

Matlab 仿真实验 2. 根轨迹与频率特性分析

基于根轨迹的性能分析上机指导书

一、 实验目的

1. 学会利用根轨迹工具对系统性能进行分析。
2. 观察和理解增加零极点对闭环根轨迹的影响。
3. 初步掌握利用 Matlab 产生根轨迹的基本指令和方法。

二、 实验内容

1. 初步熟悉根轨迹命令。对如下开环传递函数画关于根轨迹参数 k 的闭环根轨迹图。

$$G(s) = \frac{k(s^2 + 2s + 4)}{s(s + 4)(s + 6)(s^2 + 1.4s + 1)}$$

2. 确定使闭环系统稳定的根轨迹参数 k 的范围。先考察上式中的开环传递函数 $G(s)$ ，确定参数 k 的范围使闭环系统稳定。除此以外，再考察以下开环传递函数

$$G_1(s) = \frac{k}{s(s + 4)(s^2 + 4s + 20)}$$

$$G_2(s) = \frac{k(s^2 + 5s + 5)}{s(s + 1)(s^2 + 2s + 2)}$$

3. 确定某闭环极点对应的根轨迹增益，以及该增益所对应的其它闭环极点。确定根轨迹的分离点、与虚轴的交点等及其对应的根轨迹增益。
4. 在根轨迹图上利用等阻尼比射线选择闭环极点，理解主导极点对系统行为的影响。
5. 增加开环零点对闭环性能的影响。考虑如下开环传递函数

$$G_3(s) = \frac{k(s - z_1)}{s(s^2 + 2s + 2)}$$

其中， z_1 为附加的开环实数零点，其值可在 s 左半平面内任意选择。根据根轨迹分析增加开环零点对系统性能的影响。

三、 实验过程

1. 编写最基本的绘制根轨迹的 m 文件

编写如下内容的m文件，对该文件命令，如命令为rootplot

```
num=[1, 2, 4];  
den=conv(conv([1, 4, 0], [1, 6]), [1, 1.4, 1]);  
sys=tf(num, den);  
rlocus(sys);
```

有两种方式可以运行该 m 文件。第一种是在 Matlab 的命令窗口建 rootplot 后回车；另一种是在如图 1 所示的界面点击绿色的 Run 按钮 。运行后出现图 2 所示的界面。对该 m 文件中的语句做以下说明。

- (1) `num=[1,2,4]` 该语句生成开环传递函数的 $G(s)$ 分子多项式。1, 2 和 4 分别 2 次项、一次项和 0 次项的系数。
- (2) `den=conv(conv([1,4,0],[1,6]),[1,1.4,1])` 该句的目的是生成开环传递函数的 $G(s)$ 分母多项式的系数。在该语句中有一个命令 `conv`，其作用是求取两个向量的卷积。如果两个向量分别由两个多项式的系数构成，则这两个向量的卷积由这两个多项式乘积构成。在该语句中，`[1,4,0]` 和 `[1,6]` 分别为 $s^2 + 4s$ 和 $s + 6$ 的系数构成得向量，则 `conv([1,4,0],[1,6])` 生成由 $(s^2 + 4s)(s + 6)$ 的系数构成的向量。再次使用命令 `conv` 则可得到开环传递函数的 $G(s)$ 的分母多项式。
由于根轨迹分析是以开环系统的零极点为出发点，因此在基于根轨迹的分析中，命令 `conv` 会经常用到。
- (3) `sys=tf(num,den)` 生成以 `num` 为分子多项式的系数，以 `den` 为分子多项式的系数的传递函数。
- (4) `rlocus(sys)` 画出以 k 为变量的闭环系统根轨迹。

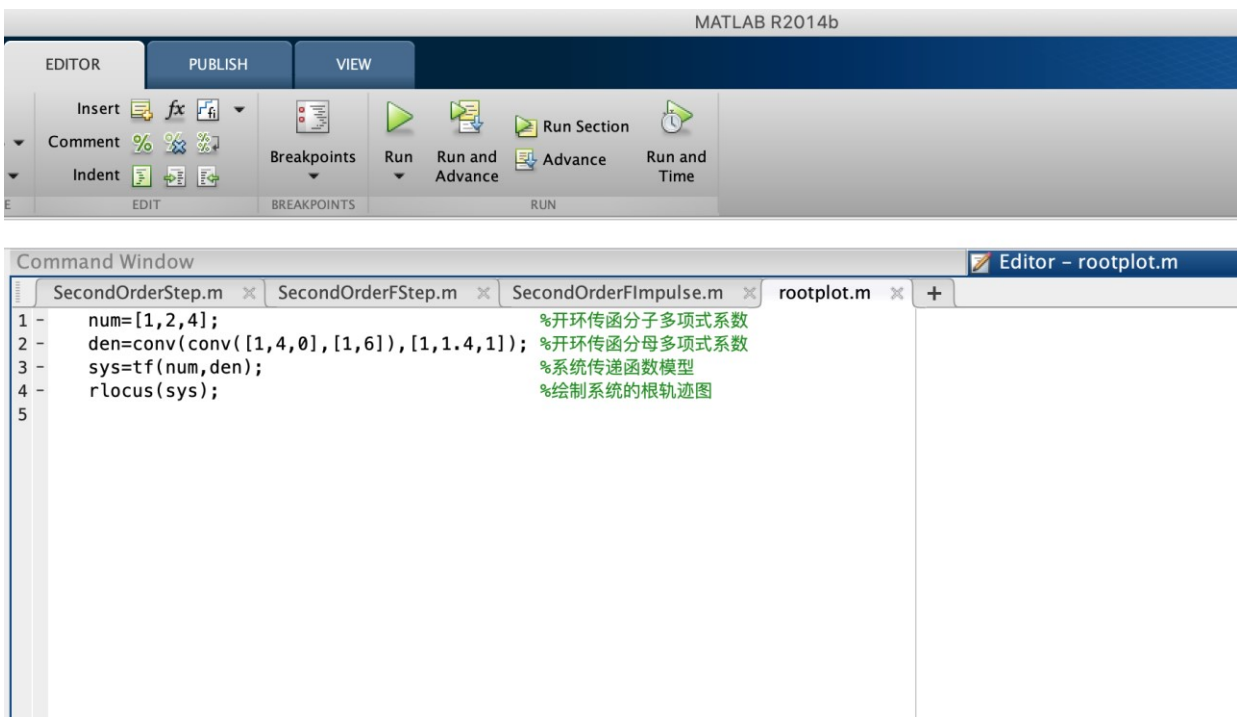


图 1 m 文件编辑界面

2. 编写限制根轨迹实轴和虚轴范围的 m 文件

可以看到图 2 中的根轨迹是宏观图, 如果需要画出局部的根轨迹图可以通过限制坐标轴的范围来实现。编写如下内容的 m 文件

```
num=[1,2,4];  
den=conv(conv([1,4,0],[1,6]),[1,1.4,1]);  
sys=tf(num,den);  
rlocus(sys);  
axis([-8 2 -6 6]);
```

将该 m 文件命名为 rootplotax, 运行后得到如图 3 所示的界面。也就是说新添加的命令 `axis([-8 2 -6 6])` 的将根轨迹图的实轴限制在-8 至 2 间, 将虚轴限制在-6 至 6 之间。

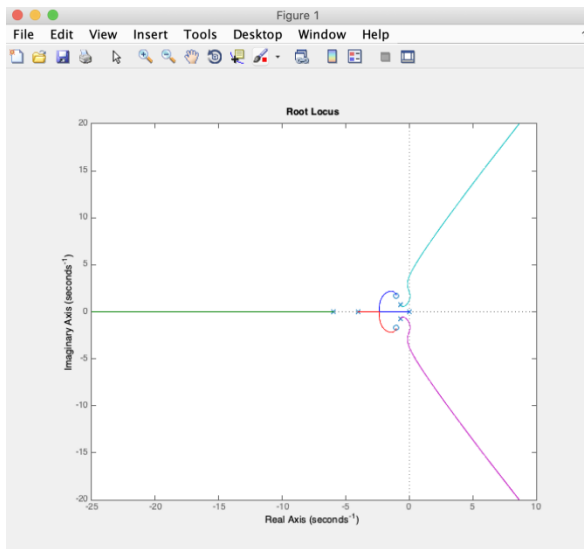


图 2 根轨迹图

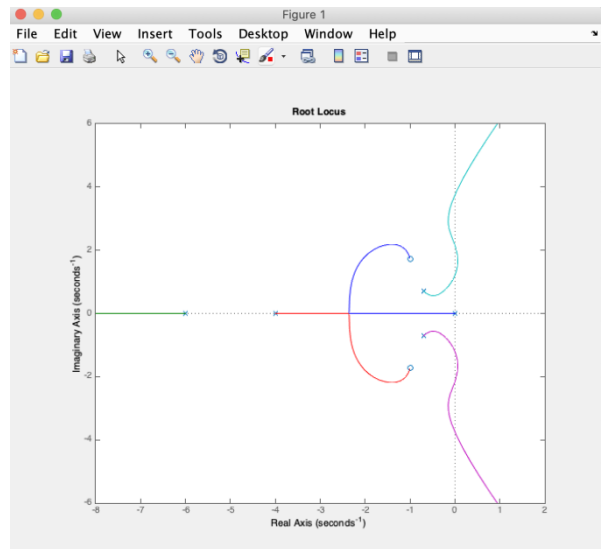


图 3 限制了范围的根轨迹图

3. 改变根轨迹图的颜色和线型

也可以利用命令 `rlocus` 改变根轨迹的颜色和线型。对于上面提到的 m 文件 `rootplotax`, 如果将语句 `rlocus(sys)` 换为 `rlocus(sys,'g-.')`, 运行后出现图 4 所示的界面。可以看到, 此图的根轨迹是绿色的点划线。

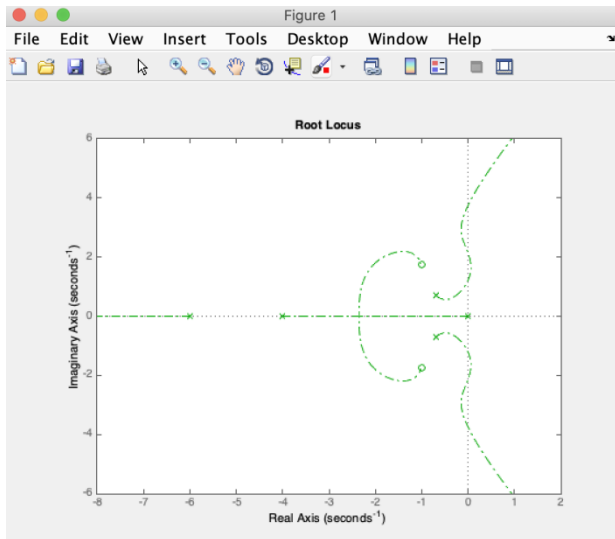


图 4 更改颜色后的根轨迹

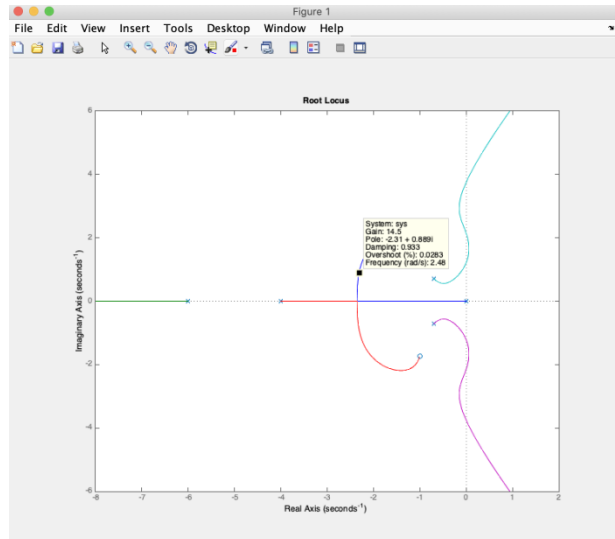


图 5 Sources 模块

4. 确定使闭环系统稳定的参数范围

在图 3 所示的界面上，可以确定使闭环系统稳定的 k 的范围。在图 3 所示的根轨迹图上，用鼠标点击根轨迹上的任意一点，都会弹出的一个黄底色的框，框内显示了该点的位置，所对应的开环增益 k 的值，还有系统超调，阻尼等信息。

把鼠标挪到框里点击右键，弹出如图 6 所示的对话框，选择适当的选项可以对信息框进行适当的操作，如更改信息框里字的大小、文字对齐方式等。选择“Delete”可以将信息框删除。

依次选中根轨迹与虚轴的三个交点，分别点击后弹出三个信息框，由图 7 可以看到它们对应的开环增益 k 分别为 15.6，67.4 和 159。根据根轨迹理论可知使闭环系统稳定的开环增益的范围是

$$0 < k < 15.6, 67.9 < k < 159$$

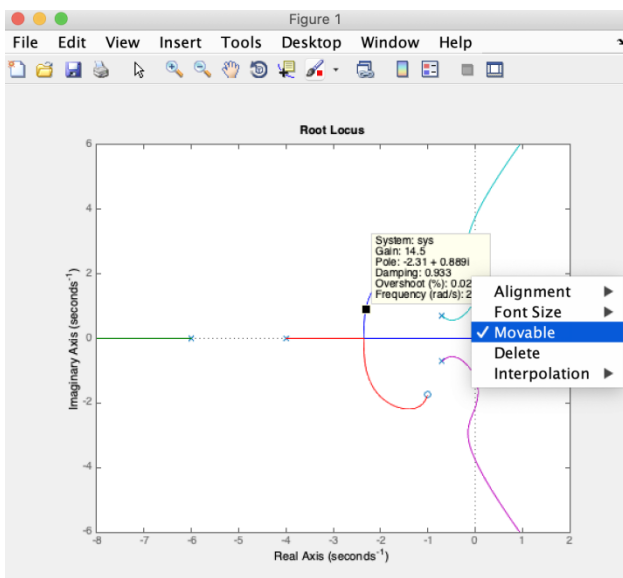


图 6 根轨迹信息框

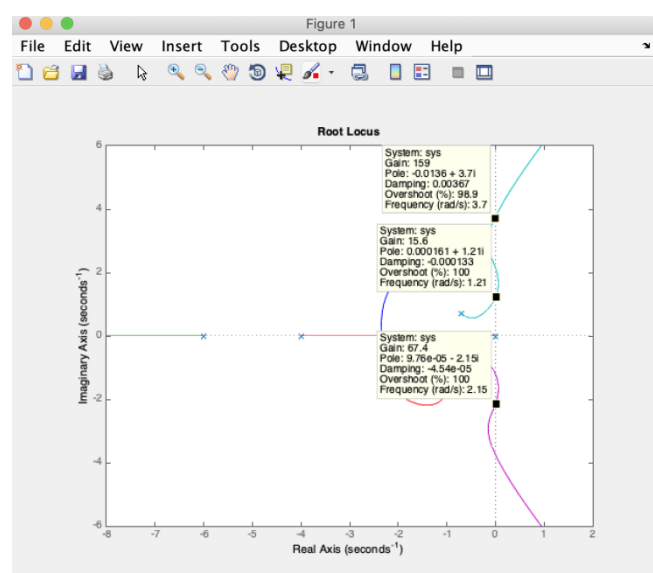


图 7 虚轴上的跟对应的参数值

5. 确定某闭环极点对应的开环增益，以及该增益对应的其它闭环极点

有两种方法完成该任务。

方法一：在画好的根轨迹上点击感兴趣的闭环极点，弹出的信息框里会出现对应的开环增益 k_0 。有了这个开环增益，`p=rlocus(sys,k0)` 即可返回所对应的闭环极点。对于 `m` 文件 `rootplotax` 中的系统，由图 7 可以看到虚轴上的闭环极点 $1.21j$ 对应的增益为 15.6，在命令框口键入如下命令

```
>> p=rlocus(sys,15.6)
```

返回一下结果

```
p =
```

```
-6.7971 + 0.0000i  
-2.3014 + 0.9725i  
-2.3014 - 0.9725i  
-0.0001 + 1.2127i  
-0.0001 - 1.2127i
```

方法二：使用命令 `rlocfind`

编写如下内容的 `m` 文件

```
num=[1,2,4];  
den=conv(conv([1,4,0],[1,6]),[1,1.4,1]);  
sys=tf(num,den);  
rlocus(sys);  
axis([-8 2 -6 6]);  
[k,p]=rlocfind(sys)
```

将该 `m` 文件命名为 `rootfind`，运行后得到如图 8 所示的界面。移动鼠标可以改变两条横纵线的位置。将两条线的相交点移动到所感兴趣的点，比如说分离点，然后单击鼠标左键，图形界面就变为图 9 所示的界面。红色“+”号的四个闭环极点具有相同的开环增益。与此同时，`Matlab` 命令窗口也返回图 10 所示的结果。

采用类似的步骤可以确定根轨迹的分离点及其对应的开环增益。

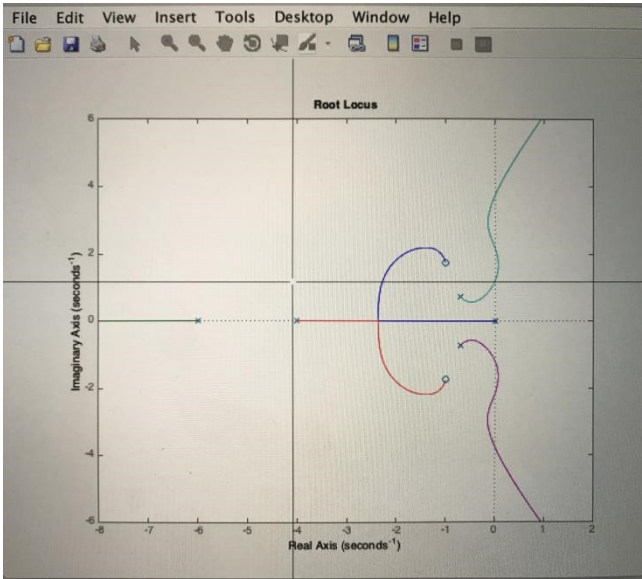


图 8 运行 rootfind 后的界面

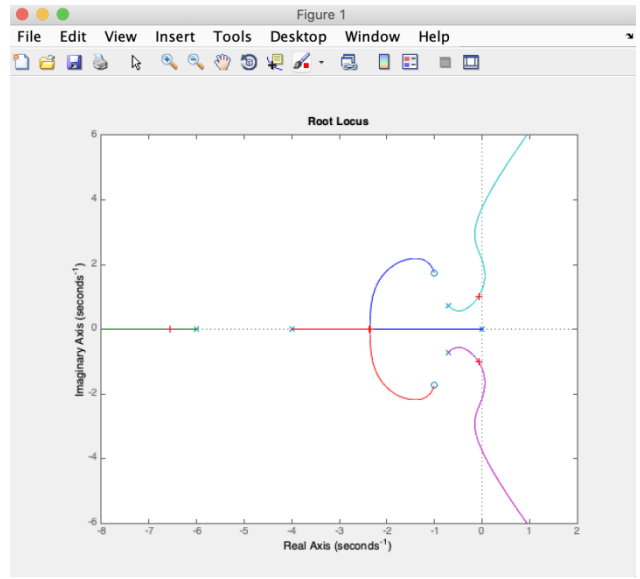


图 9 选中某一闭环极点后的根轨迹界面

6. 借助等阻尼比射线进行辅助分析

利用命令利用命令 `rlocus` 画出根轨迹后，再跟上命令 `grid` 可以在根轨迹图上画出等阻尼比射线。为完成此目的，可以对刚刚编写好的 `m` 文件 `rootfind` 进行简单修改即可，修改后的代码如下

```
num=[1,2,4];
den=conv(conv([1,4,0],[1,6]),[1,1.4,1]);
sys=tf(num,den);
rlocus(sys);
grid
axis([-8 2 -6 6]);
```

运行该 `m` 文件后出现如下图形界面。

```

>> rootfind
Select a point in the graphics window

selected_point =

    -2.3483 - 0.0000i

k =

    9.4865

p =

    -6.5521 + 0.0000i
    -2.3631 + 0.0000i
    -2.3483 + 0.0000i
    -0.0683 + 1.0193i
    -0.0683 - 1.0193i
>>

```

图 10 使用 rlocfind 后返回的结果

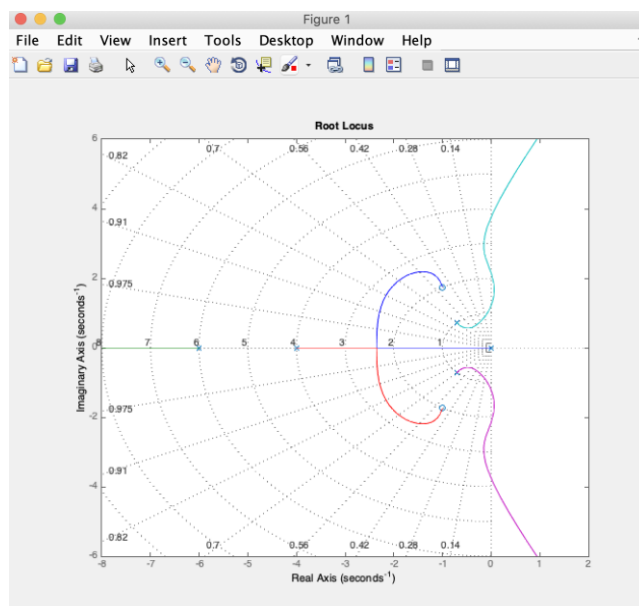


图 11 标有等阻尼比射线的根轨迹图

7. 增加零点对系统性能的影响

对开环传递函数为 $G_3(s)$ 的系统编写以开环零点 z_1 为输入的 m 文件，其文件名为 rootzero，具体代码如下

```

function rootzero(z)
num=[1,-z];
den=[1,2,2,0];
sys=tf(num,den);
rlocus(sys);
axis([-8 2 -6 6]);

```

首先画出不含开环零点的根轨迹，然后根据编写好的 m 文件 rootzero 画出开环零点 z_1 从 -100 变化到 0 的根轨迹。对比这些根轨迹图，能看出什么规律？增加开环零点对闭环根轨迹有何影响？

四、 实验前的准备工作

1. 复习绘制根轨迹的规则。
2. 利用根轨迹规则手工绘制实验内容中 $G(s)$, $G_1(s)$, $G_2(s)$ 和 $G_3(s)$ 的根轨迹。

五、 本实验指导书的使用方法

本实验指导书按照实验内容给出了完成相应实验任务的 m 文件代码，介绍了在根轨迹界面的一些操作方法。分散在不同实验中逐步介绍了一些指令的使用方法，这些指令如下表所示。

(1) rlocus	(2) rlocfind	(3) conv
(4) grid	(5) axis	

六、 实验报告要求

针对实验内容编写相应的 m 文件，将代码和运行得到的根轨迹图形界面及在 Command Window 返回的结果截图到实验报告。针对仿真结果对实验结果进行分析，得出结论。

线性系统的频率特性分析

一、 实验目的

1. 增强对典型环节频率特性的理解。
2. 加深二阶振荡环节的阻尼比对 Nyquist 曲线和 Bode 图的影响。
3. 初步掌握利用 Matlab 绘制 Bode 图和 Nyquist 图的基本指令和方法。

二、 实验内容

1. 初步熟悉绘制 Bode 图的指令。

对以下传递函数绘制不同开环增益时的 Bode 图

$$G(s) = \frac{K(2s + 1)}{s(2.5s + 1)(0.04s^2 + 0.24s + 1)}$$

2. 初步熟悉绘制 Nyquist 图的指令。

对以下传递函数绘制不同开环增益时的 Nyquist 图

$$G(s) = \frac{K(2s + 1)}{s(2.5s + 1)(4s^2 + s + 1)}$$

3. 一阶最小相位和非最小相位惯性环节的频率特性。

(1) 一阶惯性环节 $G(s) = \frac{K}{Ts+1}$ 和非最小相位的惯性环节 $G(s) = \frac{K}{Ts-1}$ 的 Nyquist 图之间的对比，
Bode 图之间的对比。

(2) 一阶惯性环节 $G(s) = \frac{K}{Ts+1}$ 中 K 和 T 的参数对 Nyquist 图和 Bode 图影响。

4. 二阶振荡环节的频率特性。

(1) 二阶振荡环节

$$G(s) = \frac{1}{Ts^2 + 2T\xi s + 1}$$

当 T 固定，阻尼比的变化对 Nyquist 图的影响。

(2) 二阶振荡环节当 T 固定，阻尼比的变化对 Bode 图的影响。

5. 几类典型传递函数的频率特性。

(1) $G(s) = \frac{1}{s(Ts^2 + 2T\xi s + 1)}$

(2) $G(s) = \frac{K}{(T_1s+1)(T_2s+1)(T_3s+1)}$

(3) $G(s) = \frac{K}{s(Ts+1)}$

$$(4) G(s) = \frac{K(1+T_1s)}{s(T_2s+1)(T_3s+1)}$$

$$(5) G(s) = \frac{K(\tau s+1)}{s(Ts-1)}$$

$$(6) G(s) = \frac{aTs+1}{Ts+1}$$

$$(7) G(s) = \frac{K}{(s+2)(s+5)(s-1)}$$

$$(8) G(s) = \frac{K}{s^2(T_1s+1)(T_2s+1)}$$

$$(9) G(s) = \frac{K}{s^2(T_1s+1)(T_2s-1)}$$

$$(10) G(s) = \frac{K(1+T_1s)}{s^2(T_2s+1)(T_3s+1)}$$

$$(11) G(s) = \frac{K(1+T_1s)}{s^2(T_2s+1)(T_3s-1)}$$

三、 实验过程

1. 编写以开环增益为输入绘制 Bode 图的 m 文件

编写如下内容的m文件，对该文件命名，如命名为testbode

```
function testbode(K)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[0.04,0.24,1]);
sys=tf(num,den);
bode(sys);
grid;
```

在以上代码中，指令 bode 的作用是画出系统的 Bode 图，而命令 grid 的作用是在画好的 Bode 图上加网格线。如果不希望出现网格线，可以把指令 grid 删掉。

在 Matlab 命令窗口输入

```
>> testbode(1)
```

回车后出现图 1 所示的图形界面。在该图形界面上用鼠标左键点击 Bode 曲线上的点，会弹出一个信息框，如图 2 所示。在信息框内显示了该点所对应的频率和幅值。如果是在相频曲线上点击对应的点，则弹出的信息框会显示该点所对应的频率和相位。

选中出现的那个点，摁住鼠标左键可以拖动该点在 Bode 图曲线上移动。对于 Bode 图来

说，一般关注剪切频率和与之对应的相位（这个相位加上 180 度就是相位裕度）。对于该系统，我们在幅频曲线和相频曲线上拖动信息框到这个位置后，得到图 3 所示的界面。由该图可以看出，该系统的剪切频率为 0.846，相位裕度是 $180-107=73$ 度。该界面可以另存为 jpg 格式的图像，如图 4 所示。

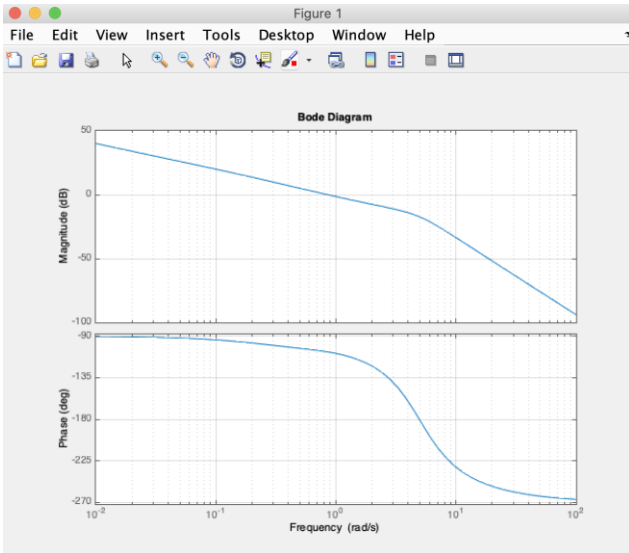


图 1 带网格线的 Bode 图

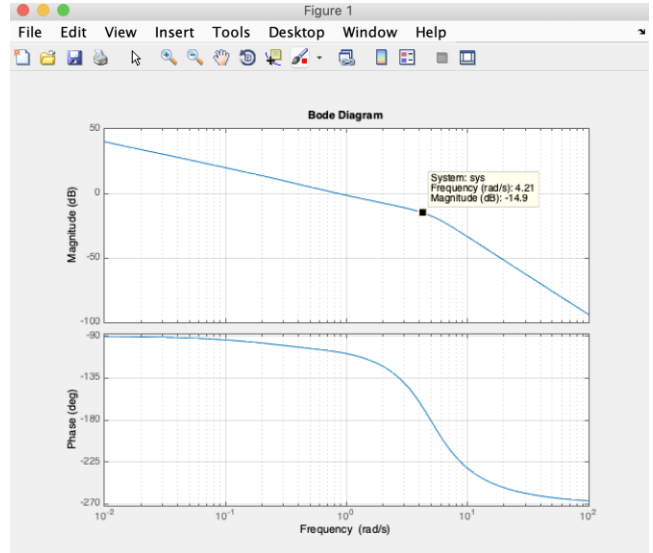


图 2 Bode 图上的信息框

前面所述及的方法是把增加网格线放在 m 文件里。也可以在 m 文件不包含这条语句，而在命令窗口运行后，再在命令窗口简单跟上指令 `grid` 即可画出网格线。接下来，我们示范在同一个图里画出不同开环增益 K 的 Bode 图。编写以下.m 文件，保存为 `testbode1`

```
function testbode1(K)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[0.04,0.24,1]);
sys=tf(num,den);
bode(sys);
hold on
```

在命令窗口依次渐入以下指令，

```
>> testbode1(1)
>> testbode1(10)
>> testbode1(100)
>> grid
```

运行后得到如图 5 所示的界面。

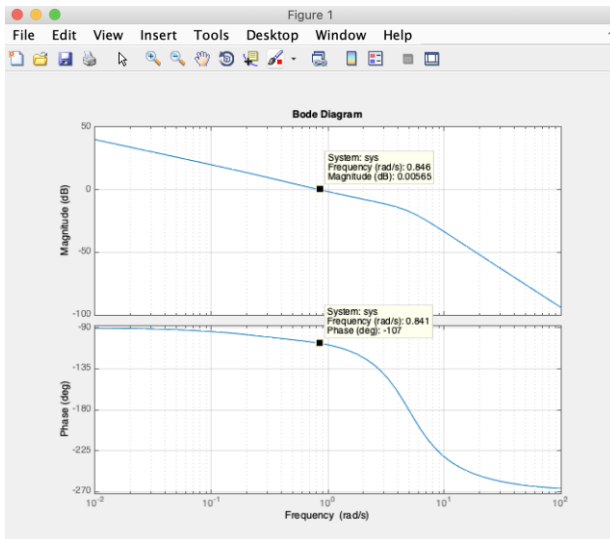


图3 幅频曲线和相频曲线同时有信息框

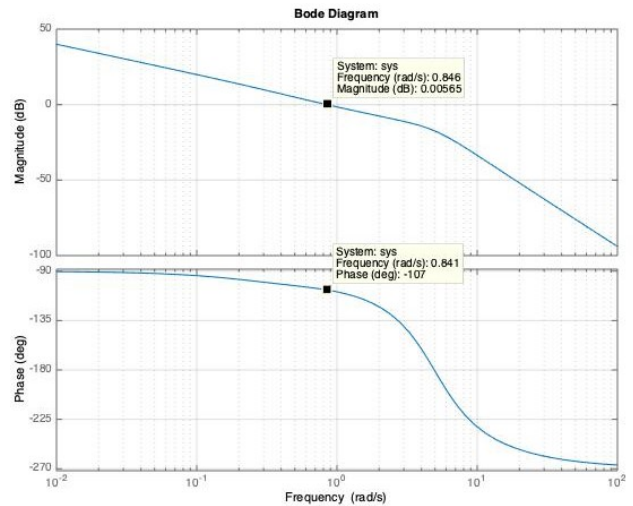


图4 幅频曲线和相频曲线同时有信息框的jpg文件

2. 编写以开环增益为输入绘制给定频率区间的 Bode 图的 m 文件

前面在使用命令bode画Bode图时，所画图形的频率区间是系统制定的。也可以自己指定频率区间，这需要给指令bode增加更多的输入。编写如下m文件，保存为testbodewm

```
function testbodewm(K)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[0.04,0.24,1]);
sys=tf(num,den);
bode(sys,{0.001,1000});
```

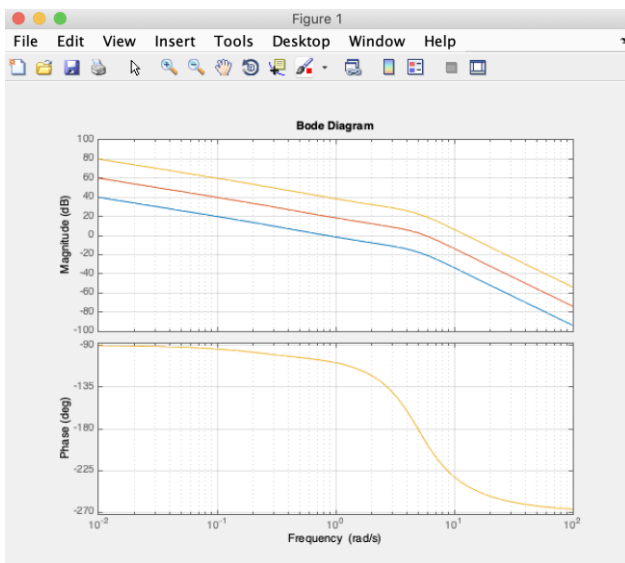


图5 在同一幅图花出多条 Bode 图

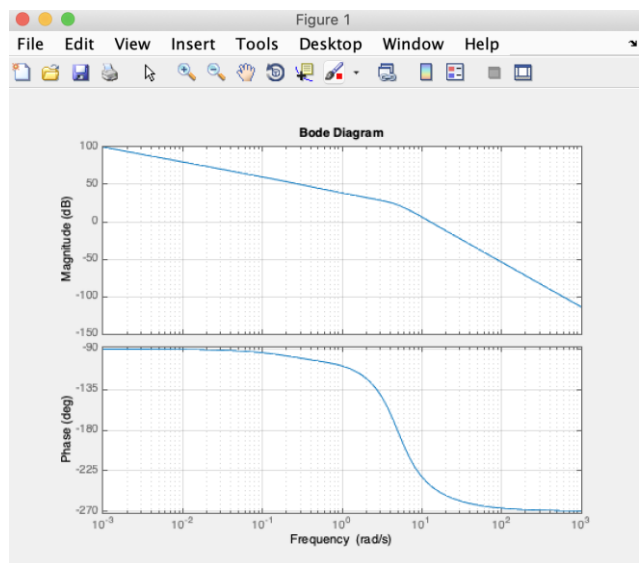


图6 具有指定区间的 Bode 图

在上面的 m 代码中, 语句 `bode(sys, {0.001, 1000})` 中的 `{0.001, 1000}` 即为指定的频率区间。注意语法规则: 大括号 `{0.001, 1000}` 中表示频段的两个端点一定都是大于 0 的正数。

在命令窗口键入以下命令

```
>> testbodewm(100)
>> grid
```

运行后得到如图 6 所示的界面。

3. 编写以开环增益和指定频率点为输入的 m 文件

指令 `bode` 除了可以画出 Bode 图外, 还能返回给定频率点的幅值和相位。编写如下函数, 把 .m 文件保存为 `testbodefre`

```
function testbodefre(K,w)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[0.04,0.24,1]);
sys=tf(num,den);
[mag,ph]=bode(sys,w)
```

在命令窗口键入以下命令

```
>> testbodefre(10,400)
```

运行后得到如图 7 所示的结果。这说明, 开环传递函数

$$G(s) = \frac{10(2s + 1)}{s(2.5s + 1)(0.04s^2 + 0.24s + 1)}$$

的频率特性在频率 $\omega = 400$ 时的幅值为 3.1251×10^{-3} , 相位为 -269.1548 度。

```
>> testbodefrec(10,400)
```

```
mag =
```

```
3.1251e-06
```

```
ph =
```

```
-269.1548
```

图 7 在给定频率点的幅值和相位

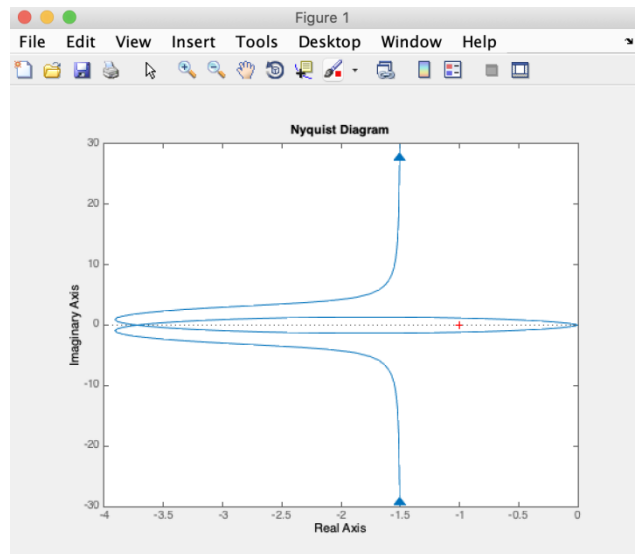


图 8 全频段 Nyquist 图

4. 编写以开环增益为输入绘制 Nyquist 图的 m 文件

编写如下内容的m文件，对该文件命名，如命名为testnyquist

```
function testnyquist(K)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[4,1,1]);
sys=tf(num,den);
nyquist(sys);
```

在以上代码中，指令 `nyquist` 的作用是画出系统的全频段 Nyquist 图，既有正频率段，也有负频率段。在命令窗口键入

```
>> testnyquist(1)
```

运行后得到如图 8 所示的 Nyquist 图。该图也包括了负频段的部分，为了只显示正频段的 Nyquist 图，可以在图 8 所示的界面，点击鼠标右键，出现图 9 所示的界面。接着将鼠标的光标移到“Show”，出现如图 10 所示的界面，此时把“Negative Frequencies”的勾去掉，就得到如图 11 所示的正频段 Nyquist 图。

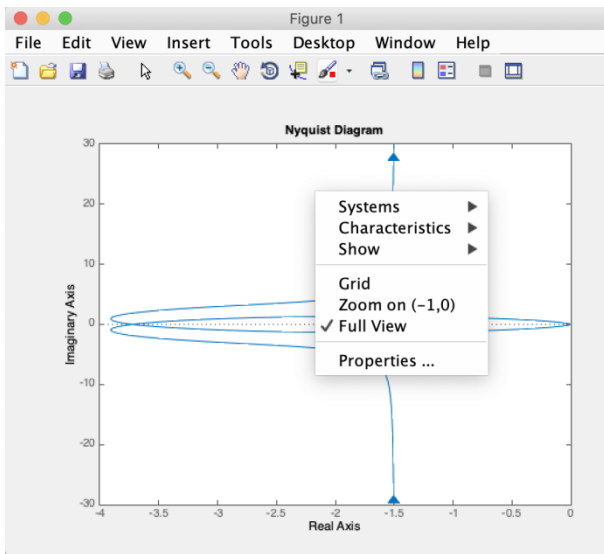


图 9 点击鼠标右键后的全频段 Nyquist 图

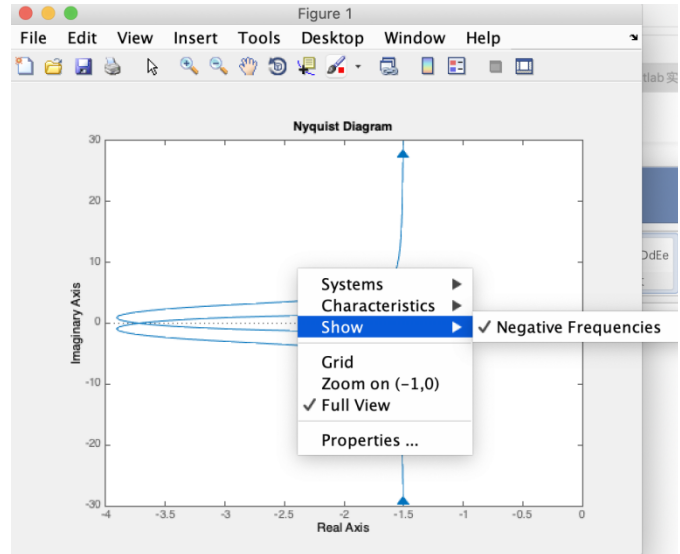


图 10 光标移到 Show 后的 Nyquist 图

5. 编写以开环增益为输入绘制给定频率区间的 Nyquist 图的 m 文件

前面在使用命令 `nyquist` 画 Nyquist 图时，所画图形的频率区间是系统制定的。也可以自己指定频率区间，这需要给指令 `bode` 增加更多的输入。编写如下 m 文件，命名为 `testnyquistwm`

```
function testnyquistwm(K)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[4,1,1]);
sys=tf(num,den)
nyquist(sys,{0.1,1000});
```

类似于指令 `bode` 的用法，在上面的 m 代码中，语句 `nyquist(sys,{0.001,1000})` 中的 `{0.001,1000}` 即为指定的频率区间。注意语法规则：大括号 `{0.001,1000}` 中表示频段的两个端点一定都是大于 0 的正数。在命令窗口键入

```
>> testnyquistwm(1)
```

运行后得到如图 12 所示的 Nyquist 图。可以看到，虽然指定的是正区间的频段，但是画出的 Nyquist 图也包含了对称的幅频段。如果只需要正频段的 Nyquist 图，可以按照前述方法去掉负频段的 Nyquist 曲线。

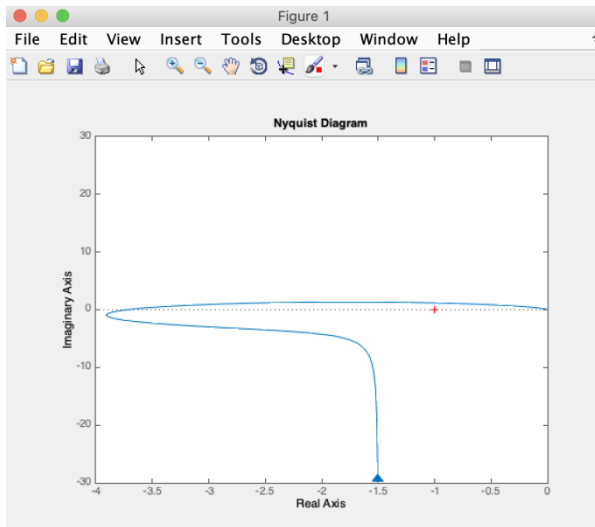


图 11 正频段 Nyquist 图

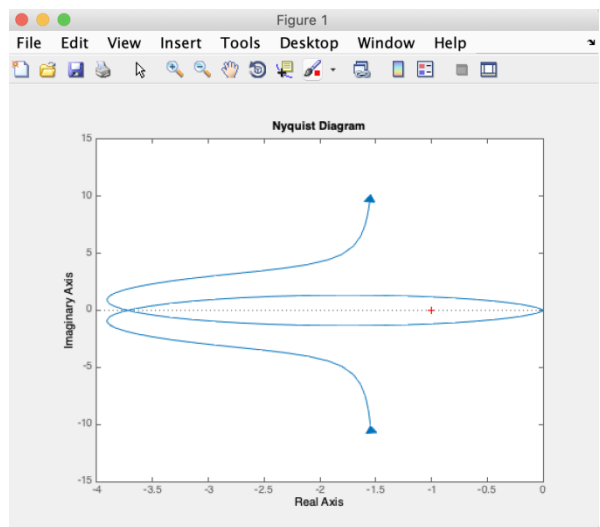


图 12 有限频段的 Nyquist 图

6. 编写以开环增益和指定频率点为输入的 m 文件

指令nyquist除了可以画出Nyquist图外，还能返回给定频率点的频率响应的实部和虚部。编写如下函数，保存为testnyquistfre

```
function testnyquistfre(K,w)
num=[2*K,K];
den=conv(conv([1,0],[2.5,1]),[4,1,1]);
sys=tf(num,den);
[Re,Im]=nyquist(sys,w)
```

在命令窗口键入以下命令

```
>> testnyquistfre(2,0.2)
```

运行后得到如图 13 所示的结果。这说明，开环传递函数

$$G(s) = \frac{2(2s + 1)}{s(2.5s + 1)(4s^2 + s + 1)}$$

的频率特性在频率 $\omega = 0.2$ 时的实部是-3.4764，虚部为-10.6009。

频率点也可以选多个，这个用向量表示。在命令窗口键入以下命令

```
>> testnyquistfre(2,[0.2,0.4,1])
```

运行后得到如图 14 所示的结果。

7. 其余任务

其余任务可以根据前述方法完成。以二阶振荡环节的阻尼比对 Nyquist 图的影响为例。编写如下函数，保存为 `secondnyquist`

```
function secondnyquist(T,xi)
num=1;
den=[T^2,2*T*xi,1];
sys=tf(num,den);
nyquist(sys);
axis([-3 3 -6 1]);
hold on;
```

在 Matlab 命令窗依次渐入以下命令

```
>> secondnyquist(1,0.1)
>> secondnyquist(1,0.3)
>> secondnyquist(1,0.5)
>> secondnyquist(1,0.7)
>> secondnyquist(1,0.8)
```

运行后得到如图 15 所示的全频段 Nyquist 图。按前述方法可以得到如图 16 所示的正频段 Nyquist 图。根据对比图可知，二阶振荡环节阻尼比越小，Nyquist 图越胖。

```
>> testnyquistfre(2,0.2)
|
Re =
    -3.4764

Im =
   -10.6009
```

图 13 在给定频率点的频率响应的实部和虚部

```
>> testnyquistfre(2,[0.2,0.4,1])
Re(:,:,1) =
    -3.4764
Re(:,:,2) =
    -6.8370
Re(:,:,3) =
    -0.1241
Im(:,:,1) =
   -10.6009
Im(:,:,2) =
    -4.9033
Im(:,:,3) =
     0.5103
```

图 14 在给定频率点的频率响应

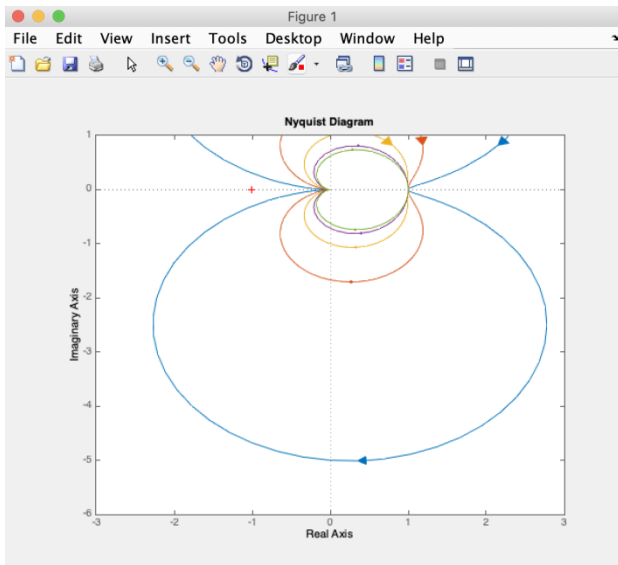


图 15 二阶振荡环节全频段 Nyquist 图对比

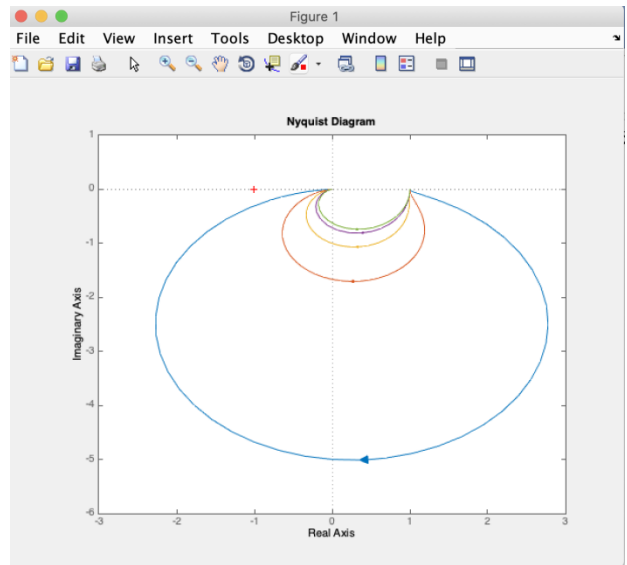


图 16 二阶振荡环节正频段 Nyquist 图对比

四、 实验前的准备工作

1. 复习 Nyquist 图的概略图画法。
2. 复习 Bode 图的概略图画法。
3. 对实验内容中出现的传递函数，用手绘的方式画出 Nyquist 图和 Bode 图。

五、 实验报告要求

针对实验内容编写代码，将代码截图和获取相应曲线的代码写入实验报告。对实验曲线进行分析，给出结论。