



安徽大学
人工智能学院
School of Artificial Intelligence
Anhui University

《自动控制原理》实验报告

控制系统建模及响应曲线绘制

学 院: 人工智能学院

专 业: 机器人工程

学 号: WA2224013

姓 名: 郭义月

指导老师: 赵冬

课程编号: ZH52164

课程学分: 1

提交日期: 2024.11.29

目 录

1	实验内容及要求	3
1.1	实验目的	3
1.2	实验内容	3
1.3	实验要求	3
2	任务 1: 画出系统的结构图	3
2.1	实验框图绘制思路	3
2.1.1	明确系统各环节的作用	3
2.1.2	确定系统的信号流向	4
2.1.3	分块绘制各环节	4
2.1.4	闭环反馈	4
2.1.5	标注传递函数	4
2.2	绘制系统结构图	4
3	任务 2: 求出 K_1, K_m 和 T_m	4
4	任务 3: 利用 MATLAB 求出闭环传递函数	5
4.1	相关 MATLAB 函数介绍	5
4.1.1	tf 函数	5
4.1.2	series 函数	5
4.1.3	feedback 函数	5
4.2	MATLAB 代码与实验结果	6
4.2.1	MATLAB 代码	6
4.2.2	实验结果	6
5	任务 4: 绘制系统在不同输入下的响应曲线	7
5.1	输入轴为幅度为 10 的阶跃响应	7
5.1.1	step 函数	7
5.1.2	MATLAB 代码	7
5.1.3	响应曲线	8
5.2	输入为幅值为 60 的速度函数	8
5.2.1	lsim 函数	8
5.2.2	MATLAB 代码	8
5.2.3	响应曲线	9
6	任务 5: 分析系统的性能	10
6.1	稳定性	10
6.2	准确性	10
6.2.1	输入为阶跃信号时	10
6.2.2	当输入为速度信号时	10
6.3	快速性	11
7	实验总结	11

1 实验内容及要求

1.1 实验目的

实验的目的是研究直流位置随动系统的动态特性和稳态性能，深入分析系统中各环节的作用及其对系统响应的影响。通过实验数据，确定误差检测器的传递系数、电动机的传递系数及时间常数等关键参数，进一步推导系统的开环和闭环传递函数。实验还旨在通过 MATLAB 仿真绘制系统在不同输入条件下的动态响应曲线，观察误差电压随时间的变化规律，分析系统的稳定性和控制精度。此外，通过调整实验参数并验证理论计算结果，探讨系统性能优化的方法，为设计更加高效、精确的控制系统提供参考依据。这一实验有助于掌握随动系统的基本原理和特性，同时提高对控制系统分析与设计的综合能力。

1.2 实验内容

如下图 (a) 所示的直流位置随动系统中, 已知放大器增益 $K_a = 10$, 减速齿轮的齿数 $z_1 = 20, z_2 = 50$ 。当输入轴 $\theta_i = 10$ 时, 实验测得误差电压的曲线如图下图 (b) 所示, 当输入 $\theta_i = 60 * i$ 时, 实验测得稳态误差 $u_s(\infty) = 0.145$

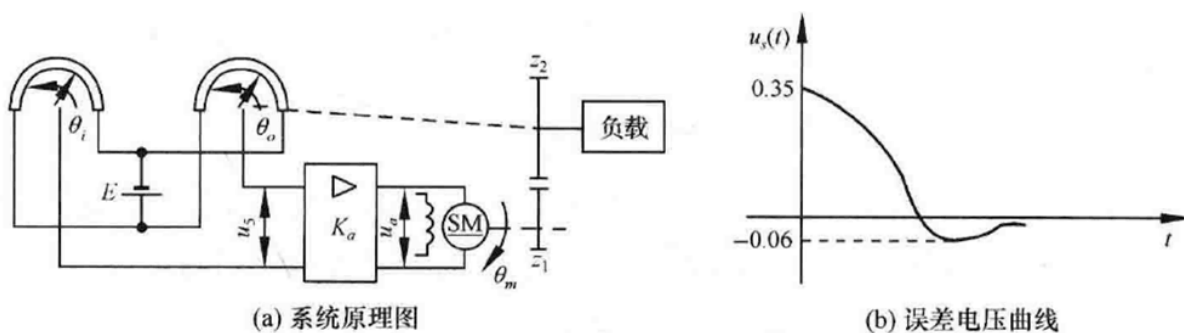


图 1: 直流位置随动系统

1.3 实验要求

- (1) 画出系统的结构图;
- (2) 求出电位器误差检测器传递函数 K_1 , 直流电动机的传递系数 K_m 和时间常数 T_m ;
- (3) 利用 MATLAB 求出闭环传递函数并绘制输入轴为 $\theta_i = 10 * 1(t)$ 和 $\theta_i = 60 * t$ 时的响应曲线。

2 任务 1: 画出系统的结构图

2.1 实验框图绘制思路

2.1.1 明确系统各环节的作用

根据原理图, 系统包含输入轴角度 θ_i 、输出轴角度 θ_o 、误差电压 u_s 、放大器增益 K_a 、电动机 (SM) 及其角度输出 θ_m 、减速齿轮传动比 $\frac{z_1}{z_2}$ 和负载。

2.1.2 确定系统的信号流向

输入信号 θ_i 与反馈信号 θ_o 经过比较，产生误差电压 $u_s = \theta_i - \theta_o$ 。经过放大 K_a 产生驱动电压 u_a ，作为电动机的输入信号。电动机输出角速度被传递至减速齿轮，经过减速比 $\frac{z_1}{z_2}$ 后作为输出轴角度 θ_o 。

2.1.3 分块绘制各环节

误差比较器：绘制比较器，输入 θ_i 和 θ_o ，输出误差 u_s 。

放大器：绘制增益块 K_a ，输入 u_s ，输出 u_a 。

电动机：绘制电动机块，输入 u_a ，输出轴角速度 ω_m 。

减速器：绘制减速比块 $\frac{z_1}{z_2}$ ，输入 ω_m ，输出 θ_o 。

2.1.4 闭环反馈

将输出角度 θ_o 通过反馈回路回输入，完成闭环控制结构。

2.1.5 标注传递函数

标注每个模块的传递函数：误差检测器的系数 K_1 ，放大器增益 K_a ，电动机的传递函数 K_m 及时间常数 T_m ，以及减速比 $\frac{z_1}{z_2}$ 。

2.2 绘制系统结构图

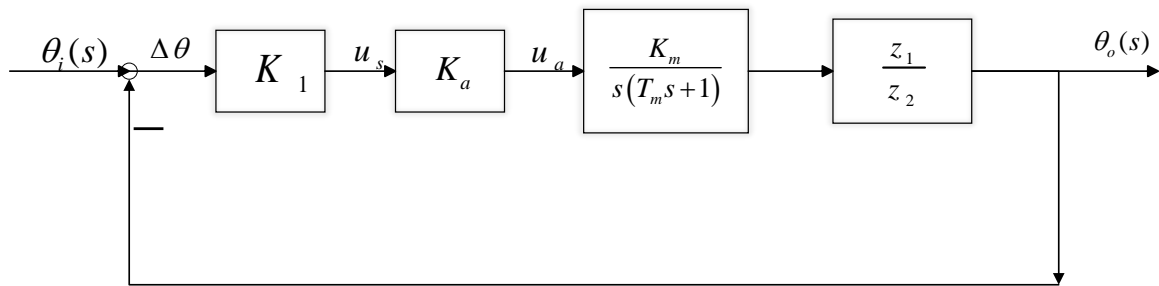


图 2: 系统的结构图

3 任务 2：求出 K_1, K_m 和 T_m

由流程图可知，系统开环传递函数为 $G(s) = \frac{K_1 K_a K_m}{s(T_m s + 1)} = \frac{4K_1 K_m}{s(T_m s + 1)}$

当输入为 $\theta = 60 * t$ 时，稳态误差为

$$u_s(\infty) = K_1 \Delta\theta(\infty) = \frac{16K_1}{4K_1 K_m} = \frac{15}{K_m} = 0.145$$

可以推出， $K_m = 103.45$

当输入为 $\theta_i = 10 * 1(t)$ 时, 由图 (b) 可知,

$$u_s(0) = 10K_1 = 0.35 \rightarrow K_1 = 0.035$$

当输入为 $\theta_i = 10 * 1(t)$ 时, 可得到系统的实际误差曲线. 根据超调量, 可知

$$\sigma\% = e^{\frac{-\pi\xi}{\sqrt{1-\xi^2}}} * 100\% = 17.14\%$$

根据开环传递函数, 可知

$$G(s) = \frac{K_1 K_a K_m}{is(T_m s + 1)} = \frac{14.483/T_m}{s(s + 1/T_m)} = \frac{\omega_n^2}{s(s + 2\xi\omega_n)} \rightarrow T_m = 0.072$$

4 任务 3: 利用 MATLAB 求出闭环传递函数

4.1 相关 MATLAB 函数介绍

4.1.1 tf 函数

在 MATLAB 中, *tf* 函数用于创建传递函数模型。它将传递函数的分子和分母以向量的形式传递给 MATLAB, 从而生成对应的传递函数对象。

tf(num, den): 创建传递函数, *num* 是分子系数, *den* 是分母系数, 传递函数的形式为:

$$G(s) = \frac{num(s)}{den(s)}$$

其中 *num(s)* 和 *den(s)* 是分子和分母的多项式。

例如, $G_4 = tf([K_m], [T_m, 1, 0])$; 通过 *tf* 函数创建一个电机传递函数 G_4 , 其分子为 K_m , 表示增益项, 分母为 $[T_m, 1, 0]$, 表示一个二阶系统的传递函数, 具体的形式为

$$G_4(s) = \frac{K_m}{T_m s^2 + s}$$

4.1.2 series 函数

series 函数用于将多个函数按顺序串联, 基本用法为 $sys = series(sys1, sys2)$, 形成新的传递函数 *sys*, 总传递函数为 $G_{total}(s) = G_1(s)G_2(s)$

在本实验中, $sys1 = series(K_1, K_a)$ 将增益 K_1 和 K_a 串联, 得到系统的第一个串联传递函数 *sys1*。 $sys2 = series(sys1, i)$ 将 *sys1* 和 *i* 串联, 得到 *sys2*。这里 $i = 2/5$ 。 $sys3 = series(sys2, G_4)$ 将 *sys2* 和 G_4 串联, 得到系统的传递函数 *sys3*

4.1.3 feedback 函数

feedback 函数用于在控制系统中加入负反馈。其基本用法是

$$sys_{out} = feedback(sys_{in}, feedback_{gain})$$

其中 *sys_{in}* 是输入的传递函数。 *feedback_{gain}* 是反馈增益。常见的用法是设置为 1, 表示单位反馈。

如果反馈增益为 1, 则反馈系统的传递函数为:

$$G_{feedback}(s) = \frac{G_{in}(s)}{1 + G_{in}(s)}$$

在本例中, $sys = feedback(sys3, 1)$; 将 $sys3$ 和单位反馈结合, 得到闭环系统 sys 。负反馈会影响系统的稳定性和响应速度, 通常用于控制系统中以改善系统的性能。

4.2 MATLAB 代码与实验结果

4.2.1 MATLAB 代码

```
1  clc; clear; close all;
2  % 定义增益 K_1 (标量增益)
3  K_1 = tf(0.035);
4  % 使用 tf 函数定义传递函数 K_1 = 0.035, 表示系统的一部分增益
5  % 定义增益 K_a
6  K_a = tf(10);
7  % 定义另一个增益传递函数 K_a = 10
8  % 定义比例增益 i
9  i = tf(2/5);
10 % 定义比例增益 i = 2/5, 表示系统中某一部分的增益
11 % 定义电机传递函数 G4
12 K_m = 103.45;
13 T_m = 0.072;
14 G4 = tf([K_m], [T_m, 1, 0]);
15 % 创建电机的传递函数 G4, K_m 为电机增益, T_m 为时间常数, 分母 [T_m, 1, 0] 是二阶
    系统的系数。
16 % 将传递函数串联起来
17 sys1 = series(K_1, K_a);
18 % 使用 series 函数将 K_1 和 K_a 串联起来, 得到传递函数 sys1
19 sys2 = series(sys1, i);
20 % 将 sys1 和 i 串联起来, 得到新的传递函数 sys2
21 sys3 = series(sys2, G4);
22 % 将 sys2 和 G4 串联起来, 得到传递函数 sys3
23 % 构建闭环系统, 添加负反馈
24 sys = feedback(sys3, 1)
25 % 使用 feedback 函数实现负反馈, 1 表示单位反馈, 返回闭环传递函数 sys
```

4.2.2 实验结果

运行 MATLAB 代码, 可得系统的传递函数为

$$sys = \frac{14.48}{0.072s^2 + s + 14.48}$$

运行结果如图所示:

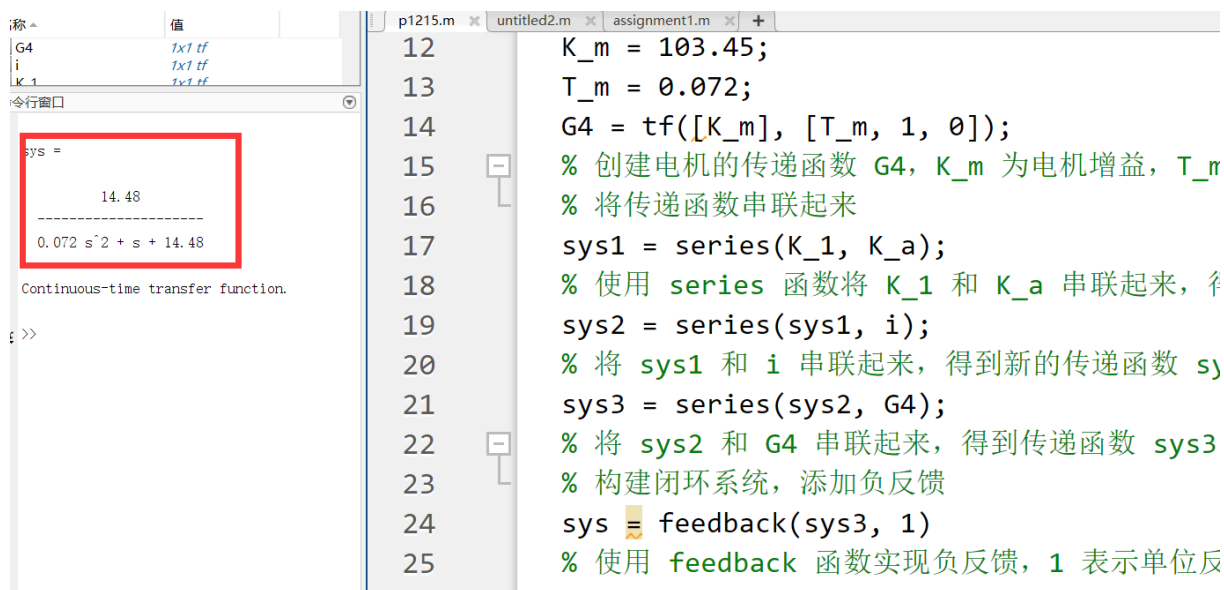


图 3: MATLAB 求解传递函数

5 任务 4: 绘制系统在不同输入下的响应曲线

5.1 输入轴为幅度为 10 的阶跃响应

5.1.1 step 函数

step 用于对传递函数系统进行阶跃响应分析, 可以计算并绘制阶跃响应图, *step* 函数的形式为

$$\text{step}(\text{sys}, t)$$

其中 *sys* 是传递函数或者零极点增益 (ZPK), *t* 为可选的时间向量, 默认的阶跃幅度为 1, 当输入轴为 $\theta_i(t) = 10 * 1(t)$ 时, 可以调整为 $\text{step}(10 * \text{sys})$, 即可实现输出 $10 * 1(t)$ 响应的功能。

5.1.2 MATLAB 代码

```
1 step(10*sys, 1.25, 'r')
2 legend('系统阶跃响应', 'Location', 'Best', 'FontName', 'SimHei');
3 % 设置坐标轴标签
4 xlabel('时间 t (秒)', 'FontSize', 12, 'FontWeight', 'bold', 'FontName', 'SimHei');
5 % 设置 X 轴标签
6 ylabel('输出响应', 'FontSize', 12, 'FontWeight', 'bold', 'FontName', 'SimHei');
7 % 设置 Y 轴标签
8 % 设置标题
9 title('系统阶跃响应 (幅度为10)', 'FontSize', 14, 'FontWeight', 'bold', 'FontName', 'SimHei');
```

```

10 grid on;
11 axis tight;
12 % 调整坐标轴刻度和字体大小
13 set(gca, 'FontSize', 12, 'FontWeight', 'bold', 'FontName', 'SimHei');
14 % 设置坐标轴刻度字体
15 % 导出为pdf
16 exportgraphics(gcf, 'G:\LateX\Template_for_DSP\Template_for_DSP\pic\result4.pdf',
    'ContentType', 'vector', 'BackgroundColor', 'none');

```

5.1.3 响应曲线

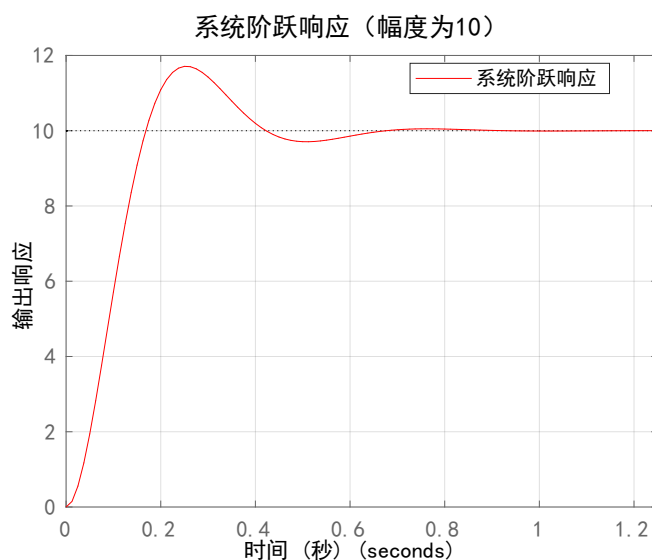


图 4: 幅度为 10 的单位阶跃响应

5.2 输入为幅值为 60 的速度函数

5.2.1 lsim 函数

lsim 函数用于模拟和绘制线性动态系统对指定输入信号的响应, 语法为

$$lsim(sys, u, t)$$

sys 表示系统的动态模型, 可以由传递函数获知零极点增益模型, u 表示输入信号, 在本例中时增益为 60 的速度响应, t 表示时间向量, 定义输入信号之间的间隔, $lsim$ 函数会计算系统对指定输入信号 u 的响应, 并绘制响应曲线

5.2.2 MATLAB 代码


```

1 figure;
2 u = [0:0.001:1];
3 t = 60.*u;
4 % 绘制系统响应
5 lsim(sys, t, u, 'r');
6 % 添加标题、坐标轴标签
7 title('幅值为60的速度响应', 'FontSize', 14, 'FontWeight', 'bold', 'FontName', 'SimHei');
8 xlabel('时间 (秒)', 'FontSize', 12, 'FontName', 'SimHei');
9 ylabel('输出响应', 'FontSize', 12, 'FontName', 'SimHei');
10 % 设置网格
11 grid on;
12 % 添加图例
13 legend('幅值为60的速度响应', 'FontSize', 10, 'Location', 'best', 'FontName', 'SimHei');
14 % 美化图形
15 set(gca, 'FontSize', 12); % 设置坐标轴字体大小
16 exportgraphics(gcf, 'G:\LateX\Template_for_DSP\Template_for_DSP\pic\result5.pdf',
    'ContentType', 'vector', 'BackgroundColor', 'none');

```

5.2.3 响应曲线

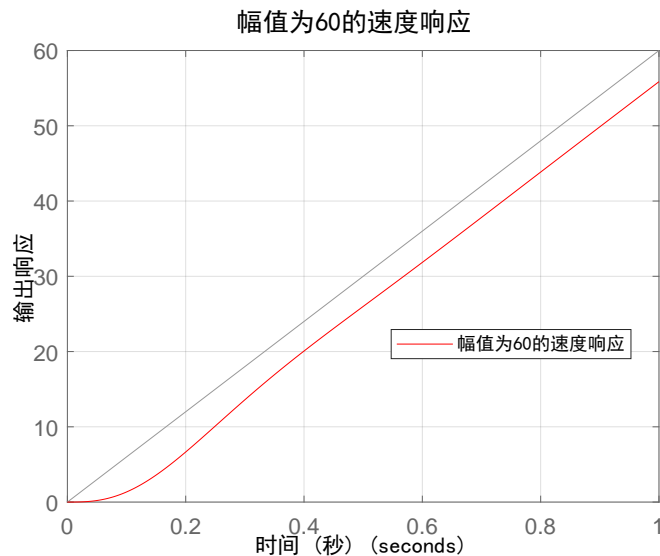


图 5: 幅度为 60 的单位速度响应

6 任务 5: 分析系统的性能

6.1 稳定性

由之前的分析可得, 系统的传递函数为

$$sys = \frac{14.48}{0.072s^2 + s + 14.48}$$

二阶系统特征方程同号且不缺项, 可得出结论: 系统是稳定的.

6.2 准确性

6.2.1 输入为阶跃信号时

当输入为阶跃响应时, 输入的幂次为 0, 系统的型别为 1, 由自动控制原理的知识可知, 当型别大于幂次时, 系统是无差的.

由于闭环传递函数的极点均在 s 平面左侧, 也可以通过终值定理判断. 阶跃信号的拉氏反变换

$$\mathcal{L}\{10u(t)\} = \frac{10}{s}$$

由闭环传递函数可以求得开环传递函数为

$$G(s) = \frac{14.48}{s(0.072s + 1)}$$

由终值定理可知,

$$e_s s = \lim_{s \rightarrow 0} s * \frac{10}{s} * \frac{1}{1 + G(s)} = \lim_{s \rightarrow 0} \frac{s(0.072s + 1)}{0.072s^2 + s + 14.48} = 0$$

6.2.2 当输入为速度信号时

速度信号的幂次为 1, 系统的型别为 1, 当型别等于幂次是, 误差为有限值, 速度信号的拉氏变换为:

$$\mathcal{L}\{60t\} = \frac{60}{s^2}$$

速度函数的幅值为 60, 系统的静态误差系数

$$K_v = \lim_{s \rightarrow 0} s \frac{14.48}{s(0.072s + 1)} = 14.48$$

可得稳态误差

$$e_s s = \frac{R}{K_v} = \frac{60}{14.48} = 4.14$$

也可以用终值定理:

$$e_s s = \lim_{s \rightarrow 0} s \frac{60}{s^2} \frac{1}{1 + G(s)} = \lim_{s \rightarrow 0} \frac{60(0.072s + 1)}{s(0.072s^2 + s + 14.48)} = \frac{60}{14.48} = 4.14$$

通过观察输出的响应曲线, 可验证计算的稳态误差的正确性.

6.3 快速性

系统的传递函数为

$$\Phi(s) = \frac{201.1}{s^2 + 13.89s + 201.1}$$

由传递函数可以确定系统的无阻尼振荡频率为 $\omega_n = \sqrt{201.1} = 14.18$, 系统的阻尼比为 $\xi = \frac{13.89}{2\omega_n} = 0.49$, 这与第二部分求参数时通过超调量算出来的 ξ 相对应.

由于 $\cos\beta = \xi$, 可得 $\beta = 1.06$, 可求上升时间

$$t_r = \frac{\pi - \beta}{\omega_n(1 - \xi^2)} = \frac{2.08}{12.36} = 0.17s$$

峰值时间

$$t_p = \frac{\pi}{\omega_n(1 - \xi^2)} = \frac{3.14}{12.36} = 0.25s$$

取允许误差带 $\Delta = 0.2\%$, 可计算调节时间为

$$t_s = \frac{4}{\xi\omega_n} = \frac{4}{6.85} = 0.58s$$

通过观察输出的响应曲线可验证上升时间, 峰值时间, 调节时间的正确性.

7 实验总结

通过这次实验, 我不仅巩固了控制系统分析的基础知识, 还学会了如何将理论应用到实际问题中。在绘制系统结构图的过程中, 我更好地理解系统的各个组成部分及其相互关系。在求解传递函数时, 我加深了对系统动态特性的认识, 并通过 MATLAB 工具进行系统建模和响应仿真, 提升了自己的编程能力和解决问题的思维方式。此外, 通过对系统稳定性、准确性和快速性的分析, 我学会了如何评估控制系统的性能, 并在实际设计中进行优化。这次实验不仅加深了我对控制理论的理解, 也为今后的学习和工作提供了宝贵的实践经验。