

Seung Jun Baek

Topics: FIR filter design

1. The frequency response of a desired filter $h_d[n]$ is shown in Fig. 1. In this problem, we wish to design an $(M + 1)$ -point causal linear-phase FIR filter $h[n]$ that minimizes the integral-squared error

$$\epsilon^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |A(e^{j\omega}) - H_d(e^{j\omega})|^2 d\omega,$$

where the frequency response of the filter $h[n]$ is

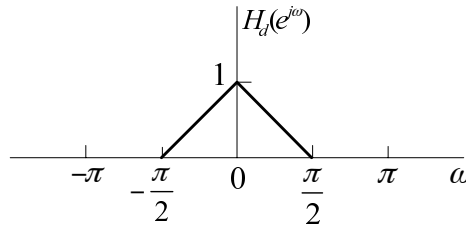


Fig 1:

$$H(e^{j\omega}) = A(e^{j\omega})e^{-j\omega M/2}$$

and M is an even integer

- Determine $h_d[n]$.
 - What symmetry should $h[n]$ have in the range $0 \leq n \leq M$? Briefly explain your reasoning.
 - Determine $h[n]$ in the range $0 \leq n \leq M$.
 - Determine an expression for the minimum integral-squared error ϵ^2 as a function of $h_d[n]$ and M .
2. Consider the function $D(x) = e^x$ and

$$H(x) = \sum_{k=0}^L b_k x^k$$

where $L = 1$, so $H(x)$ is a first-order polynomial function. In the interval $0 \leq x \leq 1$, find the coefficients b_0, b_1 that minimizes the maximum error between $H(x)$ and $D(x)$,

$$\max_{0 \leq x \leq 1} |H(x) - D(x)|.$$

(Hints: The number of points where the maximum error occurs is $L + 2 = 3$ where two of the m occur at $x = 0$ and $x = 1$. Also note that you need three points to solve for two coefficients b_0 and b_1 .)

3. An optimal equiripple FIR linear-phase filter was designed by the Parks-McClellan algorithm. The magnitude of its frequency response is shown in Fig. 2. The maximum approximation error in the passband and stopband is $\delta_1 = \delta_2 = 0.085$. The passband and stopband cutoff frequencies are $\omega_p = 0.4\pi$ and $\omega_s = 0.58\pi$.

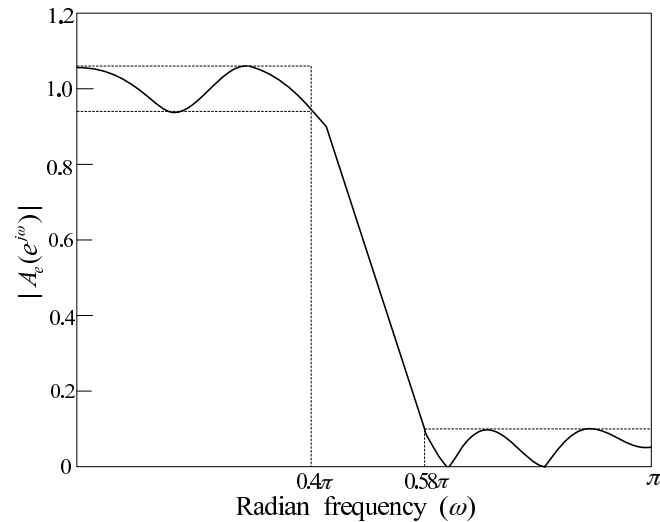


Figure 2:

- (a) Carefully sketch the approximation error; i.e., sketch

$$E(\omega) = H_d(e^{j\omega}) - A_e(e^{j\omega}).$$

(Note that Fig. 2 shows $|A_e(e^{j\omega})|$.)

- (b) What is the length of the impulse response of the system?

4. **(Migrating to Python.)** From now on, we will use Python for computation and plotting for signal processing. Python is a free tool where Matlab is proprietary software – that is you have to pay for the tool. Typically the development environment like IDE, Matlab is convenient, but other open-source benefits can be obtained from Python, especially if you have to organize a big project because Python is a well structured, general purpose programming language (compared with Matlab built for matrix operation).

An example website for Matlab to Python is

<https://towardsdatascience.com/python-for-matlab-users-ac3e0b8463a5> (you can search web for many other examples), and also for more detailed Matlab to Python function/notation mapping is given

<https://docs.scipy.org/doc/numpy-1.15.0/user/numpy-for-matlab-users.html>

Use your favorite IDE for python – I personally use Pycharm at (www.jetbrains.com/pycharm).

- (a) **Reproducing previous HW with Python:** Convert the MATLAB problem in HW2 to an equivalent code in python. You may want to import libraries such as numpy, matplotlib and signal from scipy. For example, if you want to do a convolution between two sequences, you may use `signal.convolve`; if you want to plot figures using matlab-like commands (`subplot`, `xlabel`, `ylabel`, ...), you may want to use `matplotlib.pyplot`. For example, your python code may start with

```

from scipy import signal
import numpy as np
import matplotlib.pyplot as plt

```

- (b) **Gibb's phenomenon:** Consider the following impulse response of truncated low-pass filter

$$h_M[n] = \begin{cases} \frac{\sin(\frac{\pi n}{5})}{\pi n} & -M \leq n \leq M \\ 0 & \text{otherwise} \end{cases}$$

Consider a Gaussian-like input signal given by

$$x[n] = \begin{cases} \exp\left(-\frac{n^2}{2}\right), & -N \leq n \leq N \\ 0 & \text{otherwise} \end{cases}$$

Let $N = 100$. For the variable in frequency domain, consider ω vector $-2\pi \leq \omega \leq 2\pi$ with step size $\Delta\omega = 0.001\pi$. Also let us denote the output signal by $y_M[n]$.

- i. Assume $M = 20$. Let DTFTs of $x[n]$, $h_{20}[n]$ and $y_{20}[n]$ be $X(e^{j\omega})$, $H_{20}(e^{j\omega})$ and $Y_{20}(e^{j\omega})$. Plot the magnitude of DTFT $X(e^{j\omega})$, $H_{20}(e^{j\omega})$ and $Y_{20}(e^{j\omega})$ in one page using subplot.
- ii. Repeat the same with $M = 70$: plot the magnitudes of $X(e^{j\omega})$, $H_{70}(e^{j\omega})$ and $Y_{70}(e^{j\omega})$. Do you observe the Gibb's phenomenon?