BME 50500: Image and Signal Processing in Biomedicine

Lecture 4: Filtering

Lucas C. Parra
Biomedical Engineering Department
City College of New York

CCNY

http://bme.ccny.cuny.edu/faculty/parra/teaching/signal-and-image/
parra@ccny.cuny.edu

Content (Lecture Schedule)

Linear systems in discrete time/space
Impulse response, shift invariance (4)
Convolution (4)
Discrete Fourier Transform (3)
Power spectrum (7)

Medial imaging modalities
MRI (2)
Tomography, CT, PET (5)
Ultrasound (8)

Engineering tradeoffs
Sampling, aliasing (1)
Time and frequency resolution (3)
Wavelength and spatial resolution (9)
Aperture and resolution (9)

Filtering
➡ Magnitude and phase response (6)
➡ Filtering (6)
Correlation (7)
Template Matching (10)

Intensity manipulations
A/D conversion, linearity (1)
Thresholding (10)
Gamma correction (11)
Histogram equalization (11)

Matlab

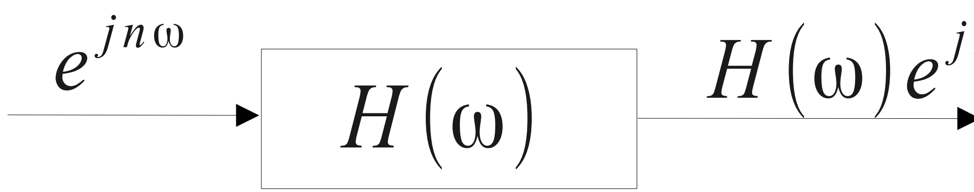
DTFT - System frequency response

Consider stationary oscillatory input to a LSI system $h[k]$:

$$x[n]=e^{jn\omega}$$

$$y[n]=\sum_{k=-\infty}^{\infty} h[k]x[n-k]=\sum_{k=-\infty}^{\infty} h[k]e^{j(n-k)\omega}=H(\omega)e^{jn\omega}$$

The output is the input times the DTFT of the impulse response



The oscillation with frequency ω has been modified in phase by the "phase delay" $\Delta\Phi$ and in amplitude by "gain" G

$$G(\omega)=|H(\omega)| \qquad \Delta\Phi(\omega)=angle(H(\omega))$$

$$H(\omega)=Ge^{j\Delta\Phi}$$

Gain and phase delay

Gain of a filter determines the output/input power ratio

$$G = \sqrt{\frac{P_y}{P_x}} \qquad P_x = \frac{1}{N}\sum_{n=1}^{N} \left| x[n] \right|^2$$

Gain is often specified in decibel:

$$dB(G) = 20 \log_{10} G = 10 \log_{10} \frac{P_y}{P_x}$$

Phase delay $\Delta\Phi$ corresponds to a shift in time, Δt, called "group delay" which depends on the frequency f

$$\Delta\Phi = 2\pi f \Delta t = \omega \Delta t$$

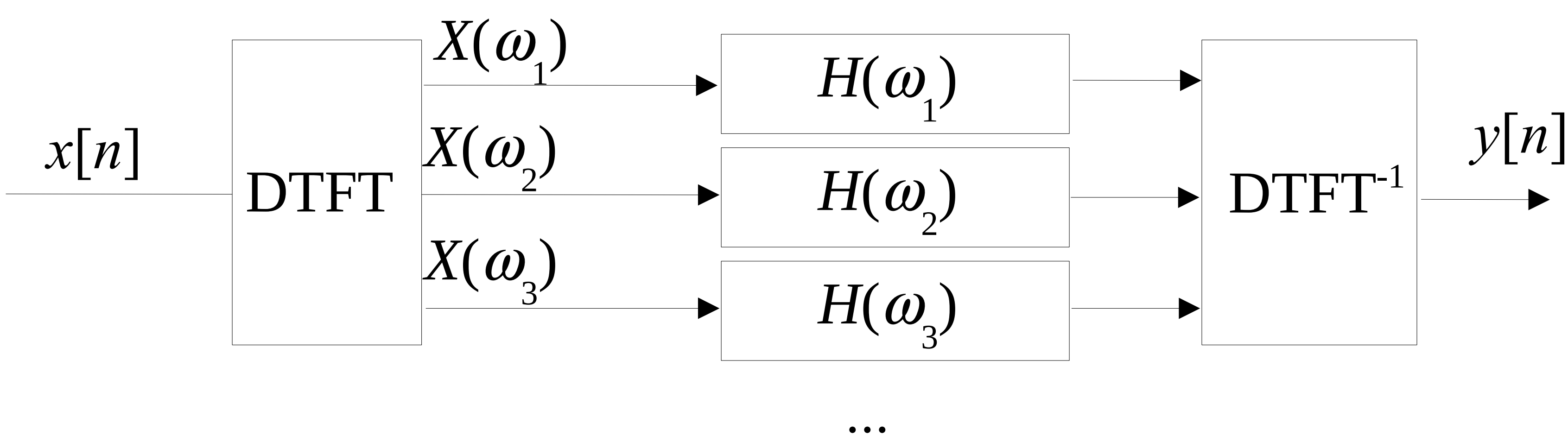
DTFT - System frequency response

DTFT inversion formula tells us that arbitrary input x[n] can be decomposed into sum of oscillations

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} d\omega \, X(\omega) e^{jn\omega}$$

The system response to that is given by the convolution theorem

$$Y(\omega) = H(\omega) X(\omega)$$



Moving Average Filter - FIR

A FIR filter is sometimes also called a **Moving Average (MA)** filter :

$$y[n]=\sum_{k=0}^{Q} b[k]x[n-k]$$

All symmetric MA filters will have a linear phase.

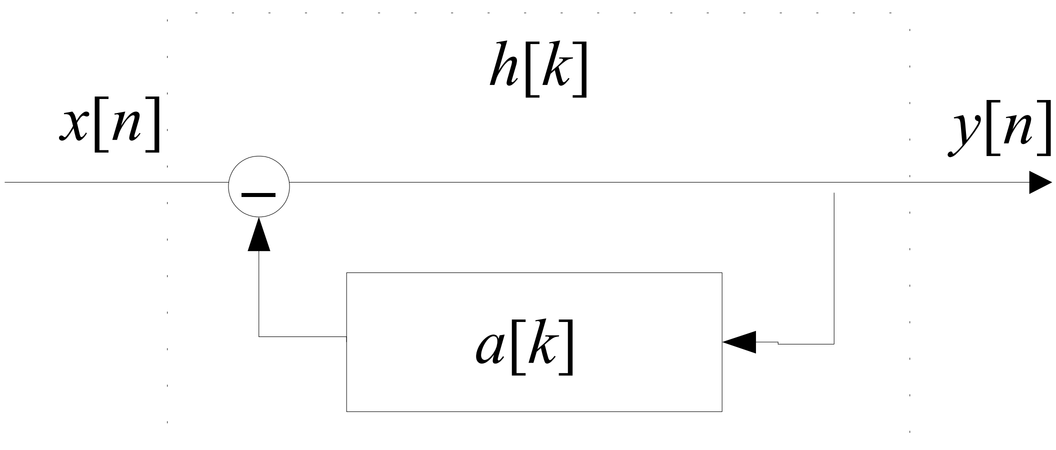
However, to generate a very narrow frequency response one may need very long filters (remember the uncertainty principle!)

To generate long filters therefore we may need a infinite impulse response.

Auto Regressive Filter - IIR

An **Infinite Impulse Response (IIR)** can be easily implemented with an **Auto Regressive** (**AR**) filter:

$$y[n]=x[n]-\sum_{k=1}^{P} a[k]\,y[n-k]$$



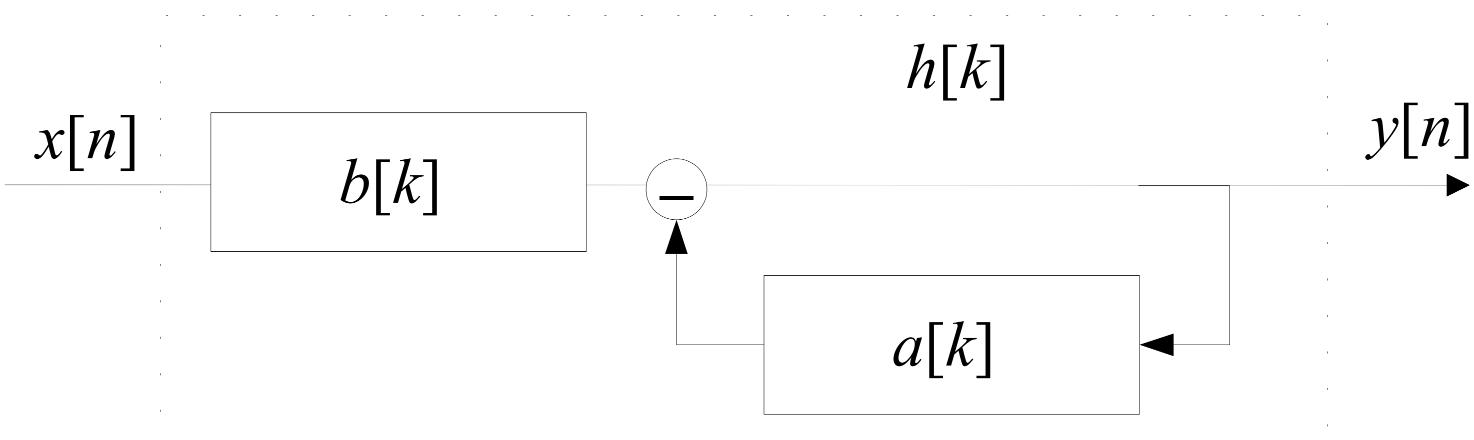
However, $h[t]$ **may not be stable!** Filter $h[k]$ is stable if:

$$\sum_{k=-\infty}^{\infty} |h[k]| < \infty$$

ARMA filter

More generally an **Infinite Impulse Response (IIR)** can be represented by an **ARMA** filter (also called difference model):

$$y[n] = -\sum_{k=1}^{P} a[k]\, y[n-k] + \sum_{k=0}^{Q} b[k]\, x[n-k]$$



Since ARMA filter is LSI there is a corresponding $h[k]$ that characterizes the system impulse response.

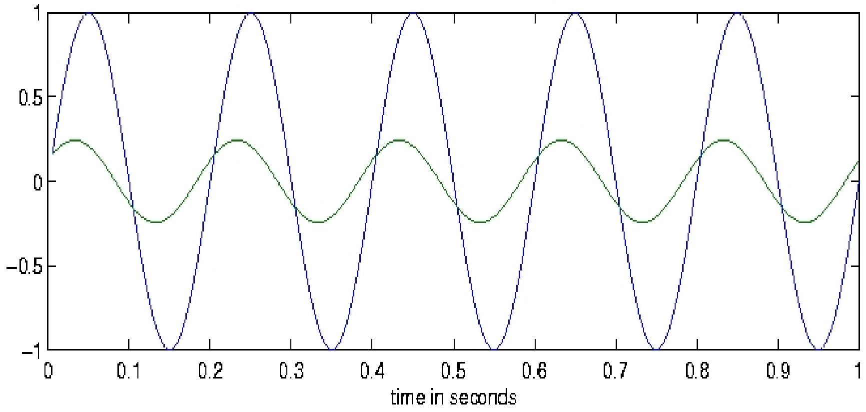
```
>> y = filter(b,a,x);
```

DTFT - System frequency response

Example: $h[0] = 1$, $h[1]=-0.8$

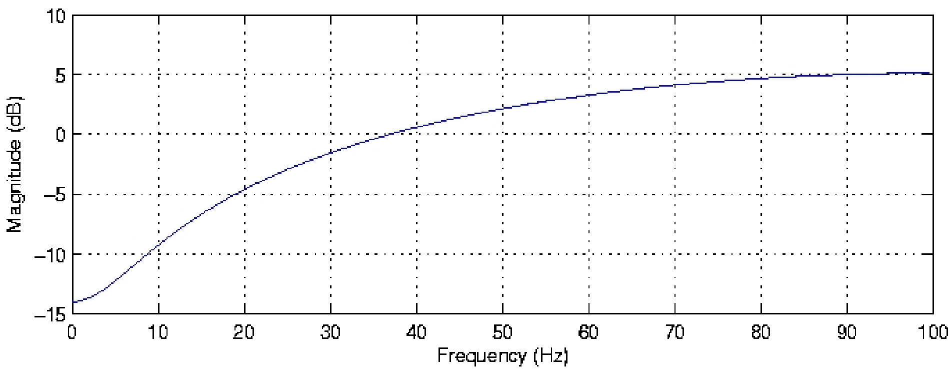
Time domain response

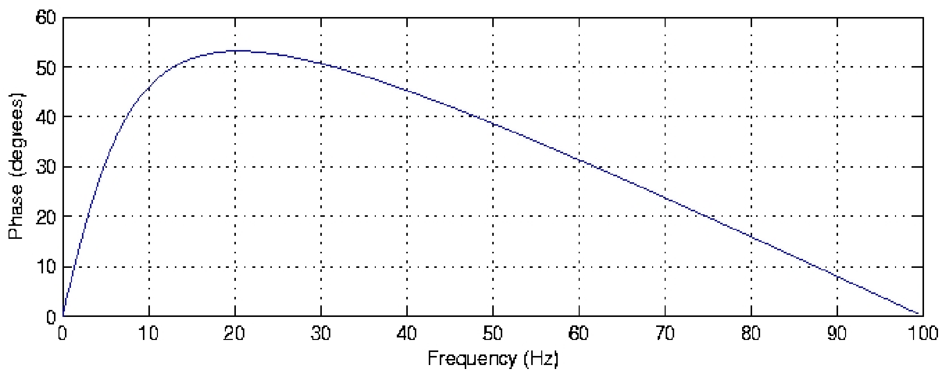
```
>> x = sin(2*pi*5*t);

>> plot(t,filter(h,1,x))
```



Frequency domain response

```
>> plot(f,db(abs(H)))
```






```
>> plot(f,angle(H))
```

DTFT - System frequency response

Assignment 6:

1. Pick some arbitrary FIR filter and show the magnitude and phase response from 0Hz to Nyquist frequency. Make sure the axis are labeled correctly.
2. Generate a 1 second steady state sinusoid with some frequency of your choice and filter this input $x[n]$ with your filter to generate output $y[n]$. Show the input $x[n]$ and output $y[n]$ in a single graph.
3. Compute from these two signals the magnitude and phase response of the filter at that frequency. Plot your measurement as a point on the phase and magnitude plots from task 2.
4. Repeat the process for various frequencies.
5. Add -10dB noise to the output $y[n]$ and repeat your estimation of phase and magnitude response.

DTFT - System frequency response

Assignment 6 - Alternate:

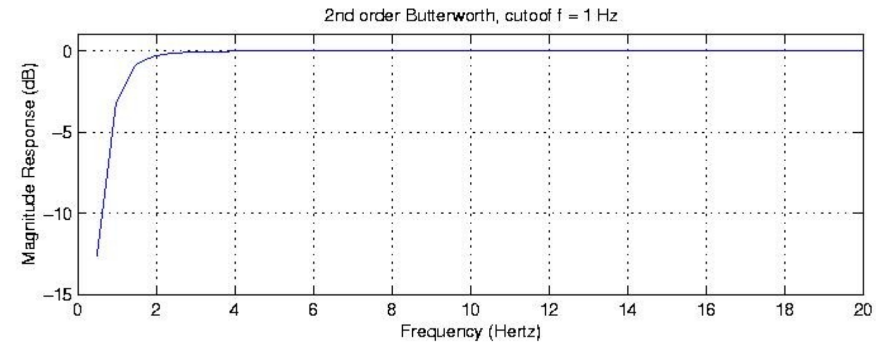
1. Use the impulse response you have measured from your system (Assignment 3) and show the magnitude and phase response from 0Hz to Nyquist frequency. Make sure the axis are labeled correctly.
2. Generate a steady state sinusoid with the function generator at some frequency of your choice and apply this voltage to the "input" of your system. Sample the output and show input and output in a single graph. Repeat this for 4 different frequencies.
3. For each frequency compute from these input and output signals the magnitude and phase response of your system at that frequency. Plot your measurement as a point on the phase and magnitude plots from task 1.

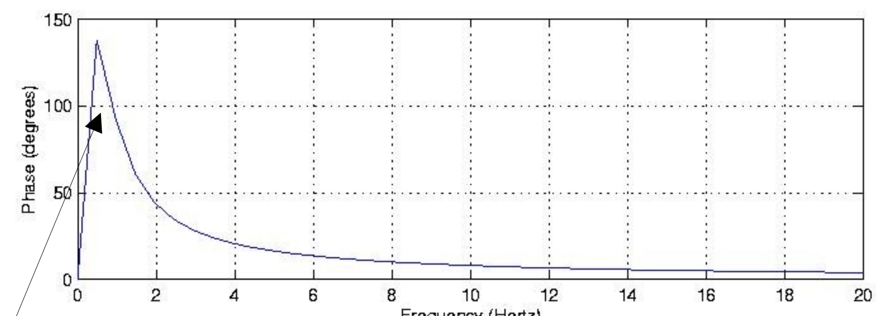
DTFT - Filter design example

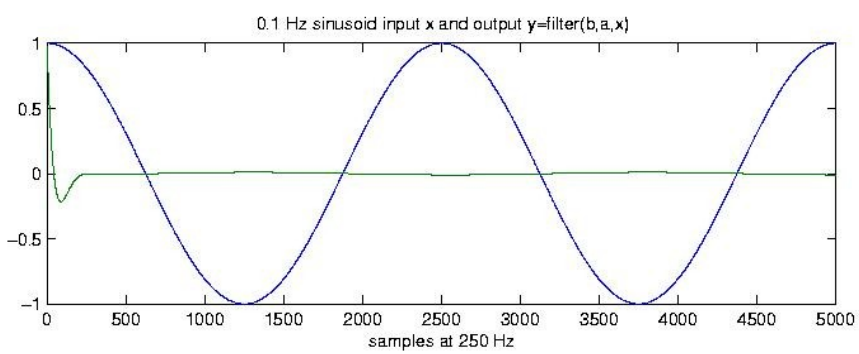
Goal: Highpass filter the DC drift in EEG with minimal latency and memory.

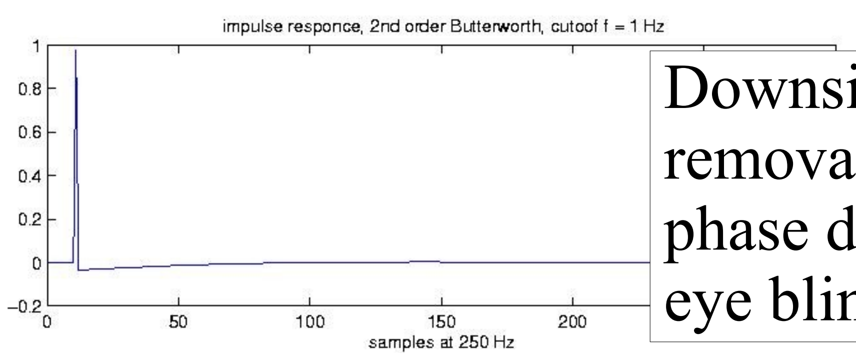
```
>> sptool
```

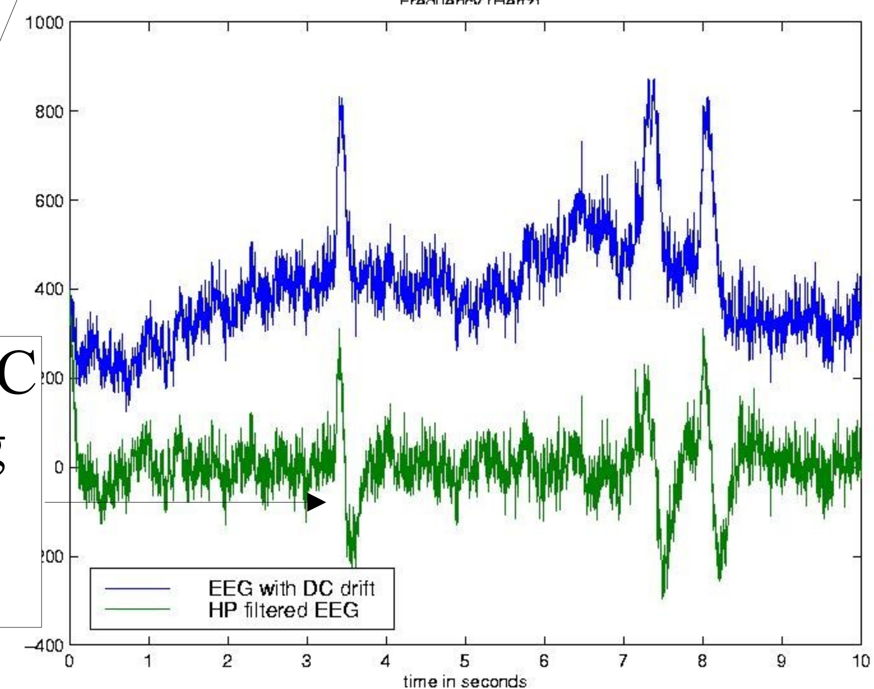
Solution: Select IIR filter, 2nd order Butterworth with cutoff at 1Hz.











Downside to DC removal is long phase delay for eye blinks.

DTFT - Zero phase and linear phase

A filter is said to have *zero phase* if it introduces no phase delay

$$H(\omega) = |H(\omega)|$$

True for all symmetric FIR filters: $h[-n] = h^*[n]$

A filter is said to have *linear phase* if

$$H(\omega) = |H(\omega)| e^{j\omega n_0}$$

Linear phase corresponds to shift in time. Because delay in time corresponds to a multiplication with a linear phase term:

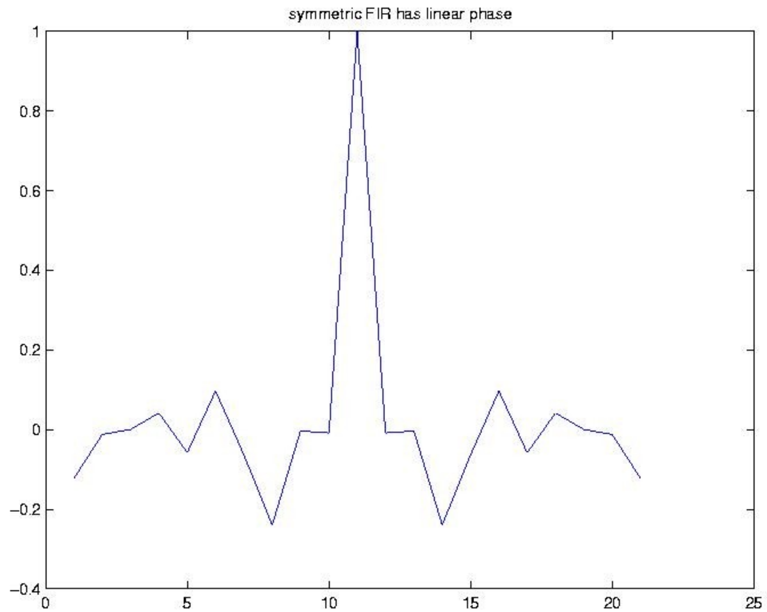
$$x_{n_0} = x[n+n_0] \quad \Rightarrow \quad X_{n_0}(\omega) = e^{j\omega n_0} X(\omega)$$

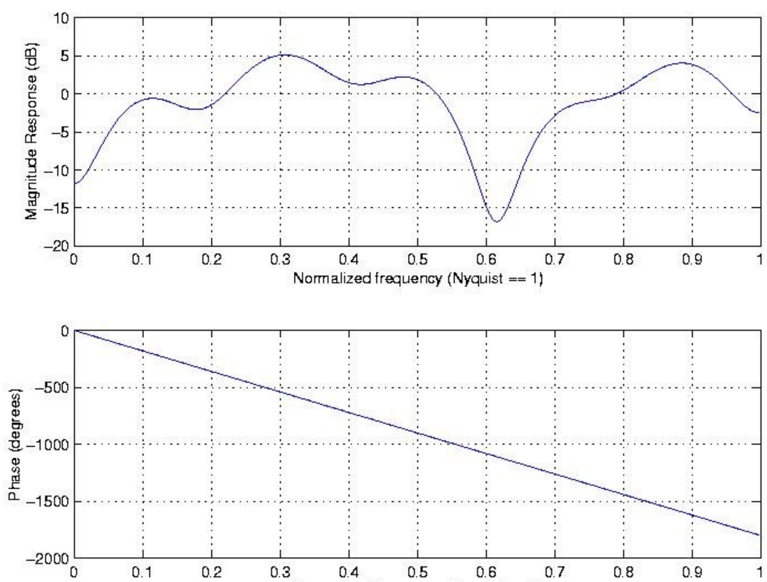
$$Y(\omega) = |H(\omega)| e^{j\omega n_0} X(\omega) = |H(\omega)| X_{n_0}(\omega)$$

DTFT - Linear Phase

- A filter with linear phase delays all frequencies by the same amount.
- If we add a constant delay to a zero phase filter we obtain a linear phase filter.
- The shift in time can be removed if the filter can be non-causal. In which case we get a zero phase filter.

Example: delay here 11 samples



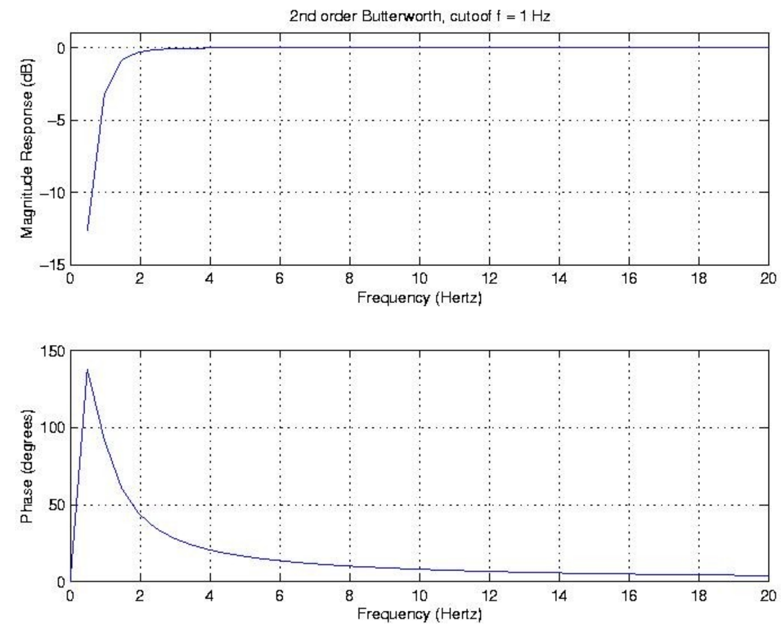


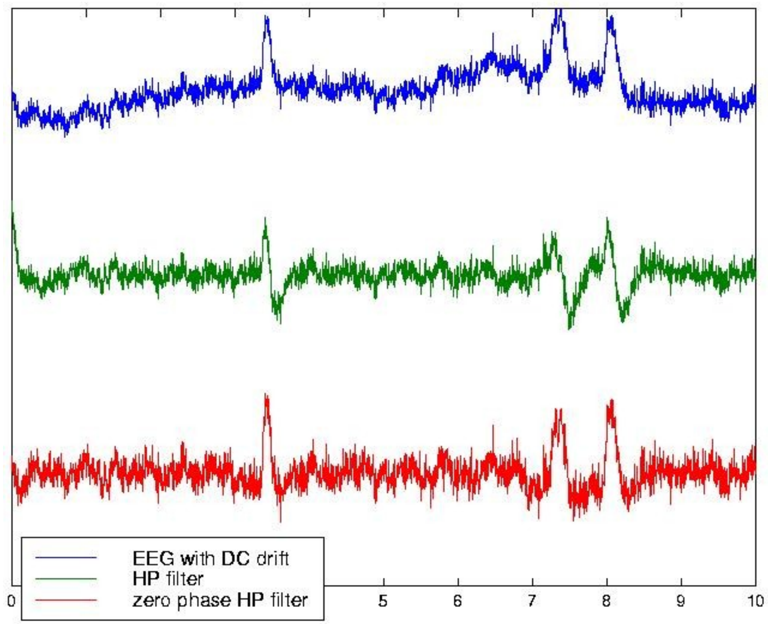
DTFT - Pragmatic filter design example

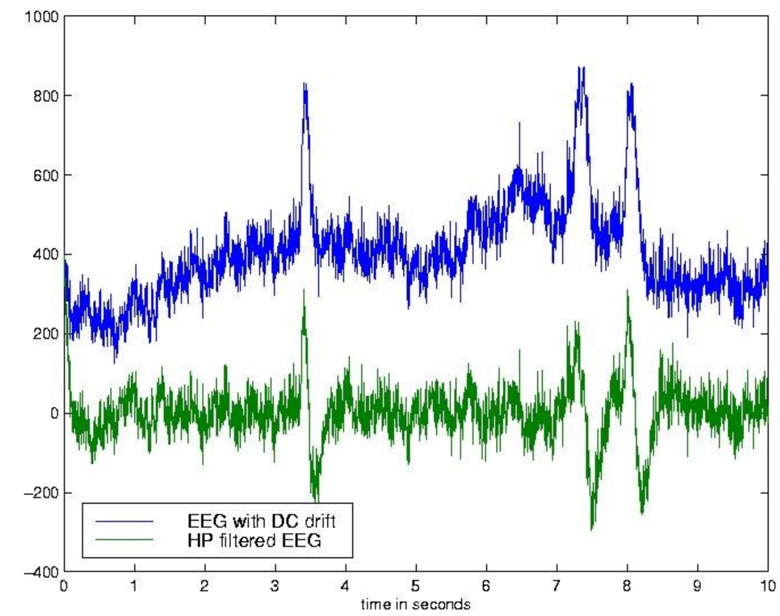
If the filter is allowed to be non causal a **quick fix** is to apply the filter to the time inverted signal resulting in a **zero phase** filter:

$$Y(\omega) = \left| H(\omega) \right|^2 X(\omega)$$

```
>>  filtfilt(b,a,x)
```







DTFT - Pragmatic filter design

Assignment 7:

1. Use `sptool()` to design 60Hz bandstop filter using both an IIR filter of low order and an FIR filter with linear phase.

   Hints: After finding a good filter take note of the parameters, and use the corresponding filter design function to compute within your code the ARMA filter coefficients. To find the corresponding filter function use for instance:

   ```
   >> lookfor chebyshev
   ```

   Alternatively, you may export the filter coefficients from sptool and save the numerical values into a matlab file that your code will load. For a FIR filter it will save coefficients *Num* which is simply the moving average part of the filter:

   ```
   b = Num; a = 1;
   ```

   For a IIR filter it will save Coefficients *SOS* and *G*. You can get coefficient *a*,*b* with this code:

   ```
   [b,a] = sos2tf(SOS,G)
   ```

2. For both IIR and FIR filters, show the corresponding impulse response, magnitude and phase response (compute these from filter coefficient *b*,*a*).
3. Apply the filters to any signal you choose contaminated with 60Hz additive noise. Show the signal before and after filtering.