**实验7 离散信号的DFT&FFT频谱MATLAB仿真**

# 一、实验目的

1、掌握序列傅氏变换的计算机实验方法，利用序列的傅氏变换对离散信号、系统及系统响应进行频域分析。

2、加深对离散信号的DTFT、DFT及其相互关系的理解。

3、在理论学习的基础上，加深对快速傅里叶变换FFT的理解，熟悉FFT算法

4、熟悉应用FFT对典型信号进行频谱分析的方法。

5、加深对离散信号FFT算法的运用，以便在实际中正确应用FFT。

# 二、实验原理与方法

## 1、DFT基础

一个连续信号 的频谱可以用它的傅里叶变换表示为:

|  |  |
| --- | --- |
|  | (7-1) |

如果对该信号进行理想采样， 可以得到采样序列:

|  |  |
| --- | --- |
|  | (7-2) |

同样可以对该序列进行Z变换,其中T为采样周期

|  |  |
| --- | --- |
|  | (7-3) |

当的时候，我们就得到了序列的傅里叶变换(DTFT):

|  |  |
| --- | --- |
|  | (7-4) |

其中ω称为数字频率，它和模拟域频率的关系为:

|  |  |
| --- | --- |
|  | (7-5) |

式中的是采样频率，上式说明数字频率是模拟频率对采样率的归一化。 同模拟域的情况相似，数字频率代表了序列值变化的速率，而序列的傅立叶变换称为序列的频谱。序列的傅立叶变换和对应的采样信号频谱具有下式的对应关系。

|  |  |
| --- | --- |
|  | (7-6) |

即序列的频谱是采样信号频谱的周期延拓。从式（7-6）看出，只要分析采样序列的频谱，就可以得到相应的连续信号的频谱。注意：这里的信号必须是带限信号，采样也必须满足Nyquist定理。

在各种信号序列中，有限长序列在数字信号处理中占有很重要的地位。无限长的序列也往往可以用有限长序列来逼近。对于有限长的序列我们可以使用离散傅立叶变换(DFT),这一变换可以很好地反映序列的频域特性，并且容易利用快速算法FFT在计算机上实现。当序列的长度是N时，我们定义离散傅立叶变换为：

|  |  |
| --- | --- |
|  | (7-7) |

其中，它的反变换定义为：

|  |  |
| --- | --- |
|  | (7-8) |

根据式(7-3)和(7-7)，令，则有

|  |  |
| --- | --- |
|  | (7-9) |

可以得到，是z平面单位圆上幅角为的点，就是将单位圆进行N等分以后第k个点。所以，是z变换在单位圆上的等距采样，或者说是序列傅立叶变换的等距采样。时域采样在满足

Nyquist定理时就不会发生频谱混淆。

若将DFT变换的定义写成矩阵形式(设x，X均为列矩阵)，则得到

|  |  |
| --- | --- |
|  | (7-10) |

其中DFT变换矩阵A为

|  |  |
| --- | --- |
|  | (7-11) |

（1）Dftmtx 函数：用来计算 DFT 变换矩阵 A 的函数

A＝dftmta（n）：返回 n×n 的 DFT 变换矩阵 A。若 x 为给定长度的行向量，则40

y＝x\*A，返回 x 的 DFT 变换 y。

Ai＝conj（dftmtx（n））/n；返回 n×n 的 IDFT 变换矩阵 Ai。

(2)**离散傅里叶变换dft函数(自编函数)**

function Xk=dft(xn,N) %实现离散傅里叶变换（DFT)的计算

n=[0:N-1];

k=n;

Wn=exp(-j\*2\*pi/N);

nk=n'\*k;

Wnnk=Wn.^nk;

xn=[xn zeros(1,N-length(xn))];%对信号进行补零

Xk=xn\*Wnnk;

end

程序中：Wn 为旋转因子；xn代表离散时间序列x(n)；N为离散时间序列x(n)的长度；Xk为离散序列x(n)的傅里叶变换。

(3)**离散傅里叶逆变换（IDFT)(自编函数)**

function xn=idft(Xk,N) %实现离散傅里叶逆变换（IDFT)的计算

n=[0:N-1];

k=[0:N-1];

Wn=exp(-j\*2\*pi/N);

nk=n'\*k;

Wnnk=Wn.^(-nk);

xn=(Xk\*Wnnk)/N;

(4)离散时间傅里叶变换DTFT**(自编函数)**

function [X]=dtft(x,w)

% 计算x序列离散时间傅立叶变换

% [X]=dtft(x,n,w),

% X = 在w频率点上的DTFT数组,

% x = 沿n的有限长度序列,

% n = 样本位置向量

% w = 频率点位置向量

n=1:length(x);

ewn=exp(-n'\*w\*i);

X=x\*ewn;

%调用时可选如下方案1或方案2

%方案1

%w=linspace(-2\*pi,2\*pi,1000)/dt;dt为时间间隔

% w是1000点行向量

% X0 = dtft(x0,w)\*dt;

%方案2

% k=0:511; f=fs\*k/512; %由wk=2πk/512可求得模拟频率f

% Xa=dtft(xa,2\*pi\*k/512); % 近似模拟信号频谱

end

【例7-1】设x(n)是4点序列{1 1 1 1 },调用dft、idft、dft函数实现以下要求：

1)求解序列DTFT，并画出幅频曲线，

2)分别求解序列4点、8点、16点DFT，并绘制幅频曲线，观察与DTFT幅频之间的关系。

程序如下:

clear;

xn=[1 1 1 1];

k=0:511; %由wk=2πk/512可求得模拟频率f

Xw=dtft(xn,2\*pi\*k/512); % 近似模拟信号频谱

subplot(221),plot(2\*pi\*k/512,abs(Xw));title('DTFT幅频'),xlabel('w')

Xk4=dft(xn,4);

subplot(222),stem(abs(Xk4));title('4点DFT幅频')

Xk8=dft(xn,8);

subplot(223),stem(abs(Xk8));title('8点DFT幅频')

Xk16=dft(xn,16);

subplot(224),stem(0:15,abs(Xk16));title('16点DFT幅频')

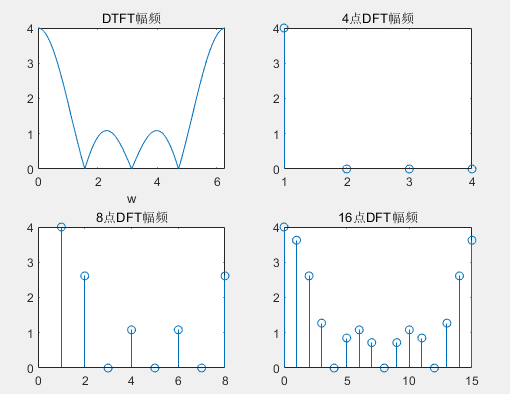


图7-1 例7-1结果图

【例7-2】

已知*x*(*n*)=sin(*n*π/8)+sin(*n*π/4)，用 MATLAB 求解N=8,16,32时 DFT 的结果，并绘制幅频曲线，比较结果。

程序如下(给出N=16,请自行补充N=18,32)：

N=16;

n=0:1:N-1; %时域采样

xn=sin(n\*pi/8)+sin(n\*pi/4); %以下DFT求解也可以调用自编函数dft实现

k=0:1:N-1; %频域采样

WN=exp(-j\*2\*pi/N);

nk=n'\*k;

WNnk=WN.^nk;41

Xk=xn\*WNnk;

subplot(2,1,1)

stem(n,xn);

subplot(2,1,2)

stem(k,abs(Xk))

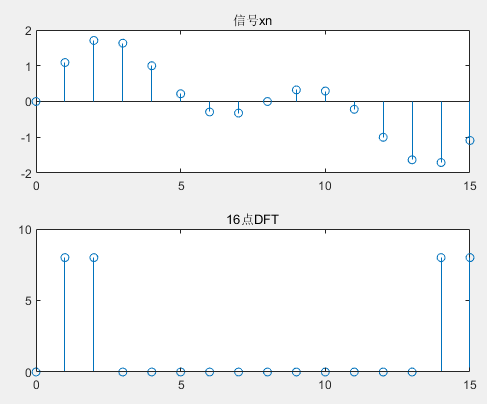


图7-2 例7-2结果图

## 2、DFT性质

DFT的性质：

两个序列和都是N点有限长序列,设

**1）线性**

|  |  |
| --- | --- |
|  | (7-12) |

式中a,b为任意常数。

**2）圆周移位**

一个有限长序列*x*(*n*)的圆周移位定义

|  |  |
| --- | --- |
|  | (7-13) |

式中，表示*x*(*n*)的周期延拓序列的移位

|  |  |
| --- | --- |
|  | (7-14) |

有限长序列圆周移位后的DFT为

|  |  |
| --- | --- |
|  | (7-15) |

[例7-3] 求有限长序列的圆周移位,并画出其结果图。

程序如下：

N=10;

m=4;

n=0:1:N-1;

x=8\*(0.4).^n;

n1=mod((n+m),N);

xm=x(n1+1);

subplot(2,1,1)

stem(n,x);

title('原始序列');

xlabel('n');

ylabel('x(n)');

subplot(2,1,2)

stem(n,xm);

title('圆周移位4位后的序列');

xlabel('n');

ylabel('x((n+4))mod20');

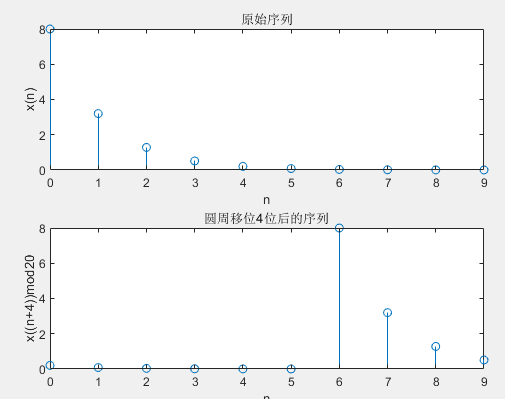


图7-3 例7-3结果图

**3）圆周卷积**

假设,则有

|  |  |
| --- | --- |
|  | (7-16) |

用⊗表示圆周卷积，则上式可化简为

|  |  |
| --- | --- |
|  | (7-17) |

在MATLAB中，只有线卷积函数conv，而不具备圆周移位函数和圆周卷积函数，可以自己编程构造这两个函数，如下所示。

* **圆周移位cirshiftd函数**

function y=cirshiftd(x,m,N) %直接实现序列x的圆周移位

if length (x)>N

error('the length of x must be less than N');

end

x=[x,zeros(1,N-length(x))];

n=[0:1:N-1];

y=x(mod(n-m,N)+1);

end

程序中： x是长度小于N的序列；m是移动的位数；N是圆周长度；y是移位后的输出序列。

* **圆周卷积circonv函数**

function yc=circonv(x1,x2,N)

if length(x1)>N %以下两个if语句判断两个序列的长度是否小于N

error('N must not be less than length of x1');

end

if length(x2)>N

error('N must not be less than length of x2');

end

x1=[x1,zeros(1,N-length(x1))]; %填充序列x1(n)使其长度为N1+N2-1

%（序列x1(n)的长度为N1，序列x2(n)的长度为N2)

x2=[x2,zeros(1,N-length(x2))]; %填充序列x2(n)使其长度为N1+N2-1

n=[0:1:N-1];

x2=x2(mod(-n,N)+1); %生成序列x2((-n))N

H=zeros(N,N);

for n=1:1:N

H(n,:)=cirshiftd(x2,n-1,N); %该矩阵的k行为x2((k-1-n))N

End

yc=x1\*H'; %计算圆周卷积

end

函数中，x1,x2为需要计算圆周卷积的序列；N为圆周卷积的点数。yc为圆周卷积结果。

[实例7-4] **：**已知4点矩形脉冲R4(N)

1）求解其和自己的线卷积与4点圆卷积。

2）在R4(N)后面添加3个零点，将它扩展成长度为7的序列后再计算它和自己的7点圆周卷积。

【解】： clear all;

xn=[1 1 1 1];

y1=conv(xn,xn); %矩形序列与其自身的线卷积

N=length(xn);

XK=dft(xn,N); %圆周卷积定理

YK=XK.\*XK;

yc=idft(YK,N); %用圆周卷积定理实现的4点圆周卷积

xn1=[1 1 1 1 0 0 0];

N1=length(xn1);

XK1=dft(xn1,N1);

YK1=XK1.\*XK1;

yc1=idft(YK1,N1); %用圆周卷积定理实现的7点圆周卷积

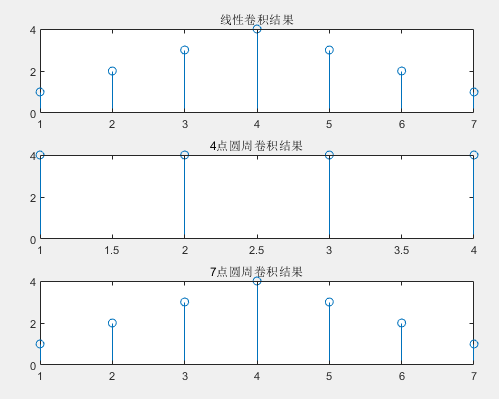


图7-4 例7-4结果图

[实例7-5] **：**

已知序列

，

求它们的线卷积yl(n)=h(n)\*x(n)和不同N点的圆周卷积yN(n)=h(n)\*x(n), 并研究两者之间的关系。

【解】x(n)的长度为N1 = 12点，h(n)的长度为N2 = 6点，我们用如下程序求出h(n)和x(n)的线卷积、N1点的圆周卷积、N1+N2-1点圆周卷积。

程序如下：

clear all

n=[0:1:11];

m=[0:1:5];

N1=length(n);

N2=length(m);

xn=n; %生成x(n)

hn=ones(1,N2); %生成h(n)

yln=conv(xn,hn); %直接用函数conv计算线性卷积

ycN1=circonv(xn,hn,N1); %用函数circonv计算N1点圆周卷积

ycn=circonv(xn,hn,N1+N2-1); %用函数circonv计算N1+N2-1点圆周卷积

nyl=[0:1:length(yln)-1];

ny1=[0:1:length(ycN1)-1];

ny2=[0:1:length(ycn)-1];

subplot(3,1,1); %画图

stem(nyl,yln,'.');

ylabel('线性卷积');

axis([0,18,0,60]);

subplot(3,1,2);

stem(ny1,ycN1,'.');

ylabel('圆周卷积N1')

axis([0,18,0,60]);

subplot(3,1,3);

stem(ny2,ycn,'.');

ylabel('圆周卷积N1+N2-1')

axis([0,18,0,60]);

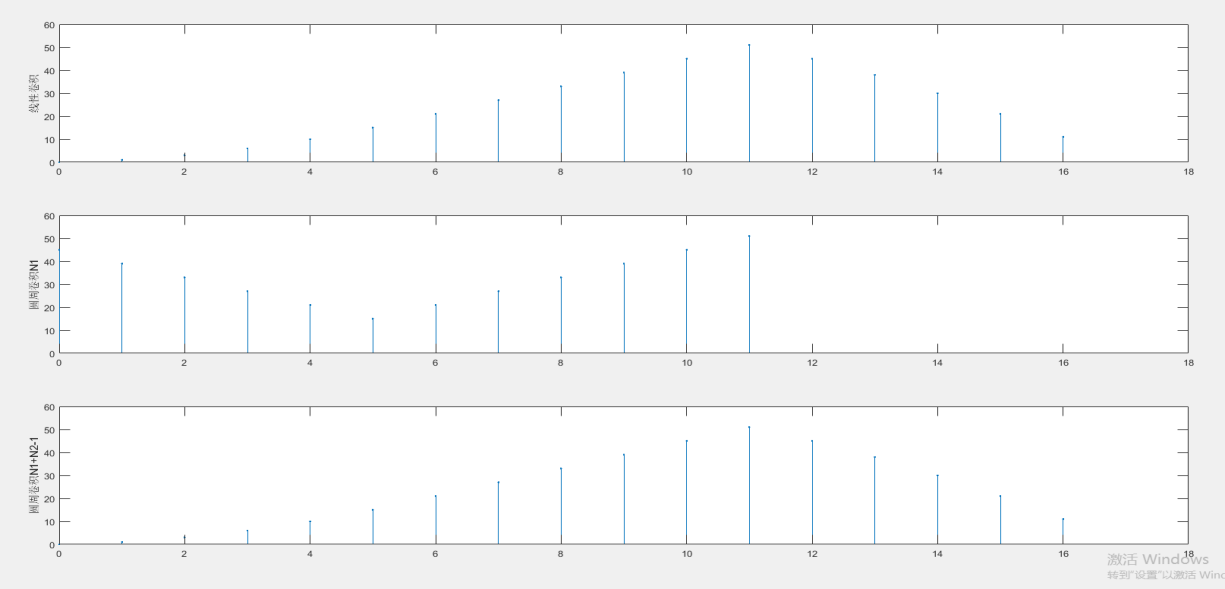


图7-5 例7-5结果图

线性卷积的长度为被卷积的两序列的长度之和减一，即此例中的N1+N2-1。从图中的仿真结果可看出，只有当圆周卷积的长度大于等于线卷积的长度时，圆周卷积的结果才与线卷积的结果一致。

## 3、FFT

**(1) FFT算法**

N点序列的DFT和IDFT变换定义式如下：

|  |  |
| --- | --- |
|  | (7-18) |
|  | (7-19) |

利用旋转因子的周期性，可以得到快速算法（FFT）。快速傅立叶变换FFT并不是与DFT不相同的另一种变换，而是为了减少DFT运算次数的一种快速算法。它是对变换式(7-18)进行一次次的分解，使其成为若干小点数DFT的组合，从而减小运算量。常用的FFT是以2为基数，其长度。它的运算效率高，程序比较简单，使用也十分方便。当需要进行变换的序列的长度不是2的整数次方的时候，为了使用以2为基的FFT,可以用末尾补零的方法，使其长度延长至2的整数次方。**IFFT一般可以通过FFT程序来完成，比较式(7-18)和(7-19)，只要对取共轭，进行FFT运算，然后再取共轭，并乘以因子1/N,就可以完成IFFT.**

**【**例7-6**】**研究当 1≤N≤64 时，FFT 函数的执行时间。最后画出执行时间相对于N 的图。

Nmax=256;

Fft\_time=zeros(1,Nmax);

for n=1:1:Nmax;

x=rand(1:n);

t=clock;

fft(x);fft\_time(n)=etime(clock,t);

end

n=[1:1:Nmax];

plot(n,Fft\_time,'r.');

xlabel('N');ylabel('时间(单位：秒)');title('FFT 执行时间')

**结果图7-7略**

**(2)频谱分析**

DFT是对序列傅立叶变换的等距采样，因此可以用于序列的频谱分析。在运用DFT进行频谱分析的时候可能有三种误差，分析如下：

* **混淆现象**

从式（7-6）中可以看出，序列的频谱是采样信号频谱的周期延拓，周期是T, 因此当采样速率不满足Nyquist定理，即采样频率小于两倍的信号 (这里指的是实信号)频率时，经过采样就会发生频率混淆。这导致采样后的信号序列频谱不能真实地反映原信号的频谱。所以，在利用DFT分析连续信号频谱的时候，必须注意这一问题。避免混淆现象的唯一方法是保证采样的速率足够高，使频谱交叠的现象不出现。这就告诉我们，在确定信号的采样频率之前，需要对频谱的性质有所了解。在一般的情况下，为了保证高于折叠频率的分量不会出现，在采样之前，先用低通模拟滤波器对信号进行滤波。

* **泄露现象**

实际中的信号序列往往很长，甚至是无限长序列。为了方便，我们往往用裁

短的序列来近似它们。这样可以使用较短的DFT来对信号进行频谱分析。这种裁短等价于给原信号序列乘以一个矩形窗函数。而矩形窗函数的频谱不是有限带宽的，从而它和原信号的频谱进行卷积以后会扩展原信号的频谱。值得一提的是，泄露是不能和混淆完全分离开的，因为泄露导致频谱的扩展，从而造成混淆。为了减小泄漏的影响，可以选择适当的窗函数使频谱的扩散减到最小。

* **栅栏效应**

因为DFT是对单位圆上Z变换的均匀采样，所以它不可能将频谱视为一个连续函数，这样就产生了栅栏效应，从某种角度来看，用DFT来观看频谱就好像通过一个栅栏观看一幅景象，只能在离散点上看到真实的频谱。这样的话就会有一些频谱的峰点或谷点被“栅栏”挡住，不能被我们观察到。减小栅栏效应的一个方法是在原序列的末端补一些零值，从而变动DFT的点数。这种方法的实质是人为地改变了对真实频谱采样的点数和位置，相当于搬动了“栅栏”的位置，从而使得被挡住的一些频谱的峰点或谷点显露出来。注意，这时候每根谱线所对应的频率和原来的已经不相同了。

从上面的分析过程可以看出，DFT可以用于信号的频谱分析，但必须注意可能产生的误差，在应用过程中要尽可能减小和消除这些误差的影响。

【例7-7】 已知信号x(t)=0.15sin(2πf1t)+sin(2πf2t)-0.1sin(2πf3t)，f1=1Hz，f2=2Hz，f3=3Hz。取fs=32Hz作频谱分析。

【解】程序如下：

clear all

fs=32;

N=fs;

n=0:N-1;

f1=1;f2=2;f3=3;

xn=0.15\*sin(2\*pi\*f1\*n/fs)+sin(2\*pi\*f2\*n/fs)-0.1\*sin(2\*pi\*f3\*n/fs);XK=fft(xn,N);

magXK=abs(XK);

phaXK=angle(XK);

subplot(1,2,1)

stem(n,xn,'.');

xlabel('n');ylabel('x(n)');

axis([0,32,-1.2,1.2]);

grid;

subplot(1,2,2)

k=0:length(magXK)-1;

stem(k,magXK,'.');

xlabel('k');ylabel('│X(k)│');

axis([0,32,0,17]);

grid

仿真图如图所示。(a)是信号的时域波形图，(b)是信号的频谱图。

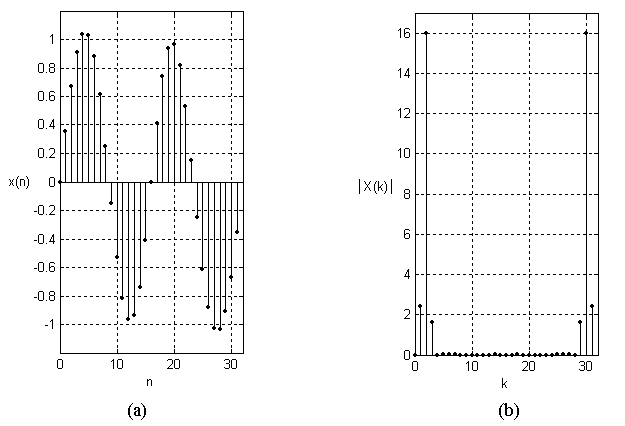


图7-7 例7-7结果

【例7-8】构造一个信号频率f = 1Hz，抽样频率fs = 32Hz的正弦序列，分别在整周期抽样间隔和非整周期抽样间隔两种情况下作谱分析。

MATLAB仿真程序如下：

clear all;fs=32;N=32;n=0:N-1;f=1;

d=(1/fs)\*0.05; %抽样间隔误差

Ts=1/fs; %整周期抽样间隔，Ts\*N=T

Ts1=1/fs-d; %非整周期抽样间隔，Ts\*N≠T

xn=sin(2\*pi\*f\*n\*Ts); %整周期抽样得到的序列xn

xn1=sin(2\*pi\*f\*n\*Ts1); %非整周期抽样得到的序列xn1

XK=fft(xn,N); %由整周期抽样序列xn的DFT得到的谱

magXK=abs(XK);phaXK=angle(XK);

XK1=fft(xn1,N); %由非整周期抽样序列xn1的DFT得到的谱

magXK1=abs(XK1);phaXK1=angle(XK1);

subplot(2,2,1);stem(n,xn,'.');

xlabel('n');ylabel('整周期抽样得到的序列xn');

axis([0,N,-1.2,1.2]);grid;

subplot(2,2,2)

k=0:length(magXK)-1;

stem(k,magXK,'.');

xlabel('k');ylabel('整周期抽样序列xn的幅值谱│X(k)│');

axis([0,N,0,(fs/2)+1]);grid

subplot(2,2,3);stem(n,xn1,'.');

xlabel('n');ylabel('非整周期抽样得到的序列xn1');

axis([0,N,-1.2,1.2]);grid;

subplot(2,2,4)

k=0:length(magXK1)-1;

stem(k,magXK1,'.');

xlabel('k');ylabel('非整周期抽样序列xn1的幅值谱│X(k)1│');

axis([0,N,0,(fs/2)+1]);grid

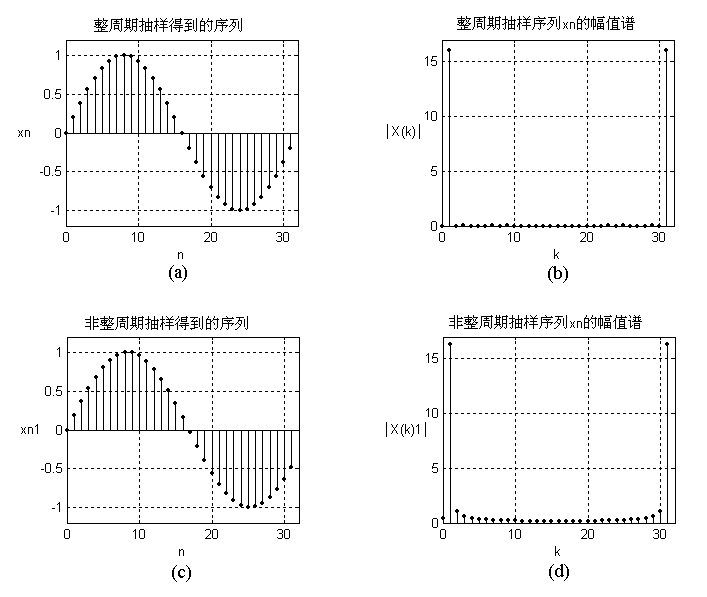


图7-8 例7-8结果

仿真结果，比较整周期抽样序列的幅值谱图(b)和非整周期抽样序列的幅值谱图(d)，可清楚地看到**频谱泄露**现象。

**(3) 快速卷积**

用来计算有限长序列卷积的方法。若选择为N1点序列，为N2点序列，则，的线性卷积为

|  |  |
| --- | --- |
|  | (7-20) |

为使有限长序列的线性卷积可用其圆周卷积来代替而不产生混叠,必须N≥N1+N2-1且N为2的整幂次，现在线性卷积可通过计算两个N点的FFT，一个N点的IFFT和一次N点点积得到。

|  |  |
| --- | --- |
|  | (7-21) |

**(4) 函数应用**

MATLAB为计算数据的离散快速傅立叶变换，提供了一系列丰富的数学函数，主要有fft、ifft、fft2、ifft2、和fftshift、ifftshift等。当所处理的数据的长度为2的幂次时，采用基-2算法进行计算，计算速度会显著增加。所以，要尽可能使所要处理的数据长度为2的幂次或者用添零的方式来添补数据使之成为2的幂次。

【例7-9】用FFT（即快速卷积法）实现,两序列的线卷积。

实现程序：

clear all;

n=[0:11];m=[0:5];

N1=length(n);N2=length(m);

xn=n;hn=ones(1,N2);

N=N1+N2-1;

XK=fft(xn,N);

HK=fft(hn,N);

YK=XK.\*HK;

yn=ifft(YK,N);

if all (imag(xn)==0)&(all(imag(hn)==0))

yn=real(yn);

end

x=0:N-1;

stem(x,yn,'.');

axis([0,18,0,60]);

grid on;

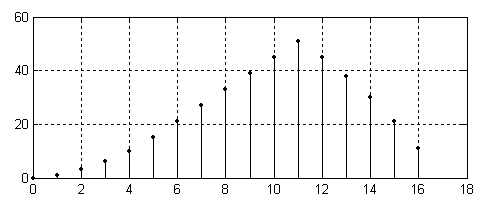


图7-9 例7-9结果

[例7-10] fft在信号分析中的应用

使用频域分析方法从受噪声污染的信号x(t)中鉴别出有用的信号。

程序：

t=0:0.001:1;

%采样周期为 0.001s,即采样频率为 1000Hz;

%产生受噪声污染的正正弦波信号；

x=sin(2\*pi\*100\*t)+sin(2\*pi\*200\*t)+rand(size(t));

subplot(2,1,1)

plot(x(1:100));title(‘信号’)

%画出时域内的信号；

Y=fft(x,512);

%对 X 进行 512 点的傅立叶变换；

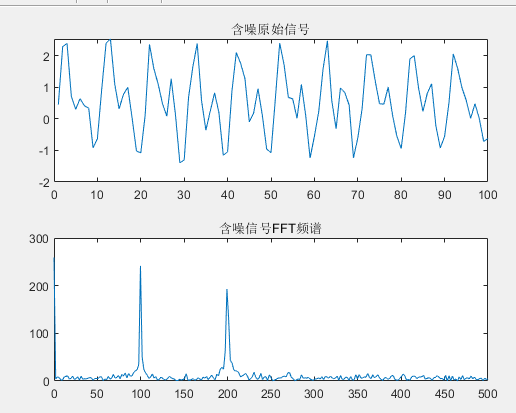
f=1000\*(0:256)/512;

%设置频率轴（横轴）坐标，1000 为采样频率；

subplot(2,1,2)

plot(f,Y(1:257));

%画出频域内的信号



从受噪声污染信号的时域波中，很难看出正弦波的成分。但是通过对 x(t)作傅立叶变换，把时域信号变换到频域进行分析，可以明显看出信号中 100Hz 和 200Hz 的两个频率分量。

图7-10 例7-10结果

# 三、实验内容及步骤

1、验证所有例程，理解原理，观察分析结果，总结实验所得结论。

2、分别计算16点序列的16点和32点fft,绘出幅度谱图形，并绘出该序列的DTFT幅频，观察分析结果。

3、已知某序列在单位圆上的N=64等分样点的Z变换为

用N点IFFT程序计算，绘出。

4、在实验四基础上继续完善（选做）

1）请查询资料，MATLAB实现：采集一段10秒电话拨号音(长按1或其他数字)，选择合适的采样频率抽样，转换为离散时间信号，存储在MATLAB中，并对其添加强度不同的随机噪声后播放出来，描述一下听见的效果如何？

2）选择合适的点数，对采样后的纯净信号和含噪信号分别求解FFT，绘制其幅频曲线，观察结果。  
四、思考题

1、分析ZT、DTFT和DFT之间的相互关系。

2、分析比较N点DFT和FFT的复乘、复加运算量。

3、在什么条件下，两序列的圆周卷积和线性卷积相等？

4、利用FFT求解连续信号频谱需经过哪些环节/会遇到哪些问题？如何解决？

# 五、实验报告要求

1、简述实验目的及原理。

2、在实验报告中附上在实验过程中记录的各个信号序列的特性图，分析所得到的结果图形，总结实验中的主要结论。

3、简要回答思考题。