西安电子科技大学网络与信息安全学院
信号与系统 实验报告

班　　级：_

学　　号：_

姓　　名：_

电子邮箱：

指导教师：_

2024 年 5 月 24 日

实验题目：

1. 使用Matlab函数fourier()计算 $f(t)=e^{-2|t|}$ 的傅里叶变换，用函数ifourier()计算 $F(j\omega)=\dfrac{1}{1+\omega^2}$ 傅里叶反变换。

实验摘要

用 **MATLAB** 的 **fourier** 和 **ifourier** 函数计算双边指数函数的傅里叶正变换和反变换

题目描述

这题很简单，直接输入函数，计算变换，然后画出计算结果，与理论相比较。

实验过程与分析

首先从理论上考虑，双边指数函数 $e^{-\alpha|t|}$ 的傅里叶变换是 $\dfrac{2\alpha}{\alpha^2+\omega^2}$

在本题中，$\alpha=2, F(j\omega)=\dfrac{4}{4+\omega^2}$

(1)计算正变换

MATLAB 代码：

```
syms t;
func1 = exp(-2 * abs(t));
F = fourier(func1)
ezplot(F);
grid on;
```

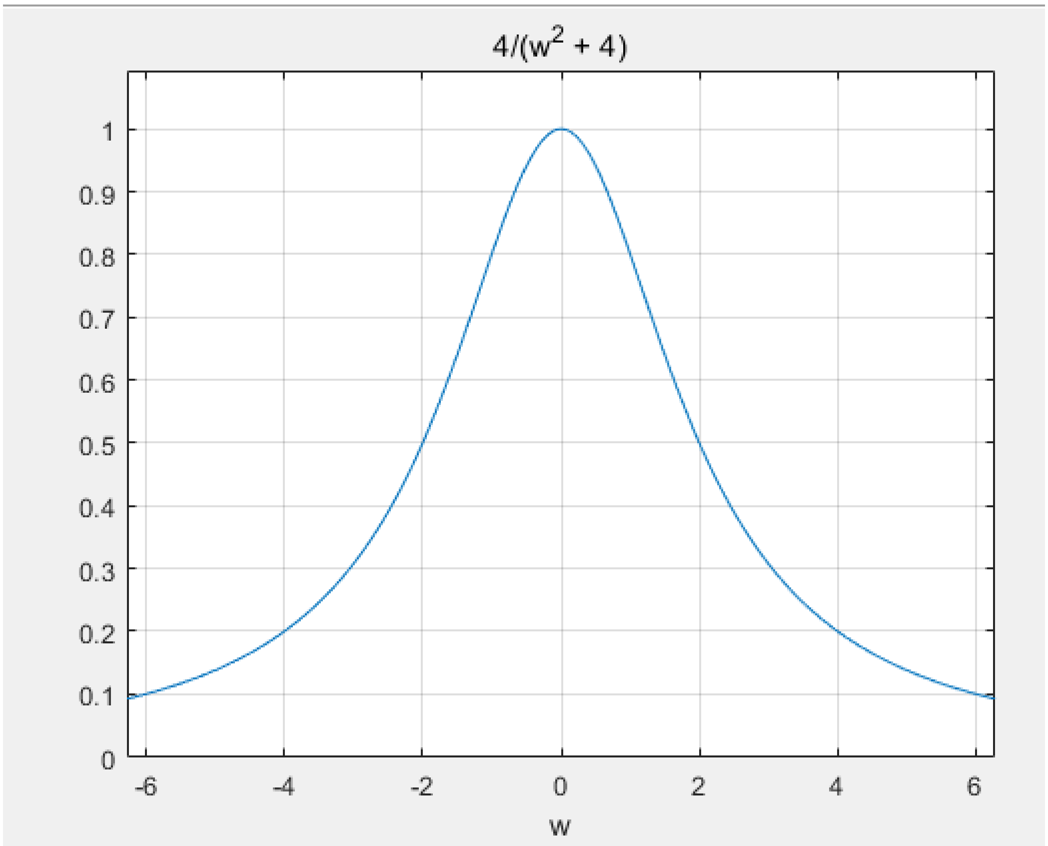
运行结果：

```
>> exp3

F =

4/(w^2 + 4)
```

傅里叶变换的函数与理论完全一致，现在用 **ezplot** 绘制其在 **-2**π到 **2**π之间的频谱



(1)计算反变换

先从理论上分析，双边指数函数$e^{-\alpha|t|}$的傅里叶变换是 $\dfrac{2\alpha}{\alpha^2+\omega^2}$

$F(j\omega)=\dfrac{1}{1+\omega^2}$，其时域的信号应该为 $e^{-0.5|t|}$

MATLAB 代码

```
syms w;
F = 1/(1+w^2);
f = ifourier(F)
ezplot(f);
grid on;
```

运行结果

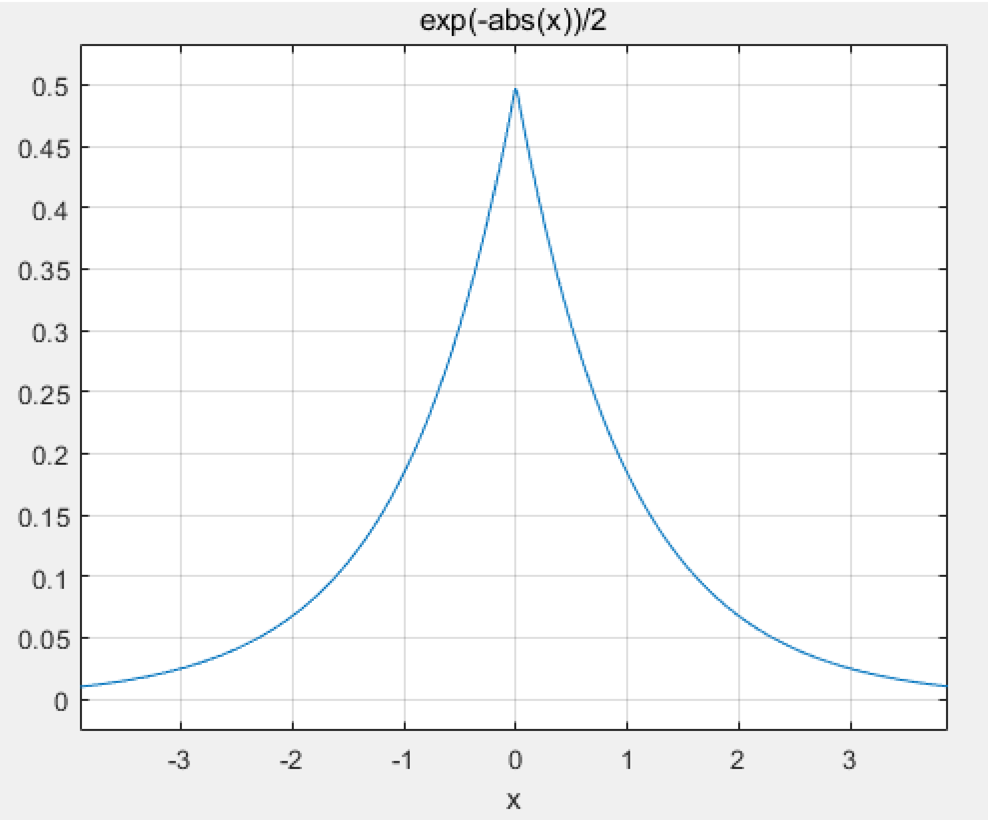
```
>> exp3

f =

exp(-abs(x))/2
```

与理论一致

时域信号的图像为



实验总结: 这一题比较简单，没有遇到什么问题。 在这一题中学到了 **Fourier** 和 **iFourier** 的使用，以及用 **ezplot** 方便地绘制一个函数的图像

参考文献：

Fourier 的用法

语法

1. `F = fourier(f)`
   - 计算符号表达式 `f` 的傅里叶变换，默认的变量由 `symvar(f,1)` 确定。
2. `F = fourier(f, x)`
   - 计算符号表达式 `f` 相对于变量 `x` 的傅里叶变换。
3. `F = fourier(f, x, w)`
   - 计算符号表达式 `f` 相对于变量 `x` 的傅里叶变换，并以频率变量 `w` 表示结果。

示例

1. **基本傅里叶变换**

matlab 复制代码

```matlab
syms t
f = exp(-t^2);
F = fourier(f)
```

这将计算 `exp(-t^2)` 相对于 `t` 的傅里叶变换。

2. **指定变换变量**

matlab 复制代码

```matlab
syms t w
f = exp(-t^2);
F = fourier(f, t, w)
```

这将计算 `exp(-t^2)` 相对于 `t` 的傅里叶变换，并以 `w` 表示结果。

Ezplot 的用法

1. **绘制单变量函数**

matlab 复制代码

```matlab
ezplot(f)
```

- 其中 `f` 是一个符号表达式或字符串，表示单变量函数。

2. **绘制单变量函数，并指定范围**

matlab 复制代码

```matlab
ezplot(f, [xmin, xmax])
```

- 在指定的 `xmin` 到 `xmax` 范围内绘制函数 `f`。

3. **绘制隐函数**

matlab 复制代码

```matlab
ezplot(f == g)
```

- 绘制隐函数 `f == g` 的图形，其中 `f` 和 `g` 是符号表达式。

4. **绘制参数方程**

matlab 复制代码

```matlab
ezplot(x, y)
```

- 绘制由参数变量定义的参数方程 `x` 和 `y`，其中 `x` 和 `y` 是函数表达式，参数通常是 `t`。

5. **绘制参数方程，并指定参数范围**

matlab 复制代码

```matlab
ezplot(x, y, [tmin, tmax])
```

- 在指定的参数范围 `tmin` 到 `tmax` 内绘制参数方程 `x` 和 `y`。

实验题目：

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。(注：其它可能使用到的函数有abs()，angle()，heaviside())

实验摘要

两个函数，f2 是 f1 时移了一个单位。

计算傅里叶变换，画出幅度谱、相位谱、然后观察时移特性

题目描述

根据傅里叶变换的时移特性

五、时移性质

如果 $f(t)\longleftrightarrow F(j\omega)$，则 $\boxed{f(t\pm t_0)\longleftrightarrow e^{\pm j\omega t_0}F(j\omega)}$

其中“t_0”是实常数。

如果 **f1(t)**的傅里叶变换是 **F（jw）**，则

f2(t)的傅里叶变换应该只是在 **F(jw)**的基础上多了一个 **e^(-jw)**

从复数的极坐标表示分析：

理论上，这对幅度不产生任何影响

对相位而言，相位会被加上**-w**

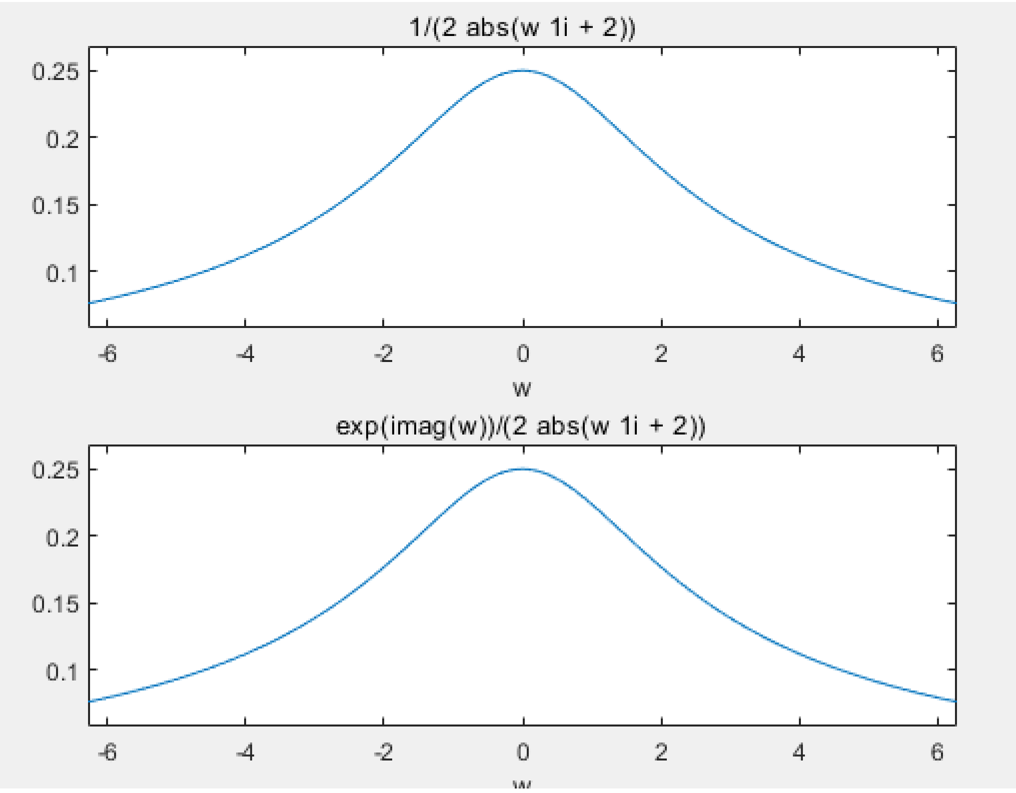
实验过程与分析

MATLAB 代码

```
syms t;
f1 = 0.5 * exp(-2*t) * heaviside(t);
f2 = 0.5 * exp(-2*(t-1)) * heaviside(t-1);

F1 = fourier(f1);
F2 = fourier(f2);

subplot(2,1,1)%用于同时展示两个图
ezplot(abs(F1));
subplot(2,1,2)%用于同时展示两个图
ezplot(abs(F2));
```

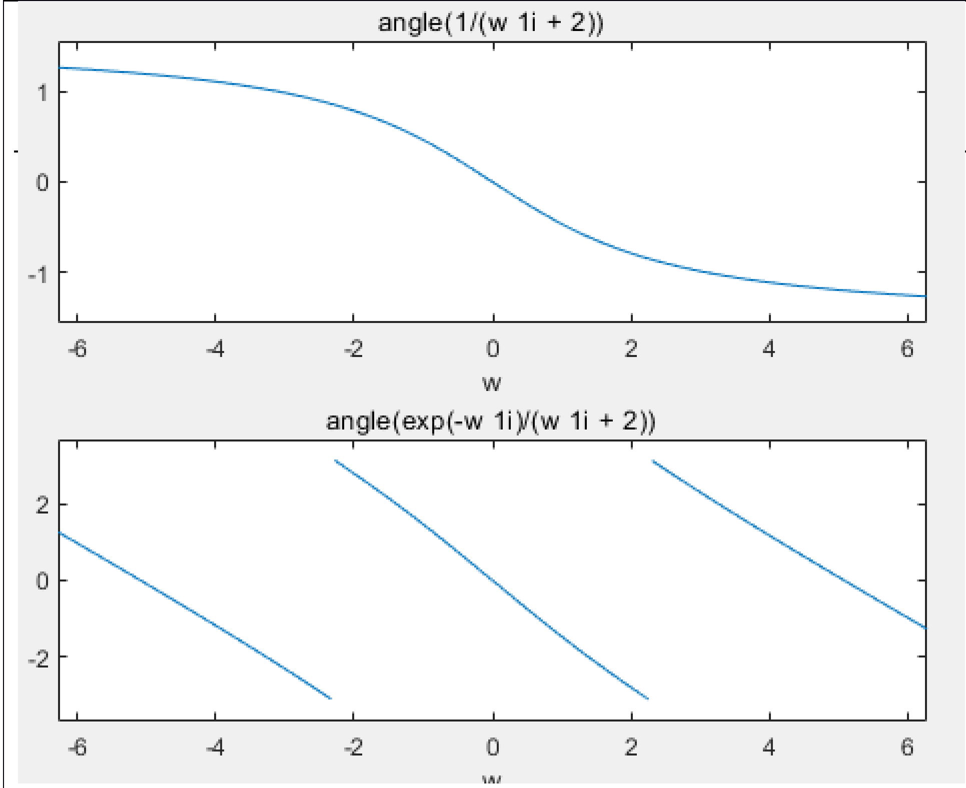


上图为 **f1** 的幅度谱，下图为 **f2** 的幅度谱

可见，幅度谱是完全一样的

将绘制图形部分的代码改为以下内容，用于画出相位谱

```
subplot(2,1,1)%用于同时展示两个图
ezplot(angle(F1));
subplot(2,1,2)%用于同时展示两个图
ezplot(angle(F2));
```



上图为 **f1** 的相位谱，下图为 **f2** 的相位谱

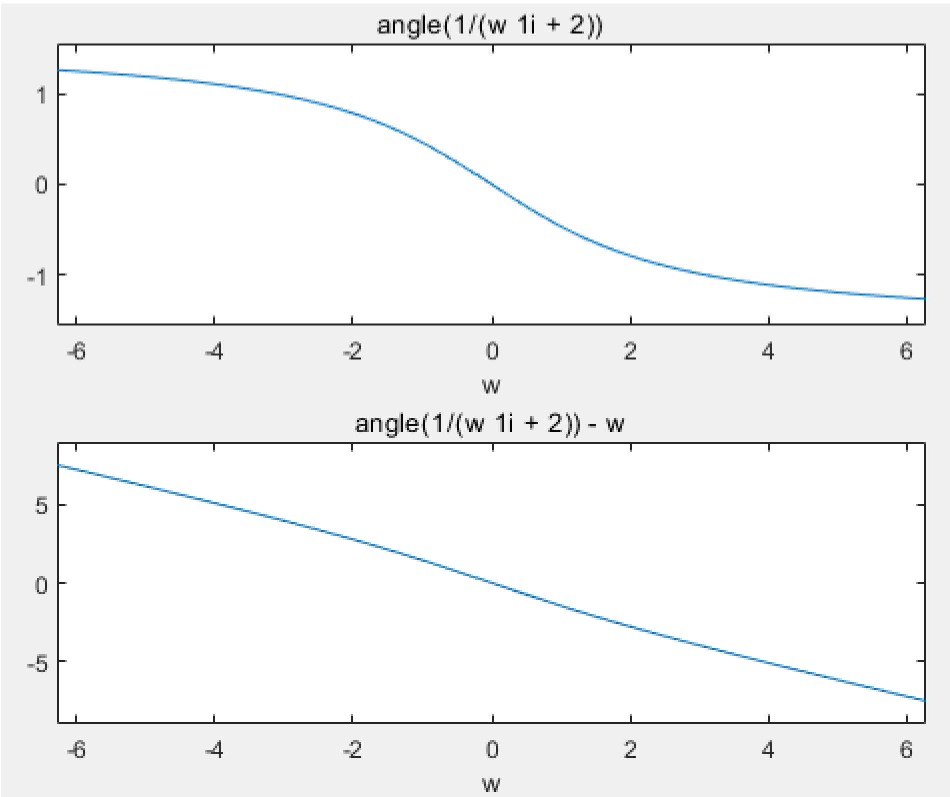
接下来用理论验证结果是否正确

在“题目描述”中，已经提到，如果 **f1** 的相位谱函数记为 $\varphi_1(\omega)$

那么 **f2** 的相位谱函数应该是 $\varphi_2(\omega)=\varphi_1(\omega)-\omega$

我最开始使用这行代码画出 $\varphi_1(\omega)-\omega$：

```
subplot(2,1,1)
ezplot(angle(F1));
subplot(2,1,2)
ezplot(angle(F1)-w);
```

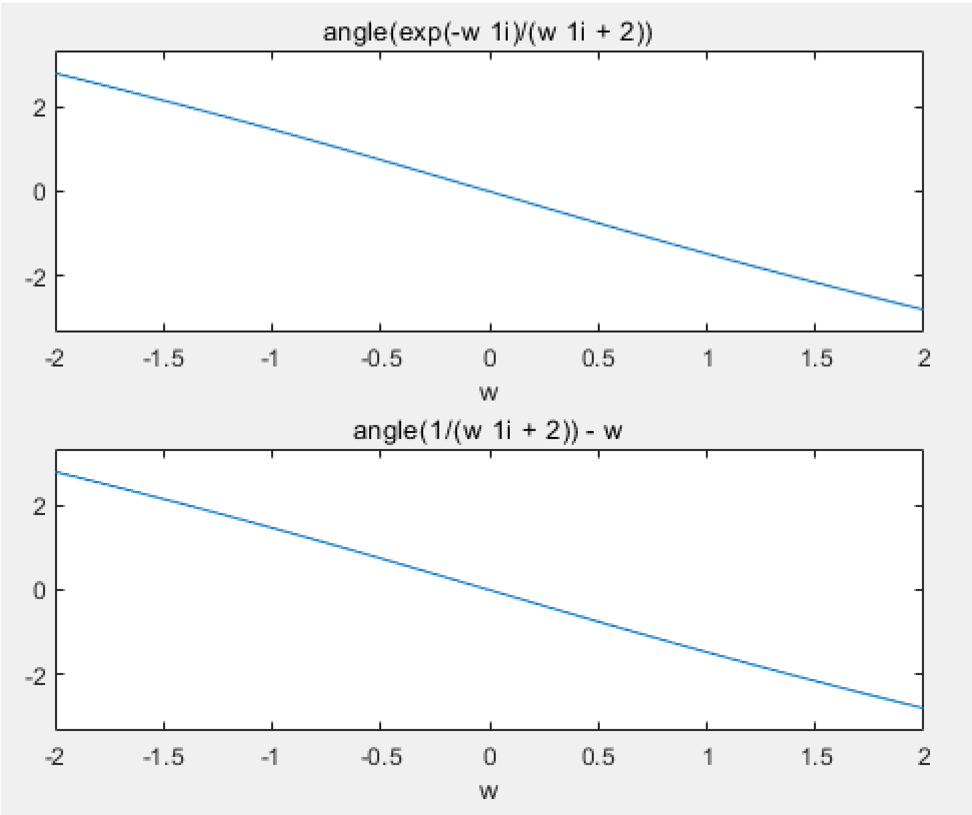


可以观察到$\varphi_1(\omega)-\omega$图像是连续的，而直接 ezplot(angle(F2));的结果是有断点的，这是因为相位的范围是-π到π，一旦超过了这个范围，就要+或者-2kπ

所以，设置 **ezplot** 的绘图范围为**-2** 到 **2**，这样图像不会出现断点。便于我们观察比较、

比较 **angle(F2)**与 **angle(F1)-w** 的图像，如果一致，就说明结果正确

```
subplot(2,1,1)
ezplot(angle(F2), [-2,2]);
subplot(2,1,2)
ezplot(angle(F1)-w, [-2,2]);
```



可见，完全一致，说明我们的理论与结果均正确。

实验总结：

这次实验没有出现什么 bug 和问题。

从理论分析到编程实现均顺利。

学习到了 MATLAB 中 angle、abs 的用法、以及 ezplot 改变绘图范围的用法

之前一直认为 abs 只是一个绝对值函数，通过这题才了解到，也可以用于计算复数的模值。

参考文献：

在 MATLAB 中，`angle` 函数用于计算复数的相位角（也称为辐角）。相位角表示复数在复平面上的角度，单位是弧度。

语法

matlab　　复制代码

```matlab
theta = angle(z)
```

- `z` 是一个复数或复数数组。
- `theta` 是 `z` 的相位角，单位为弧度。

五、时移性质

如果 $f(t) \longleftrightarrow F(j\omega)$，则 $\boxed{f(t \pm t_0) \longleftrightarrow e^{\pm j\omega t_0} F(j\omega)}$

其中“t_0”是实常数。

实验题目：

3. 计算 $f_1(t)=g_4(t)$、$f_2(t)=\cos(\omega_c t)$，以及 $f_3(t)=g_4(t)\cos(\omega_c t)$ 的傅里叶变换，其中 $f_c=100Hz$，画出三个信号的幅度谱，并观察讨论它们之间的关系。

实验摘要

门函数 **f1**，余弦载波信号 **f2**,以及调制信号 **f3**

画出三个信号的幅度谱

题目描述

由理论：

$$f_1(t)\leftrightarrow 4Sa(2\omega)=\frac{2\sin(2\omega)}{\omega}=F_1(j\omega)$$

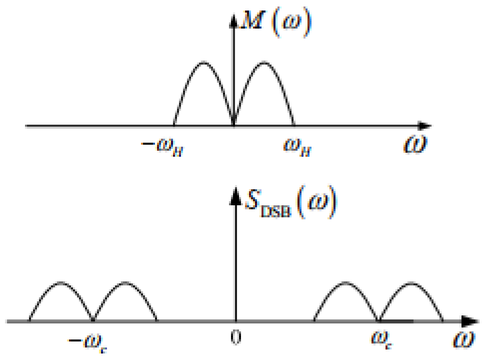
$$f_2(t)\leftrightarrow \pi\left[\delta(\omega-\omega_c)+\delta(\omega+\omega_c)\right]$$

由傅里叶变换的频移特性，理论上

$$F_3(j\omega)=\frac{1}{2}\left[F_1(j(\omega-\omega_c))+F_1(j(\omega+\omega_c))\right]$$

等于将 **F1** 幅值缩小一半后，向左搬移 **wc**，向右搬移 **wc**，然后合起来

如图



实验过程与分析

MATLAB 代码

```
syms t;
syms w;
width = 4;%门的宽度
wc = 2 * pi * 100;

f1 = heaviside(t + width/2) - heaviside(t - width/2);
f2 = cos(wc * t);
f3 = f1 * f2;
```

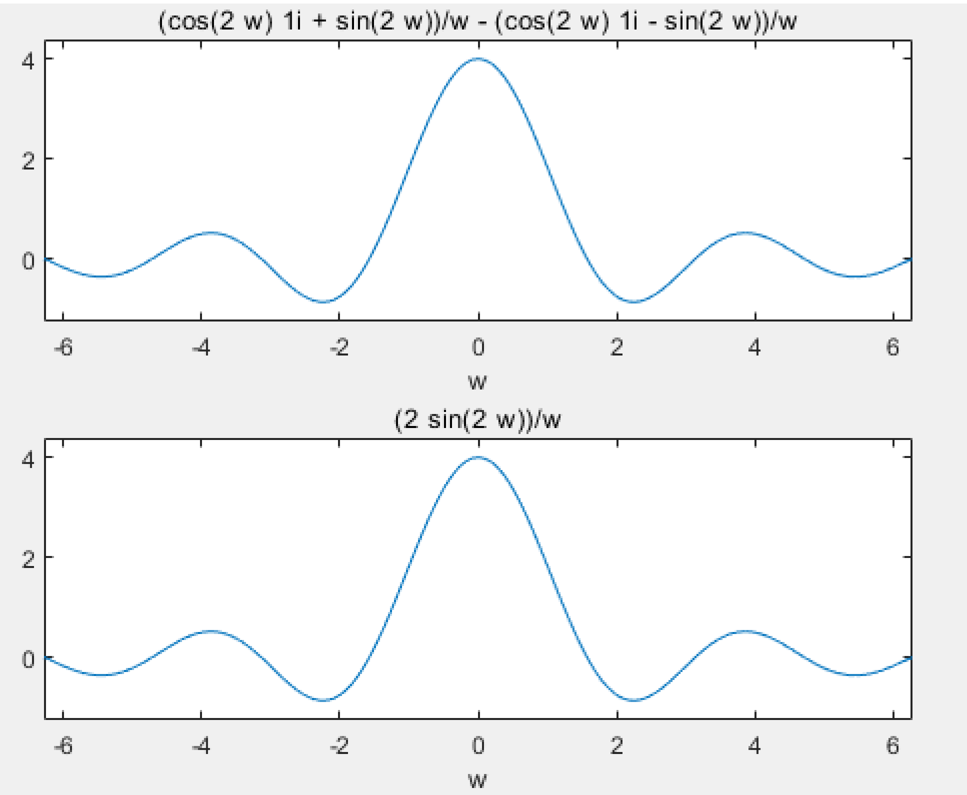
(1)f1 的幅度谱

由理论可知 **f1** 的幅度谱是实的，无需用 **abs** 计算幅值

用这几行代码画出 **f1** 幅度谱，(2*sin(2*w))/w 即理论值 $4Sa(2\omega)=\dfrac{2\sin(2\omega)}{\omega}$

比较电脑计算与理论是否一致

```
subplot(2,1,1)
ezplot(fourier(f1));
subplot(2,1,2)
ezplot((2*sin(2*w))/w);
```



二者一致，说明幅度谱就长这样

(2) f2 的幅度谱

由上述理论可知，**f2** 的傅里叶变换是两个冲激，

分别位于 **w=-200π**和 **200π**

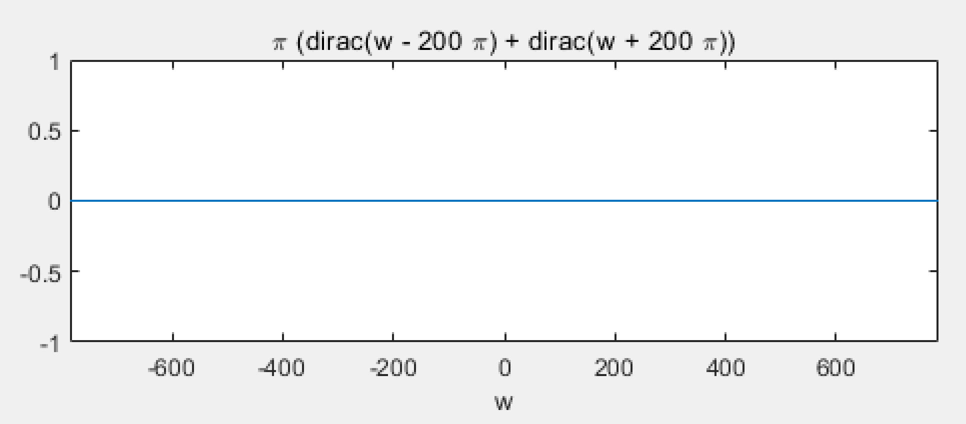
Ezplot 的默认绘图范围是**-2pi** 到 **2pi**

因此我们需要调整 **ezplot** 的绘图范围，不然得到的只是一条**-2pi** 到 **2pi** 的水平线

如果直接 `ezplot(fourier(f2),[-250*pi, 250*pi]);`虽然 **fourier(f2)**显示的结果与

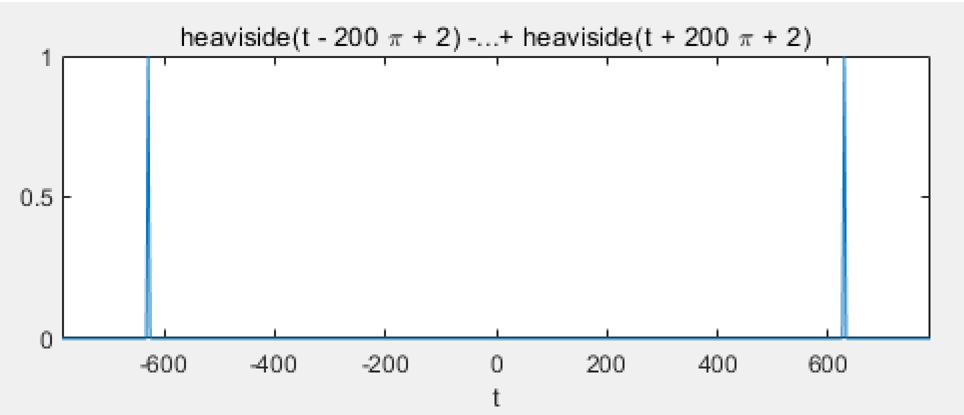
理论值$\pi[\delta(\omega-200\pi)+\delta(\omega+200\pi)]$一致

但是并不会得到两个无限高的尖峰（或是 1），仍然是水平线



于是我采用两个很窄的门函数相加模拟冲激，使得 **f2** 的幅度谱在 **MATLAB** 上可视化

f2 的幅度谱长这样



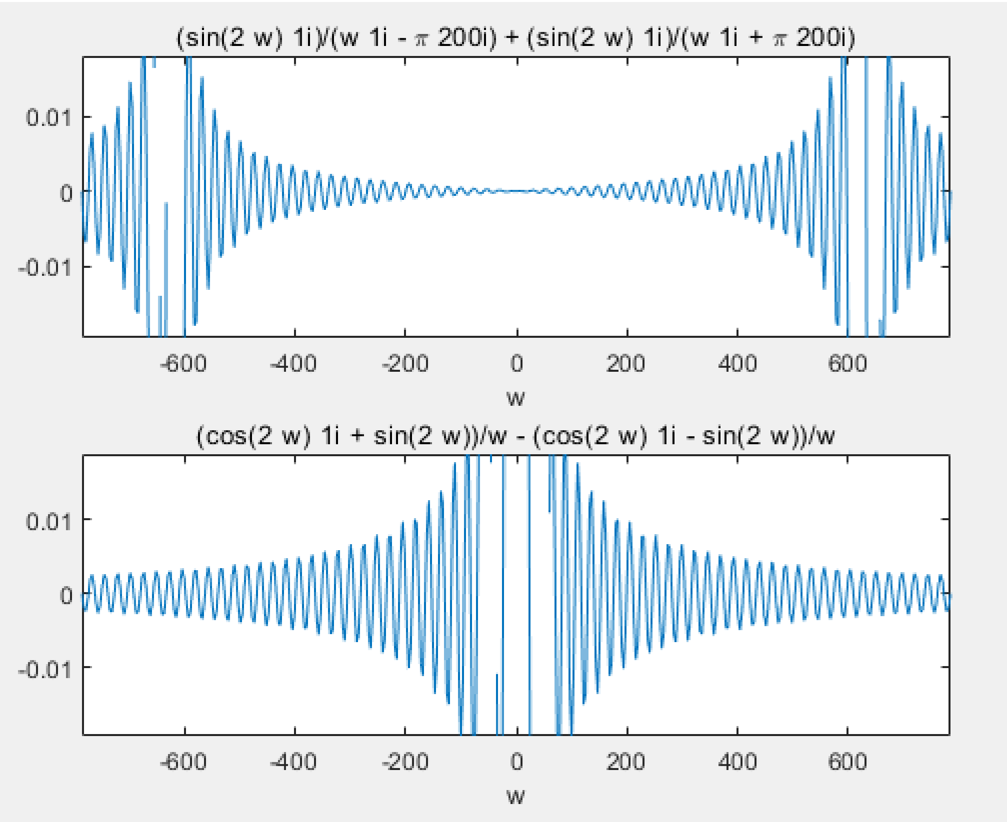
(3)f3 的幅度谱

在上文，已经分析了 **f3** 的幅度谱是 **f1** 的幅度谱左右“分裂”两个，式子如下

$$F_3(j\omega)=\frac{1}{2}\left[F_1(j(\omega-\omega_c))+F_1(j(\omega+\omega_c))\right]$$

把 **f3** 和 **f1** 的幅度谱画在一起，看看是不是这么回事

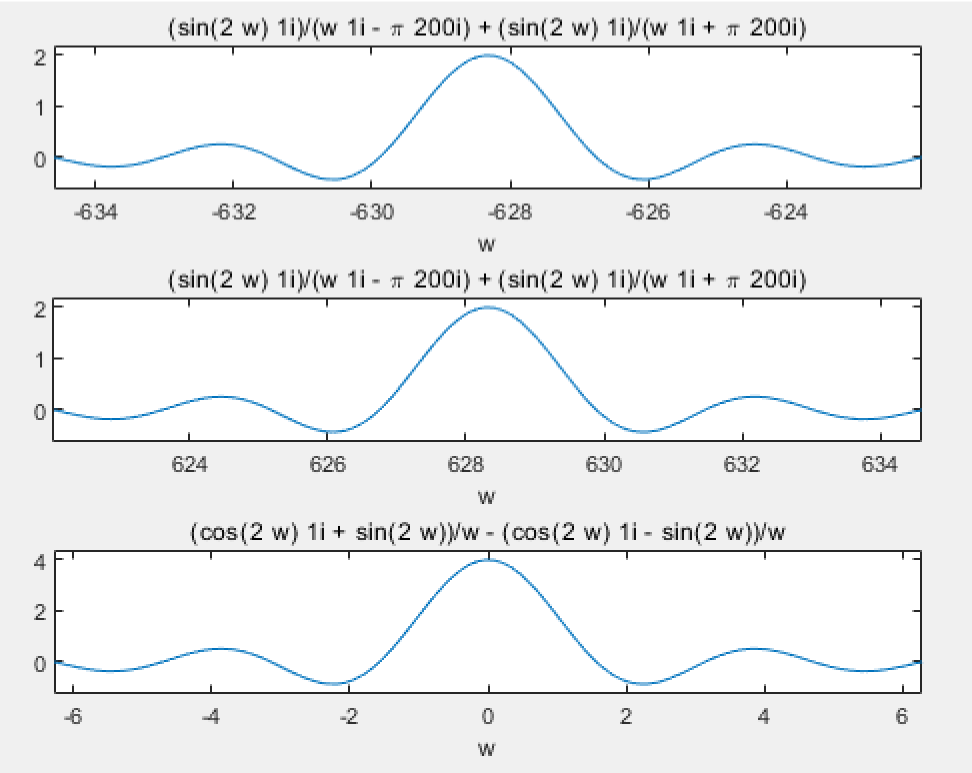
```
subplot(2,1,1)
ezplot(fourier(f3),[-250*pi,250*pi]);
subplot(2,1,2)
ezplot(fourier(f1),[-250*pi,250*pi]);
```



由于傅里叶变换的收敛性。到 **w** 比较大的位置时，幅值已经很小了
图形的效果很差。但是可以看出确实是左右搬移。

所以缩小区间，让图像可读性更强
对 **f3** 在**(-202π，-198π)**和**(198，202π)**的频谱
与 **f1** 在**(-2π，2π)**的频谱作比较

```
subplot(3,1,1)
ezplot(fourier(f3),[-202*pi,-198*pi]);
subplot(3,1,2)
ezplot(fourier(f3),[198*pi,202*pi]);
subplot(3,1,3)
ezplot(fourier(f1));
```



可见：

f3 在(-202π，-198π)和(198，202π)的频谱，与 f1 在(-2π，2π)的频谱形状基本完全一样，前者幅值是后者的一半。

符合 **DSB** 调制的理论。

结论：$F_3(j\omega)=\dfrac{1}{2\pi}F_1(j\omega)*F_2(j\omega)$

$F_3(j\omega)$ 就是 $F_1(j(\omega-200\pi))+F_1(j(\omega+200\pi))$

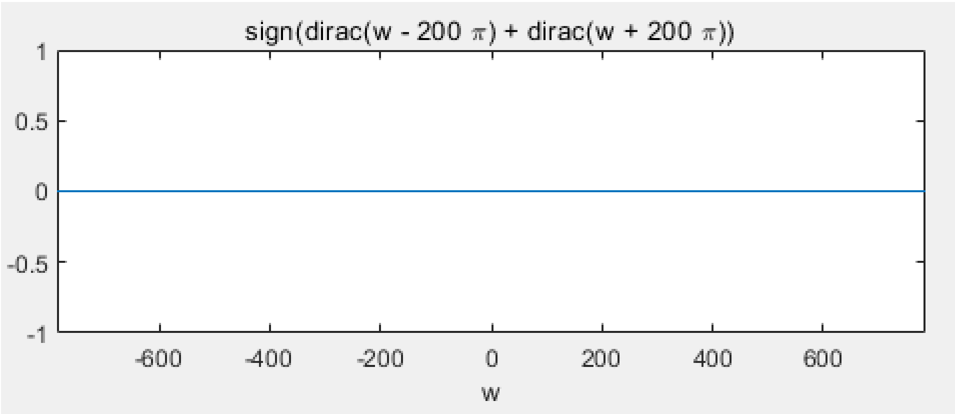
实验总结:

在绘制 **f2** 的幅度谱时,本想使用老师上课提到的 **sign()**对冲激函数进行可视化。

但是实际操作中存在很多我无法解决的问题

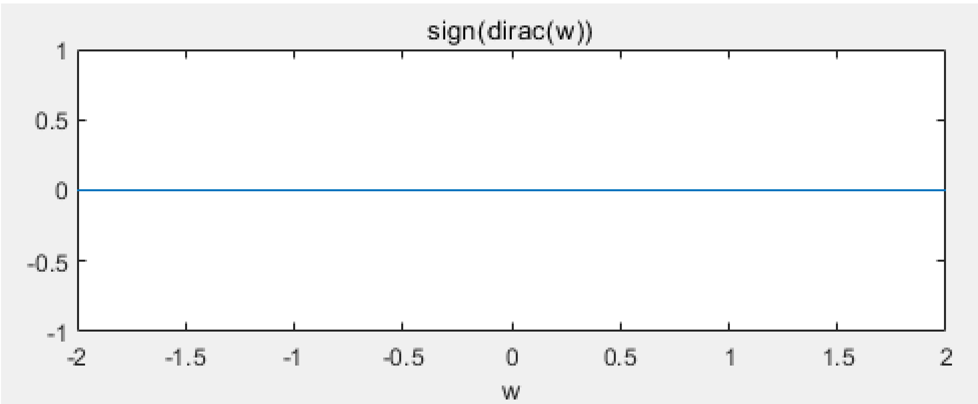
直接 **fourier(f2)**,返回的是 **dirac** 函数。

如果绘制 **sign(fourier(f2))**:



依然是一条水平线。

如果直接绘制 **sign(dirac(w))**



在 **0** 处依然也是什么都没有

所以无奈选择了用很窄的门函数可视化冲激函数。

通过这次实验，复习到了 **DSB** 调制、傅里叶变换的性质。

也提高了利用 **MATLAB** 解决信号与系统问题的能力

参考文献：

4.3 节 PPT

例2： $f(t)=\cos\omega_0 t \longleftrightarrow F(j\omega)=?$

解： $f(t)=\frac{1}{2}e^{j\omega_0 t}+\frac{1}{2}e^{-j\omega_0 t}$

$$F(j\omega)=\pi[\delta(\omega-\omega_0)+\delta(\omega+\omega_0)]$$

信号与系统 电子教案　　4.5　傅里叶变换的性质

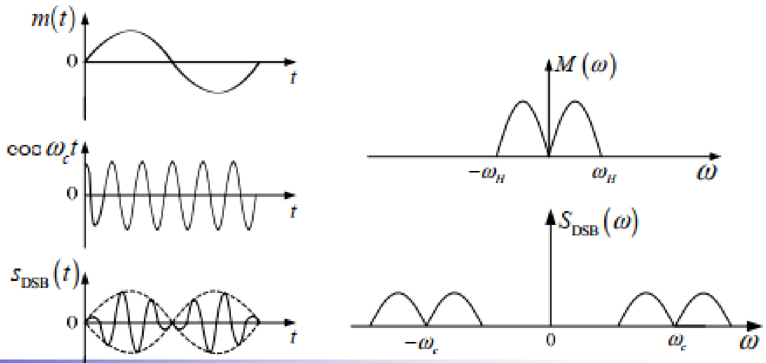
例3： 已知$f(t)\longleftrightarrow F(j\omega)$，调制信号 $f(t)\cos\omega_0 t \longleftrightarrow ?$

双边带调制(DSB)

时域表示式：$s_{DSB}(t)=m(t)\cos\omega_c t$

频谱：$S_{DSB}(\omega)=\frac{1}{2}[M(\omega+\omega_c)+M(\omega-\omega_c)]$

曲线：



第1−20页

在 MATLAB 中，`dirac` 函数用于表示和计算狄拉克 δ 函数（Dirac delta function）。狄拉克 δ 函数是一个理想化的函数，用于表示无限窄且无限高的脉冲，其积分值为 1。它在信号处理、物理学和工程学中有广泛应用。

`dirac` 函数的用法

语法

matlab　复制代码

```matlab
y = dirac(x)
```

- `x` 是输入变量，可以是标量、向量或矩阵。
- `y` 是输出值，对应于 Dirac delta 函数的值。

当 `x` 为 0 时，`dirac(x)` 的值为无穷大 (Inf)；对于非零 `x`，`dirac(x)` 的值为 0。

MATLAB 中的冲激函数