MinMax Filter Implementations

- Frequency Response, Impulse Response, Maximal Errors

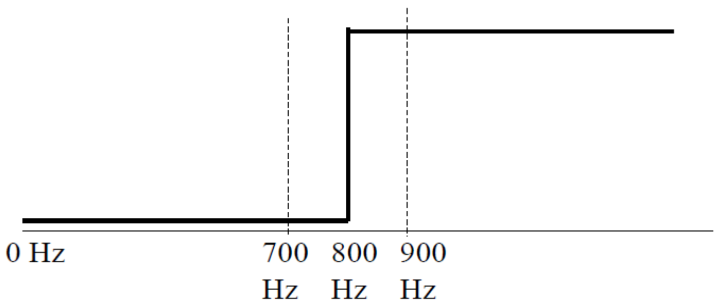
(1) Design a Mini-max **lowpass** FIR filter such that (40 scores)

① Filter length = 19, ② Sampling frequency f_s = 4000Hz,

③ Pass Band 800~2000Hz ④ Transition band: 700~900 Hz,

⑤ Weighting function: $W(F)$ = 1 for passband, $W(F)$ = 0.5 for stop band .

⑥ Set Δ = 0.0001 in Step 5.



(sol)

(b) the frequency response

1. Given that **Sampling freq.=** ***4000Hz*** , so I map the **freq. domain** to normalized ***F*-domain** , which has 2000 intervals, i.e. ***F = 0: 1/fs : 0.5***

2. The design process converges after ***6* iterations** , and the final Maximal error is ***0.050168***

3. The **intermediate** ***R(F)***, **impulse response** ***h[n]*** and **frequency response** ***H(f)*** of designed FIR filter are showed below.

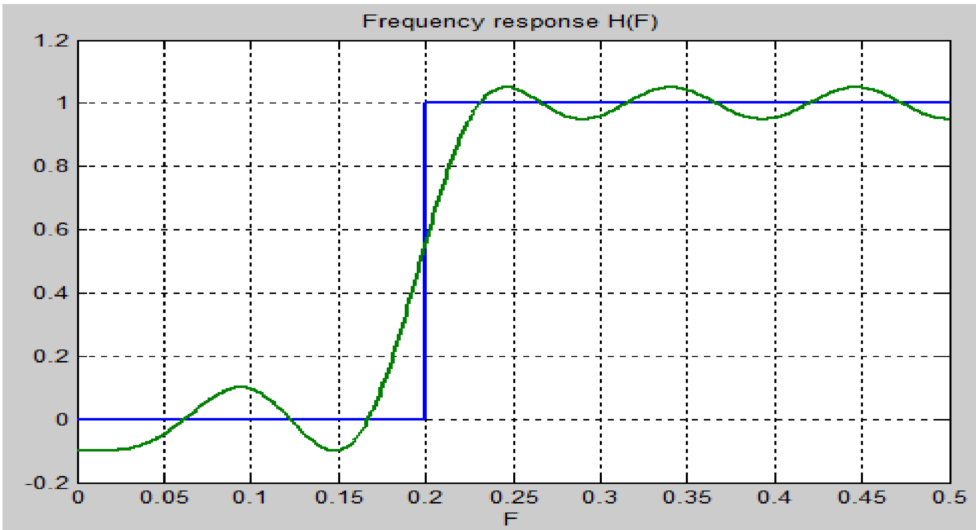


Figure *1*. Mini-Max FIR HPF & Ideal HPF

Figure *1*. shows the frequency response of our FIR filter.

Since I've assigned different weighting function to pass-band and stop-band , we could see that the amplitude ripples are different in these two bands.

For the Weight value of pass-band is larger , we could find that the amplitude ripple of the pass-band is smaller , and vice versa in the case of stop-band.

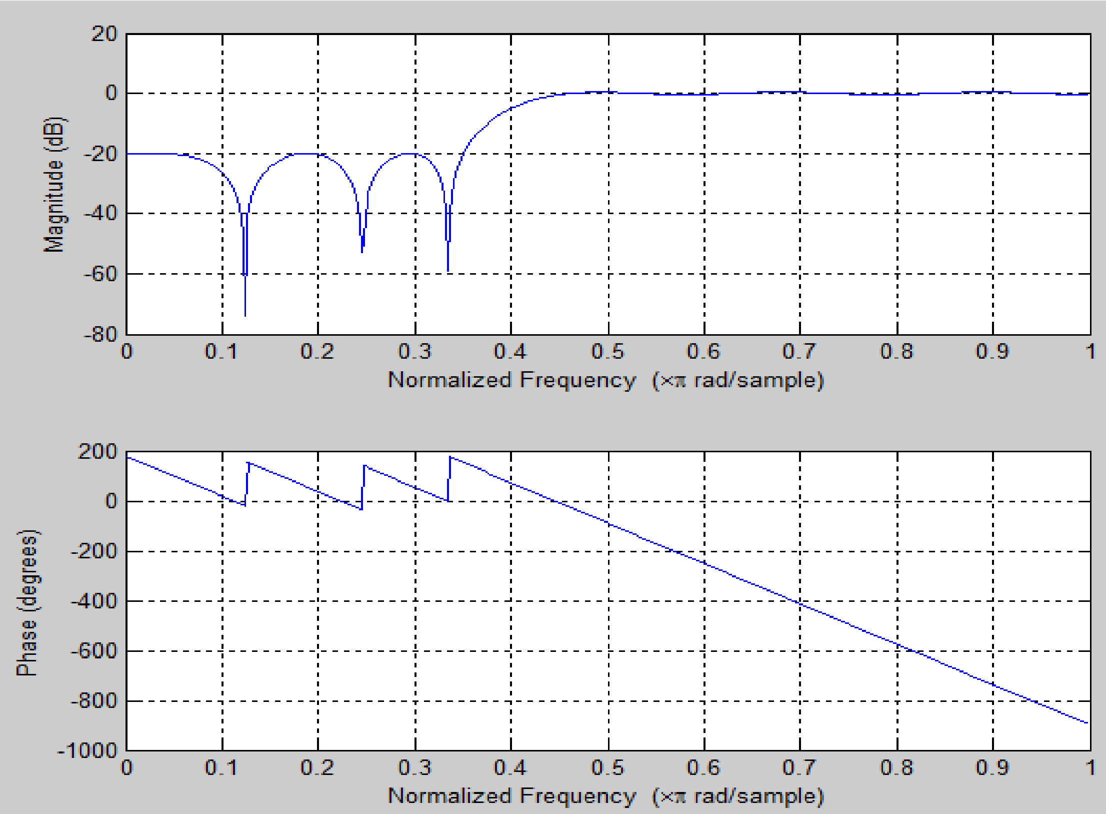


Figure 2. **The Amplitude & Frequency Response of h[n]**

(c) the impulse response $h[n]$

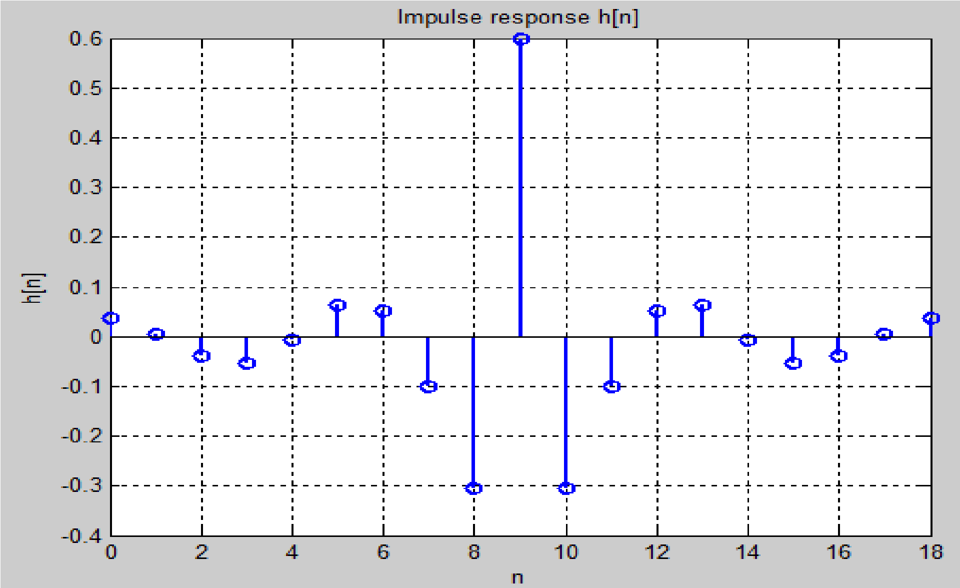


Figure *3*. Impulse Response of Mini-Max FIR HPF

Figure *3* is the Impulse response of the Mini-max FIR filter.

Given that h[n] has 19 points as we expected and the whole response is even-symmetric to the middle point n = 9.

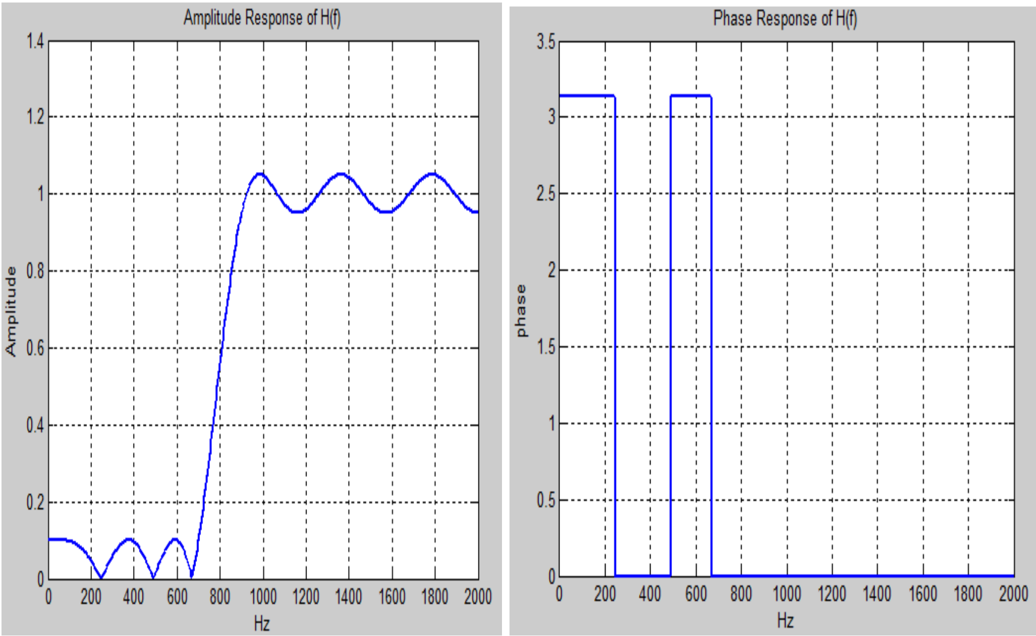


Figure *4.*Amplitude Response & Phase Response of Mini-Max FIR HPF

(d) the maximal error for each iteration.

Iteration	Maximal error
1	**0.191856**
2	**0.123296**
3	**0.069735**
4	**0.051108**
5	**0.050177**
6	**0.050168**

Table 1. The Max. Error of each Iteration

The final max. error is the value of the ***6th*** iteration , for the reason that the difference of max. error between ***5th*** and ***6th*** is $|0.050168-0.050177| < delta = 0.0001$, so we conclude that the program stops at the ***6th*** iteration.

(a) the Matlab program

Editor - D:_home\Desktop\ADSP HW\HW1\HW1.m

HW1.m | MiniMaxFilter.m

This file can be published to a formatted document. For more information, see the publishing video or help.

```matlab
%% ADSP HW1
clc;clear;
%% Parameters
N = 19; k=(N-1)/2;      % Filter length N=19, k=(N-1)/2=9
fs = 4000;              % Sampling frequency = 4000Hz
delta = 0.001;
F = 0:delta:0.5; % Normalized frequency interval
Hd = [zeros(1, 0.2/delta) ones(1,0.3/delta+1)]; % Desired HPF  , Stop-band 0~800Hz <=> F: 0~ 0.2 (800/4000)
trans = [700/fs , 900/fs]; % Normalized transition band: (700~900)/fs
    % ==== Weighting function: W(F) = 1 for passband, W(F) = 0.5 for stopband.
W = [ 0.5*ones(1,round(trans(1)/delta+1)) zeros(1,round((trans(2)-trans(1))/delta)-1) ones(1,round((0.5-trans(2))/delta+1))]
    % ================================================================
Delta = 0.0001; %maxima error DELTA = 0.0001 in Step 5.

%% Step 1. Guessing (k+2) Extreme points
% N = 19, k = (N-1)/2 = 9, k+2 = 11 and exclude the transition band F=[0.175 0.225]
exf = [0.01 0.03 0.08 0.13 0.24 0.29 0.32 0.37 0.42 0.47 0.5]; % Random Assign by Myself
                                                      % Just avoid the transition band
```

HW1.m | MiniMaxFilter.m | +

This file can be published to a formatted document. For more information, see the publishing video or help.

```matlab
%% Output Function
% ========= Input =========
% F: Normalized Frequency Interval       Hd: Desired Fiter Hd(F)
% W: weighting function W(F)              N: filter length N
% exF: arbitrary extreme frequency      DEL: Set the maxima error DELTA in Step 5.
% ============= Output ========
% R: designed filter R(f)
% h: Impulse response h[n] of the designed filter R(f)
% MaxErrF: The maximal error for each iteration
n = 0:N-1;
[R, h, MaxerrF] = MiniMaxFilter(F,Hd,W,N,exf,Delta);

%% Frequency response (Normalized freq.)
figure(1);
plot(F,Hd,F,R,'linewidth',1.5);
title('Frequency response H(F)');
xlabel('F');
grid on;
```

HW1.m | MiniMaxFilter.m | +

This file can be published to a formatted document. For more in

```matlab
%% Impulse response h[n]
figure(2);
stem(n,h,'b','linewidth',1.5);
title('Impulse response h[n]');
xlabel('n');
ylabel('h[n]');
grid on;
%% Amplitude Response of H(f) (Hz)
figure(3);
plot(F*fs,abs(R),'linewidth',1.5);
title('Amplitude Response of H(f)');
xlabel('Hz');
ylabel('Amplitude');
grid on;
%% Phase Response of H(f) (Hz)
figure(4);
plot(F*fs,phase(R),'linewidth',1.5);
title('Phase Response of H(f)');
xlabel('Hz');
ylabel('phase');
grid on;
```

```
    disp('The maximal error for each iteration');
    disp(sprintf('Iteration \t Maximal error'));
for itera = 1:length(MaxerrF)
        disp(sprintf('\t%d\t\t\t%f', itera, MaxerrF(itera)));
end

```

(2) Suppose that $x[n] = y(0.0005n)$ and the length of $x[n]$ is 10000 and $X[m]$ is the FFT of $x[n]$. Find m_1 and m_2 such that $X[m_1]$ and $X[m_2]$ correspond to the 100Hz and 250Hz components of $y(t)$, respectively. (10 scores)

(ANS)

(3) In IIR filter design, why the step invariant method and the bilinear transform method can reduce the aliasing effect? (10 scores)

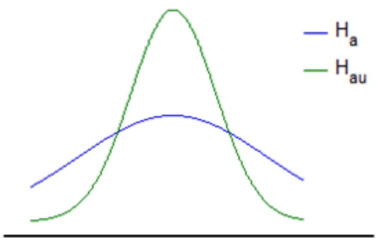
(ANS)

a. Step Invariant method

Do sampling on the impulse response of step function, just like:

$$h_{a,u}(t)=h_a(t)*u(t)=\int_{-\infty}^{\infty}h_a(\tau)u(t-\tau)d\tau=\int_{-\infty}^{t}h_a(\tau)d\tau$$

$$H_{a,u}(f)=\frac{H_a(f)}{j2\pi f}$$



So that, the high-freq. term will be compressed that we could avoid the aliasing effect

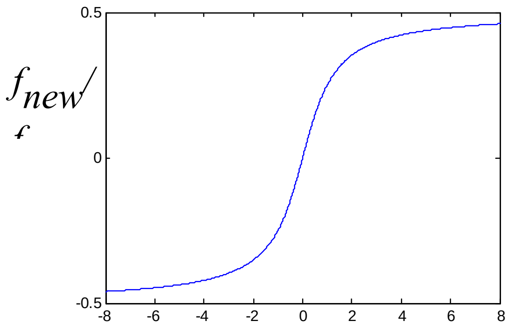
b. ***Bilinear transform method***

$$\boxed{f_{new} = \frac{1}{\pi\Delta_t}\operatorname{atan}\left(\frac{2\pi}{c}f_{old}\right) = \frac{f_s}{\pi}\operatorname{atan}\left(\frac{2\pi}{c}f_{old}\right)}$$

$$\boxed{H\left(f_{new}\right) = H_a\left(f_{old}\right)} \qquad f_s = 1/\Delta_t \text{ (sampling frequency)}$$

$$f_{old} \in (-\infty, \infty) \qquad f_{new} \in (-f_s/2,\ f_s/2)$$

By this method , we transform the original frequency-domain to a new one.(and it’s also a better one)



By the figure above, we could perfectly confirm that the normalized frequency term will never larger than ***0.5*** , so that we could conclude the aliasing effect won’t happen.

(4) Suppose that the IIR filter $h[n]$ is $(1.5\times0.8^n + 0.95^n)u[n]$. What is the efficient way to implement $y[n] = x[n]*h[n]$ (* means convolution)? (10 scores)

(ANS)

(5) Why ① the transition band and ② the weighting function play important roles in digital filter design? (10 scores)

(ANS)

① the transition band :

From ***1979 F.Mintzer and L.Bede Paper***,
The width of the transition band ΔF and the filter length N have the relation equation that:

δ_1 is the passband ripple (error)

$$N = \frac{2}{3}\frac{1}{\Delta F} log_{10}\left(\frac{1}{10\delta_1\delta_2}\right)$$

δ_2 is the stopband ripple (error)

When ***N***(filter length) is fixed ,once we increase the transition band ***ΔF*** , we could increase the accuracy of filter in passband and stopband.

② the weighting function

The Weighting function could control the accuracy between passband and stopband, which means that we could make the accuracy between passband and stopband different.

For example, if we setting that the weight is larger in stopband, smaller in passband, then we could get a smaller ripple in the stopband, which means a higher accuracy, so that we view stopband accuracy is more important , and vice versa.

(6) Make a comparison among the methods of MSE, Minimax, and frequency sampling for FIR filter design and show their advantages and disadvantages. (20 scores)

(ANS)

Method	**Feature**	**Advantage(s)**	**Drawbacks**

MSE	Minimizing the mean square error	1.Flexible designing 2. could also be optimized by weighting function & transition band	Large error near the transition band
Mini-Max	Minimizing the max. error	Min. max. error	1. Transition band is necessary 2. Design duration is slow because of recursion 3. Impulse response h[n] must be odd or even
Frequency Sampling	The discrete-time Fourier transform of the desired filter	Simplest for designing	1. Hard to optimize 2. *Gibb's Phenomenon* in time-domain