Design of FIR and IIR Filters

FIR as a class of LTI Filters

Transfer function of the filter is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

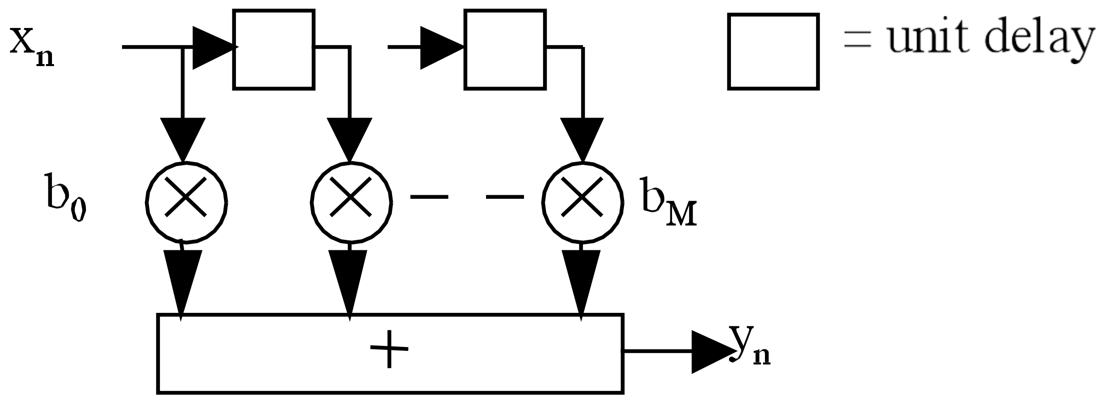
Finite Impulse Response (FIR) Filters:

N = 0, no feedback

FIR Filters

Let us consider an FIR filter of length M (order N=M-1, watch out!
order – number of delays)

$$y(n) = \sum_{k=0}^{M-1} b_k x(n-k) = \sum_{k=0}^{M-1} h(k)\, x(n-k)$$



FIR filters

Can immediately obtain the impulse response, with $x(n) = \delta(n)$

$$h(n) = y(n) = \sum_{k=0}^{M-1} b_k \delta(n-k) = b_n$$

The impulse response is of finite length M, as required

Note that FIR filters have only zeros (no poles). Hence known also as **all-zero** filters

FIR filters also known as **feedforward** or **non-recursive,** or **transversal**

FIR Filters

Digital FIR filters cannot be derived from analog filters – rational analog filters cannot have a finite impulse response.

Why bother?

1. They are inherently stable
2. They can be designed to have a linear phase
3. There is a great flexibility in shaping their magnitude response
4. They are easy and convenient to implement

Remember very fast implementation using FFT?

FIR Filter using the DFT

FIR filter:

$$y(n) = \sum_{k=0}^{M-1} h(k)\, x(n-k)$$

Since $h(n)$ and $x(n)$ are finite-duration sequences, their convolution is also finite in duration. The duration of the sequence $y(n)$ is $L + M - 1$.

Let us consider $N \geq L + M - 1$. Let us pad the sequences $h(n)$ and $x(n)$ with zeros to increase their lengths to N and perform the $N-$point DFT. The frequency-domain equivalent is

$$Y(k) = H(k)\, X(k)$$

with $k = \frac{2\pi k}{N}$.

Now *N*-point DFT (*Y(k)*) and then *N*-point IDFT (*y(n)*) can be used to compute standard convolution product and thus to perform linear filtering (given how efficient FFT is)

Linear-phase filters

The ability to have an exactly linear phase response is the one of the most important of FIR filters

$$H(\omega) = |H(\omega)| e^{j\phi(\omega)} \quad \text{where } \phi(\omega) = -\omega n_0$$

A general FIR filter does not have a linear phase response but this property is satisfied when

$$h(n) = \pm h(M-1-n), \quad n = 0, 1, \ldots, M-1.$$

four linear phase filter types

Impulse response	# coefs	$H(\omega)$	Type
$h(n) = h(M-1-n)$	Odd	$e^{-j\omega(M-1)/2}\left(h\left(\frac{M-1}{2}\right) + 2\sum_{k=1}^{(M-3)/2} h\left(\frac{M-1}{2} - k\right)\cos(\omega k)\right)$	1
$h(n) = h(M-1-n)$	Even	$e^{-j\omega(M-1)/2} 2\sum_{k=1}^{(M-3)/2} h\left(\frac{M}{2} - k\right)\cos\left(\omega\left(k - \frac{1}{2}\right)\right)$	2
$h(n) = -h(M-1-n)$	Odd	$e^{-j[\omega(M-1)/2 - \pi/2]}\left(2\sum_{k=1}^{(M-1)/2} h\left(\frac{M-1}{2} - k\right)\sin(\omega k)\right)$	3
$h(n) = -h(M-1-n)$	Even	$e^{-j[\omega(M-1)/2 - \pi/2]} 2\sum_{k=1}^{(M-1)/2} h\left(\frac{M}{2} - k\right)\sin\left(\omega\left(k - \frac{1}{2}\right)\right)$	4

FIR Design Methods

- Impulse response truncation – the simplest design method, has undesirable frequency domain-characteristics, not very useful but intro to …
- Windowing design method – simple and convenient but not optimal, i.e. order achieved is not minimum possible
- Optimal filter design methods

Back to Our Ideal Low- pass Filter Example

Let us consider for example a simple ideal lowpass filter defined by

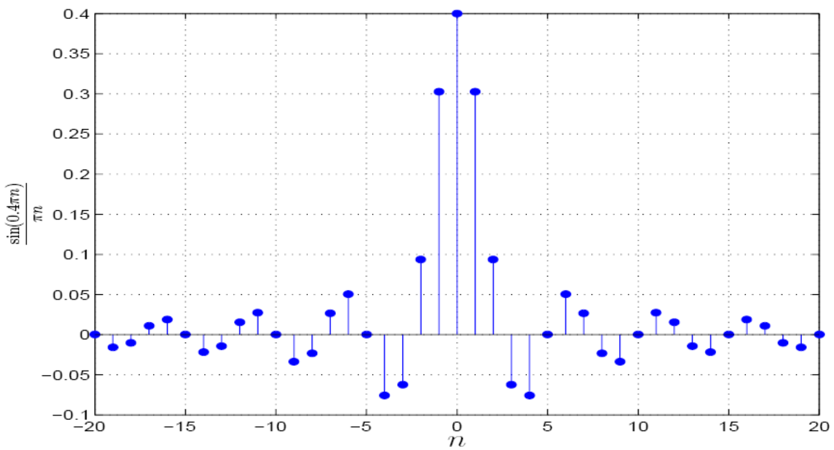
$$H_d(\omega) = \begin{cases} 1 \text{ if } |\omega| \le \omega_c \\ 0 \text{ if } \omega_c < |\omega| < \pi. \end{cases}$$

It can be shown easily that the impulse response is given by

$$h_d(n) = \frac{\omega_c}{\pi} \frac{\sin \omega_c n}{\omega_c n}.$$

Desired impulse response has a sinc shape which is non-causal and infinite in duration.

clearly cannot be implemented in practice



Approximation via truncation

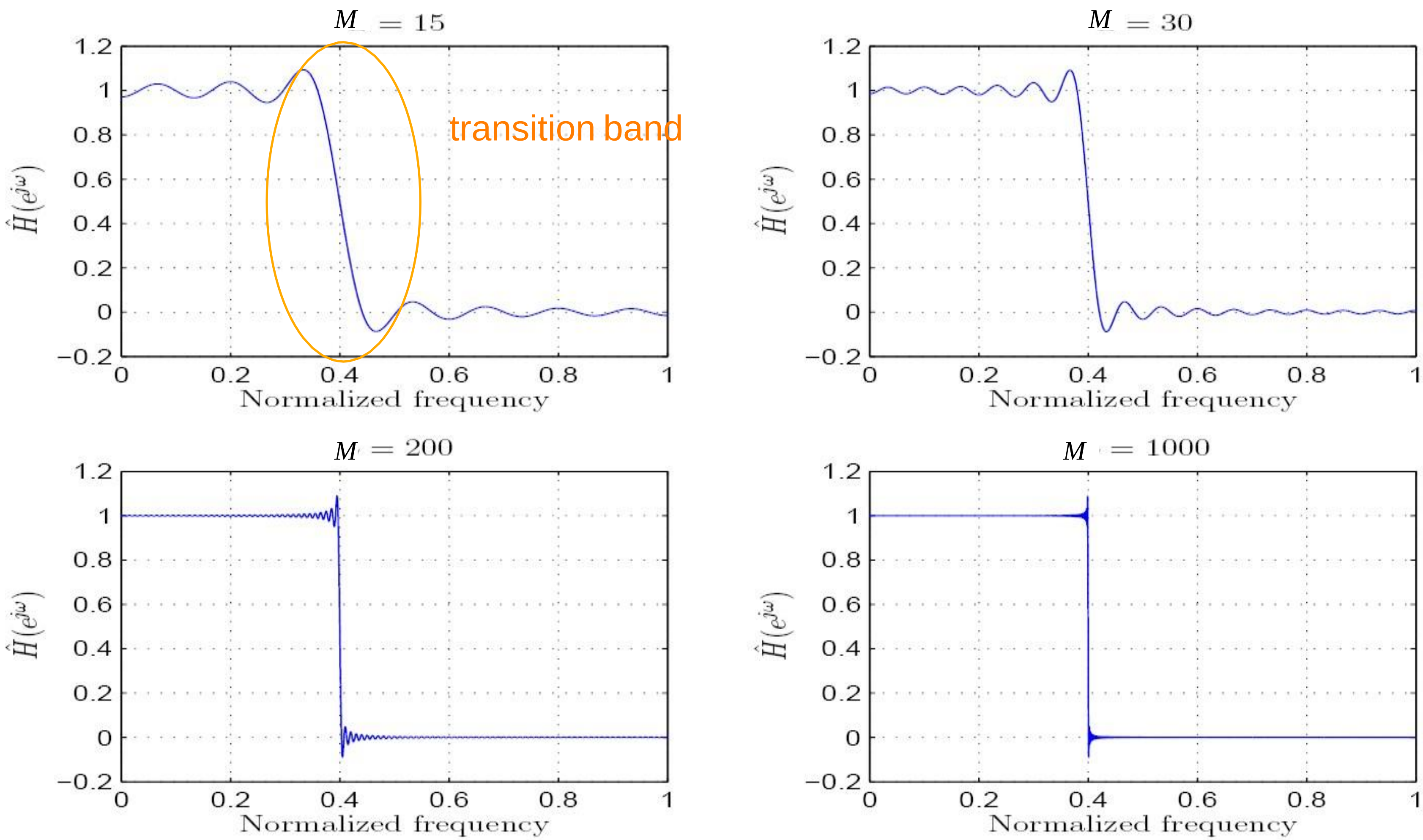
• $h(n)$ infinite in duration $\Rightarrow$ to compute any sample of the output, we need to know all samples of the input, both in the past and in the future!!
$\Rightarrow$ Unacceptable, so what if we just clipp off the sinc at some large n

$$\widehat{h}(n) = \frac{\sin(n\omega_c)}{\pi n} \text{ for } |n| \leq M \text{ and 0 otherwise.}$$

• Here is what the frequency response now looks like for $\omega_c = 0.4\pi$ and different values of M.

• One observes *ripples* in both passband/stopband and transition not abrupt (leading to *transition band*).

Approximated filters obtained by truncation



Though the transition band gets narrower as $M \to \infty$, the ripple remains!

Window Design Method

To be expected …

Truncation is just pre-multiplication by a rectangular window

– This is not very clever

– obviously one introduces a delay

$$w(n) = \begin{cases} 1 \text{ if } n = 0, 1, \ldots, M-1 \\ 0 \text{ otherwise.} \end{cases}$$

Fourier transform $H(\omega)$ of the truncated filter $h(n) = h_d(n)\, w(n)$ is

$$H(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(\lambda)\, W(\omega - \lambda)\, d\lambda$$

spectrum convolution

where $W(\omega)$ is the Fourier transform of the rectangular window

$$W(\omega) = e^{-j\omega(M-1)/2} \frac{\sin(\omega M/2)}{\sin(\omega/2)}$$

Rectangular Window Frequency Response

Fourier transform of the rectangular window

$$W(\omega) = e^{-j\omega(M-1)/2} \frac{\sin(\omega M/2)}{\sin(\omega/2)}$$

admits as magnitude and phase responses

$$\begin{aligned} |W(\omega)| &= \frac{|\sin(\omega M/2)|}{|\sin(\omega/2)|}, \ \ |\omega| < \pi, \\ \theta(\omega) &= \begin{cases} -\omega(M-1)/2 & \text{when } \sin(\omega M/2) \geq 0 \\ -\omega(M-1)/2 + \pi & \text{when } \sin(\omega M/2) < 0 \end{cases} \end{aligned}$$

⇨ i.e. $W(\omega)$ has a piecewise linear phase

Window Design Method

- Practically, one uses **truncated** & **delayed** impulse response

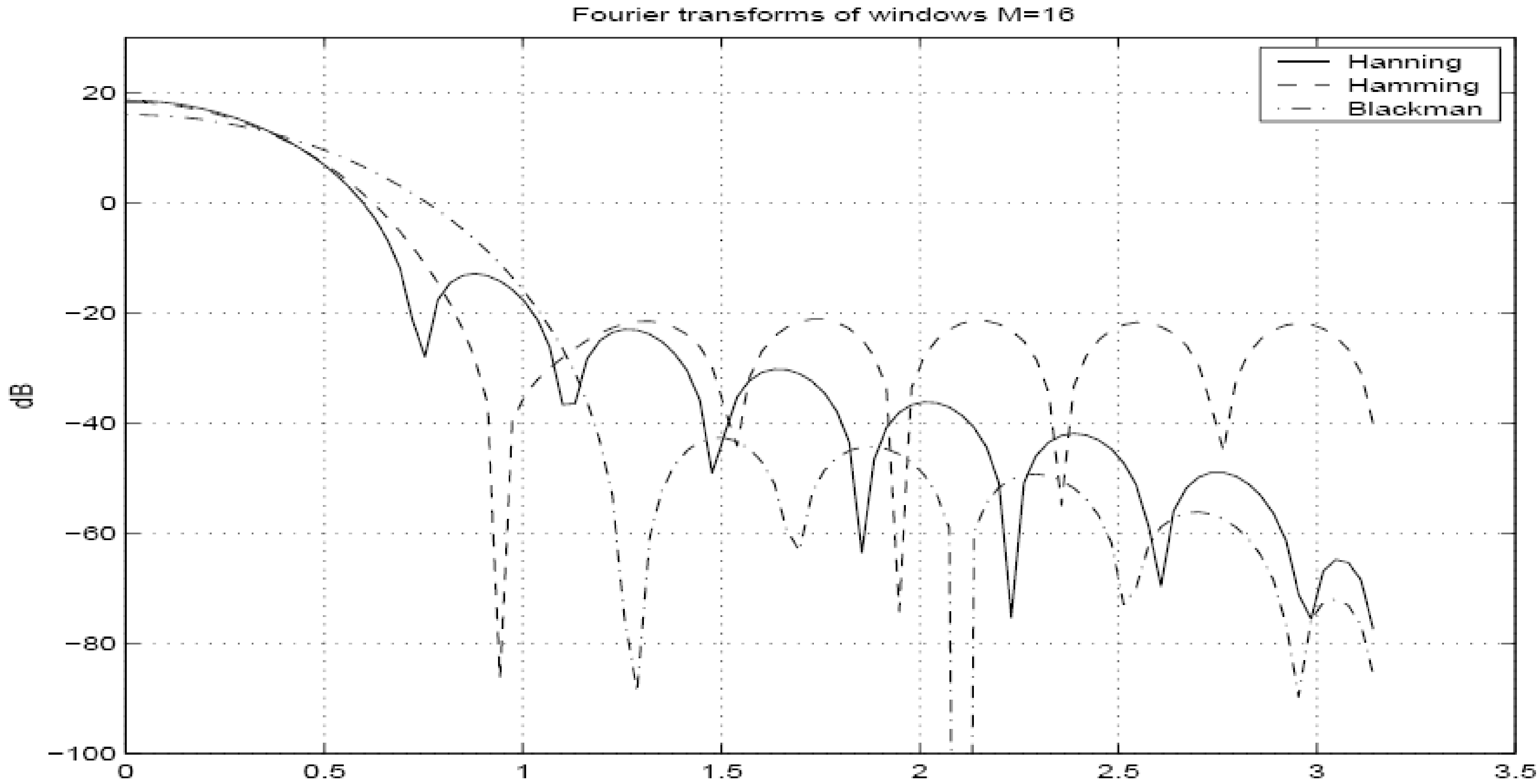
$$\widetilde{h}(n) = \widehat{h}\left(n - \frac{M-1}{2}\right) \text{ where } \widehat{h}(n) = \frac{\omega_c}{\pi}\frac{\sin(\omega_c n)}{\omega_c n}1_{\left\{-\frac{M-1}{2},\ldots,\frac{M-1}{2}\right\}}(n)$$

where M is the *filter length* & $N = M - 1$ is known as *filter order*.

- Delaying operation $\rightarrow$ introduce linear phase term.

$\Rightarrow$ The resulting filter is causal and has a linear phase.

Windows –Magnitude of Frequency Response

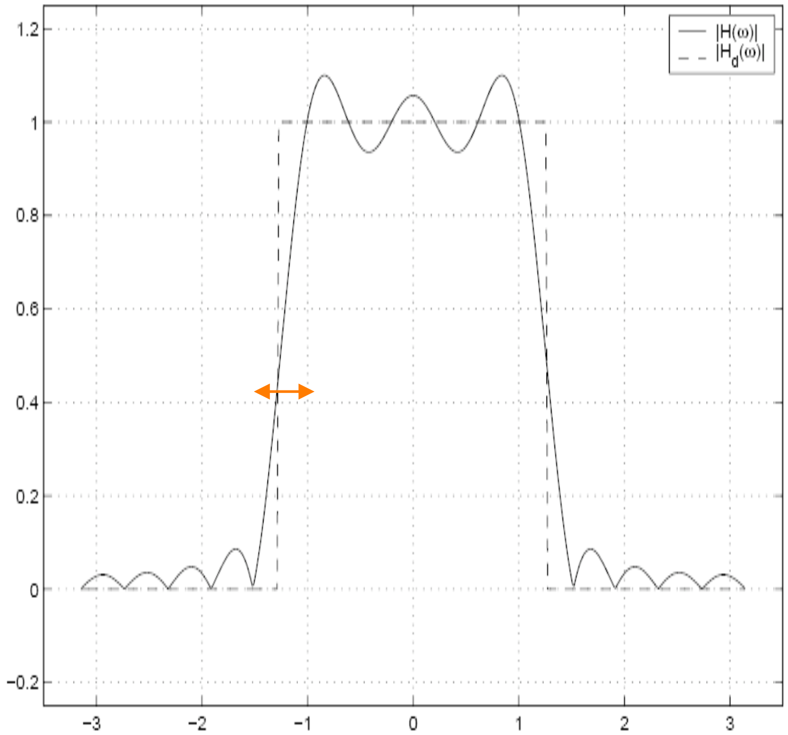


Summary of Windows Characteristics

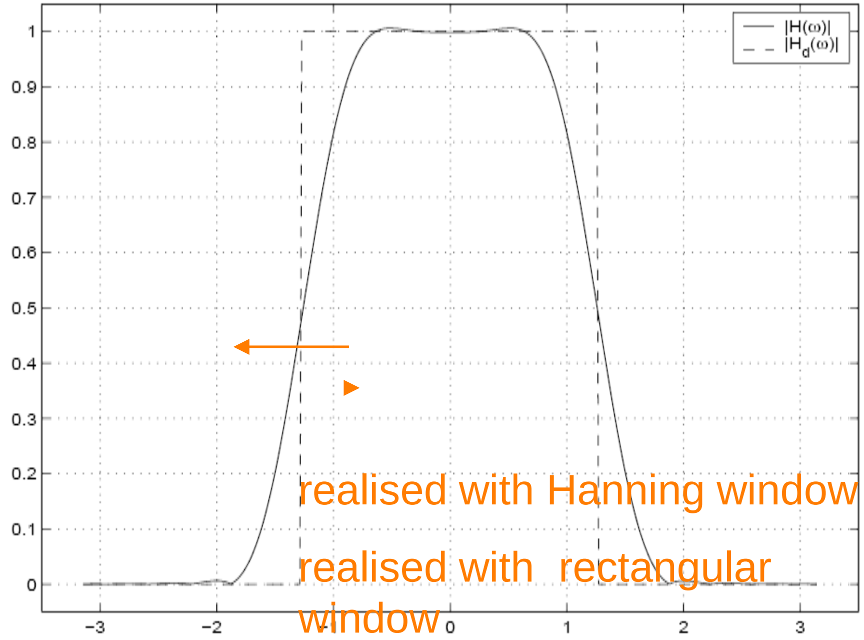
Window's name	Mainlobe	Mainlobe/sidelobe	Peak $20\log_{10}\delta$
Rectangular	$4\pi/\ M$	-13dB	-21dB
Hanning	$8\pi/M$	-32dB	-44dB
Hamming	$8\pi/M$	-43dB	-53dB
Blackman	$12\pi/M$	-58dB	-74dB

We see clearly that a wider transition region (wider main-lobe) is compensated by much lower side-lobes and thus less ripples.

Filter realised with rectangular/Hanning windows



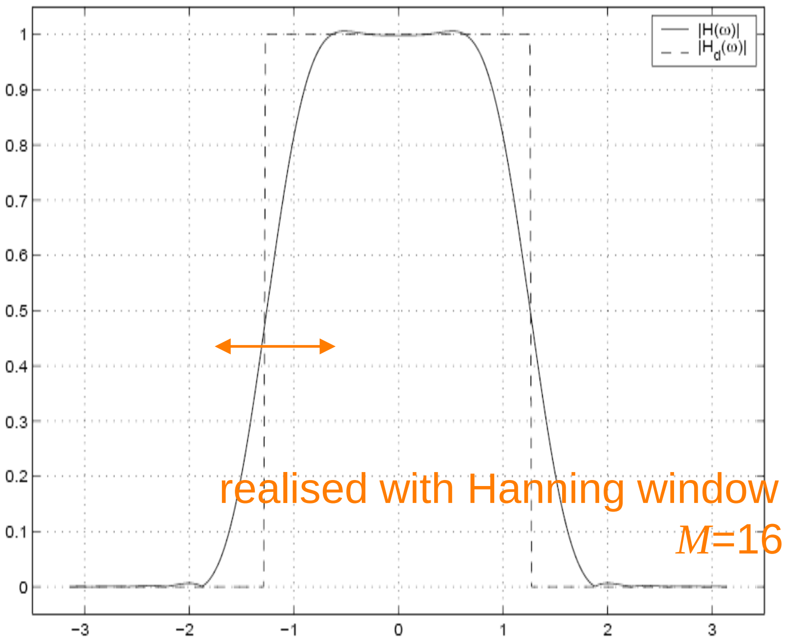
Back to our ideal filter

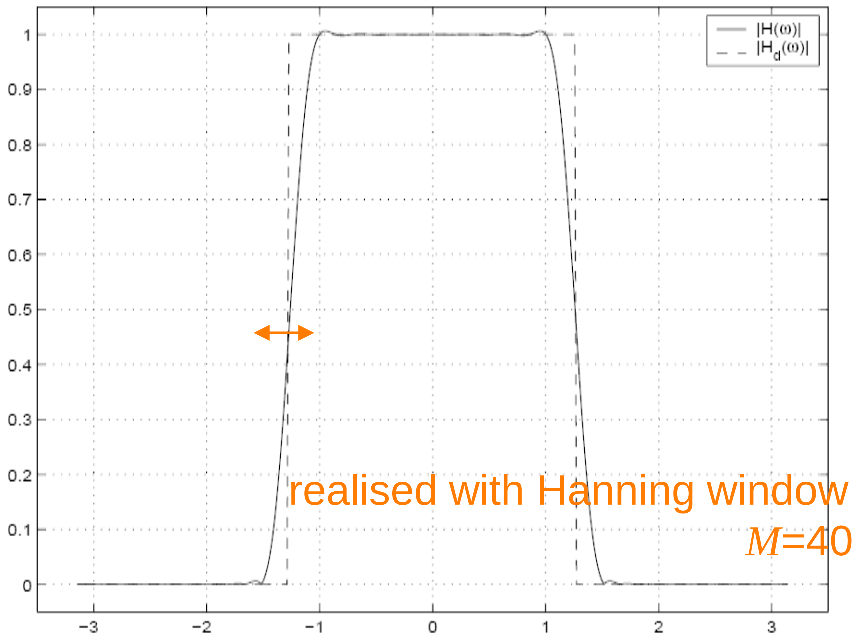


M=16 M=16

There are much less ripples for the Hanning window but that the transition width has

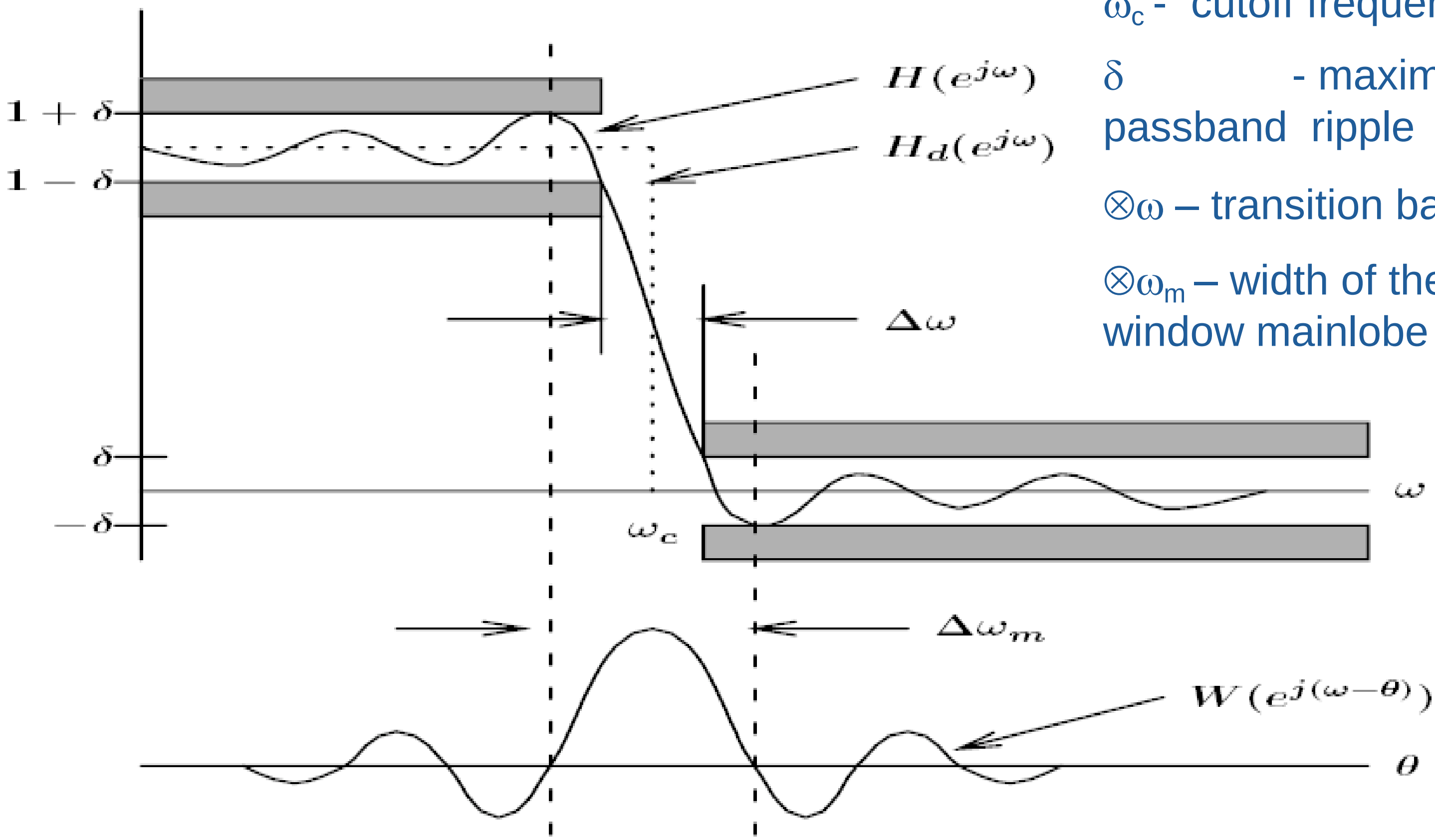
Filter realised with Hanning windows





Transition width can be improved by increasing the size of the Hanning window to $M = 40$

Specification necessary for Window Design Method



ω_c - cutoff frequency

δ - maximum passband ripple

⊗ω – transition bandwidth

$\otimes\omega_m$ – width of the window mainlobe

Response must not enter shaded regions

Passband / stopband ripples

Passband / stopband ripples are often expressed in dB:

passband ripple = 20 $log_{10}(1+\delta_p)$ dB,
or peak-to-peak passband ripple ≅ 20 $log_{10}(1+2\delta_p)$ dB;
minimum stopband attenuation = -20 $log_{10}(\delta_s)$ dB.

Example: $\delta_p = 6\%$ ⇨ peak-to-peak passband ripple ≅ 20 $log_{10}(1+2\delta_p)$ = 1dB;

$\delta_s = 0.01$ ⇨ minimum stopband attenuation = -20 $\log_{10}(\delta_s)$ = 40dB.

The band-edge frequencies ω_s and ω_p are often called **corner frequencies**, particularly when associated with specified gain or attenuation (e.g. gain = -3dB).

Summary of Window Design Procedure

- Ideal frequency response has infinite impulse response

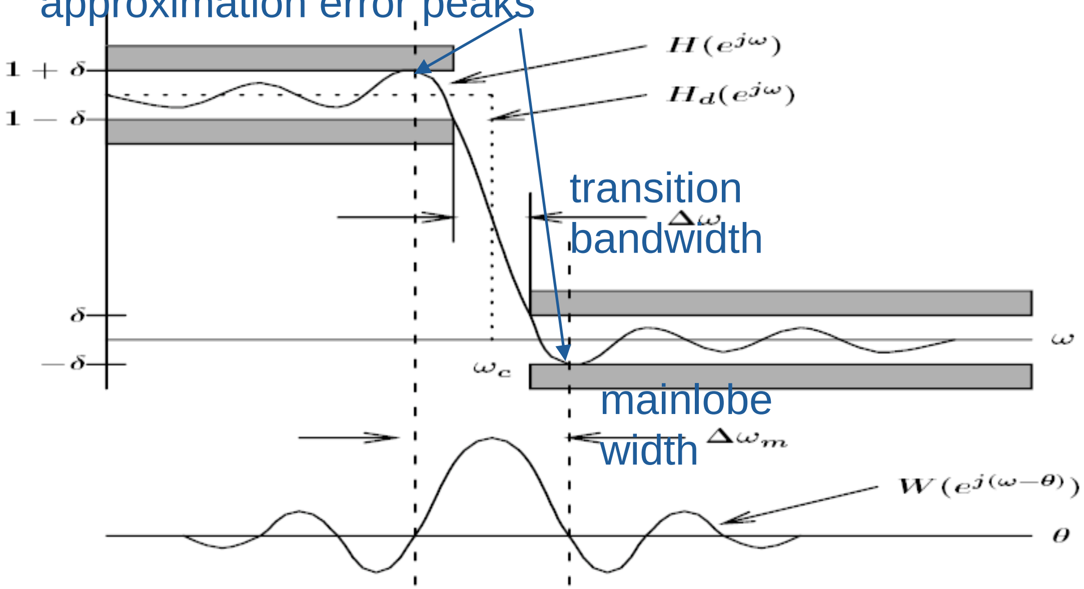
- To be implemented in practice it has to be
  - truncated
  - shifted to the right (to make is causal)

- Truncation is just pre-multiplication by a rectangular window
  - the filter of a large order has a narrow transition band
  - however, sharp discontinuity results in side-lobe interference independent of the filter's order and shape Gibbs phenomenon

- Windows with no abrupt discontinuity can be used to reduce Gibbs oscillations (e.g. Hanning, Hamming, Blackman)

Summary of the Key Properties of the Window Design Method

1. Equal transition bandwidth on both sides of the ideal cutoff frequency.
2. Equal peak approximation error in the pass-band and stop-band.
3. Distance between approximation error peaks is approximately equal to the width of the window main-lobe.
4. The width of the main-lobe is wider than the transition band.



5. Peak approximation error is determined by the window shape, independent of the filter order.

Summary of the windowed FIR filter design procedure

1. Select a suitable window function

2. Specify an ideal response $H_d(\omega)$

3. Compute the coefficients of the ideal filter $h_d(n)$

4. Multiply the ideal coefficients by the window function to give the filter coefficients

5. Evaluate the frequency response of the resulting filter and iterate if necessary (typically, it means increase M if the constraints you have been given have not been satisfied)

Multi-band Design

- So far, only lowpass filter: how do we design highpass, bandpass, etc. filters?
$\Rightarrow$ treat them as sums and differences of lowpass filters.

- *Example*: design the following highpass filter

$$H_d(\omega) = 1_{(-\pi,-\omega_c)\cup(\omega_c,\pi)}(\omega)$$

It can be rewritten as

$$H_d(\omega) = \underbrace{1_{(-\pi,\pi)}(\omega)}_{\text{LP filter cutoff } \pi} - \underbrace{1_{(-\omega_c,\omega_c)}(\omega)}_{\text{LP filter cutoff } \omega_c} \Rightarrow h(n) = \frac{\sin(\pi n)}{\pi n} - \frac{\sin(\omega_c n)}{\pi n}.$$

- Now we use window and delay this answer by $M/2$ to make it causal.

Frequency sampling method

- Drawbacks of the window design method:
  - Start with $H_d(\omega)$ and end up with approximation $H(\omega)$, difficult to predict values of $H(\omega)$ at some specific frequencies (not big problem...)
  - Computation of the IDTFT of arbitrary $H_d(\omega)$ may be difficult.

- *Simple idea*: sample the desired frequency response $H_d(\omega)$ at N frequencies unif. spaced over $[0, 2\pi)$, compute the IDFT of these N samples $\Rightarrow \{h(n)\}$.

- **Advantage**: The filter frequency response lands exactly on the specified values at the sampling points.

- **Drawback**: Difficult to control between those points (i.e. sinc-interpolated).

IIR vs FIR Filters

- FIR filters often employed in problems where linear phase required.
- When *phase distortion tolerable*, *IIR* are usually *favoured*
  - Typically require *less parameters* to achieve sharp cutoff filters.
  - Thus for given response specification, *lower computational complexity/less memory* (despite FFT cannot be used)
- Main problems of IIR filters.
  - *Difficult design.*
  - *Stability problems.*

IIR as a class of LTI Filters

Difference equation:

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=1}^{M} b_k x(n-k)$$

Transfer function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

To give an Infinite Impulse Response (IIR), a filter must be recursive, that is, incorporate feedback $N \neq 0$, $M \neq 0$

the recursive (previous output) terms feed back energy into the filter input and keep it going.

(Although recursive filters are not *necessarily* IIR)

IIR Filters Design from an Analogue Prototype

- Given filter specifications, direct determination of filter coefficients is too complex.

- Well-developed design methods exist for analogue low-pass filters

- Almost all methods rely on converting an analogue filter to a digital one

Analogue filter Rational Transfer Function

Assume an analog filter can be described by a rational transfer function ($\{\alpha_k\}$ and $\{\beta_k\}$ real-valued)

$$H_a(s) = \frac{\sum_{k=0}^{M} \beta_k s^k}{\sum_{k=0}^{N} \alpha_k s^k}$$

where $H_a(s)$ is the Laplace transform of the impulse response $h_a(t)$

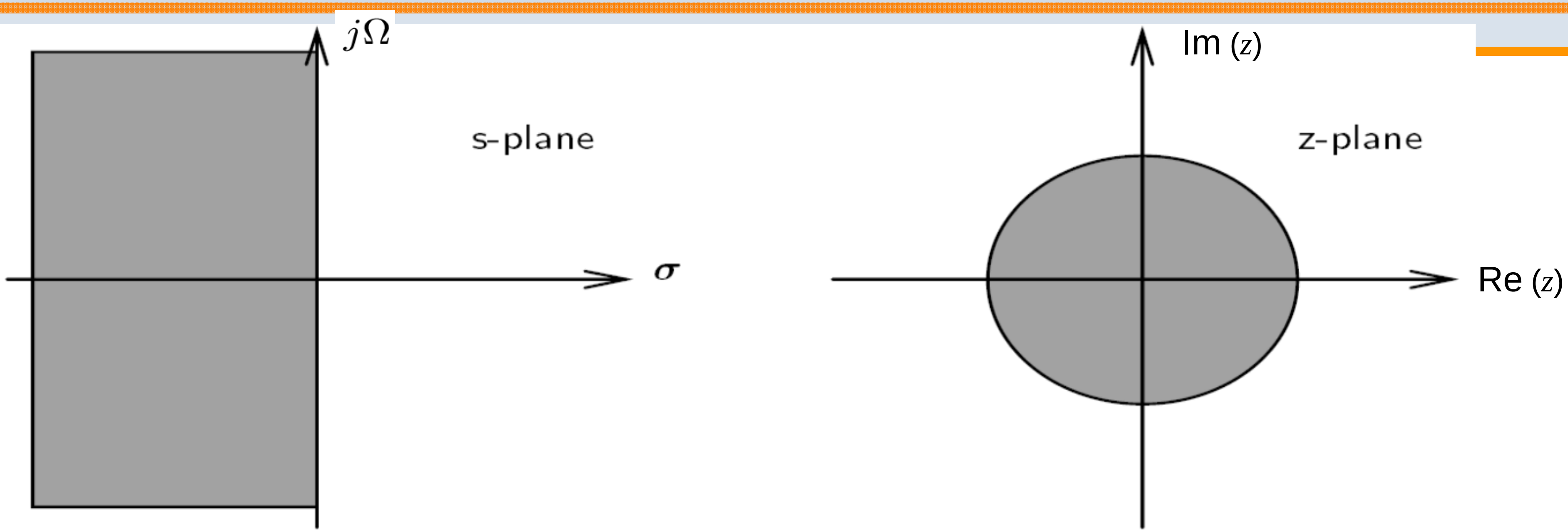
$$H_a(s) = \int_{-\infty}^{\infty} h_a(t)\, e^{-st} dt.$$

In the time domain, it means that the input $x(t)$ and the output $y(t)$ are related by

$$\sum_{k=0}^{M} \alpha_k \frac{d^k y(t)}{dt^k} = \sum_{k=0}^{M} \beta_k \frac{d^k x(t)}{dt^k}.$$

Problem: how to convert sensibly analogue filters into digital ones.

Analogue to Digital Conversion



$$H_c(s) \leftrightarrow H(z)$$

- Analogue filters stable if poles on left half of the s-plane / Digital filters stable if poles inside unit circle

➡ Left half of the s-plane should map inside the unit circle in the z-plane.

- The $j\Omega$ axis in the s-plane should map the unit circle in the z-plane; i.e. direct relationship between frequencies variables [Mathematically = *one-to-one mapping* between $(-\infty, \infty)$ and $(-\pi, \pi)$].

Impulse Invariant method

Start with suitable analogue transfer function $h_c(t)$ and discretize it

$$h(n) \triangleq h_c(nT) \quad \text{where } T = 1/F_s \text{ sampling period.}$$

Sampling in time $\Leftrightarrow$ Periodic repetition in frequency

$$H(\omega) = F_s \sum_{k=-\infty}^{\infty} H_c[(\omega - 2\pi k) F_s]$$

where $\omega = 2\pi f$ and $f = F/F_s$ is the normalized frequency

Impulse Invariant method: Steps

1. Compute the Inverse Laplace transform to get impulse response of the analogue filter

2. Sample the impulse response (quickly enough to avoid aliasing problem)

3. Compute z-transform of resulting sequence

Example 1 – Impulse Invariant Method

Consider first order analogue filter

$$H_c(s) = \frac{s}{s+\alpha} = 1 - \frac{\alpha}{s+\alpha}$$

Corresponding impulse response is

$$h_c(t) = \underbrace{\delta(t)} - \alpha e^{-\alpha t} (t) V$$

The presence of delta term prevents sampling of impulse response which thus cannot be defined

Fundamental problem: high-pass and band-stop filters have functions with numerator and denominator polynomials of the same degree and thus cannot be designed using this method

Example 2 – Impulse Invariant Method

Consider an analogue filter

$$H_c(s) = \frac{C}{s - \alpha}$$

Step 1. Impulse response of the analogue filter

$$h_c(t) = Ce^{\alpha t}$$

Step 2. Sample the impulse response

$$h(n) = Ce^{\alpha nT}$$

Step 3. Compute z-transform

$$H(z) = \frac{C}{1 - e^{\alpha T} z^{-1}}$$

The poles are mapped as $\alpha \rightarrow e^{\alpha T}$

Impulse Invariant Method

Indeed, in the general case the poles are mapped as

$$\alpha_k \to e^{\alpha_k T}$$

since any rational transfer function with the numerator degree strictly less than the denominator degree can be decomposed to partial fractions

$$H_c(s) = \sum \frac{C}{s - \alpha_k}$$

and similarly it can be shown

$$H(z) = \sum_k \frac{C}{1 - e^{\alpha_k T} z^{-1}}$$

Impulse Invariant Method: Stability

Since poles are mapped as:

$$\alpha_k \to e^{\alpha_k T}$$

stable analogue filter is transformed into stable digital filter

$$s = \sigma + j\Omega \leftrightarrow z = re^{j\omega}$$

$$\sigma < 0 \Rightarrow \left|e^{\alpha_k T}\right| < 1$$

Summary of the Impulse Invariant Method

- Determine analogue filter $H_c(s)$ satisfying specifications for desired digital filter (not discussed here!).
- If necessary, expand $H_c(s)$ using partial fractions.
- Obtain the z-transform of each partial fraction z
- Obtain $H(z)$ by combining the z-transforms of the partial fractions.

Summary of the Impulse Invariant Method

- Advantage:
  - preserves the order and stability of the analogue filter

- Disadvantages:
  - Not applicable to all filter types (high-pass, band-stop)
  - There is distortion of the shape of frequency response due to aliasing

Matched z-transform method

- **Matched z-transform:** very simple method to convert analog filters into digital filters.

$$H(s) = \frac{\prod_{k=1}^{M}(s - z_k)}{\prod_{k=1}^{N}(s - p_k)} \overset{\text{matched z-transform}}{\rightarrow} H(z) = \frac{\prod_{k=1}^{M}\left(1 - e^{z_k T} z^{-1}\right)}{\prod_{k=1}^{N}\left(1 - e^{p_k T} z^{-1}\right)};$$

i.e. poles and zeros are transformed according to

$$z_k \rightarrow e^{z_k T}, \; p_k \rightarrow e^{p_k T}$$

where T is the sampling period.

- Poles using this method are similar to impulse invariant method.
- Zeros are located at a new position.

⇒ This method suffers from aliasing problems.

Example of Impulse Invariant vs Matched z transform methods

- Consider the following analog filter into a digital IIR filter

$$H(s) = \frac{s+2}{(s+1)(s+3)} = \frac{1/2}{(s+1)} + \frac{1/2}{(s+3)}$$

- **Impulse invariant method**

$$H(z) = \frac{1/2}{1-e^{-T}z^{-1}} + \frac{1/2}{1-e^{-3T}z^{-1}} = \frac{1-\frac{1}{2}\left(e^{-3T}+e^{-T}\right)z^{-1}}{(1-e^{-T}z^{-1})(1-e^{-3T}z^{-1})}.$$

- **Matched z-tranform**

$$H(z) = \frac{\left(1-e^{-2T}z^{-1}\right)}{(1-e^{-T}z^{-1})(1-e^{-3T}z^{-1})}.$$

$\Rightarrow$ Same poles but different zero.

Backward Difference Method

The analogue-domain variable *s* represents differentiation.

We can try to replace s by approximating differentiation operator in the digital domain:

$$\left.\frac{dx(t)}{dt}\right|_{t=nT} = \frac{x(nT) - x((n-1)T)}{T} = \frac{x(n) - x(n-1)}{T}$$

Thus,

$$y(t) = \frac{dx(t)}{dt} \quad \Rightarrow \quad y(n) \approx \frac{x(n) - x(n-1)}{T}$$

$$Y(z) = T^{-1}\left(1 - z^{-1}\right) X(z)$$

Which suggests the *s*-to-*z* transformation:

$$s \leftarrow T^{-1}\left(1 - z^{-1}\right)$$

delay

backward difference operator

Backward Difference Operator

Consider now the second order derivative

$$\begin{aligned}\left.\frac{d^2x(t)}{dt^2}\right|_{t=nT} &= \frac{d}{dt}\left[\left.\frac{dx(t)}{dt}\right|\right]_{t=nT} \\ &= T^{-1}\frac{[x(nT)-x((n-1)T)]-[x((n-1)T)-x((n-2)T)]}{T} \\ &= \frac{x(n)-2x(n-1)+x(n-2)}{T^2}.\end{aligned}$$

This means that

$$s^2 = \frac{1-2z^{-1}+z^{-2}}{T^2} = \frac{\left(1-z^{-1}\right)^2}{T^2}.$$

Similarly, one can easily check by induction that

$$s^k = \frac{\left(1-z^{-1}\right)^k}{T^k}.$$

Backward Difference method - Stability

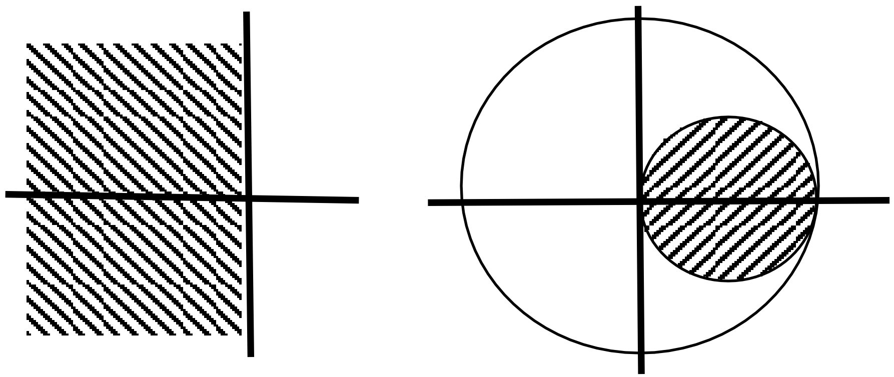
So to convert the analogue filter into a digital one, we simply use

$$H(z) = H_a(s)\big|_{s=(1-z^{-1})/T}\,.$$

$$s = T^{-1}\left(1 - z^{-1}\right) \Rightarrow z = \frac{1}{1 - sT}$$

For $s = j\Omega$, we obtain

$$z = \frac{1}{1 - j\Omega T} \Rightarrow z - 0.5 = 0.5\,\frac{(1 + j\Omega T)}{(1 - j\Omega T)} \Rightarrow |z - 0.5| = 0.5$$



The Left half s-plane onto the interior of the circle with radius 0.5 and centre at 0.5 in the z-plane

➡ **Stable** analogue filters become **stable** digital filters.
However, poles are conned to a relatively small set of frequencies, no highpass filter possible!

Summary of the Backward Difference method

- Since the imaginary axis in the s domain are not mapped to the unit circle we can expect that the frequency response will be considerably distorted

- An analogue high-pass filter cannot be mapped to a digital high-pass because the poles of the digital filter cannot lie in the correct region

⇨ method is crude and rarely used

Bilinear transform

- Bilinear transform is a correction of the backwards difference method

- The bilinear transform (also known as Tustin's transformation) is defined as the substitution:

$$s = \psi(z) = \frac{1 - z^{-1}}{1 + z^{-1}}$$

- It is the most popular method

- The bilinear transform produces a digital filter whose **frequency response** has the same characteristics as the frequency response of the analogue filter (but its impulse response may then be quite different).

The bilinear transform

The bilinear transform

$$s = \psi(z) = \frac{1 - z^{-1}}{1 + z^{-1}}$$

- Note 1 : Although the ratio could have been written (z-1)/(z+1), that causes unnecessary algebra later, when converting the resulting transfer function into a digital filter;

- Note 2 : In some sources you will see the factor (2/T) multiplying the RHS of the bilinear transform; this is an optional scaling, but it cancels and does not affect the final result.

Where is the Bilinear Transform coming from?

- Consider the following analogue system

$$H(s) = \frac{b}{s+a} \Leftrightarrow \frac{dy(t)}{dt} = -ay(t) + bx(t)\,. \tag{1}$$

- Approximate the derivative by a trapezoidal approximation; i.e.

$$\begin{aligned} y(nT) &= \int_{(n-1)T}^{nT} \frac{dy(u)}{dt} du + y((n-1)T) \\ &\simeq \frac{T}{2}\left[\frac{dy(nT)}{dt} + \frac{dy(nT-T)}{dt}\right] + y(nT-T) \end{aligned} \tag{2}$$

- Now plugging (1) in (2) then

$$\left(1+\frac{aT}{2}\right) y(n) - \left(1-\frac{aT}{2}\right) y(n-1) = \frac{bT}{2}(x(n)+x(n-1))$$

where $x(k) \triangleq x(kT) / y(k) \triangleq y(kT)$. Applying the z-transform, one obtains

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b}{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) + a} \Rightarrow \quad s \leftarrow \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right).$$

Properties of the Bilinear Transform

To derive the properties of the bilinear transform, solve for z, and put $s = \sigma + j\Omega$

$$s = \frac{1 - z^{-1}}{1 + z^{-1}}$$

$$z = \frac{1+s}{1-s} = \frac{1+\sigma + j\Omega}{1-\sigma - j\Omega}; \text{ hence } |z|^2 = \frac{(1+\sigma)^2 + \Omega^2}{(1-\sigma)^2 + \Omega^2}$$

Properties of the Bilinear Transform

Look at two important cases:

1. The imaginary axis, i.e. σ=0. This corresponds to the boundary of stability for the analogue filter's poles.

   With σ=0, we have

$$|z|^2 = \frac{(1+0)^2 + \Omega^2}{(1-0)^2 + \Omega^2} = 1$$

the imaginary (frequency) axis in the s-plane maps to the unit circle in the z-plane

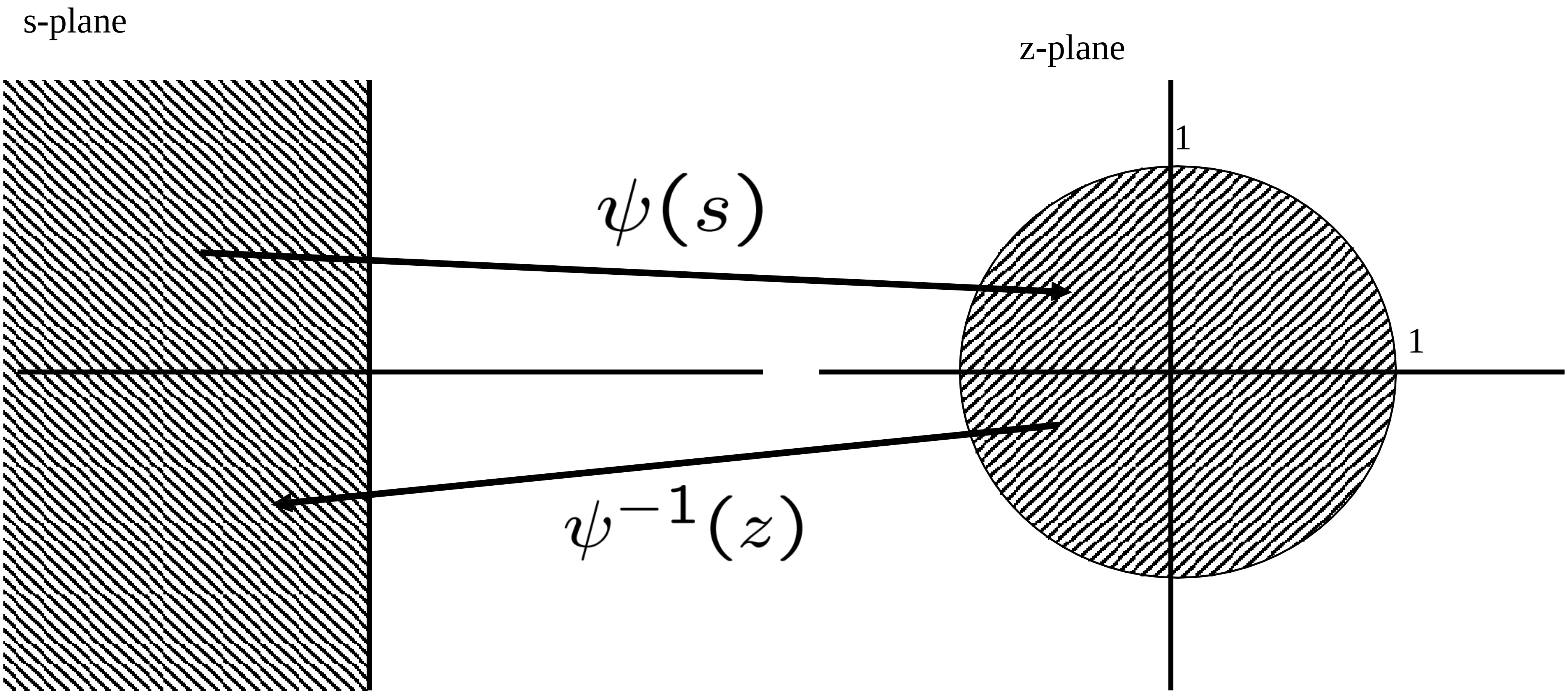
2. With σ<0, i.e. the left half-plane in the s-plane we have

$$|z|^2 = \frac{(1+\sigma)^2 + \Omega^2}{(1-\sigma)^2 + \Omega^2} =< 1, \quad (\sigma < 0)$$

left half s-plane maps onto the interior of the unit circle

Properties of the Bilinear Transform

Thus the bilinear transform maps the Left half s-plane onto the interior of the unit circle in the z-plane:



This property allows us to obtain a suitable frequency response for the digital filter, and also to ensure the stability of the digital filter.

Properties of the Bilinear Transform

If $s = \sigma + j\Omega$ and $z = re^{j\omega}$, then one can easily establish that

$$s = \frac{z - 1}{z + 1}$$

$$= \frac{re^{j\omega} - 1}{re^{j\omega} + 1}$$

$$= \left(\frac{r^2 - 1}{1 + r^2 + 2r\cos\omega} + j\,\frac{2r\sin\omega}{1 + r^2 + 2r\cos\omega} \right)$$

Clearly if $r = 1$ then $\sigma = 0$ (unit circle maps onto imaginary axis)

$$\Omega = \frac{\sin\omega}{1 + \cos\omega} = \tan\left(\frac{\omega}{2}\right) \Leftrightarrow \omega = 2\arctan(\Omega)$$

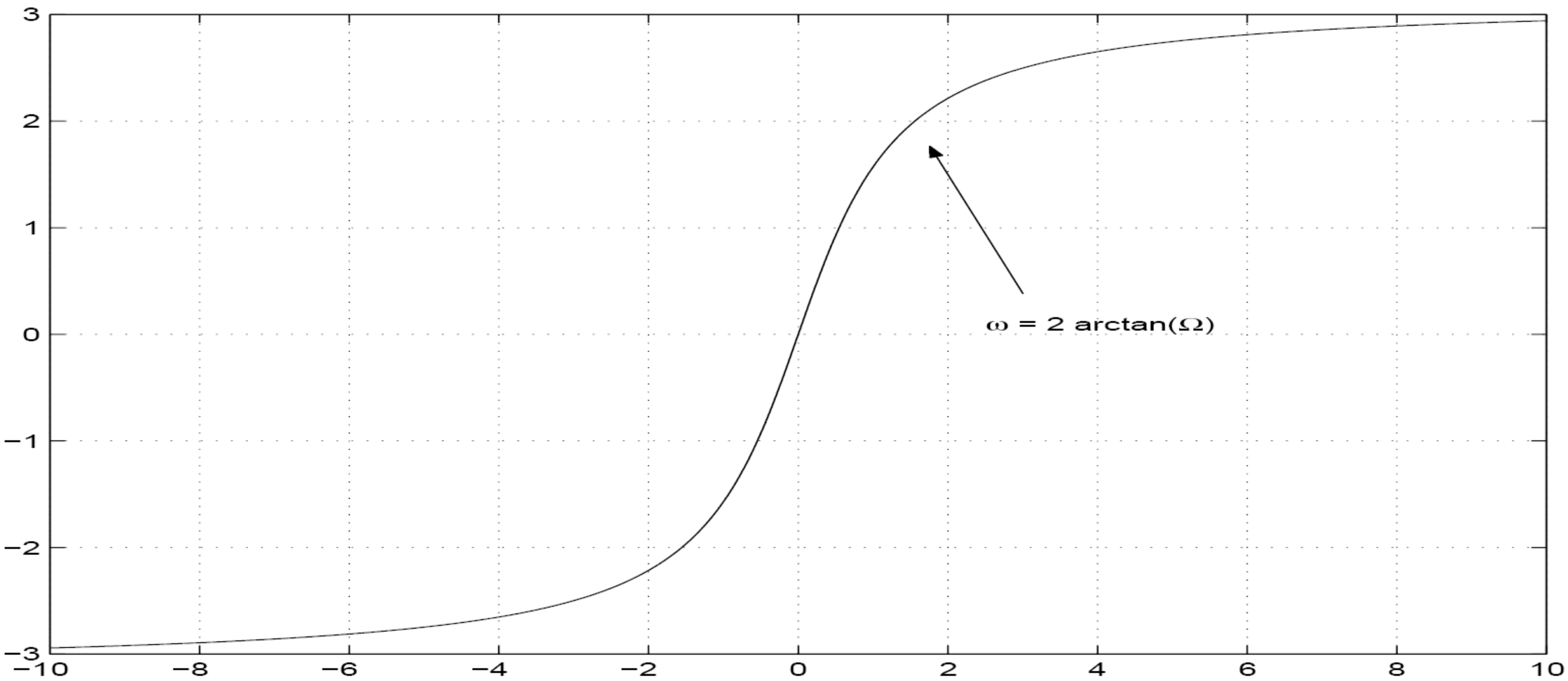
Properties of the Bilinear Transform

Hence the Bilinear Transform preserves the following important features of the frequency response :

1. the $\Omega \leftrightarrow \omega$ mapping is <u>monotonic</u>, and
2. $\Omega = 0$ is mapped to $\omega = 0$, and $\Omega = \infty$ is mapped to $\omega = \pi$ (half the sampling frequency). Thus, for example, a low-pass response that decays to zero at $\Omega = \infty$ produces a low-pass digital filter response that decays to zero at $\omega = \pi$.
3. Mapping between the frequency variables is

$$\Omega = \tan\left(\frac{\omega}{2}\right) \Leftrightarrow \omega = 2\arctan(\Omega)$$

Properties of the Bilinear Transform



If the frequency response of the analogue filter at frequency Ω is $H(j\Omega)$, then the frequency response of the digital filter at the corresponding frequency ω = 2 arctan(ω) is also H(j Ω). Hence -3dB frequencies become -3dB frequencies, minimax responses remain minimax, etc.

Proof of Stability of the Filter

$$s = \psi(z) = \frac{1 - z^{-1}}{1 + z^{-1}}; \quad z = \psi^{-1}(s) = \frac{1 + s}{1 - s}$$

Suppose the analogue prototype $H(s)$ has a stable pole at $\sigma + j\Omega$, i.e.

$$H(\sigma + j\Omega) \rightarrow \infty, \quad a < 0$$

Then the digital filter $\hat{H}(z)$ is obtained by substituting $s = \psi(z)$

$$\hat{H}(z) = H(\psi(z))$$

Since $H(s)$ has a pole at $\sigma + j\Omega$, $H(\psi(z))$ has a pole at $\psi^{-1}(\sigma + j\Omega)$ because

$$\hat{H}(\psi^{-1}(\sigma + j\Omega)) = H(\psi(\psi^{-1}(\sigma + j\Omega))) = H(\sigma + j\Omega) \rightarrow \infty$$

However, we know that $\psi^{-1}(\sigma + j\Omega)$ lies within the unit circle. Hence the filter is *guaranteed stable* provided $H(s)$ is stable.

Frequency Response of the Filter

The frequency response of the analogue filter is

$$H(j\omega)$$

$$s = \psi(z) = \frac{1 - z^{-1}}{1 + z^{-1}}; \quad z = \psi^{-1}(s) = \frac{1 + s}{1 - s}$$

The frequency response of the digital filter is

$$\begin{aligned}\hat{H}(\exp(j\Omega)) &= H(\psi(\exp(j\Omega))) \\ &= H(j\tan(\Omega/2))\end{aligned}$$

$$\begin{aligned}\psi(\exp(j\Omega)) &= \frac{1 - \exp(-j\Omega)}{1 + \exp(-j\Omega)} \\ &= \frac{\exp(-j\Omega/2)(\exp(j\Omega/2) - \exp(-j\Omega/2))}{\exp(-j\Omega/2)(\exp(+j\Omega/2) + \exp(-j\Omega/2))} \\ &= \frac{j\sin(\Omega/2)}{\cos(\Omega/2)} \\ &= j\tan(\Omega/2)\end{aligned}$$

Hence we can see that the frequency response is *warped* by a function

$$\omega = \tan(\Omega/2)$$

Analogue Frequency

Digital Frequency

Design using the bilinear transform

The steps of the bilinear transform method are as follows:

1. "Warp" the digital critical (e.g. band-edge or "corner") frequencies ω_i , in other words compute the corresponding analogue critical frequencies $\Omega_i = \tan(\omega_i/2)$.

2. Design an analogue filter which satisfies the resulting filter response specification.

3. Apply the bilinear transform

$$s = \frac{1-z^{-1}}{1+z^{-1}}$$

to the s-domain transfer function of the analogue filter to generate the required z-domain transfer function.

Example: Application of Bilinear Transform

Design a first order low-pass digital filter with -3dB frequency of 1kHz and a sampling frequency of 8kHz using a the first order analogue low-pass filter

$$H(s) = \frac{1}{1 + s/\Omega_c}$$

which has a gain of 1 (0dB) at zero frequency, and a gain of -3dB (= √0.5) at Ω_c rad/sec (the "cutoff frequency ").

Example: Application of Bilinear Transform

- First calculate the normalized digital cutoff frequency:

$$\omega_c = \frac{1kHz}{8kHz} 2\pi = \pi/4$$

3dB cutoff frequency

sampling frequency

- Calculate the equivalent pre-warped analogue filter cutoff frequency (rad/sec)

$$\Omega_c = \tan(\omega_c/2) = \tan(\pi/8) = 0.4142$$

- Thus, the analogue filter has the system function

$$H(s) = \frac{1}{1 + s/\Omega_c}$$

$$= \frac{1}{1 + s/0.4142} = \frac{0.4142}{s + 0.4142}$$

Example: Application of Bilinear Transform

Apply Bilinear transform

$$H(s) = \left. \frac{0.4142}{s + 0.4142} \right|_{s = \frac{1 - z^{-1}}{1 + z^{-1}}}$$

$$\Rightarrow \quad H(z) = \frac{0.2929\left(1 + z^{-1}\right)}{1 - 0.4142 z^{-1}}$$

Normalise to unity for recursive implementation

As a direct form implementation:

Keep 0.2929 factorised to save one multiply

$$y_n = 0.4142 y_{n-1} + 0.2929(x_n + x_{n-1})$$

Designing high-pass, band-pass and band-stop filters

- The previous examples we have discussed have concentrated on IIR filters with low-pass characteristics.

- There are various techniques available to transform a low-pass filter into a high-pass/band-pass/band-stop filters.

- The most popular one uses a frequency transformation in the analogue domain.