实验报告

课程名称： 自动控制原理专题实验

姓名： 周湛昊

学院： 电信学部

班级： 自动化 2305

学号： 2233712088

2025 年 11 月 9 日

目录

实验一 线性系统时域特性分析

一、实验目的

1. 研究二阶系统在阶跃信号输入作用下的输出响应，分析其动态性能指标。

2. 研究二阶系统的特征参量阻尼比ζ和自然频率ω_n对阶跃响应动态性能的影响。

二、实验内容与要求

1. 根据二阶系统的标准传递函数，自定义参数ζ和ω_n

（1）令ω_n不变，绘制 4 种阻尼状态的 5 条阶跃响应曲线，分别是无阻尼、欠阻尼（2 条）、临界阻尼和过阻尼状态，要求在一个坐标上绘制。

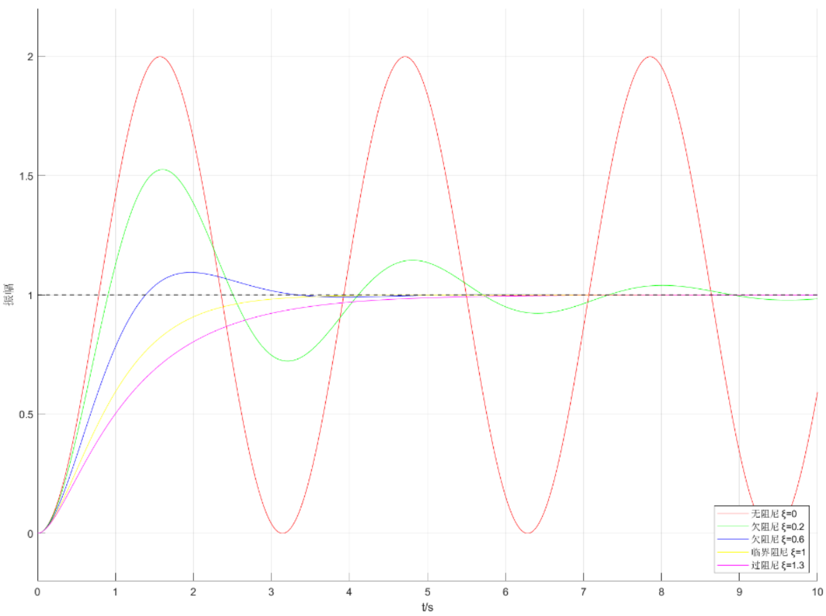
（2）令ζ不变，在欠阻尼状态下，ω_n分别取$\frac{1}{5}\omega_n$、ω_n、$5\omega_n$，画出 3 条阶跃响应曲线；在过阻尼状态下，ω_n分别取$\frac{1}{10}\omega_n$、ω_n、$10\omega_n$，画出 3 条阶跃响应曲线。要求在一个坐标上绘制。

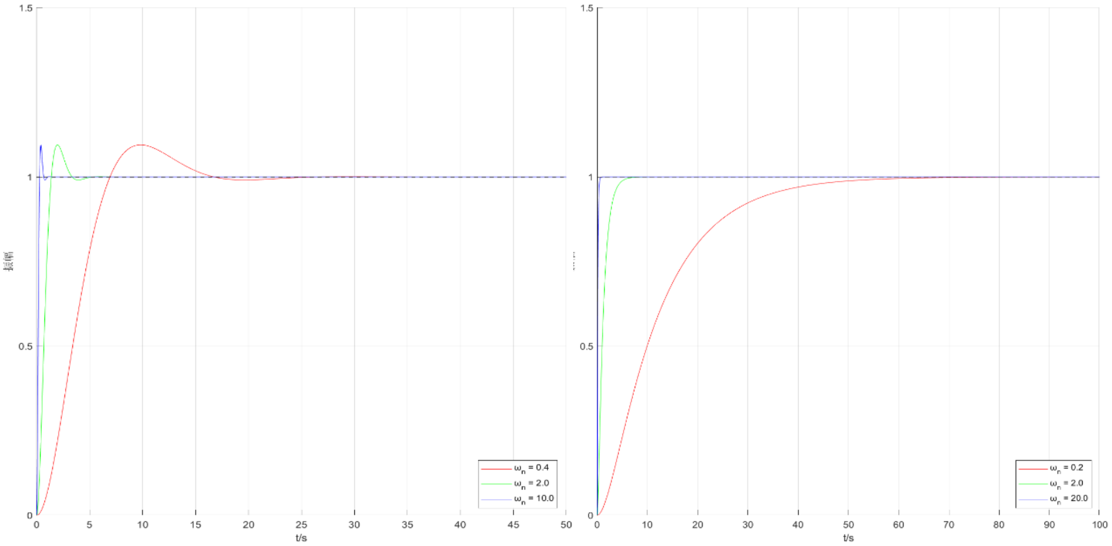
（3）编程获取以上参数ζ和ω_n对应的系统性能指标，超调量、稳态时间（Δ2 ）、上升时间、峰值时间、稳态误差（Δ2 ）。并以表格的形式进行对比，分析写出标准二阶系统中阻尼比和自然频率参数变化对系统阶跃响应曲线的影响。

三、代码部分

见附录

四、运行结果






```
>> exp1
超调量 σ%:    99.9999    52.6610     9.4778    -0.0000    -0.0000
峰值时间 tp:    1.5700     1.6000     1.9600    18.5500    20.0000
上升时间 tr:    0.5100     0.6000     0.9300     1.6800     2.4300
调节时间 ts:   20.0000     9.8000     2.9700     2.9100     4.4300
稳态误差 ess:      0      0      0      0      0
超调量 σ%:     9.4780     9.4778     9.4745
峰值时间 tp:    9.8200     1.9600     0.3900
上升时间 tr:    4.6300     0.9300     0.1900
调节时间 ts:   14.8500     2.9700     0.5900
稳态误差 ess:      0      0      0
超调量 σ%:    -0.0108    -0.0000    -0.0000
峰值时间 tp:  100.0000    35.0900     3.7100
上升时间 tr:   24.3800     2.4300     0.2500
调节时间 ts:   44.3200     4.4300     0.4400
稳态误差 ess:      0      0      0
```

五、思考题

1. 二阶系统的显著特点是什么？为什么控制系统把二阶系统作为主要分析对象？

答：（1）二阶系统的行为可以通过ζ和ωn两个参数精确、完整地描述和预测，其响应从振荡到平滑，性能指标可计算，方便分析和设计。

（2）在数学上方便处理计算，同时又广泛存在，并且其特性能够表现动态系统的行为

2. 二阶系统的动态特性分析为什么使用阶跃信号作为输入。

答：能直接定义和测量上升时间、超调量、调节时间等所有关键动态性能标。

实验二 线性系统稳定性分析

一、实验目的

1. 掌握使用特征根、零极点图和劳斯判据判别系统稳定性的方法。

2. 研究系统开环增益、零极点的变化对特征根和稳定性的影响。

二、实验内容与要求

1. 根据自动控制理论的内容，自定义开环传递函数$G(s)=\frac{K}{s^4+a_3s^3+a_2s^2+a_1s+a_0}$的参数$K$，$a_3$，$a_2$，$a_1$，$a_0$，要求：

利用劳斯判据判定系统稳定性，求出K的稳定范围，并画出稳定、不稳定、临界稳定的 3 种K值的阶跃响应曲线；

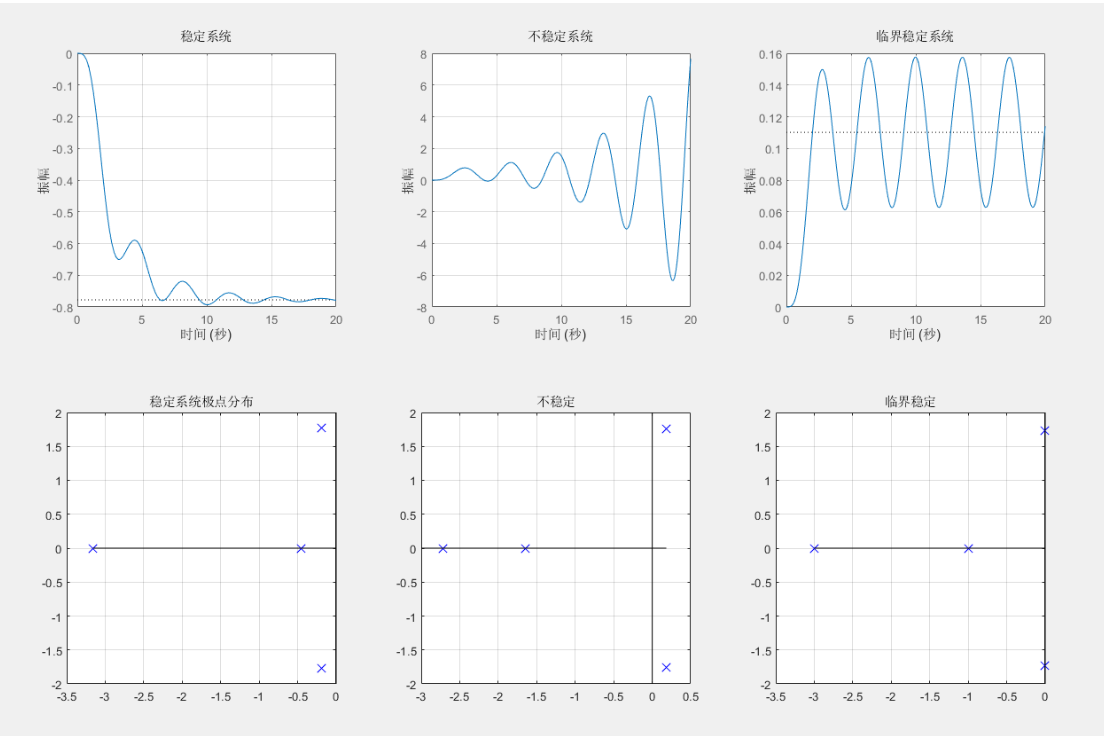
2. 并求出稳定、不稳定、临界稳定 3 中情况下系统特征根，验证特征根与稳定性关系；

3. 绘制出稳定、不稳定、临界稳定 3 种情况下对应的零极点图。

三、代码部分

见附录

四、运行结果



五、思考题

1. 特征根与开环零极点、闭环零极点的关系。

答：特征根是闭环传递函数分母的根，也就是闭环极点。开环传递函数中包含开环零极点，令开环函数等于-1 的根就是特征根。

2. 绝对稳定性和相对稳定性指的是什么？

答：绝对稳定性是指所有闭环极点是否在 S 左半平面，而相对稳定性指的是系统稳定程度的大小，取决于阻尼比、相位裕度和增益裕度的大小。

六、实验总结

通过本次实验，使用 MATLAB 进行自动控制二阶系统的动态响应分析和线性系统稳定性的分析，学会了用 MATLAB 辅助原理学习，同时实际应用中 MATLAB 在自动控制分析方面具有重要的实践应用场景，这对于提升理论与实践的结合具有重要作用。

七、附录（程序代码）

1. 实验一

```
% 1(1)
w_n = 2;
ei_values = [0, 0.2, 0.6, 1, 1.3];
ei_labels = {'无阻尼 ξ=0', '欠阻尼 ξ=0.2', '欠阻尼 ξ=0.6', '临界阻尼 ξ=1', '过阻尼 ξ=1.3'};
colors = ['r', 'g', 'b', 'y', 'm'];
line_styles = {'-', '-', '-', '-', '-'};
figure('Position', [100, 100, 800, 600]);
t = 0:0.01:10;
hold on;
for i = 1:length(ei_values)
    ei = ei_values(i);
    if ei == 0
        num = w_n^2;
        den = [1, 0, w_n^2];
    else
        num = w_n^2;
        den = [1, 2*ei*w_n, w_n^2];
    end

    sys = tf(num, den);
    [y, t] = step(sys, t);
    plot(t, y, 'Color', colors(i), 'LineStyle', line_styles{i}, 'DisplayName', ei_labels{i});
end
grid on;
xlabel('t/s');
ylabel('振幅');
legend('show', 'Location', 'southeast');
yline(1, 'k--', 'LineWidth', 1, 'HandleVisibility', 'off');
xlim([0, 10]);
ylim([-0.2, 2.2]);
```

```matlab
% 1(2)
ei_1 = 0.6;   % 欠阻尼
ei_2 = 1.3;    % 过阻尼
t = 0:0.01:100;
colors = ['r', 'g', 'b'];
line_styles = {'-', '-', '-'};
figure('Position', [100, 100, 1200, 500]);

subplot(1, 2, 1);
w_n_values1 = [1/5, 1, 5];
base_w_n = 2;
hold on;
for i = 1:length(w_n_values1)
    w_n = base_w_n * w_n_values1(i);
    num = w_n^2;
    den = [1, 2*ei_1*w_n, w_n^2];
    sys = tf(num, den);
    [y, t] = step(sys, t);
    plot(t, y, 'Color', colors(i), 'LineStyle', line_styles{i}, 'DisplayName', sprintf('ω_n = %.1f', w_n));
end
grid on;
xlabel('t/s');
ylabel('振幅');
legend('show', 'Location', 'southeast');
yline(1, 'k--', 'LineWidth', 1, 'HandleVisibility', 'off');
xlim([0, 50]);
ylim([0, 1.5]);

subplot(1, 2, 2);
w_n_values2 = [1/10, 1, 10];
hold on;
for i = 1:length(w_n_values2)
    w_n = base_w_n * w_n_values2(i);
    num = w_n^2;
    den = [1, 2*ei_2*w_n, w_n^2];
    sys = tf(num, den);
    [y, t] = step(sys, t);
    plot(t, y, 'Color', colors(i), 'LineStyle', line_styles{i}, 'DisplayName', sprintf('ω_n = %.1f', w_n));
end
grid on;
xlabel('t/s');
ylabel('振幅');
legend('show', 'Location', 'southeast');
yline(1, 'k--', 'HandleVisibility', 'off');
```

```matlab
% 1(3)
ei_values = [0, 0.2, 0.6, 1, 1.3];
sigma_percent = [];
tp = []; tr = []; ts = []; ess = [];
for i = 1:length(ei_values)
    ei = ei_values(i); wn = 2; num = wn^2; den = [1, 2*ei*wn, wn^2];
    sys = tf(num, den);
    t = 0:0.01:20;
    [y, t] = step(sys, t); yss = dcgain(sys);
    upper_bound = yss * 1.02; lower_bound = yss * 0.98;
    [ymax, idx_peak] = max(y);
    sigma_percent = [sigma_percent, (ymax - yss) / yss * 100];
    tp = [tp, t(idx_peak)];
    idx_10 = find(y >= 0.1*yss, 1); idx_90 = find(y >= 0.9*yss, 1);
    tr = [tr, t(idx_90) - t(idx_10)];
    idx_settle = find(y < lower_bound | y > upper_bound, 1, 'last');
    if isempty(idx_settle)
        ts = [ts, 0];
    else
        ts = [ts, t(idx_settle)];
    end
    ess = [ess, abs(1 - yss)];
end
fprintf('超调量 σ%%:'); disp(sigma_percent);
fprintf('峰值时间 tp:'); disp(tp);
fprintf('上升时间 tr:'); disp(tr);
fprintf('调节时间 ts:'); disp(ts);
fprintf('稳态误差 ess:'); disp(ess);


wn_values = [0.4, 2, 10];
sigma_percent = [];
tp = []; tr = []; ts = []; ess = [];
for i = 1:length(wn_values)
    ei = 0.6; wn = wn_values(i); num = wn^2; den = [1, 2*ei*wn, wn^2];
    sys = tf(num, den);
    t = 0:0.01:20;
    [y, t] = step(sys, t); yss = dcgain(sys);
    [ymax, idx_peak] = max(y);
    sigma_percent = [sigma_percent, (ymax - yss) / yss * 100];
    tp = [tp, t(idx_peak)];
    idx_10 = find(y >= 0.1*yss, 1); idx_90 = find(y >= 0.9*yss, 1);
    tr = [tr, t(idx_90) - t(idx_10)];
    idx_settle = find(y < lower_bound | y > upper_bound, 1, 'last');
    if isempty(idx_settle)
        ts = [ts, 0];
    else
        ts = [ts, t(idx_settle)];
    end
    ess = [ess, abs(1 - yss)];
end
fprintf('超调量 σ%%:'); disp(sigma_percent);
fprintf('峰值时间 tp:'); disp(tp);
fprintf('上升时间 tr:'); disp(tr);
fprintf('调节时间 ts:'); disp(ts);
fprintf('稳态误差 ess:'); disp(ess);
```

```
wn_values = [0.2, 2, 20];
sigma_percent = [];
tp = []; tr = []; ts = []; ess = [];
for i = 1:length(wn_values)
    ei = 1.3; wn = wn_values(i); num = wn^2; den = [1, 2*ei*wn, wn^2];
    sys = tf(num, den);
    t = 0:0.01:100;
    [y, t] = step(sys, t); yss = dcgain(sys);
    [ymax, idx_peak] = max(y);
    sigma_percent = [sigma_percent, (ymax - yss) / yss * 100];
    tp = [tp, t(idx_peak)];
    idx_10 = find(y >= 0.1*yss, 1); idx_90 = find(y >= 0.9*yss, 1);
    tr = [tr, t(idx_90) - t(idx_10)];
    idx_settle = find(y < lower_bound | y > upper_bound, 1, 'last');
    if isempty(idx_settle)
        ts = [ts, 0];
    else
        ts = [ts, t(idx_settle)];
    end
    ess = [ess, abs(1 - yss)];
end
fprintf('超调量 σ%%:'); disp(sigma_percent);
fprintf('峰值时间 tp:'); disp(tp);
fprintf('上升时间 tr:'); disp(tr);
fprintf('调节时间 ts:'); disp(ts);
fprintf('稳态误差 ess:'); disp(ess);
```

2. 实验二

```
a0 = 8;
a1 = 12;
a2 = 6;
a3 = 4;

syms K_sym s;

% s4: 1,  a2, a0+K
% s3: a3, a1, 0
% s2: b1, b2, 0
% s1: c1, 0,  0
% s0: d1, 0,  0

b1 = (a3*a2-1*a1)/a3;
b2 = (a3*(a0+K_sym))/a3;
c1 = (b1*a1-a3*b2)/b1;
d1 = b2;

K_values = -10:0.01:10;
stable_K = [];
```

```matlab
for K_test = K_values
    b1_val = double(subs(b1, K_sym, K_test));
    c1_val = double(subs(c1, K_sym, K_test));
    d1_val = double(subs(d1, K_sym, K_test));
    if a3 > 0 && b1_val > 0 && c1_val > 0 && d1_val > 0
        stable_K = [stable_K, K_test];
    end
end
if ~isempty(stable_K)
    K_min = min(stable_K);
    K_max = max(stable_K);
    fprintf('稳定范围: %.2f<K<%.2f\n', K_min, K_max);
    K_stable = (K_max+K_min)/2;      % 稳定系统
    K_unstable = K_max + 5;              % 不稳定系统
    K_critical = K_max;                  % 临界稳定系统
else
    fprintf('未找到稳定的K值范围\n');
    K_stable = 1;
    K_unstable = 10;
    K_critical = 5;
end
fprintf('稳定:%f\n',K_stable);
fprintf('不稳定:%f\n',K_unstable);
fprintf('临界:%f\n',K_critical);
figure('Position', [100, 100, 1200, 800]);

subplot(2,3,1);
G_stable = tf(K_stable, [1, a3, a2, a1, a0+K_stable]);
step(G_stable, 20);
title('稳定系统');
grid on;

subplot(2,3,2);
G_unstable = tf(K_unstable, [1, a3, a2, a1, a0+K_unstable]);
step(G_unstable, 20);
title('不稳定系统');
grid on;

subplot(2,3,3);
G_critical = tf(K_critical, [1, a3, a2, a1, a0+K_critical]);
step(G_critical, 20);
title('临界稳定系统');
grid on;
```

```matlab
% 绘制极点分布
subplot(2,3,4);
p_stable = pole(G_stable);
plot(real(p_stable), imag(p_stable), 'bx', 'MarkerSize', 10);
hold on;
plot([0,0], ylim, 'k-');
plot(xlim, [0,0], 'k-');
title('稳定系统极点分布');
grid on;

subplot(2,3,5);
p_unstable = pole(G_unstable);
plot(real(p_unstable), imag(p_unstable), 'bx', 'MarkerSize', 10);
hold on;
plot([0,0], ylim, 'k-');
plot(xlim, [0,0], 'k-');
title('不稳定');
grid on;

subplot(2,3,6);
p_critical = pole(G_critical);
plot(real(p_critical), imag(p_critical), 'bx', 'MarkerSize', 10);
hold on;
plot([0,0], ylim, 'k-');
plot(xlim, [0,0], 'k-');
title('临界稳定');
grid on;
```