实验三 使用快速傅里叶变换进行频谱分析

MATLAB

1.

代码：

```
clear;
clc;
%1.三角波
figure (1)%创建画布
figure (2)%创建画布
fs=1000;
x=zeros(1,9);
for i=0:8 %获得三角波序列
if i<=4
x(i+1)=i+1;
end
if i>=5
x(i+1)=9-i;
end
end
n=0:1:8; %画三角波序列
figure(1);
subplot(3,1,1);
stem(n,x);
title('1.(1) 三角波序列');
xlabel('n');
ylabel('x1[n]');
figure(2);
subplot(2,3,1);
stem(n,x);
title('1.(1) 三角波序列');
xlabel('n');
ylabel('x1[n]');
%频谱用模拟频率做横轴
N=10000;
xk=fft(x,N); %计算 10000 点 fft
o=0:fs/N:fs-fs/N;
figure(1);
subplot(3,1,2);
plot(o,abs(xk)); %画 10000 点 fft 幅度谱
xlabel('f/Hz (模拟频率)');
```

```
ylabel('10000 点 FFT 频谱幅度');
title('1.(1) N=10000 三角波幅度谱');
figure(1);
subplot(3,1,3); %画 10000 点 fft 相位谱
plot(o,angle(xk));
xlabel('f/Hz (模拟频率)');
ylabel('\theta/rad');
title('1.(1) N=10000 三角波相位谱');

%频谱用数字角频率做横轴
o=0:2*pi/N:2*pi-2*pi/N;
figure(2);
subplot(2,3,2); %画 10000 点 fft 幅度谱(连续)
plot(o,abs(xk));
xlabel('\Omega/rad (数字角频率)');
ylabel('10000 点 FFT 频谱幅度');
xlim([0 2*pi]);
title('1.(1) N=10000 三角波幅度谱(连续)');
subplot(2,3,3); %画 10000 点 fft 幅度谱(离散)
stem(o,abs(xk));
xlabel('\Omega/rad (数字角频率)');
ylabel('10000 点 FFT 幅度频谱');
xlim([0 2*pi]);
title('1.(1) N=10000 三角波幅度谱(离散)');
figure(2);
subplot(2,3,5); %画 10000 点 fft 相位谱(连续)
plot(o,angle(xk));
xlabel('\Omega/rad (数字角频率)');
ylabel('\theta/rad');
xlim([0 2*pi]);
title('1.N=10000 三角波相位谱(连续)');
figure(2);
subplot(2,3,6); %画 10000 点 fft 相位谱(离散)
stem(o,angle(xk));
xlabel('\Omega/rad (数字角频率)');
ylabel('\theta/rad');
xlim([0 2*pi]);
title('1.N=10000 三角波相位谱(离散)');

%1.方波
figure(3);%创建画布
figure(4);%创建画布

x=ones(1,7);
```

```
x=[x 0 0]; %创建方波序列
n=0:1:8;
figure(3); %画方波序列
subplot(3,1,1);
stem(n,x);
title('1.(2)方波序列');
ylim([0,1.5]);
xlabel('n');
ylabel('x2[n]');

figure(4); %画方波序列
subplot(2,3,1);
stem(n,x);
title('1.(2)方波序列');
ylim([0,1.5]);
xlabel('n');
ylabel('x2[n]');

N=10000;
xk=fft(x,N); %计算 10000 点 FFT
%横轴用模拟频率
o=0:fs/N:fs-fs/N;
figure(3);
subplot(3,1,2);
plot(o,abs(xk)); %画 FFT 幅度谱
xlabel('f/Hz （模拟频率)');
ylabel('10000 点 FFT 频谱幅度');
title('1.(2) 方波幅度谱');
figure(3);
subplot(3,1,3); %画 FFT 相位谱
plot(o,angle(xk));
xlabel('f/Hz （模拟频率)');
ylabel('\theta/rad');
title('1.(2) 方波相位谱');


%横轴用数字频率
o=0:2*pi/N:2*pi-2*pi/N;
figure(4);
subplot(2,3,2);
plot(o,abs(xk)); %画 FFT 幅度谱 (连续)
xlabel('\Omega/rad （数字角频率)');
ylabel('10000 点 FFT 频谱幅度');
title('1.(2) 方波幅度谱(连续)');
```

```
xlim([0 2*pi]);
figure(4);
subplot(2,3,5); %画 FFT 相位谱(连续)
plot(o,angle(xk));
xlabel('\Omega/rad (模拟频率)');
ylabel('\theta/rad');
title('1.(2) 方波相位谱(连续)');
xlim([0 2*pi]);
figure(4);
subplot(2,3,3);
stem(o,abs(xk)); %画 FFT 幅度谱 (离散)
xlabel('\Omega/rad (数字角频率)');
ylabel('10000 点 FFT 频谱幅度');
title('1.(2) 方波幅度谱(离散)');
xlim([0 2*pi]);
figure(4);
subplot(2,3,6); %画 FFT 相位谱(离散)
stem(o,angle(xk));
xlabel('\Omega/rad (模拟频率)');
ylabel('\theta/rad');
title('1.(2) 方波相位谱(离散)');
xlim([0 2*pi]);
```

运行截图：

(1) 三角波序列

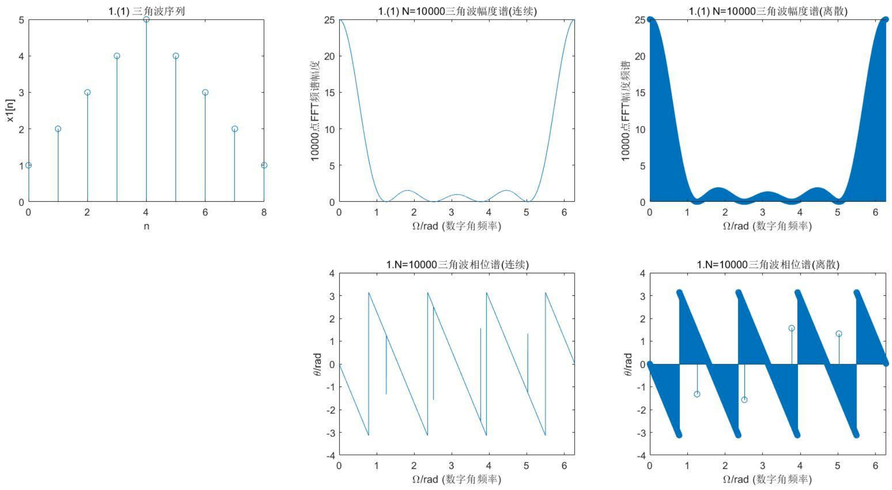
第一列为 x1[n]

第二列第一行为 10000 点 FFT 以数字角频率为横轴通过插值得到的连续幅度谱

第二列第 2 行为 10000 点 FFT 以数字角频率为横轴通过插值得到的连续相位谱

第 3 列第一行为 10000 点 FFT 以数字角频率为横轴的离散幅度谱

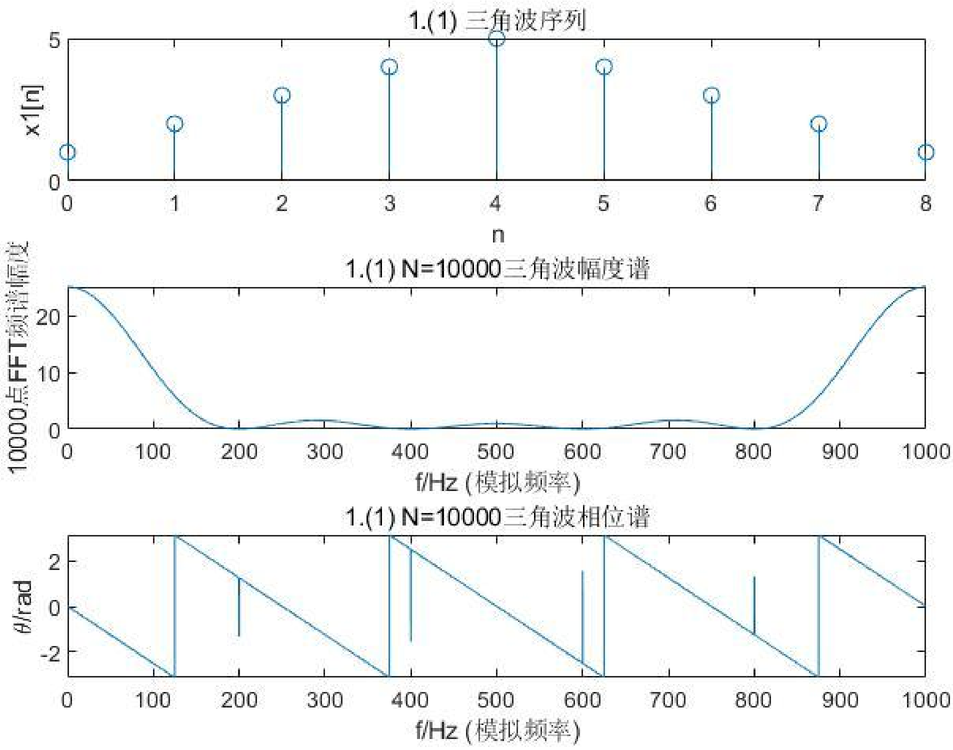
第 3 列第 2 行为 10000 点 FFT 以数字角频率为横轴的离散相位谱



第一行为 x1[n]

第二行为以模拟频率为横轴的 10000 点 FFT 幅度谱

第二行为以模拟频率为横轴的 10000 点 FFT 相位谱



(2) 方波

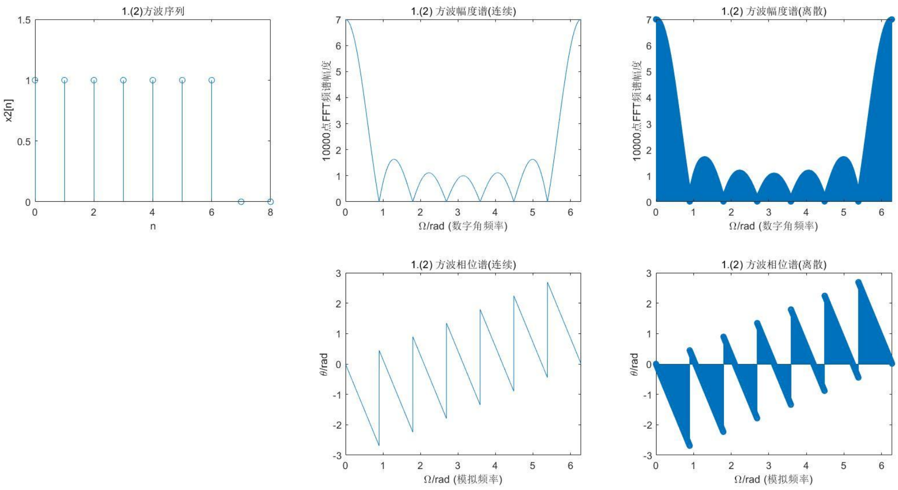
第一列为 x2[n]

第二列第一行为 10000 点 FFT 以数字角频率为横轴通过插值得到的连续幅度谱

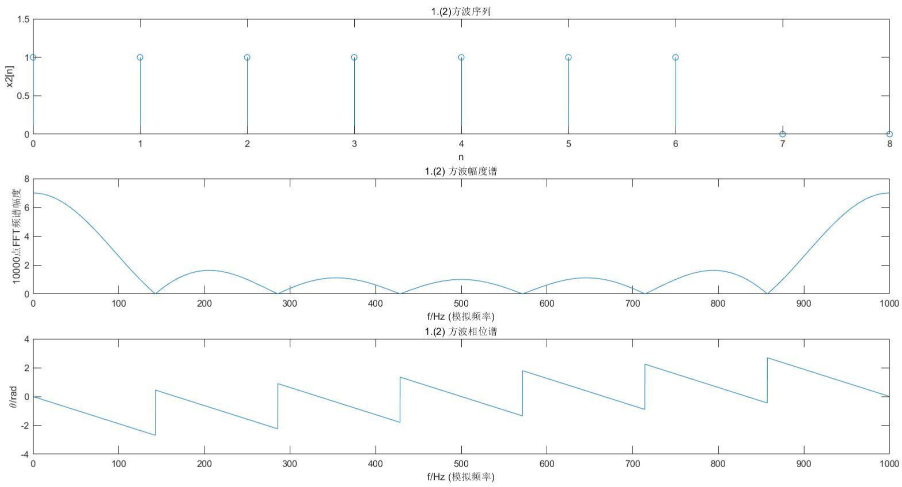
第二列第 2 行为 10000 点 FFT 以数字角频率为横轴通过插值得到的连续相位谱

第 3 列第一行为 10000 点 FFT 以数字角频率为横轴的离散幅度谱

第 3 列第 2 行为 10000 点 FFT 以数字角频率为横轴的离散相位谱



第一行为 x2[n]
第二行为以模拟频率为横轴的 10000 点 FFT 幅度谱
第二行为以模拟频率为横轴的 10000 点 FFT 相位谱



分析
注意到方波序列 FFT 相位谱出现不连续点，且此时幅度谱为 0，这些不连续点是由于非常小的复数在求相角时的计算误差引起的
同时 FFT 得到的频谱和对三角波，矩形波信号以 f=1000Hz 采样得到的混叠频谱一致，二者只在幅度上差一个常数，可见在不满足采样定理时采样得到的频谱会发生频谱混叠

2.
代码：
```
clear;
clc;
%2(1)
N=20000;
fs=1000;
x_1=zeros(1,15);
```

```
x_2=zeros(1,15);
for i=0:1:9 %计算 x1[n]
if i<=4
x_1(i+1)=i+1;
end
if i>=5
x_1(i+1)=10-i;
end
end
for i=0:1:11 %计算 x2[n]
if i<=5
x_2(i+1)=2^i;
end
if i>=6
x_2(i+1)=-2^(i-6);
end
end
g=conv(x_1,x_2); %计算两序列线性卷积
gc=cconv(x_1,x_2,12); %计算两序列长度为 12 的圆周卷积
gc1=cconv(x_1,x_2,21); %计算两序列长度为 21 圆周卷积

x_1k=fft(x_1,N);
x_2k=fft(x_2,N); %计算两序列 FFT
x_k=x_1k.*x_2k; %计算两序列 FFT 相乘
g_k=fft(g,N); %计算线性卷积序列 FFT
gc_k=fft(gc,N); %计算长度为 12 的圆周卷积序列 FFT
gc_1k=fft(gc1,N); %计算长度为 21 的圆周卷积序列 FFT
g_ifft=ifft(x_k); %计算两序列 FFT 相乘后的 IFFT 序列

figure; %创建图层
n=0:1:14;
subplot(2,3,1); %绘制 x1[n]序列
stem(n,x_1);
title("2.(1) x1[n]");
ylabel('x1[n]');
xlabel('n');

subplot(2,3,4);
stem(n,x_2);
title("2.(1) x2[n]"); %绘制 x2[n]序列
ylabel('x2[n]');
xlabel('n');

subplot(2,3,2);
```

```
stem(0:length(g)-1,g);
title("2.(1) x1[n]线性卷积 x2[n]"); %绘制线性卷积序列
ylabel('y[n]');
xlabel('n');

subplot(2,3,5);
stem(0:length(gc)-1,gc);
title("2.(1) x1[n]与 x2[n]长度为 12 的圆周卷积"); %绘制两序列长度为 12 的圆周卷积
ylabel('y1[n]');
xlabel('n');

subplot(2,3,6);
stem(0:length(gc1)-1,gc1);
title("2.(1) x1[n]与 x2[n]长度为 21 的圆周卷积"); %绘制两序列长度为 21 的圆周卷积
ylabel('y2[n]');
xlabel('n');

subplot(2,3,3);
stem(0:length(g)-1,g_ifft(1:length(g)));
title("2.(1) IFFT(FFT(x1[n]).FFT(x2[n])序列"); %绘制两序列 FFT 相乘后的 IFFT 序
列
ylabel('y3[n]');
xlabel('n');

%频谱绘制
figure %创建画布

n=0:1:14;
subplot(3,3,1); %绘制 x1[n]序列
stem(n,x_1);
title("2.(1) x1[n]");
ylabel('x1[n]');
xlabel('n');

subplot(3,3,4);
stem(n,x_2);
title("2.(1) x2[n]"); %绘制 x2[n]序列
ylabel('x2[n]');
xlabel('n');

subplot(3,3,2);
o=0:2*pi/N:2*pi-2*pi/N;
plot(o,abs(x_1k));
title("2.(1) FFT(x1[n])幅度谱"); %绘制 FFT(x1[n])幅度谱
```

```
ylabel('|X1(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,5);
plot(o,abs(x_2k));
title("2.(1) FFT(x2[n])幅度谱"); %绘制 FFT(x2[n])幅度谱
ylabel('|X2(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,8);
plot(o,abs(x_k));
title("2.(1) FFT(x1[n]).FFT(x2[n])幅度谱"); %绘制 FFT(x1[n]).FFT(x2[n])幅度谱
ylabel('|X2(\Omega).X1(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,7);
plot(o,abs(g_k));
title("2.(1) FFT(x1[n]线性卷积 x2[n])幅度谱"); %绘制 FFT(x1[n]线性卷积 x2[n])幅
度谱
ylabel('|Y(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,3);
plot(o,abs(gc_k));
title("2.(1) FFT(x1[n]与 x2[n]长度为 12 的圆周卷积)幅度谱"); %绘制 FFT(x1[n]与 x2[n]
长度为 12 的圆周卷积)幅度谱
ylabel('|Y1(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,6);
plot(o,abs(gc_1k));
title("2.(1) FFT(x1[n]与 x2[n]长度为 21 的圆周卷积)幅度谱"); %绘制 FFT(x1[n]与 x2[n]
长度为 21 的圆周卷积)幅度谱
ylabel('|Y2(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

%(2)
N=20000;
```

```
x_3=zeros(1,15);
x_4=zeros(1,15);
for i=0:1:11 %计算 x3[n]
if i<=6
x_3(i+1)=0.8^i;
end
if i>=7
x_3(i+1)=i-3;
end
end
for i=0:1:12 %计算 x4[n]
if i<=4
x_4(i+1)=i-1;
end
if i>=5
x_4(i+1)=-0.6^(i-6);
end
end
g=conv(x_3,x_4); %计算两序列线性卷积
gc=cconv(x_3,x_4,15); %计算两序列长度为 15 的圆周卷积
gc1=cconv(x_3,x_4,24); %计算两序列长度为 24 圆周卷积

x_3k=fft(x_3,N);
x_4k=fft(x_4,N); %计算两序列 FFT
x_k=x_3k.*x_4k; %计算两序列 FFT 相乘
g_k=fft(g,N); %计算线性卷积序列 FFT
gc_k=fft(gc,N); %计算长度为 15 的圆周卷积序列 FFT
gc_1k=fft(gc1,N); %计算长度为 24 的圆周卷积序列 FFT
g_ifft=ifft(x_k); %计算两序列 FFT 相乘后的 IFFT 序列

figure; %创建图层
n=0:1:14;
subplot(2,3,1); %绘制 x3[n]序列
stem(n,x_3);
title("2.(1) x3[n]");
ylabel('x3[n]');
xlabel('n');

subplot(2,3,4);
stem(n,x_4);
title("2.(1) x4[n]"); %绘制 x4[n]序列
ylabel('x4[n]');
xlabel('n');
```

```
subplot(2,3,2);
stem(0:length(g)-1,g);
title("2.(1) x3[n]线性卷积 x4[n]"); %绘制线性卷积序列
ylabel('y[n]');
xlabel('n');

subplot(2,3,5);
stem(0:length(gc)-1,gc);
title("2.(1) x3[n]与 x4[n]长度为 15 的圆周卷积"); %绘制两序列长度为 15 的圆周卷积
ylabel('y1[n]');
xlabel('n');

subplot(2,3,6);
stem(0:length(gc1)-1,gc1);
title("2.(1) x3[n]与 x4[n]长度为 24 的圆周卷积"); %绘制两序列长度为 24 的圆周卷积
ylabel('y2[n]');
xlabel('n');

subplot(2,3,3);
stem(0:length(g)-1,g_ifft(1:length(g)));
title("2.(1) IFFT(FFT(x3[n]).FFT(x4[n])序列"); %绘制两序列 FFT 相乘后的 IFFT 序
列
ylabel('y3[n]');
xlabel('n');

%频谱绘制
figure %创建画布

n=0:1:14;
subplot(3,3,1); %绘制 x3[n]序列
stem(n,x_3);
title("2.(1) x3[n]");
ylabel('x3[n]');
xlabel('n');

subplot(3,3,4);
stem(n,x_4);
title("2.(1) x4[n]"); %绘制 x4[n]序列
ylabel('x4[n]');
xlabel('n');

subplot(3,3,2);
o=0:2*pi/N:2*pi-2*pi/N;
plot(o,abs(x_3k));
```

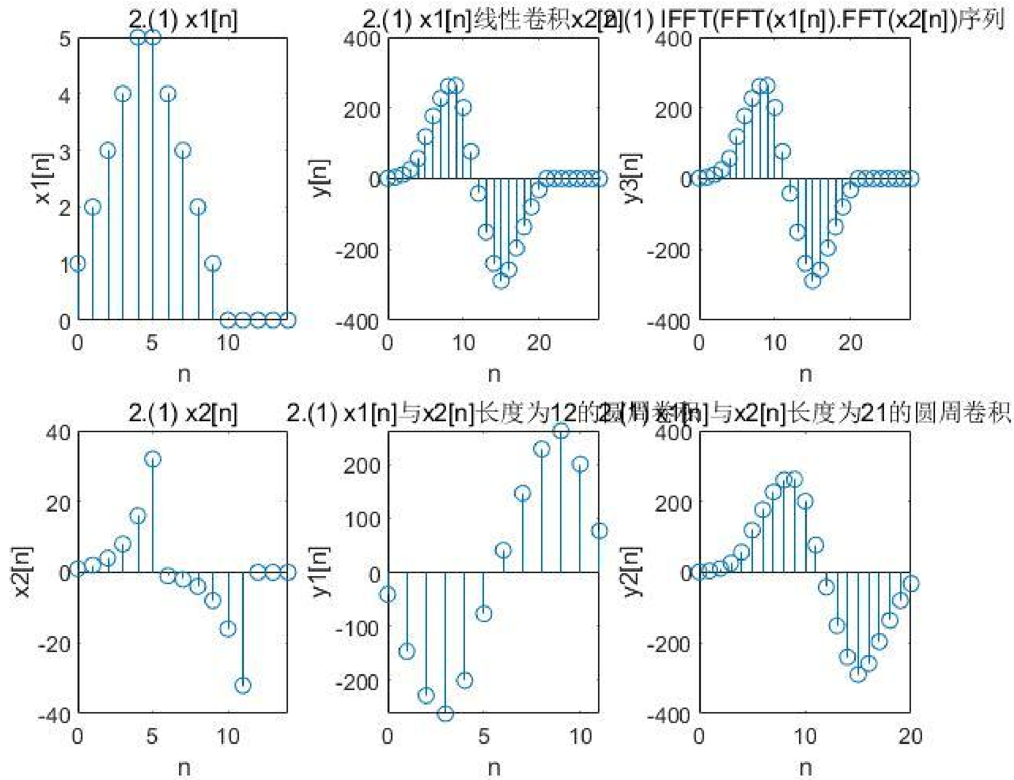
```
title("2.(1) FFT(x3[n])幅度谱"); %绘制 FFT(x3[n])幅度谱
ylabel('|X3(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,5);
plot(o,abs(x_4k));
title("2.(1) FFT(x4[n])幅度谱"); %绘制 FFT(x4[n])幅度谱
ylabel('|X4(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,8);
plot(o,abs(x_k));
title("2.(1) FFT(x3[n]).FFT(x4[n])幅度谱"); %绘制 FFT(x3[n]).FFT(x4[n])幅度谱
ylabel('|X4(\Omega).X3(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,7);
plot(o,abs(g_k));
title("2.(1) FFT(x3[n]线性卷积 x4[n])幅度谱"); %绘制 FFT(x3[n]线性卷积 x4[n])幅度谱
ylabel('|Y(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,3);
plot(o,abs(gc_k));
title("2.(1) FFT(x3[n]与x4[n]长度为15的圆周卷积)幅度谱"); %绘制FFT(x3[n]与x4[n]长度为15的圆周卷积)幅度谱
ylabel('|Y1(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);

subplot(3,3,6);
plot(o,abs(gc_1k));
title("2.(1) FFT(x3[n]与x4[n]长度为24的圆周卷积)幅度谱"); %绘制FFT(x3[n]与x4[n]长度为24的圆周卷积)幅度谱
ylabel('|Y2(\Omega)|');
xlabel('\Omega/rad');
xlim([0 2*pi]);
```

运行结果：

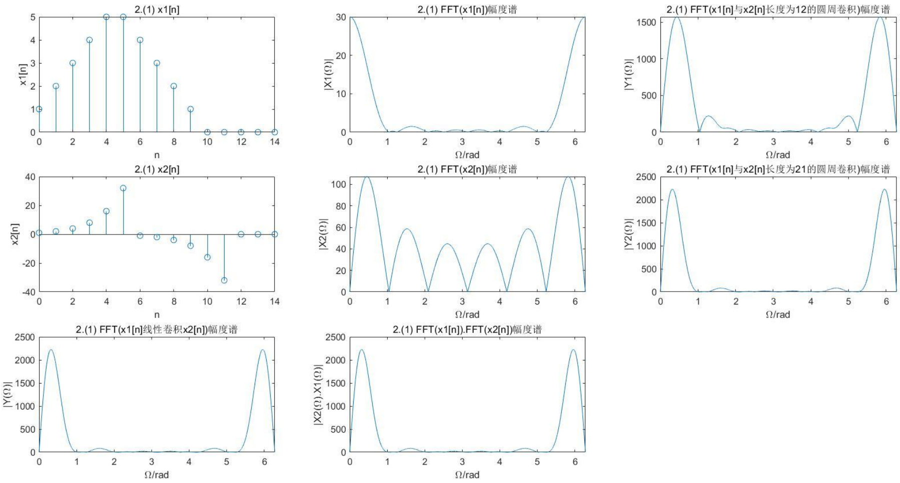
(1)：

第一列第一行为 x1[n] 第二行为 x2[n]
第 2 列第一行为 x1[n] 与 x2[n]通过函数 conv 线性卷积得到的序列，第二行为 x1[n] 与 x2[n] 通过函数 cconv 得到的长度为 12 的圆周卷积序列序列

第 3 列第一行为 x1[n] , x2[n]两序列 FFT 相乘后的 IFFT 序列
第 3 列第二行为 x1[n] 与 x2[n]通过函数 cconv 得到的长度为 21 的圆周卷积序列序列



频谱：
第一行第一列为 x1[n]序列，第二列为 FFT(x1[n])的幅度谱，第三列为 x1[n]与 x2[n]长度为 12 的圆周卷积序列 FFT 的幅度谱
第 2 行第一列为 x2[n]序列，第二列为 FFT(x1[n])的幅度谱，第三列为 x1[n]与 x2[n]长度为 21 的圆周卷积序列 FFT 的幅度谱
第三行第一列为 FFT(x1[n]线性卷积 x2[n])幅度谱 ，第二列为 x1[n]的 FFT 序列乘 x2[n]的 FFT 序列得到的幅度谱



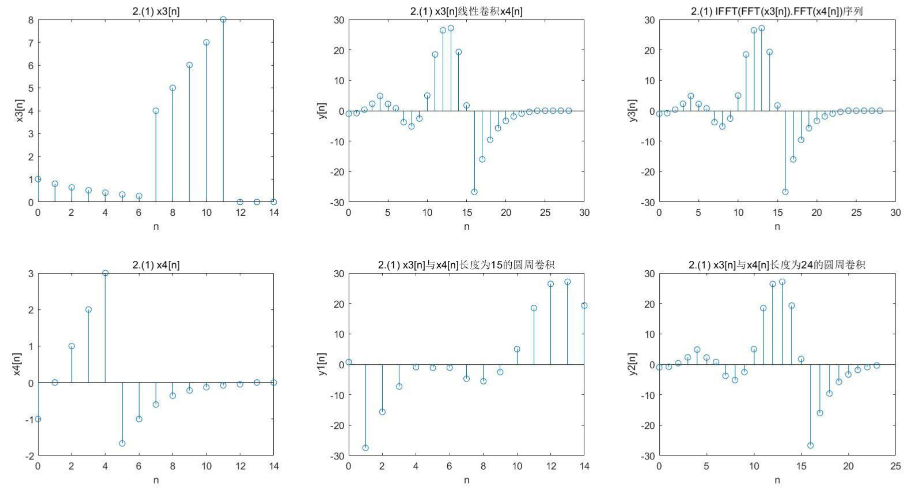
(2)

第一列第一行为 x3[n] 第二行为 x4[n]

第 2 列第一行为 x3[n] 与 x4[n]通过函数 conv 线性卷积得到的序列，第二行为 x3[n] 与 x4[n]通过函数 cconv 得到的长度为 15 的圆周卷积序列序列

第 3 列第一行为 x3[n] , x4[n]两序列 FFT 相乘后的 IFFT 序列

第 3 列第二行为 x3[n] 与 x4[n]通过函数 cconv 得到的长度为 24 的圆周卷积序列序列

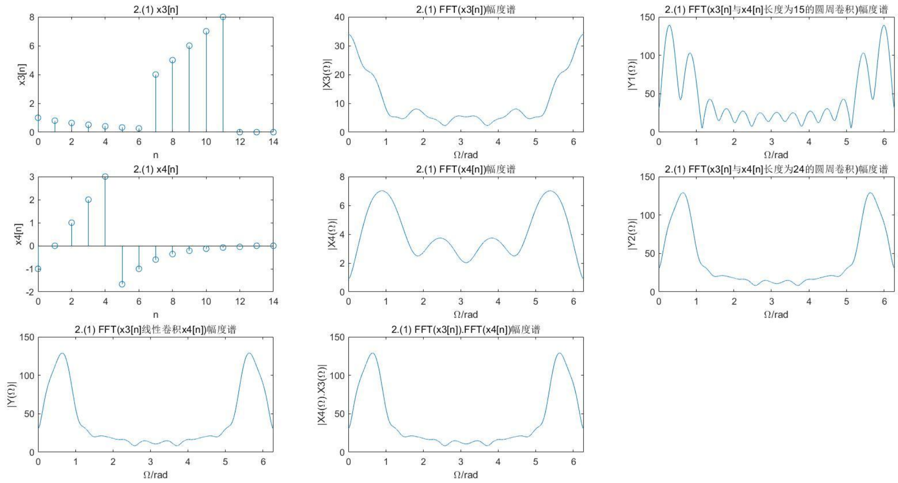


频谱：

第一行第一列为 x3[n]序列，第二列为 FFT(x3[n])的幅度谱，第三列为 x3[n]与 x4[n]长度为 15 的圆周卷积序列 FFT 的幅度谱

第 2 行第一列为 x4[n]序列，第二列为 FFT(x3[n])的幅度谱，第三列为 x3[n]与 x4[n]长度为 24 的圆周卷积序列 FFT 的幅度谱

第三行第一列为 FFT(x3[n]线性卷积 x4[n])幅度谱 ，第二列为 x3[n]的 FFT 序列乘 x4[n]的 FFT 序列得到的幅度谱



分析：

N=20000 点的 FFT，由两个序列的 FFT 序列相乘 和两个序列线性卷积的 FFT 序列 频谱相同
同时两个序列线性卷积的序列和 由两个序列的 FFT 序列相乘之后 IFFT 得到的序列相同
当圆周卷积长度 L 大于等于两个序列长度之和减一时，圆周卷积与两序列的线性卷积相同，
当圆周卷积长度 L 大于等于两个序列长度之和减一时，同时圆周卷积的序列的 FFT 频谱和两个序列的 FFT 序列相乘 和两个序列线性卷积的 FFT 序列 频谱相同

3.

代码：

```
clear;
clc;
%3.(1)
f1=5;
f2=15;
f3=40;
x1=zeros(1,128);
x2=zeros(1,128);
x3=zeros(1,128);
f_1=5;
f_2=9;
for i=0:1:127 %不同频率采样
x1(i+1)=sin(2*pi*f_1*i/f1)+sin(2*pi*f_2*i/f1);
x2(i+1)=sin(2*pi*f_1*i/f2)+sin(2*pi*f_2*i/f2);
x3(i+1)=sin(2*pi*f_1*i/f3)+sin(2*pi*f_2*i/f3);
end

x1_k=fft(x1,128);
x2_k=fft(x2,128);
x3_k=fft(x3,128); %计算 128 点 FFT

figure; %创建画布
```

```
%画时域采样序列
subplot(3,3,1); %画 x1(n)
N=0:1:127;
stem(N,x1);
title('3.(1) f1=5Hz128 点采样序列');
xlabel('n');
ylabel('x1(n)')

subplot(3,3,4); %画 x2(n)
stem(N,x2);
title('3.(1) f2=15Hz128 点采样序列');
xlabel('n');
ylabel('x2(n)')


subplot(3,3,7); %画 x3(n)
stem(N,x3);
title('3.(1) f3=40Hz128 点采样序列');
xlabel('n');
ylabel('x3(n)')
%画 FFT 频谱 (连续)
o=0:2*pi/128:2*pi-2*pi/128; %画 x1(n)的 FFT 幅度谱
subplot(3,3,2);
plot(o,abs(x1_k));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f1=5Hz128 点 FFT 幅度谱');

subplot(3,3,5); %画 x2(n)的 FFT 幅度谱
plot(o,abs(x2_k));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f2=15Hz128 点 FFT 幅度谱');

subplot(3,3,8); %画 x3(n)的 FFT 幅度谱
plot(o,abs(x3_k));
xlabel('\Omega/rad');
ylabel('|X3(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f3=40Hz128 点 FFT 幅度谱');

%画 FFT 频谱 (离散)
```

```
subplot(3,3,3); %画 x1(n)的 FFT 幅度谱
stem(o,abs(x1_k));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f1=5Hz128 点 FFT 幅度谱序列');

subplot(3,3,6); %画 x2(n)的 FFT 幅度谱
stem(o,abs(x2_k));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f2=15Hz128 点 FFT 幅度谱序列');

subplot(3,3,9); %画 x3(n)的 FFT 幅度谱
stem(o,abs(x3_k));
xlabel('\Omega/rad');
ylabel('|X3(\Omega)|');
xlim([0 2*pi]);
title('3.(1) f3=40Hz128 点 FFT 幅度谱序列');

%3.(2)
figure;%创建画布
f=60;
x=zeros(1,64);
for i=0:1:63 %采样
x(i+1)=sin(2*pi*f_1*i/f)+sin(2*pi*f_2*i/f);
end
N=0:1:63;
subplot(2,3,1);
stem(N,x);
xlabel('n');
ylabel('x1(n)');
title('3.(2) f=60Hz 64 点采样序列');
x_=zeros(1,64);
y=[x x_]; %补零
N=0:1:127;
subplot(2,3,4);
stem(N,y);
title('3.(2) f=60Hz 64 点采样后补零 64 点的 128 点序列');
xlabel('n');
ylabel('x2(n)');
%计算 FFT
xk=fft(x,64);
```

```
yk=fft(y,128);
%绘制 64 点采样序列的 fft 幅度谱
subplot(2,3,2);
o=0:2*pi/64:2*pi-2*pi/64;
plot(o,abs(xk));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(2) f=60Hz 64 点采样序列的 FFT 幅度谱');

subplot(2,3,3);
o=0:2*pi/64:2*pi-2*pi/64;
stem(o,abs(xk));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(2) f=60Hz 64 点采样序列的 FFT 幅度谱序列');
%绘制 64 点采样后补零 64 点的 128 点序列的 fft 幅度谱
subplot(2,3,5);
o=0:2*pi/128:2*pi-2*pi/128;
plot(o,abs(yk));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(2) f=60Hz 64 点采样后补零 64 点的 128 点序列的 FFT 幅度谱');

subplot(2,3,6);
o=0:2*pi/128:2*pi-2*pi/128;
stem(o,abs(yk));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(2) f=60Hz 64 点采样后补零 64 点的 128 点序列的 FFT 幅度谱序列');


%3.(3)
figure;%创建画布
f=60;
x=zeros(1,128);
for i=0:1:127
x(i+1)=sin(2*pi*f_1*i/f)+sin(2*pi*f_2*i/f);
end
N=0:1:127;
subplot(2,3,1); %画时域采样图
```

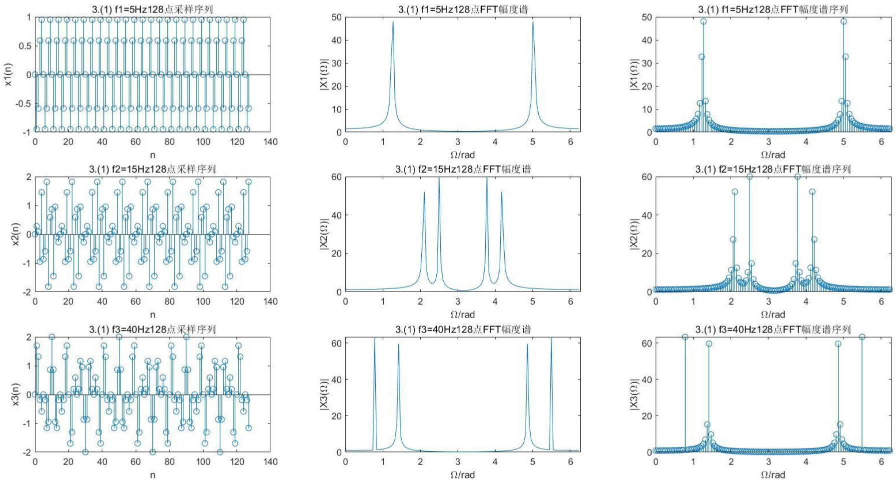
```
stem(N,x);
xlabel('n');
ylabel('x1(n)');
title('3.(3) f=60Hz 128 点采样序列');

subplot(2,3,4); %画时域采样图
stem(N,y);
xlabel('n');
ylabel('x2(n)');
title('3.(3) f=60Hz 64 点采样后补零 64 点的 128 点序列');
%画 FFT 频谱
xk=fft(x,128);
subplot(2,3,2);
o=0:2*pi/128:2*pi-2*pi/128;
plot(o,abs(xk));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(3) f=60Hz 128 点采样序列的 FFT 幅度谱');

subplot(2,3,3);
stem(o,abs(xk));
xlabel('\Omega/rad');
ylabel('|X1(\Omega)|');
xlim([0 2*pi]);
title('3.(3) f=60Hz 128 点采样序列的 FFT 幅度谱序列');
%绘制 64 点采样后补零 64 点的 128 点序列的 fft 幅度谱
subplot(2,3,5);
plot(o,abs(yk));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(3) f=60Hz 64 点采样后补零 64 点的 128 点序列的 FFT 幅度谱');

subplot(2,3,6);
stem(o,abs(yk));
xlabel('\Omega/rad');
ylabel('|X2(\Omega)|');
xlim([0 2*pi]);
title('3.(3) f=60Hz 64 点采样后补零 64 点的 128 点序列的 FFT 幅度谱序列');
```

运行截图:

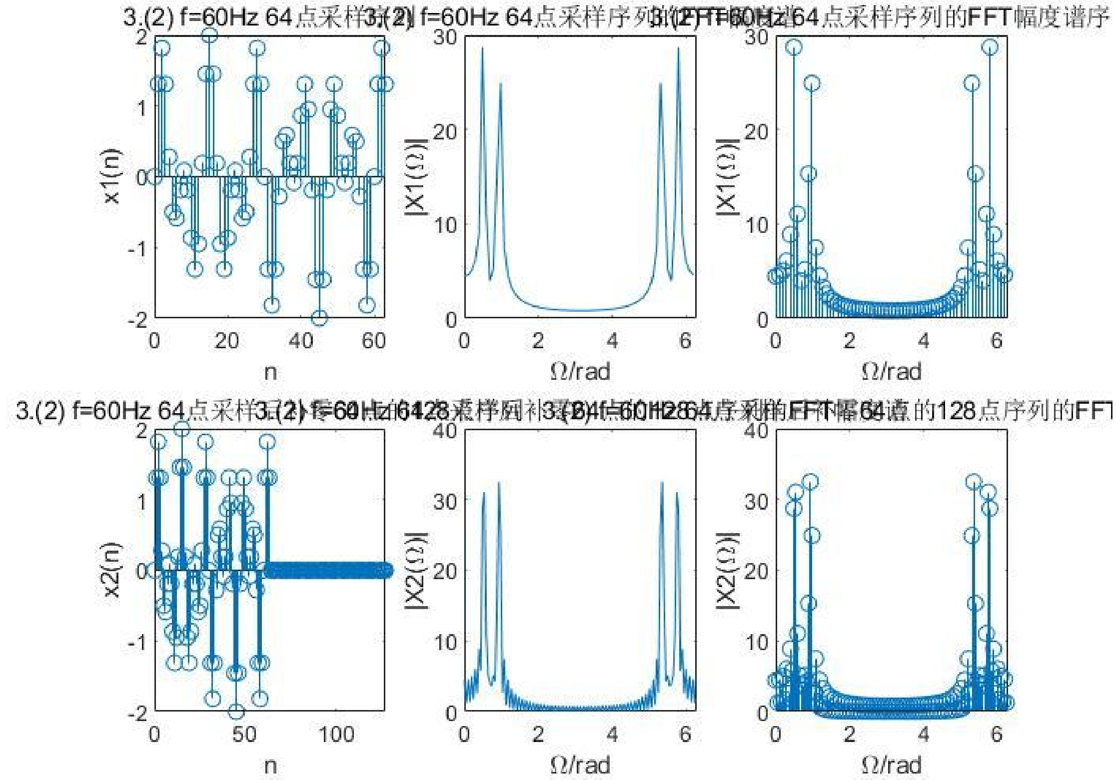
(1)

第一列从上到下依次是 f=5Hz,15Hz,40Hz 的时域采样图

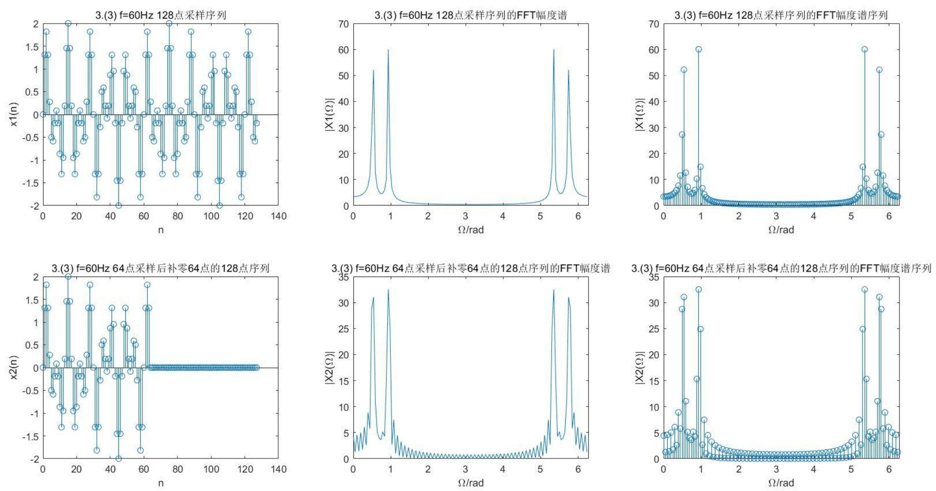
第 2 列从上到下依次是 f=5Hz,15Hz,40Hz 的 128 点 FFT 幅度谱
第 3 列从上到下依次是 f=5Hz,15Hz,40Hz 的 128 点 FFT 幅度谱序列



(2)
第一列从上到下依次是 f=60Hz 的 64 点时域采样图 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的序列图
第 2 列从上到下依次是 f=60Hz 的 64 点 FFT 幅度谱 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的 128 点 FFT 幅度谱
第 3 列从上到下依次是 f=60Hz 的 64 点 FFT 幅度谱序列 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的 128 点 FFT 幅度谱序列



(3)
第一列从上到下依次是 f=60Hz 的 128 点时域采样图 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的序列图
第 2 列从上到下依次是 f=60Hz 的 128 点 FFT 幅度谱 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的 128 点 FFT 幅度谱
第 3 列从上到下依次是 f=60Hz 的 128 点 FFT 幅度谱序列 和 f=60Hz 的 64 点时域采样加末尾补 64 个零的 128 点 FFT 幅度谱序列



对比发现补零后序列的 FFT 幅度谱出现震荡，虽然整体变化特征一致，但是补零序列的 FFT 幅度谱包络线出现锯齿状震荡
由于采样是 60Hz 而对原信号,采样 64 点和 128 点已经发生频谱泄露，对 64 点补零相当于

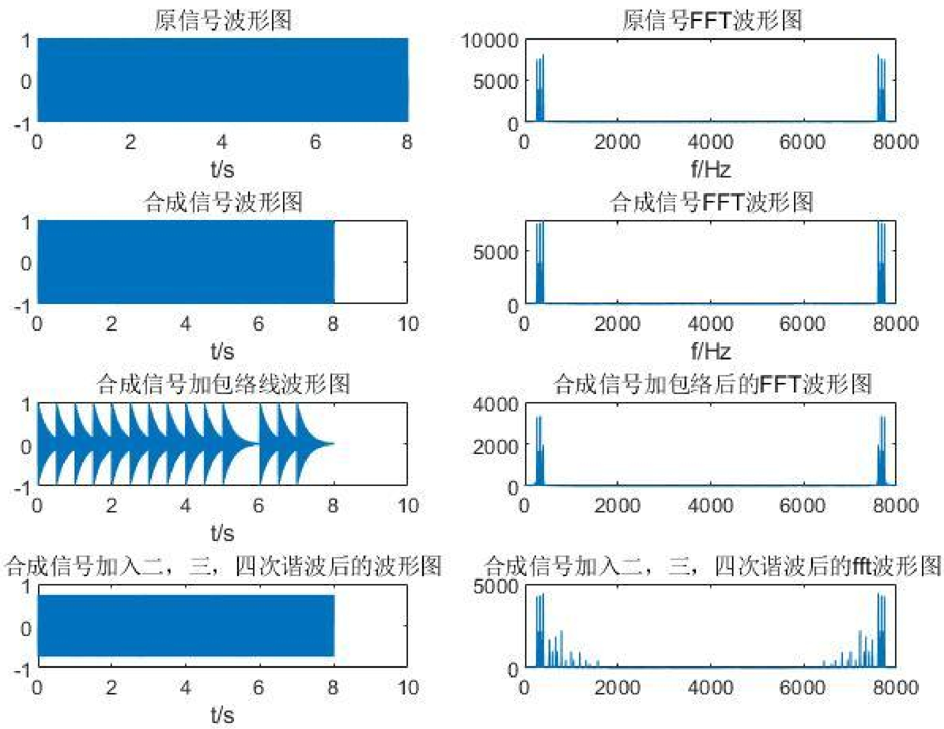
提高 DFT 的分辨率，由于 DTFT 频谱已经泄露，提高 DFT 分辨率让频谱泄露现象更加明显。
这里不止发生了频谱泄露还发生了频谱混叠
混叠和泄露都是对正弦信号而言。

4.
代码：
```
clear
clc
%4
[y,Fs]=audioread('laohu.wav');%读取音频信息
yk=fft(y);%做 FFT
o=0:Fs/length(y):Fs-Fs/length(y);
t=1/Fs:1/Fs:length(y)/Fs;
figure;
subplot(4,2,1);
plot(t,y);
title('原信号波形图');
xlabel('t/s');
subplot(4,2,2);
plot(o,abs(yk));
xlabel('f/Hz');
title('原信号 FFT 波形图');

%合成信号
fs=8000;
T=8;
f1=261.63;
f2=293.66;
f3=329.63;
f4=349.23;
f5=392;
f=[f1 f2 f3 f4 f5];%所需频率数组
a=1;

y=0;%合成信号
y1=0;%合成信号加包络
y3=0;%合成信号加谐波
k=0.7;
t1=0:1/fs:0.5;
t2=0:1/fs:1;
so=[1 2 3 1 1 2 3 1 3 4 5 3 4 5 ];%乐谱
lt=[0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 1 0.5 0.5 1];%节拍持续时间
for i=1:1:14%合成信号
if lt(i)==0.5
```

```
y=[y a*sin(2*pi*f(so(i))*t1)];
y1=[y1 a*sin(2*pi*f(so(i))*t1).*exp(-4*t1)];%加指数包络
y3=[y3
(0.5+1/16)*a*sin(2*pi*f(so(i))*t1)+0.25*a*sin(2*pi*2*f(so(i))*t1)+0.125*a*s
in(2*pi*3*f(so(i))*t1)+0.0625*a*sin(2*pi*4*f(so(i))*t1)];
end
if lt(i)==1
y=[y a*sin(2*pi*f(so(i))*t2)];
y1=[y1 a*sin(2*pi*f(so(i))*t2).*exp(-4*t2)];%加指数包络
y3=[y3
(0.5+1/16)*a*sin(2*pi*f(so(i))*t2)+0.25*a*sin(2*pi*2*f(so(i))*t2)+0.125*a*s
in(2*pi*3*f(so(i))*t2)+0.0625*a*sin(2*pi*4*f(so(i))*t2)];
end
end
%输出音频文件
audiowrite("5-3.wav",y,fs);
audiowrite("5-4.wav",y1,fs);
audiowrite("5-5.wav",y3,fs);
%画合成信号
t=1/fs:1/fs:length(y)/fs;
subplot(4,2,3);
plot(t,y);
xlabel('t/s');
title('合成信号波形图');
yk=fft(y);
o=0:fs/length(y):fs-fs/length(y);
subplot(4,2,4);
plot(o,abs(yk));
xlabel('f/Hz');
title('合成信号 FFT 波形图');
%画合成信号加入包络后的信号
subplot(4,2,5);
plot(t,y1);
xlabel('t/s');
title('合成信号加包络线波形图');
yk=fft(y1);
o=0:fs/length(y1):fs-fs/length(y1);
subplot(4,2,6);
plot(o,abs(yk));
title('f/Hz');
title('合成信号加包络后的 FFT 波形图');
%画合成信号加入谐波后的信号
subplot(4,2,7);
plot(t,y3);
```

```
xlabel('t/s');
title('合成信号加入二，三，四次谐波后的波形图');
yk=fft(y3);
o=0:fs/length(y3):fs-fs/length(y3);
subplot(4,2,8);
plot(o,abs(yk));
title('f/Hz');
title('合成信号加入二，三，四次谐波后的 fft 波形图');
%播放合成信号加入二，三，四次谐波后的音频
sound(y3);
pause(9);
%播放合成信号加入包络线后的音频
sound(y1);
pause(9);
%播放合成信号音频
sound(y);
```

运行截图：



音频文件在附件

第一列从上到下分别为原信号波形图，合成信号波形图，合成信号加包络的波形图，合成信号加入二三四次谐波后的波形图，右侧为对应的 FFT 幅度谱

实验思考题

1. 阐述线性卷积、圆周卷积和周期卷积的区别和（或）联系。用 FFT 计算线性卷积时， FFT 的长度 N 应满足什么条件？

线性卷积直接使用两个序列进行反向、移位、相乘、求和得到

线性卷积是对两个序列所有点进行计算，周期卷积是对两个周期序列内一个周期求和，线性卷积对序列长度无要求，周期卷积要求两个序列都是周期一致的周期序列

圆周卷积是其中一个序列圆周移位在进行相乘求和运算这个与线性卷积不同。

圆周卷积是对两个序列进行相同周期的周期延拓后进行周期卷积再取主值区间的结果，也就是说周期卷积得到的还是周期序列，圆周卷积是只取主值区间作为结果

用 FFT 计算线性卷积时， FFT 的长度 N 应满足

N>=N1+N2-1,N1,N2 分别为两个序列长度

或者 N>=2*L-1 L 为两个序列周期延拓长度

2. 实数序列的频域幅度谱和相位谱有什么规律？虚数序列的频域幅度谱和相位谱有什么规律？（结合课上所学知识，可以自行设定序列，仿真观察）

实数序列的幅度谱是偶函数，相位谱是奇函数；虚数序列的幅度谱也是偶函数，相位谱也是奇函数；若一个虚数序列是一个实数序列乘 i，两个序列的幅度谱完全相同，相位谱差 pi。

3. 同一连续信号离散化后有两种情况，第一种是取较长的离散序列求 FFT；第二种是取较短的离散序列，结尾补零扩展成与第一种中的长度相等，再求 FFT。在上述两种情况下，信号的频谱有何异同点？

如果两个序列都把时域信号非零区间覆盖，则这两个信号 FFT 频谱一样

如果不是则长须列保留了更多信息，两者包络线形状比较近似，同时长序列的 FFT 幅度比短序列补零后的 FFT 幅度大，由于采样带来频谱混叠，得到序列的 DTFT 本身具有波动，如果补零相当与增加分辨率，得到的频谱波动也会更加明显，短序列的频谱幅度比长序列小，同时有更加明显的波动，二者的包络线形状类似