

控制理论基础实验教程

(基于 MATLAB 语言)



二〇一七年五月

目 录

实验 1 控制系统的模型建立.....	1
一、实验目的	1
二、实验原理	1
三、实验内容	7
四、实验报告要求	8
实验 2 控制系统的暂态特性分析	9
一、实验目的	9
二、实验原理	9
三、实验内容	11
四、实验报告要求	11
实验 4 系统的频率特性分析	15
一、实验目的	15
二、实验原理	15
三、实验内容	16
四、实验报告要求	16
实验 6 极点配置与全维状态观测器的设计	19
一、实验目的	19
二、实验原理	19
三、实验内容	19
四、实验报告要求	20
参考文献	21

实验 1 控制系统的模型建立

一、实验目的

1. 掌握利用 MATLAB 建立控制系统模型的方法。
2. 掌握系统的各种模型表述及相互之间的转换关系。
3. 学习和掌握系统模型连接的等效变换。

二、实验原理

1. 系统模型的 MATLAB 描述

系统的模型描述了系统的输入、输出变量以及内部各变量之间的关系，表征一个系统的模型有很多种，如微分方程、传递函数、状态空间方程等。这里主要介绍系统多项式型传递函数（TF）模型、零极点型传递函数（ZPK）模型和状态空间方程（SS）模型的 MATLAB 描述方法。

1) 传递函数（TF）模型

传递函数是描述线性定常系统输入-输出关系的一种最常用的数学模型，其表达式一般为

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

在 MATLAB 中，直接使用行向量分子分母多项式的表示系统，即

`num = [bm, bm-1, ..., b1, b0]`

`den = [an, an-1, ..., a1, a0]`

调用 `tf` 函数可以建立传递函数 TF 对象模型，调用格式如下：

`Gtf = tf(num,den)`

`Tfdata` 函数可以从 TF 对象模型中提取分子分母多项式，调用格式如下：

`[num,den] = tfdata(Gtf)` 返回 `cell` 类型的分子分母多项式系数

`[num,den] = tfdata(Gtf,'v')` 返回向量形式的分子分母多项式系数

例 E2-1 采用 MATLAB 建立某一系统的传递函数模型，已知其微分方程为

$$y^{(4)} + 3y^{(3)} + 8y'' + 4y' + 2y = 3u'' + 2u' + 8u$$

解：首先写出描述该系统的传递函数模型的分子分母多项式系数向量：

`>> num = [3 2 8];`

`>> den = [1 3 8 4 2];`

然后调用 `tf` 函数建立系统模型：

```
>> G = tf(num,den)
```

运行结果为:

Transfer function:

$$\frac{3s^2 + 2s + 8}{s^4 + 3s^3 + 8s^2 + 4s + 2}$$

2) 零极点增益 (ZPK) 模型

传递函数因式分解后可以写成

$$G(s) = \frac{k(s-z_1)(s-z_2)\dots(s-z_m)}{(s-p_1)(s-p_2)\dots(s-p_n)} \quad (1-2)$$

式中, z_1, z_2, \dots, z_m 称为传递函数的零点, p_1, p_2, \dots, p_n 称为传递函数的极点, k 为传递系数 (系统增益)。

在 MATLAB 中, 直接用 $[z,p,k]$ 矢量组表示系统, 其中 z, p, k 分别表示系统的零极点及其增益, 即:

```
z=[z1,z2,...,zm];
```

```
p=[p1,p2,...,pn];
```

```
k=[k];
```

调用 `zpk` 函数可以创建 ZPK 对象模型, 调用格式如下:

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

同样, MATLAB 提供了 `zpkdata` 命令用来提取系统的零极点及其增益, 调用格式如下:

```
[z,p,k] = zpkdata(Gzpk) 返回 cell 类型的零极点及增益
```

```
[z,p,k] = zpkdata(Gzpk,'v') 返回向量形式的零极点及增益
```

函数 `pzmap` 可用于求取系统的零极点或绘制系统的零极点图, 调用格式如下:

`pzmap(G)` 在复平面内绘出系统模型的零极点图。

`[p,z] = pzmap(G)` 返回的系统零极点, 不作图。

3) 状态空间 (SS) 模型

由状态变量描述的系统模型称为状态空间模型, 由状态方程和输出方程组成:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (1-3)$$

其中: x 为 n 维状态向量; u 为 r 维输入向量; y 为 m 维输出向量; A 为 $n \times n$ 方阵, 称为系统矩阵; B 为 $n \times r$ 矩阵, 称为输入矩阵或控制矩阵; C 为 $m \times n$ 矩阵, 称为输出矩阵; D 为 $m \times r$ 矩阵, 称为直接传输矩阵。

在 MATLAB 中, 直接用矩阵组 $[A,B,C,D]$ 表示系统, 调用 `ss` 函数可以创建 SS 对象模

型，调用格式如下：

$G_{ss} = ss(A,B,C,D)$

同样，MATLAB 提供了 `ssdata` 命令用来提取系统的 A、B、C、D 矩阵，调用格式如下：

$[A,B,C,D] = ssdata(G_{ss})$ 返回系统模型的 A、B、C、D 矩阵

例 E2-2 已知控制系统的状态空间方程如下

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ -8 & -4 \end{bmatrix} x + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u \\ y = [1 \ 0] x \end{cases}$$

试用 MATLAB 建立系统模型。

解：首先写出系统的 A、B、C、D 矩阵：

```
>> A = [0 1;-8 -4];
```

```
>> B = [0;2];
```

```
>> C = [1 0];
```

```
>> D = [0];
```

然后调用 `ss` 函数建立系统模型：

```
>> Gss = ss(A,B,C,D)
```

运行结果为：

a =

```
      x1  x2
x1      0   1
x2     -8  -4
```

b =

```
      u1
x1      0
x2      2
```

c =

```
      x1  x2
y1      1   0
```

d =

```
      u1
y1      0
```

4) 三种模型之间的转换

上述三种模型之间可以互相转换，MATLAB 实现方法如下

TF 模型→ZPK 模型：zpk(SYS)或 tf2zp(num,den)

TF 模型→SS 模型：ss(SYS)或 tf2ss(num,den)

ZPK 模型→TF 模型：tf(SYS)或 zp2tf(z,p,k)

ZPK 模型→SS 模型：ss(SYS)或 zp2ss(z,p,k)

SS 模型→TF 模型：tf(SYS)或 ss2tf(A,B,C,D)

SS 模型→ZPK 模型：zpk(SYS)或 ss2zp(A,B,C,D)

例 E2-3 已知某系统的传递函数模型，试建立其零极点增益模型，并绘制零极点图。其传递函数为

$$G(s) = \frac{s^2 + 9s + 20}{s^3 + 6s^2 + 11s + 6}$$

解：首先建立系统的传递函数模型描述：

```
>> num = [1 9 20];
>> den = [1 6 11 6];
>> Gtf = tf(num,den)
```

运行结果为：

Transfer function:

```
      s^2 + 9 s + 20
-----
s^3 + 6 s^2 + 11 s + 6
```

然后调用 zpk 函数，实现从传递函数模型到零极点增益模型的转换：

```
>> Gzpk = zpk(Gtf)
```

运行结果为：

Zero/pole/gain:

```
      (s+5) (s+4)
-----
(s+3) (s+2) (s+1)
```

调用 pzmap 函数绘制系统零极点图，结果如图 1-1 所示：

```
>> pzmap(Gzpk);
>> grid on
```

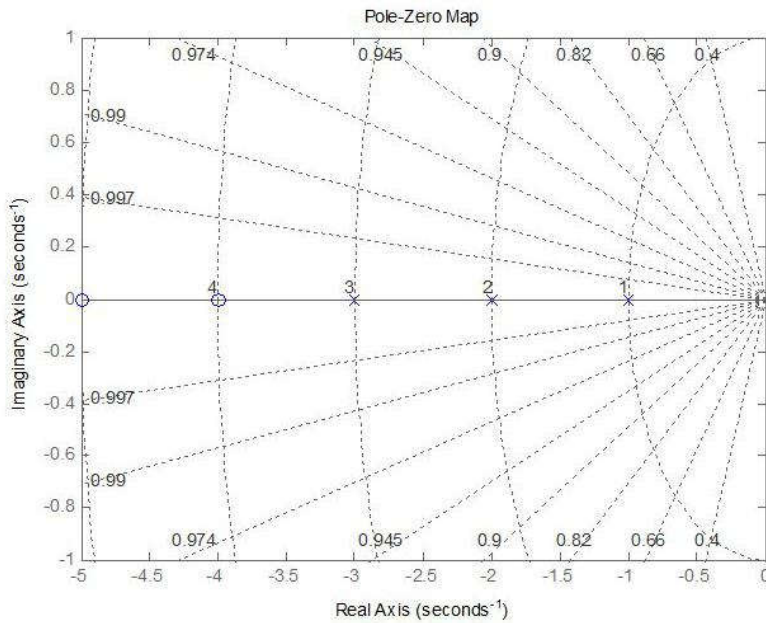


图 1-1 系统零极点图

2. 系统模型的连接

在实际应用中，整个控制系统是由多个单一的模型组合而成，基本的组合方式有串联连接、并联连接和反馈连接。图 1-2 分别为串联连接、并联连接和反馈连接的结构框图和等效总传递函数。

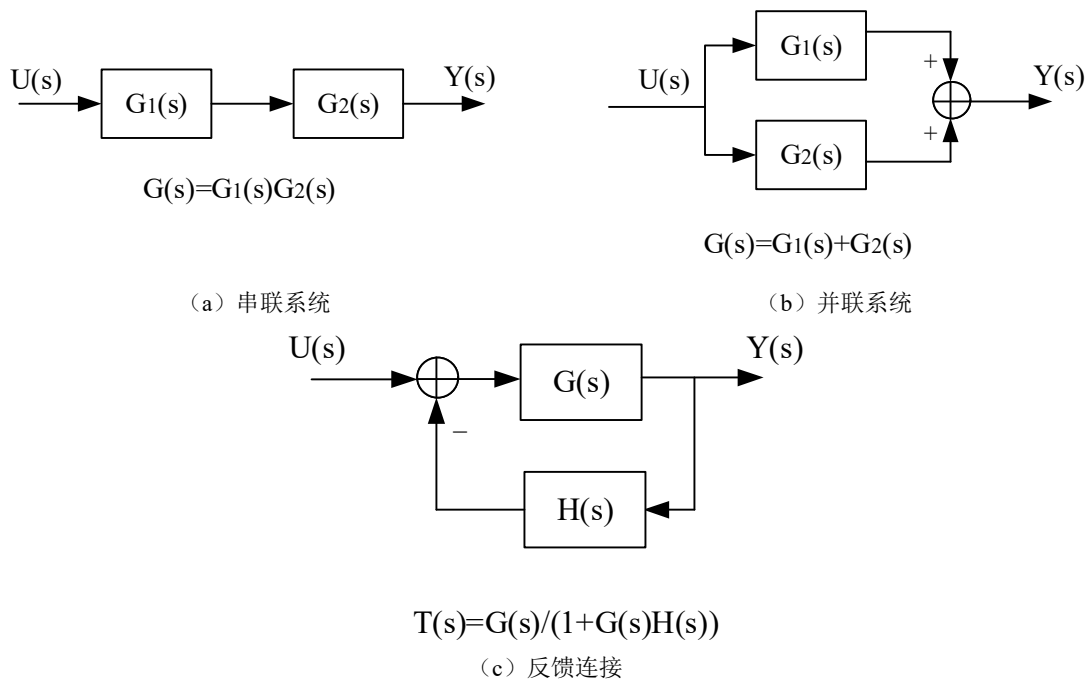


图 1-2 串联连接、并联连接和反馈连接

在 MATLAB 中可以直接使用 “*” 运算符实现串联连接，使用 “+” 运算符实现并联连接。反馈系统传递函数求解可以通过命令 `feedback` 实现，调用格式如下：

```
T = feedback(G,H)
```

```
T = feedback(G,H,sign)
```

其中， G 为前向传递函数， H 为反馈传递函数；当 $\text{sign} = +1$ 时， GH 为正反馈系统传递函数；当 $\text{sign} = -1$ 时， GH 为负反馈系统传递函数；默认值是负反馈系统。

例 E2-3 两个系统串联，已知两个系统传递函数分别为

$$G_1(s) = \frac{s+3}{(s+1)(s+2)} \quad G_2(s) = \frac{3s^2 + s + 4}{5s^2 + 12s + 3}$$

解：首先分别建立两个系统的传递函数模型：

```
>> num1 = [1 3];
```

```
>> den1 = conv([1 1],[1 2]); %使用 conv 命令实现多项式相乘
```

```
>> G1=tf(num1,den1) %创建 G1(s)描述的传递函数模型
```

Transfer function:

$s + 3$

$s^2 + 3s + 2$

```
>> num2 = [3 1 4];
```

```
>> den2 = [5 12 3];
```

```
>> G2 = tf(num2,den2) %创建 G2(s)描述的传递函数模型
```

Transfer function:

$3s^2 + s + 4$

$5s^2 + 12s + 3$

使用 “*” 运算符实现串联连接：

```
>> G = G2*G1
```

运行结果为：

Transfer function:

$3s^3 + 10s^2 + 7s + 12$

$5s^4 + 27s^3 + 49s^2 + 33s + 6$

三、实验内容

1. 已知控制系统的传递函数如下

$$G(s) = \frac{2s^2 + 18s + 40}{s^3 + 5s^2 + 8s + 6}$$

试用 MATLAB 建立系统的传递函数模型、零极点增益模型及系统的状态空间方程模型，并绘制系统零极点图。

2. 已知控制系统的状态空间方程如下

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & -2 & -3 & -4 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u \\ y = [10 \quad 2 \quad 0 \quad 0] x \end{cases}$$

试用 MATLAB 建立系统的传递函数模型、零极点增益模型及系统的状态空间方程模型，并绘制系统零极点图。

3. 已知三个系统的传递函数分别为

$$G_1(s) = \frac{2s^2 + 6s + 5}{s^3 + 4s^2 + 5s + 2}$$

$$G_2(s) = \frac{s^2 + 4s + 1}{s^3 + 9s^2 + 8s}$$

$$G_3(s) = \frac{5(s+3)(s+7)}{(s+1)(s+4)(s+6)}$$

试用 MATLAB 求上述三个系统串联后的总传递函数。

4. 已知如图 E2-1 所示的系统框图

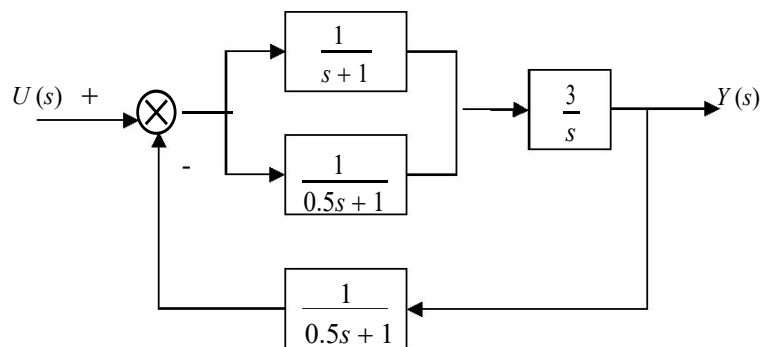


图 E2-1

试用 MATLAB 求该系统的闭环传递函数。

5. 已知如图 E2-2 所示的系统框图

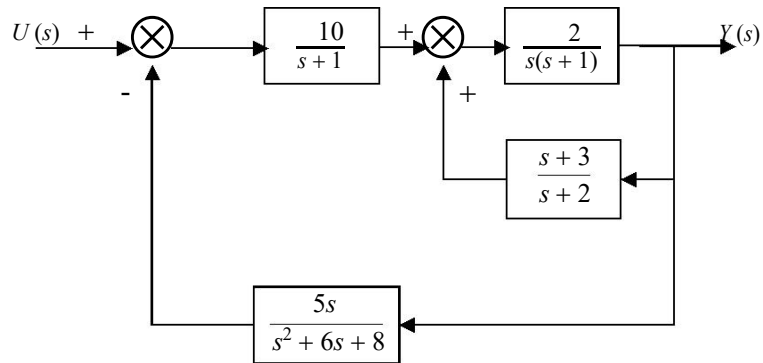


图 E2-2

试用 MATLAB 求该系统的闭环传递函数。

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 2 控制系统的暂态特性分析

一、实验目的

1. 学习和掌握利用 MATLAB 进行系统时域响应求解和仿真的方法。
2. 考察二阶系统的时间响应，研究二阶系统参数对系统暂态特性的影响。

二、实验原理

1. 系统的暂态性能指标

控制系统的暂态性能指标常以一组时域量值的形式给出，这些指标通常根据系统的单位阶跃响应给出定义，指标分别为：

- (1) 延迟时间 t_d ：响应曲线首次到达稳态值的 50%所需的时间。
- (2) 上升时间 t_r ：响应曲线从稳态值的 10%上升到 90%所需要的时间，对于欠阻尼系统，指响应曲线首次到达稳态值所需的时间。
- (3) 峰值时间 t_p ：响应曲线第一次到达最大值的时间。
- (4) 调整时间 t_s ：响应曲线开始进入并保持在允许的误差（ $\pm 2\%$ 或 $\pm 5\%$ ）范围内所需要的时间。
- (5) 超调量 σ ：响应曲线的最大值和稳态值之差，通常用百分比表示

$$\sigma = \frac{y(t_p) - y(\infty)}{y(\infty)} \times 100\%$$

其中 $y(t)$ 为响应曲线。

在 MATLAB 中求取单位阶跃响应的函数为 `step`，其使用方法如下

`step(sys)` 在默认的时间范围内绘出系统响应的时域波形

`step(sys,T)` 绘出系统在 $0 - T$ 范围内响应的时域波形

`step(sys,ts:tp:te)` 绘出系统在 $t_s - t_e$ 范围内，以 t_p 为时间间隔取样的响应波形

`[y,t] = step(...)` 该调用格式不绘出响应波形，而是返回响应的数值向量及其对应的时间向量。

系统的暂态性能指标可以根据上述定义，在响应曲线上用鼠标读取关键点或通过搜索曲线对应的数值向量中关键点来确定。

2. LTI Viewer 工具

在 MATLAB 中提供了线性时不变系统仿真的工具 `LTI Viewer`，可以方便地观察系统的响应曲线和性能指标。在命令窗口中键入 `ltiview` 即可启动 `LTI Viewer`。这里简要介绍 `LTI Viewer` 工具（如图 2-1 所示）的使用方法。

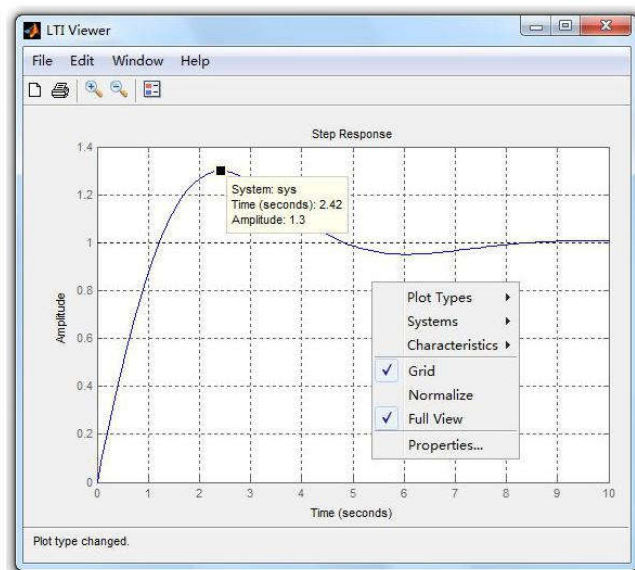


图 2-1 LTI Viewer 工具运行界面

1) 【File】菜单

Import 选项：可以从 Workspace 或 MAT 文件中导入系统模型。

Export 选项：将当前窗口中的对象模型保存到 Workspace 或文件中。

Toolbox preferences 选项：属性设置功能，可以设置控制系统中的各种属性值。

Page Setup 选项：页面设置功能，可以对打印输出和显示页面进行设置。

2) 【Edit】菜单

Plot Configuration 选项：对显示窗口及显示内容进行配置。

Line Style 选项：线型设置功能，可以对输出响应曲线的线型进行设置。

Viewer Preferences 选项：对当前窗口的坐标、颜色、字体、响应曲线的特性参数等属性进行设置。

3) 右键菜单

在运行界面上点击鼠标右键，将会弹出一个弹出式菜单，菜单上各选项的功能分别为：

Plot Types：选择绘制的系统曲线类型，可选的类型有单位阶跃响应、单位冲激响应、波特图、奈奎斯特图、零极点图等。

System：选择需要仿真的系统。

Characteristic：系统的性能指标选项。

Grid：显示和关闭网格。

Normalize：正常显示模式。

Full View：满界面显示模式。

Properties：性能编辑器选项，可以设置画面的标题、坐标标志、坐标范围、线型、颜色、性能指标等。

三、实验内容

1. 已知单位负反馈系统前向通道的传递函数为

$$G(s) = \frac{80}{s^2 + 2s}$$

试用 MATLAB 绘制系统的单位阶跃响应曲线。

2. 已知二阶系统

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

(1) $\zeta = 0.6$, $\omega_n = 5$, 试用 MATLAB 绘制系统单位阶跃响应曲线, 并求取系统的暂态性能指标。

(2) $\omega_n = 1$, ζ 为 0、0.707、1、2 时, 求此系统的单位阶跃响应。

(3) $\zeta = 0.5$, ω_n 为 1、5、10 时, 求此系统的单位阶跃响应。

(4) 观察上述实验结果, 分析这两个特征参数对系统暂态特性的影响。

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果, 程序代码中在必要的地方应加上注释, 必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法, 谈谈你的收获和体会。

实验 4 系统的频率特性分析

一、实验目的

为学习和掌握利用 MATLAB 绘制系统 Nyquist 图和 Bode 图的方法。

为学习和掌握利用系统的频率特性分析系统的性能。

二、实验原理

系统的频率特性是一种图解方法，运用系统的开环频率特性曲线，分析闭环系统的性能，如系统的稳态性能、暂态性能。常用的频率特性曲线有 Nyquist 图和 Bode 图。在 MATLAB 中，提供了绘制 Nyquist 图和 Bode 图的专门函数。

1. Nyquist 图

nyquist 函数可以用于计算或绘制连续时间 LTI 系统的 Nyquist 频率曲线，其使用方法如下：

nyquist(sys) 绘制系统的 Nyquist 曲线。nyquist(sys,w) 利用给定的频率向量 w 来绘制系统的 Nyquist 曲线。

[re,im]=nyquist(sys,w) 返回 Nyquist 曲线的实部 re 和虚部 im，不绘图。

2. Bode 图

bode 函数可以用于计算或绘制连续时间 LTI 系统的 Bode 图，其使用方法如下：

bode(sys) 绘制系统的 Bode 图。

bode(sys,w) 利用给定的频率向量 w 来绘制系统的 Bode 图。

[mag,phase]=bode(sys,w) 返回 Bode 图数据的幅度 mag 和相位 phase，不绘图。

3. 幅值裕度和相位裕度计算

margin 函数可以用于从频率响应数据中计算出幅值裕度、相位裕度及其对应的角频率，其使用方法如下：

margin(sys)

margin(mag,phase,w)

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

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

其中不带输出参数时，可绘制出标有幅值裕度和相位裕度的 Bode 图；带输出参数时，返回幅值裕度 Gm、相位裕度 Pm 及其对应的角频率 Wcg 和 Wcp。

三、实验内容

1. 已知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。

2. 已知单位负反馈系统的开环传递函数为

$$G(s) = \frac{10[(\frac{5}{4}s)^2 + \frac{5}{4}s + 1]}{s^2(\frac{10}{3}s + 1)(\frac{0.2}{3}s + 1)(\frac{1}{40}s + 1)}$$

- (1) 绘制闭环系统的零极点图，根据零极点分布判断系统的稳定性。
 (2) 绘制系统 Bode 图，求出幅值裕度和相位裕度，判断闭环系统的稳定性。

3. 已知系统的开环传递函数为

$$G(s) = \frac{K}{s(s+1)(0.1s+1)}$$

分别判断当开环放大系数 $K = 5$ 和 $K = 20$ 时闭环系统的稳定性，并求出幅值裕度和相位裕度。

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

实验 6 极点配置与全维状态观测器的设计

一、实验目的

1. 加深对状态反馈作用的理解。
2. 学习和掌握状态观测器的设计方法。

二、实验原理

在 MATLAB 中，可以使用 `acker` 和 `place` 函数来进行极点配置，函数的使用方法如下：

$K = \text{acker}(A, B, P)$ A , B 为系统系数矩阵， P 为配置极点， K 为反馈增益矩阵。

$K = \text{place}(A, B, P)$ A , B 为系统系数矩阵， P 为配置极点， K 为反馈增益矩阵。

$[K, \text{PREC}, \text{MESSAGE}] = \text{place}(A, B, P)$ A , B 为系统系数矩阵， P 为配置极点， K 为反馈增益矩阵， PREC 为特征值， MESSAGE 为配置中的出错信息。

三、实验内容

1. 已知系统

$$\dot{x} = \begin{bmatrix} -2 & -1 & 1 \\ 1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} u$$

- (1) 判断系统稳定性，说明原因。
- (2) 若不稳定，进行极点配置，期望极点：-1，-2，-3，求出状态反馈矩阵 k 。
- (3) 讨论状态反馈与输出反馈的关系，说明状态反馈为何能进行极点配置？
- (4) 使用状态反馈进行零极点配置的前提条件是什么？

2. 已知系统

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = [2 \quad 0] x$$

设计全维状态观测器，使观测器的极点配置在 $-12 \pm j$ 。

- (1) 给出原系统的状态曲线。（设 $x(0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ ， $u(t) = 1(t)$ ）
- (2) 给出观测器的状态曲线并加以对比。（观测器的初始状态可以任意选取）

观察实验结果，思考以下问题：

- (1) 说明反馈控制闭环期望极点和观测器极点的选取原则。
- (2) 说明观测器的引入对系统性能的影响。

四、实验报告要求

1. 简述实验目的和实验原理。
2. 列出完成各项实验内容所编写的程序代码并给出实验结果，程序代码中在必要的地方应加上注释，必要时应对实验结果进行分析。
3. 总结实验中遇到的问题及解决方法，谈谈你的收获和体会。

参考文献

1. 王士宏, 周思永. 控制理论基础. 北京: 北京理工大学出版社, 2002.
2. 韩绍坤, 许向阳, 王晓华. 自动控制原理. 北京: 北京理工大学出版社, 2009.
3. 刘坤. MATLAB 自动控制原理习题精解. 北京: 国防工业出版社, 2004.
4. 黄忠霖. 自动控制原理的 MATLAB 实现. 北京: 国防工业出版社, 2009.
5. 王海英, 袁丽英, 吴勃. 控制系统的 MATLAB 仿真与设计. 北京: 高等教育出版社, 2009.
6. Frederick D. K., Chow J. H. 著. 张彦斌译. 反馈控制问题—使用 MATLAB 及其控制系统工具箱. 西安: 西安交通大学出版社, 2001.