实验二 时域采样理论的验证

一、 实验原理

奈奎斯特(Nyquist)采样定理：为了保证模拟信号采样后的采样序列能够不失真的还原出原信号，抽样频率必须大于等于两倍原信号谱的最高频率。

二、 实验目的

（1）学习连续时间信号的傅立叶变换

（2）利用傅立叶变换分析采样后信号的频谱，验证奈奎斯特采样定理

（3）信号的重构

三、实验内容

（1）对信号进行采样分析，讨论信号的抽样的最低频率，验证奈奎斯特采样定理。要求对信号分析，确定采样频率，求出采样后的序列的频谱，比较不同采样频率情况下（满足和不满足采样定理）频谱和关系。

（2）利用采用信号重建原始信号，比较不同采样频率情况下（满足和不满足采样定理）和误差。

四、实验过程

1. (1)首先画出信号的时域波形与傅里叶变换后的幅频图



由波形可知，信号在t>0.008和t<-0.008时间内几乎为0，因此，我们在对其离散化获得离散序列时，可以先做截断处理，时域的采样长度为0.016s。最高频率大约是1500HZ，那么在后续实验中，我们取高于1500HZ的采用频率和等于1500HZ的采样频率以及小于1500HZ的采样频率才分析。

(2)若采样率为5000HZ



若采用率为1500HZ



若采样率为1000HZ



(3)主要代码如下

Fs1=5000;

Ts1=1/F1;

L=0.008;

Fs2=1500;

Ts2=1/F2;

N1=L/Ts1;

N2=L/Ts2;

t1=[-L:Ts1:L];

x1=exp(-1000\*abs(t1));

t2=[-L:Ts2:L];

x2=exp(-1000\*abs(t2));

f1=[-Fs1:Fs1/N1:Fs1];

y1=Ts1\*fft(x1);

f2=[-Fs2:Fs2/N2:Fs2];

y2=Ts2\*fft(x2);

figure(1);

subplot(2,2,1);

stem(t1,x1);

title('采样频率为5000HZ的采样信号');

subplot(2,2,2);

plot(f1,y1);

title('采样频率为5000HZ的信号频谱');

subplot(2,2,3);

stem(t2,x2);

title('采样频率为1500HZ的采样信号');

subplot(2,2,4);

plot(f2,y2);

title('采样频率为1500HZ的信号频谱');

2. 利用采用信号重建原始信号 ，比较不同采样频率情况下（满足和不满足采样定理）和误差。

由分析可知，信号的最高有效频率为1500HZ，因此，在不同采样频率情况下，重建信号与重建信号误差如下。

1. 采样频率为5000HZ



1. 采样频率为3000HZ时



（3）采样率为1000HZ时



通过以上三个图可以看出，当采样频率时，采样后的信号与原信号非常接近，重建后的误差也很小；当采样频率时, 采样后的信号与原信号偏差明显，没有保留原始信号特征，重建后的误差也很大；当采样频率时，效果介于两者之间。可见，为了避免产生混叠现象，能从抽样信号无失真地恢复出原始信号，抽样频率必须大于或等于信号频谱最高频率的两倍。

（4）主要代码

dt=0.0005;

t=-0.008:dt:0.008;

xa=exp(-1000\*abs(t));

W=2\*pi\*1500;

K=500;

k=0:1:K;

omega=k\*W/K;

Xa=xa\*exp(-j\*t'\* omega)\*Dt;

Xa=real(Xa);

omega =[-fliplr(omega), omega (2:501)];

Xa=[fliplr(Xa),Xa(2:501)]; Ts=0.00033;

Fs=1/Ts;

n=-25:1:25;

nTs=n\*Ts;

x=exp(-1000\*abs(n\*Ts));

w=k\*pi/K;

X=x\*exp(-j\*n'\*w);

X=real(X);

w=[-fliplr(w),w(2:K+1)];

X=[fliplr(X),X(2:K+1)]; x1=x\*sinc(Fs\*(ones(length(n),1)\*t-nTs'\*ones(1,length(t))));

subplot(2,2,1);

stem(n\*Ts\*1000,x);

xlabel('n');

ylabel('x(n)');

title('采样信号 ');

subplot(2,2,2);

plot(t\*1000,x1);

xlabel('t');

ylabel('x(n)');

title('重建的信号');

subplot(3,1,3);

plot(t\*1000,abs(x1-xa));

xlabel('t');

ylabel('误差绝对值');

title('误差');