



**本科实验报告**

|  |  |
| --- | --- |
| 姓名： | Awslasasd |
| 学院： |  |
| 系： |  |
| 专业： |  |
| 学号： |  |
| 指导教师： | 韩涛 |

2024年 9 月 20 日

**实验报告**

装订线

课程名称： Mworks仿真实验 实验类型： 仿真实验

实验项目名称： 现代控制理论仿真实验(1)

学生姓名： 专业： 学号：

同组学生姓名： 指导老师： 韩涛

实验地点： 实验日期： 2024 年 9 月 19 日

一、实验目的和要求（必填）

1、熟悉 MATLAB 中频域的分析方法，掌握用 MATLAB 进行系统频域分析与设

计。

2、熟悉 MATLAB 中离散系统的分析方法，掌握用 MATLAB 进行离散系统分析

与设计。

3、熟悉 MATLAB 中状态空间系统的分析方法，掌握用 MATLAB 进行状态空间系统分析与设计

二、实验内容和原理（必填）

2.1 频域分析与设计

2.1.1 Nyquist 稳定性判据

单位反馈开环系统  绘制系统 Nyquist 曲线，判断闭环系统的稳

定性，绘制出闭环系统的脉冲响应。

2.1.2 稳定裕度

控制系统的传递函数为

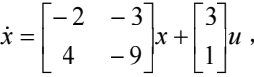
用对数频率特性确定相位裕度大于 45°时的 Km 值

2.2 离散系统分析与设计

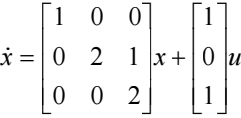
离散二阶系统*H*求当输入为幅值± 1 的方波信号时系统的输出响应。

2.3状态空间系统的分析方法

2.3.1 状态反馈矩阵的设计

已知受控系统为  设计状态反馈阵 K，使系统闭环极点为-1+j2, -1-j2. (分别采用上课所讲方法直接编程和 Mworks函数 place 或 acker方法)

2.3.2 全维状态观测器的设计

已知系统  设计全维状态观测器，使其极点为 3, 4, 5。(分别采用

*y* = [1 1 0]*x*

书上的方法直接编程和 Mworks 函数 estim 方法)

三、主要仪器设备（系统、软件或平台）

1.Windows电脑一台

2.Mworks-sylab 2024b 软件

四、实验代码

4.1 频域分析与设计

4.1.1 Nyquist 稳定性判据

具体代码如下：

1. clc()
2. clear()
3. *# 定义极点和增益*
4. z = []
5. p = [-1 -5 2]
6. k = 50
7. *# 创建零极点增益模型*
8. sys = zpk(z,p,k)
9. *# 绘制系统的奈奎斯特图*
10. nyquist(sys)

得到的奈奎斯特图如下图所示

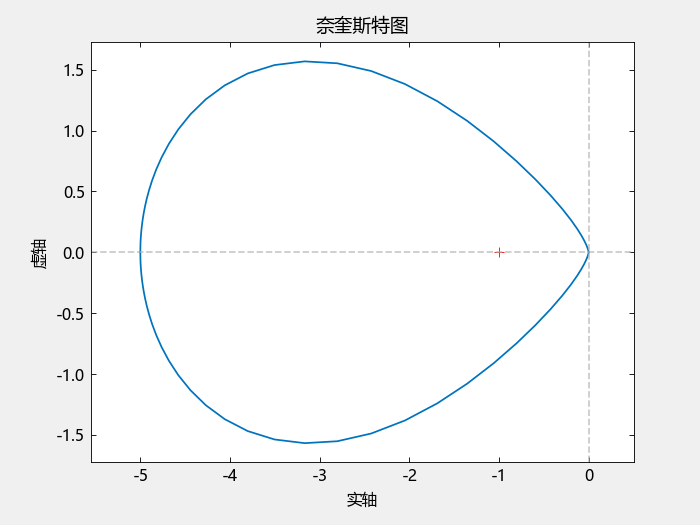


图 1 奈奎斯特图

1. *# 创建负反馈系统*
2. sys1 = feedback(sys,-1)
3. flag = isstable(sys1)
4. println(flag)
5. *# 绘制系统脉冲响应*
6. impulse(sys1)

脉冲响应图像如图 2 所示，系统稳定性结果如下图 3 所示。

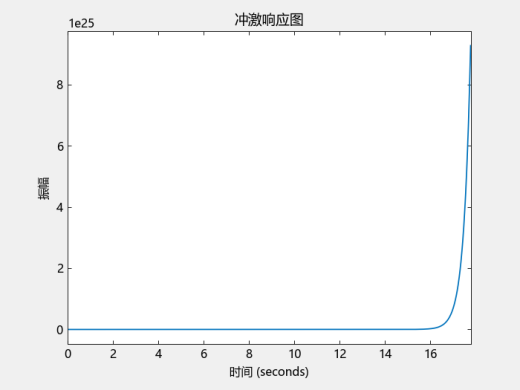
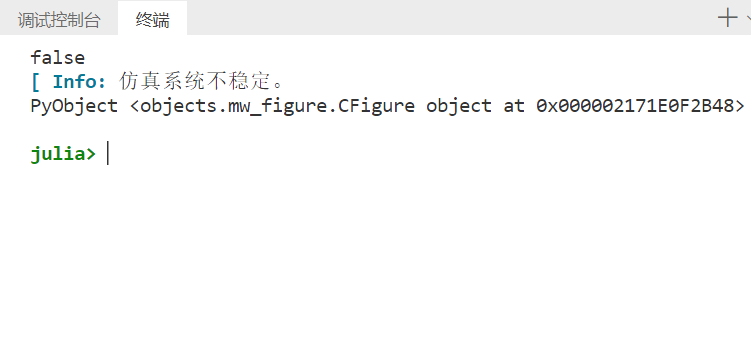
 

图 2 脉冲响应 图 3 稳定性结果

从图中可以看出，Nyquist 图顺时针绕点 (-1,0) 一圈，N=-1，又开环极点为-1，-5，2，有一个在右半平面，PR = 1，可得 ZR = PR − N = 2，因而系统不稳定。

调用 isstable 函数判断闭环系统的稳定性，输出结果为 false，说明系统不稳定，与Nyquist 判据相符。

4.1.2 稳定裕度

代码如下

1. clc()
2. clear()
3. *# 循环遍历k的值，从0到100*
4. for k = 0.00:0.01:100.00
5. *# 定义分子和分母多项式系数*
6. global num = [1 1]
7. global den = conv([1 2],[1 4 5])
8. *# 创建传递函数模型*
9. global sys = tf(num,den)
10. *# 计算增益为k倍的系统增益裕度和相位裕度*
11. global Gm,Pm,Wcg,Wcp = margin(k\*sys,fig = false);
12. if Pm[1] <= 45.0
13. println("km的值为:$k")
14. break;
15. end
16. end

由上面的代码可以得到下面的结果K=**43.69**



图 4 实验1.2结果

4.2 离散系统分析与设计

代码如下

1. clc()
2. clear()
3. *# 定义分子分母的多项式系数*
4. num = [0.632]
5. den = [1 -1.368 0.568]
6. *# 定义采样时间*
7. Ts = 0.001
8. *# 创建离散传递函数*
9. sys = tf(num,den,Ts)
10. *# 方波参数定义*
11. tau = 2 *#方波周期*
12. Tf = 6 *#总时间*
13. Ts = 0.001 *#采样时间*
14. *# 生成方波信号*
15. u0,t = gensig(:square,tau,Tf=Tf,Ts=Ts)
16. *# 调整方波范围到[-1,1]*
17. u = 2\*u0 .- 1
18. lsim(sys,u,t)

运行结果如下图所示

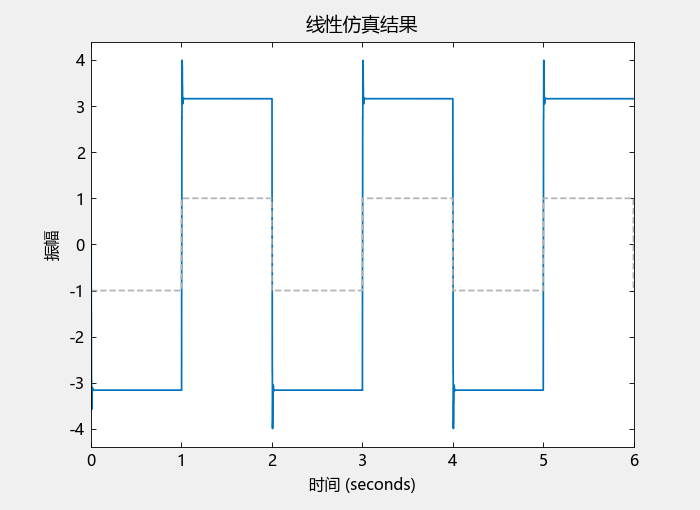


图 5 系统输出响应

4.3状态空间系统的分析方法

4.3.1 状态反馈矩阵的设计

代码如下

1. *# 能控标准型法*
2. clear()
3. clc()
4. *# 定义矩阵和向量*
5. a = [-2 -3;4 -9]
6. b = [3;1]
7. *# 计算可控性矩阵 M*
8. M = ctrb(a,b)
9. println("矩阵的秩：$M")
10. *# 判断系统是否完全能控*
11. if rank(M) < size(a,1)
12. println("系统不完全能控")
13. else
14. println("系统完全能控")
15. *# 定义极点*
16. s1 = [1 1-2im]
17. s2 = [1 1+2im]
18. dest = conv(s1,s2)#目标多项式
19. println("dest is : $dest")
20. *# 求多项式系数*
21. pa = poly(a)
22. println("pa is :$pa")
23. *# 矩阵L*
24. L = [pa[2] 1;1 0]
25. println("L is :$L")
26. *# 控制能观性矩阵Qc*
27. Qc = [b a\*b]
28. *# 计算T矩阵*
29. T = Qc \* L
30. *# 计算T向量*
31. t = dest - pa
32. println("t is:$t")
33. t = t'
34. println("t' is:$t")
35. *# 从t矩阵提取反馈增益向量Kc*
36. Kc = t[:,end:-1:2]
37. *# 计算反馈增益矩阵K*
38. K = Kc\*inv(T)
39. println("K = ",K)
40. end

得到的结果如下图 6 所示

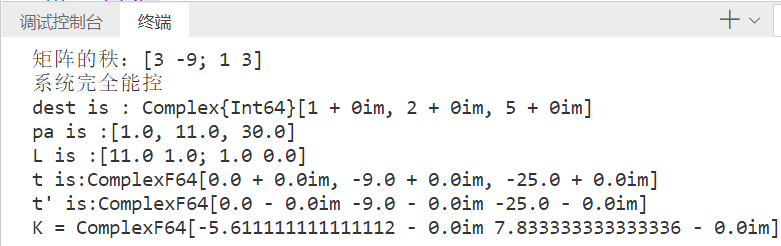


图 6 利用原理计算结果

接下来利用Mworks的函数进行求解

1. # 利用系统函数
2. a = [-2 -3;4 -9];
3. b = [3 1]';
4. p = [-1+2im;-1-2im];
5. #计算可控性矩阵 M
6. M = ctrb(a,b)
7. #判断系统是否完全能控
8. if rank(M) < size(a,1)
9. println("系统不完全能控")
10. else
11. println("系统完全能控")
12. k = acker(a,b,p)
13. m = place(a,b,p)
14. display(m)
15. end

得到的结果如下所示

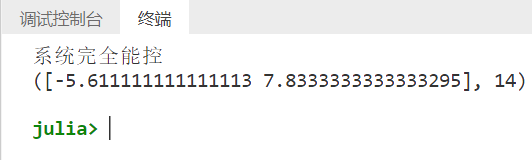


图 7 调用函数结果

可以看到，通过能控标准型的原理进行计算的结果与利用MWorks的place与acker函数的结果基本一致，都为[-5.61,7.83]左右。

4.3.2 全维状态观测器的设计

代码如下

1. clear()
2. clc()
3. *# 定义矩阵*
4. A = [1 0 0;0 2 1;0 0 2]
5. B = [1;0;1]
6. C = [1 1 0]
7. D = [0]
8. M = ctrb(A,B);
9. if rank(M) < size(A,1)
10. println("系统不完全能控")
11. else
13. *# 计算当前系统的多项式*
14. pA = poly(A)
15. println("pA is : $pA")
16. *# 定义目标多项式*
17. s1 = [1 3]
18. s2 = [1 4]
19. s3 = [1 5]
20. dest = conv(conv(s1,s2),s3)
21. println("dest is :$dest")
22. *# 构造L矩阵*
23. L = [pA[3] pA[2] 1;pA[2] 1 0; 1 0 0]
24. println("L is :$L")
25. *#计算可观测性矩阵*
26. Q\_o = [C;C\*A;C\*A^2]
27. *# To\_inv = L\*Qo,计算矩阵的逆*
28. T\_Cinv = L\*Q\_o
29. println("T\_C is:$T\_Cinv")
30. T\_c = inv(T\_Cinv)
31. println("T\_cinv is :$T\_c")
32. *# 计算Ho矩阵*
33. t = dest - pA
34. t = t'
35. println("t is :$t")
36. Ho = t[end:-1:2]
37. println("Ho is :$Ho")
38. *# 计算H矩阵*
39. H = T\_c\*Ho
40. println("H is :$H")
41. *# 计算新的A\_e矩阵*
42. A\_e = A - H\*C
43. println("A\_e is :$A\_e")
44. end

得到的结果如下图所示

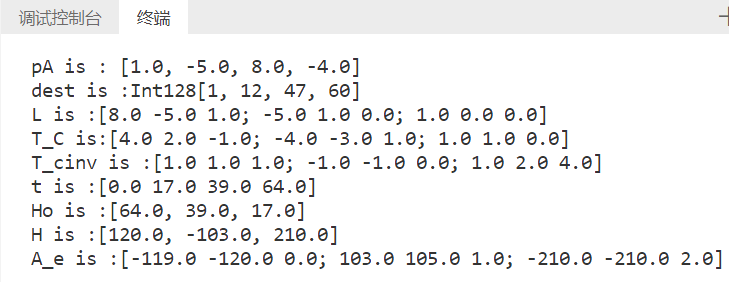


图 9 利用原理计算结果

利用MWorks的系统函数求解的代码如下所示

1. *# 利用系统函数*
2. clear()
3. clc()
4. *# 定义矩阵*
5. A = [1 0 0;0 2 1;0 0 2]
6. B = [1;0;1]
7. C = [1 1 0]
8. D = [0]
9. p = [-3,-4,-5]
10. *# 计算极点配置反馈增益矩阵 k*
11. k = place(A',C',p)
12. *# 使用k'估计系统*
13. G1 = ss(A,B,C,D)
14. est = estim(G1,k[1]')
15. println("k = ",k)

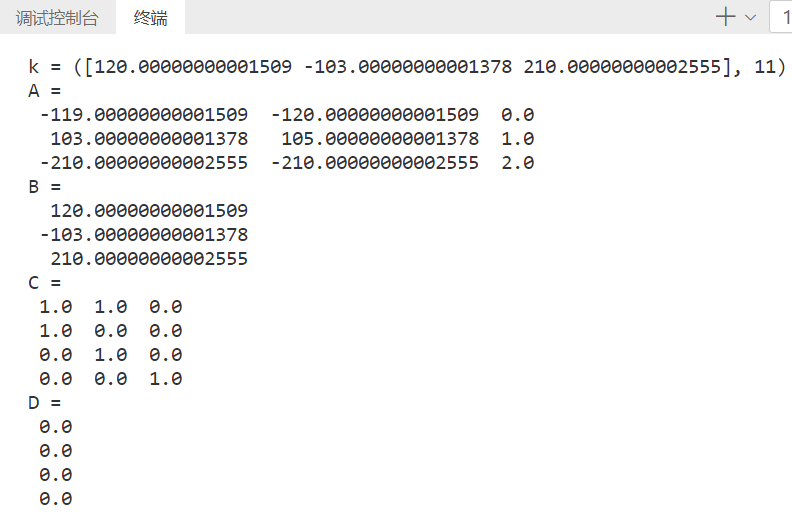


图 10 调用函数结果

从图9与图10可以看出两者的计算结果相同。

五、讨论、心得

5.1 讨论

通过这次实验，我对MWorks在控制系统分析与设计中的应用有了更深入的理解。实验过程中，我熟悉了频域分析方法，掌握了如何利用Nyquist稳定性判据来判断系统的稳定性，并通过实际操作绘制了系统的脉冲响应。同时，我也学会了如何确定系统的稳定裕度，这对于设计具有良好性能的控制系统至关重要。

在离散系统分析与设计部分，我通过编写代码模拟了离散二阶系统对特定输入信号的响应，这让我对离散时间系统的动态行为有了更直观的认识。此外，状态空间系统的分析方法让我对系统的内部动态有了更深刻的理解，我也学会了如何设计状态反馈矩阵和全维状态观测器，以调整系统的特性。

5.2 心得

通过这次实验，我深刻体会到了控制系统分析与设计的重要性，不仅增强了我的专业技能，也为我将来在自动化、数字经济和智能城市建设等领域为国家科技进步和社会发展做出贡献打下了坚实的基础。这次学习经历让我更加坚信，作为工程师，我们有能力也有责任通过技术创新来推动社会向前发展。