Handout 16 Signal Processing for Communications
Solution 8 April 18, 2011, INF1 – 10:15am-12:00

Problem 1 (Gibbs Phenomenon)

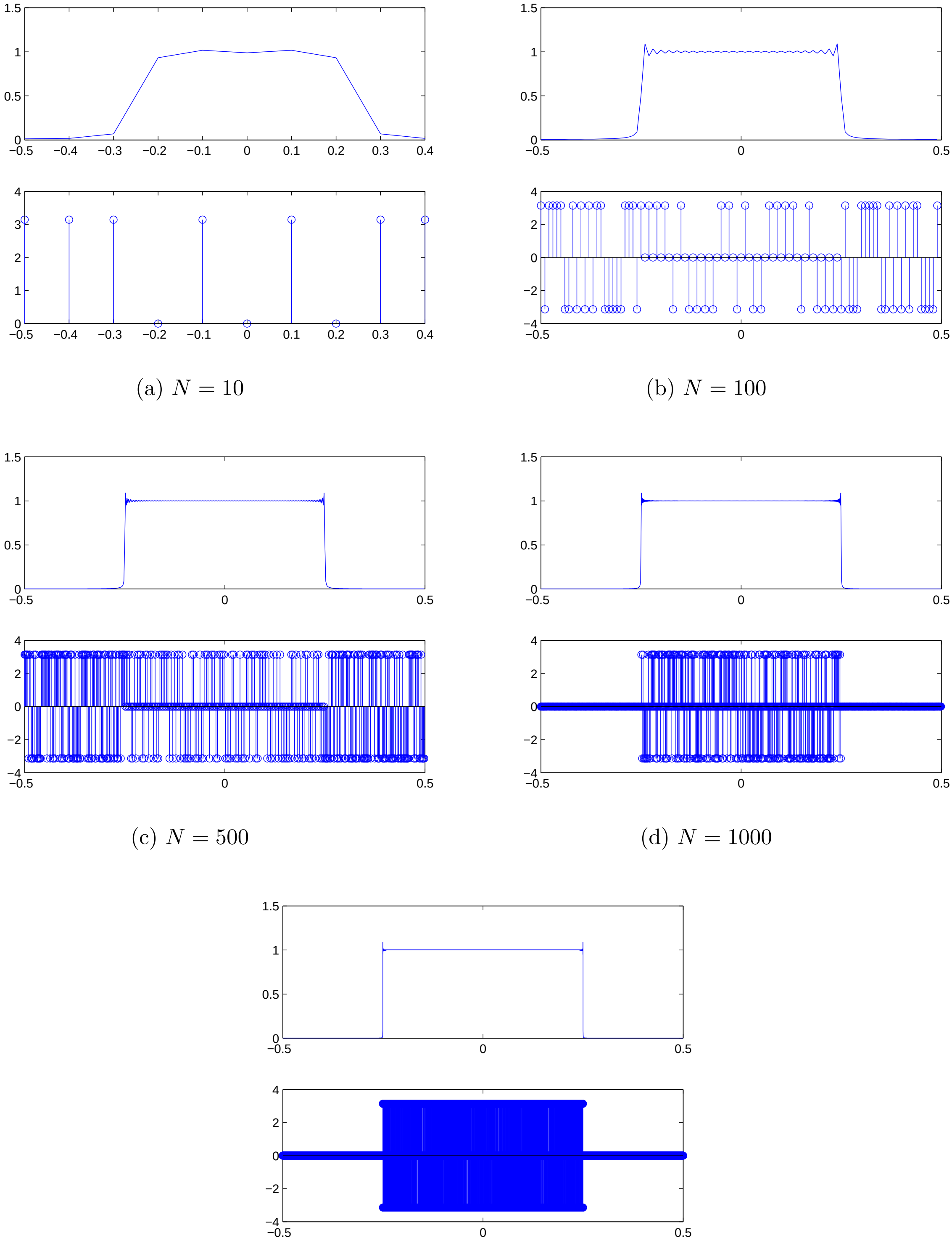
i) We obtained $h[n]$ in problem 2 of HW 6 and it is equal to

$$h[n] = \frac{\sin(2\pi f_c n)}{\pi n}$$

ii) According to problem 2 of HW 6 :

```
function H = LowPass(fc,N)


    fc = 0.25;
    if (N/2 == floor(N/2)) %Check N is even
    h = [sin(2*fc*pi*[-N/2:-1])./([-N/2:-1]*pi),2*fc...
        ,sin(2*fc*pi*[1:N/2-1])./([1:N/2-1]*pi)];
    else %N is odd
    h = [sin(2*fc*pi*[-(N-1)/2:-1])./([-(N-1)/2:-1]*pi),2*fc...
        ,sin(2*fc*pi*[1:(N-1)/2])./([1:(N-1)/2]*pi)];
    end

    H = fft(h,N);
    subplot(2,1,1)
    plot((-N/2:N/2-1)/N,abs(fftshift(H)));
    subplot(2,1,2)
    stem((-N/2:N/2-1)/N,angle(fftshift(H)));
```

iii) The maximum value of frequency responses does not depend on N. It always remains fixed around 1.09 (9% overshoot). See figure 1 for the results.



(a) $N = 10$

(b) $N = 100$

(c) $N = 500$

(d) $N = 1000$

(e) $N = 10000$

Figure 1: Plots of the magnitude (upper sub figures) and phase (lower sub figures) of H for different N.

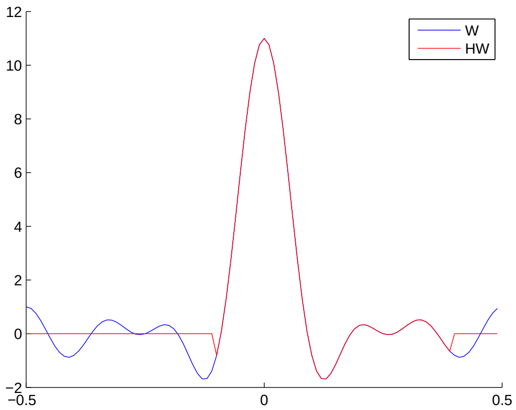
iv) Here is the code to compute it :

```
% Fourier Transform of w[n]:
% Set a maximum for sequence:
close all
Max = 1000;
% The length windowing:
N = 100;
w = zeros(1,Max);
w(Max/2-N/2:1:Max/2+N/2) = ones(1,N+1);
W=fft(w);
% Supress the Phase shift:
% The Phase shift for DFT coefficeint k is
% e^{j*2*pi*(k/Max) * Max/2} = (-1)^k
W = W.* (-1).^(0:Max-1);

subplot(2,1,1)
plot((-Max/2:Max/2-1)/Max,fftshift(W));
title('Fourier transform W');
subplot(2,1,2)
stem(w);
title('Initial signal w[n]');
```

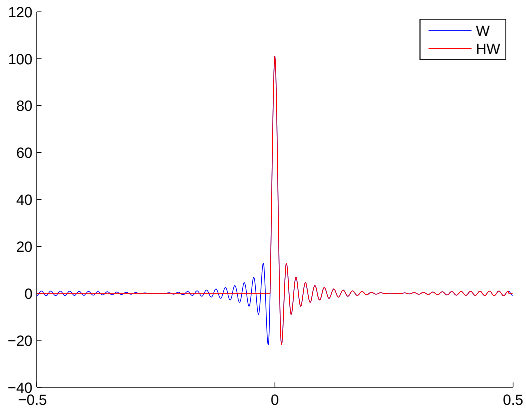
v)

```
%SHOULD RUN AFTER hw84.m
% Fourier Transform of Ideal LowPass filter with cut-off .25 Hz:
H = zeros(1,Max);
% Frequency responses for frequencies -.25 ≤ k/Max ≤.25 are equal to 1:
H(1:Max/4) = ones(1,Max/4);
H(Max-Max/4+1:Max)= ones(1,Max/4);
h = ifft(H);
figure
subplot(2,1,1)
plot((-Max/2:Max/2-1)/Max,abs(fftshift(H)));
subplot(2,1,2)
plot(abs(h));

Hhat = cconv(H,W,Max)/Max;
figure
%Find the index that the frequency response gets maximazed:
ind = find(Hhat(1:Max/2) ≥ max(Hhat(1:Max/2)));
% The frequency of maximazed frequency response.
frequency = ind/Max

% Do the Convolution to find out why it is maximum:
circH = circshift(H,[1,ind]);
D = circH.*W;
hold on

plot((-Max/2:Max/2-1)/Max,fftshift(W));
plot((-Max/2:Max/2-1)/Max,fftshift(D),'r');
```

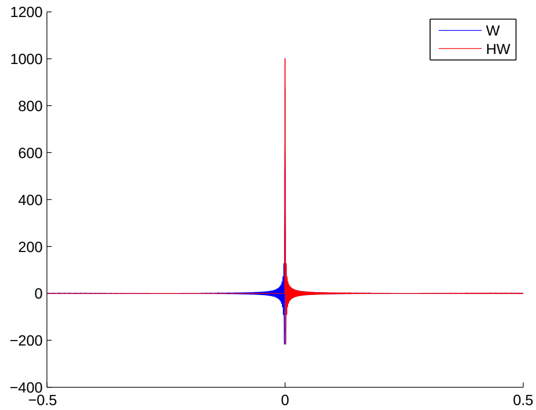
f_N increases by growing N and for large values of N, it gets close to 0.25 (from left). See figure 2 for the results.



(a) $N = 10$



(b) $N = 100$



(c) $N = 1000$

Figure 2: Plots of $W(e^{j2\pi\theta})$ and $H(e^{j2\pi(f-\theta)}) \cdot W(e^{j2\pi\theta})$ for different N

vi) We know that, $W(e^{j2\pi\theta}) = \frac{\sin((N+\frac{1}{2})2\pi\theta)}{\sin(\pi\theta)}$ for $\frac{-1}{2} \leq \theta \leq \frac{1}{2}$.
The maximum value of frequency response takes place when the window of $H(e^{j2\pi(f-\theta)})$ doesn't cover one of the main negative lope and the remaining lopes before that. Indeed, the integral value behind this part of $W(e^{j2\pi\theta})$ is about -0.089. Therefore, since

$$\int_{-1/2}^{1/2} W(e^{j2\pi\theta})d\theta = 1,$$

we can conclude that

$$\int_{-1/2}^{1/2} W(e^{j2\pi\theta})H(e^{j2\pi(f-\theta)})d\theta \approx 1.089$$

By changing N, the integral under the rescaled frequency response does not change. Therefore, the maximum value remains about 1.089.

Problem 2 (Weighted Least-Squares Filter Design)

i) Due to

$$E(f) = W(e^{j2\pi f})\left[D(e^{j2\pi f}) - H_d(e^{j2\pi f})\right]$$

we can easily compute the matrices **W** and **U** and the vector **d**.
Consider $f = f_i = \frac{i}{K}$

$$e_i = E(f_i) = W(e^{j2\pi\frac{i}{K}})\left[D(e^{j2\pi\frac{i}{K}}) - H_d(e^{j2\pi\frac{i}{K}})\right].$$

Therefore, for the vector $\mathbf{d} = [d_0, \ldots, d_K]$:

$$d_i = \begin{cases} 0 & i < f_p K \\ 1 & i \geq f_p K \end{cases}$$

In fact the interval between $f_s K$ and $f_p K$ is not important since the weight considered for this interval is zero. Therefore, we can set it anything, so we let that 0.
And then the i-th row of matrix U is

$$U_i = [1, 2\cos(2\pi\frac{i}{K}), 2\cos(2\pi\cdot 2\cdot\frac{i}{K}), \ldots, 2\cos(2\pi\frac{M-1}{2}\frac{i}{K})]$$

The matrix W is a diagonal matrix such that :

$$W_{ii} = W(e^{j2\pi\frac{i}{K}}) = \begin{cases} \frac{\delta_p}{\delta_s} & i \leq f_s K \\ 1 & i \geq f_p K \\ 0 & \text{otherwise} \end{cases}$$

ii) We know that

$$e = W(d - Uh)$$

for every e_i, we have two conditions :

$$e_i \leq \delta_p \qquad \text{and} \qquad -e_i \leq \delta_p$$

Therefore we make the following vector

$$\begin{bmatrix} \mathbf{e} \\ \cdots \\ -\mathbf{e} \end{bmatrix} = \begin{bmatrix} e_1 \\ \vdots \\ e_K \\ \cdots \\ -e_1 \\ \vdots \\ e_K \end{bmatrix} \leq \begin{bmatrix} \delta_p \\ \delta_p \\ \vdots \\ \delta_p \end{bmatrix} = \delta_{\mathbf{p}}$$

Hence,

$$\begin{bmatrix} \mathbf{e} \\ \cdots \\ -\mathbf{e} \end{bmatrix} = \underbrace{\begin{bmatrix} W_{K\times K} & 0_{K\times K} \\ 0_{K\times K} & W_{K\times K} \end{bmatrix}}_{W'} \left[\underbrace{\begin{bmatrix} d \\ \cdots \\ d \end{bmatrix}}_{d'} \underbrace{\begin{bmatrix} U_{K\times K} \\ \cdots \\ U_{K\times K} \end{bmatrix}}_{U'} h \right] \leq \delta_p$$

$$W'(d' - U'h) \leq \delta_p \Rightarrow -W'U'h \leq \delta_p - W'd'$$

Thus, $A = -W'U'$ and $b = \delta_p - W'd'$

In part (iii) we should insert the above matrices in the code to obtain the desired WLS filter.