

FIR 滤波器设计

1.设计原理

设计数字滤波器的任务就是寻求一个因果稳定的线性时不变系统，并使系统函数 $H(z)$ 具有指定的频率特性。

与 IIR 滤波器相比，FIR 滤波器在保证幅度特性满足技术要求的同时，很容易做到严格的线性相位特性。稳定和线性相位特性是 FIR 滤波器突出的优点。

FIR 滤波器的设计方法有多种，如窗函数法、频率采样法及其它各种优化设计方法。窗函数法是使用矩形窗、三角窗、巴特利特窗、汉明窗、汉宁窗和布莱克曼窗等设计出标准响应的高通、低通、带通和 FIR 滤波器。

2. 相关函数

◆ 语音信号的读取和播放；

(1) 将给定的语音信号存放在 MATLAB 的工作目录下，可用 **wavread** 读取，用 **sound** 播放；

(2) 用 **audioread()** 函数读取电脑中的音频文件，用 **audioplayer(Y,FS); play()** 播放；

```
[Y,FS] = audioread('filename.mp3');
```

(Y 保存音频信号数据，FS 是音频采样率，MP3 格式的采样率一般为 44100,)

```
[Y,FS] = audioread('1.mp3');
```

p = audioplayer(Y,FS); %播放读入的数据

play(p);

补充：如果播放文件较长，不想播完，可用 clear sound 停止播放

◆ Matlab 中关于 FIR 滤波器设计的命令

(1) fir1(数字1) 函数

在 MATLAB 下设计标准响应 FIR 滤波器可使用 fir1 函数。fir1 函数以经典方法实现加窗线性相位 FIR 滤波器设计，它可以设计出标准的低通、带通、高通和带阻滤波器。fir1 函数的用法为：

b=fir1(n,Wn,'ftype',Window)

各个参数的含义如下：

b—滤波器系数。对于一个 n 阶的 FIR 滤波器，其 n+1 个滤波器系数可表示为：

$$b(z)=b(1)+b(2)z^{-1}+\dots+b(n+1)z^{-n}。$$

n—滤波器阶数。

Wn—截止频率， $0 \leq Wn \leq 1$ ， $Wn=1$ 对应于采样频率的一半。当设计带通和带阻滤波器时， $Wn=[W1 \ W2]$, $W1 \leq \omega \leq W2$ 。

ftype—当指定 ftype 时，可设计高通和带阻滤波器。Ftype=high 时，设计高通 FIR 滤波器；

ftype=stop 时设计带阻 FIR 滤波器。

低通和带通 FIR 滤波器无需输入 ftype 参数。

Window—窗函数。窗函数的长度应等于 FIR 滤波器系数个数，即阶

数 $n+1$ 。默认汉明窗 (Hamming)。

(2) 窗函数的使用

在 MATLAB 下, 这些窗函数分别为:

产生窗函数的文件有八个:

1. bartlett (三角窗);——两端为零
2. blackman (布莱克曼窗);
3. boxcar (矩形窗);
4. hamming (哈明窗);
5. hanning (汉宁窗);
6. triang (三角窗);——两端不为零
7. chebwin (切比雪夫窗);
8. kaiser (凯赛窗);

例: $w = \text{boxcar}(n)$, 产生一个 n 点的矩形窗函数。

[例]设计一个长度为 8 截止频率为 0.4π 的线性相位 FIR 滤波器。

用矩形窗:

```
Window=boxcar(8);
```

```
b=fir1(7,0.4,Window);
```

用 blackman 窗:

```
Window=blackman(8);
```

```
b=fir1(7,0.4,Window);
```

[例] 设计线性相位带通滤波器，其长度 $N=16$ ，上下边带截止频率分别为 $W1=0.3\pi$ ， $W2=0.5\pi$

```
Window=blackman(16);
```

```
b=fir1(15,[0.3 0.5],Window);
```

(3) N 的计算

$$N = \frac{(\Delta B)}{\omega_s - \omega_p}$$
, ΔB 为过渡带宽，根据窗函数的类型查表获得

ceil(x)：大于 x 的最小整数

$$\omega_n = (\omega_p + \omega_s) / 2$$

实验步骤：

(1) 语音信号的频谱分析

首先画出语音信号的时域波形；然后对语音号进行快速傅里叶变换，得到信号的频谱特性。

程序如下：

```
clc;clear;close all;
```

```
fs=44100; %语音信号采样频率为22050
```

```
x1=wavread('我的祖国原.wav'); %读取语音信号的数据，赋给变量 x1
```

```
sound(x1,fs); %播放语音信号
```

```
y1=fft(x1,1024); %对信号做1024点 FFT 变换
```

```
f=fs*(0:511)/1024;
```

```
figure(1);
```

```
plot(x1) %做原始语音信号的时域图形
```

```
title('原始语音信号');xlabel('time n');ylabel('fuzhi n');
```

```
figure(2);freqz(x1) %绘制原始语音信号的频率响应图
```

```
title('频率响应图')
```

```
figure(3);subplot(2,1,1);
```

```
plot(abs(y1(1:512))) %做原始语音信号的 FFT 频谱图
```

```
title('原始语音信号 FFT 频谱');
```

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

```
plot(f,abs(y1(1:512)));
```

```
title('原始语音信号频谱')
```

```
xlabel('频率/Hz');ylabel('幅值');
```

(2) 给原始的语音信号加高频余弦噪声，频率为5kHz。

画出加噪后的语音信号时域和频谱图，与原始信号对比，可以很明显的看出区别。

程序如下：

```
X1=fft(x1);
```

```
L=length(X1);
```

```
k=0:L-1;
```

```
c1=0.1*sin (2*pi*6000*k/fs);
```

```
x2=x1+c1';
```

```
sound(x2,44100)
```

```
y2=fft(x2,1024);
```

```
figure(1);
```

```
plot(k,x2);
```

```
title('加噪后的信号');
```

```
xlabel('time n');
```

```
ylabel('fuzhi n');
```

```
figure(2);
```

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

```
plot(f,abs(y1(1:512)));
```

```
title('原始语音信号频谱');
```

```
xlabel('频率/Hz');ylabel('幅值');hold on
```

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

```
plot(f,abs(y2(1:512)));
```

```
title('加噪后的信号频谱');
```

```
xlabel('频率/Hz');ylabel('幅值');hold on
```

(3) 设计 **IIR** 滤波器进行滤波（相关函数见上次实验函数说明）

(4) 设计 **FIR** 滤波器进行滤波：

程序如下：

```
wp=0.15*pi;%实际的通带截止频率为(wp/pi)*fs/2
```

```
ws=0.2*pi;
```

```
wdelta=ws-wp;
```

```
N=.....; % 后面程序自己设计，可用 ceil 取整
```

窗口函数名称↵	旁瓣峰值衰减↵ (dB) ↵	主瓣宽度↵	阻带最小衰减↵ (dB) ↵	过渡带宽↵
矩形窗↵	13↵	$4\pi/N$ ↵	21↵	$1.8\pi/N$ ↵
汉宁窗↵	32↵	$8\pi/N$ ↵	44↵	$6.2\pi/N$ ↵
哈明窗↵	43↵	$8\pi/N$ ↵	53↵	$6.6\pi/N$ ↵
布拉克曼窗↵	58↵	$12\pi/N$ ↵	74↵	$11\pi/N$ ↵
三角形窗↵	27↵	$8\pi/N$ ↵	25↵	$4.2\pi/N$ ↵
凯塞窗↵	65↵	$12\pi/N$ ↵	80↵	$10\pi/N$ ↵

补充：函数 filter

函数 filter 的调用格式为 $y = \text{filter}(b,a,x)$

该格式采用数字滤波器对数据进行滤波，既可以用于 IIR 滤波器，也可以用于 FIR 滤波器。其中向量 b 和 a 分别表示系统函数的分子、分母多项式的系数，若 $a=1$ ，此时表示 FIR 滤波器，否则就是 IIR 滤波器。该函数是利用给出的向量 b 和 a，对 x 中的数据进行滤波，结果放入向量 y。