

# 大连理工大学

## 本科实验报告二

课程名称： 随机信号分析实验

学院（系）： 电子信息与电气工程学部

专    业： 电子信息工程

班    级： 电信 1806 班

学    号： 201871080

学生姓名： 刘祎铭

2020 年 11 月 17 日

# 大连理工大学实验预习报告

学院（系）： 电子信息与电气工程学部 专业： 电子信息工程 班级： 电信 1806

姓 名： 刘祎铭 学号： 201871080 组：                     

实验时间： 2020.11.11 实验室： 创新园 C220 实验台：                     

指导教师： 李小兵

## 实验 II：系统对随机信号响应的统计特性分析、功率谱分析及应用实验

### 一、实验目的和要求

掌握直接法估计随机信号功率谱的原理和实现方法；掌握间接法估计随机信号功率谱的原理和实现方法；掌握系统对随机信号响应的统计特性分析及仿真实现方法。熟悉 MATLAB 信号处理软件包的使用。

### 二、实验原理和内容

#### （一）实验原理：

##### 1. 直接法估计随机信号功率谱原理

直接法又称为周期图法，它是把随机信号  $x(n)$  的  $N$  点观察数据  $x_N(n)$  视为一能量有限信号，直接取  $x_N(n)$  的傅里叶变换，得到  $X_N(e^{j\omega})$ ，然后取其模值的平方，并除以  $N$ ，作为对  $x(n)$  真实的功率谱  $P(e^{j\omega})$  的估计。工程上，常使用离散 Fourier 变换（DFT，编程上使用其快速算法 FFT），即  $P_X(k) = \frac{1}{N} |X_N(k)|^2$ ，进行计算。

##### 2. 间接法估计随机信号功率谱

间接法的理论基础是 Wiener-Khintchine 定理，具体的实现方法是先由  $x_N(n)$  估计出自相关函数  $\hat{r}(m)$ ，然后对  $\hat{r}(m)$  求傅里叶变换得到  $x_N(n)$  的功率谱，记之为  $X_N(e^{j\omega})$ ，并以此作为对真实功率谱  $P(e^{j\omega})$  的估计。工程上，常使用离散 Fourier 变换（DFT，编程上使用其快速算法 FFT），即：

$$P_X(k) = \sum_{m=-M}^M \hat{r}(m) e^{-j \frac{2\pi km}{2M+1}}, \quad |M| \leq N-1$$

进行计算。因为由这种方法求出的功率谱是通过自相关函数间接得到的，所以又称为间接法或 Blackman-Tuckey(BT)法，该方法是 FFT 出现之前常用的谱估计方法。

### 3. 时域中系统对随机信号响应的统计特性分析及仿真

根据系统卷积性质，计算系统输出信号的统计特性。有如下性质：

$$m_Y = m_X \sum_n h(n)$$

$$R_Y(m) = \sum_j \sum_k R_X(m+j-k)h(j)h(k)$$

### 4. 频域中系统对随机信号响应的统计特性分析及仿真

根据卷积定理，输入、输出信号功率谱的关系为：

$$R_Y(e^{j\omega}) = R_X(e^{j\omega})|H(e^{j\omega})|^2$$

在计算系统输出信号功率谱时，如果在时域时计算困难，可以按照上式在频域计算。

## （二）实验内容：

### 1. 直接法估计随机信号功率谱

#### (1) 生成 1024 点数据的随机信号

$$X(n) = 2\cos(2\pi f_1 t + \varphi_1) + 5\cos(2\pi f_2 t + \varphi_2) + N(n)$$

其中  $f_1 = 30\text{Hz}$ ,  $f_2 = 100\text{Hz}$ ,  $\varphi_1, \varphi_2$  为在  $[0, 2\pi]$  内的均匀分布的随机变量,  $N(n)$  是数学期望为 0, 方差为 1 的高斯白噪声。

#### (2) 用周期图法计算 $X(n)$ 的功率谱，并绘图。

#### (3) 用 MATLAB 函数 periodogram 重新计算 $X(n)$ 的功率谱，并与(2)做比较。

### 2. 间接法估计随机信号功率谱

#### (1) 计算以上 $X(n)$ 的自相关函数。

#### (2) 通过计算自相关函数的 Fourier 变换，求 $X(n)$ 的功率谱并绘图。

#### (3) 利用 MATLAB 函数 psd、pwelch 重新计算 $X(n)$ 的功率谱，并与(2)做比较。

### 3. 系统对随机信号响应的统计特性分析及仿真

- (1) 生成含 500 点数据的高斯分布白噪声随机信号  $x(n)$ 。
- (2) 设计一个带通系统  $H(e^{j\omega})$ ，其上、下截止频率分别为 4KHz 和 3KHz。
- (3) 计算  $x(n)$  通过以上带通滤波器的自相关函数和功率谱密度。

#### 4. 探究式实验内容

### DTMF 电话号码识别

#### 【研究内容和要求】

研究随机信号分析的相关分析方法/谱分析方法在实际中的应用。

教师设计一路语音信号，根据 DTMF 标准生成 11 位电话号码拨号音信，加上加性标准的随机分布噪声数据；或者采集一组实际拨号语音信号。要求学生根据信号数据自行找出所携带的电话号码。

学生需要研究 DTMF 信号构成规则，掌握信号分割方法，利用相关法或谱分析方法来找到电话号码。

如果采用其他处理方法，需要详细说明原理和操作过程。

#### 【数据设计】

提供一路声音信号文件：学号-实验二案例 2\_无噪 DTMF.wav。

实际数据的生成，可以采用 Matlab 编写的代码生成，或者用 Cooledit 等软件生成，再叠加高斯/均匀/指数等分布律的数据作为噪声。

电话拨号音的号码要与学生的学号相关，并且不容易看出之间的关系。

DTMF 信号构成基本原理：

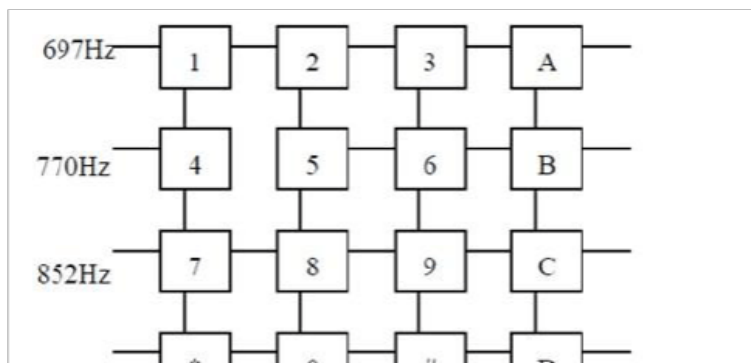


图 1 DTMF 信号构成基本原理：

图 1 中每个按键音包含两个信号频率，通过按键拨号发出的音频可以推断出是按了哪个拨号键。

### 【实验布置】

在实验一（相关法）、二（谱分析）的内容中加入，可以在实验二中或者另外安排时间验收。

只提供一路信号文件：学号-实验二案例 2\_无噪 DTMF.wav；或者给学生 p 代码，要求学生输入自己的学号生成自己的数据。可以提示学生用 `audioread` 函数取得采样率和数据，或者用其他工具取得数据信息。

提示学生可以用相关/谱分析的方法。

提醒学生数据具有特异性，不能抄袭答案。

### 【验收内容】

每个学生均需要经过验收，验收时，教师根据学号，运行验收代码，生成每个学生的正确结果以供核对。学生需要：

下载与个人学号相关联的实验数据，分析、实验。

根据名单确认学号，提供根据数据解码得到的电话号码。

解释实验原理和过程，回答教师提问。

上交自行撰写的实验/研究报告（此部分不需要预习报告，可单独另行成文）。

## 三、实验步骤

### 1. 直接法估计随机信号功率谱

- (1) 准备实验环境。
- (2) 生成随机信号样本。

```
%两个带随机相位的单谱信号与白噪声之和
N=1024;fs=1000;           %序列长度和采样频率
t=(0:N-1)/fs;             %时间序列
fai=random('unif',0,2*pi,1,2); %产生 2 个[0, 2pi]内均匀随机数
xn=cos(2*pi*30*t+fai(1))+3*cos(2*pi*100*t+fai(2))+randn(1,N); %产生含噪声的随机序列
```

- (3) 显示随机信号时域波形。

```
figure,plot(xn);
title('随机信号时域波形')
```

- (4) 周期图法直接估计随机信号功率谱，通过 Fourier 变换取上述高斯白噪声 1024 点计算

对应的功率谱并绘图。

```
%对样本进行傅里叶变换，取模平方时间平均
Sx1=abs(fft(xn)).^2/N;           %估计功率谱
f=(0:N/2-1)*fs/N;               %频率轴坐标
figure
subplot(211);
plot(f,10*log10(Sx1(1:N/2)));grid on; %用 dB/Hz 做功率谱单位，画图
xlabel('f (Hz)');
ylabel('Sx1(f) (dB/Hz)');
title('周期图法估计功率谱');
```

(5) 与内置函数 periodogram 对比。

```
Sx2=periodogram(xn); %用 periodogram 函数生成功率谱
subplot(212);
plot(f,10*log10(Sx2(1:N/2)));grid on; %用 dB/Hz 做功率谱单位，画图
xlabel('f (Hz)');
ylabel('Sx2(f) (dB/Hz)');
title('periodogram 函数估计功率谱');
```

(6) 尝试改变信号、噪声的幅度和频率观察效果  
使用 GUI 设计幅值、频率、相位可实时调整的随机信号，使用各方法估计功率谱

## 2. 间接法估计随机信号功率谱

- (1) 准备环境。
- (2) 生成随机信号样本。

```
%两个带随机相位的单频谱信号与白噪声之和
N=1024;fs=1000;                %序列长度和采样频率
t=(0:N-1)/fs;                  %时间序列
fai=random('unif',0,2*pi,1,2); %产生 2 个 [0, 2pi] 内均匀随机数
xn=cos(2*pi*30*t+fai(1))+3*cos(2*pi*100*t+fai(2))+randn(1,N); %产生含噪声的随机序列
```

(3) 显示随机信号时域波形。

```
figure,plot(xn);grid on
title('随机信号时域波形')
```

(4) 计算高斯白噪声的自相关函数。

```
Rxx=xcorr(xn,'biased');          %估计自相关函数 Rxx
```

(5) 间接法估计随机信号的功率谱

```
Rxx=xcorr(xn,'biased');          %估计自相关函数 Rxx
Sx1=abs(fft(Rxx));               %对 Rxx 进行 FFT 得到功率谱
f=(0:N-1)*fs/N/2;               %频率轴坐标
figure;plot(f,10*log10(Sx1(1:N)));grid on; %用 dB/Hz 做功率谱单位，画图
```

```
xlabel('f (Hz)');
ylabel('Sx(f) (dB/Hz)');
title('自相关函数法估计功率谱');
```

(6) 与内置函数对比。

### 1. 使用 MATLAB 内置的 psd 函数生成功率谱并比较

```
Nseg=256; %分段间隔为 256
window=hanning(Nseg); %汉宁窗
noverlap=Nseg/2; %重叠点数为 128
f=(0:Nseg/2)*fs/Nseg; %频率轴坐标
Sx2=psd(xn,Nseg,fs>window,noverlap,'none'); %psd 函数估计功率谱
figure;plot(f,10*log10(Sx2));grid on;
xlabel('f (Hz)');
ylabel('Sx(f) (dB/Hz)');
title('Bartlett 法估计功率谱');
```

### 2. 使用 MATLAB 内置的 pwelch 函数生成功率谱并比较

```
Sx3=pwelch(xn>window,128,Nseg,fs,'onesided')*fs/2; %Welch 函数估计功率谱
figure;plot(f,10*log10(Sx3));grid on; %用 dB/Hz 做功率谱单位，画图
xlabel('f (Hz)');
ylabel('Sx(f) (dB/Hz)');
title('Welch 法估计功率谱');
```

(7) 比较三者误差。

```
Sx11=Sx1';
err1=Sx11(1:16:N*2-1)-Sx2(1:N/8); % 间接法谱估计与 psd 函数估计的误差
ff=f(1:length(f)-1);
figure;subplot(311);plot(ff,10*log10(err1));
xlabel('f (Hz)');
ylabel('Sx1(f)-Sx2(f) (dB/Hz)');
title('间接法谱估计与 psd 函数估计的误差');
err2=Sx11(1:16:N*2-1)-Sx3(1:N/8); % 间接法谱估计与 pwelch 函数估计的误差
subplot(312);plot(ff,10*log10(err2));
xlabel('f (Hz)');
ylabel('Sx1(f)-Sx3(f) (dB/Hz)');
title('间接法谱估计与 pwelch 函数估计的误差');
err3=Sx2-Sx3; % psd 函数估计与 pwelch 函数估计的误差
subplot(313);plot(f,10*log10(err3));
xlabel('f (Hz)');
ylabel('Sx2(f)-Sx3(f) (dB/Hz)');
title('psd 函数估计与 pwelch 函数估计的误差');
```

(8) 尝试改变信号、噪声的幅度和频率观察效果  
使用 GUI 设计幅值、频率、相位可实时调整的随机信号，使用各方法估计功率谱。

### 3. 系统对随机信号响应的统计特性分析及仿真

- (1) 准备环境。
- (2) 生成高斯随机信号样本。

```
N=500; %样本长度 N=500，对应时长 25ms
xt=random('norm',0,1,1,N); %产生 1*N 个高斯随机数
% 显示时域波形
figure,plot(xt);
title('随机信号时域波形')
```

- (3) 设计一个带通滤波器，截止频率为3KHz，4KHz，显示其时域频域波形和频域波形

```
%冲激响应
ht=fir1(101,[0.3 0.4]); % 101 阶带通滤波器，数字截止频率为 0.3 和 0.4
% 显示冲激响应函数 ht
figure,plot(ht)
title('冲激响应函数 ht')
HW=fft(ht,2*N); %2N 点滤波器频率响应（系统传输函数）
% 显示传递函数 HW
figure,plot((1:N)/N,abs(HW(1:N))); %频域波形输出
title('传递函数 HW')
```

- (4) 计算输出信号的自相关函数、功率谱等统计量。

```
Rxx=xcorr(xt,'biased'); %直接法估计白噪声的自相关函数
% 显示自相关函数 Rxx
figure,stem(Rxx)
title('输入自相关函数 Rxx')
```

```
Sxx=abs(fft(xt,2*N)).^2/(2*N); %周期图法估计白噪声的功率谱
```

```
HW2=abs(HW).^2; %系统的功率传输函数
Syy=Sxx.*HW2; %输出信号的功率谱
```

```
Ryy=fftshift(iffshift(Syy)); %用 IFFT 求输出信号的自相关函
数 %函数 fftshift 对数组进行移位
%Ryy=iffshift(Syy);
```

- (5) 图形展示系统输出观察效果。



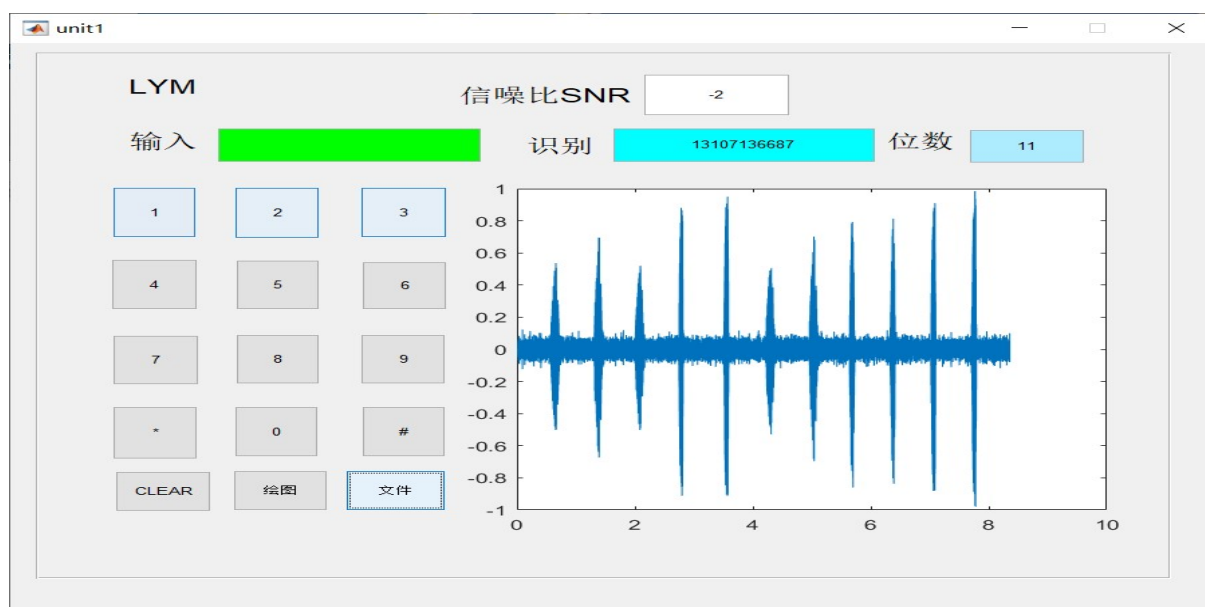
```

w=(1:N)/N; %功率谱密度横轴坐标
t=(-N:N-1)/N*(N/20000); %自相关函数横轴坐标
subplot(4,1,1);plot(w,abs(Sxx(1:N))); %输入信号功率谱密度
xlabel('归一化频率 f');ylabel('Sxx(f)');title('输入信号功率谱密度');
subplot(4,1,2);plot(w,abs(HW2(1:N))); %系统的功率传输函数
xlabel('归一化频率 f');ylabel('H2(f)');title('系统的功率传输函数');
subplot(4,1,3);plot(w,abs(Syy(1:N))); %输出信号的功率谱密度
xlabel('归一化频率 f');ylabel('Syy(f)');title('输出信号的功率谱密度');
subplot(4,1,4);plot(t,Ryy); %输出信号的自相关函数
xlabel('归一化频率 f');ylabel('Ryy(f)');title('输出信号的自相关函数');

```

#### 4. 探究式实验内容 DTMF 电话号码识别

##### 1. 利用 GUI 设计电话音号码识别界面



##### 2. 将电话音信号进行分割，获取每一个拨号号码对应的信号

```

n=max(sampledata(1:length(sampledata)))%获得音频数据峰值
count1=0;%初始化
i=1;%初始化
numcount1=[];%初始化
while(i<=length(sampledata)-2640/48000*Fs)%最后一段长2640，如果按键发音，
大小2640的样本长度正好容纳，对之外的每一个点进行讨论
    for j=i:ceil(i+2640/48000*Fs)%对于采样率48000，用48000合适，但48000*Fs
适用于各种音频。%对讨论是否认为是噪声并需要置0的点的后面的2640个点进行讨论
        if(abs(sampledata(j))>0.3*n)%我们认为峰值的0.39是区分按键音
和环境噪音的标志，这个是由峰值1.79的取高度0.7作为分界点，这个是由图像特点，
就是噪音和按键音幅度有显著差异决定的
            count1=count1+1;%计数
        end
    end
end

```

if(count1>550/48000\*Fs)%统计有多少个大于0.3的点，用550作为分界点进行保留，不是算出来的，而是使用matlab试出来的。不写count1的‘；’很容易区分出噪声含有的count1数目和按键音数目

numcount1=[numcount1,sampleddata(i:i+ceil(4799/48000\*Fs))'];%!!!!!!!  
正常录音不需转置，但老师的测试音频需要!!!!!!%把按键音合并起来，该点和该点之后4800点，即100ms，会丢掉一部分按键音尾端数据，设计中也会舍弃很少一些首端数据，不过数量少，而且按键音中间的声音更具备特征，信号质量更好。

i=ceil(i+5000/48000\*Fs);%取完按键音后跳过该段，不重复截取声音片段。

```
%           else
%           i=ceil(i+600/48000*Fs)-count1;
end
i=i+1;%计数
count1=0;%复位
end
t=0:1/Fs:(length(sampledata)-1)/Fs;%输出原始波形的时间转换
plot(t,sampledata);%输出原始波形
numlength=length(numcount1);%接收到的信号长度
count=numlength/(1+ceil(4799/48000*Fs));%计算出发了多少个数字
set(handles.receiveScreen,'string','');%清屏
```

更加简单但效果更好的基于频域的电话音信号分割

```
L=length(sampledata);
while(i<L)
    if(i<L-Fs/20)
        xh=sampleddata(i:i+Fs/20);
    else
        xh=sampleddata(i:L);
    end
    f=fft(xh,Fs);
    b=abs(f);
    p=b.*(Fs);
    if(max(p(1:1000))>1.2&&max(p(1000:1700))>1.2)
        numcount1=[numcount1,sampleddata(i:i+4409)'];
        i=i+Fs/12;
    else
        i=i+Fs/200;
    end
end
```

### 3. 判断每个号码信号对应的电话号码

for i=1:count%一共count个数字

d=numcount1(1+(1+ceil(4799/48000\*Fs))\*(i-1):(1+ceil(4799/48000\*Fs))\*i);%取

## 每位拨号音的采样点

```

f=fft(d,Fs); % 以N=2048 作FFT 变换d是取出来每位拨号音的采样点
a=abs(f);
p=a.*a/Fs; % 计算功率谱
num(1)=find(p(1:1000)==max(p(1:1000))); % 找行频
num(2)=1000+find(p(1000:1700)==max(p(1000:1700))); % 找列频
if (num(1) < 730)
    row=1; % 确定行数
elseif (num(1) < 810)
    row=2;
elseif (num(1) < 900)
    row=3;
else
    row=4;
end
if (num(2) < 1260)
    coloum=1; % 确定列数
elseif (num(2) < 1395)
    coloum=2;
elseif (num(2) < 1550)
    coloum=3;
else coloum=4;
end
if row==4&&coloum==2%计算出对应的数字值
    num=0;
elseif(row==4&&coloum==1)
    num='*';
elseif(row==4&&coloum==3)
    num='#';
elseif(coloum==4)
    if(row==1)
        num='A';
    elseif(row==2)
        num='B';
    elseif(row==2)
        num='C';
    else
        num='D';
    end
else
    num=coloum+3*(row-1); %根据行列关系计算出对应的数字值
end
strnum=strcat(get(handles.receiveScreen, 'string'), num2str(num));
set(handles.receiveScreen, 'string', strnum); %将结果显示在屏幕上

```

```

end
    set(handles.edit4,'string',count);%将位数显示在屏幕上
end
4. 添加信噪比可改变的噪声信号
    [filename,filepath]=uigetfile('.wav','选择音频文件');%打开文件
    if(filename==0)
        return%判断文件是否为空
    else
        audeofile=strcat(filepath,filename);%形成音频文件路径
        [sampledata,Fs]=audioread(audeofile);%从音频文件路径中读取音频文件，获得
        采样率和音频数据
        SNR=get(handles.edit5,'String');
        sampledata=awgn(sampledata,str2num(SNR),'measured');
5. 设计滤波器，对噪声信号进行滤波
Hd=band_pass(600,690,1710,1800);                                %带通滤波器

sampledata=filter(Hd,sampledata);

function Hd = band_pass(Fstop1,Fpass1,Fpass2,Fstop2);

%BAND_PASS Returns a discrete-time filter object.

% MATLAB Code

% Butterworth Bandpass filter designed using FDESIGN.BANDPASS.

% Generated by MATLAB(R) 9.0 and the Signal Processing Toolbox 7.2.

% Generated on: 18-Dec-2019 14:51:03

% All frequency values are in Hz.

Fs = 44100; % Sampling Frequency

% Fstop1 = 500;           % First Stopband Frequency

% Fpass1 = 690;           % First Passband Frequency

% Fpass2 = 1710;          % Second Passband Frequency

% Fstop2 = 1800;          % Second Stopband Frequency

Astop1 = 70;              % First Stopband Attenuation (dB)

Apass  = 1;               % Passband Ripple (dB)

Astop2 = 70;              % Second Stopband Attenuation (dB)

match  = 'passband';     % Band to match exactly

% Construct an FDESIGN object and call its BUTTER method.

```

```
h = fdesign.bandpass(Fstop1, Fpass1, Fpass2, Fstop2, Astop1, Apass, ...
                    Astop2, Fs);
```

```
Hd = design(h, 'butter', 'MatchExactly', match);
```

#### 四、实验数据记录表格

##### 1. 直接法估计随机信号功率谱

	实验点 1
直接法功率谱	
内置函数功率谱	

##### 2. 间接法估计随机信号功率谱

	实验点 1
间接法功率谱	
内置函数功率谱	

##### 3. 系统对随机信号响应的统计特性分析及仿真

输入信号功率谱密度	
系统的功率传输函数	
输出信号的功率谱密度	
输出信号的自相关函数	

识别的号码	13107136687
-------	-------------