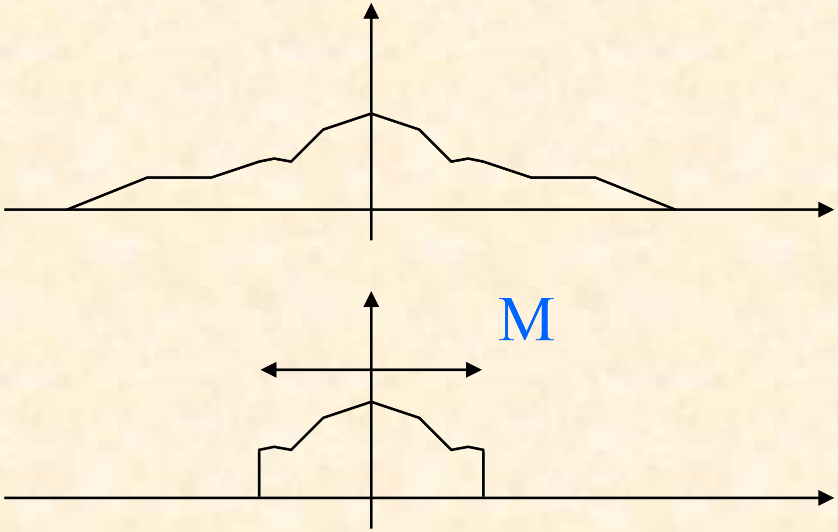
Parks-McClellan Method

- Often called the Remez exchange method.
- This method designs an optimal linear phase filter directly from the design specifications.
- This is the standard method for FIR filter design.
- In matlab, this method is available as remez().

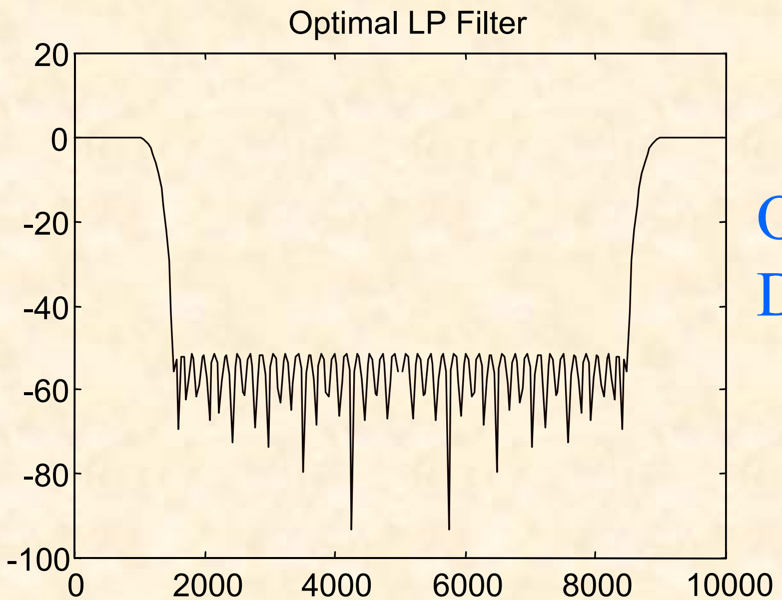
Approximation Errors

- From the theory of the Fourier series, the rectangular window design method (truncation of the impulse response) gives the best mean square (L^2) approximation to a desired frequency response for a given filter length M.



Minimax Design

- However simple truncation leads to adverse behaviour near discontinuity's and in the stop band.
- Better filters generally result from minimization of the maximum error (L_∞) or a frequency weighed error criterion.



Optimal Filter Design

Optimal Filters

Consider a linear phase FIR filter for which h[n] = h[-n]

Take Fourier Transforms

$$A(e^{j\omega}) = \sum_{n=-L}^{L} h[n]e^{-j\omega n}, \omega = 2\pi f / f_s$$

with $L = M / 2$

$$A(e^{j\omega}) = h[0] + \sum_{n=1}^{L} 2h[n]\cos(\omega n)$$

Now the terms $\cos(\omega n)$ can be expressed as polynomials of degree n in $\cos\omega$

Chebyshev Polynomials

$$\cos \omega n = T_n(\cos \omega)$$

where $T_n(x)$ is the nth order Chebyshev polynomial of the first kind defined by the recursion

$$T_0(x) = 1\,, T_1(x) = x$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$$

for example

$$T_2(x) = 2x^2 - 1;\; T_3(x) = 4x^3 - 3x$$

Thus A can be rewritten as a polynomial of degree L in cos ω.

$$A(e^{j\omega}) = \sum_{k=0}^{L} a_k (\cos\omega)^k$$

If we replace cos ω with x we obtain

$$A(e^{j\omega}) = P(x)\Big|_{x=\cos\omega}$$

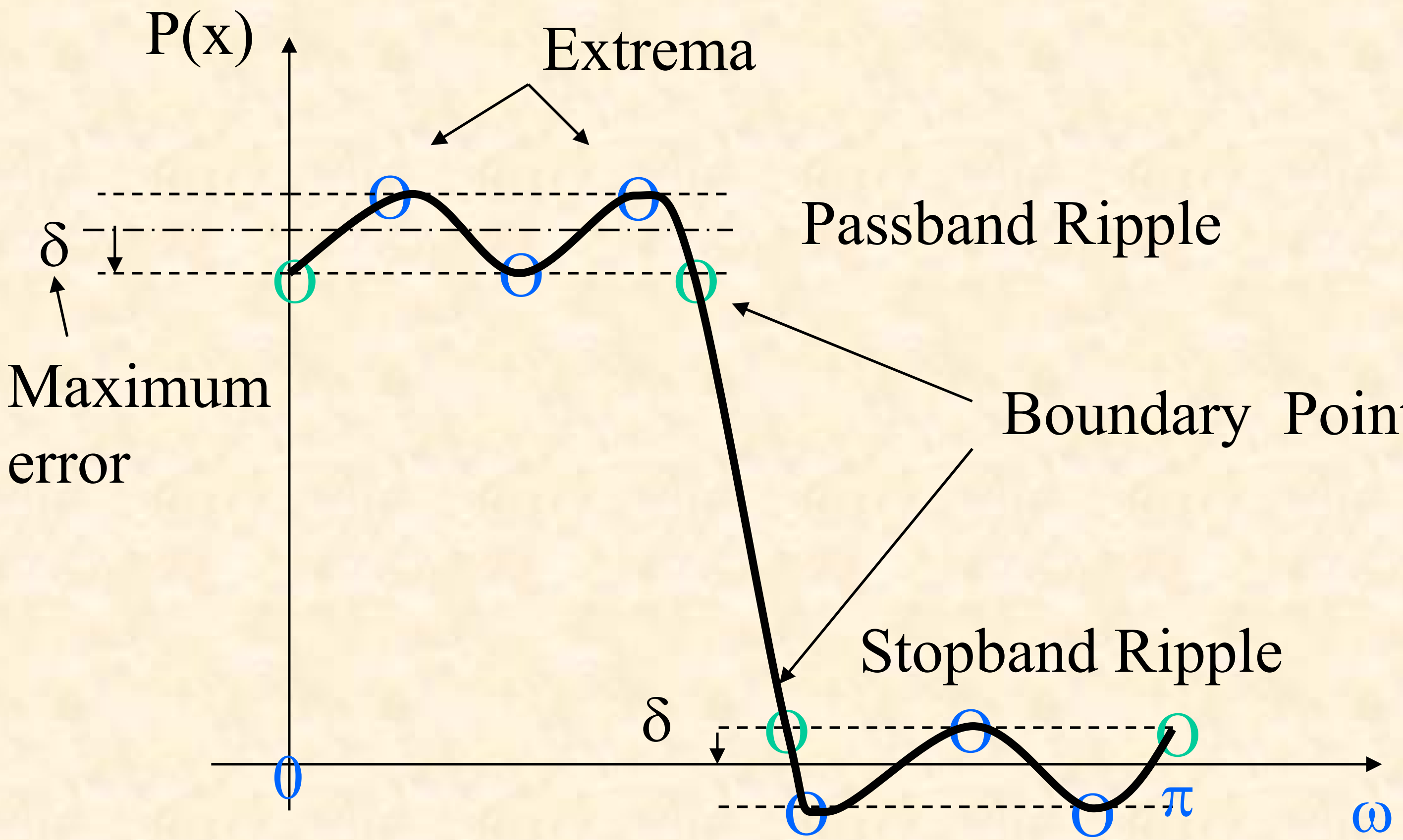
where $P(x)$ is the Lth order polynomial

$$P(x) = \sum_{k=0}^{L} a_k x^k$$

Optimal Filters

- **Alternation Theorem** : The polynomial of degree L that minimises the maximum error will have at least L+2 extrema. The optimal frequency response will just “kiss” the maximum ripple bounds
- Extrema must occur at the pass and stop band edges and at either $\omega=0$ or π or both.
- Now the derivative of a polynomial of degree L is a polynomial of degree L-1, which can be zero in at most L-1 places. So the maximum number of local extrema is the L-1 local extrema plus the 4 band edges. That is L+3.

Optimal Response



L=7th order polynomial, number of extrema = 10

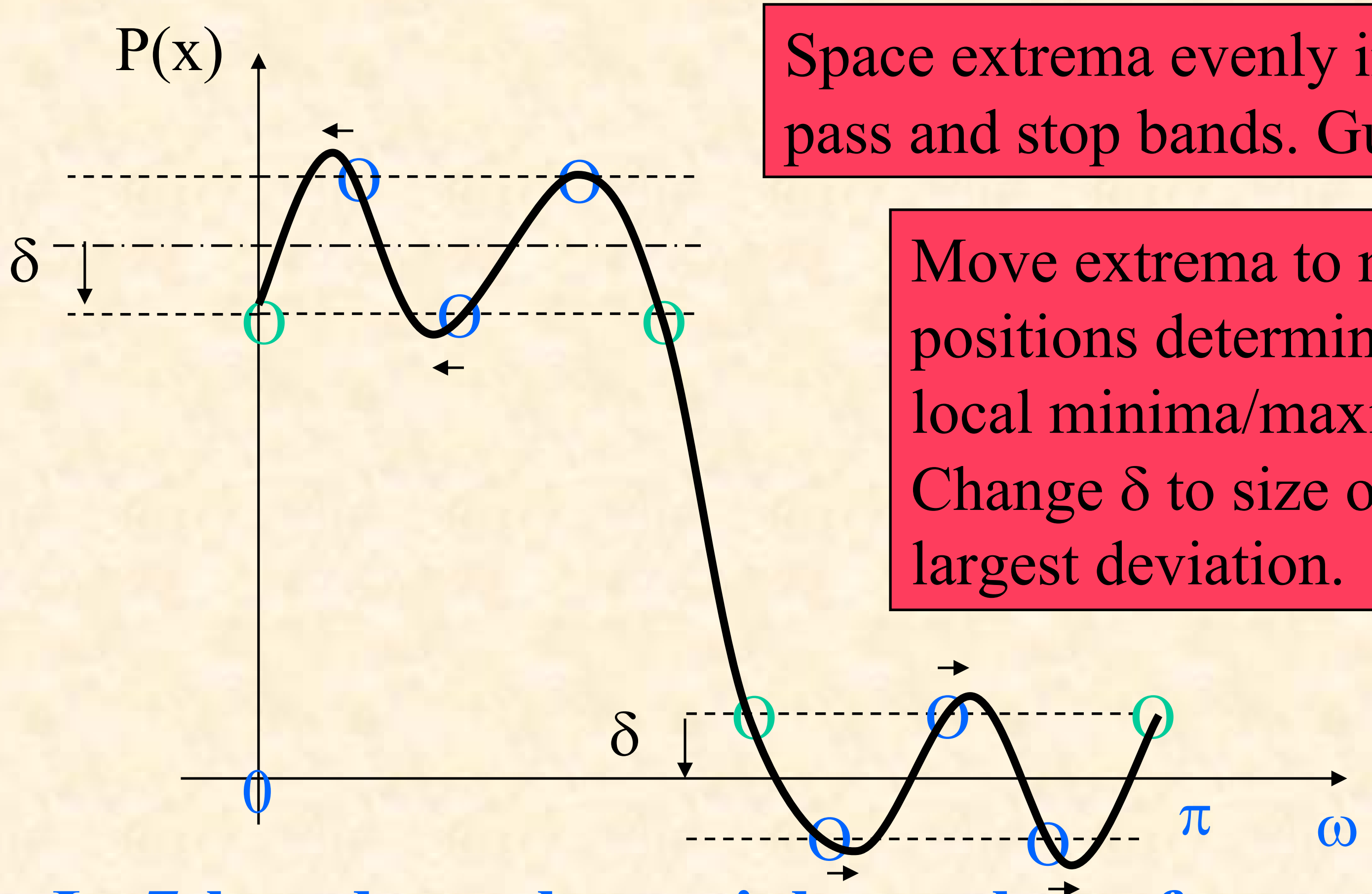
The Method

- We know where the boundary points are from the band edge specifications. At least 3 of these points must be extrema.
- We know how many local extrema there are from the estimated filter length (harris formula or similar) but we don’t know their positions.
- Guess the positions of the extrema are evenly spaced in the pass and stop bands.
- Perform polynomial interpolation and reestimate positions of local extrema.
- Move extrema to new positions and iterate until the extrema stop shifting.

Comments

- Given the positions of the extrema, there exists a formula for the optimum δ. However we don't know the optimum δ nor the exact positions of the extrema.
- Thus we need to iterate. Assume the positions of the extrema, calculate δ, move the extrema, recalculate δ, until δ stops changing.
- The algorithm generally converges in about 12 iterations.

Progress of Algorithm

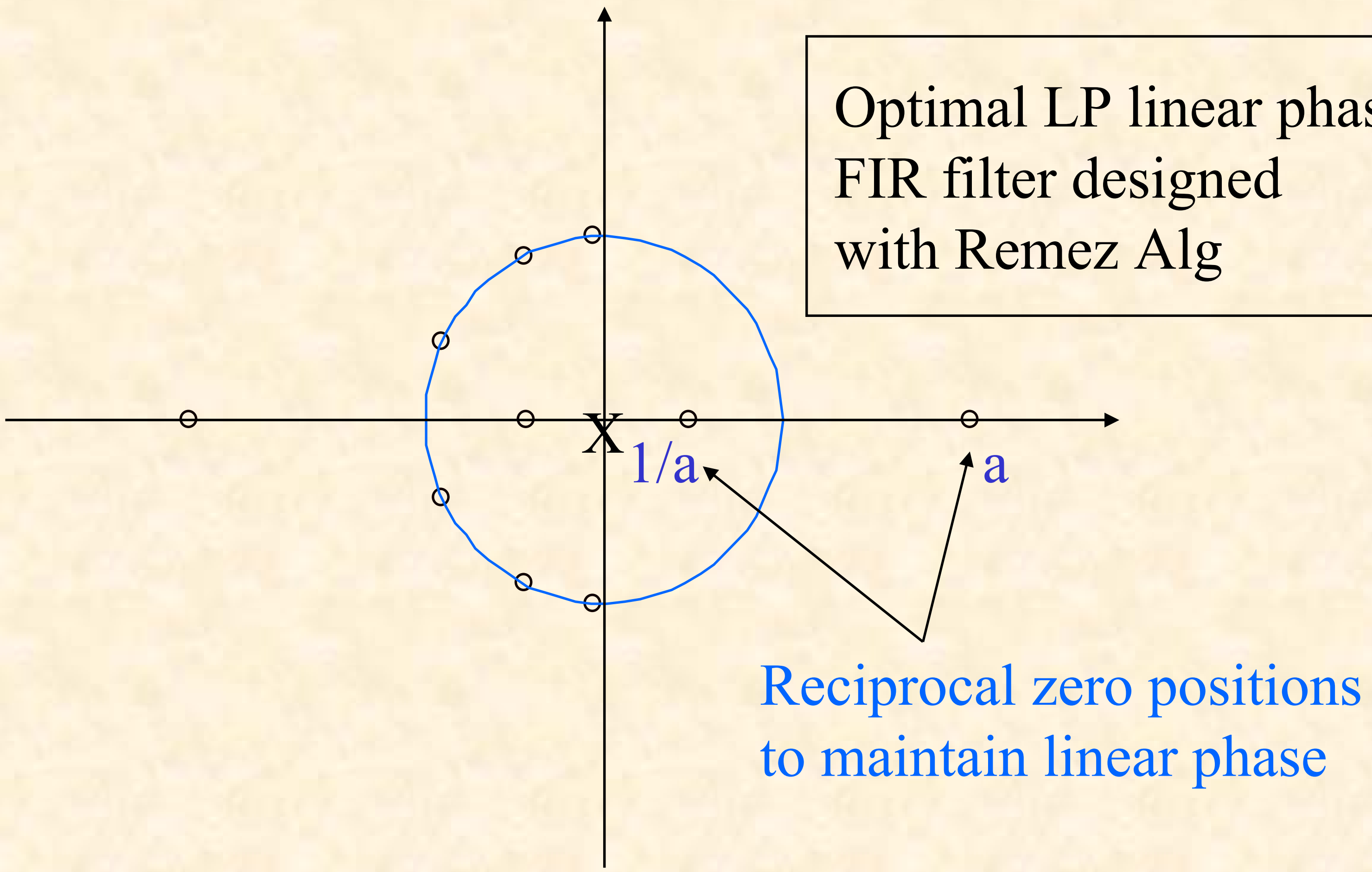


Space extrema evenly in pass and stop bands. Guess δ

Move extrema to new positions determined by local minima/maxima. Change δ to size of largest deviation.

L=7th order polynomial, number of extrema = 10

Pole-Zero Diagram



Optimal LP linear phase FIR filter designed with Remez Alg

Example

lpremez

- Design a low pass filter with the following specifications
  Fs = 10 kHz
  Fc = 1 kHz (-3dB)
  A = 60dB at 1.5 kHz
- From harris formula, N >10000x60/(500x22) = 55

Types of Filters

- These types of filters are easy to design with the standard remez algorithm
  - Low Pass, High Pass, Band Pass, Band Stop,Multiband, Arbitrary Magnitude
  - In Matlab may need to use the 'hilbert' switch for some filters (e.g. even length high pass, hpremez.m)
- The following are easy to design with a slight modification to the algorithm. These filters generally have zero response near DC and Fs/2.
  - Differentiator $H(f) = j2\pi f$
  - Hilbert Transform, $H(f) = -j\,\mathrm{sgn}(f)$

Differentiator

- Differentiation corresponds to a filter $H(f)=j2\pi f$.
- The inverse transform of this filter is

$$h(n) = (-1)^n \frac{1}{n}, n \neq 0$$

- The first three terms of this response are (-1,0,1) which is the standard approximation for differentiation learnt in calculus. (central finite difference)

The spectral response of this truncated filter is

$$H(z) = z^{+1} - z^{-1}$$

$$H(f) = e^{+j2\pi f} - e^{-j2\pi f}$$

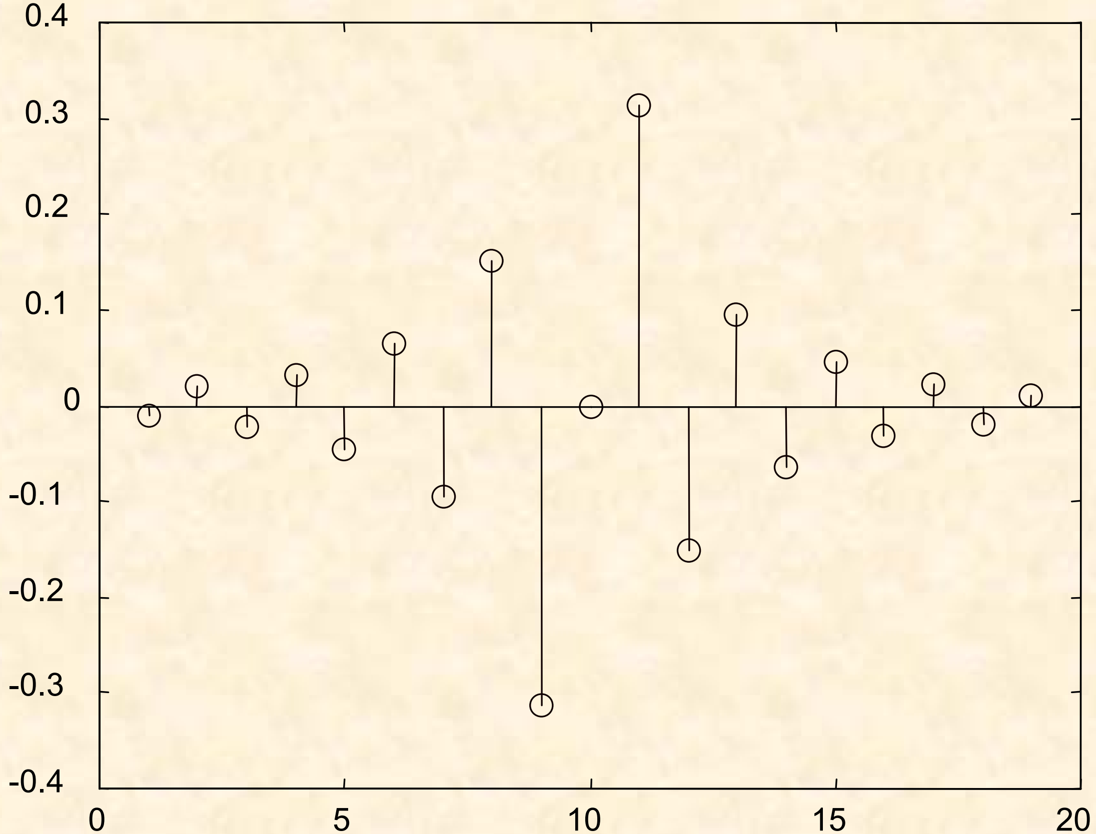
$$= 2j\sin(2\pi f)$$

which for small f is approximately

$$H(f) \cong j4\pi f$$

In the design we generally specify a bandwidth over which the filter acts as a differentiator.

Impulse Response of Differentiator



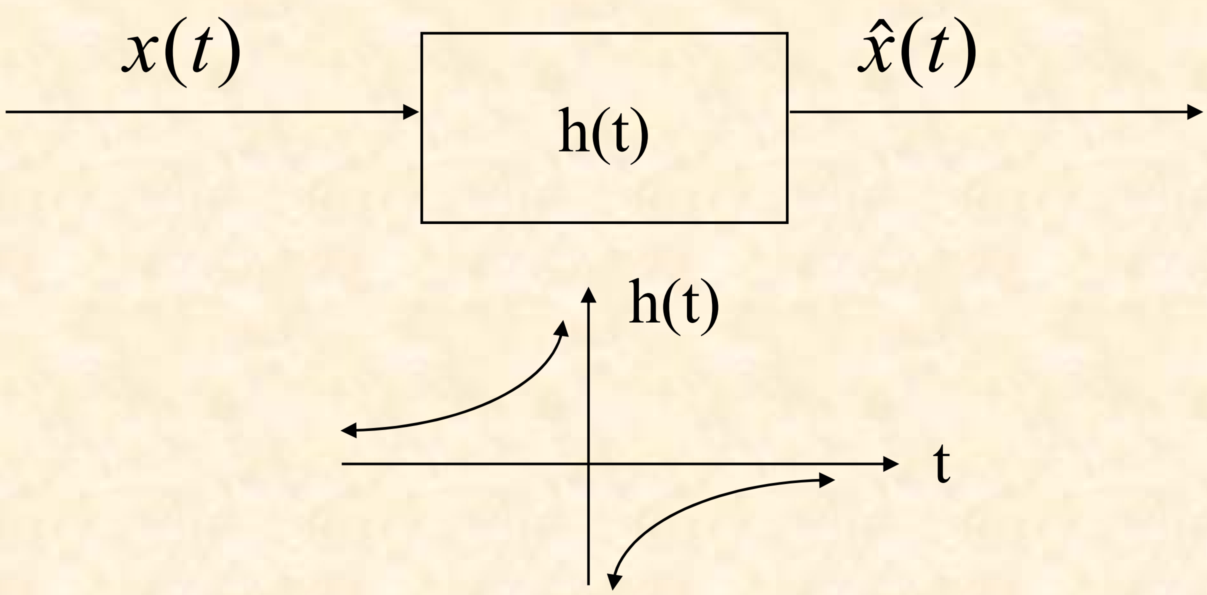
Hilbert Transform

- A Hilbert Transform is used to form the Analytic Signal. This signal is a complex signal with the property that its FT has only positive (or negative) frequencies. [e.g., the analytic signal corresponding to $\cos(2\pi ft)$ is $\exp(j2\pi ft)$]
- This signal is classically used in Single Sideband Modulation.
- The Hilbert Transform is defined by

$$\hat{h}(t) = \int_{-\infty}^{+\infty} \frac{h(\tau)}{t-\tau}\,d\tau$$

Hilbert Transform

What does this mean? This is the instruction to perform a convolution of the input signal h(t) with a filter response 1/t.



Hilbert Transform

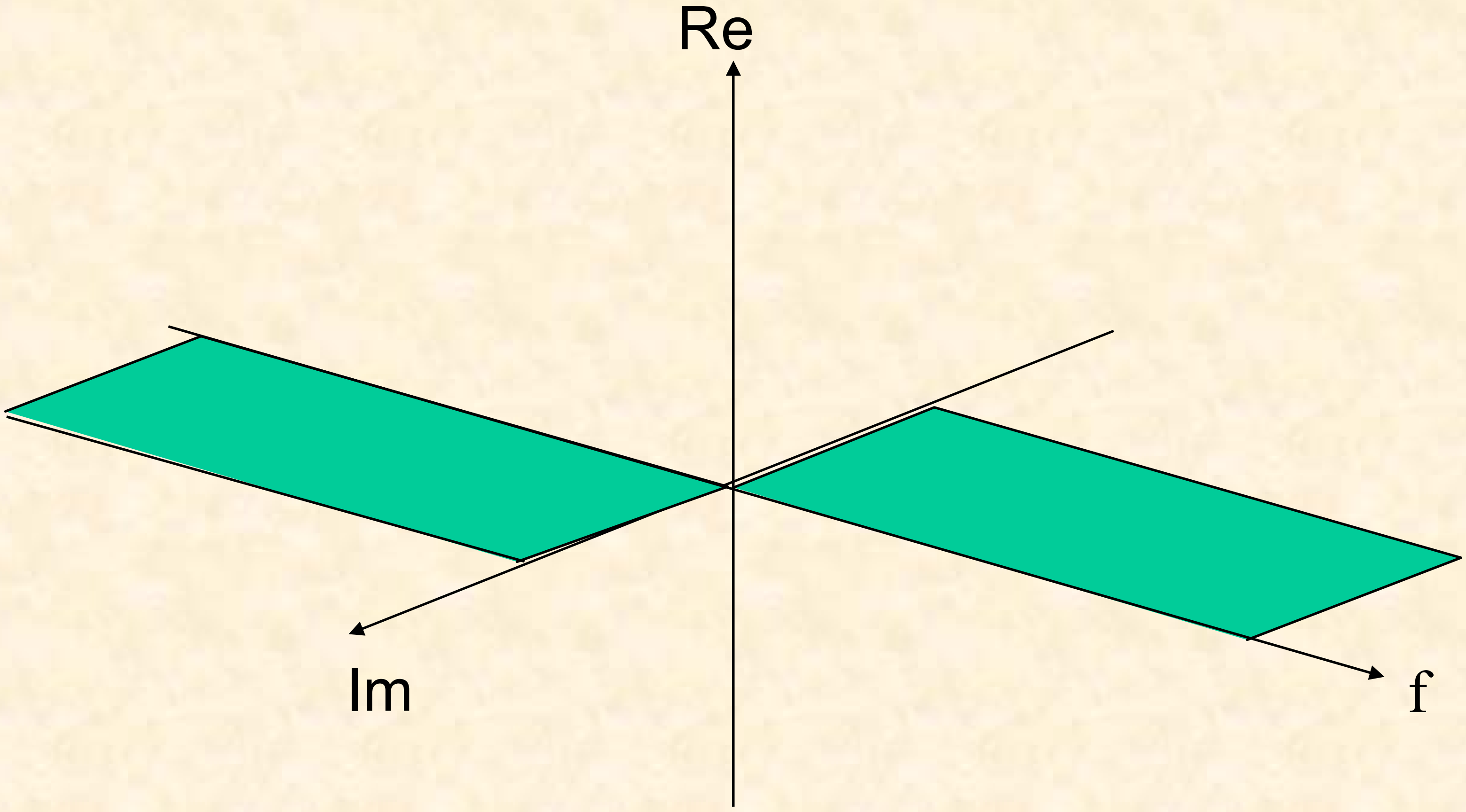
We can form the HT by designing a filter with the frequency response corresponding to the impulse response 1/t. This response is

$$h(f) = -j\,\mathrm{sgn}(f)$$

This frequency response describes a wide-band 90 degrees phase shifter...that is positive frequencies are shifted -90 degrees and negative frequencies are shifted +90 degrees.

Note: HT of $\sin(2\pi ft)$ is $\cos(2\pi ft)$

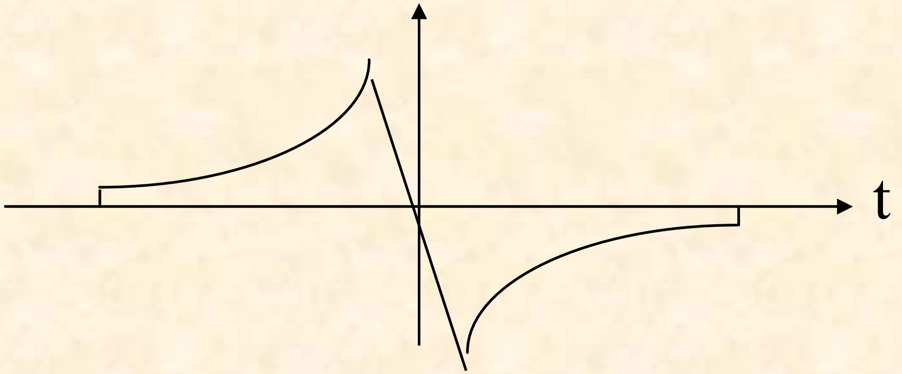
Hilbert Transform



This is a terrible filter! The impulse response has a bad singularity at the origin. Further, the function extends to $\pm\infty$ on the time line.

To realize this filter we need to modify the response

- Truncate to make finite
- Change behaviour at origin, affects low freq performance



Discrete Time HT

For discrete time implementation, the spectrum must be periodic in the sample rate. The form of the sampled data spectrum is seen to be

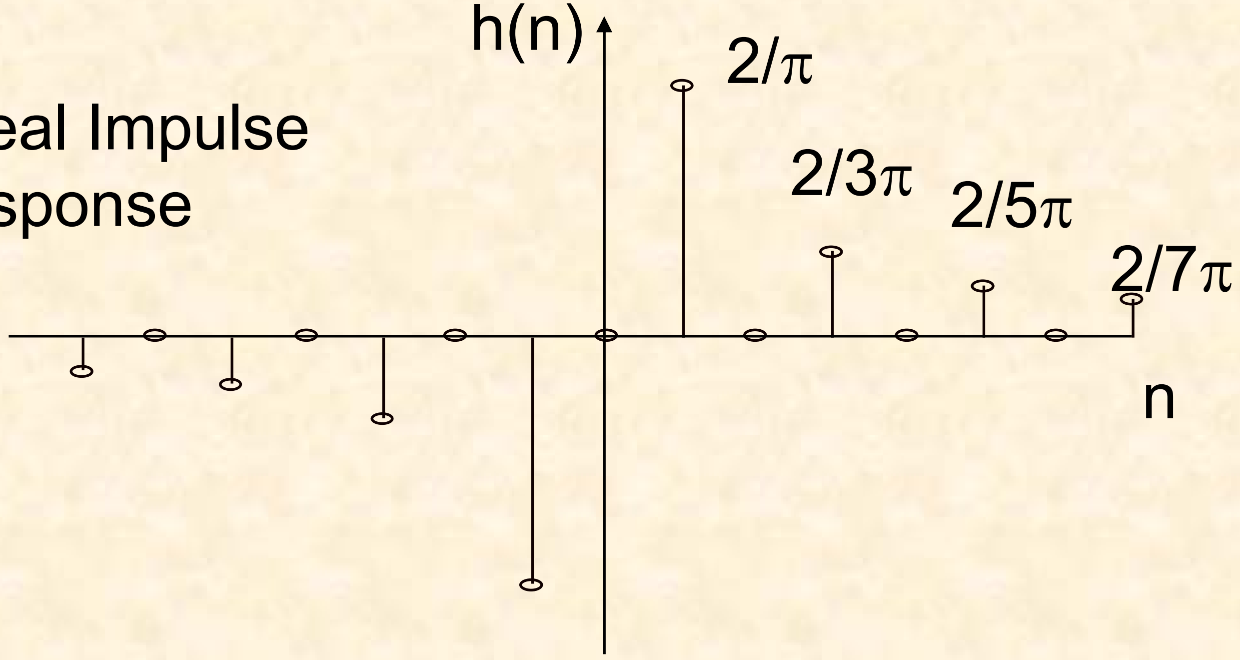
$$H(f) = -j, 0 < f < f_s / 2$$
$$+j, -f_s / 2 < f < 0$$

The impulse response is

$$h(n) = \frac{2}{\pi} \sum_{-\infty}^{+\infty} \frac{1}{n} \forall \text{ odd} \quad n$$

Discrete Time HT

Ideal Impulse response



Other Effects

- Coefficient Quantization in Finite Precision
  - As a rough guide the maximum stopband attenuation must be about 5*n dB, where n is the number of bits. For example the noise floor is 80 dB with 16 bit arithmetic.
  - Can optimise filter design taking into account finite precision by performing an exhaustive search around desired function. There are algorithms available for this but the computational load becomes prohibitive for N greater than about 40.