西北工業大學

NORTHWESTERN POLYTECHNICAL UNIVERSITY

信号与系统实验报告

实验三：信号时域抽样和恢复

学院 软件学院

班级 14012104

学号 2021303423

姓名 申铭

2023 年 12 月 17 日

一、实验目的

- 掌握抽样定理，验证抽样定理；
- 掌握利用Matlab完成信号抽样的方法，并对抽样信号的频谱进行分析；
- 了解运用Matlab对抽样信信号进行恢复的方法。

二、实验环境

- 操作系统：Windows10
- 编程软件：Matlab2019b

三、实验涉及的部分 MATLAB 函数

1、stem

- 功能：绘制离散序列数据
- 调用格式：stem(Y)，stem(X,Y)

2、sinc

- 功能：计算 Sa 函数
- 调用格式：sinc(x)

3、heaviside

- 功能：实现阶跃信号
- 调用格式：heaviside(x)
- 当 x < 0 时返回 0，x > 0 时返回 1，x = 0 时返回 1/2

4、interp1

- 功能：一维数据插值
- 调用格式：vq = interp1(x,v,xq)

四、实验内容

1、实验1：抽样定理验证实验

已知连续信号为$f(t)=0.5(1+cost),-\pi\leq t\leq\pi$

（1）绘制$f(t)$时域波形和频谱；

- 手动求解$f(t)$的傅里叶变换

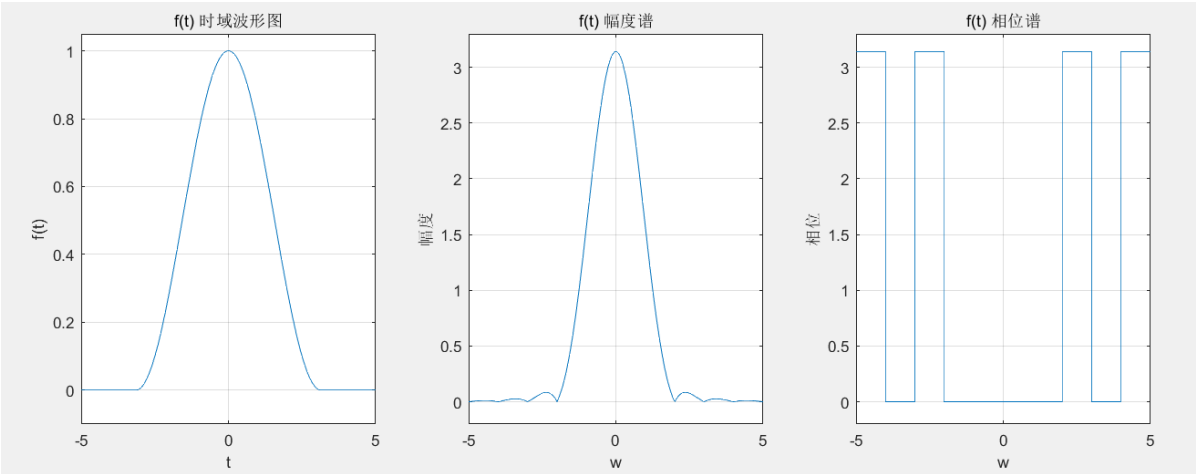
$$F(j\omega)=\int_{-\infty}^{\infty}f(t)e^{-j\omega t}\mathrm{d}t=\int_{-\infty}^{\infty}0.5(1+cost)*(cos(-\omega t)+jsin(-\omega t))\mathrm{d}t=\frac{sin(\pi\omega)}{\omega(1-\omega^2)}$$

- 绘制时域波形和频谱

代码：

```
% 绘制信号波形
t = -pi:0.001:pi
% syms t w
f = 0.5 * (1 + cos(t)) .* (heaviside(t + pi) - heaviside(t - pi)) % 定义信号
% F = fourier(f,t,w)
subplot(3,1,1)
plot(t,f)
grid()
xlabel('t')
ylabel('f(t)')
title('f(t) 时域波形图')
% 绘制幅度频谱
w = -10:0.001:10
F = sin(pi.*w)./(w.*(1-w.*w)) % 定义信号的FT（手动求解得到）
subplot(3,1,2)
plot(w,F)
grid()
xlabel('w')
ylabel('F(jw)')
title('f(t) 幅度谱')
% 绘制相位谱
subplot(3, 1, 3)
fplot(angle(ft))
title('f(t) 相位谱');
xlabel('w')
ylabel('相位');
axis([-5, 5, -0.2, 3.3])
```

结果图：



（2）分别绘制抽样间隔为0.5s、1s、2s时的抽样信号的时域波形和频谱；

> 注：抽样信号的幅度谱绘制三个周期即可。

- 以0.5s为抽样间隔

$$周期T=0.5s,抽样频率\omega_s=\frac{2\pi}{T}=4\pi$$

$$抽样信号信号f(k),假设f(k)\iff f_\delta(t)$$

$$抽样信号的频谱F_\delta(jw)=\frac{1}{T}\sum_{-\infty}^{\infty}F|j(\omega-k\omega_s)|=2\sum_{-\infty}^{\infty}F|j(\omega-4\pi k)|,取k=0,\pm1,\pm2,\pm3$$

- 以1s为抽样间隔

$$周期T=1s，抽样频率\omega_s=\frac{2\pi}{T}=2\pi$$

$$抽样信号的频谱F_\delta(jw)=\frac{1}{T}\sum_{-\infty}^{\infty}F|j(\omega-k\omega_s)|=\sum_{-\infty}^{\infty}F|j(\omega-2\pi k)|,取k=0,\pm1,\pm2,\pm3$$

-

- y

$$周期T=2s，抽样频率\omega_s=\frac{2\pi}{T}=\pi$$

$$抽样信号的频谱F_\delta(jw)=\frac{1}{T}\sum_{-\infty}^{\infty}F|j(\omega-k\omega_s)|=0.5\sum_{-\infty}^{\infty}F|j(\omega-\pi k)|,取k=0,\pm1,\pm2,.....,\pm10$$

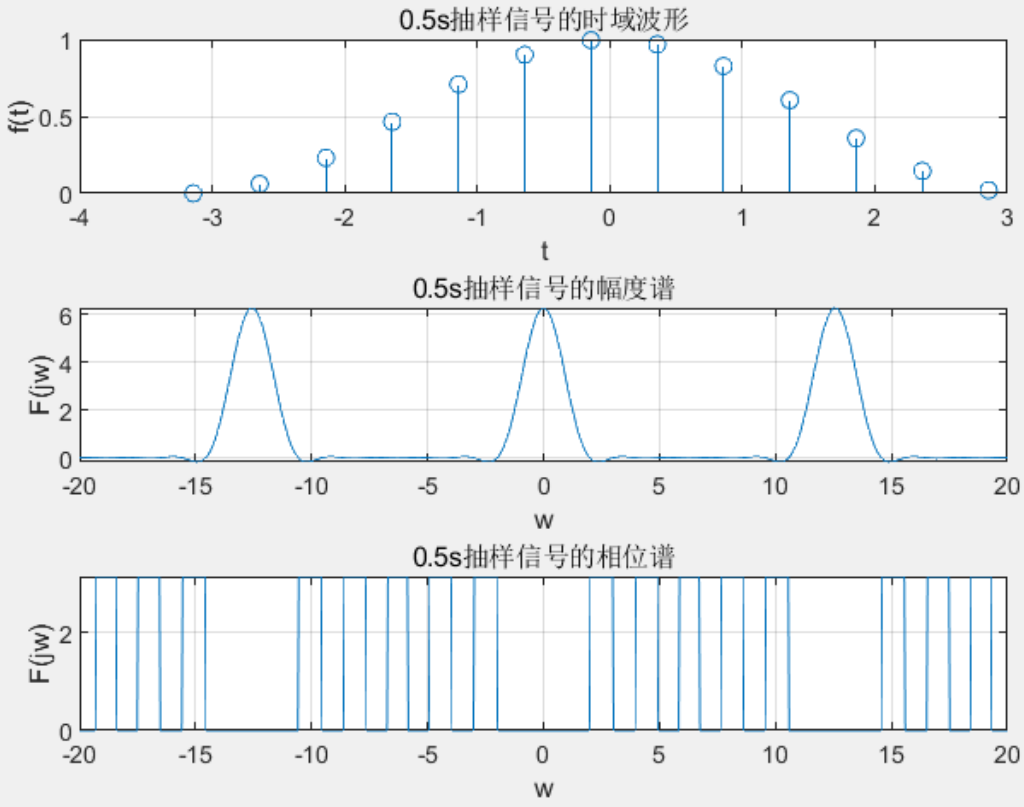
- 源代码

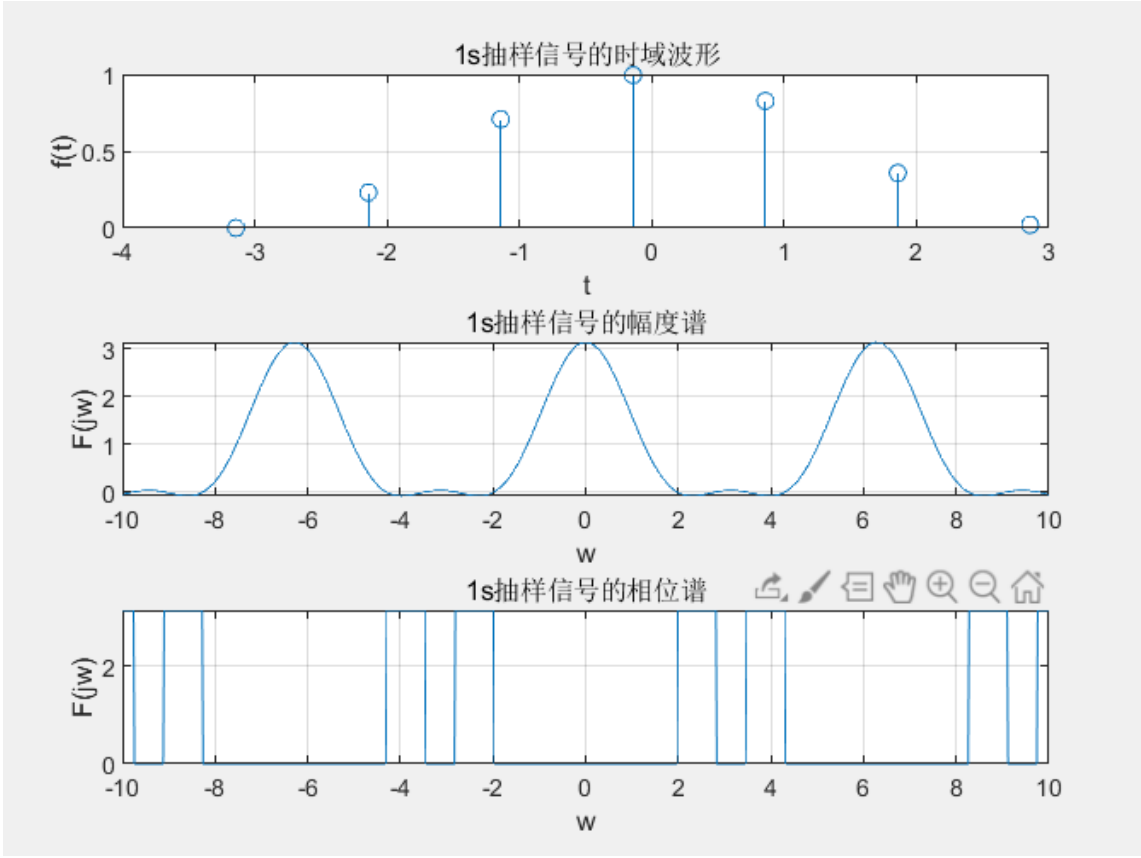
```
%% 1.2.1
t = -pi :0.5:pi
f = 0.5 * (1 + cos(t)) % 定义函数 f(t)
% 绘制抽样信号的时域波形
subplot(3,1,1)
stem(t, f) % 用stem函数绘制离散序列图
grid()
xlabel('t')
ylabel('f(t)')
title('0.5s抽样信号的时域波形')

w = linspace(-20, 20, 1000)
F = sin(pi.*w)./(w.*(1-w.*w)) % 定义信号的FT

% 循环计算频谱 F
for i = 1:3
    F = F + sin(pi.*(w + i * 4 * pi))./((w + i * 4 * pi).*(1-(w + i * 4 * pi).*
(w + i * 4 * pi)));
    F = F + sin(pi.*(w - i * 4 * pi))./((w - i * 4 * pi).*(1-(w - i * 4 * pi).*
(w - i * 4 * pi)));
end
% 绘制抽样信号的频谱
subplot(3,1,2)
plot(w, F * 2)
xlabel('w')
ylabel('F(jw)')
grid()
title('0.5s抽样信号的幅度谱')
% 绘制抽样信号的相位谱
subplot(3,1,3)
plot(w,angle(F))
xlabel('w')
ylabel('F(jw)')
```

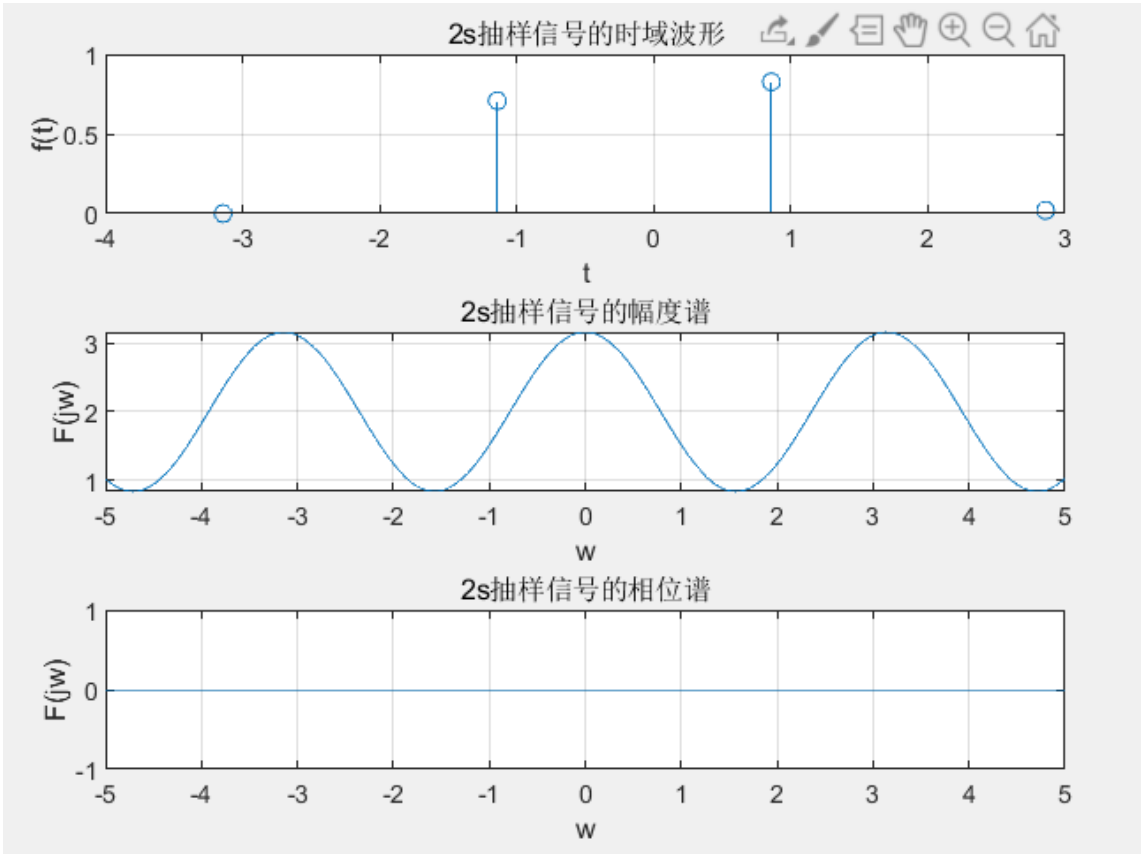
```
grid()
title('0.5s抽样信号的相位谱')

%% 1.2.2
t = -pi :1:pi
f = 0.5 * (1 + cos(t)) % 定义函数 f(t)
% 绘制抽样信号的时域波形
subplot(3,1,1)
stem(t, f) % 用stem函数绘制离散序列图
grid()
xlabel('t')
ylabel('f(t)')
title('1s抽样信号的时域波形')

w = linspace(-10, 10, 1000)
F = sin(pi.*w)./(w.*(1-w.*w)) % 定义信号的FT

for i=1:3
    F = F+sin(pi.*(w + i * 2 * pi))./((w + i * 2 * pi).*(1-(w + i * 2 * pi).*(w
+ i * 2 * pi)))
    F = F+sin(pi.*(w - i * 2 * pi))./((w - i * 2 * pi).*(1-(w - i * 2 * pi).*(w
- i * 2 * pi)))
end
% 绘制抽样信号的幅度谱
subplot(3,1,2)
plot(w,F)
xlabel('w')
ylabel('F(jw)')
grid()
title('1s抽样信号的幅度谱')
% 绘制抽样信号的相位谱
subplot(3,1,3)
plot(w,angle(F))
xlabel('w')
ylabel('F(jw)')
grid()
title('1s抽样信号的相位谱')

%% 1.2.3
t = -pi :2:pi
f = 0.5 * (1 + cos(t)) % 定义函数 f(t)
% 绘制抽样信号的时域波形
subplot(3,1,1)
stem(t, f) % 用stem函数绘制离散序列图
grid()
xlabel('t')
ylabel('f(t)')
title('2s抽样信号的时域波形')

w = linspace(-5, 5, 1000)
F = sin(pi.*w)./(w.*(1-w.*w)) % 定义信号的FT

for i=1:10
    F = F+sin(pi.*(w + i * pi))./((w + i * pi).*(1-(w + i * pi).*(w + i * pi)))
    F = F+sin(pi.*(w - i * pi))./((w - i * pi).*(1-(w - i * pi).*(w - i * pi)))
end
% 绘制抽样信号的幅度谱
subplot(3,1,2)
```

```
plot(w,F)
xlabel('w')
ylabel('F(jw)')
grid()
title('2s抽样信号的幅度谱')
% 绘制抽样信号的相位谱
subplot(3,1,3)
plot(w,angle(F))
xlabel('w')
ylabel('F(jw)')
grid()
title('2s抽样信号的相位谱')
```

- 结果图







（3）观察抽样信号的频谱混叠程度，验证抽样定理。

抽样定理： 信号的抽样频率必须至少是信号最高频率的两倍。如果一个信号的最高频率为ω_m，那么为了准确地重构这个信号，抽样频率ω_s必须满足$\omega_s \geq 2\omega_m$。如果抽样频率小于两倍的信号最高频率，会导致混叠效应，即不同频率成分之间的信息混合在一起。

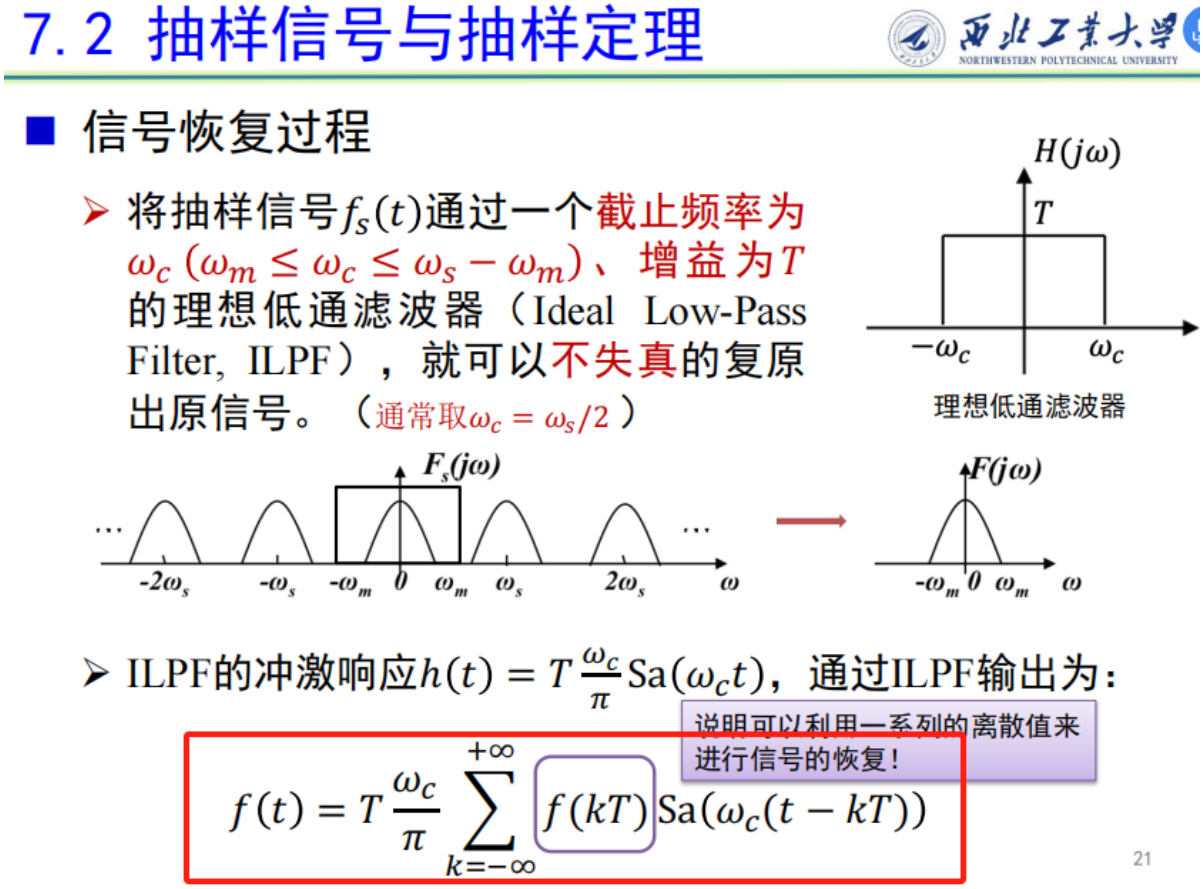
验证抽样定理：从（2）中三张图可以看出，$\omega_s = 2$，当$T = 0.5s$或$T = 1s$时，抽样后的频谱没有发生重叠，而当$T = 2s$时，抽样后相邻频谱发生混叠。综上，验证了抽样定理。

2、实验2：信号恢复实验

2.1

对实验1中的信号，观察到$\omega_m = 2$。对于抽样之后的信号，采用截止频率为$\omega_c = 1.2\omega_m$的ILPF进行信号恢复

（1）画出三种抽样间隔下抽样信号通过ILPF后的信号时域波形图；



使用下面的公式进行信号恢复：

$$f(t) = T\frac{\omega_c}{\pi}\sum_{-\infty}^{\infty} f(kt)Sa(\omega_c(t-kT))$$

根据公式，得到恢复信号的MATLAB计算表达式：

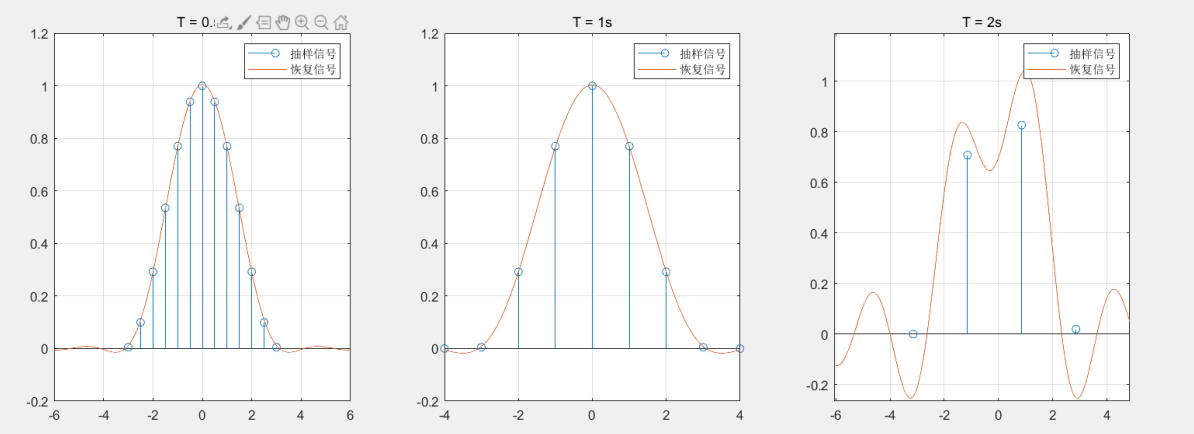
```
t_kT = ones(length(k),1)*t-kT'*ones(1,length(t));
xa=x*T(i)/pi*(sin(wc*t_kT)./(t_kT));
```

- 源代码

```
%% 2.1.1
t = -6 : 0.01 : 6;
f = ((1 + cos(t)) / 2) .* (heaviside(t + pi) - heaviside(t - pi)) %定义函数 f(t)

% 抽样间隔0.5s下 抽样信号通过ILPF后的信号时域波形图
Ts = 0.5; %抽样间隔
n = -6 : 6; %抽样点
nTs = n * Ts; %时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)); %抽样信
号
w_c = 2.4; %截止频率
```

```
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))); %恢复信号
figure, subplot(1, 3, 1);
stem(nTs, fs)
hold on
plot(t, ft),
title('T = 0.5s')
grid on,
legend('抽样信号', '恢复信号')

% 抽样间隔1s下 抽样信号通过ILPF后的信号时域波形图
Ts = 1 %抽样间隔
nTs = -6 : Ts : 6 %时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)) %抽样信
号
w_c = 2.4 %截止频率
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))) %恢复信号
subplot(1, 3, 2)
stem(nTs, fs)
hold on
plot(t, ft)
title('T = 1s')
grid on
legend('抽样信号', '恢复信号')
axis([-4, 4, -0.2, 1.2])

% 抽样间隔1s下 抽样信号通过ILPF后的信号时域波形图
Ts = 2 %抽样间隔
nTs = -pi : Ts : pi %时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)) %抽样信
号
w_c = 2.4; %截止频率
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))); %恢复信号
subplot(1, 3, 3)
stem(nTs, fs)
hold on
plot(t, ft)
title('T = 2s')
grid on
legend('抽样信号', '恢复信号')
```

- 结果图



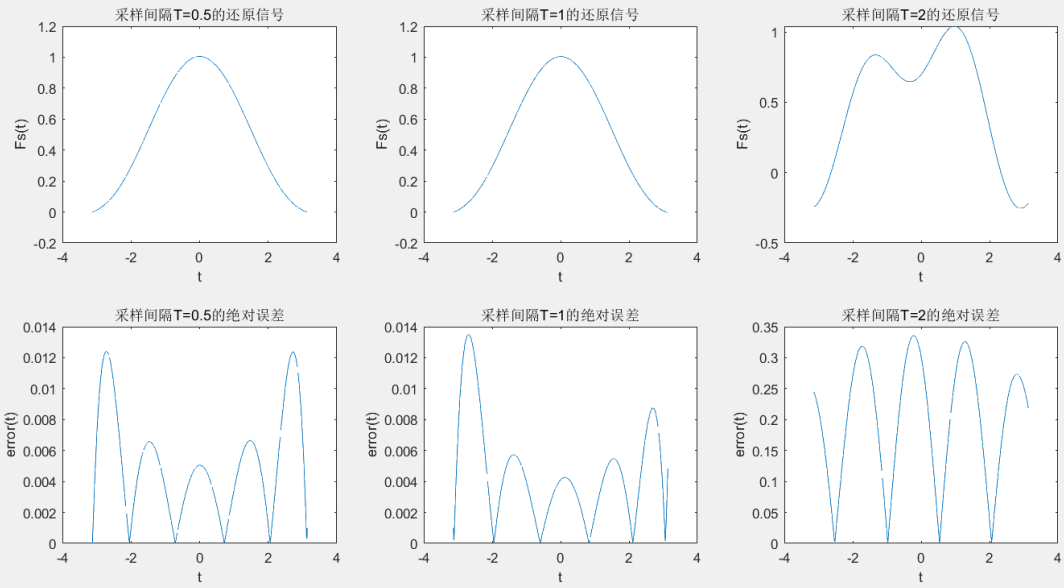
（2）绘制三种抽样间隔下的恢复信号与原信号的绝对误差图，观察并总结抽样间隔对于信号恢复过程的影响。

- 源代码

```
%% 2.1.2
t = -6 : 0.01 : 6
f = ((1 + cos(t)) / 2) .* (heaviside(t + pi) - heaviside(t - pi))  %定义函数 f(t)

% 抽样间隔0.5s下 恢复信号与原信号的绝对误差图
Ts = 0.5; %抽样间隔
n = -6 : 6; %抽样点
nTs = n * Ts; %抽样时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)); %抽样信
号
w_c = 2.4; %截止频率
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))); %恢复信号
subplot(2, 3, 1)
plot(t, ft)
title('采样间隔T=',num2str(T(i)),'的还原信号')
subplot(2, 3, 2)
plot(t, abs(f - ft))
axis([-6, 6, 0, 0.018])
title('采样间隔T=',num2str(T(i)),'的绝对误差')
grid on

% 抽样间隔1s下 恢复信号与原信号的绝对误差图
Ts = 1; %抽样间隔
n = -6 : 6; %抽样点
nTs = n * Ts; %抽样时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)); %抽样信
号
w_c = 2.4; %截止频率
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))); %恢复信号
figure(2)
subplot(2, 3, 3)
plot(t, ft)
title('采样间隔T=',num2str(T(i)),'的还原信号')
subplot(2, 3, 4)
plot(t, abs(f - ft))
axis([-6, 6, 0, 0.018])
title('采样间隔T=',num2str(T(i)),'的绝对误差')
grid on

% 抽样间隔2s下 恢复信号与原信号的绝对误差图
Ts = 2; %抽样间隔
n = -6 : 6; %抽样点
nTs = n * Ts; %抽样时间向量
fs = ((1 + cos(nTs)) / 2) .* (heaviside(nTs + pi) - heaviside(nTs - pi)); %抽样信
号
w_c = 2.4; %截止频率
ft = (Ts * w_c / pi) * fs * sinc((w_c ./ pi) * (ones(length(nTs), 1) * t - nTs'
* ones(1,length(t)))); %恢复信号
figure(3)
subplot(2, 3, 5)
```

```
plot(t, ft)
title('采样间隔T=',num2str(T(i)),'的还原信号')
subplot(2, 3, 6)
plot(t, abs(f - ft))
title('采样间隔T=',num2str(T(i)),'的绝对误差')
grid on
```

- 结果图



可以看到，当$T=0.5s$或$T=1s$时， 恢复信号与原信号的绝对误差较小，信号在理想情况下（ILPF）不失真；当$T=2s$时， 恢复信号与原信号的绝对误差较大，此时信号发生了混叠，还原出来的信号波形变形，发生了失真。

综上，抽样间隔满足$\omega_s \geq \omega_m$时，恢复信号与原信号的绝对误差较小，信号不失真；抽样间隔不满足$\omega_s \geq \omega_m$时，恢复信号与原信号的绝对误差较大，信号失真。符合抽样定理。

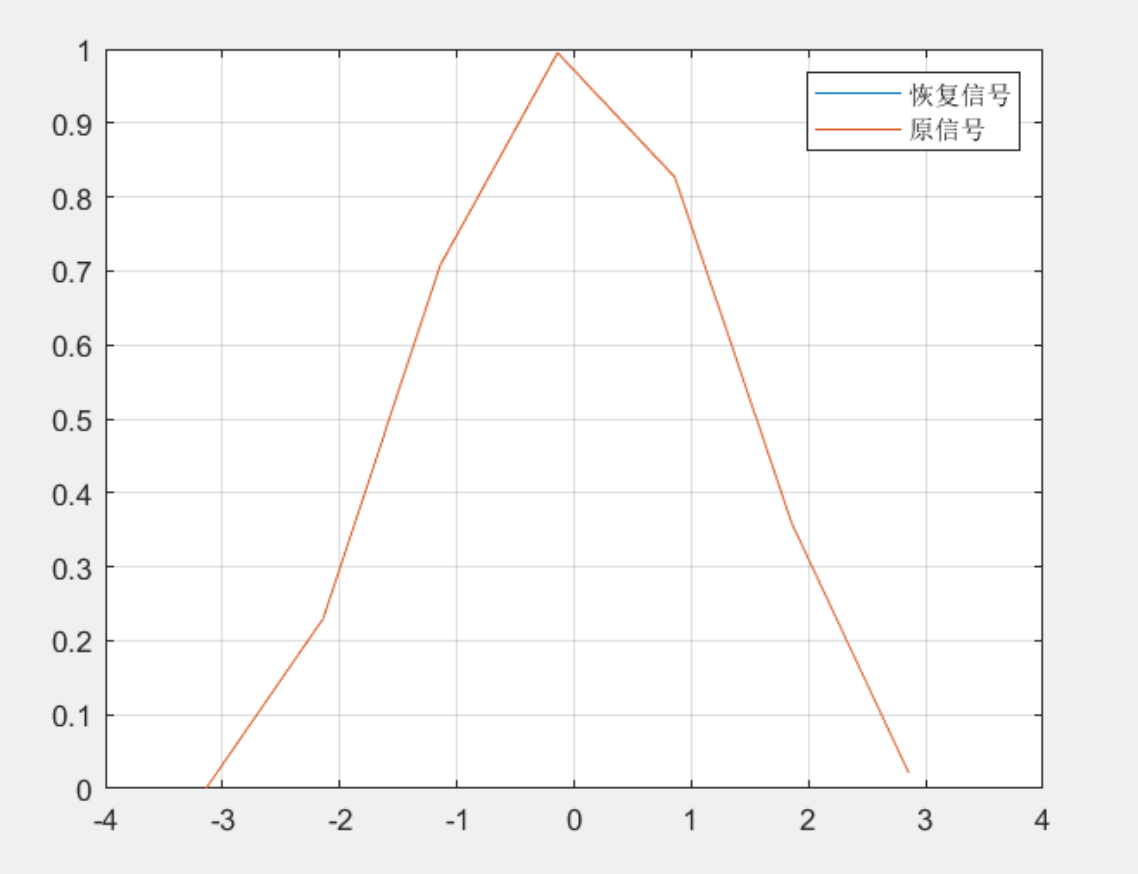
2.2

对实验1中的信号，绘制抽样间隔为1s下的抽样信号经过DAC一阶保持器后的恢复信号时域波形，体会一阶保持器的基本原理和作用。

- 源代码

```
%% 2.2
T_s = 1;
t = -pi:T_s:pi;
% 定义原始信号
f_t = 0.5 * (1 + cos(t));
% 定义保持器的参数 a 和 b
a = 1;
b = [1 0];
% 使用 filter 函数实现一阶保持器
y = filter(b, a, f_t);
% 绘制时域波形
figure;
plot(t,y,t,f_t);
legend('恢复信号','原信号')
axis([-4 4 0 1]);
title('经过 DAC 一阶保持器后的恢复信号时域波形')
```

- 结果图



- 一阶保持器的基本原理和作用

  保持器是一种电子元件或电路，它能够保持或记忆其状态，直到受到外部信号的触发而改变。

  基本原理：一节保持器可以通过传统的数位类比转换器（DAC）及称为积分器的模拟电路完成。

  作用：

  1. **存储信息**： 主要作用是存储二进制信息，因此常用于存储电子存储器中的位信息。
  2. **时序逻辑**： 用于时序逻辑电路，例如时序逻辑电路中的寄存器。
  3. **触发器**： 保持器常被用作触发器的基本构建块。在数字电路中，触发器用于存储和传递信息。
  4. **控制开关**： 保持器可用于控制开关，保持某些设备或电路的状态。

五、实验体会和感悟

在本次实验中，我遇到了一些挑战，但通过查阅资料和尝试不同的方法，我成功地克服了这些困难。特别是对于DAC一阶保持器的理解，我深入学习了相关资料，这让我对这个概念有了更清晰的认识。

另外，关于恢复信号的表示，我一开始采用了循环语句的方法，但在进一步研究资料时发现了更高效的矩阵向量乘法运算的方式。这个发现让我认识到在编程中选择合适的数据结构和算法是提高效率的关键。

通过这次实验，我不仅深化了对抽样定理的理解，还提升了在MATLAB中编程的技能。我学到了信号处理中的一些实用技巧，也对信号抽样和恢复的原理有了更直观和全面的认识。这次实验为我提供了一个学习和成长的机会，使我更加熟练地运用所学知识解决实际问题。