



北京航空航天大学
B E I H A N G U N I V E R S I T Y

《数字信号处理》软件部分 实验报告

院(系)名称	高等理工学院
专业名称	自动化
学 号	16231235
姓 名	李谨杰
同组同学姓名	李世珪
指导教师	袁梅

2019 年 5 月

实验一 利用离散傅里叶变换（DFT）计算声发射信号频谱

实验时间：2019.05.18

同组同学：李世焱

一、实验结果

1. 实验程序代码：

getWN.m:

```
function a = getWN(k,N)
a=exp(-1j*(2*pi/N)*k);
```

ruan1.m:

```
clc;clear;
filename='LeadBreak.txt';
Data=load(filename);
Data=Data(1:2001);
F=1*10^6;
t_all=2*10^-3;
t=0:1/F:t_all;
figure(1); %发射信号
plot(t,Data);title('发射信号波形');
xlabel('时间t/s'),ylabel('声波信号大小x(t)');
```

```
N=2048;
DN=ones(N,N); %构造DFT矩阵
for n=2:N
    for k=2:N
        DN(n,k)=getWN((n-1)*(k-1),N);
    end
end
```

```
size=size(Data,2);
if N>size %若N>数据长度, 补零
    Dataplus=zeros(1,N);
    Dataplus(1:size)=Data;
    Dataplus=Dataplus';
else
    Dataplus=Data';
end
```

```
X=DN*Dataplus; %DFT变换
Xtest=fft(Data,N); %FFT, 用于检验
```

```
n=0:2/N:2-2/N;
figure(2); %DFT
subplot(2,1,1);
stem(n,abs(X),'.');title('DFT变换的幅频特性');
xlabel('ω/π'),ylabel('幅度X(e^jω)');
subplot(2,1,2);
plot(n,angle(X));title('DFT变换的相频特性');
xlabel('ω/π'),ylabel('相位');
```

```
% figure(3); %FFT
% subplot(2,1,1);
% stem(n,abs(Xtest));title('FFT变换的幅频特性');
% xlabel('ω/π'),ylabel('幅度X(e^jω)');
% subplot(2,1,2);
%
% plot(n,angle(Xtest));title('FFT变换的相频特性');
% xlabel('ω/π'),ylabel('相位');
```

```
f=n(1:N/2)*F/2; %0-F/2
XA=X(1:N/2)*(1/F); %模拟信号在频域的采样
figure(4); %模拟信号
subplot(2,1,1);
plot(f,abs(XA));title('模拟信号的幅频特性');
xlabel('Hz'),ylabel('幅度X_a(jf)');
subplot(2,1,2);
```

```
plot(f,angle(XA));title('模拟信号      相位'  
的相频特性'),xlabel('Hz'),ylabel(''
```

2. 实验典型界面

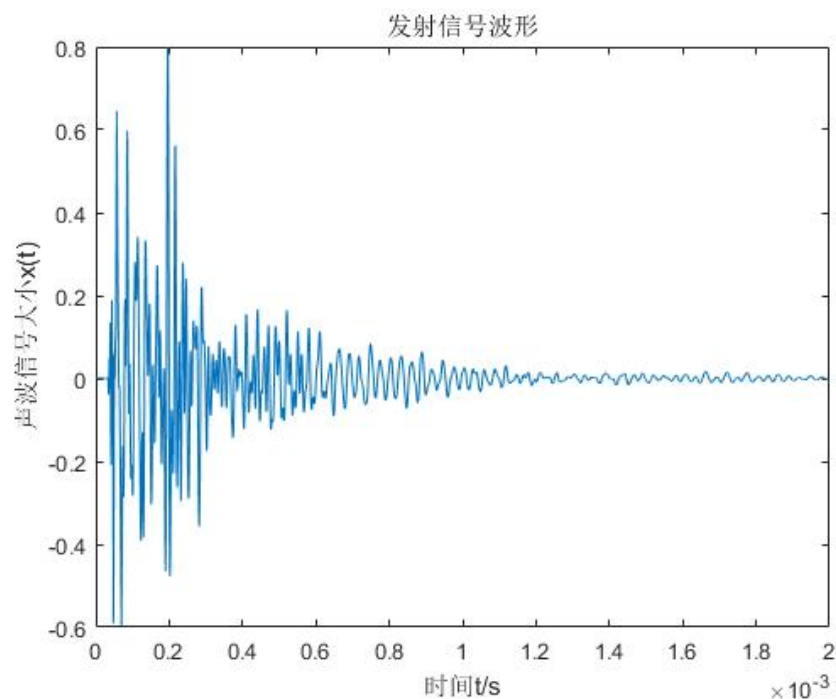


图 2.1 时域信号

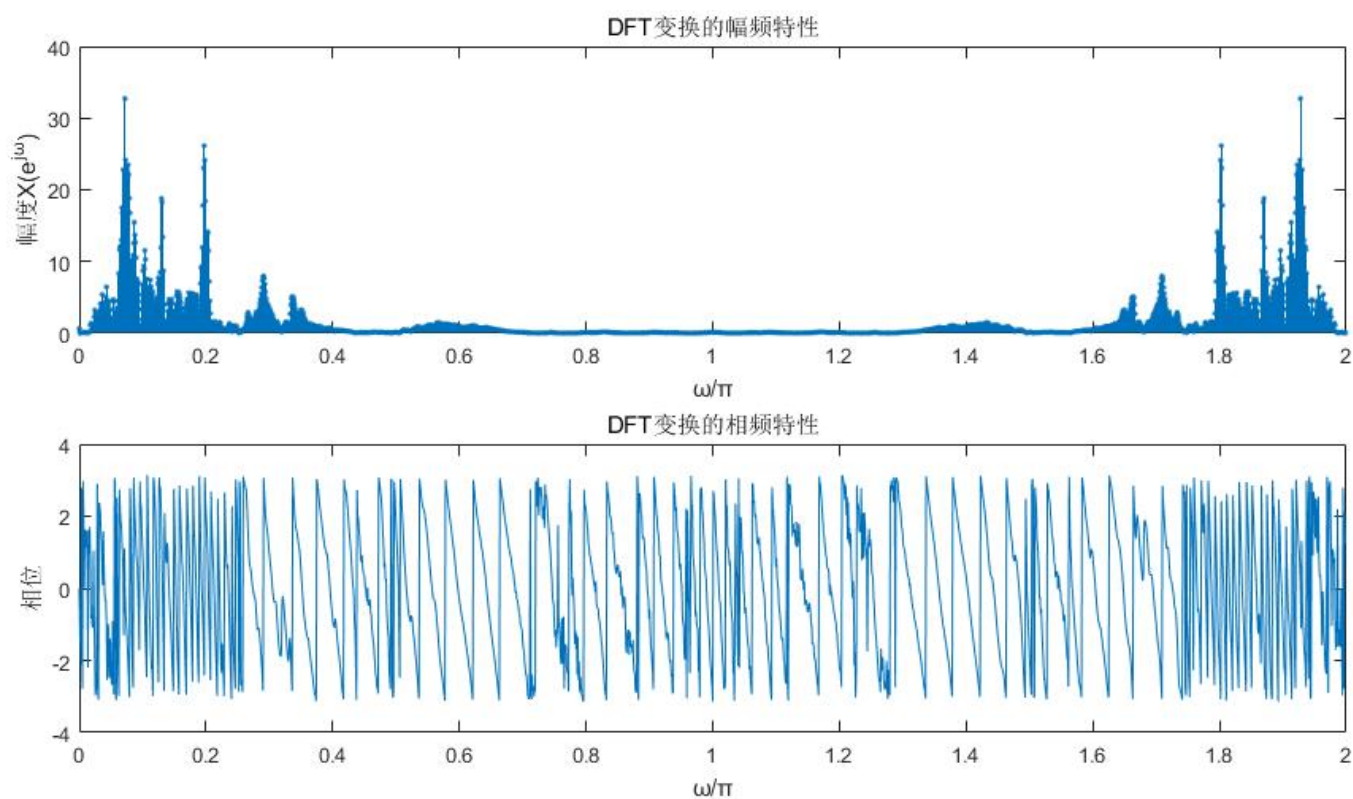


图 2.2 经 DFT 变换得到的数字频谱

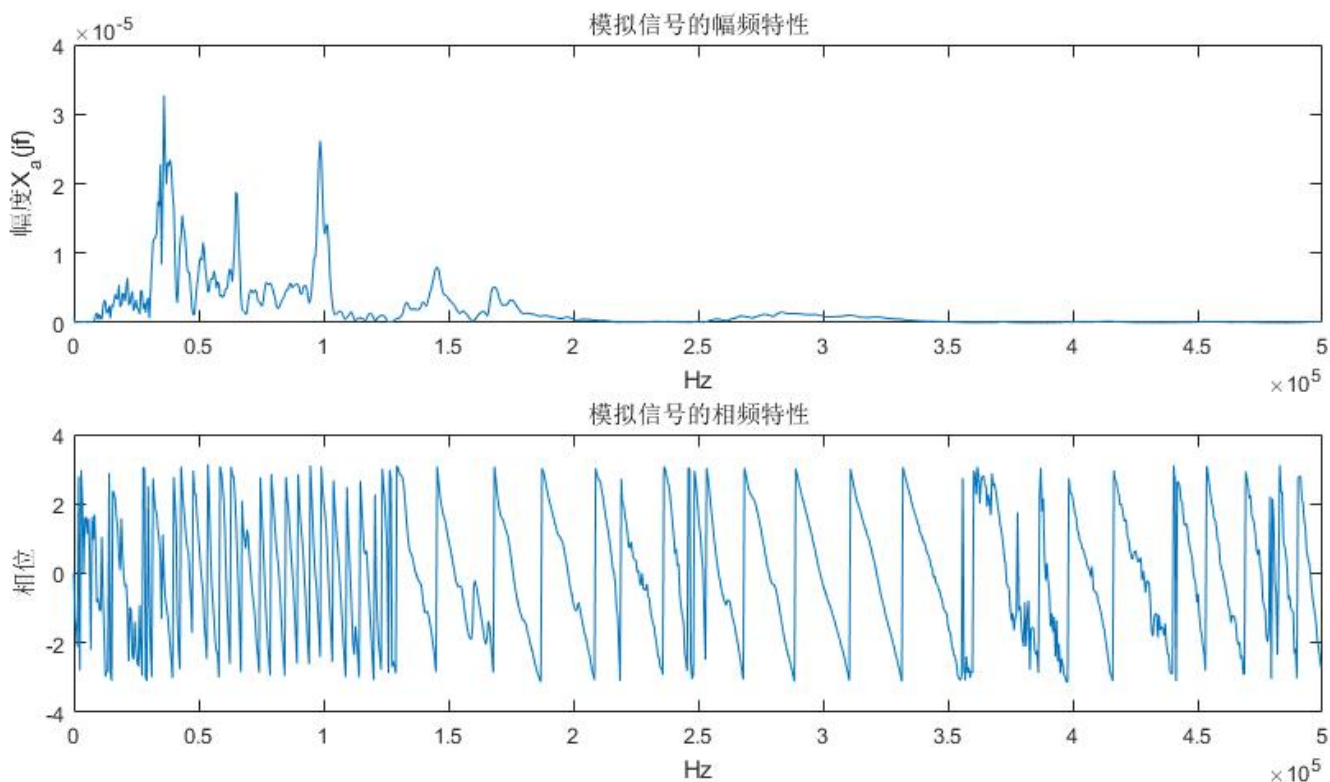


图 2.3 模拟信号频谱图

二、结果分析

1. 实验结果及分析

(1) 本实验的数据来自于真实的铅笔铅芯断裂时产生的声发射信号。通过判断 DFT 变换后的幅值是否超过某个阈值，可以判断铅笔断裂的情况。老师实际给出的数据采集了 4ms，共 4002 个数据，因此在实际实验时修改了代码，采集前 2000 个数据。另外，我在实验中采集了 2001 个数据，严格来说只需要 2000 个就行，因为 $t=0s$ 时不采样， $t=Ts$ 时采集第一个数据。

(2) DFT 变换利用 D 矩阵进行：

$$\mathbf{X} = \mathbf{D}_N \mathbf{x}, \quad \mathbf{D}_N = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & \cdots & W_N^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & \cdots & W_N^{(N-1)(N-1)} \end{pmatrix}, \quad W_N^{kn} = e^{-j\frac{2\pi}{N}kn}$$

与 MATLAB 自带的 FFT 处理的图对比后，可知 DFT 结果正确。注意，DFT 变换后的横坐标是数字角频率 ω ，范围是 $0-2\pi$ ，幅值的物理量是 $X(e^{j\omega})$ 。另外，由于 DFT 所得频谱信号是离散的，用茎叶图画比较好。

(3) 数字角频率 ω 的 π 点对应着模拟频率 $F_s/2$ 。由采样定理，信号的最高频率不应该超过 $F_s/2$ ，因此频谱分析的范围是 $0-F_s/2$ ，即数字角频率的 $0-\pi$ 范围。

另外，由公式 $X_a(jf) = T \cdot \text{DFT}[x(n)]$ ，需要将 DFT 的频谱值乘以采样周期 T 。经过

坐标和幅值的变化，即可将 DFT 频谱变化为具有真实物理意义的模拟信号频谱。

2. 本次实验收获、体会及建议

经过这次实验，我更加深刻地理解了时域、数字频域、模拟频域之间的关系，锻炼了 MATLAB 编程能力。老师检查的时候会询问许多问题，从这些问题中，我发现自己对某些 DSP 知识掌握得不够扎实。今后要更加细致地复习上课讲解的内容，这样才能全面深刻地理解 DSP 这门学科。

实验二 离散太阳黑子信号的时域滤波及其周期性分析

实验时间：2019.05.25

同组同学：李世焱

一、实验结果

1. 实验程序代码：

getWN.m: (得到 WN 的子函数)

```
function a = getWN(k,N)
a=exp(-1j*(2*pi/N)*k);
```

juanji.m: (用于滑动平均的子函数)

```
function S = juanji(K,Spot_N) %K
为窗口长度
N=length(Spot_N); %太阳黑子数据数量
hn=ones(1,K); %滑动平均时域信号
```

```
L=K+N-1; %刚好循环卷积完毕
```

```
x1=zeros(1,L); %补零
```

```
x2=zeros(1,L);
```

```
x1(1,1:K)=hn;
```

```
x2(1,1:N)=Spot_N;
```

```
R_M=zeros(L); %构造循环矩阵
```

```
for j=1:L
```

```
R_M(j,:)=circshift(x1,[0,j-1]); %列移动0, 行移动j-1
```

```
end
```

```
R_M=R_M';x2=x2';
```

```
S1 =(R_M*x2)';
```

```
for i=1:K-1
```

```
S1(i)=S1(i)/i;
```

```
end
```

```
S1(K:L)=S1(K:L)/K;
```

```
% S1 =(R_M*x2)'/K; %这里有问题, 应该是用了几个数据除以几
```

```
S=S1(1:N); %舍去末尾,理论上末尾也应该像开头一样处理
```

```
end
```

%无法实时得到滤波结果。实时运算无法用循环卷积。第一个数据用第一个的,第二个用前两个的均值,这样直到第五个数据。

%循环卷积的开头和末尾都是不完全的,我的

做法是去尾,这样开头的几个数据是不完全的

DDFFT.m: (用 DFT 矩阵计算 DFT)

```
function X = DDFFT(Data,N) %a为
传进来的时域序列, N为点数 Data为列向量
DN=ones(N,N); %构造DFT矩阵
for n=1:N %从N=1开始, 第一行第一列
频率为0, 值为1 %为加快运算速度, 针对
第三次实验进行优化
```

```
for k=n:N
```

```
DN(n,k)=getWN((n-1)*(k-1),N);
```

```
DN(k,n)=getWN((n-1)*(k-1),N);
```

```
end
```

```
end
```

```
Size=size(Data);
```

```
if N>Size %若N>数据长度, 补零
```

```
Dataplus=zeros(N,1);
```

```
Dataplus(1:Size)=Data;
```

```
else
```

```
Dataplus=Data;
```

```
end
```

```
X=DN*Dataplus; %DFT变换
```

```
end
```

ruan2.m (主函数)

```
clc;clear;
```

```
Data=load('Sunspot20190525.txt')
;
```

```
Spot_N=Data(:,4);
```

```
[N,~]=size(Spot_N);
```

```
time=datetime(Data(:,3),Data(:,2),1);
```

```
figure(1);
```

```

stem(time,Spot_N,'. ');
axis([638811,736665,0,420]);
dateaxis('x',10);
title('太阳黑子数目随时间的变化
'),xlabel('时间t (年, 月)
'),ylabel('太阳黑子数x(t) (个)
');grid on;

K=5;
S1=juanji(K,Spot_N); %调用自己写的
卷积函数
figure(2);
subplot(2,1,1);
plot(time,S1);
axis([638811,736665,0,350]);
dateaxis('x',10);
title('太阳黑子数目随时间的变化, 宽度
为5, 滑动平均'),xlabel('时间t (年)
'),ylabel('太阳黑子数x(t) (个)
');grid on;

K=13;
S2=juanji(K,Spot_N); %调用自己写的
卷积函数
% figure(3);
subplot(2,1,2);
plot(time,S2);
axis([638811,736665,0,350]);
dateaxis('x',10);
title('太阳黑子数目随时间的变化, 宽度
为13, 滑动平均'),xlabel('时间t (年,

```

```

月) '),ylabel('太阳黑子数x(t) (个)
');grid on;

X=DDFFT(Spot_N,N);
% X=fft(Spot_N);
n=0:2/N:2-2/N;
figure(3);
X(1)=0; %频率为0处的数据无意义,周期无
限大
plot(n,abs(X));title('太阳黑子DTFT
变换的幅频特性
'),xlabel('ω/π'),ylabel('幅度
X(e^jω)');grid on;

XA=X(1:N/2)*(1); % 黑子数目
na=0:2/N:1-2/N;
for i= 1:N/2;
nat(i)=2/(na(i));
end
figure(4);
plot(nat,abs(XA));title('具有物理
意义的太阳黑子DTFT变换的幅频特性
'),xlabel('月'),ylabel('幅度
X(e^jω)');grid on;

[m,i]=max(abs(X(1:N/2)));
F=n(i);
Y=2/(F);

%w/π=1处对应着Fs/2=1/2月
%幅值最大处0.01493, 故周期为2/
(0.01493*12) =11.16年

```

2. 实验典型界面

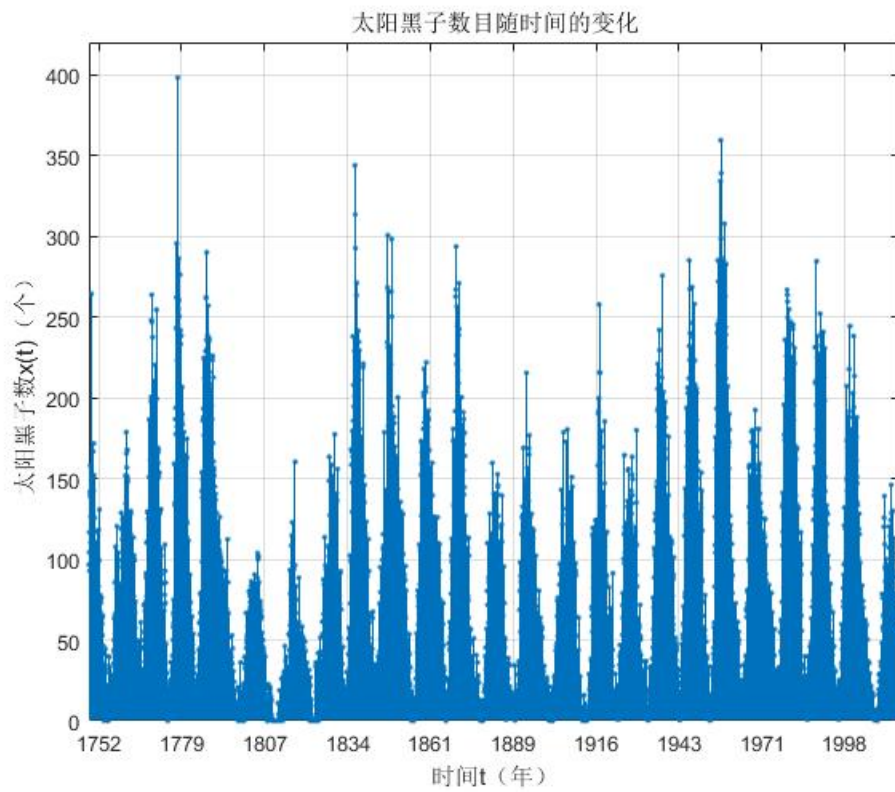


图 2.1 太阳黑子数目随时间变化图

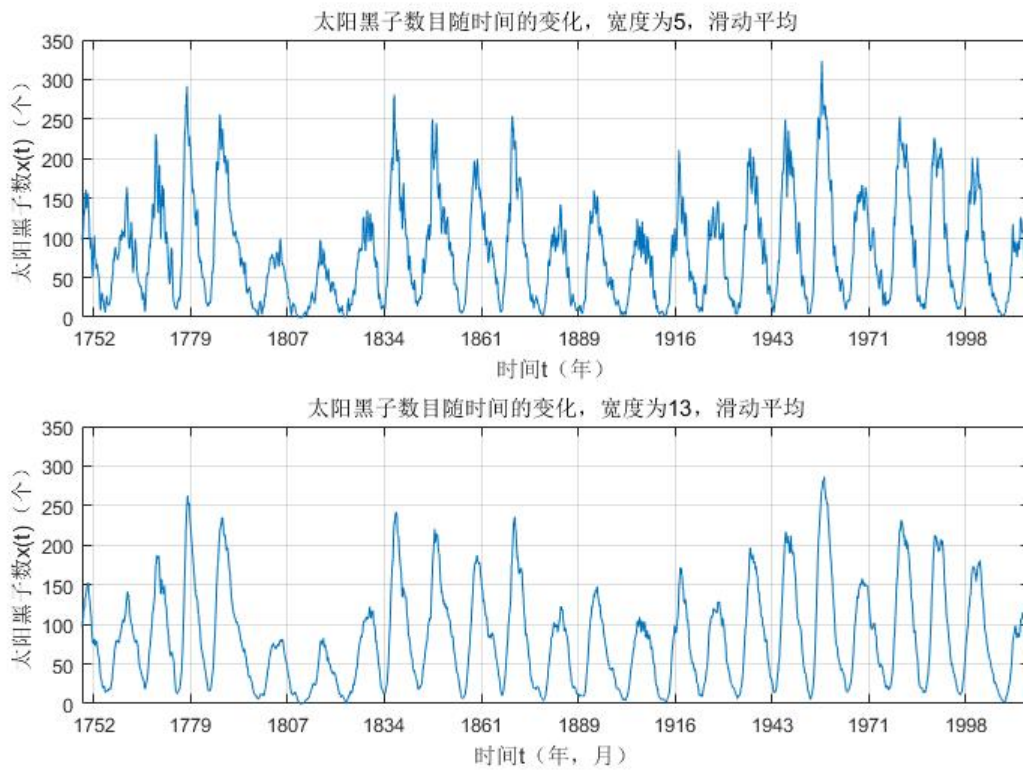


图 2.2 滑动平均处理后的太阳黑子数目随时间变化图

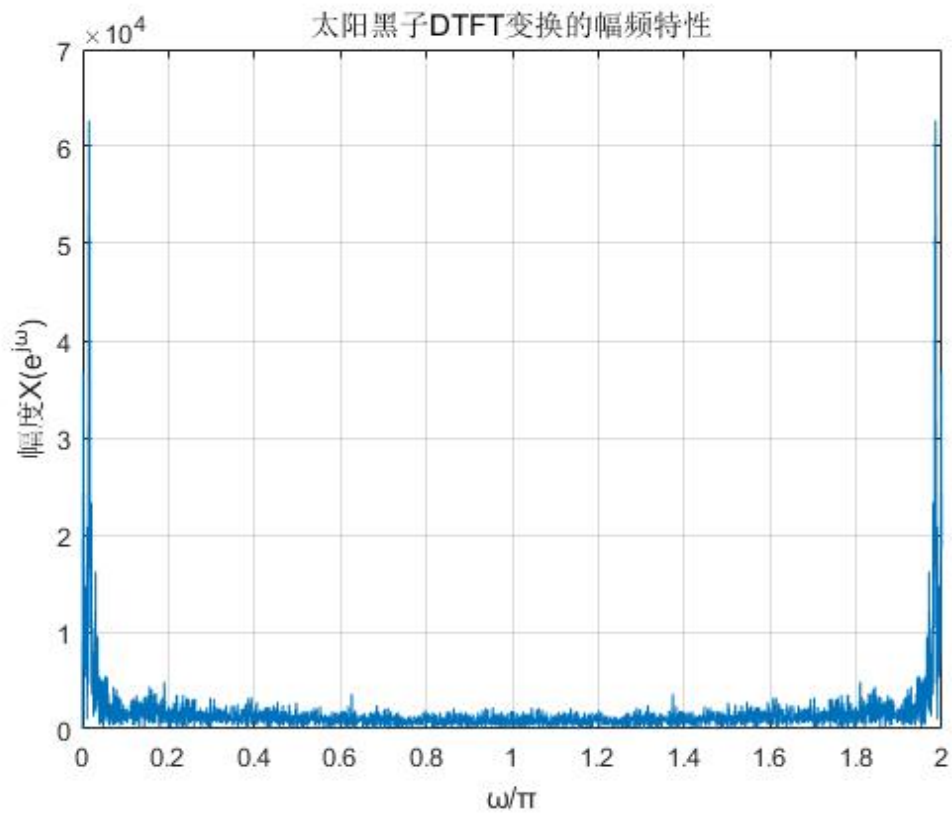


图 2.3 太阳黑子经 DFT 变换得到的数字频谱

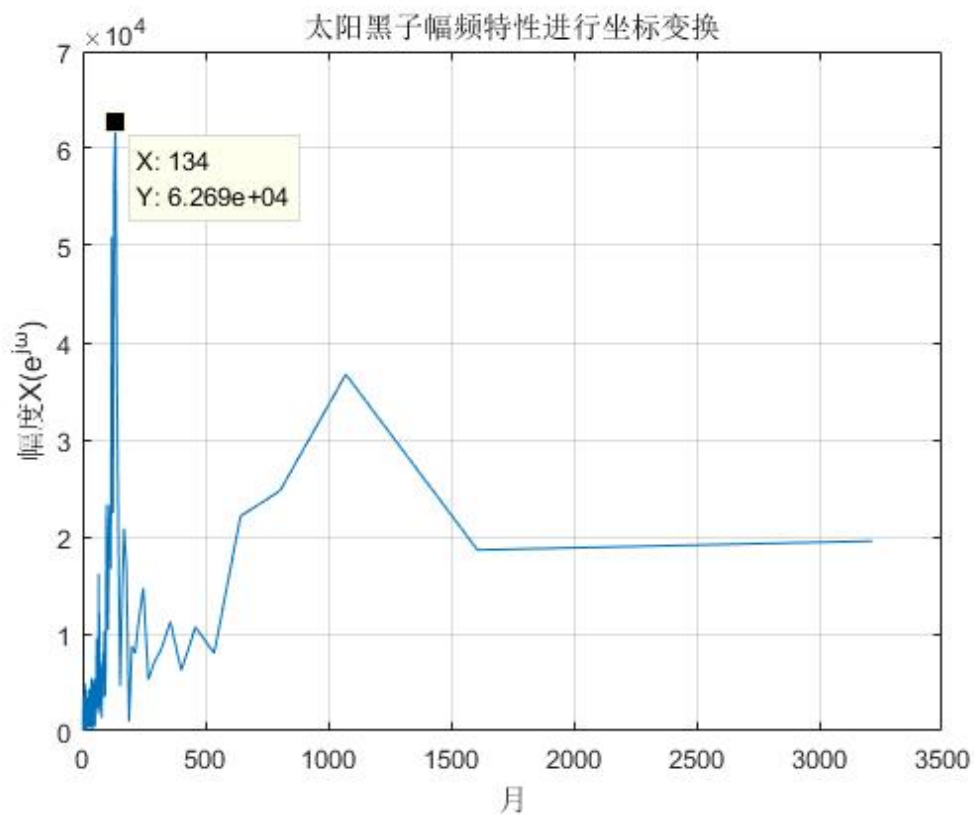


图 2.4 为便于看出周期，将横坐标变为月份得到的频谱图

二、结果分析

1. 实验结果及分析

(1) 本实验的数据来自于 1749 年 1 月至 2017 年 6 月观测得到的太阳黑子数目序列。通过对观测序列进行 DSP 分析，可以观察到太阳黑子数目变化的周期，从而可以进一步研究太阳活动与自然现象之间的联系。本实验所用数据中，第二列为月份，第三列为年份，第四列为这一个月内的太阳黑子数目平均值，其余数据无用。本题一定要通过 `datenum` 数据类型将年份和月份联系在一起，使用年份为横坐标作图。

(2) 太阳黑子的时域波形（图 2.1）非常混乱，经过滑动平均滤波后可以看到，时域波形滤除了一部分噪声干扰，平滑了许多，但一些本来陡峭的部分也被磨平了，如图 2.2。在我的程序中，使用循环卷积方法进行线性卷积，求得滤波序列的循环卷积矩阵，与待处理信号进行矩阵相乘，实现滑动平均滤波。我一开始处理的程序出现了两个问题：

1). 对于两端无法使用足够数据的卷积结果，我均按窗口宽度进行平均，造成结果过小。正确的方法是，采用了几个数据，就平均几个数据。

2). 如果考虑数据处理的实时性，应该用线性卷积定义编程，并保留开头结果，舍去末尾的多余数据。第一个数据即为第一个信号，第二个数据为前两个信号的平均，以此类推直至达到窗口宽度。我之前的程序去除了开头的结果，保留了末尾的多余数据，没有考虑实时性。经过老师的提醒，我修改了这两个问题。

(3) 太阳黑子的时域波形（图 2.1）非常混乱，难以看出规律。进行 DFT 变换后，由图 2.3 可以看到 $0-\pi$ 的一个峰值，但依然不够直观。之后我们想将横坐标通过公式 $\frac{1}{T} = f = \omega * \frac{F_s}{2\pi}$ 从数字角频率变为时间，但 MATLAB 会自动按横坐标的大小进行排序，因此频谱图变为图 2.4。可以看到幅值最大点的横坐标为 134 个月，故太阳黑子爆发的周期为 134 个月 ≈ 11.17 年。

2. 本次实验收获、体会及建议

经过这次实验，我掌握了滑动平均滤波的使用，同时熟练使用频谱特性对信号进行处理分析，体会到了频率域分析的神奇所在。另外我还锻炼了 `matlab` 编程技巧。同时我发现，实验内容与平时作业有很多重复的地方，可以直接调用平常写的程序。

实验三 受谐波噪声干扰的数字心电信号图（ECG）滤波技术

实验时间：2019.06.01

同组同学：李世焱

一、实验结果

1. 实验程序代码

%实验：处理心电信号，去掉工频干扰

```
clear;clc;
```

```
Data=load('Ecginf041.txt');
```

```
F=360; %Sample frequency
```

```
T=60; %Sample time in total
```

```
%%
```

%原信号时域波形

```
time=1/F:1/F:T;
```

```
data=Data(1:F*T,4);
```

```
figure(1);
```

```
plot(time,data,'-');title('数字心  
电图时域波形'),xlabel('时间
```

```
t/s'),ylabel('心电信号x(t)');grid  
on;
```

```
%%
```

%原信号频域波形

```
[N,~]=size(data);
```

```
% X=DDFFT(data,N); %data必须为列  
向量，速度太慢
```

```
X=fft(data,N);
```

```
Xa=X(1:N/2).*(1/F); %幅值改变
```

```
Xa(1)=0; %去除直流分量
```

```
ff=0:F/N:F/2-F/N;
```

```
%%
```

%低通滤波器

```
fc=40;
```

```
wc=fc*pi/(F/2);
```

```
L=51; %窗口长度
```

```
Nw=L-1;
```

```
for i=1:L
```

```
wn(i)=(0.54-  
0.46*cos(2*pi*i/(L-  
1))); %hamming窗
```

```
end
```

```
wn=wn';
```

```
% wn=hamming(L);
```

```
n=0:1:Nw;
```

```
hd=sin(wc*(n-Nw/2))./(pi*(n-  
Nw/2)); %理想时域信号。奇数，故中间  
那个是无穷，一定要洛必达法则改了中间  
的值
```

```
hd(Nw/2+1)=wc/pi;
```

```
H=hd.*wn';
```

```
% H=firl(Nw,wc/pi,'low',hamming  
(L));
```

```
[hf,wf]=freqz(H,1,N,'whole');
```

```
figure(3);
```

```
%
```

```
plot(wf/pi,20*log10(abs(hf)));xl  
abel('归一化频率
```

```
 $\omega/\pi$ '),ylabel('20lg|H( $\omega$ )|'),title  
('矩形窗FIR幅频响应');grid on;
```

```
plot(wf(1:N/2)/pi*F/2,20*log10(a  
bs(hf(1:N/2))));xlabel('频率
```

```
/Hz'),ylabel('20lg|H( $\omega$ )|'),title  
('Hamming窗FIR幅频响应');grid on;
```

```
%%
```

%频域滤波

```
FFT=fft(H,N);
```

```
XN=FFT.*X'; %fft是对的。
```

XaN=hf.*X; %使用freqz的数据。由于相
位的问题，一定要整体滤波（0-2pi），千
万不能用0-pi滤波后对称补上另一半

```
figure(4);
```

```
subplot(2,1,2);
```

```
plot(ff,abs(XaN(1:N/2)/F));title  
('FIR滤波后数字心电图实际频谱
```

```
'),xlabel('频率f/Hz'),ylabel('幅度  
X_aN(jf)');grid
```

```
on;axis([0,180,0,5]);
```

```

subplot(2,1,1);
plot(ff,abs(Xa));title('数字心电图
实际频谱'),xlabel('频率
f/Hz'),ylabel('幅度X_a(jf)');grid
on;axis([0,180,0,5]);
%%
%时域波形
datap1=ifft(XaN);
datap2=ifft(XN);
datas=conv(H,data); %时域滤波

figure(5);
subplot(3,1,1);
plot(time,datap1);title('freqz频
域滤波后数字心电图时域波形'),xlabel('
时间t/s'),ylabel('心电信号
x(t)');grid on;
subplot(3,1,2);
plot(time,datap2);title('fft频域
滤波后数字心电图时域波形'),xlabel('时
间t/s'),ylabel('心电信号
x(t)');grid on;
subplot(3,1,3);
plot(time,datas(1:F*T));title('时
域滤波后数字心电图时域波形'),xlabel('
时间t/s'),ylabel('心电信号
x(t)');grid on;

```

2. 实验典型界面

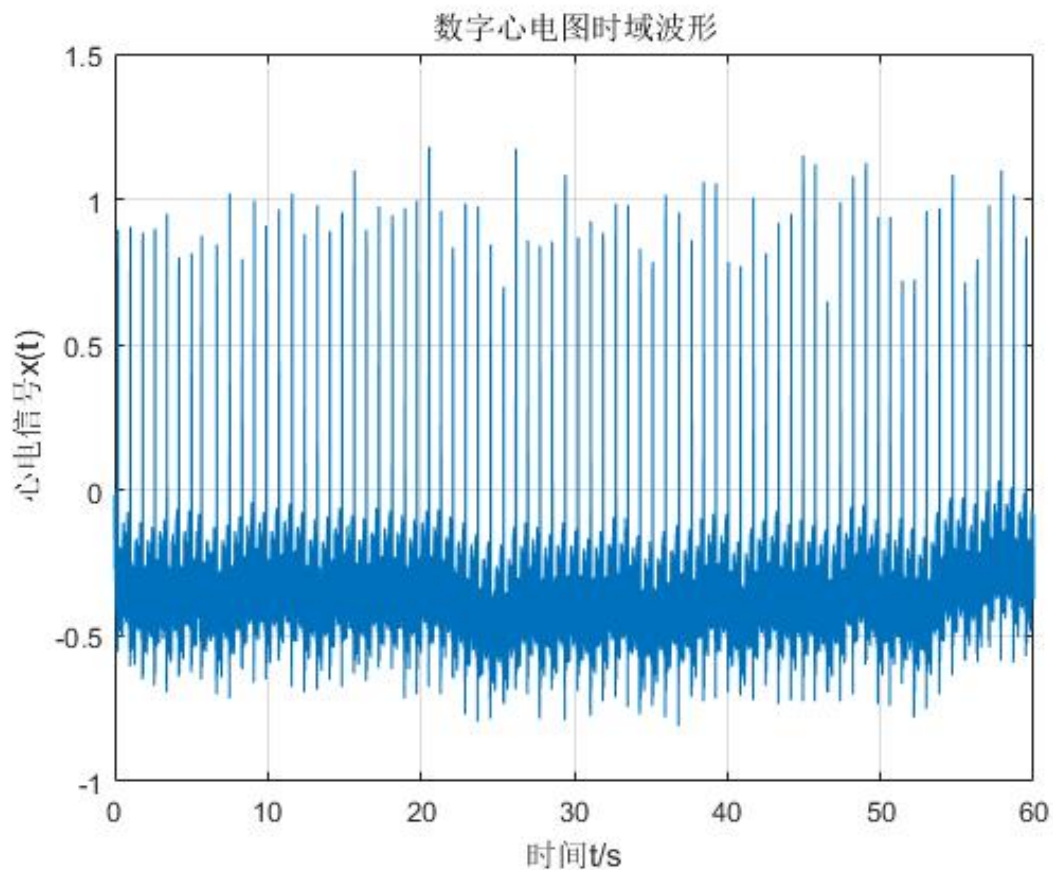


图 2.1 数字心电图时域波形

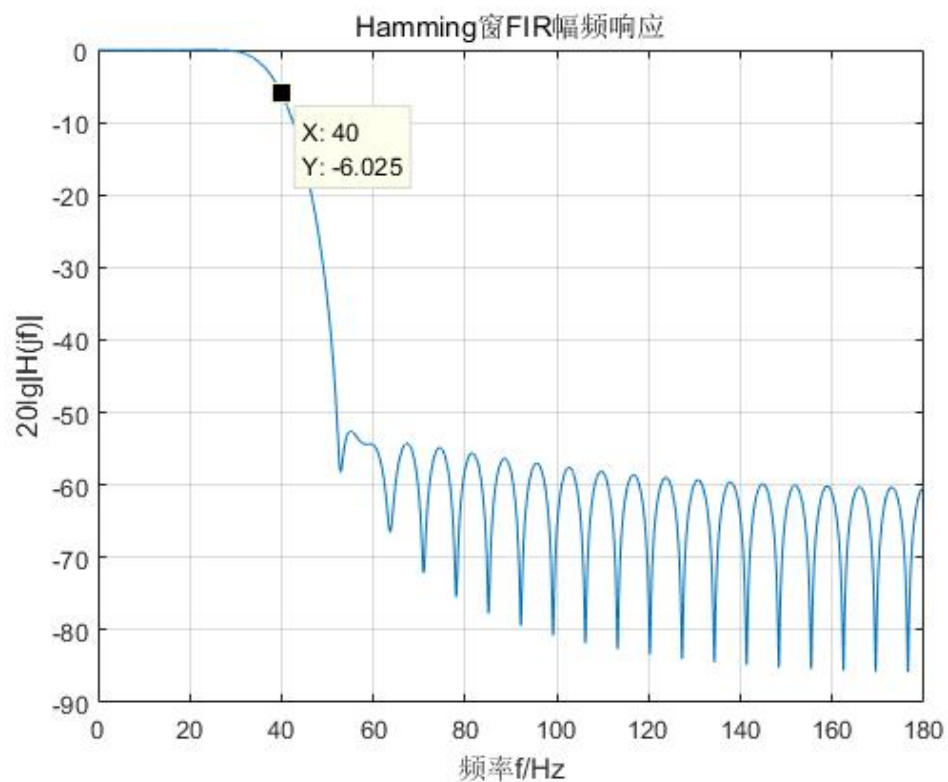


图 2.2 Hamming 窗 FIR 幅频响应

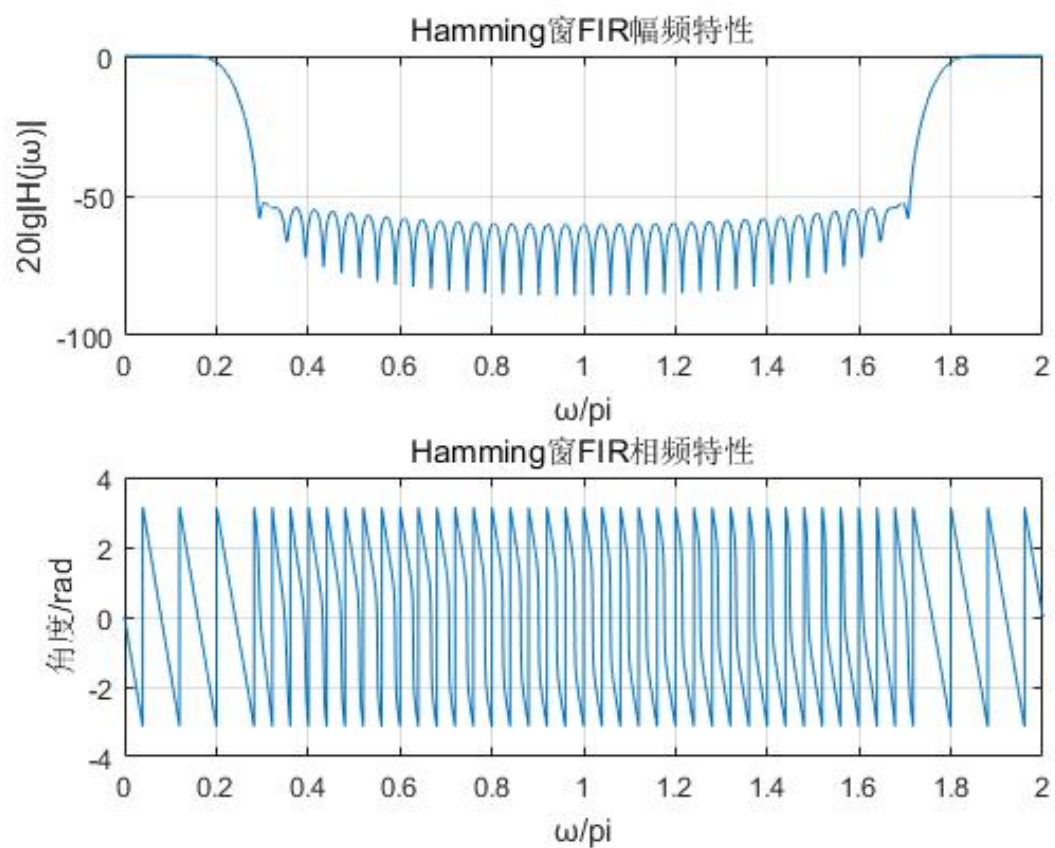


图 2.3 完整的 Hamming 窗 FIR 幅频、相频响应

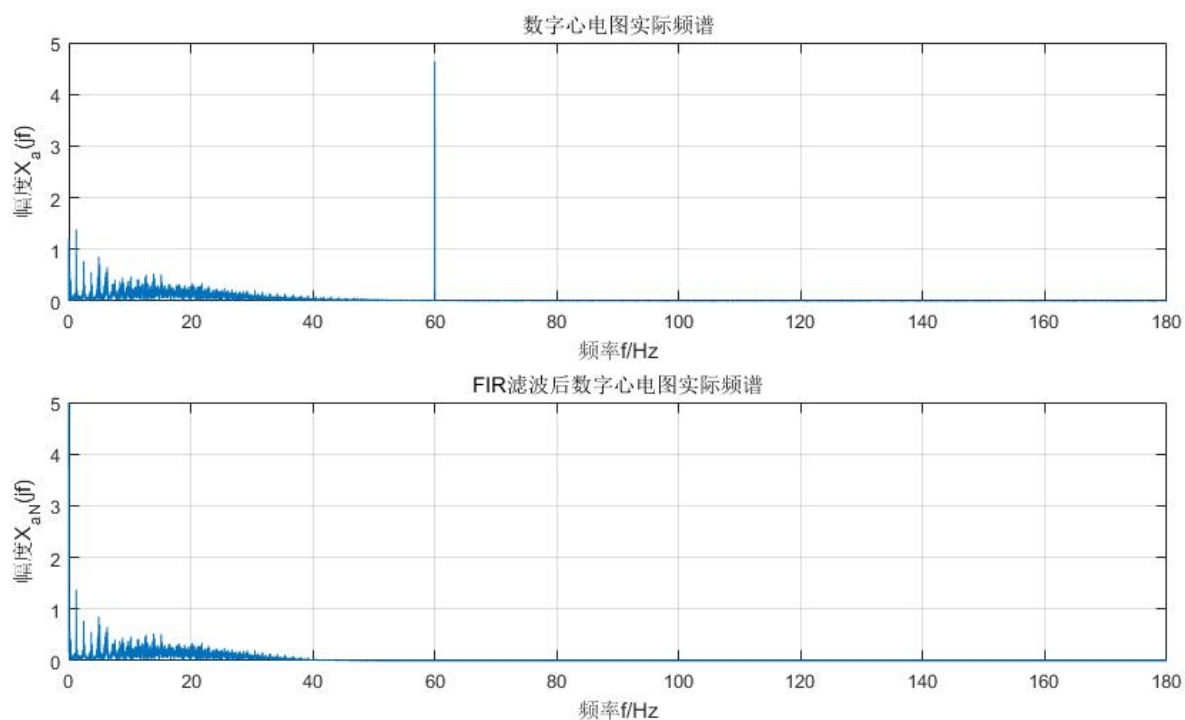


图 2.4 滤波前后的数字心电图模拟频谱

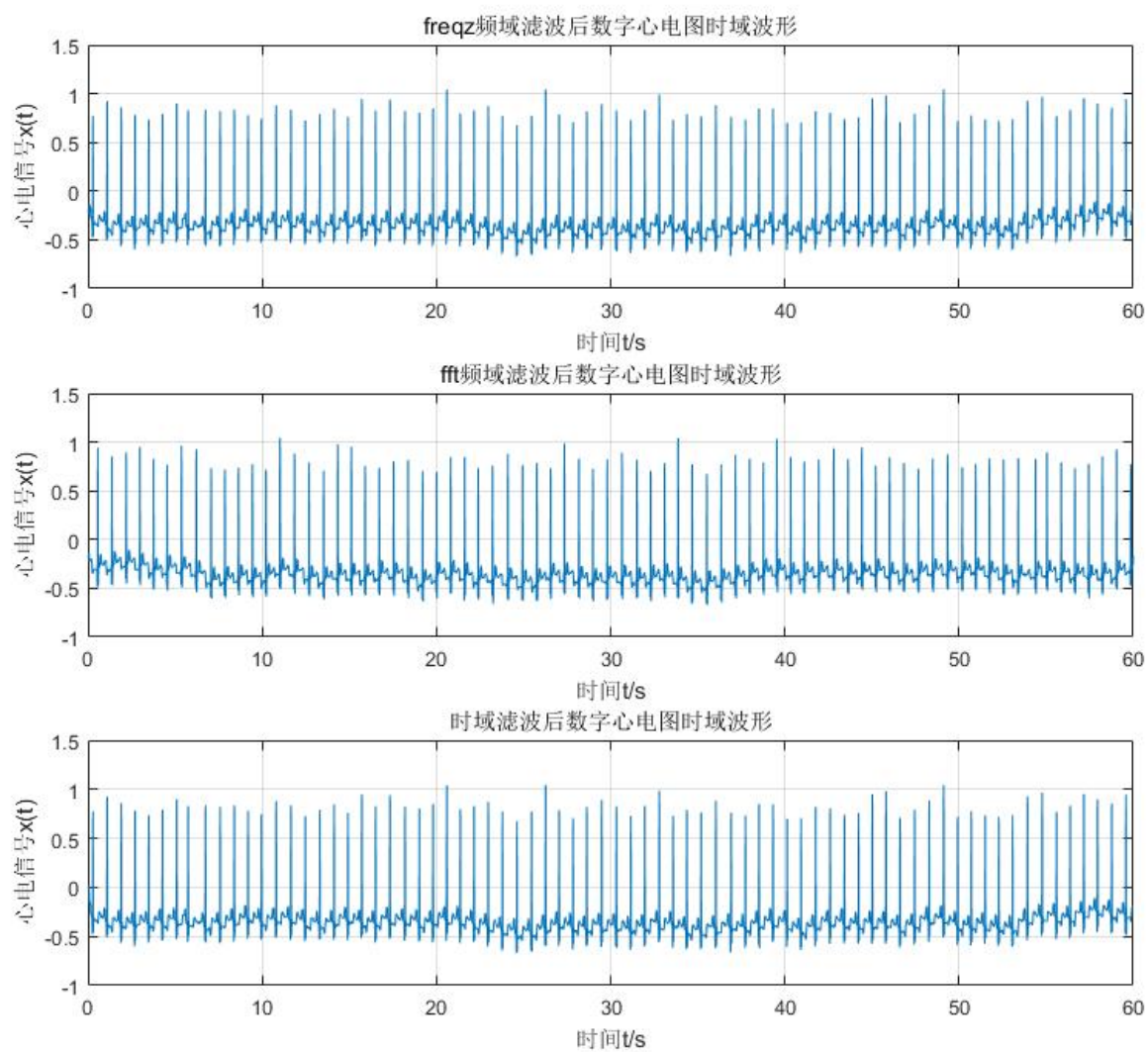


图 2.5 使用频域滤波和时域滤波后得到的时域波形

注: freqz 表示通过 freqz()函数得到窗函数的频域值, fft 表示通过 fft()函数得到窗函数的频域值, 这里没有用自己写的函数, 因为自己写的函数计算太慢了。

二、结果分析

1. 实验结果及分析

(1) 本次实验背景来自于心电图机记录的心肌激动产生的生物电信号, 它可以为临床诊断和科研提供有效地依据。由于生物电信号非常微弱, 非常容易受到电网的干扰。这种干扰被称为工频干扰, 频率往往是电网频率的奇数倍。在我国, 工频信号的干扰集中在 50Hz, 美国为 60Hz。工频干扰常常使用梳状滤波器进行去除。本实验原始数据采样频率 360Hz, 采样时间 60s, 共 $360 \times 60 = 21600$ 个数据。绘制时域波形如图 2.1, 可以看到心电图波形被掩盖在干扰中, 使医生难以诊断。

(2) 使用 fft() 函数计算原信号的频谱特性 $X(j\omega)$ 如图 2.4 上图, 可以看到在 $f=60\text{Hz}$ 处有很高的值, 这就是工频干扰, 与题设背景相符。

(3) 根据公式 $\omega_c = f_c * \frac{2\pi}{F_s}$, 得到截止频率 $\omega_c = \frac{2\pi}{9}$, 长度为 $L=51$, 阶数为 $N=L-1=50$ 。FIR 窗函数设计方法如下:

首先根据 Hamming 窗公式 $w[n] = \frac{1}{2} \left[1 - \cos\left(\frac{2\pi n}{N-1}\right) \right] R_N(n)$ 计算得到时域下窗

信号 $w(n)$, 之后根据理想时域信号公式 $h_d[n] = \frac{\sin(\omega_c(n-\tau))}{\pi(n-\tau)}$, $\tau = N/2$ 计算出

$H_d(n)$, 最后经过时域乘积, $h[n] = w[n] \cdot h_d[n]$, 得到 FIR 滤波器的时域形式 $H(n)$ 。

之后需要求 $H(n)$ 的 DFT 变换 $H(j\omega)$, 进行频域滤波, 如图 2.2。从频域图可以看出, 滤波器在 $f=40\text{Hz}$ 的地方衰减 6dB, 满足截止频率设计要求。自己写的函数计算 DFT 速度太慢, 舍弃不用。这里用 freqz 函数和 fft 函数均可求得 DFT。Freqz 函数和 fft 函数的区别在于: freqz 根据公式计算 DFT, 得到的值要更准确一些; 同时 freqz 默认为计算 $0-\pi$ 范围, 如果计算 N 点全范围, 需要写语句 `[hf,wf]=freqz(H,1,N,'whole');`

之后根据公式 $X_a[j\omega] = X[j\omega] \cdot h[j\omega]$, 频域点乘, 进行频域滤波。需要注意的是, 这里采用的频谱特性 $X[j\omega]$, $h[j\omega]$ 均为 $0-2\pi$ 范围。如果要显示实际采样频率, 需要进行坐标变换 $f = \omega * \frac{F_s}{2\pi}$, 结果如图 2.4 下图。

最后将 $X_a[j\omega]$ 进行 ifft 变换即可得到滤波后的时域信号, 如图 2.5 的上、

中图。

(4) 最后通过时域卷积进行时域滤波, 如图 2.5 下图。比较时域滤波与频域滤波的结果可知, 通过 `freqz` 函数计算 DFT 得到的最终结果与时域滤波完全一样, 通过 `fft` 函数计算 DFT 最终得到的结果有微小差异, 这可能是因为快速算法产生很小的误差。

由于 `freqz` 函数默认得到的是 $0-\pi$ 范围的频域响应, 我一开始的做法是: 将这部分响应与 FIR 滤波器 $0-\pi$ 范围的响应相乘, 再将结果沿 π 对称, 进行 `ifft`。最后结果出现严重失真。由于书上的滤波器特性都是幅值响应, 我误认为整体是偶对称的, 但从图 2.3 可以看出, 相频响应是奇对称的, 因此滤波时一定要将频谱整体在频域上相乘, 一定不能乘完再对称。

2. 本次实验收获、体会及建议

经过这次实验, 我掌握了 FIR 滤波器的设计、使用过程, 更重要的是纠正了一个错误概念。DSP 实验就此全部做完, 我受益匪浅, 感谢袁老师三个星期的付出! 如果对报告有任何问题, 请老师联系我, 手机: 15652587808, E-mail: lijinjie362@outlook.com, 谢谢老师!