统分析函数有：

```

dstep    dimpulse  dinitial  dlsim    dlyap    staris
dbode    dnyquist  dnichols  margin
rlocus   rlocfind   zgrid

```

2. 若系统为纯数字系统，则直接利用以“d”开头的函数进行分析计算；若系统中存在连续环节，会涉及到连续系统数学模型离散化问题，要将连续系统数学模型转换为离散系统数学模型，然后再分析计算。

例 9.5.1 已知系统如图 9.5.1 所示，采样周期 $T=0.25s$ ，PID 控制器的参数为 $K_p=1.0468$ ， $K_d=0.2134$ ， $K_i=1.2836$ ，试绘制离散系统的根轨迹图和阶跃响应曲线。

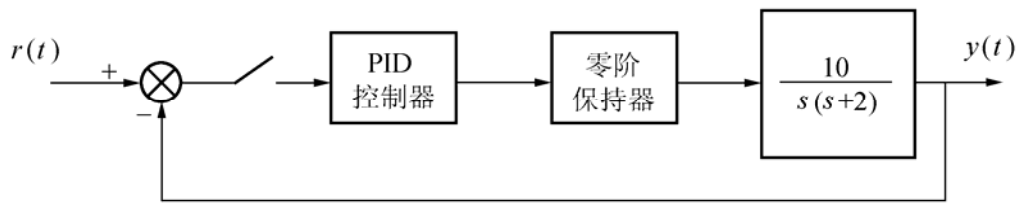


图 9.5.1 系统的方框图

解：首先用零阶保持器将连续系统离散化，然后计算 PID 控制器的脉冲传递函数，最后绘制出带 PID 控制器及零阶保持器的离散系统阶跃响应曲线如图 9.5.2 所示和根轨迹如图 9.5.3 所示。

```

nc=10; dc=[1,2,0]; ts=0.25;
kp=1.0468; kd=0.2134; ki=1.2836;
[nz1,dz1]=c2dm(nc,dc,ts);
nz2=[kp*ts+kd+ki*ts*ts,-(kp*ts+2*kd),kd];dz2=[ts,-ts,0];
[nzk,dzk]=series(nz2,dz2,nz1,dz1);
[nt,dt]=cloop(nzk,dzk,-1);
dstep(nt,dt)
figure; zgrid('new');hold on
rlocus(nzk,dzk);

```

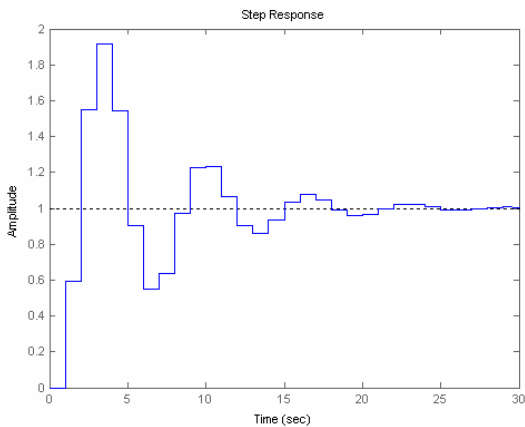


图9.5.2 离散系统阶跃响应曲线

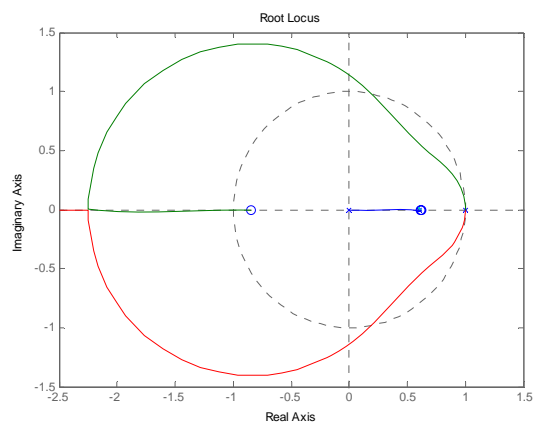


图9.5.3 离散系统的根轨迹

小结

本章是在介绍控制系统基本理论和控制系统工具箱函数的基础上，利用MATLAB语言及其工具箱来解决控制系统的分析问题，包括线性系统的时域分析、频域分析、根轨迹分析和系统的稳定性分析，为系统的仿真和设计做准备。