http://elec3004.com

Digital Filters
IIR & FIR

ELEC 3004: **Systems**: Signals & Controls
Dr. Surya Singh

Lecture 11

elec3004@itee.uq.edu.au

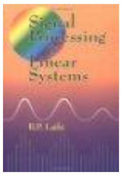
http://robotics.itee.uq.edu.au/~elec3004/

April 3, 2019

Lecture Schedule:

Week	Date	Lecture Title
1	27-Feb	Introduction
	1-Mar	Systems Overview
2	6-Mar	Systems as Maps & Signals as Vectors
	8-Mar	Systems: Linear Differential Systems
3	13-Mar	Sampling Theory & Data Acquisition
	15-Mar	Aliasing & Antialiasing
4	20-Mar	Discrete Time Analysis & Z-Transform
	22-Mar	Second Order LTID (& Convolution Review)
5	27-Mar	Frequency Response
	29-Mar	Filter Analysis
6	**3-Apr**	**Digital Filters (IIR) & Filter Analysis**
	5-Apr	PS 1: Q & A
7	10-Apr	Digital Filter (FIR) & Digital Windows
	12-Apr	FFT
8	17-Apr	Active Filters & Estimation & Holiday
	19-Apr	**Holiday**
	24-Apr	
	26-Apr	
9	1-May	Introduction to Feedback Control
	3-May	Servoregulation/PID
10	8-May	PID & State-Space
	10-May	State-Space Control
11	15-May	Digital Control Design
	17-May	Stability
12	22-May	State Space Control System Design
	24-May	Shaping the Dynamic Response
13	29-May	System Identification & Information Theory
	31-May	Summary and Course Review

Follow Along Reading:



B. P. Lathi
Signal processing and linear systems
1998
TK5102.9.L38 1998

Today

- Chapter 10
  (**Discrete-Time System Analysis Using the *z*-Transform**)
  - § 10.3 Properties of DTFT
  - § 10.5 Discrete-Time Linear System analysis by DTFT
  - § 10.7 Generalization of DTFT to the $\mathcal{Z}$–Transform

- Chapter 12
  (**Frequency Response and Digital Filters**)
- § 12.1 Frequency Response of Discrete-Time Systems
- § 12.3 Digital Filters
- § 12.4 Filter Design Criteria
- § 12.7 Nonrecursive Filters

Next Time

ELEC3004 is

$$-e^{\pi i}$$

Periodic Signals:
Writing them in the Fourier Domain & z-Domain

- Synthesis:

The function $X(e^{j\Omega})$ defined by

$$X(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega n} \qquad (7.1.1)$$

(if it converges) is called the *discrete-time Fourier transform* (*DTFT*) of the signal $x[n]$. In particular, if the region of convergence for the z transform

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

includes the unit circle, then the DTFT equals $X(z)$ evaluated on the unit circle, that is,

$$X(e^{j\Omega}) = X(z)\big|_{z=e^{j\Omega}}. \qquad (7.1.2)$$

Euler's Identity: $sin([\omega t]) = \frac{1}{2i}\left(e^{i\cdot[\omega t]} - e^{-i\cdot[\omega t]}\right)$

- **The Discrete-Time Fourier Transform** of a sinusoid

$$x[n] = \sin \Omega_0 n = \frac{1}{2j}(e^{j\Omega_0 n} - e^{-j\Omega_0 n})$$

is simply

$$X(e^{j\Omega}) = 2\pi\left(\frac{1}{2j}\right)[\delta(\Omega - \Omega_0) - \delta(\Omega + \Omega_0)]$$

$$= -j\pi[\delta(\Omega - \Omega_0) - \delta(\Omega + \Omega_0)]$$

for $|\Omega|, |\Omega_0| \leq \pi$, while that of the cosine signal

$$y[n] = \cos \Omega_0 n = \tfrac{1}{2}(e^{j\Omega_0 n} + e^{-j\Omega_0 n})$$

is likewise

$$Y(e^{j\Omega}) = 2\pi(\tfrac{1}{2})[\delta(\Omega - \Omega_0) + \delta(\Omega + \Omega_0)]$$

$$= \pi[\delta(\Omega - \Omega_0) + \delta(\Omega + \Omega_0)].$$

In addition, the DTFT pair for the dc signal $x[n] = 1$ is simply

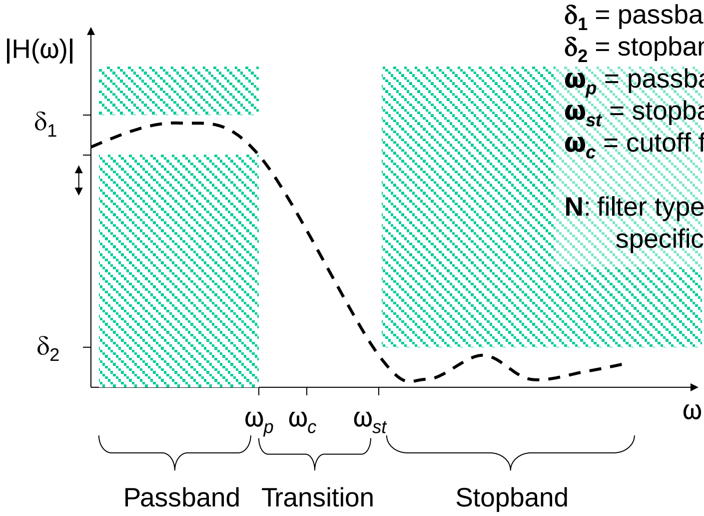
$$1 \leftrightarrow 2\pi\,\delta(\Omega), \qquad |\Omega| \leq \pi,$$

as opposed to the dual relationship

$$\delta[n] \leftrightarrow 1, \qquad \text{all } \Omega.$$

Recap:
IIR Filters = “Analog Filters” in Digital Form ☺

Filter Specification in the Frequency Domain



Where:

δ_1 = passband ripple (dB)

δ_2 = stopband attenuation (dB)

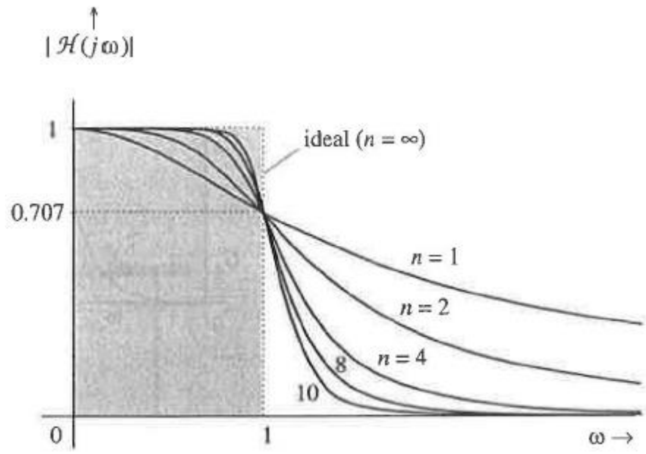
$\boldsymbol{\omega_p}$ = passband edge (Hz)

$\boldsymbol{\omega_{st}}$ = stopband edge (Hz)

$\boldsymbol{\omega_c}$ = cutoff frequency (@ 3dB)

N: filter type/order to meet specification

Butterworth Filters



Butterworth Filters

- Butterworth: Smooth in the pass-band
- The amplitude response $|H(j\omega)|$ of an n^{th} order Butterworth low pass filter is given by:

$$|H(j\omega)| = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}}}$$

- The normalized case (ω_c=1)

$$|\mathcal{H}(j\omega)| = \frac{1}{\sqrt{1+\omega^{2n}}} \Rightarrow \mathcal{H}(j\omega)\mathcal{H}(-j\omega) = |\mathcal{H}(j\omega)|^2 = \boxed{\frac{1}{1+\omega^{2n}}}$$

Recall that: $|H(j\omega)|^2 = H(j\omega)\,H(-j\omega)$

Analog Filter Summary ☺

Filter Type	Passband Ripple	Stopband Ripple	Transition Band	MATLAB Design Command
Butterworth	No	No	Loose	`butter`
Chebyshev	Yes	No	Tight	`cheby`
Chebyshev Type II (Inverse Chebyshev)	No	Yes	Tight	`cheby2`
Eliptic	Yes	Yes	Tightest	`ellip`

IIR Filter Design Methods

IIR Filter Design Methods

- Normally based on analogue prototypes
  - **Butterworth**, Chebyshev, Chebyshev 2, Elliptic, etc.

- Then transform $H(s) \rightarrow H(z)$

- Three popular methods:

1. Impulse invariant
   - $H(z)$: whose impulse response is a sampled version of $h(t)$ (also step invariant)
2. Matched ***z***–Transform
   - poles/zeros $H(s)$ directly mapped to poles/zeros $H(z)$
3. **Bilinear z – transform**
   - left hand s – plane mapped to unit circle in z – plane

Impulse Invariant

Simplest approach, proceeds as follows:

1. Select prototype analogue filter
2. Determine H(s) for desired $\underbrace{\omega_c}_{cutoff\ freq.}$ and $\underbrace{\omega_s}_{stop\ freq.}$
3. Inverse Laplace
   - Calculate impulse response, $h(t)$
4. Sample impulse response $h(t)|_{t=n\Delta t_d}$
   - $h[n] = \Delta t_d\ h(n\Delta t_d)$
5. Take ***z***-Transform of $h[n] \Rightarrow H(z)$
   - Poles: p_i maps to $\exp(p_i \Delta t_d)$
   - Zeros: have no simple mapping ☹

Impulse Invariant [2]

Useful approach when:

- Impulse (or step) invariance is required
  - e.g., control applications
- Designing Lowpass or Bandpass filters

Has problems when:

- $H(\omega)$ does not $\rightarrow 0$ as $\omega \rightarrow \infty$
  - Ex: highpass or bandstop filters
- If H(w) is not bandlimited, aliasing occurs!

Matched z - transform

- Maps poles/zeros in s – plane directly
  – to poles/zeros in z – plane
- No great virtues/problems
- Fairly old method
  – not commonly used
  – so we won't consider it further ☺

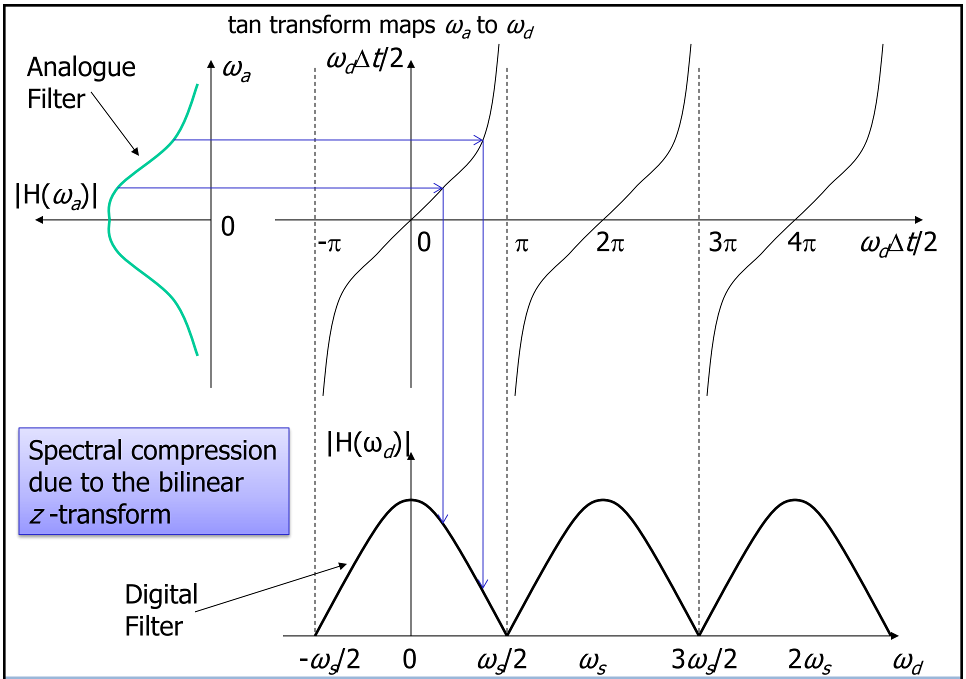
Bilinear z - transform

- Maps complete imaginary s –plane (±∞)
  - to unit circle in z -plane
- That is: map analogue frequency ω_a to discrete frequency ω_d
- Uses continuous transform:

$$\omega_a = \frac{2}{\Delta t}\tan\left(\frac{\omega_d \Delta t}{2}\right)$$

This compresses (warps) ω_a to have finite extent $\pm\frac{\omega_s}{2}$

➔ this removes possibility of any aliasing ☺



Note, $H(\omega_d)$ periodic, due to sampling

Bilinear Transform

The bilinear transform	$\omega_a = \frac{2}{\Delta t}\tan(\frac{\omega_d \Delta t}{2})$
Transforming to s-domain Remember: $s = j\omega_a$ and $\tan\theta = \sin\theta/\cos\theta$ Where $\theta = \omega_d\Delta t/2$	$s = \frac{2}{\Delta t}\frac{j\sin(\frac{\omega_d\Delta t}{2})}{\cos(\frac{\omega_d\Delta t}{2})}$
Using Euler's relation This becomes… (note: j terms cancel)	$s = \frac{2}{\Delta t}\frac{½(\exp(\frac{j\omega_d\Delta t}{2}) - \exp(\frac{-j\omega_d\Delta t}{2}))}{½(\exp(\frac{j\omega_d\Delta t}{2}) + \exp(\frac{-j\omega_d\Delta t}{2}))}$
Multiply by $\exp(-j\theta)/\exp(-j\theta)$	$s = \frac{2}{\Delta t}\frac{(1-\exp(-j\omega_d\Delta t))}{(1+\exp(-j\omega_d\Delta t))}$
As $z = \exp(s_d\Delta t) = \exp(j\omega_d\Delta t)$	$s = \frac{2(1-z^{-1})}{\Delta t(1+z^{-1})}$

Bilinear Transform

- Convert $H(s) \Rightarrow H(z)$ by substituting,

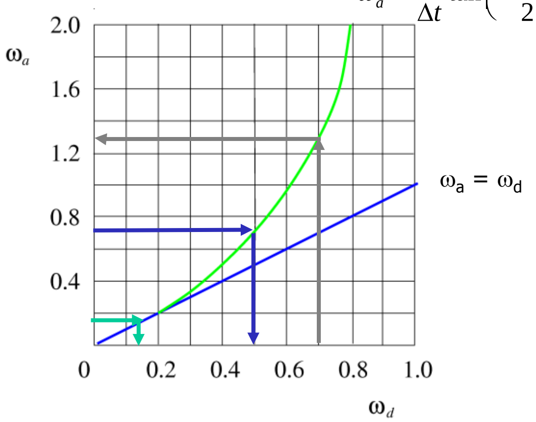
$$s = \frac{2(1-z^{-1})}{\Delta t(1+z^{-1})}$$

Note: this comes directly from **tan** transform

- However, this transformation compresses the analogue frequency response, which means
  - digital cut off frequency will be lower than the analogue prototype
- Therefore, analogue filter must be "pre-warped" prior to transforming $H(s) \Rightarrow H(z)$

Bilinear Pre-warping

$$\omega_a = \frac{2}{\Delta t}\tan\left(\frac{\omega_d \Delta t}{2}\right)$$



Bilinear Transform: Example

- Design digital Butterworth lowpass filter
  - order, n = 2, cut off frequency ω_d = 628 rad/s
  - sampling frequency ω_s = 5024 rad/s (800Hz)

- Pre-warp to find ω_a that gives desired ω_d

$$\omega_a = \left(\frac{2}{1/800}\right)\tan\left(\frac{628}{2\times 800}\right) = 663 \text{ rad/s}$$

Note: $\omega_d < \omega_a$ due to compression

- Butterworth prototype (unity cut off) is,

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

Bilinear Transform: Example [2]

- De-normalised analogue prototype ($s' = \frac{s}{\omega_c}$)
  - $\omega_c = 663$ rad/s (required ω_a to give desired ω_d)

$$H(s_d) = \frac{1}{\left(\frac{s}{663}\right)^2 + \frac{\sqrt{2}s}{663} + 1}$$

  - Convert H(s) ⇒ H(z) by substituting $s = \frac{2(1-z^{-1})}{\Delta t(1+z^{-1})}$

$$H(z) = \frac{1}{\left(\frac{2\times 800(1-z^{-1})}{663(1+z^{-1})}\right)^2 + \sqrt{2}\left(\frac{2\times 800(1-z^{-1})}{663(1+z^{-1})}\right) + 1}$$

$$H(z) = \frac{0.098z^2 + 0.195z + 0.098}{z^2 - 0.942z + 0.333}$$

Note: H(z) has both poles and zeros
H(s) was all-pole

Bilinear Transform: Example [3]

$$H(z) = \frac{Y(z)}{X(z)} = \frac{0.098z^2 + 0.195z + 0.098}{z^2 - 0.942z + 0.333}$$

- Multiply out and make causal:

$$Y(z)(z^2 - 0.942z + 0.333) = X(z)(0.098z^2 + 0.195z + 0.098)$$

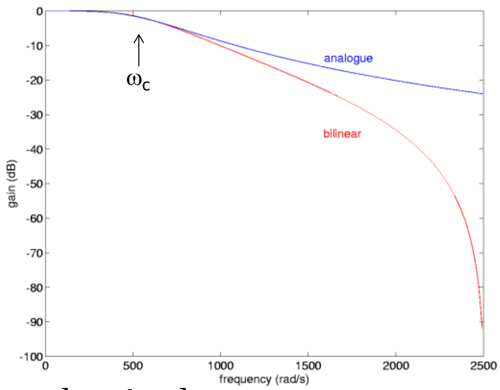
$$Y(z)(1 - 0.942z^{-1} + 0.333z^{-2}) = X(z)(0.098 + 0.195z^{-1} + 0.098z^{-2})$$

- Finally, apply inverse z-transform to yield the difference equation:

$$y[n] = 0.098x[n] + 0.195x[n-1] + 0.098x[n-2] + 0.942y[n-1] - 0.333y[n-2]$$

Bilinear Transform: Example [4]

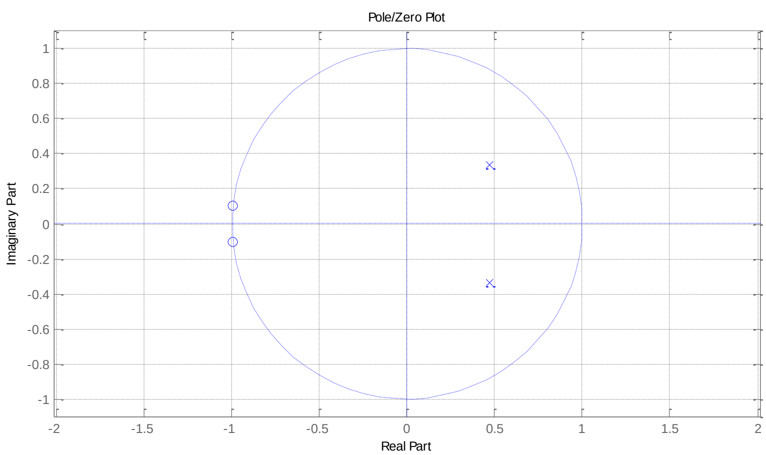
Magnitude response



Digital compared to Analog:

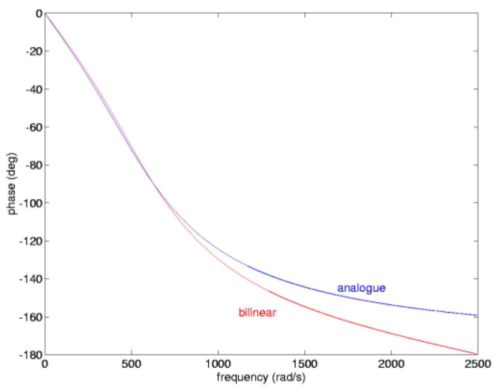
1. Increased roll off and attenuation in stopband
2. Nearly ∞ attenuation at $\frac{\omega_s}{2}$

Bilinear Transform: Example [5]



Bilinear Transform: Example [6]

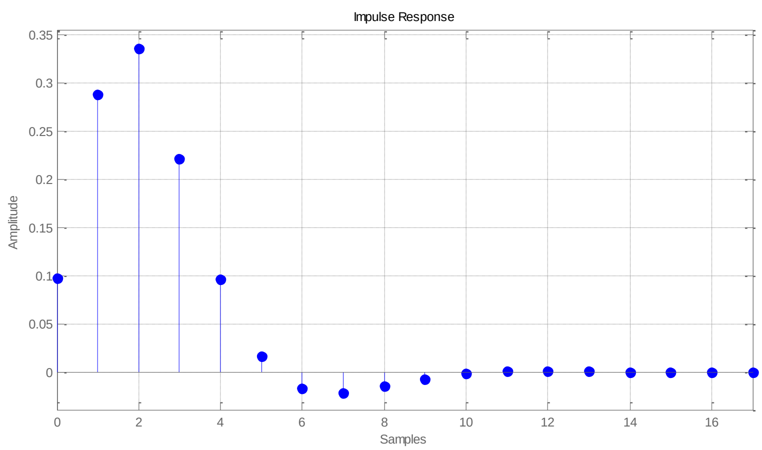
Phase response



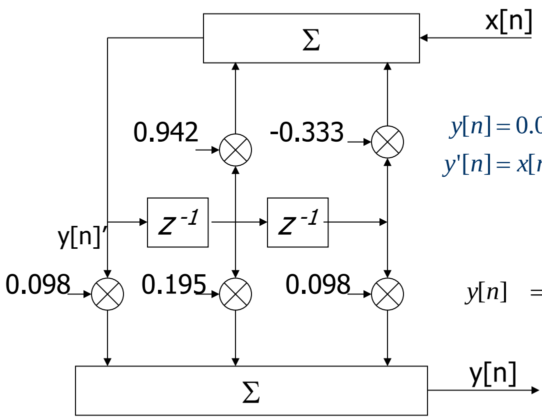
Increased phase delay

Bilinear transform has effectively increased digital filter order (by adding zeros)

Bilinear Transform: Example [7]



Bilinear Transform: Example [8]



Canonical Implementation

$$y[n] = 0.098y'[n] + 0.195y'[n-1] + 0.098y'[n-2]$$
$$y'[n] = x[n] + 0.942y'[n-1] - 0.333y'[n-2]$$

of the difference equation

$$y[n] = 0.098x[n] + 0.195x[n-1] + 0.098x[n-2] + 0.942y[n-1] - 0.333y[n-2]$$

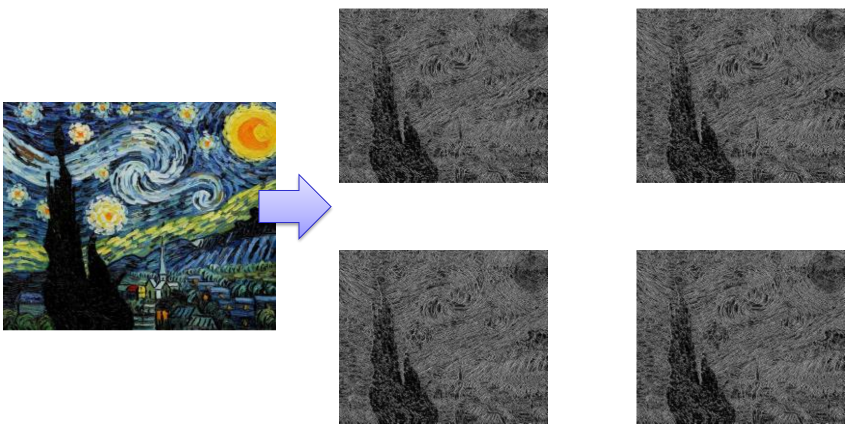
Bilinear Design Summary

- Calculate pre-warping analogue cutoff frequency
- De-normalise filter transfer function using pre-warping cut-off
- Apply bilinear transform and simplify
- Use inverse z-transform to obtain difference equation

BREAK

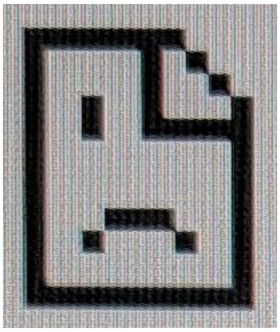
Tutorial 3 (Week 6 & 7!)

- Week 6: PS 1 Review
- Week 7: FIR Filters
- DCT, FFT & More!

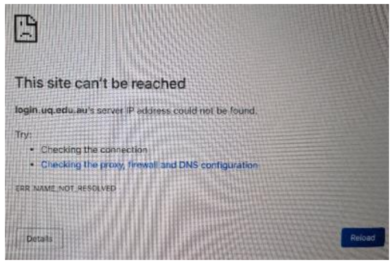


Fun Demo! Resize a "Screen Shot"

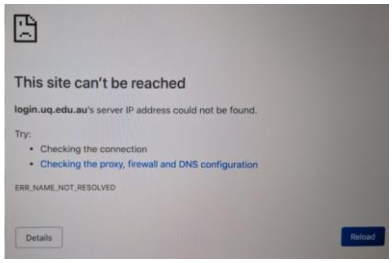
- **<u>10:1 Reduction</u>** of Picture of an LCD Screen…



Without Gaussian Filter First



WITH Gaussian Filter (σ=2)



- **Moire** ∵ of Aliasing of a Non Band-limited Signal

Direct Synthesis (in the Z-Domain)

Direct Synthesis

- Not based on analogue prototype
  - **But direct placement of poles/zeros**

- Useful for
  - First order lowpass or highpass
    - simple smoothers
  - **Resonators** and equalisers
    - Single frequency **amplification**/removal
  - Comb and notch filters
    - Multiple frequency amplification/removal

First Order Filter: Example

- General first order transfer function
  - Gain, G, zero at –b, pole at a (a, b both < 1)

Remember: $H(\omega) = H(z)|_{z = \exp(jw\Delta t)}$

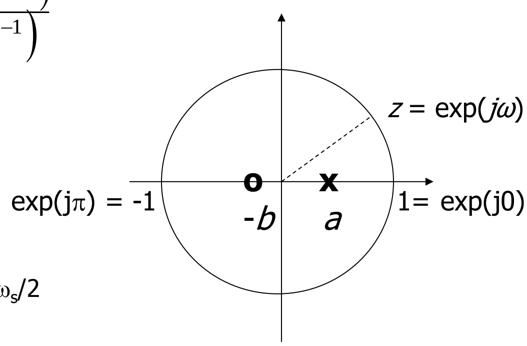
$$H(z) = \frac{G(1 + bz^{-1})}{(1 - az^{-1})}$$

with a +ve & b –ve this is a lowpass filter i.e.,

$$H(0) = \frac{G(1+b)}{(1-a)}$$

$$H(\pi) = \frac{G(1-b)}{(1+a)}$$

$\omega_s/2$



First Order Filter: Example

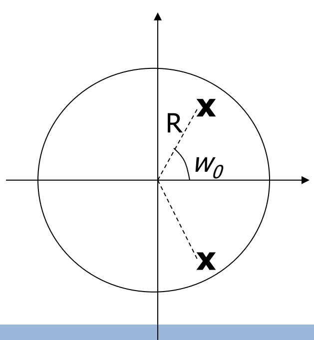
- Possible design criteria
  - cut-off frequency, ω_c
    - $3dB = 20\log(|H(\omega_c)|)$
    - e.g., at $\omega_c = \frac{\pi}{2}$,
    - $\frac{1+b}{1+a} = \sqrt{2}$
  - stopband attenuation
    - assume $\omega_{stop} = \pi$ (Nyquist frequency)
    - e.g., $\delta_2 = \frac{H(\pi)}{H(0)} = \frac{1}{21}$ i.e.,

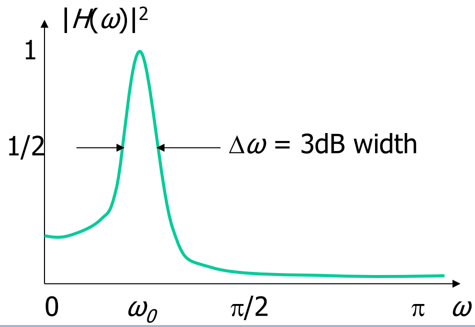
$$\frac{H(\pi)}{H(0)} = \frac{(1-b)(1-a)}{(1+b)(1+a)} = \frac{1}{21}$$

two unknowns (*a,b*) two (simultaneous) design equations.

Digital Resonator Design Prototype

- Second order 'resonator'
  - single narrow peak frequency response
  - i.e., peak at resonant frequency, ω_0





Quality factor (Q-factor)

- Dimensionless parameter that compares
  - Time constant for oscillator decay/bandwidth (Δω) to
  - Oscillation (resonant) period/frequency (ω0)
    - High Q = less energy dissipated per cycle

$$Q = \frac{\omega_0}{\Delta\omega} = \frac{f_0}{\Delta f}$$

- Alternative to damping factor (ζ) as

$$Q = \frac{1}{2\zeta} \qquad H(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} = \frac{\omega_0^2}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2}$$

- Note: Q < ½ overdamped (not an oscillator)

Digital Resonator Design

- To make a peak at ω_0 place pole
  - Inside unit circle (for stability)
  - At angle ω_0 distance R from origin
    - i.e., at location $p = R\exp(j\omega_0)$
      - R controls Δω
        - » Closer to unit circle → sharper peak
    - plus complex conj pole at p* = R exp(-jw0)

$$\begin{aligned} H(z) &= \frac{1}{\left(1 - R\cdot\exp(j\omega_0)z^{-1}\right)\left(1 - R\cdot\exp(-j\omega_0)z^{-1}\right)} \\ &= \frac{1}{1 - R\left(\exp\left(j\omega_0\right) + \exp\left(-j\omega_0\right)\right)z^{-1} + R^2 z^{-2}} \\ &= \frac{G}{1 + a_1 z^{-1} + a_2 z^{-2}} \end{aligned}$$

Where (via Euler's relation)

$$a_1 = -2R\cos(\omega_0) \ \text{ and } \ a_2 = R^2$$

Digital Resonator Design

- Frequency response $H(w) = H(z)|_{z = \exp(jw)}$

$$H(\omega) = \frac{G}{\left(1 - R\cdot\exp(j\omega_0)\exp(-j\omega)\right)\left(1 - R\cdot\exp(-j\omega_0)\exp(-j\omega)\right)}$$

$$= \frac{G}{1 + a_1\exp(-j\omega) + a_2\exp(-2j\omega)}$$

Note: we know the form of the 2nd equation from the previous slide
And a_1 and a_2 remain the same

Digital Resonator Design

- Fixing unity gain at w_0 (pole frequency)
  - i.e., $|H(w_0)| = 1$

$$|H(w_0)| = 1 = \frac{G}{\left|\left(1 - R\cdot\exp(j\omega_0)\exp(-j\omega_0)\right)\left(1 - R\cdot\exp(-j\omega_0)\exp(-j\omega_0)\right)\right|}$$

solving for G,

$$G = (1-R)\sqrt{1 - 2R\cos(2\omega_0) + R^2}$$

Design relationship between gain G and pole radius R (at resonant frequency w_0)

Digital Resonator Design

- Magnitude squared response is given by

$$|H(\omega)|^2 = \frac{G^2}{\left(1-2R\cos(\omega-\omega_0)+R^2\right)\left(1-2R\cos(\omega+\omega_0)+R^2\right)}$$

- The 3dB bandwidth, Δw, occurs when
  - $|H(\omega)|^2 = ½$ (remember G selected for $|H(\omega_0)| = 1$)
  - two points ω_1 and ω_2 (on either side of ω_0)
  - $\Delta\omega = \omega_1 - \omega_2$
- when pole, p, is close to the unit circle (R ≈< 1)

$$\Delta\omega \approx 2(1-R)$$

i.e., closer pole is to unit circle, the sharper the peak

Note: (1 - R) is pole distance to unit circle

Resonator Design "Formula"

1. For specified resonant frequency ω_0
   - and 3 dB bandwidth $\Delta\omega$
2. Calculate pole angle $\theta = \pm 2\pi\omega_0/\omega_s$
   - i.e., $\frac{\theta}{\pi} = \frac{\omega_0}{\omega_s/2}$
3. Calculate pole radius $R = 1 - \frac{\Delta\omega}{2}$
4. Calculate $G = (1-R)\sqrt{1-2R\cos(2w_0)+R^2}$
5. Calculate filter coefficients (a_1, a_2)

$$a_1 = -2R\cos(\omega_0) \text{ and } a_2 = R^2$$

Digital Resonator: Example

- Design a 2-pole resonator with
  - peak, $f_0 = 500Hz$
  - 3dB width, $\Delta f = 32Hz$
  - sampling frequency $f_s = 10kHz$

- Normalise specification
  - $\omega_0 = 2\pi \frac{f_0}{f_s} = 0.1\pi$
  - $\Delta\omega = 2\pi \frac{\Delta f}{f_s} = 0.02$

- Calculate R (from $\Delta\omega \approx 2(1 - R)$)
  - R = 0.99
- Then calculate G and a_1 and a_2
  - $\boldsymbol{G} = 0.0062$, $a_1 = -1.8831$ and $a_2 = 0.9801$

Discrete Filter Transformations

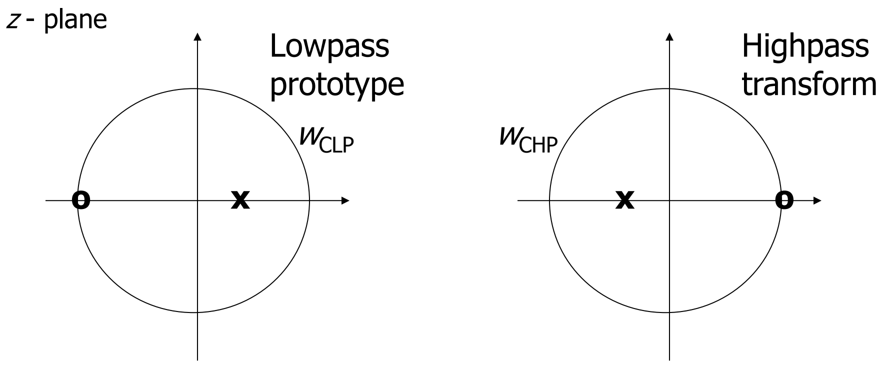
- By convention, design Lowpass filters
  - transform to HighPass, BandPass, BandStop, etc.

- Simplest transformation
  - Lowpass H(z’) → highpass H(z)
  - $H_{HP}(z) = H_{LP}(z)|_{z' \rightarrow -z}$
    - reflection about imaginary axis ($\frac{\omega_s}{4}$)
    - changing signs of poles and zeros

- LP cutoff frequency, ω_{CLP} becomes
    - HP cut-in frequency:

$$\omega_{C_{HP}} = {}^{1}/_{2} - \omega_{C_{LP}}$$

Lowpass → highpass (z' = -z)



$p_L = 1/4$, $z_L = -1$ $\qquad$ $p_H = -1/4$, $z_H = 1$

Poles/zeros reflected in imaginary axis: $W_{CHP} = 1/2 - W_{CLP}$

Same gain @ $w_s/4$ (i.e., $\pi/4$)

$|H(w_{HP})| = |H(\pi/2 - w_{LP})|$

Discrete Filter Transformations

- Lowpass H(z') → highpass H(z)
  - Cut-off (3dB) frequency = wc (remains same)
- Lowpass H(z') → Bandpass H(z)
  - Centre frequency = w0 & 3dB bandwidth = wc

$$z' = \frac{\cos(w_c \Delta t) - z}{1 - \cos(w_c \Delta t) z}$$

$$z' = \frac{\alpha z - z^2}{-\alpha z + 1} \qquad \alpha = \frac{\cos(w_0 \Delta t)}{\cos(w_c \Delta t)}$$

Note: these are not the only possible BP and BS transformations!

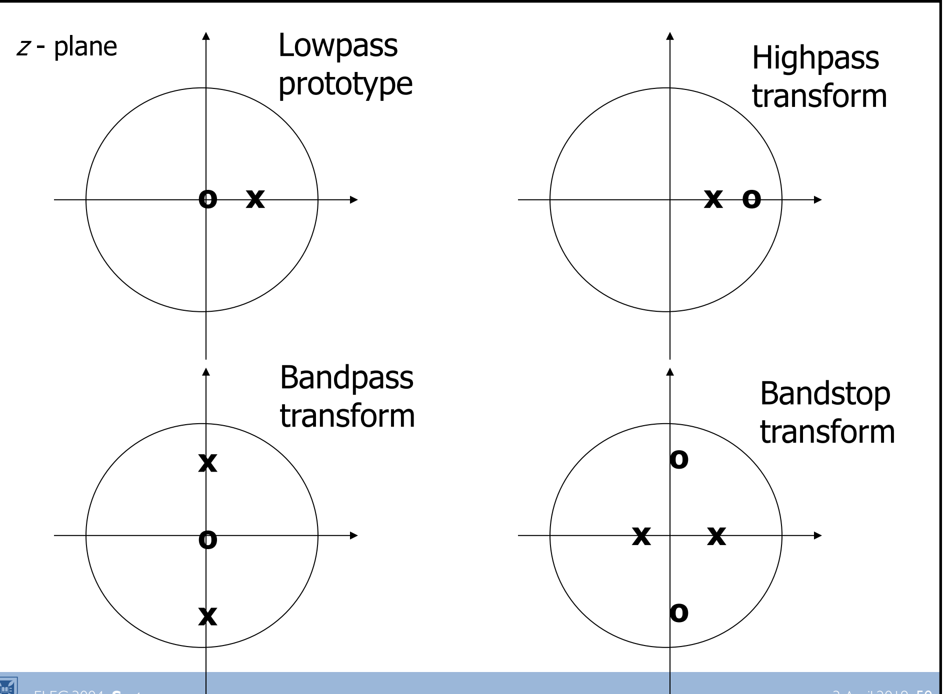
Discrete Filter Transformations

- Lowpass H(z') → Bandstop H(z)
  - Centre frequency = ω_0 3dB bandwidth = ω_c

$$z' = \frac{z^2 - \left(2\alpha/(k+1)\right)z + (1-k)/(1+k)}{1 + \left(2\alpha/(k+1)\right)z + \left((1-k)/(1+k)\right)z^2}$$

$$\alpha = \frac{\cos(\omega_0 \Delta t)}{\cos(\omega_c \Delta t)} \qquad k = \tan^2(\omega_c \Delta t)$$

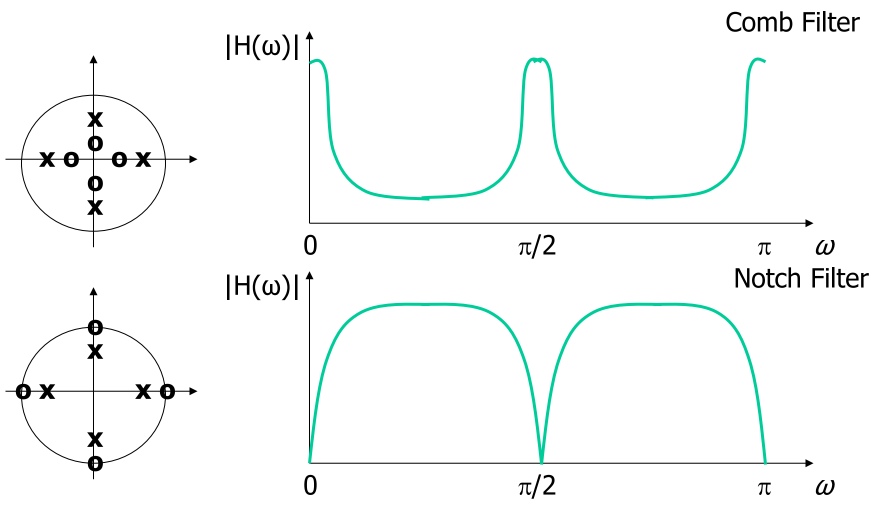
Note: order doubles for bandpass/bandstop transformations



Notch and Comb Filters

- By positioning multiple pole/zero pairs
  - equally spaced around the unit circle
- Design a filter that removes/amplifies
  - frequencies at $n \cdot \omega_0$
  - i.e., frequency harmonics
- Can also remove/amplify multiple arbitrary frequencies
- Notch filter
  - removes multiple/single frequencies
- Comb filter
  - amplifies multiple/single frequencies

Comb and Notch Filters



Summary

- Digital Filter Structures
  - Direct form (simplest)
  - Canonical form (minimum memory)
- IIR filters
  - Feedback and/or feedforward sections
- FIR filters
  - Feedforward only
- Filter design
  - Bilinear transform (LP, HP, BP, BS filters)
  - Direct form (resonators and notch filters)
  - Filter transformations (LP $\rightarrow$ HP, BP, or BS)
- Stability & Precision improved
  - Using cascade of 1st/2nd order sections

Next Time…

- **Digital Windows**

  & **FIR!**

- Review:
  - Chapter 10 of Lathi

- A signal has many signals ☺
  [Unless it's bandlimited. Then there is *the one*]