

目录

1. 自动控制原理仿真实验.....	3
1.1 实验一 控制系统的建模	3
1.2 实验二 控制系统的稳定性分析实验.....	8
1.3 实验三 控制系统的时域分析实验	10
1.4 实验四 控制系统的根轨迹分析.....	12
1.5 实验五 控制系统的频域分析	14

自动控制原理真实验

1.说明

《自动控制原理仿真实验指导书》包括四部分内容：MATLAB 语言基础、MATLAB 语言基础实验、控制理论仿真实验和实验作业。要求每位学生完成规定的实验内容后，独立做完实验作业，以巩固知识，增强应用能力。

本课程上机实验学时为 8 学时，建议分配如下：

实验内容	学时数
实验一 控制系统的建模	1
实验二 控制系统的稳定性分析实验	1
实验三 控制系统的时域分析实验	2
实验四 控制系统的根轨迹分析	2
实验五 控制系统的频域分析	2

上述学时仅为参考学时，学生可根据自己对 MATLAB 的熟悉程度作适当的调整。

1. 自动控制原理仿真实验

1.1 实验一 控制系统的建模

一、实验目的

1. 学习在 MATLAB 命令窗口建立系统模型的方法；
2. 学习如何在三种模型之间相互转换；

二、相关知识

1. 传递函数模型

设连续系统的传递函数为：

$$G(s) = \frac{num(s)}{den(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n}$$

设离散系统的传递函数为：

$$G(z) = \frac{num(z)}{den(z)} = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_{m-1} z + b_m}{a_0 z^n + a_1 z^{n-1} + \cdots + a_{n-1} z + a_n}$$

则在 MATLAB 中，都可直接用分子/分母多项式系数构成的两个向量 **num** 与 **den** 构成的矢量组[**num,den**]表示系统，即

$$num = [b_0, b_1, \cdots, b_m]$$

$$den = [a_0, a_1, \cdots, a_n]$$

建立控制系统的传递函数模型（对象）

的函数为 tf(), 调用格式为：

sys=tf(num,den)

sys=tf(num,den,Ts)

sys=tf(othersys)

sys=tf(num,den) 返回的变量 **sys** 为连续系统的传递函数模型。

sys=tf(num,den,Ts) 返回的变量 **sys** 为离散系统的传递函数模型，Ts 为采样周期，当 Ts=-1 或 Ts=[] 时，系统的采样周期未定义。

sys=tf(othersys) 将任意的控制系统对象转换成传递函数模型。

离散系统的传递函数的表达式还有一种表示为 z^{-1} 的形式（即 DSP 形式），转换为 DSP 形式的函数命令为 filt(), 调用格式为：

sys=filt(num,den)

sys=filt(num,den,Ts)

sys=filt(num,den) 函数用来建立一个采样时间未指定的 DSP 形式传递函数。

sys=filt(num,den,Ts) 函数用来建立一个采样时间为 Ts 的 DSP 形式传递函数。

2. 零极点增益模型

设连续系统的零极点增益模型传递函数为：

$$G(s) = k \frac{(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)}$$

设离散系统的零极点增益模型传递函数为：

$$G(z) = k \frac{(z - z_0)(z - z_1) \cdots (z - z_m)}{(z - p_0)(z - p_1) \cdots (z - p_n)}$$

则在 MATLAB 中，都可直接用向量 $\mathbf{z}, \mathbf{p}, \mathbf{k}$ 构成的矢量组 $[\mathbf{z}, \mathbf{p}, \mathbf{k}]$ 表示系统，即

$$\mathbf{z} = [z_0, z_1, \cdots z_m]$$

$$\mathbf{p} = [p_0, p_1, \cdots p_n]$$

$$\mathbf{k} = [k]$$

在 MATLAB 中，用函数 `zpk()` 来建立控制系统的零极点增益模型，调用格式为：

`sys=zpk(z,p,k)`

`sys=zpk(z,p,k,Ts)`

`sys=zpk(othersys)`

`sys=zpk(z,p,k)` 返回的变量 `sys` 为连续系统的零极点增益模型。

`sys=zpk(z,p,k,Ts)` 返回的变量 `sys` 为离散系统的零极点增益模型，`Ts` 含义同前。

`sys=zpk(othersys)` 将任意的控制系统对象转换成零极点增益模型模型。

3. 状态空间模型

设连续系统的状态空间模型为：

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

设离散系统的状态空间模型为：

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

在 MATLAB 中，连续与离散系统都可直接用矩阵组 $[A, B, C, D]$ 表示系统。

在 MATLAB 中，用函数 `ss()` 来建立系统的状态空间模型，调用格式为：

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

`sys=ss(a,b,c,d,Ts)`

`sys=ss(othersys)` 将任意的控制系统对象转换成状态空间模型。

`sys=ss(a,b,c,d)` 返回的变量 `sys` 为连续系统的状态空间模型。

`sys=ss(a,b,c,d,Ts)` 返回的变量 `sys` 为离散系统的状态空间模型，`Ts` 含义同前。

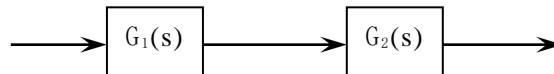
4. 三种系统数学模型之间的转换

解决实际问题时，常常需要对自控系统的数学模型进行转换，MATLAB 提供了用于转换的函数，如下表所示：

函数名	函数功能
ss2tf	将系统状态空间模型转换为传递函数模型
ss2zp`	将系统状态空间模型转换为零极点增益模型
tf2ss	将系统传递函数模型转换为状态空间模型
tf2zp	将系统传递函数转换为零极点增益模型
zp2ss	将系统零极点增益模型转换为状态空间模型
zp2tf	将系统零极点增益模型转换为传递函数模型

5. 环节方框图的化简

1. 环节串联连接的化简



对于上图的两个环节串联，它们的传递函数分别为：

$$G_1(s) = \frac{num1(s)}{den1(s)}, \quad G_2(s) = \frac{num2(s)}{den2(s)}$$

则两个环节串联连阶的等效传递函数为

$$G(s) = G_1(s)G_2(s) = \frac{num1(s)num2(s)}{den1(s)den2(s)}$$

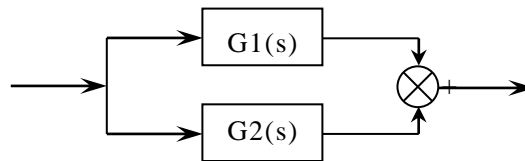
在 MATLAB 中，实现两个环节传递函数串联连接的运算为：

```
sys1=tf(num1,den1)
```

```
sys2=tf(num2,den2)
```

```
sys=sys1*sys2
```

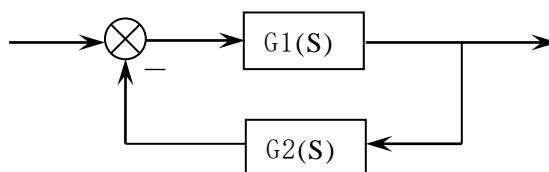
2. 环节并联连接的化简



实现两个环节传递函数串联连接的运算为：

```
sys=sys1+sys2
```

3. 反馈环节的化简



实现反馈环节化简的运算为：

```
sys=feedback(sys1,sys2,sign)
```

式中 sign 为反馈符号，‘+’表示正反馈，‘-’为负反馈。缺省为‘-’。

6. 用 SIMULINK 仿真工具建模

MATLAB 集成有 SIMULINK 工具箱，为用户提供了用方框图进行系统建模的图形窗口。

1. 建立新的结构图文件

在 matlab 环境下，点击 simulink 图标，或输入命令 simulink，启动 simulink 程序，出现 Simulink 模块库，如图 3.1 所示。

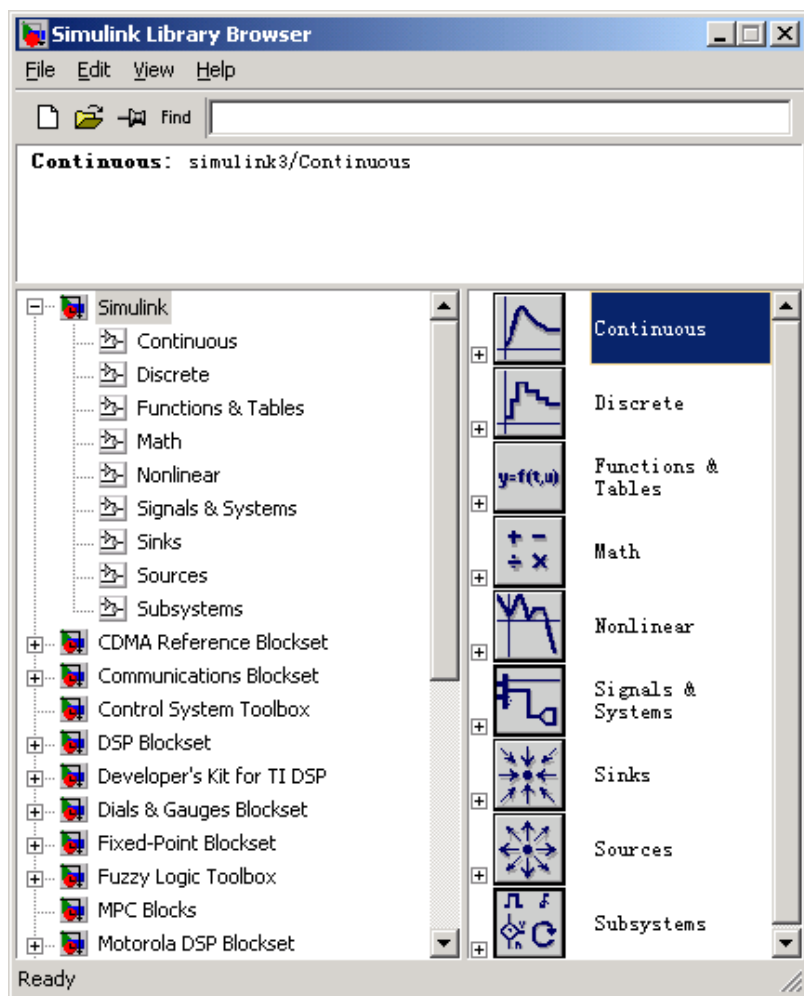



图 3.1SIMULINK 窗口

点击  或选择“File”菜单中的“New”选项下的子选项“Model”，出现“untitled”空白文档。

2. 打开结构图模块组

在 Simulink Library Browser 中，点击 simulink 边上的“+”就有下列模块组：

Sources (输入信号单元模块组)

Sinks (数据输出单元模块组)

Contionuous (线性单元模块组)

Discrete (离散单元模块组)

.....

用鼠标单击任何一个模块组的图标，即可打开该模块组，从中选择仿真实验所需的单元模块。

3. 建立用户的仿真结构图

将所需的结构图模块用鼠标拖到新建的结构图文件的空白处，依次完成仿真结构图。用鼠标作结构图单元之间的连线，按结构图模块的信号连接关系，完成仿真试验的结构图。

1) 连线方法：一般是选中一个输出口，按下鼠标左键拖动至另一个模块的输入口，快捷的方法是先单击选中源模块，按下 Ctrl 键，再单击目标模块。画信号的分支线，用鼠标右键单击信号线并拖动。

2) 模块的旋转和翻转：选中模块，右击鼠标，打开 Format 子菜单，选择 Flip Block、Rotate Block、Flip Name 等完成响应的功能。

4. 结构图单元参数设置

用鼠标器双击任何一个结构图单元即激活结构图单元的参数设置窗口。

5. 仿真参数设置

选择“simulink”菜单项中的“Simulink parameters”，即出现仿真参数设置子窗口，用于设置仿真参数，例如，仿真起始时间，仿真终止时间，仿真布长，允许误差，返回变量名称等。

6. 仿真操作

选中“simulink”中的“Start”启动系统的仿真。在系统仿真中如显示器不能很好地展现波形，可以随时修改显示器的定标，达到满意为止。

二、实验内容

1. 练习上面介绍的各种函数和命令

2. 给定控制系统的传递函数为

$$G(s) = \frac{3}{s^3 + 3s^2 + 5s + 7}$$

在 MATLAB 中建立该系统的传递函数模型、零极点增益模型和状态变量模型。

4.2 实验二 控制系统的稳定性分析实验

一、实验目的

1. 学习控制系统稳定性分析的 MATLAB 实现；
2. 掌握控制系统的稳定判据；

二、相关知识

1. 控制系统的特征多项式与特征方程；

控制系统的传递函数为：

$$\Phi(s) = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} = \frac{M(s)}{D(s)}$$

线性系统的特征方程为：

$$D(s) = a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n = 0$$

2. 控制系统的稳定性判据

稳定判据的方法有：代数稳定判据，根轨迹法，Bode 图法判定系统的稳定性等等。

3. 有关的 MATLAB 函数

roots(P)：是求多项式根函数。P 为特征多项式降幂排列的系数向量。

rlocus(num,den,k)：作系统的根轨迹函数，无开环增益 k，则 k 的范围自动设定，有 k，则 k 的范围可以由人工给定。

[k,poles]=rlocfind(num,den)，执行该命令前，先作根轨迹图，会出现提示，要求在根轨迹上选一点，会出现“+”标记。在 MATLAB 命令窗口会有 k 和 poles 的值。

[Gm,Pm,Wcp,Wcg]=margin(sys0)：在命令窗口得到系统 sys0 的频域性能指标；

margin(sys0)：MATLAB 绘制系统的伯德图，计算伯德图上的稳定裕度，并将计算结果表示在图的上方。

bode(sys0)：MATLAB 绘制系统的伯德图，但无系统的分析结果。

4. m 文件和函数

m 文件是使用 MATLAB 语言编写的程序代码文件。之所以称为 m 文件，是因为这种文件都是以“.m”作为文件的扩展名。我们可以通过任何文本编辑器来生成或编辑 m 文件，但是在 MATLAB 提供的编辑器里生成或编辑 m 文件最为简单、方便且高效。m 文件可以分为两种类型，一种是脚本（或称命令）文件，另一种是函数文件。

脚本文件类似于 C 语言的过程，就是将一系列命令写在一个文件中，它的书写方法与命令行执行程序的方式相同。

函数也是以“.m”作为文件的扩展名。不同于 m 文件，函数带有输入输出参数，书写格式为

```
function [返回变量列表]=函数名（输入变量列表）  
%注释行  
主程序体
```

其中 function 为函数的保留字，函数名也要符合变量的命名规则，保存函数文件时，文件名应与函数名同名。注释行是对函数的功能、调用格式和参数的说明。

三、实验内容及要求

1. 已知系统的开环传递函数：

$$G(s) = 100 \frac{(s+2)}{s(s+1)(s+20)}$$

用求根的方法来判别闭环系统的稳定性。

编写程序，求特征多项式及其根（不能手工计算），判断系统的稳定性。

2. 已知一个单位负反馈开环传递函数 G(S)，当 k 分别为 1、5、10、20 时闭环系统的稳定性。

$$G(s) = k \frac{(s+3)}{s(s+5)(s+6)(s^2+2s+2)}$$

以 k 为输入参数，编写函数，画出上述 k 值对应的闭环根，并判断系统的稳定性。

3. 已知单位负反馈系统的传递函数分别为

$$1) \quad G_1(s) = \frac{2.7}{s^3 + 5s^2 + 4s} \qquad 2) \quad G_2(s) = \frac{2.7}{s^3 + 5s^2 - 4s}$$

用 Bode 图法判断系统闭环的稳定性。

编写程序，并运行程序，得出相关的数据。

4.3 实验三 控制系统的时域分析实验

一、实验目的

1. 学习控制系统时域分析的 MATLAB 实现。
2. 掌握控制系统的时域响应及性能指标。

二、相关知识

有关的 MATLAB 函数：

damp(den)

step(sys)

[y,t,x]=step(sys)

impulse(sys)

[y,t,x]=impulse(sys)

initial(sys,x0)

[y,t,x]=initial(sys,x0)

lsim(sys, u, t)

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

对于离散系统，相应的命令分别是 dstep、dimpulse、dinitial、dlsim。

三、实验内容及要求

- 1、利用 help 命令学习上述函数命令的用法，自行练习。
- 2、已知一阶系统传递函数为

$$\phi(s) = \frac{1}{s+1}$$

输入为正弦信号，求输出。编写程序，将输入和响应曲线画于同一图上。

- 3、已知二阶系统传递函数为

$$\phi(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

当 $\omega_n=1$ 时，试计算当阻尼比 ξ 值从 0.1 到 1 时二阶系统的阶跃响应。编写程序，将响应曲线画于同一图上，并加上标注。

- 4、二阶系统为

$$\phi(s) = \frac{10}{s^2 + 2s + 10}$$

编写程序，求系统的根、阻尼比、无阻尼振荡频率和响应曲线，计算（注意不是从响应图上读出）出峰值、峰值时间和过渡时间，并与理论公式计算值比较。

		实际值	理论值
峰值			
峰值时间			
过渡时间	$\pm 5\%$		
	$\pm 2\%$		

5、编写函数，输入参数为对象模型 `sys`，求该对象模型的单位斜坡输入响应，将输入和输出曲线画于同一图上。（提示：`sys` 的分子多项式为 `sys.num{1}`，分母多项式为 `sys.den{1}`。）

第四题参考程序

```
num=10; den=[1 2 10];
G=tf(num,den) %
G=zpk([],[-1+3*i,-1-3*i],3);
% 计算最大峰值时间和它对应的超调量
C=dcgain(G)
[y,t]=step(G);
plot(t,y)
grid [Y,k]=max(y); %取得最大峰值
timeopeak=t(k) %取得最大峰值时间
percentovershoot=100*(Y-C)/C %计算超调量
n=1 while y(n)<C
n=n+1
end
risetime=t(n) %计算稳态响应时间
i=length(t);
while(y(i)>0.98*C)&(y(i)<1.02*C)
i=i-1;
end
settingtime=t(i)
```

4.4 实验四 控制系统的根轨迹分析

一、实验目的

- 1、学习控制系统根轨迹分析的 MATLAB 实现。
- 2、利用根轨迹进行系统分析。

二、相关知识

有关的 MATLAB 函数：

pzmap(sys);

rlocus(sys);

[r, k]=rlocus(sys);

rlocfind(sys);

[k, r]=rlocfind(sys)

注意：没有离散系统对应的命令“drlocus”

三、实验内容及要求

- 1、利用在线帮助学习上述函数命令的用法，自行练习。
- 2、已知系统的开环传递函数为

$$G(s)H(s) = \frac{k(s+1)}{s(s-1)(s^2+4s+20)}$$

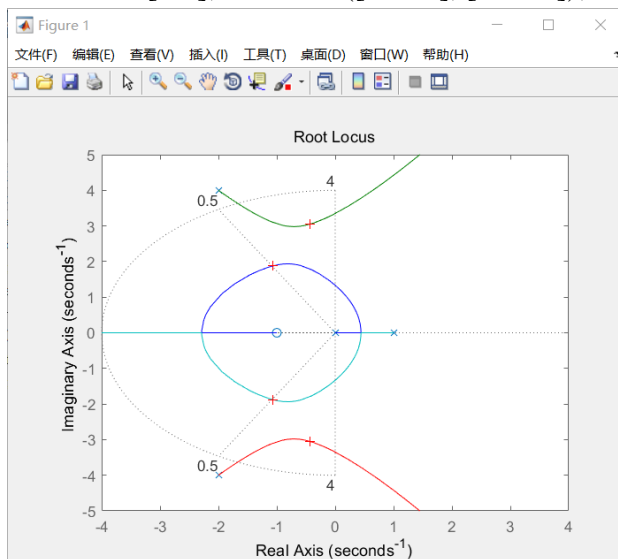
试确定使系统稳定的 k 值范围。

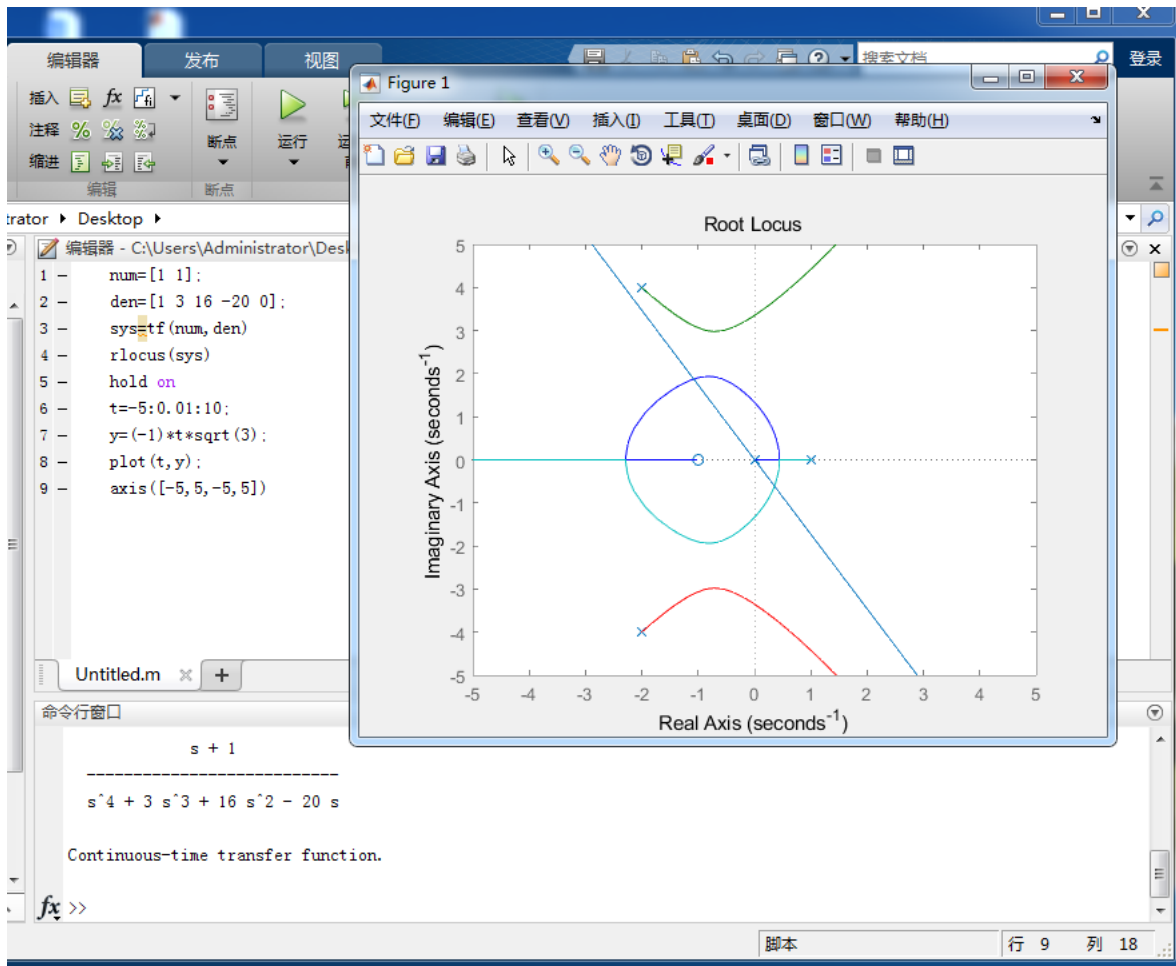
（提示：为了使坐标点定位准确，可用 axis 命令将图局部放大）

- 3、上题中，确定使系统阻尼比为 0.5 的 k 值和这时的闭环特征根。

（注意：不能用目测定位。）

num=[1 1];den=conv([1 -1 0],[1 4 20]); rlocus(num,den) axis ([-5 5 -5 5])





4.5 实验五 控制系统的频域分析

一、实验目的

- 1、学习控制系统频域分析的 MATLAB 实现。
- 2、利用频率响应进行系统分析。

二、相关知识

有关的 MATLAB 函数：

`Bode(G);`

`[m,p,w]=bode(G);`

`Margin(G);`

`[Gm,Pm,Wg,Wp]=margin(G);`

`Nyquist(G);`

`[re,im,w]=nyquist(G);`

对于离散系统，相应的命令分别是 `dbode`、`dnyquist`，没有“`dmargin`”命令，要利用 `margin` 命令，先用 `[m,p,w]=dbode(G)`，再用 `margin(m,p,w)` 命令。

三、实验内容及要求

- 1、利用在线帮助学习上述函数命令的用法，自行练习。
- 2、一单位负反馈系统

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

画出其 bode 图，求其幅值裕量和相位裕量，判断其稳定性和动态性能，用 nyquist 图分析，是否与前述分析相符。若开环增益放大 1.5 倍，又怎样？利用时域分析法验证之。