

GitHub 账号: Tonkovski**实验摘要:**

- 熟练掌握 MATLAB 基础语法，进一步使用条件和循环控制解决问题。
- 熟悉信号图像绘制，通过添加标注增强 MATLAB 图像可读性。
- 利用各内置函数及工具箱完成常见的信号相关运算。
- 综合运用 MATLAB 仿真，了解傅里叶级数逼近原函数的原理。

实验题目

1. 利用MATLAB求下列函数的卷积，并绘制出图形

$$(1) f_1(t) = \varepsilon(t) - \varepsilon(t-1), \quad f_2(t) = 2t[\varepsilon(t) - \varepsilon(t-1)]$$

$$(2) f_1(t) = \cos(30t)g_3(t), \quad f_2(t) = \varepsilon(t) - \varepsilon(t-4)$$

参考函数: conv()

2. 某系统满足的微分方程为

$$y''(t) + 4y'(t) + 3y(t) = 2f'(t) + f(t)$$

(1) 利用MATLAB求系统的单位冲击响应，并绘出图形

(2) 利用MATLAB求系统的单位阶跃响应，并绘出图形

(3) 利用MATLAB求系统对信号 $f(t) = 4\sin(2\pi t)\varepsilon(t)$ 的响应，并绘出图形

参考函数: tf(), impulse(), step(), lsim(), conv()

3. 利用MATLAB产生高斯白噪声，绘出图形，并求其自相关函数，绘出图形。

参考函数: randn(), wgn(), xcorr(), autocorr()

4. 预习关于傅里叶级数的内容，用MATLAB或者Python进行以下实验，回答问题并给出实验过程中产生的结果图。

(1) 信号 $f(t)$ 的傅里叶级数为 $\sum_1^{\infty} \frac{\sin nt}{n}$ ，代入数字去逼近或者用解析法分析，估计 $f(t)$ 的形式。

(2) 写出你估计出的 $f(t)$ 的傅里叶级数，与上式对比，说明它的谐波和正余弦分量的情况。

(3) 取 $N = 50, 100, 200, \dots$ 画出 $f_N(t) = \sum_{n=1}^N \frac{\sin nt}{n}$ ，当 $N \rightarrow \infty$ 时，判断这个部分和与 $f(t)$ 的区别。

(4) 同样，取 $N = 50, 100, 200, \dots$ 画出 $F_N(t) = \frac{f_1(t) + f_2(t) + f_3(t) + \dots + f_N(t)}{N}$ ，和上面的图对比，分析他们之间的不同。

实验内容

一 实验基本原理及步骤

利用 **MATLAB** 的内置函数及工具箱进行仿真并绘制图像。

各题目代码如下：

1.

```
%% Signal & System Experiment - Class 2 - Task 1
pace = 0.01; % Determines accuracy, reduce to increase sampling rate.
figure('name','Task 1', 'numbertitle','off');

% Fig 2.(1)
range_t = [-1 2]; % Range of t.
t0 = range_t(1):pace:range_t(2);
t = 2*range_t(1):pace:2*range_t(2);
f1 = heaviside(t0) - heaviside(t0-1);
f2 = 2*t0.*(heaviside(t0) - heaviside(t0-1));
f = conv(f1,f2)*pace;
subplot(211);plot(t,f); title('Fig 1.(1)'); xlabel('t'); ylabel('f1*f2');

% Fig 1.(2)
range_t = [-3 5]; % Range of t.
t0 = range_t(1):pace:range_t(2);
t = 2*range_t(1):pace:2*range_t(2);
f1 = cos(30*t0).*rectpuls(t0,5);
f2 = heaviside(t0) - heaviside(t0 - 4);
f = conv(f1,f2)*pace;
subplot(212);plot(t,f); title('Fig 1.(2)'); xlabel('t'); ylabel('f1*f2');
```

2.

```
%% Signal & System Experiment - Class 2 - Task 2
pace = 0.01; % Determines accuracy, reduce to increase sampling rate.
figure('name','Task 2', 'numbertitle','off');
sys = tf([2, 1],[1, 4, 3]);

% Fig 2.(1)
t = 0:pace:10;
h = impulse(sys,t);
subplot(311); plot(t,h); title('Fig 2.(1) Impulse Response'); xlabel('t');
ylabel('h');

% Fig 2.(2)
t = 0:pace:10;
g = step(sys,t);
subplot(312); plot(t,g); title('Fig 2.(2) Step Response'); xlabel('t');
ylabel('g');

% Fig 2.(3)
t = 0:pace:10;
f = 4*sin(2*pi*t);
y = lsim(sys,f,t);
subplot(313); plot(t,y); title('Fig 2.(3) Function Zero State Response');
xlabel('t'); ylabel('g');
```

3.

```
%% Signal & System Experiment - Class 2 - Task 3
figure('name','Task 3', 'numbertitle','off');

% Fig 3.(1)
y=2*wgn(1,500,0); % Generates WGN.
subplot(211); plot(y); title('Fig 3.(1) WGN');

% Fig 3.(2)
n=length(y);
[acf, lags, bounds] = autocorr(y,n-1);
subplot(212); plot(lags,acf); title('Fig 3.(2) Autocorrelation Function by "autocorr"');
```

4.

```
%% Signal & System Experiment - Class 2 - Task 4
pace = 0.01; % Determines accuracy, reduce to increase sampling rate.
range_t = [-5 5]; % Can be changed manually;
t = range_t(1):pace:range_t(2);

% Section 4.(1)
m = 999; % Can be changed manually;
f = zeros(size(t));
for n = 1:m
    temp = sin(n*t)./n;
    f = f + temp;
end
figure('name','Task 4 Section 4.(1)', 'numbertitle','off');
plot(t,f); title('Digital Approximation'); xlabel('t'); ylabel('y
(Approximate)');

% Section 4.(2)
f = 1.5*sawtooth(-t);
figure('name','Task 4 Section 4.(2)', 'numbertitle','off');
plot(t,f); title('Hypothesis Testing'); xlabel('t'); ylabel('y (Hypothesis)');
ylim([-2 2]);

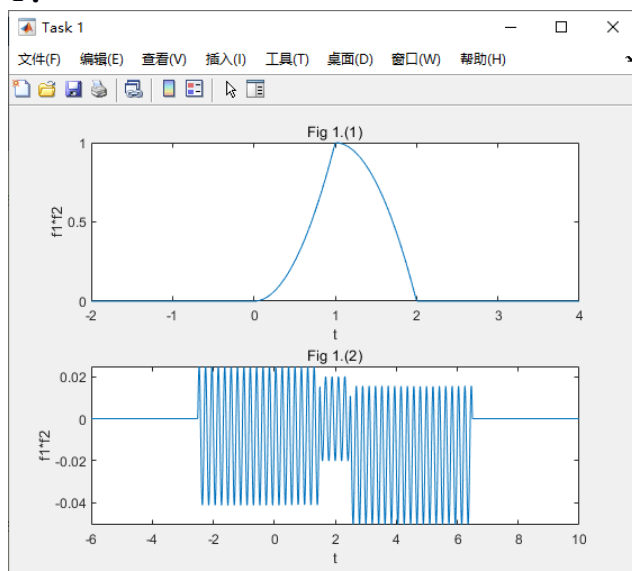
% Section 4.(3)
f = zeros(size(t));
figure('name','Task 4 Section 4.(3)', 'numbertitle','off'); hold on;
for n = 1:50
    temp = sin(n*t)./n;
    f = f + temp;
end
plot(t,f,'r');
for n = 51:100
    temp = sin(n*t)./n;
    f = f + temp;
end
plot(t,f,'g');
for n = 101:200
    temp = sin(n*t)./n;
    f = f + temp;
end
plot(t,f,'b');
legend('n = 50', 'n = 100', 'n = 200'); title('Comparison fn'); xlabel('t');
ylabel('y');

% Section 4.(4)
figure('name','Task 4 Section 4.(4)', 'numbertitle','off'); hold on;
Sum = zeros(size(t));
for m = 1:50
    f = zeros(size(t));
    for n = 1:m
        temp = sin(n*t)./n;
        f = f + temp;
    end
    Sum = Sum + f;
end
F = Sum/50;
plot(t,F,'r');
for m = 51:100
    f = zeros(size(t));
    for n = 1:m
        temp = sin(n*t)./n;
        f = f + temp;
    end
    Sum = Sum + f;
end
F = Sum/100;
plot(t,F,'g');
for m = 101:200
    f = zeros(size(t));
```

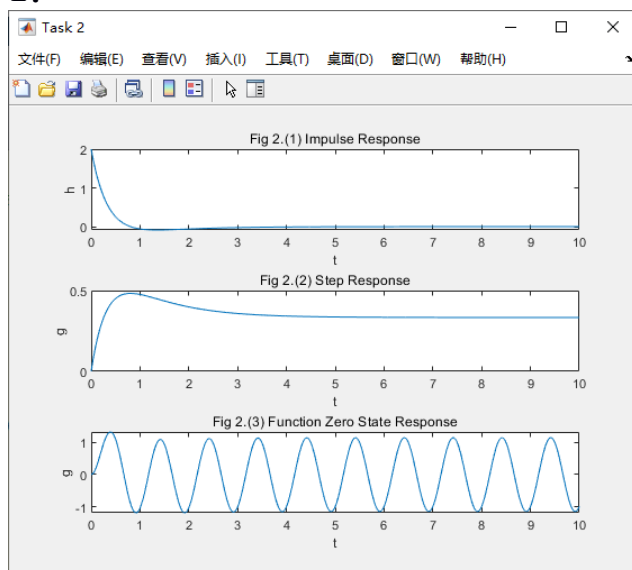
```
for n = 1:m
    temp = sin(n*t)./n;
    f = f + temp;
end
Sum = Sum + f;
end
F = Sum/200;
plot(t,F,'b');
legend('N = 50', 'N = 100', 'N = 200'); title('Comparison Fn'); xlabel('t');
ylabel('y');
```

二 实验结果

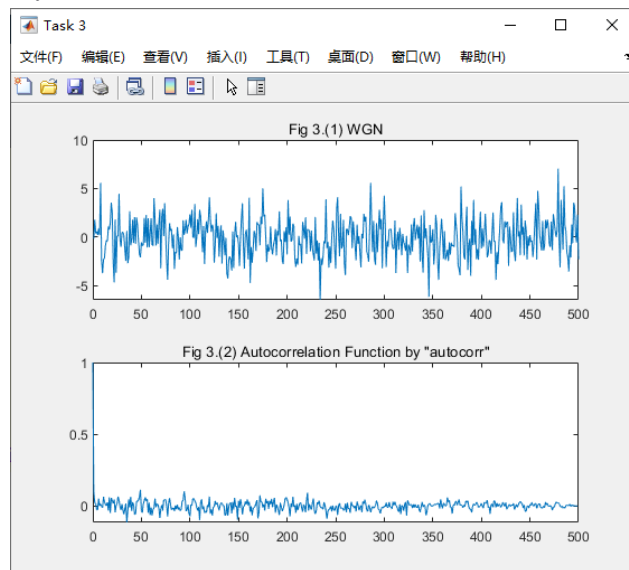
1.



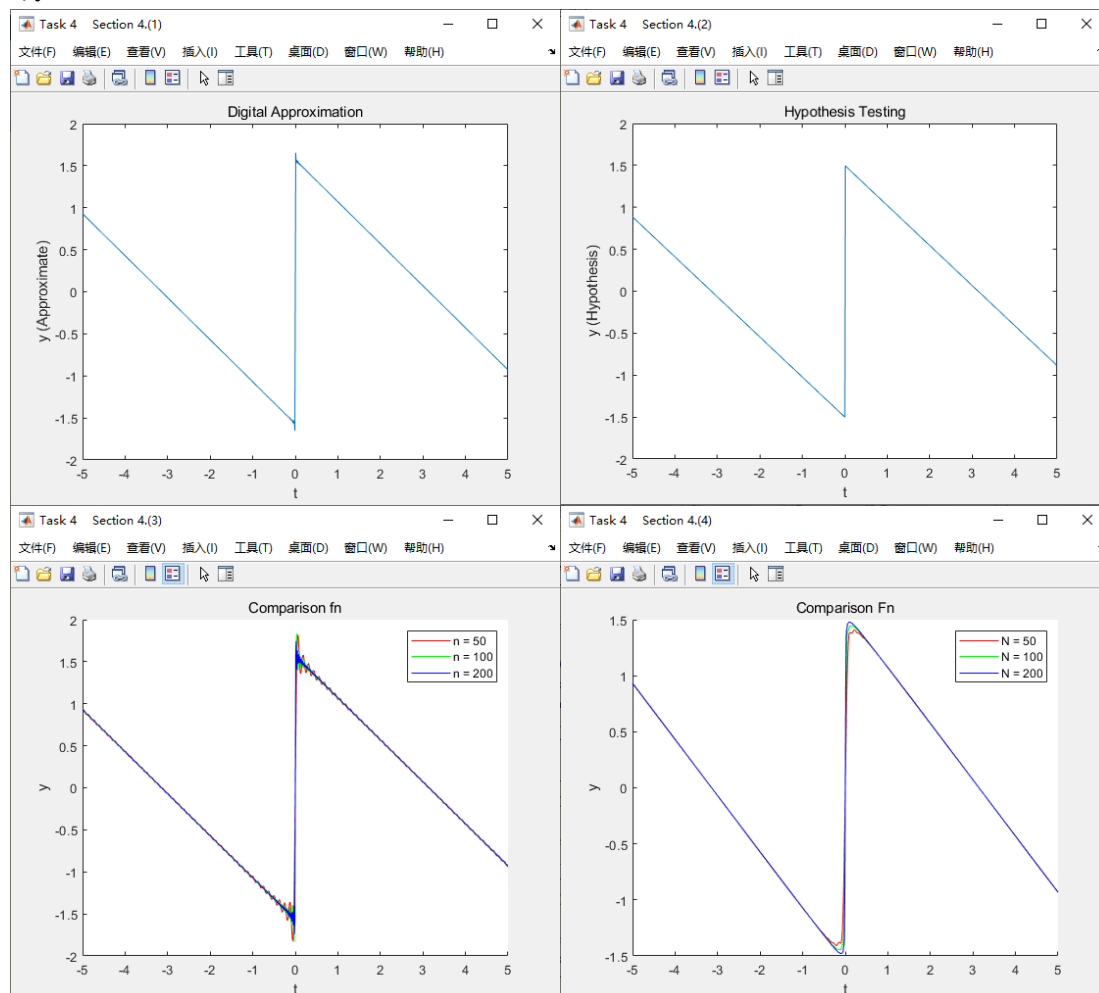
2.



3.



4.



三 实验结果的分析

符合实验预期；完成了卷积、求响应等基本运算；了解傅里叶展开的近似拟合原理并观察吉布斯现象。

实验总结

- 应加强交流，寻找最优实现方法。
- 要提前做准备，避免用到扩展组件时才在线下载。

参考文献

- **MATLAB Documentation**
- **CSDN 论坛**
- 《信号与线性系统分析》 吴大正 编