Source: http://web.stanford.edu/class/me104b/cgi-bin/

http://elec3004.org

Frequency Response & Filter Analysis

(Digital filters are next week)

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

Lecture 6

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

April 13, 2015

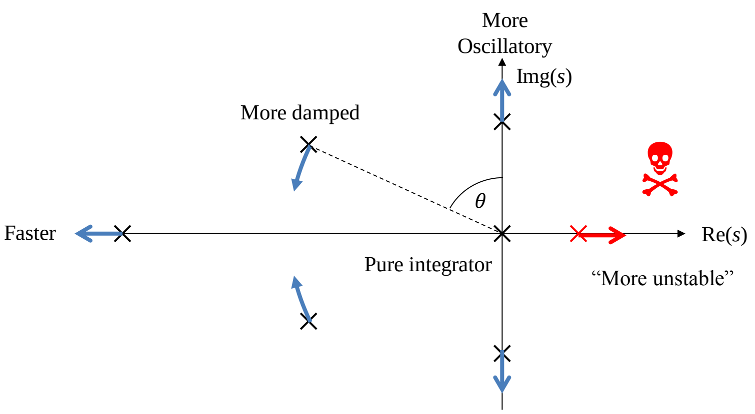
Lecture Schedule:

Week	Date	Lecture Title
1	2-Mar	Introduction
	3-Mar	Systems Overview
2	9-Mar	Signals as Vectors & Systems as Maps
	10-Mar	[Signals]
3	16-Mar	Sampling & Data Acquisition & Antialiasing Filters
	17-Mar	[Sampling]
4	23-Mar	System Analysis & Convolution
	24-Mar	[Convolution & FT]
5	30-Mar	Discrete Systems & Z-Transforms
	31-Mar	[Z-Transforms]
6	13-Apr	Frequency Response & Filter Analysis
	14-Apr	[Filters]
7	20-Apr	Digital Filters
	21-Apr	[Digital Filters]
8	27-Apr	Introduction to Digital Control
	28-Apr	[Feedback]
9	4-May	Digital Control Design
	5-May	[Digitial Control]
10	11-May	Stability of Digital Systems
	12-May	[Stability]
11	18-May	State-Space
	19-May	Controllability & Observability
12	25-May	PID Control & System Identification
	26-May	Digitial Control System Hardware
13	31-May	Applications in Industry & Information Theory & Communications
	2-Jun	Summary and Course Review

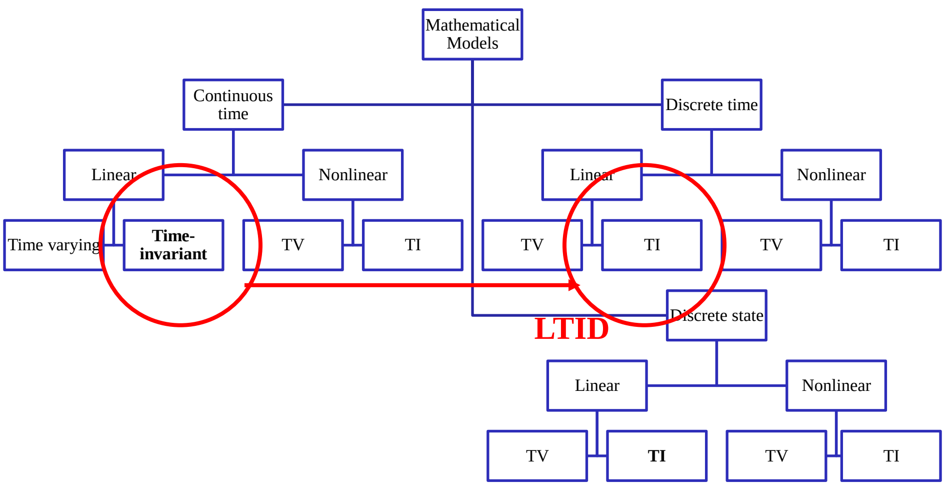
Dynamic Systems Review

Dynamic Responses (Poles & Zeros)

- Moving pole positions change system response characteristics



What about the Discrete Domain?



z Transforms

(Digital Systems Made eZ)

Review and Extended Explanation

Coping with Complexity

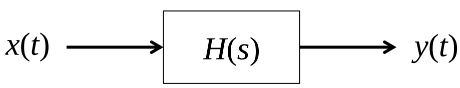
Transfer functions help control complexity

– Recall the Laplace transform:

$$\mathcal{L}\{f(t)\} = \int_0^\infty f(t)e^{-st}dt = F(s)$$

where

$$\mathcal{L}\{\dot{f}(t)\} = sF(s)$$



- Is there a something similar for sampled systems?

The z-Transform

- It is defined by:

$$z = re^{j\omega}$$

- Or in the Laplace domain:

$$z = e^{sT}$$

- That is → it is a discrete version of the Laplace:

$$f(kT) = e^{-akT} \Rightarrow \mathcal{Z}\{f(k)\} = \frac{z}{z - e^{-aT}}$$

The z-Transform [2]

- Thus:

$$Y(z) = \sum_{k=-\infty}^{\infty} h[k] z^{-k} \qquad y[n] \overset{\mathcal{Z}}{\longleftrightarrow} Y(z)$$

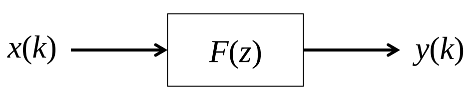
- z-Transform is analogous to other transforms:

$$\mathcal{Z}\{f(k)\} = \sum_{k=0}^{\infty} f(k) z^{-k} = F(z)$$

and

$$\mathcal{Z}\{f(k-1)\} = z^{-1} F(z)$$

∴ Giving:



The z-Transform [3]

- The z-Transform may also be considered from the Laplace transform of the impulse train representation of sampled signal

$$u^*(t) = u_0\delta(t) + u_1\delta(t-T) + \ \dots + u_{k(t-kT)} + \ \dots$$
$$= \sum_{k=0}^{\infty} u_k \delta(t-kT)$$
$$U^*(s) = u_0 + u_1 e^{-sT} + \cdots + u_k e^{-skT} + \cdots$$
$$= \sum_{k=0}^{\infty} u_k\, e^{-ksT}$$
$$U(z) = \sum_{k=0}^{\infty} u_k\, z^{-k}, \qquad z = e^{sT}$$

The *z*-transform

- In practice, you'll use look-up tables or computer tools (ie. Matlab) to find the *z*-transform of your functions

$F(s)$	$F(kt)$	$F(z)$
$\frac{1}{s}$	1	$\frac{z}{z-1}$
$\frac{1}{s^2}$	kT	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s+a}$	e^{-akT}	$\frac{z}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	kTe^{-akT}	$\frac{zTe^{-aT}}{(z-e^{-aT})^2}$
$\frac{1}{s^2+a^2}$	$\sin(akT)$	$\frac{z\sin aT}{z^2-(2\cos aT)z+1}$

z-Transform Example

- Obtain the z-Transform of the sequence:

$$x[k] = \{3, 0, 1, 4, 1, 5, \ldots\}$$

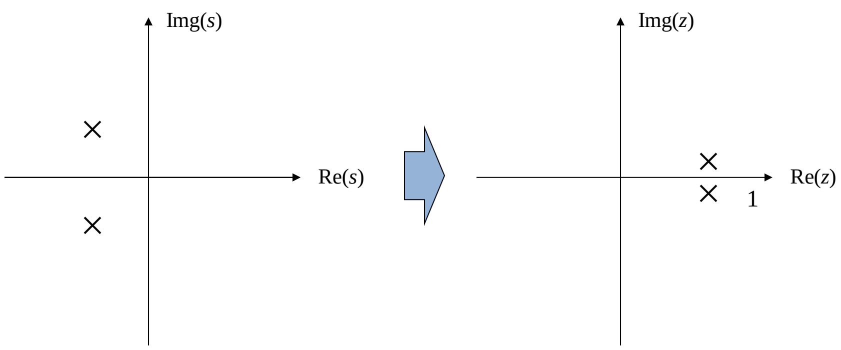
- Solution:

$$X(z) = 3 + z^{-2} + 4z^{-3} + z^{-4} + 5z^{-5}$$

The *z*-Plane

z-domain poles and zeros can be plotted just like *s*-domain poles and zeros (of the $\mathcal{L}$):

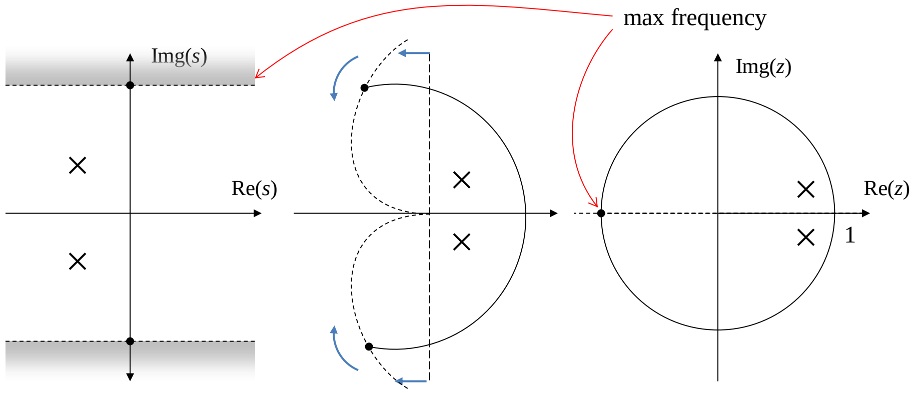
- S-plane:
- $\boldsymbol{z = e^{sT}}$ Plane



- s – Plane
- γ – Plane

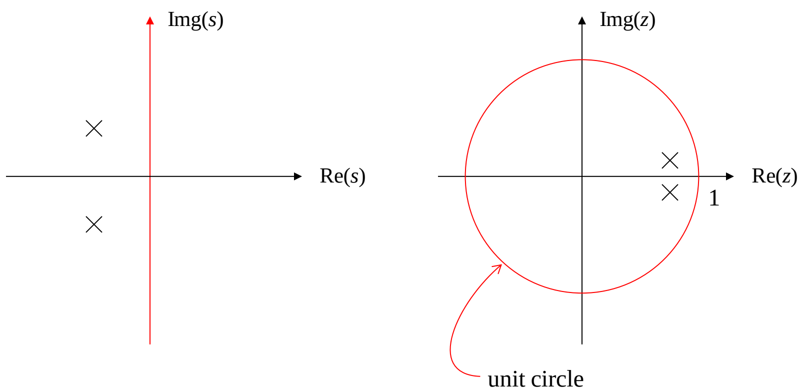
Deep insight #1

The mapping between continuous and discrete poles and zeros acts like a distortion of the plane



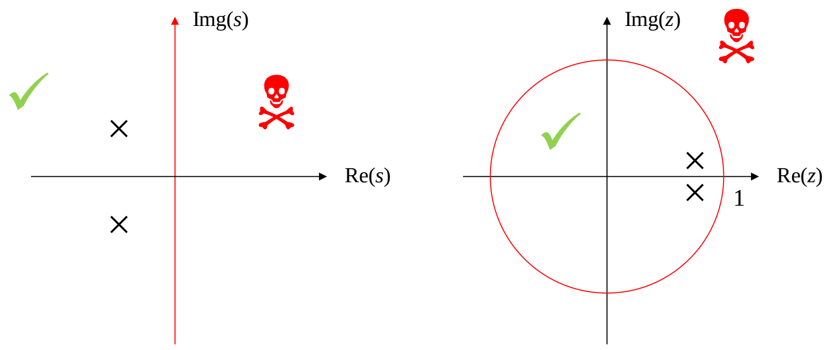
γ-plane Stability

- For a γ-Plane (e.g. the one the *z*-domain is embedded in) the unit circle is the system stability bound



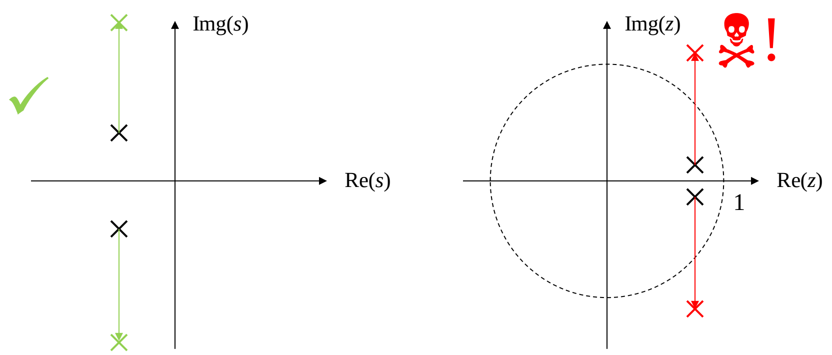
γ-plane Stability

- That is, in the *z*-domain,
  the unit circle is the system stability bound



z-plane stability

- The z-plane root-locus in closed loop feedback behaves just like the s-plane:



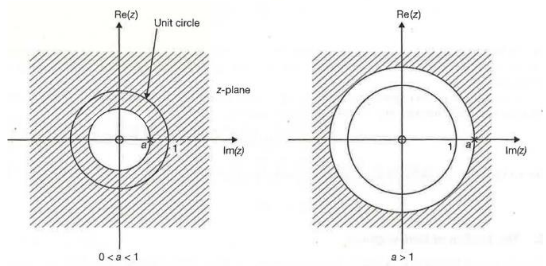
Region of Convergence

- For the convergence of X(z) we require that

$$\sum_{n=0}^{\infty} |az^{-1}|^n < \infty$$

- Thus, the ROC is the range of values of z for which $|az^{-1}| < 1$ or, equivalently, $|z| > |a|$. Then

$$X(z) = \frac{z}{z-a} \qquad |z| > |a|$$



An example!

- Back to our difference equation:

$$y(k) = x(k) + Ax(k-1) - By(k-1)$$

becomes

$$Y(z) = X(z) + Az^{-1}X(z) - Bz^{-1}Y(z)$$
$$(z+B)Y(z) = (z+A)X(z)$$

which yields the transfer function:

$$\frac{Y(z)}{X(z)} = \frac{z+A}{z+B}$$

Note: It is also not uncommon to see systems expressed as polynomials in z^{-n}

This looks familiar…

- Compare:

$$\frac{Y(s)}{X(s)} = \frac{s+2}{s+1} \text{ vs } \frac{Y(z)}{X(z)} = \frac{z+A}{z+B}$$

How are the Laplace and *z* domain representations related?

Linearity:

$$a_1 y_1[n] + a_2 y_2[n] \overset{\mathcal{Z}}{\longleftrightarrow} a_1 Y_1(z) + a_2 Y_2(z)$$

Z-Transform Properties: Time Shifting

$$y[n-n_0] \overset{\mathcal{Z}}{\longleftrightarrow} z^{-n_0} Y(z)$$

$$\begin{aligned} y_2[n] &= y[n-n_0] \\ Y_2(e^{jw}) &= \sum_{k=-\infty}^{\infty} y[k-n_0] z^{-k} \\ &= \sum_{l=-\infty}^{\infty} y[l] z^{-(l+n_0)} \\ &= z^{-n_0} Y(z) \end{aligned}$$

- Two Special Cases:
- z^{-1}: the *unit-delay operator:*

$$x[n-1] \leftrightarrow z^{-1}X(z) \qquad R' = R \cap \{0 < |z|\}$$

- z: *unit-advance operator:*

$$x[n+1] \leftrightarrow zX(z) \qquad R' = R \cap \{|z| < \infty\}$$

More Z-Transform Properties

- Time Reversal

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$x[-n] \leftrightarrow X\left(\frac{1}{z}\right) \qquad R' = \frac{1}{R}$$

- Multiplication by z^n

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$z_0^n x[n] \leftrightarrow X\left(\frac{z}{z_0}\right) \qquad R' = |z_0| R$$

- Multiplication by n (or Differentiation in z):

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$nx[n] \leftrightarrow -z\frac{dX(z)}{dz} \qquad R' = R$$

- Convolution

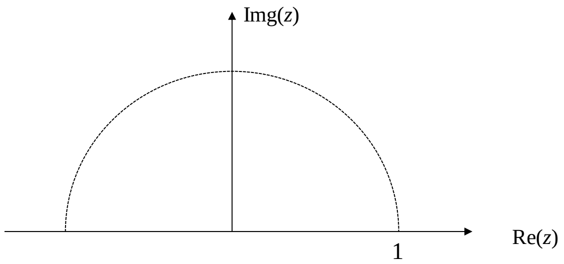
$$x_1[n] \leftrightarrow X_1(z) \qquad \text{ROC} = R_1$$
$$x_2[n] \leftrightarrow X_2(z) \qquad \text{ROC} = R_2$$

$$x_1[n] * x_2[n] \leftrightarrow X_1(z)X_2(z) \qquad R' \supset R_1 \cap R_2$$

The *z*-plane [for all pole systems]

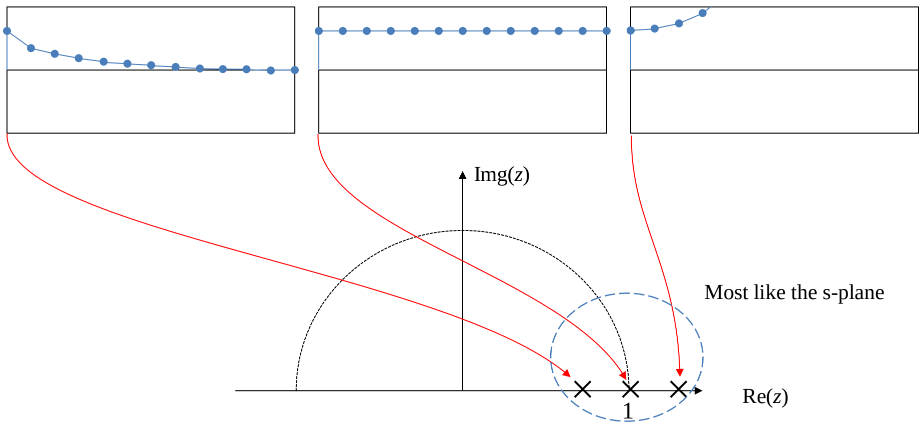
- We can understand system response by pole location in the *z*-plane

[Adapted from Franklin, Powell and Emami-Naeini]



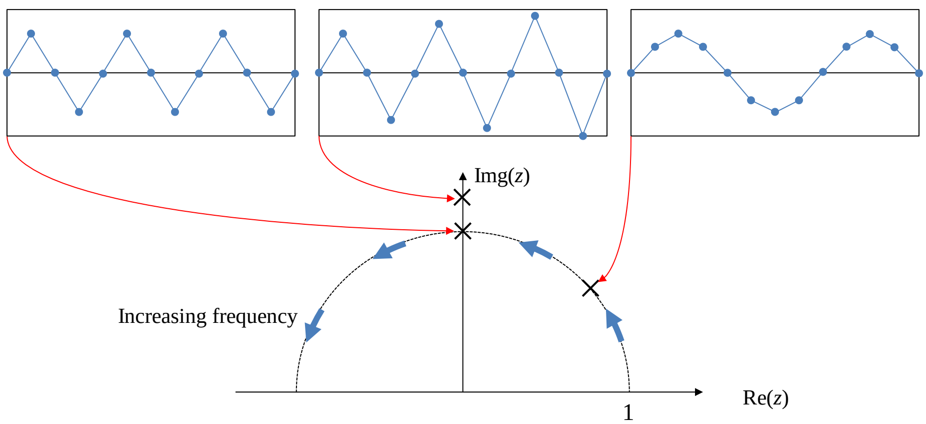
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



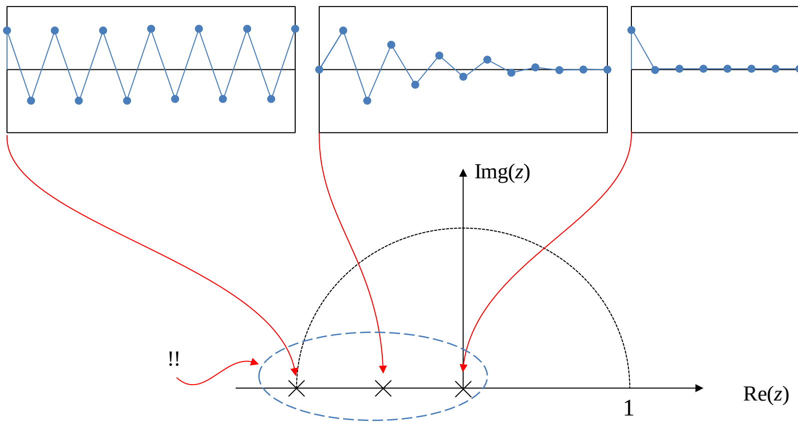
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



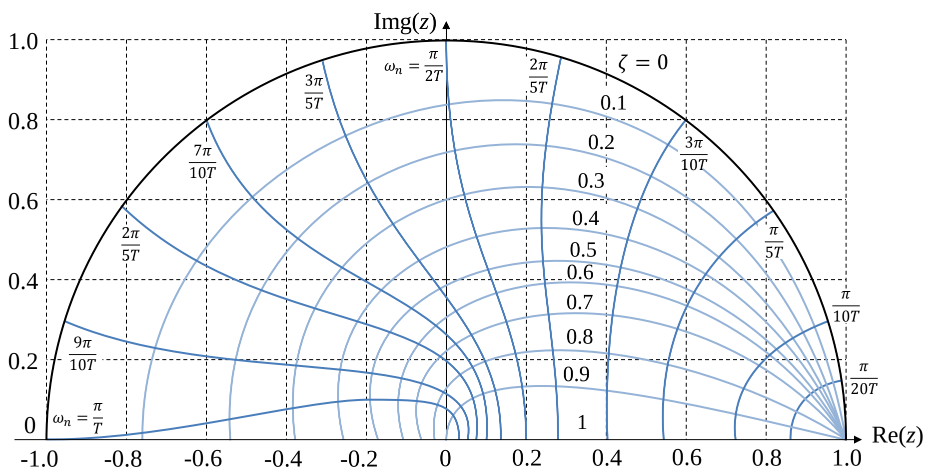
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



z-Plane Response for 2nd Order Systems: Damping (ζ) and Natural frequency (ω)

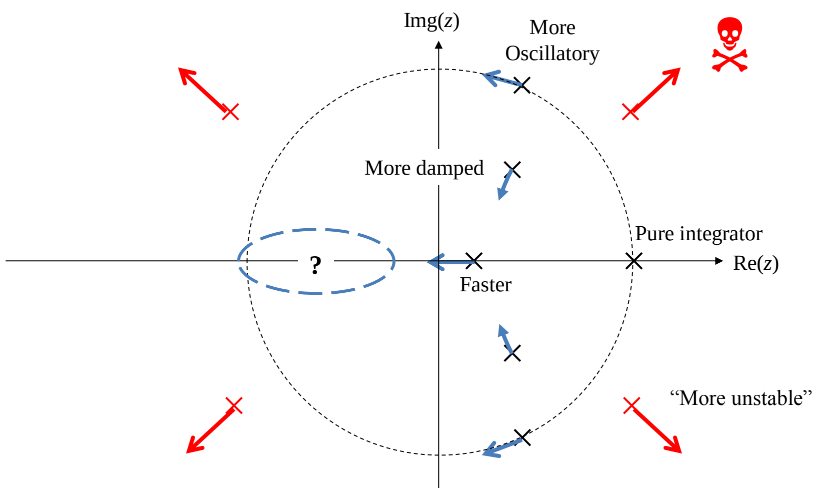
$$z = e^{sT} \text{ where } s = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$$



[Adapted from Franklin, Powell and Emami-Naeini]

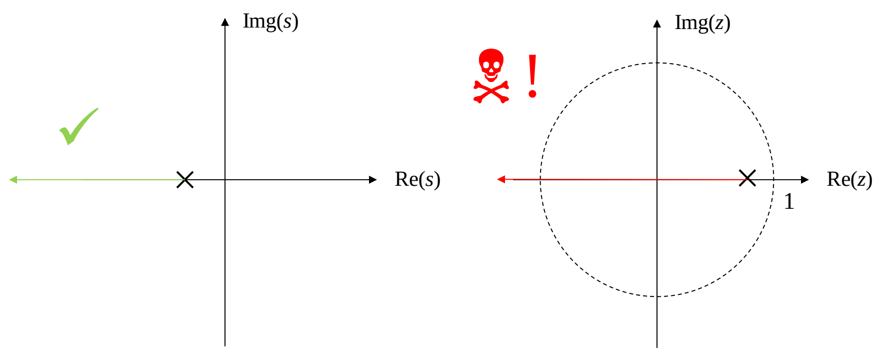
Recall dynamic responses

- Ditto the z-plane:



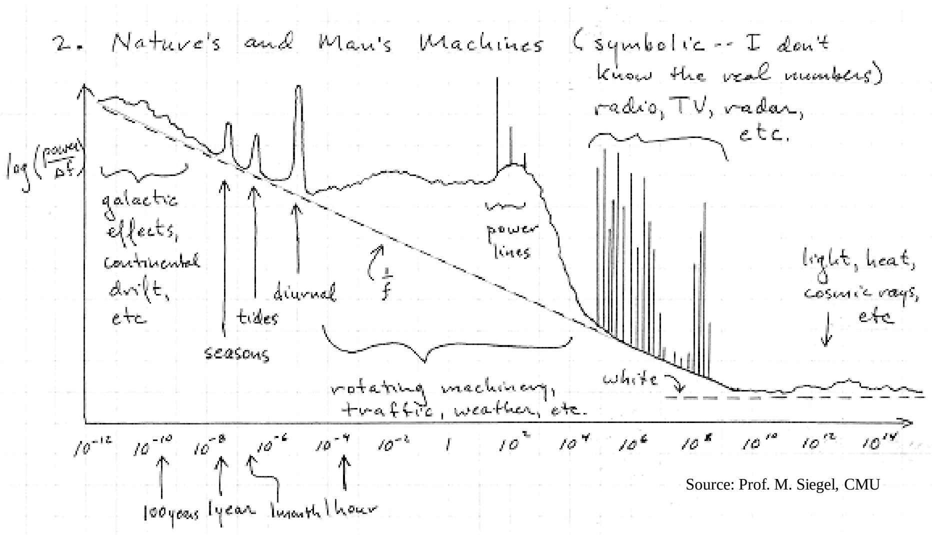
Deep insight #2

- Gains that stabilise continuous systems can actually ***destabilise*** digital systems!



First Some Noise!

Noise



Source: Prof. M. Siegel, CMU

Note: this picture illustrates the concepts but it is not quantitatively precise

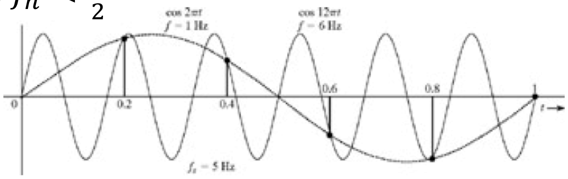
Noise [2]

Various Types:

- Thermal (white):
  - Johnson noise, from thermal energy inherent in mass.
- Flicker or 1/f noise:
  - Pink noise
  - More noise at lower frequency
- Shot noise:
  - Noise from quantum effects as current flows across a semiconductor barrier
- Avalanche noise:
  - Noise from junction at breakdown (circuit at discharge)

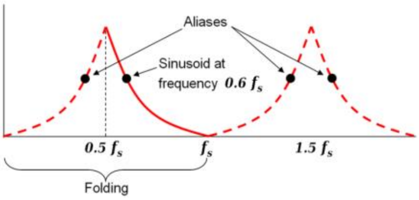
Also Aliasing & Sampling (It is a type of "noise")

- Nyquist: $f_h < \frac{f_s}{2}$



- Spectral Folding:

$$f_{image}(N) = f - Nf_s$$



Noise ⊆ Uncertainty

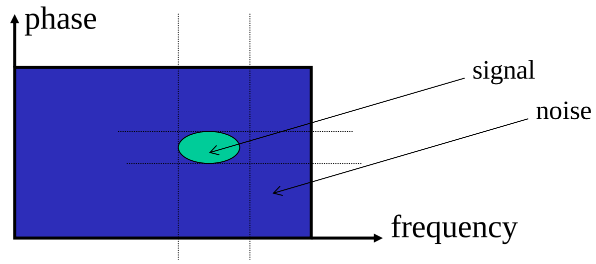
- **<u>Uncertainty</u>**:
  All measurement has some approximation
  A. <u>Statistical uncertainty</u>: quantified by mean & variance
  B. <u>Systematic uncertainty</u>: non-random error sources

- **<u>Law of Propagation of Uncertainty</u>**
  – Combined uncertainty is root squared

$$u_c = \sqrt{u_1^2 + u_2^2 + \ldots + u_n^2}$$

How to beat the noise

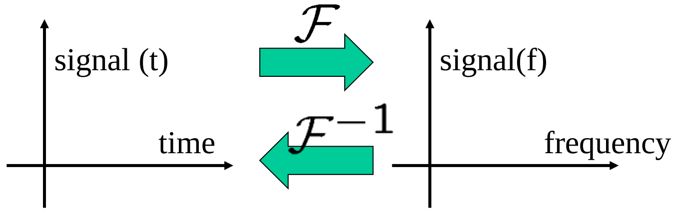
- **Filtering** (Narrow-banding):
  Only look at particular portion of **<u>frequency space</u>**
- Multiple measurements …
- Other (modulation, etc.) …



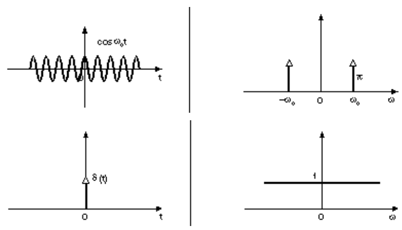
➡ By adding shared **<u>information</u>** (structure) between the sender and receiver (the noise doesn't know your structure)

Frequency

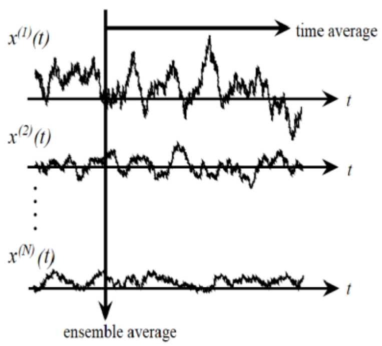
- How often the signal repeats
- Can be analyzed through Fourier Transform



- Examples:



Treating Uncertainty with Multiple Measurements



1. **Over time:** multiple readings of a quantity over time
   - "stationary" or "ergodic" system
   - Sometimes called "integrating"
2. **Over space:** single measurement (summed) from multiple sensors each distributed in space
3. **Same Measurand**: multiple measurements take of the **same observable quantity** by multiple, related instruments

   e.g., measure position & velocity simultaneously

→ Basic "sensor fusion"

.

$$\sigma_{\text{final}} = \left[\sigma_1^{-1} + \sigma_2^{-1} + \cdots + \sigma_n^{-1}\right]^{-1}$$

Multiple Measurements Example

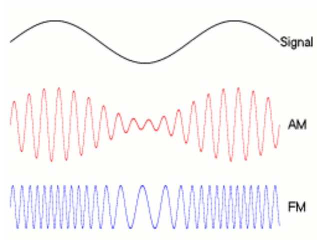
- What time was it when this picture was taken?
- What was the temperature in the room?

- Estimation (Information Fusion) Problem
  - A Solution: Linear Least Squares
    (over-determined simultaneous set of equations)

Modulation

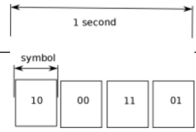
Analog Methods:

- AM - Amplitude modulation
  - Amplitude of a (carrier) is modulated to the (data)
- FM - Frequency modulation
  - Frequency of a (carrier) signal is varied in accordance to the amplitude of the (data) signal
- PM – Phase Modulation



Source: http://en.wikipedia.org/wiki/Modulation

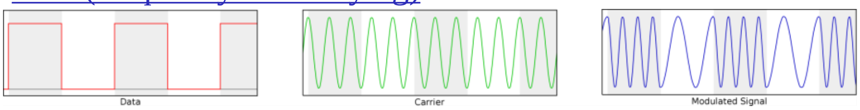
Modulation [Digital Methods]



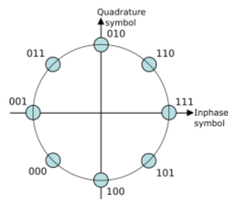
Start with a "**symbol**" & place it on a channel

- [ASK (amplitude-shift keying)](http://en.wikipedia.org/wiki/Modulation)

- FSK (frequency-shift keying)



- PSK (phase-shift keying)
- QAM (quadrature amplitude modulation)



$s(t) = A \cdot cos(\omega_c + \phi_i(t))$

$= x_i(t)\cos(\omega_c t) + x_q(t)\sin(\omega_c t)$

Source: http://en.wikipedia.org/wiki/Modulation | http://users.ecs.soton.ac.uk/sqc/EL334 | http://en.wikipedia.org/wiki/Constellation_diagram

Modulation [Example – V.32bis Modem]

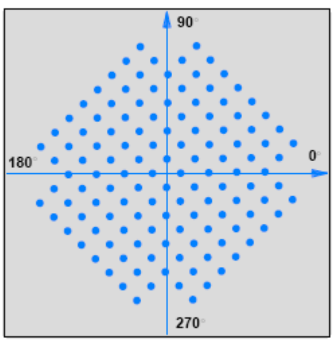
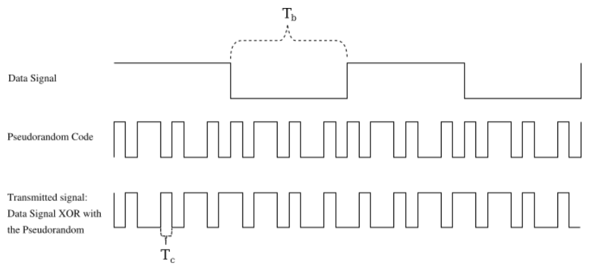


Figure 10.13 Illustration of the QAM constellation for a V.32bis dialup modem.

Source: Computer Networks and Internets, 5e, Douglas E. Comer

Multiple Access (Channel Access Method)

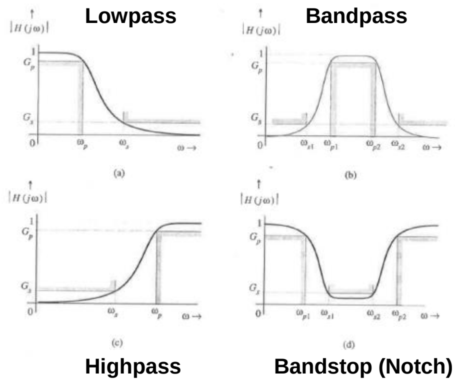
- Send multiple signals on 1 to N channel(s)
  - Frequency-division multiple access (FDMA)
  - Time-division multiple access (TDMA)
  - Code division multiple access (CDMA)
  - Space division multiple access (SDMA)
- CDMA:
  - Start with a pseudorandom code (the noise doesn't know your code)



Source: http://en.wikipedia.org/wiki/Code_division_multiple_access

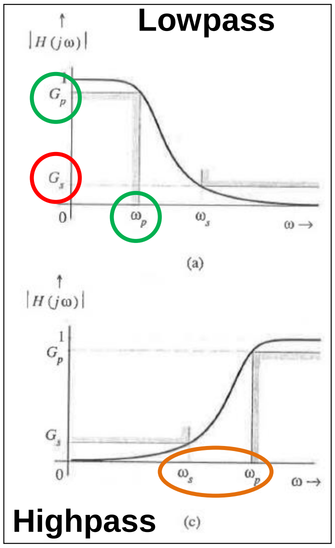
Now: (analog) **Filters!**

Filters



- *Frequency-shaping filters*: LTI systems that change the shape of the spectrum
- *Frequency-selective filters:* Systems that pass some frequencies undistorted and attenuate others

Filters



Specified Values:

- Gp = minimum passband gain

Typically:

$$G_p = \frac{1}{\sqrt{2}} = -3dB$$

- Gs = maximum stopband gain
  - **Low**, not zero (sorry!)
  - For realizable filters, the gain cannot be zero over a finite band (Paley-Wiener condition)
- Transition Band:

  transition from the passband to the stopband ➔ ωp≠ ωs

Filter Design & z-Transform

Filter Type	Mapping	Design Parameters
Low-pass	$z^{-1} \to \frac{z^{-1}-\alpha}{1-\alpha z^{-1}}$	$\alpha = \frac{\sin[(\omega_c - \omega_c')/2]}{\sin[(\omega_c + \omega_c')/2]}$ ω_c' = desired cutoff frequency
High-pass	$z^{-1} \to -\frac{z^{-1}+\alpha}{1+\alpha z^{-1}}$	$\alpha = -\frac{\cos[(\omega_c + \omega_c')/2]}{\cos[(\omega_c - \omega_c')/2]}$ ω_c' = desired cutoff frequency
Bandpass	$z^{-1} \to -\frac{z^{-2} - [2\alpha\beta/(\beta+1)]z^{-1} + [(\beta-1)/(\beta+1)]}{[(\beta-1)/(\beta+1)]z^{-2} - [2\alpha\beta/(\beta+1)]z^{-1} + 1}$	$\alpha = \frac{\cos[(\omega_{c2} + \omega_{c1})/2]}{\cos[(\omega_{c2} - \omega_{c1})/2]}$ $\beta = \cot[(\omega_{c2} - \omega_{c1})/2]\tan(\omega_c/2)$ ω_{c1} = desired lower cutoff frequency ω_{c2} = desired upper cutoff frequency
Bandstop	$z^{-1} \to \frac{z^{-2} - [2\alpha/(\beta+1)]z^{-1} + [(1-\beta)/(1+\beta)]}{[(1-\beta)/(1+\beta)]z^{-2} - [2\alpha/(\beta+1)]z^{-1} + 1}$	$\alpha = \frac{\cos[(\omega_{c1} + \omega_{c2})/2]}{\cos[(\omega_{c1} - \omega_{c2})/2]}$ $\beta = \tan[(\omega_{c2} - \omega_{c1})/2]\tan(\omega_c/2)$ ω_{c1} = desired lower cutoff frequency ω_{c2} = desired upper cutoff frequency

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:

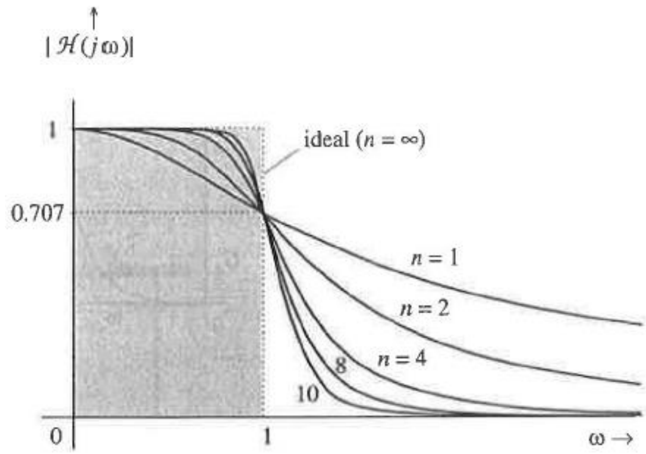
$$|H(j\omega)| = \frac{1}{\sqrt{1+\left(\frac{\omega}{\omega_c}\right)^{2n}}}$$

- The normalized case (ω_c=1)

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

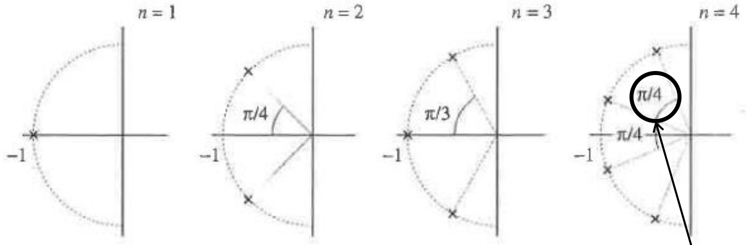
Recall that: $|H(j\omega)|^2 = H(j\omega)\,H(-j\omega)$

Butterworth Filters



Butterworth Filters of Increasing Order: Seeing this Using a Pole-Zero Diagram

- Increasing the order, increases the number of poles:



➔Odd orders (n=1,3,5…):

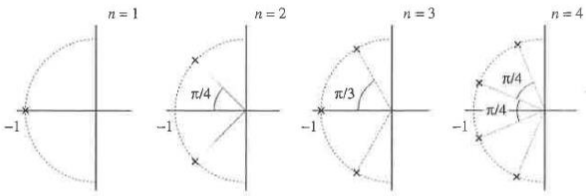
- Have a pole on the Real Axis

➔Even orders (n=2,4,6…):

- Have a pole on the off axis

Angle between poles: $\frac{\pi}{n}$

Butterworth Filters: Pole-Zero Diagram



- Since H(s) is stable and causal, its poles must lie in the LHP
- Poles of -H(s) are those in the RHP
- Poles lie on the unit circle (for a normalized filter)

n is the order of the filter

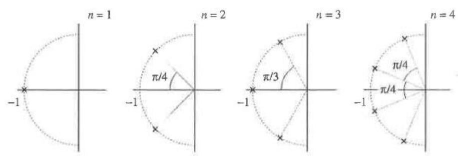
$$\Rightarrow H(s) = \frac{1}{(s - s_1)(s - s_2)\ldots(s - s_n)}$$

Where:

$$s_k = e^{\frac{j\pi}{2n}(2k+n-1)}$$

$$= \cos\frac{\pi}{2n}(2k+n-1) + j\sin\frac{\pi}{2n}(2k+n-1) \qquad k = 1, 2, 3, \ldots, n$$

Butterworth Filters: 4^{th} Order Filter Example



- Plugging in for n=4, k=1,…4:

$$\mathcal{H}(s) = \frac{1}{(s+0.3827-j0.9239)(s+0.3827+j0.9239)(s+0.9239-j0.3827)(s+0.9239+j0.3827)}$$

$$= \frac{1}{(s^2+0.7654s+1)(s^2+1.8478s+1)}$$

$$= \frac{1}{s^4+2.6131s^3+3.4142s^2+2.6131s+1}$$

- We can generalize ➔ Butterworth Table

n	a_1	a_2	a_3	a_4	a_5
2	1.41421356				
3	2.00000000	2.00000000			
4	2.61312593	3.41421356	2.61312593		
5	3.23606798	5.23606798	5.23606798	3.23606798	
6	3.86370331	7.46410162	9.14162017	7.46410162	3.86370331

This is for 3dB bandwidth at ω_c=1

Butterworth Filters: Scaling Back (from Normalized)

- Start with Normalized equation & Table
- Replace ω with $\frac{\omega}{\omega_c}$ in the filter equation

- For example:
  for f_c=100Hz → ω_c=200π rad/sec

From the Butterworth table: for n=2, $a_1=\sqrt{2}$
Thus:

$$H(s) = \frac{1}{\left(\frac{s}{200\pi}\right)^2 + \sqrt{2}\left(\frac{s}{200\pi}\right) + 1}$$
$$= \frac{1}{s^2 + 200\pi\sqrt{2} + 40{,}000\pi^2}$$

Butterworth: Determination of Filter Order

- Define $\mathbf{G_x}$ as the gain of a lowpass Butterworth filter at $\omega = \omega_x$
- Then:

$$\hat{G}_x = 20\log_{10}|H(j\omega_x)| = -10\log\left[1 + \left(\frac{\omega_x}{\omega_c}\right)^{2n}\right]$$

And thus:

$$\hat{G}_p = -10\log\left[1 + \left(\frac{\omega_p}{\omega_c}\right)^{2n}\right]$$

$$\hat{G}_s = -10\log\left[1 + \left(\frac{\omega_s}{\omega_c}\right)^{2n}\right]$$

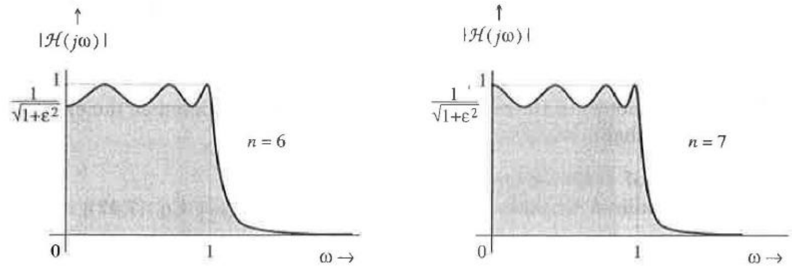
Or alternatively: $\omega_c = \frac{\omega_p}{\left[10^{-\hat{G}_p/10} - 1\right]^{1/2n}}$ & $\omega_c = \frac{\omega_s}{\left[10^{-\hat{G}_s/10} - 1\right]^{1/2n}}$

Solving for n gives:

$$n = \frac{\log\left[\left(10^{-\hat{G}_s/10} - 1\right) / \left(10^{-\hat{G}_p/10} - 1\right)\right]}{2\log(\omega_s/\omega_p)}$$

PS. See Lathi 4.10 (p. 453) for an example in MATLAB

Chebyshev Filters



- **equal-ripple:**
  Because all the ripples in the passband are of equal height
- If we reduce the ripple, the passband behaviour improves, but it does so at the cost of stopband behaviour

Chebyshev Filters

- Chebyshev Filters: Provide tighter transition bands (sharper cutoff) than the same-order Butterworth filter, but this is achieved at the expense of inferior passband behavior (rippling)

➔ For the lowpass (LP) case: at higher frequencies (in the stopband), the Chebyshev filter gain is smaller than the comparable Butterworth filter gain by about ***6(n - 1) dB***

- The amplitude response of a normalized Chebyshev lowpass filter is:

$$|\mathcal{H}(j\omega)| = \frac{1}{\sqrt{1+\epsilon^2 {C_n}^2(\omega)}}$$

Where Cn(ω), the nth-order Chebyshev polynomial, is given by:

$$C_n(\omega) = \cos\left(n \cos^{-1} \omega\right)$$
$$C_n(\omega) = \cosh\left(n \cosh^{-1} \omega\right)$$

and where C_n is given by:

n	$C_n(\omega)$
0	1
1	ω
2	$2\omega^2 - 1$
3	$4\omega^3 - 3\omega$
4	$8\omega^4 - 8\omega^2 + 1$
5	$16\omega^5 - 20\omega^3 + 5\omega$
6	$32\omega^6 - 48\omega^4 + 18\omega^2 - 1$

Normalized Chebyshev Properties

- It's normalized: The passband is $0<\omega<1$
- **Amplitude response**: has **ripples** in the passband and is **smooth** (monotonic) in the stopband
- **Number of ripples**: there is a total of ***n*** maxima and minima over the passband $0<\omega<1$
- $C_n^2(0) = \begin{cases} 0, & n: odd \\ 1, & n: even \end{cases}$ $\Rightarrow$ $|H(0)| = \begin{cases} 1, & n: odd \\ \frac{1}{\sqrt{1+\epsilon^2}}, & n: even \end{cases}$
- ϵ: ripple height $\rightarrow r = \sqrt{1+\epsilon^2}$
- The Amplitude at $\omega=1$: $\frac{1}{r} = \frac{1}{\sqrt{1+\epsilon^2}}$
- For Chebyshev filters, the ripple ***r*** dB takes the place of G_p

Determination of Filter Order

- The gain is given by: $\hat{G} = -10 \log \left[1 + \epsilon^2 {C_n}^2(\omega)\right]$

Thus, the gain at ω_s is: $\epsilon^2 {C_n}^2(\omega_s) = 10^{-\hat{G}_s/10} - 1$

- Solving:

$$n = \frac{1}{\cosh^{-1}(\omega_s)} \cosh^{-1} \left[\frac{10^{-\hat{G}_s/10} - 1}{10^{\hat{r}/10} - 1}\right]^{1/2}$$

- General Case:

$$n = \frac{1}{\cosh^{-1}(\omega_s/\omega_p)} \cosh^{-1} \left[\frac{10^{-\hat{G}_s/10} - 1}{10^{\hat{r}/10} - 1}\right]^{1/2}$$

Chebyshev Pole Zero Diagram

- Whereas Butterworth poles lie on a semi-circle,
  The poles of an n^{th}-order normalized Chebyshev filter lie on a **semiellipse** of the major and minor semiaxes:

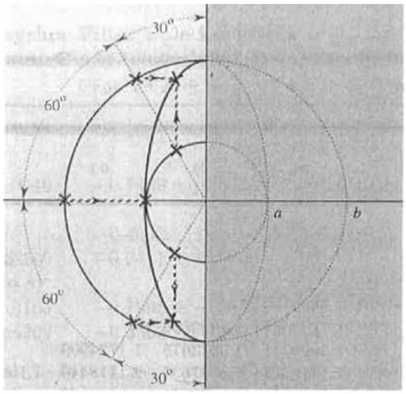
$$a = \sinh\left(\frac{1}{n}\sinh^{-1}\left(\frac{1}{\epsilon}\right)\right) \quad \& \quad b = \cosh\left(\frac{1}{n}\sinh^{-1}\left(\frac{1}{\epsilon}\right)\right)$$

And the poles are at the locations:

$$H(s) = \frac{1}{(s-s_1)(s-s_2)\dots(s-s_n)}$$

$$s_k = -\sin\left[\frac{(2k-1)\pi}{2n}\right]\sinh x + j\cos\left[\frac{(2k-1)\pi}{2n}\right]\cosh x,\ k = 1,\dots,n$$

Ex: Chebyshev Pole Zero Diagram for **n=3**



Procedure:

1. Draw two semicircles of radii **a** and **b** (from the previous slide).
2. Draw radial lines along the corresponding Butterworth angles (π/n) and locate the n^{th}-order Butterworth poles (shown by crosses) on the two circles.
3. The location of the k^{th} Chebyshev pole is the intersection of the horizontal projection and the vertical projection from the corresponding k^{th} Butterworth poles on the outer and the inner circle, respectively.

Chebyshev Values / Table

$$\mathcal{H}(s) = \frac{K_n}{C'_n(s)} = \frac{K_n}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}$$

$$K_n = \begin{cases} a_0 & n \text{ odd} \\ \frac{a_0}{\sqrt{1+\epsilon^2}} = \frac{a_0}{10^{\hat{r}/20}} & n \text{ even} \end{cases}$$

n	a_0	a_1	a_2	a_3
1	1.9652267			
2	1.1025103	1.0977343		
3	0.4913067	1.2384092	0.9883412	
4	0.2756276	0.7426194	1.4539248	0.9528114

1 db ripple
($\hat{r} = 1$)

Other Filter Types:
Chebyshev Type II = Inverse Chebyshev Filters

- Chebyshev filters passband has ripples and the stopband is smooth.
- **Instead:** this has **passband** have **smooth** response and **ripples** in the stopband.

➔ Exhibits maximally flat passband response and equi-ripple stopband

➔ **Cheby2 in MATLAB**

$$|\mathcal{H}(\omega)|^2 = 1 - |\mathcal{H}_C(1/\omega)|^2 = \frac{\epsilon^2 C_n^2(1/\omega)}{1 + \epsilon^2 C_n^2(1/\omega)}$$

Where: H_c is the Chebyshev filter system from before

- Passband behavior, especially for small ω, is **better** than Chebyshev
- **Smallest transition band** of the 3 filters (Butter, Cheby, Cheby2)
- Less time-delay (or phase loss) than that of the **Chebyshev**
- Both needs the **same order *n*** to meet a set of specifications.
- **$$$** (or number of elements):
  Cheby < Inverse Chebyshev < Butterworth (of the same **performance** [not order])

Other Filter Types:
Elliptic Filters (or Cauer) Filters

- Allow **ripple** in **both** the passband and the stopband,
  ➔ we can achieve **tighter** transition band

$$|\mathcal{H}(j\omega)| = \frac{1}{\sqrt{1+\epsilon^2 {R_n}^2(\omega)}}$$

Where: $\mathbf{R_n}$ is the n^{th}-order Chebyshev rational function determined from a given ripple spec.
ϵ controls the ripple
$Gp = \frac{1}{\sqrt{1+\epsilon^2}}$

- Most efficient (η)
  - the **largest ratio** of the passband gain to stopband gain
  - **or** for a given ratio of passband to stopband gain, it requires the **smallest transition band**

➔ **in MATLAB:** `elliord` followed by `ellip`

In 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`

Linear Difference Equations
(a sub-set of Linear, Discrete Dynamical Systems)

DT Causality & BIBO Stability [Review]

- Causality:

$$h[n] = 0,\ n < 0$$

$$\rightarrow y[n] = \sum_{k=0}^{\infty} h[k]x[n-k] \quad \text{or} \quad \Rightarrow y[n] = \sum_{k=-\infty}^{n} x[k]h[n-k]$$

- Input is Causal if: $x[n] = 0,\ n < 0$

- Then output is Causal:

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

- And, DT LTI is BIBO stable if:

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

Linear Difference Equations

$$u_k = f(e_0, \dots, e_k; u_0, \dots, u_{k-1}).$$

$$u_k = -a_1 u_{k-1} - a_2 u_{k-2} - \cdots - a_n u_{k-n} + b_0 e_k + b_1 e_{k-1} + \cdots + b_m e_{k-m}.$$

$$\begin{aligned} \nabla u_k &= u_k - u_{k-1} && \text{(first difference)}, \\ \nabla^2 u_k &= \nabla u_k - \nabla u_{k-1} && \text{(second difference)}, \\ \nabla^n u_k &= \nabla^{n-1} u_k - \nabla^{n-1} u_{k-1} && \text{(}n\text{th difference)}. \end{aligned}$$

$$\begin{aligned} u_k &= u_k, \\ u_{k-1} &= u_k - \nabla u_k, \\ u_{k-2} &= u_k - 2\nabla u_k + \nabla^2 u_k. \end{aligned}$$

$$a_2 \nabla^2 u_k - (a_1 + 2a_2)\nabla u_k + (a_2 + a_1 + 1)u_k = b_0 e_k.$$

Assume a form of the solution

z^k :

- k: “order of difference”
- k: delay

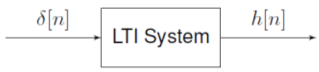
$$Az^k = Az^{k-1} + Az^{k-2}.$$

$$1 = z^{-1} + z^{-2}$$

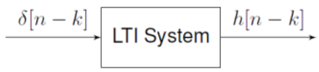
$$z^2 = z + 1.$$

Impulse Response (Graphically)

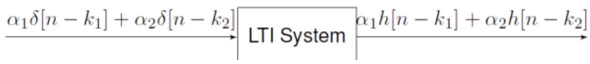
Let's define the *impulse response*, $h[n]$, as the result of applying an LTI system to the unit impulse:

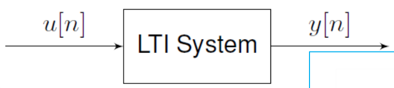


By time invariance, we know



And by linearity, we know





$$u[n] = \sum_{k=-\infty}^{\infty} u[k]\delta[n-k]$$

$$y[n] = \sum_{k=-\infty}^{\infty} u[k]h[n-k]$$

∞ matrix × ∞ vector?

How do you multiply an infinite matrix?

- First let's multiply circulant matrices…
  - A circulant matrix can be descibed completely by its first row or column

$$A = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{n-1} \\ a_{n-1} & a_0 & a_1 & \cdots & a_{n-2} \\ a_{n-2} & a_{n-1} & a_0 & \cdots & a_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & a_3 & \cdots & a_0 \end{bmatrix} = \begin{bmatrix} | & | & | & & | \\ h & Zh & Z^2h & \cdots & Z^{N-1}h \\ | & | & | & & | \end{bmatrix}$$

Z: Shift operator

- Multiply by u[k] ➔ $\begin{bmatrix} | & | & | & & | \\ h & Zh & Z^2h & \cdots & Z^{N-1}h \\ | & | & | & & | \end{bmatrix} \begin{bmatrix} u[0] \\ u[1] \\ u[2] \\ \vdots \\ u[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} u[k]Z^k h$

∴ For circulant matrices, matrix multiplication reduces to a weighted combination of shifted impulse responses

Two Types of Systems

- Linear shift-invariant:

$$y = \sum_{k=0}^{N-1} u[k] Z^k h$$

Z: Shift operator

$$Z \cdot [u_0, u_1, u_2, u_3, \ldots, u_{n-1}]^T = [u_{n-1}, u_0, u_1, u_2, \ldots, u_{n-2}]^T$$

- Linear time-invariant system

$$y = \sum_{k=-\infty}^{\infty} u[k] R^k h$$

R: Unit delay operator

$$R \cdot [\ldots, u_0, u_1, u_2, u_3, \ldots]^T = [\ldots, u_{-1}, u_0, u_1, \ldots]^T$$

Impulse Response of Both Types

$$y[n] = \frac{1}{2}u[n-1] + \frac{1}{2}u[n]$$

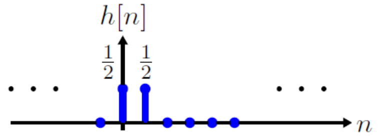
$$y[-1] = 0$$
$$y[0] = \frac{1}{2}$$
$$y[1] = \frac{1}{2}$$
$$y[2] = 0$$
$$\vdots$$

$$y[n] = \frac{1}{2}y[n-1] + u[n]$$

$$h[-1] = 0$$
$$h[0] = 1$$
$$h[1] = \frac{1}{2}$$
$$h[2] = \frac{1}{4}$$
$$\vdots$$
$$h[n] = \begin{cases} 0 & n < 0 \\ (\frac{1}{2})^n & n \geq 0 \end{cases}$$

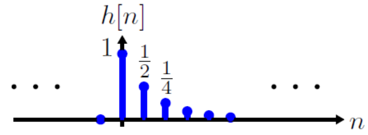
Impulse Response of Both Types

$$y[n] = \frac{1}{2}u[n-1] + \frac{1}{2}u[n]$$



'Finite impulse response" (FIR)

$$y[n] = \frac{1}{2}y[n-1] + u[n]$$



"Infinite impulse response" (IIR)