Signal and Image Processing

Lab session 4

Elliott Perryman, Jan Zavadil

12.11.2021

1 Signal windowing

A signal of finite length N corresponds to the multiplication of the signal with a rectangular window of length N, denoted by w_r. In the Fourier domain, this corresponds to the convolution with the Fourier transform of the rectangular window. Alternatively, we consider the Hamming window of length N.

$$w_h(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N}\right) & \text{if} \quad n \in \{0, \dots, N-1\} \\ 0 \quad \text{otherwise} \end{cases}$$

As the Dirac distribution is the neutral operator for the convolution, smaller distortions are obtained when the Fourier transform of the window is closer to the Dirac distribution.

1.1 Assuming the number of frequency bins is $L = 1024$ and considering $N = 10, N = 100, N = 200$, write a program that plots the modulus of the Fourier transform of the windows w_r and w_h at frequency locations $\frac{k}{L}$ for $-\frac{L}{2} \leq k \leq \frac{L}{2} - 1$.

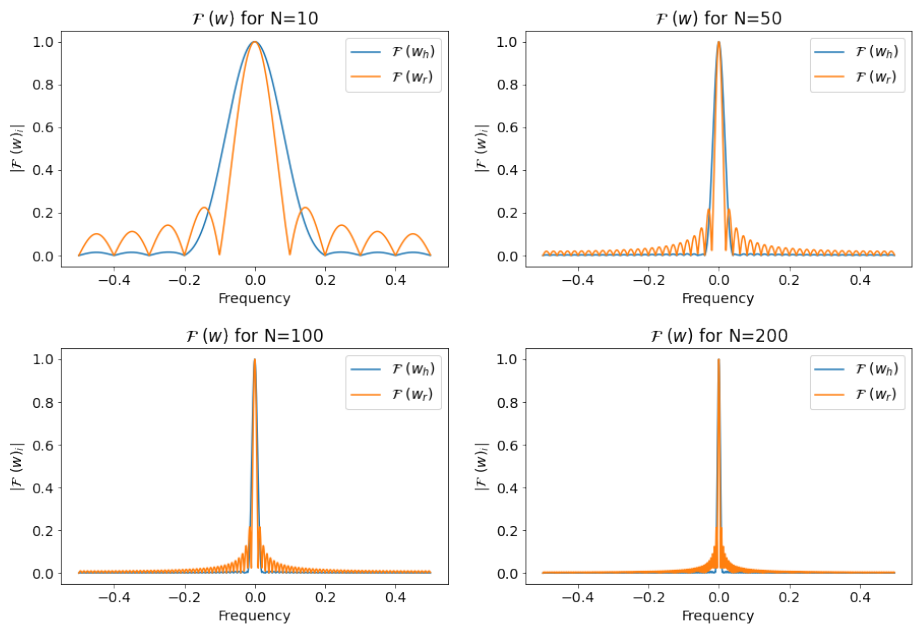

Figure 1: Comparison of the modulus of Fourier transform of w_r and w_h for $N = 10, 50, 100, 200$. N=50 is included because a 2x2 grid of figures looks better.

1.2 What is the effect of N on the frequency representation?

Both transforms converge to the Dirac distribution. In all figures, $\hat{w}_r$ goes to zero faster than $\hat{w}_h$. $\hat{w}_h$ does not have the large oscillations that $\hat{w}_r$ does. As N increases, the oscillations of $\hat{w}_r$ do not go away but instead move closer to 0. Their amplitude remains large even for large N. By analogy, this is similar to an under-damped harmonic oscillator versus a critically damped oscillator. The critically damped oscillator behaves like $\hat{w}_h$ as it goes away from frequency 0 in that it goes to the baseline much slower than the underdamped system, but its oscillations are much lower. The underdamped oscillator goes to zero quickly but overshoots and takes a long time to settle to 0.

2 Low Pass signal filtering

In this exercise, we would like to build a finite impulse response filter h whose Fourier transform approximates $\mathbf{1}_{[-f_0,f_0]}$, the indicator function of interval $[-f_0, f_0]$ for $f_0 < 1/2$.

2.1 Assuming $\mathbf{1}_{[-f_0,f_0]}$ is periodized into a 1-periodic function, compute the Fourier coefficients of the function, denoted by h_n.

$$h(\lambda) = \sum_{n\in\mathcal{Z}} h_n e^{2i\pi n\lambda} \approx \sum_{m=\frac{-N-1}{2}}^{\frac{N-1}{2}} h_n e^{2i\pi m\lambda}$$

$$h_n = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{1}_{[-f_0,f_0]}(\lambda) e^{-2i\pi n\lambda} \mathrm{d}\lambda = \int_{-f_0}^{f_0} e^{-2i\pi n\lambda} \mathrm{d}\lambda = -\frac{1}{2i\pi n}\left(e^{-2i\pi n f_0} - e^{2i\pi n f_0}\right) =$$

$$= \frac{\sin(2\pi n f_0)}{\pi n} = 2f_0 sinc(2\pi n f_0)$$

2.2 As in the previous lab session, we truncate the h_n and keep only N coefficients, and we adopt the same rule as in the previous lab session to obtain $\tilde{h}_n$ when n is even.

$$h(\lambda) \approx \sum_{-N/2+1}^{N/2} h_n e^{2i\pi n\lambda} \approx \sum_{-N/2+1}^{N/2} \tilde{h}_n e^{2i\pi(n-1/2)\lambda}$$

$$\hat{h}_n = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{1}_{[-f_0,f_0]}(\lambda) e^{-i\pi(2n-1)\lambda} \mathrm{d}\lambda = \int_{-f_0}^{f_0} e^{-i\pi(2n-1)\lambda} \mathrm{d}\lambda =$$

$$-\frac{1}{i\pi(2n-1)}\left(e^{-i\pi(2n-1)f_0} - e^{i\pi(2n-1)f_0}\right) = \frac{2}{\pi(2n-1)} \sin\left(\pi(2n-1)f_0\right)$$

2.3 Multiply h with the Hamming window of length N to obtain g, this should be done by writing a function $g = FIR(f_0, N)$.

$$g(n) = h(n) w_h(n + N/2)$$

$$\hat{h}_n w_h(n) = \frac{2}{\pi(2n-1)} \sin\left(\pi(2n-1)f_0\right)\left(0.54 - 0.46\cos\left(\frac{2\pi(n+\frac{N}{2})}{N}\right)\right)$$

$$\text{for} \quad -\frac{N}{2} \leq n \leq \frac{N}{2} - 1 \quad \text{when } N \text{ is even}$$

$$h_n w_h(n) = \frac{\sin\left(2\pi n f_0\right)}{\pi n}\left(0.54 - 0.46\cos\left(\frac{2\pi(n+\frac{N-1}{2})}{N}\right)\right)$$

$$\text{for} \quad -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \quad \text{when } N \text{ is odd}$$

2.4 Compute the modulus of the DFT of h and g and conclude which filter is better, namely h or g.

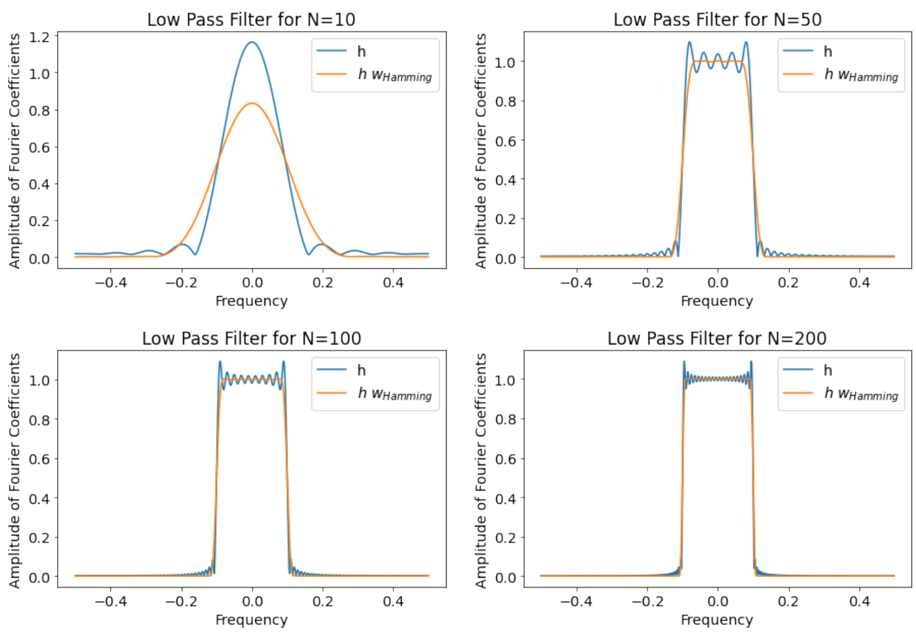

Figure 2: Comparison of DFT of an approximation of a Low pass filter using h, g for $N = 10, 50, 100, 200$

3 Band Pass signal filtering

In this exercise, we would like to build a finite impulse response filter h whose Fourier transform approximates $\mathbf{1}_{[-f_0-f_1, f_0-f_1]} + \mathbf{1}_{[-f_0+f_1, f_0+f_1]}$ for some $f_0 < 1/2$ and $f_1 > f_0$ and $f_1 + f_2 < 1/2$.

3.1 If we denote H by the Fourier transform of a sequence h, show that the Fourier transform of the sequence $2h_n cos(2\pi f_1 n)$ is $H(\xi - f_1) + H(\xi + f_1)$.

$$\sum_{n\in\mathbb{Z}} h_n e^{-2i\pi\xi n} = H(\xi)$$

$$\sum_{n\in\mathbb{Z}} 2h_n cos(2\pi f_1 n)e^{-2i\pi\xi n} = \sum_{n\in\mathbb{Z}} h_n \left(e^{2i\pi f_1 n} + e^{-2i\pi f_1 n}\right) e^{-2i\pi\xi n} =$$

$$\sum_{n\in\mathbb{Z}} h_n \left(e^{-2i\pi(\xi - f_1)n} + e^{-2i\pi(\xi+f_1)n}\right) = H(\xi - f_1) + H(\xi + f_1)$$

3.2 Modify the program of the previous question to obtain filters of size N with Fourier transform approximating $\mathbf{1}_{[-f_0-f_1, f_0-f_1]} + \mathbf{1}_{[-f_0+f_1, f_0+f_1]}$. Deal with the cases N odd and even, and plot the modulus of the corresponding discrete Fourier transform. Numerical application: $f_0 = 1/8, f_1 = 1/4$. What is the role of such a filter?

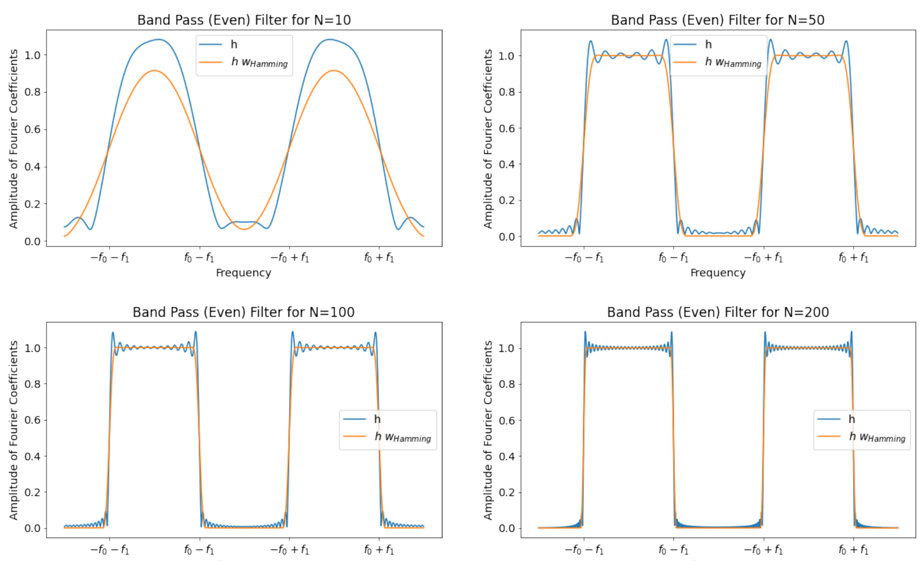

Figure 3: Fourier transform of a approximations of a band pass filter using h and $h_{hamming}$ for even values $N = 10, 50, 100, 200$

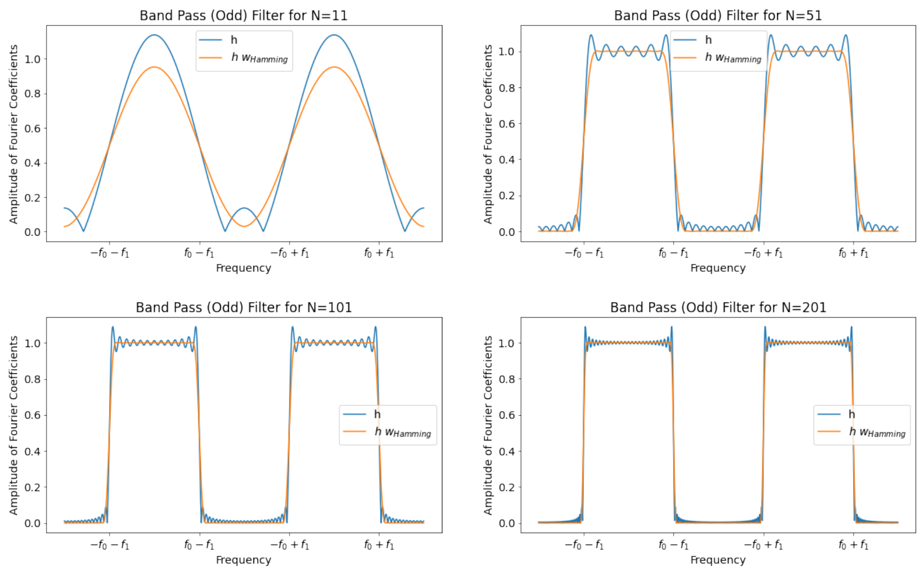

Figure 4: Fourier transform of a approximations of a band pass filter using h and $h_{hamming}$ for odd values $N = 11, 51, 101, 201$

This filter cancels out too high and too low frequencies, ignoring high and low frequency noise. One example of a use case would be in cleaning a signal that has some long slow oscillation in its baseline, possibly from some physical shaking, and a high frequency noise component. This would depend on the frequency representation of the signal not having large amplitudes in the regions that would be set to zero.