

Github 账号: [117503445](#)

实验摘要:

学习 傅里叶变换,傅里叶反变换,幅度谱,相位谱

学习 拉普拉斯变换,拉普拉斯反变换

学习 系统函数零极点求解,稳定性判断

学习 冲激响应 阶跃响应 零状态响应 的拉普拉斯求解方法

实验题目

1. 使用Matlab函数计算 $f(t) = e^{-2t}$ 的傅里叶变换, $F(\omega) = \frac{1}{1+\omega^2}$ 傅里叶反变换。

参考函数: `fourier()`, `ifourier()`

2. 计算 $f_1(t) = \frac{1}{2}e^{-2t}\varepsilon(t)$ 和 $f_2(t) = \frac{1}{2}e^{-2(t-1)}\varepsilon(t-1)$ 的傅里叶变换, 画出其幅度谱和相位谱, 并观察傅里叶变换的时移特性。

参考函数: `syms()`

3. 用部分分式展开法求 $H(s) = \frac{(s+1)(s+4)}{s(s+2)(s+3)}$ 的反变换。

参考函数: 因子形式转换多项式`conv()`, `residue()`

4. 画出系统函数 $H_1(s) = \frac{s+2}{s^3+s^2+2s+6}$ 和 $H_2(s) = \frac{s^2+1}{3s^3+5s^2+4s+6}$ 零、极点分布, 并判断系统的稳定性。

参考函数: `laplace()`, `roots()`, `pzmap()`, `sys()`

5. 用MATLAB计算拉普拉斯变换求解 $H(s) = \frac{s}{s^2+3s+2}$ 的冲激响应、阶跃响应, 以及激励 $f(t) = \cos(20t)\varepsilon(t)$ 产生的零状态响应, 给出运行结果并分析。

参考函数: `laplace()`, `ilaplace()`, `sys()`等

实验内容

1

```
syms t;  
f1 = exp(-2 * abs(t));  
F1 = fourier(f1)  
subplot(1, 2, 1);  
ezplot(F1);
```

```
grid on;
```

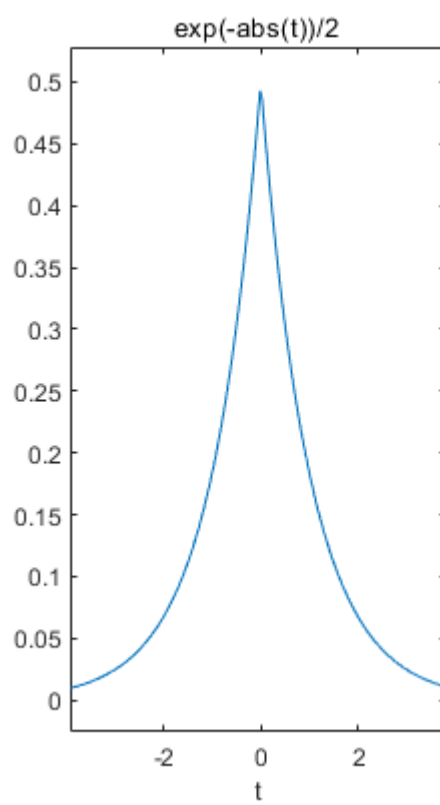
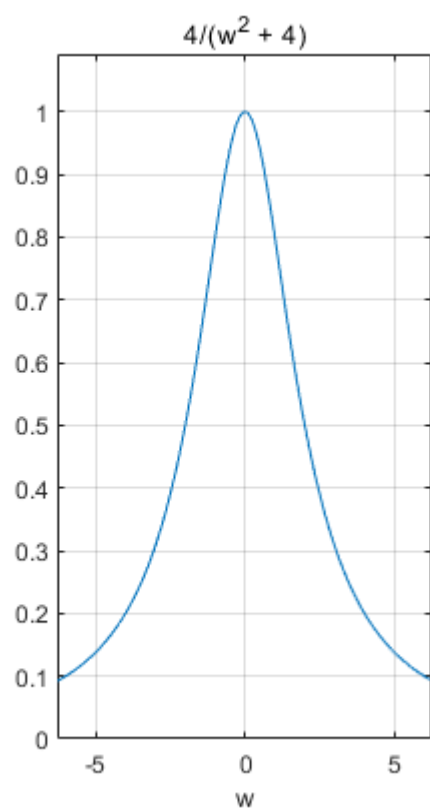
```
syms w;  
F2 = 1 / (1 + w^2);  
f2 = ifourier(F2, t)  
subplot(1, 2, 2);  
ezplot(f2);  
,
```

F1 =

$4/(w^2 + 4)$

f2 =

$\exp(-\text{abs}(t))/2$

**2**

```

syms t w;
f1 = (1/2) * exp(-2 * t) * heaviside(t);
F1 = fourier(f1)
f2 = (1/2) * exp(-2 * (t - 1)) * heaviside(t - 1);
F2 = fourier(f2)
subplot(2, 2, 1);
ezplot(f1);
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
title('f1(t)');
subplot(2, 2, 2);
ezplot(abs(F1), [-5 5]);
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
title('f1 幅度谱');
p1 = angle(F1);
%i1=imag(F1);r1=real(F1);p1=atan2(i1,r1); subplot(2,3,3); ezplot(p1);
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
xlim([-5 5]);
title('f1 相位谱');
subplot(2, 2, 3);
ezplot(f2);
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
title('f2(t)');
subplot(2, 2, 4);
ezplot(abs(F2));
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
xlim([-5 5]);
title('f2 幅度谱');
p2 = angle(F2); %i2=imag(F2);r2=real(F2);p2=atan2(i2,r2); subplot(2,3,6); ezplot(p2);
grid on;
set(gca, 'GridLineStyle', ':', 'GridColor', 'k', 'GridAlpha', 1);
xlim([-5 5]);
title('f2 相位谱');|

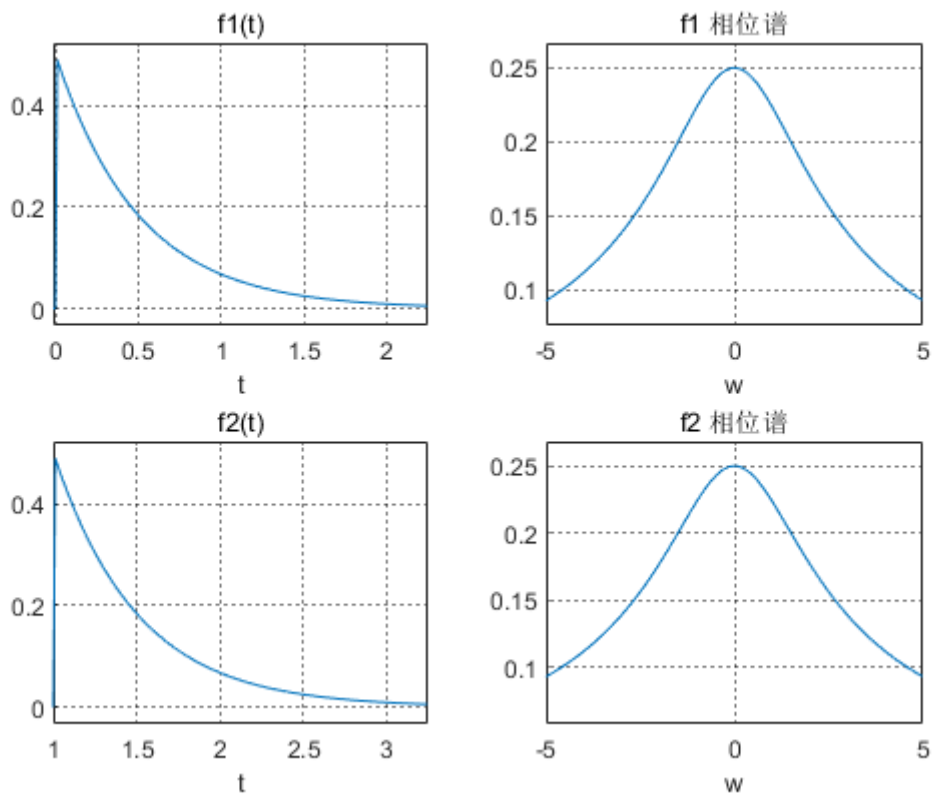
```

F1 =

$$1/(2*(2 + w*1i))$$

F2 =

$$\exp(-w*1i)/(2*(2 + w*1i))$$



3

```
syms s t;
b = conv([1, 1], [1, 4]);
a = conv(conv([1, 0], [1, 2]), [1, 3]);
[r, p, k] = residue(b, a);
h1 = ilaplace(r(1) / (s - p(1)), t);
h2 = ilaplace(r(2) / (s - p(2)), t);
h3 = ilaplace(r(3) / (s - p(3)), t);
h = h1 + h2 + h3
```

```
>> t3
```

```
h =
```

```
exp(-2*t) - (2*exp(-3*t))/3 + 2/3
```

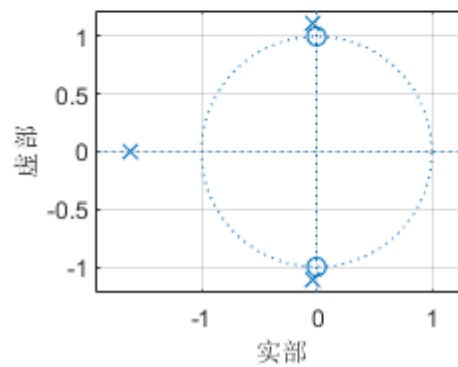
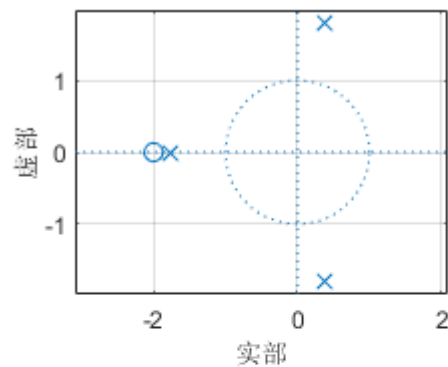
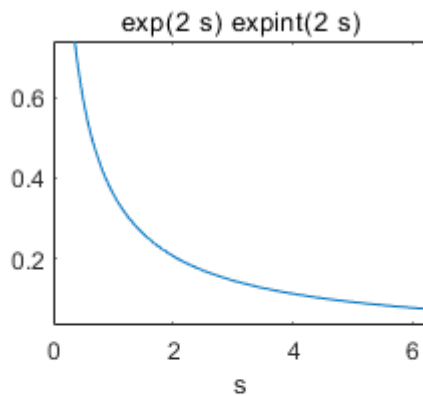
4

```

syms t;
f=1/(t+2);
subplot(2,2,1);
F=laplace(f);
ezplot(F)
A1=[0 0 1 2];
B1=[1 1 2 6];
subplot(2,2,2)
zplane(A1,B1);
grid on;

A2=[0 1 0 1];
B2=[3 5 4 6];
subplot(2,2,4)
zplane(A2,B2);

```



H1 不稳定,H2 稳定。

5

```
syms t s;  
H(s) = s / (s^2 + 3 * s + 2)  
h(t) = ilaplace(H(s))  
g(t) = ilaplace(H(s) / s)  
f(t) = cos(20 * t) * heaviside(t);  
y = ilaplace(H(s) * laplace(f(t)))  
H(s) =  
  
s/(s^2 + 3*s + 2)  
  
h(t) =  
  
2*exp(-2*t) - exp(-t)  
  
g(t) =  
  
exp(-t) - exp(-2*t)  
  
y =  
  
(300*cos(20*t))/40501 + exp(-t)/401 - exp(-2*t)/101 + (1990*sin(20*t))/40501
```

实验总结

通过本次实验，我学会了使用 **MATLAB** 处理傅里叶（反）变换和拉普拉斯（反）变换的具体做法，同时提升了 **MATLAB** 的 **Debug** 能力。

参考文献

<https://ww2.mathworks.cn/help/matlab/ref/conv.html>

<https://ww2.mathworks.cn/help/matlab/ref/residue.html>

<https://ww2.mathworks.cn/help/matlab/ref/roots.html>