CTP431: Fundamentals of Computer Music

Digital Filters

KAIST Graduate School of Culture Technology

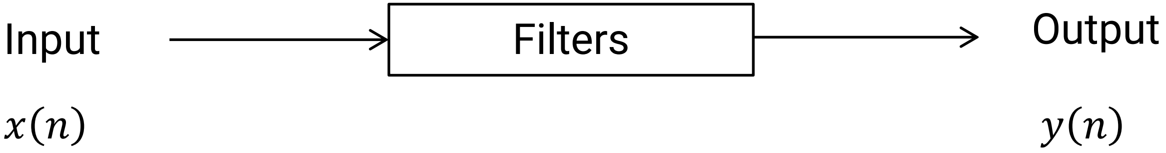
Juhan Nam

Goals

- The fundamentals of digital filters
  - Linear time-invariant (LTI) system
  - FIR / IIR Filters
  - Convolution
  - Impulse response
  - Frequency response

Digital Filters

- Take the input signal $x(n)$ as a sequence of numbers and returns the output signal $y(n)$ as another sequence of numbers
- Perform a combination of three mathematic operations upon the input or the output
  - **Multiplication**: $y(n) = b \cdot x(n)$
  - **Delay**: $y(n) = x(n - M)$
  - **Summation**: $y(n) = x(n) + a \cdot y(n - M)$



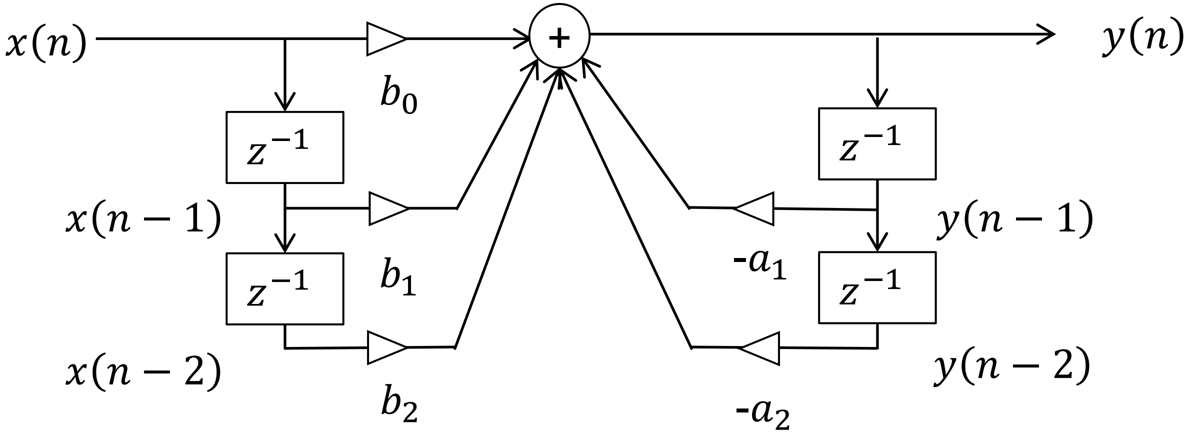
Commonly Used Digital Filters in Computer Music

- **Bi-quads filters**
  - lowpass, bandpass, highpass, equalizers
  - Change timbre
- **Comb filter (delayline)**
  - delay/echo, chorus, flanger, reverb
  - Change timbre or add spatial effect
- **Convolution**
  - Reverberation (using room impulse responses), HRTF
  - Add spatial effect

Bi-quad Filter

- Two delay elements for input or output
- The delayed output (“feedback”) causes **resonance**
- Rooted from analog circuits (R-L-C)

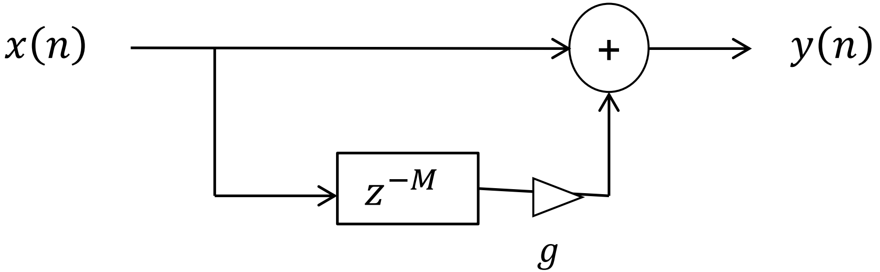
$$y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) - a_1 \cdot y(n-1) - a_2 \cdot y(n-2)$$



Comb Filter

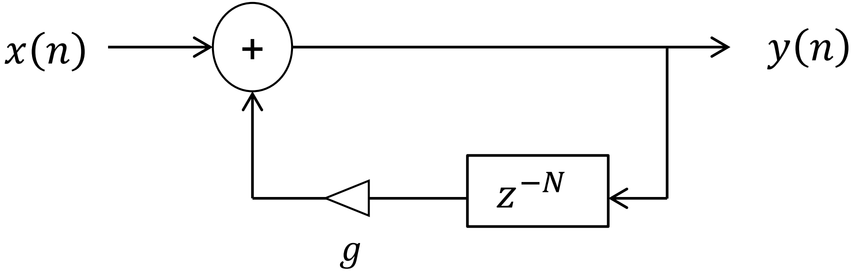
- Use a delayline
- Feedforward is used for chorus/flanger and feedback is for echo/reverb
- Rooted from magnetic tape recording

$$y(n) = x(n) + g \cdot x(n - M)$$



Feedforward Comb Filter

$$y(n) = x(n) + g \cdot y(n - N)$$

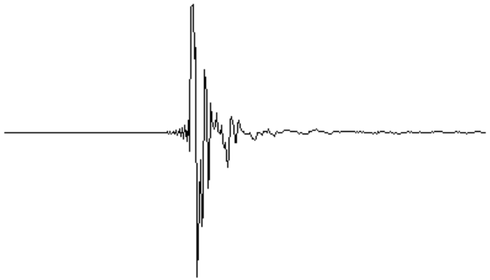


Feedback Comb Filter

Convolution Filter

- Conduct the convolution operation with an **impulse response** of a system
- The impulse response is often measured from natural objects
  - HRTF: human ears
  - Reverberation: rooms

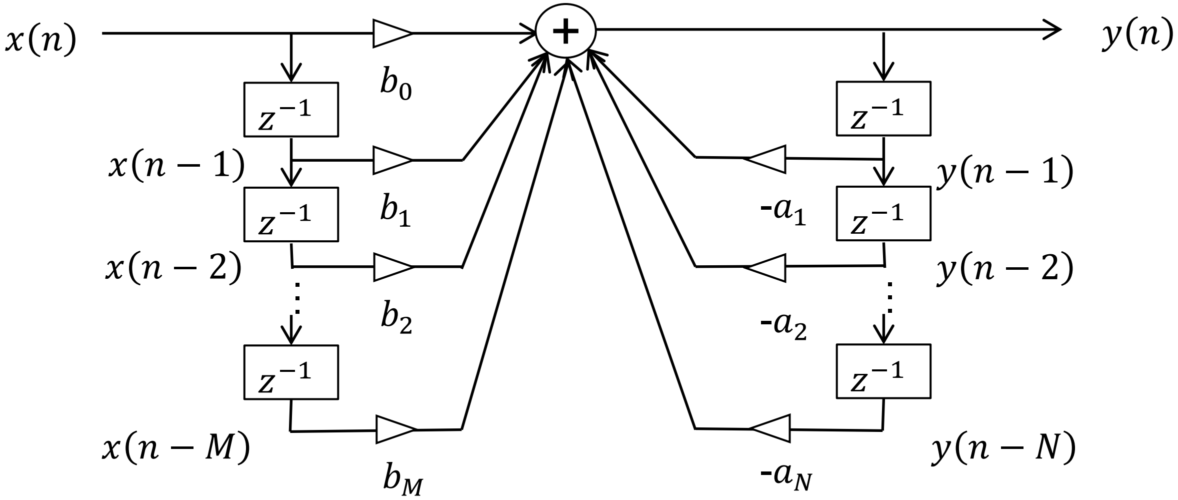
$$y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \cdots + b_M \cdot x(n-M)$$



$h(n)= [b_0, b_1, b_2 \ldots, b_M]$

General Form of Digital Filters

- Difference equation
  - $y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \ldots + b_M \cdot x(n-M)$
    $-a_1 \cdot y(n-1) - a_2 \cdot y(n-2) - \cdots - a_N \cdot y(n-N)$
- Signal flow graph

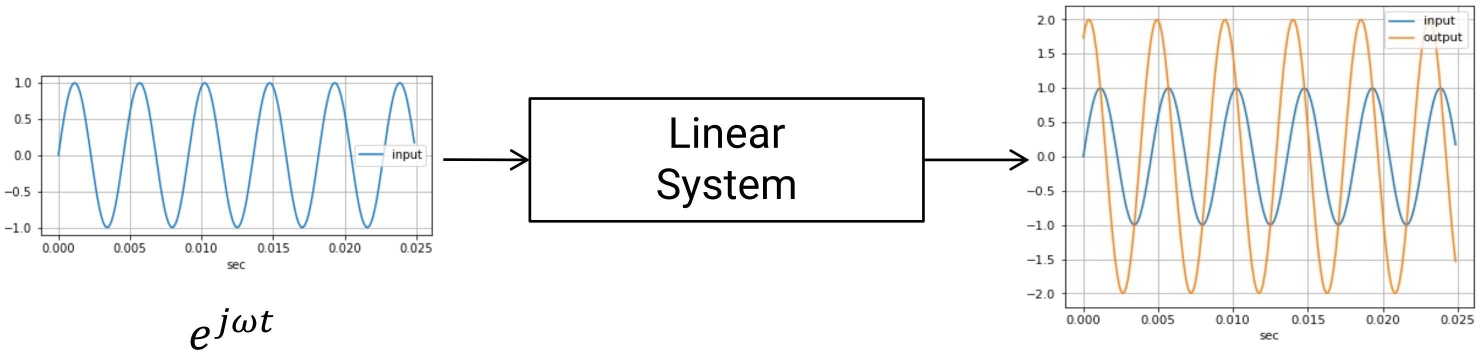


Linear Time-Invariant Filters

- They are called Linear Time-Invariant (LTI) filters in the context of digital signal processing
- Linearity
  - Scaling: if $x(n) \rightarrow y(n)$, then $\mathrm{a} \cdot x(n) \rightarrow \mathrm{a} \cdot y(n)$
  - Superposition: if $x_1(n) \rightarrow y_1(n)$ and $x_2(n) \rightarrow y_2(n)$, then $x_1(n) + x_2(n) \rightarrow y_1(n) + y_2(n)$
- Time-Invariance
  - If $x(n) \rightarrow y(n)$, then $x(n - N) \rightarrow y(n - N)$ for any N
  - This means that the system does not change its behavior over time

Linear Time-Invariant System

- Remember that sinusoids are eigenfunctions of linear system
  - The input sinusoids changes in amplitude and phase while preserving the same frequency
  - No new sinusoidal components are created



$e^{j\omega t}$

$G(\omega)e^{j\theta(\omega)}e^{j\omega t}$

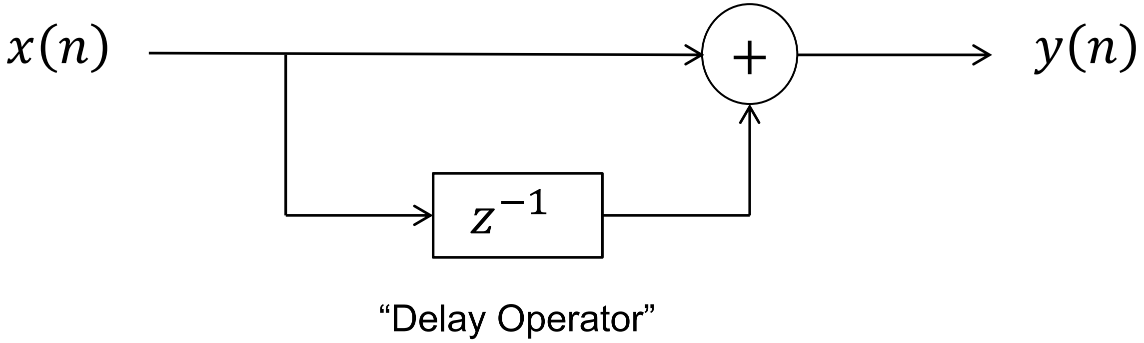
Amplitude Response

Phase Response

The Simplest Lowpass Filter

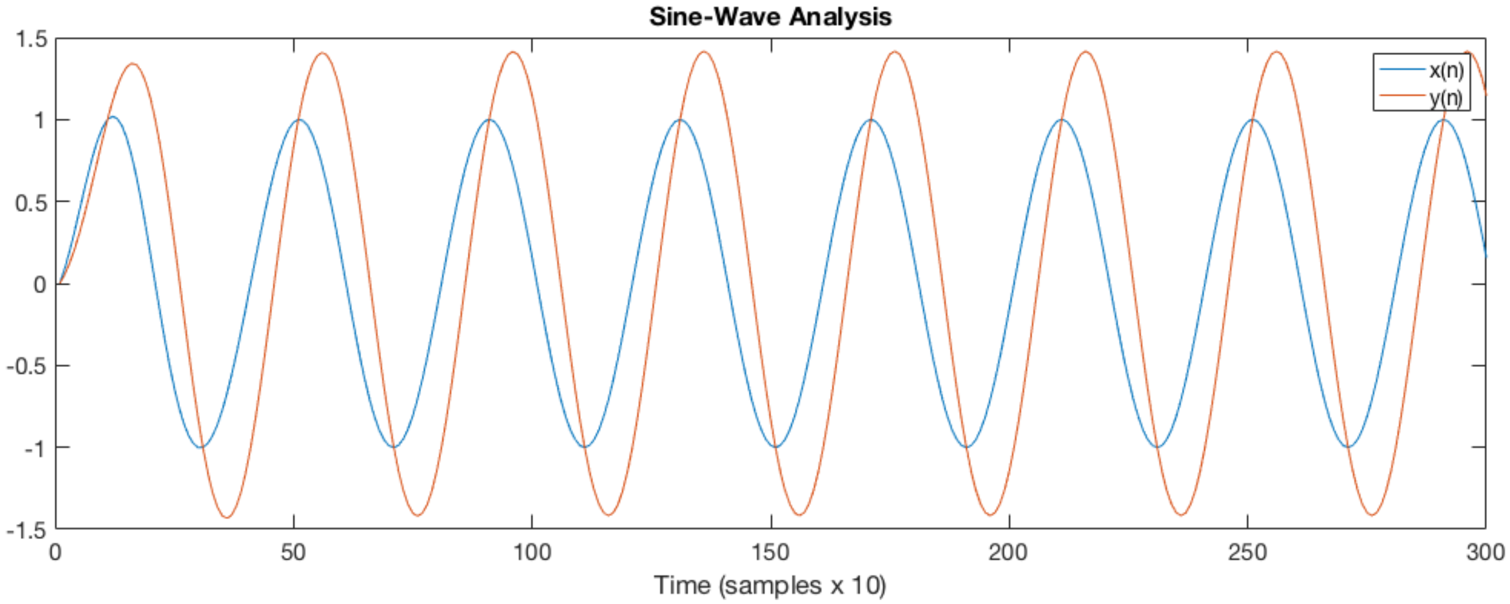
- Difference equation: $y(n) = x(n) + x(n-1)$

- Signal flow graph



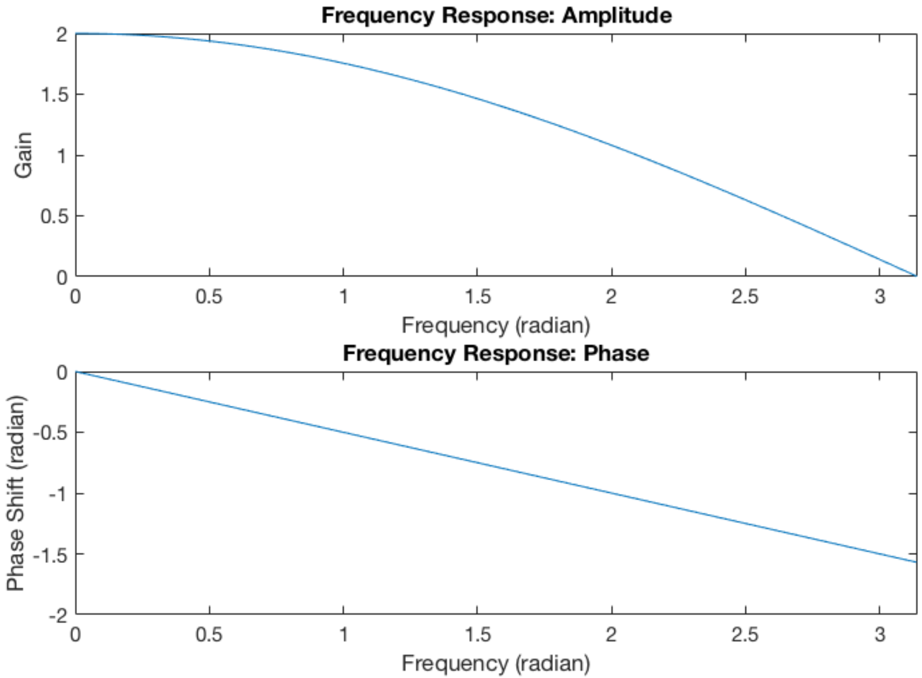
The Simplest Lowpass Filter: Sine-Wave Analysis

- Measure the amplitude and phase changes given a sinusoidal signal input



The Simplest Lowpass Filter: Frequency Response

- Plot the amplitude and phase change over different frequency
  - The frequency sweeps from 0 to the Nyquist rate



The Simplest Lowpass Filter: Frequency Response

- Mathematical approach
  - Use complex sinusoid as input: $x(n) = e^{j\omega n}$
  - Then, the output is:

    $$y(n) = x(n) + x(n-1) = e^{j\omega n} + e^{j\omega(n-1)} = \left(1 + e^{-j\omega}\right) \cdot e^{j\omega n}$$
    $$= \left(1 + e^{-j\omega}\right) \cdot x(n)$$

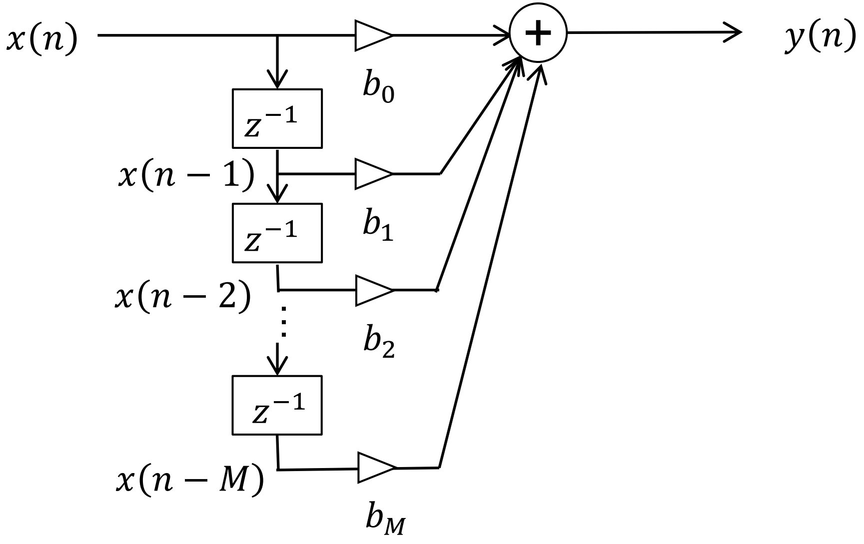
  - Frequency response: $H(\omega) = \left(1 + e^{-j\omega}\right) = \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}}\right) e^{-j\frac{\omega}{2}} = 2\cos(\frac{\omega}{2})e^{-j\frac{\omega}{2}}$
  - Amplitude response: $|H(\omega)| = 2\cos\left(\frac{\omega}{2}\right)$
  - Phase response: $\angle H(\omega) = -\frac{\omega}{2}$

Finite Impulse Response (FIR) System

- Difference equation

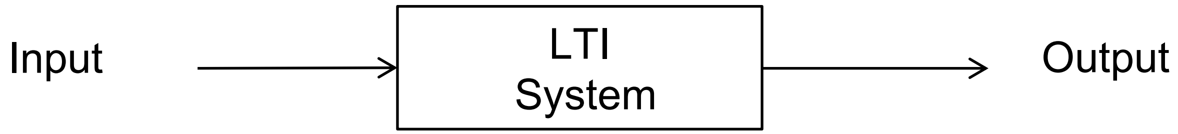
$$y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \cdots + b_M \cdot x(n-M)$$

- Signal flow graph



Impulse Response

- The system output when the input is a unit impulse
  - $x(n) = \delta(n) = [1, 0, 0, 0, \dots] \rightarrow y(n) = h(n) = [b_0, b_1, b_2 \dots, b_M]$ (for FIR system)
- Characterizes the digital system **as a sequence of numbers**
  - A system is represented just like audio samples!



$x(n) = \delta(n)$ $h(n)$ $y(n) = h(n)$

Examples: Simplest FIR filters and Moving-Average Filters

- The simplest lowpass filter
  - $\mathrm{h}(n) = [1, 1]$
- The simplest highpass filter
  - $\mathrm{h}(n) = [1, -1]$
- Moving-average filter (order=5)
  - $\mathrm{h}(n) = \left[\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right]$

Convolution

- The output of LTI digital filters is represented by the convolution operation between $x(n)$ and $h(n)$

$$y(n) = x(n) * h(n) = \sum_{i=-\infty}^{\infty} x(i) \cdot h(n-i) = \boxed{\sum_{i=-\infty}^{\infty} h(i) \cdot x(n-i)}$$

This is more practical expression
when the input is an audio streaming

- Examples
  - The simplest lowpass filter
  - $y(n) = [1, 1] * x(n) = 1 \cdot x(n) + 1 \cdot x(n-1) = x(n) + x(n-1)$

Proof: Convolution

- Method 1
  - The input is represented as the sum of weighted and delayed impulses units
    - $x(n) = [x_0, x_1, x_2 \dots, x_M] = x_0 \cdot \delta(n) + x_1 \cdot \delta(n-1) + x_2 \cdot \delta(n-2) + \cdots + x_M \cdot \delta(n-M)$
  - By the linearity and time-invariance
    - $y(n) = x_0 \cdot h(n) + x_1 \cdot h(n-1) + x_2 \cdot h(n-2) + \cdots + x_M \cdot h(n-M) = \sum_{i=0}^{M} x(i) \cdot h(n-i)$

Proof: Convolution

- Method 2
  - The impulse response can be represented as a set of weighted impulses
    - $h(n) = [b_0, b_1, b_2 \ldots, b_M] = b_0 \cdot \delta(n) + b_1 \cdot \delta(n-1) + b_2 \cdot \delta(n-2) + \cdots + b_M \cdot \delta(n-M)$
  - By the linearity, the distributive property and $x(n) * \delta(n-k) = x(n-k)$
    - $y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \cdots + b_M \cdot x(n-M) = \sum_{i=0}^{M} h(i) \cdot x(n-i)$

Properties of Convolution

- Commutative: $x(n) * h_1(n) * h_2(n) = x(n) * h_2(n) * h_1(n)$
- Associative: $(x(n) * h_1(n)) * h_2(n) = x(n) * (h_1(n) * h_2(n))$
- Distributive: $x(n) * \left(h_1(n) + h_2(n)\right) = x(n) * h_1(n) + x(n) * h_2(n)$

Example: Convolution

- Given $x(n) = [x_0, x_1, x_2, \ldots, x_5]$ and $h(n) = [h_0, h_1, h_2\ ]$
  - $y(n) = \sum_{i=0}^{M} h(i) \cdot x(n-i)$
  - $y(0) = h(0) \cdot x(0)$
  - $y(1) = h(0) \cdot x(1) + h(1) \cdot x(0)$
  - $y(2) = h(0) \cdot x(2) + h(1) \cdot x(1) + h(2) \cdot x(0)$
  - $y(3) = h(0) \cdot x(3) + h(1) \cdot x(2) + h(2) \cdot x(1)$
  - $y(4) = h(0) \cdot x(4) + h(1) \cdot x(3) + h(2) \cdot x(2)$
  - $y(5) = h(0) \cdot x(5) + h(1) \cdot x(4) + h(2) \cdot x(3)$
  - $y(6) = \qquad h(1) \cdot x(5) + h(2) \cdot x(4)$
  - $y(7) = \qquad h(2) \cdot x(5)$

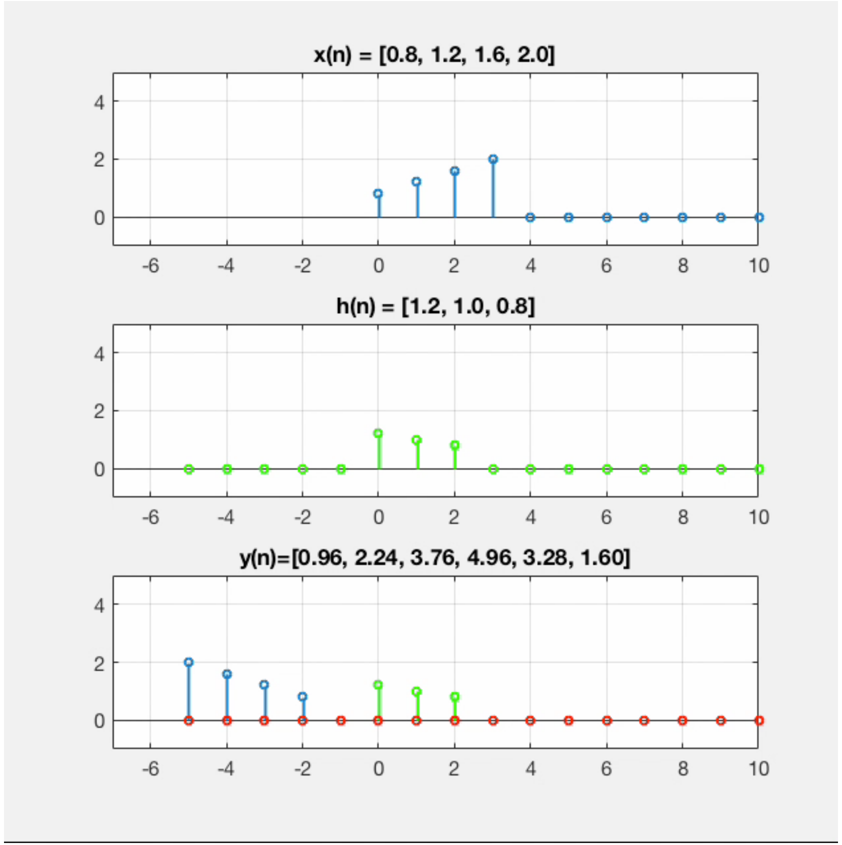
Transient state ($y(0)$–$y(1)$)

Steady state ($y(2)$–$y(5)$)

Transient state ($y(6)$–$y(7)$)

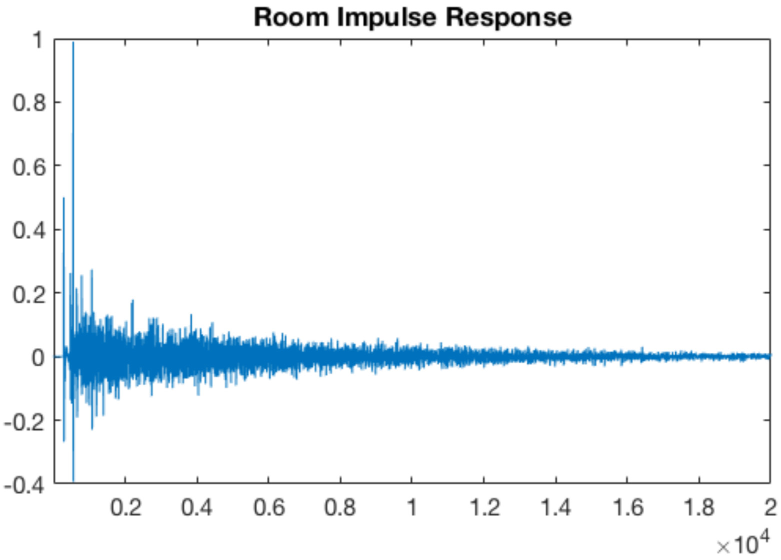
- The size of transient region is equal to the number of delay operators
- If the length of $x(n)$ is M and the length of $h(n)$ is N, then the length of $y(n)$is $M + N - 1$.

Demo: Convolution



Example: Convolution Reverb

- Convolution Reverb



Room Impulse Response

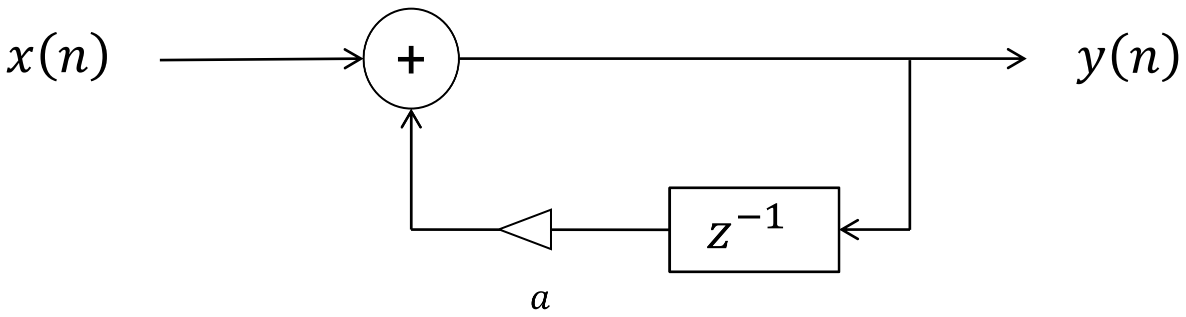
Lobby

Church

A Simple Feedback Lowpass Filter

- Difference equation: $y(n) = x(n) + a \cdot y(n-1)$

- Signal flow graph
  - When a is slightly less than 1, it is called “Leaky Integrator”



A Simple Feedback Lowpass Filter: Impulse Response

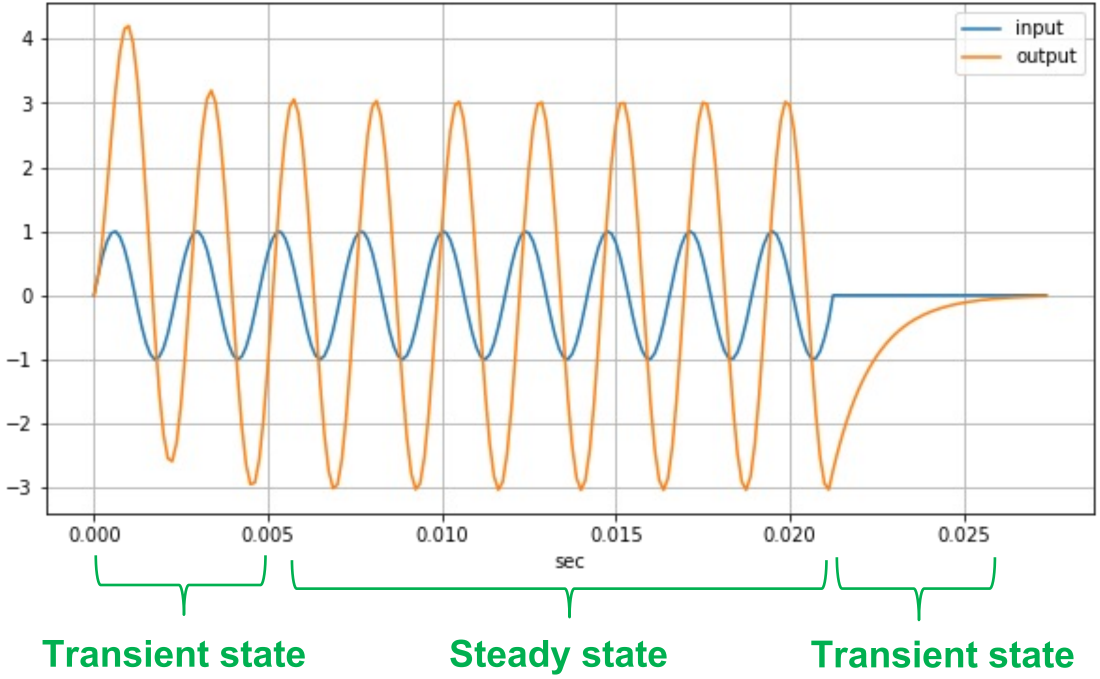
- Impulse response: exponential decays
  - $y(0) = x(0) = 1$
  - $y(1) = x(1) + a \cdot y(0) = a$
  - $y(2) = x(2) + a \cdot y(1) = a^2$
  - $y(3) = x(3) + a \cdot y(2) = a^2$
  - …

$\longrightarrow$ $h(n)$=$[1, a, a^2, a^3, \ldots]$

- **Stability Issue!**
  - If $a < 1$, the filter output converges (stable)
  - If $a = 1$, the filter output oscillates (critical)
  - If $a > 1$, the filter output diverges (unstable)

A Simple Feedback Lowpass Filter: Sine-Wave Analysis

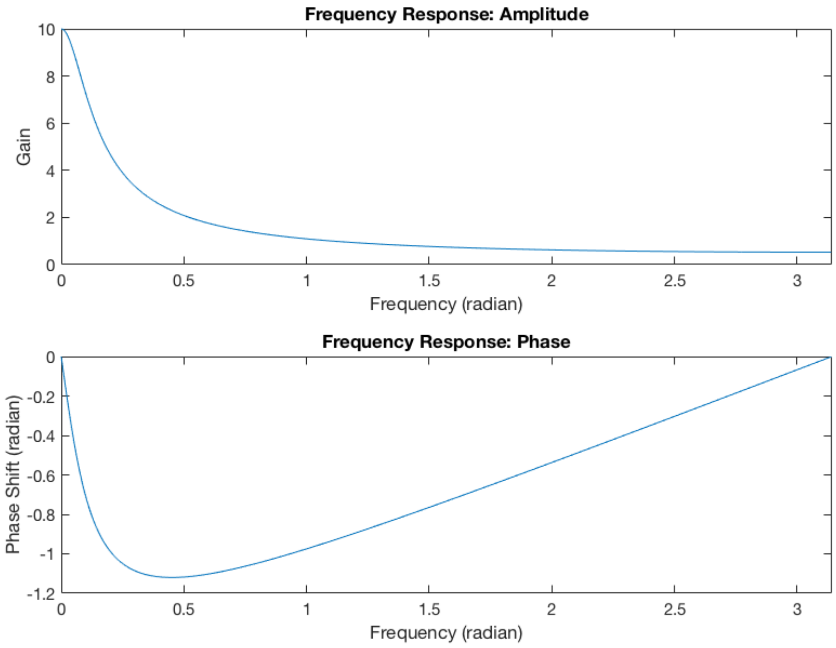
- Measure the amplitude and phase changes given a sinusoidal signal input



$a = 0.9$

A Simple Feedback Lowpass Filter: Frequency Response

- More dramatic change than the simplest lowpass filter (FIR)
  - Phase response is not linear



$$y(n) = x(n) + 0.9 \cdot y(n-1)$$

A Simple Feedback Lowpass Filter: Frequency Response

- Mathematical approach
    - Use complex sinusoid as input: $x(n) = e^{j\omega n}$
    - $y(n) = G(\omega)e^{j(\omega n+\theta(\omega))} \rightarrow y(n-m) = e^{-j\omega m}y(n)$ for any m
    - The output is: $y(n) = x(n) + a \cdot y(n-1)$

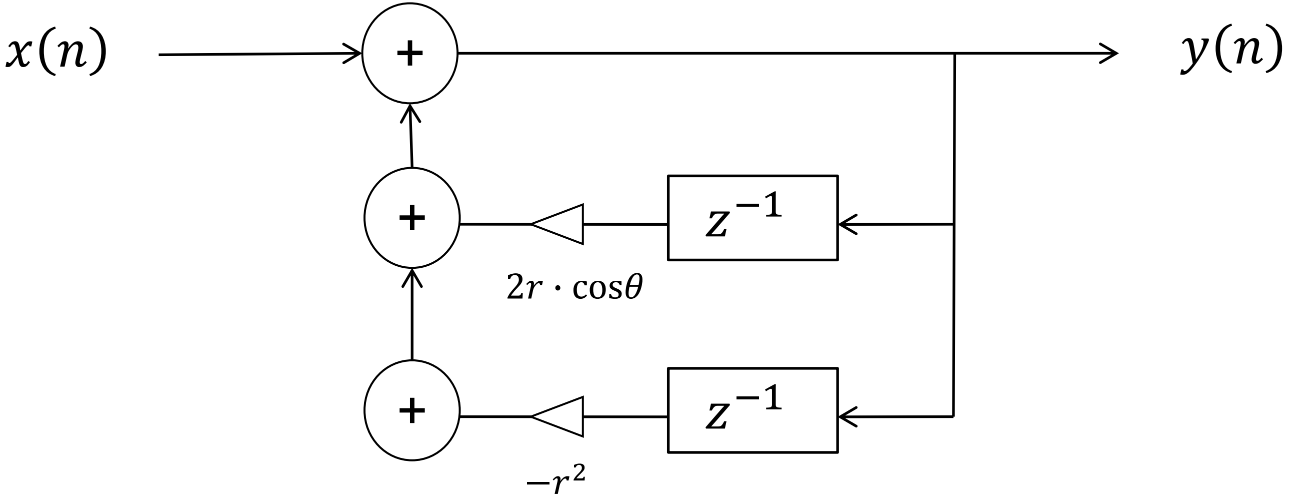
$$y(n) = x(n) + a \cdot e^{j\omega} y(n)$$

    - Frequency response: $H(\omega) = \frac{1}{(1-a \cdot e^{-j\omega})} = \frac{1}{(1-a \cdot \cos(\omega)+a \cdot j \cdot \sin(\omega))}$
    - Amplitude response: $|H(\omega)| = \frac{1}{(1-a \cdot \cos(\omega))^2+(a \cdot \sin(\omega))^2}$
    - Phase response: $\angle H(\omega) = -\operatorname{atan}(\frac{a \cdot \sin(\omega)}{1-a \cdot \cos(\omega)})$
- Note that the this approach is getting complicated

Reson Filter

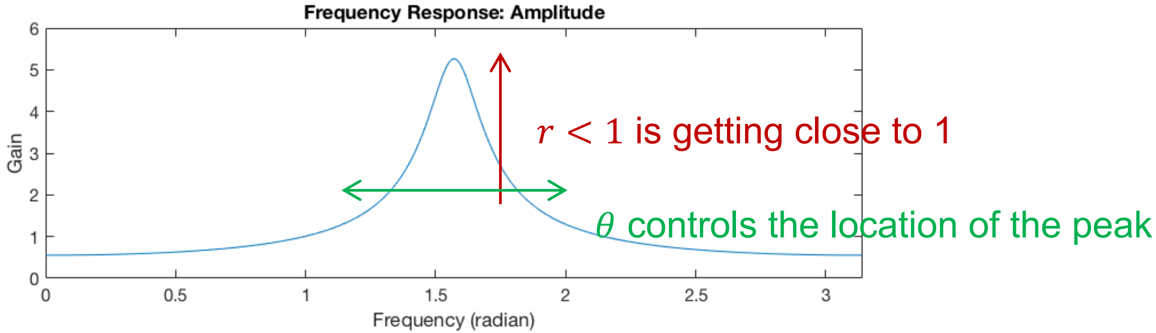
- Difference equation
  - $y(n) = x(n) + 2r \cdot \cos(\theta) \cdot y(n-1) - r^2 \cdot y(n-2)$

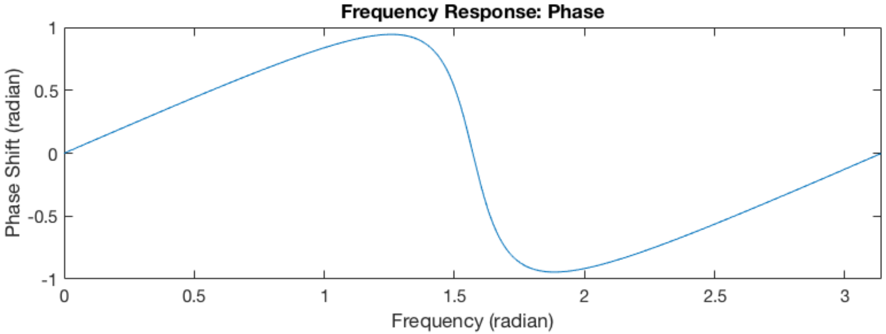
- Signal flow graph



Reson Filter: Frequency Response

- Generate resonance at a particular frequency
  - Control the peak height by r and the peak frequency by θ





For stability: $r < 1$

Reson Filter

- Mathematical approach
  - Use complex sinusoid as input: $x(n) = e^{j\omega n}$
  - $y(n) = G(\omega)e^{j(\omega n+\theta(\omega))} \rightarrow y(n-m) = e^{-jm\omega}y(n)$ for any m
  - The output is: $y(n) = x(n) + 2r \cdot \cos(\theta) \cdot y(n-1) - r^2 \cdot y(n-2)$

    $$y(n) = x(n) + 2r \cdot \cos(\theta) \cdot e^{j\omega}y(n) - r^2 \cdot e^{j2\omega}y(n-2)$$
  - Frequency response
    - $H(\omega) = \frac{1}{(1-2r\cdot\cos(\theta)\cdot e^{j\omega}+r^2\cdot e^{j2\omega})}$

      $= \frac{1}{(1-r(\cos(\theta)+j\cdot\sin(\theta))e^{j\omega})(1-r(\cos(\theta)-j\cdot\sin(\theta))e^{j\omega})}$

Amplitude response: $|H(\omega)|$?

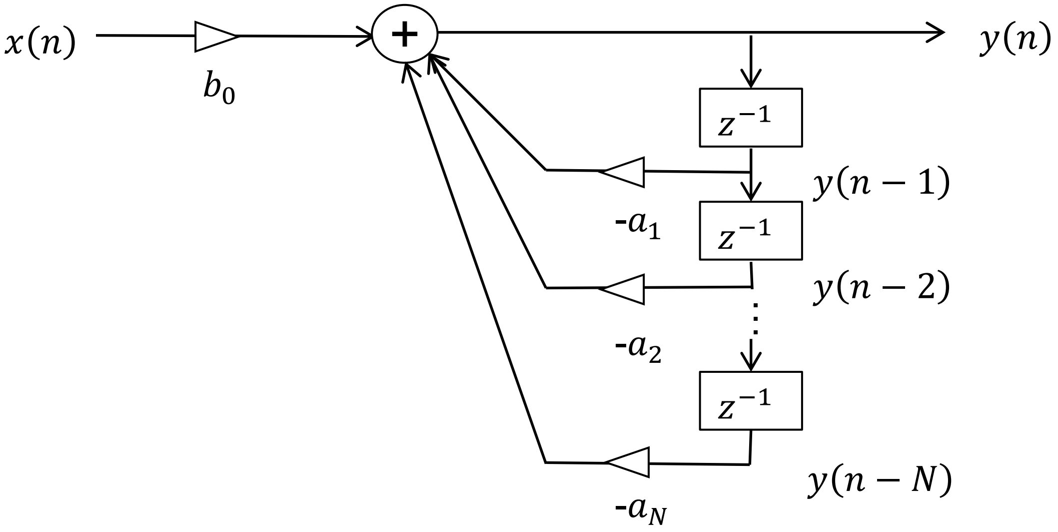
Phase response: $\angle H(\omega)$?

- Now you see that the this approach is getting even more complicated
  - We will introduce more intuitive method to obtain the frequency response

Infinite Impulse Response (IIR) Filters

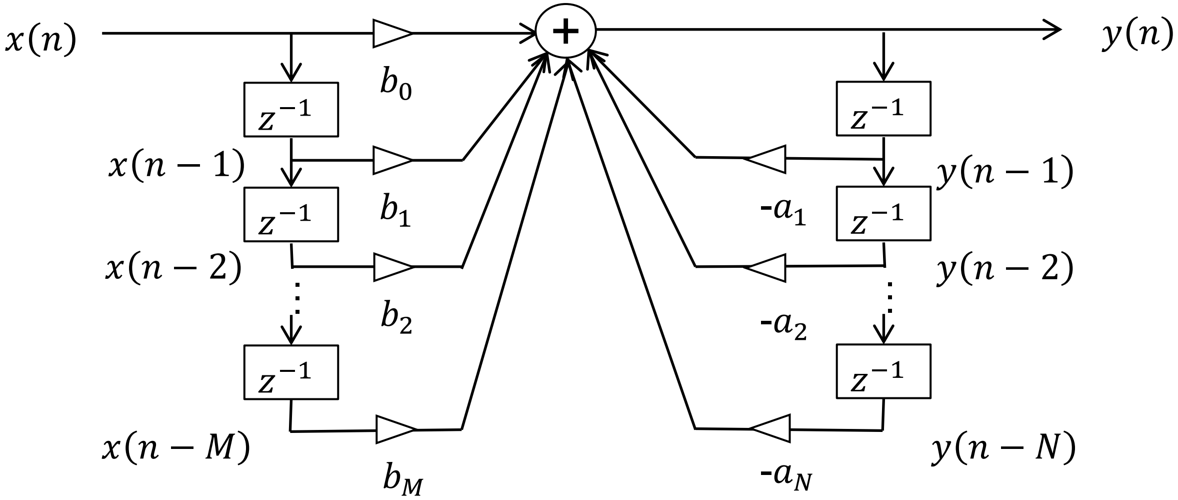
- Difference equation
  - $y(n) = b_0 \cdot x(n) - a_1 \cdot y(n-1) - a_2 \cdot y(n-2) - \cdots - a_N \cdot y(n-N)$

- Signal flow graph



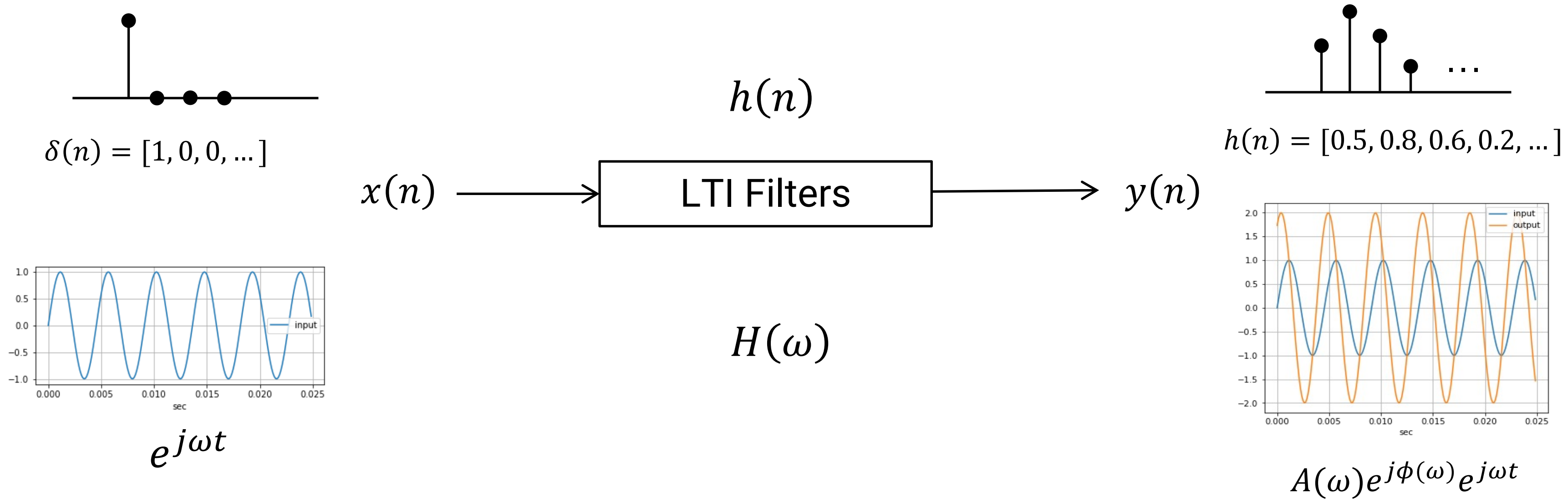
General Form of LTI Filters

- Difference equation
  - $y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) + \ldots + b_M \cdot x(n-M)$
    $-a_1 \cdot y(n-1) - a_2 \cdot y(n-2) - \cdots - a_N \cdot y(n-N)$
- Signal flow graph



LTI filters

- Characterized by
  - Impulse response (time-domain): $h(n)$
  - Frequency response (frequency-domain): $H(\omega)$



Questions

- How can we easily derive the frequency response?

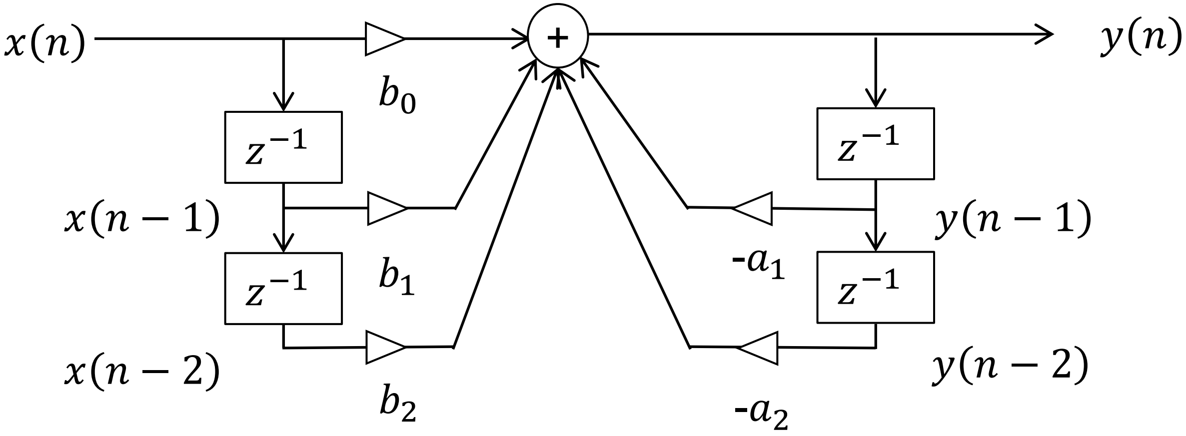
- What is the relation between the impulse response and the frequency response?

Bi-quad filter

- Difference equation

$$y(n) = b_0 \cdot x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) - a_1 \cdot y(n-1) - a_2 \cdot y(n-2)$$

- Signal flow graph



Bi-quad filter: Frequency Response

- Sine-wave analysis
  - $x(n) = e^{j\omega n} \rightarrow x(n-1) = e^{j\omega(n-1)} = e^{-j\omega}x(n),\ x(n-2) = e^{-j2\omega}x(n)$
  - $y(n) = G(\omega)e^{j(\omega n+\theta(\omega))} \rightarrow y(n-1) = G(\omega)e^{j(\omega(n-1)+\theta(\omega))} = e^{-j\omega}y(n)$, $y(n-2) = e^{-j2\omega}y(n)$
- Putting this into the different equation

$$y(n) = b_0 \cdot x(n) + b_1 \cdot e^{-j\omega} \cdot x(n) + b_2 \cdot e^{-j2\omega} \cdot x(n) - a_1 \cdot e^{-j\omega} \cdot y(n) - a_2 \cdot e^{-j2\omega} \cdot y(n)$$

$$y(n) = \frac{b_0 + b_1 \cdot e^{-j\omega} + b_2 \cdot e^{-j2\omega}}{1 + a_1 \cdot e^{-j\omega} + a_2 \cdot e^{-j2\omega}} x(n)$$

$$\boxed{H(\omega) = \frac{b_0 + b_1 \cdot e^{-j\omega} + b_2 \cdot e^{-j2\omega}}{1 + a_1 \cdot e^{-j\omega} + a_2 \cdot e^{-j2\omega}}}$$

Bi-quad filter: Amplitude and Phase Response

- Amplitude Response: $G(\omega) = |H(\omega)|$

$$G(\omega) = |H(\omega)| = \left|\frac{b_0 + b_1 \cdot e^{-j\omega} + b_2 \cdot e^{-j2\omega}}{1 + \ a_1 \cdot e^{-j\omega} + a_2 \cdot e^{-j2\omega}}\right|$$

$$= \left|\frac{b_0 + b_1 \cdot (\cos(\omega) - j\sin(\omega)) + b_2 \cdot (\cos(2\omega) - j\sin(2\omega))}{1 + \ a_1 \cdot (\cos(\omega) - j\sin(\omega)) + a_2 \cdot (\cos(2\omega) - j\sin(2\omega))}\right|$$

$$= \left|\frac{b_0 + b_1 \cdot \cos(\omega) + b_2 \cdot \cos(2\omega) - j(b_1 \cdot \sin(\omega) + b_2 \cdot \sin(2\omega))}{1 + \ a_1 \cdot \cos(\omega) + a_2 \cdot \cos(2\omega) - j(a_1 \cdot \sin(\omega) + a_2 \cdot j \sin(2\omega))}\right| = \quad \cdots$$

- Phase response: $\theta(\omega) = \angle H(\omega)$
- The analytic expression of frequency response is complicated!

Bi-quad filter: Z-Transform

- Z-transform
  - Define z as a variable in the complex plane: we call it z-plane
  - When replacing $z = e^{j\omega} = \cos(\omega) + j\sin(\omega)$ (on unit circle), the frequency response changes to the following form

$$H(\omega) = \frac{b_0 + b_1 \cdot e^{-j\omega} + b_2 \cdot e^{-j2\omega}}{1 + a_1 \cdot e^{-j\omega} + a_2 \cdot e^{-j2\omega}} \longrightarrow H(z) = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2}}$$

  - We call this **z-transform** or **transfer function** of the filter
  - "z^{-1}" corresponds to one sample delay: delay operator or delay element

Bi-quad filter: Poles and Zeros in Z-Transform

- The polynomial of z^{-1} in $H(z)$ can be factorized
  - We can find roots for both numerator and denominator
  - **Zeros**: roots of numerator
  - **Poles**: roots of denominator

$$H(z) = \frac{(1 - q_1 z^{-1})(1 - q_2 z^{-1})}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})}$$

  - Zeros and poles can be complex numbers (as a complex conjugate)
- We can analyze frequency response more easily using poles and zeros than numerator or denominator coefficient

Bi-quad filter: Pole-Zero Analysis: Amplitude Response

- Amplitude Response
  - Computed using distances between poles and unit circles and distances between zeros and units circles on Z-plane

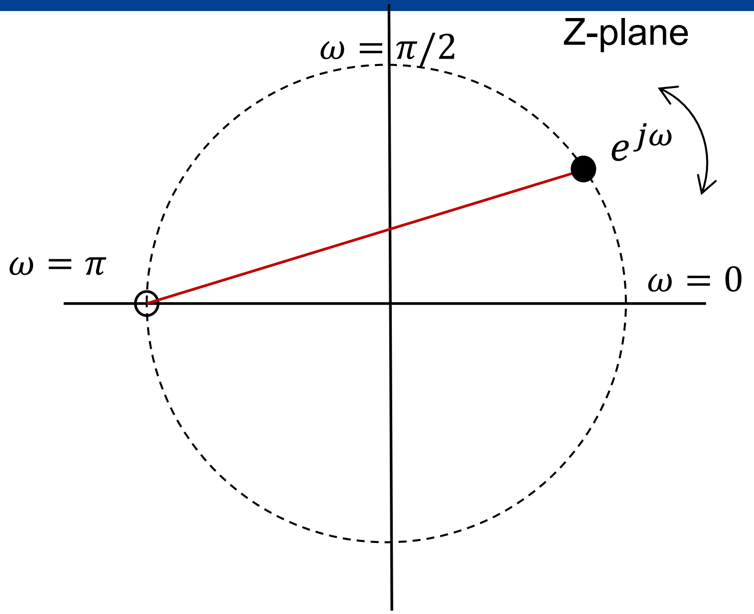
$$G(\omega) = |H(\omega)| = \left|\frac{\left(1 - q_1 e^{-j\omega}\right)\left(1 - q_2 e^{-j\omega}\right)}{\left(1 - p_1 e^{-j\omega}\right)\left(1 - p_2 e^{-j\omega}\right)}\right|$$

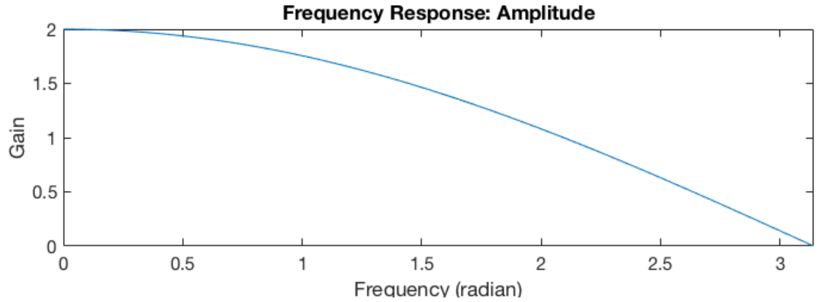
$$= \left|\frac{\left(e^{j\omega} - q_1\right)\left(e^{j\omega} - q_2\right)}{\left(e^{j\omega} - p_1\right)\left(e^{j\omega} - p_2\right)}\right|$$

$$= \frac{\left|e^{j\omega} - q_1\right|\left|e^{j\omega} - q_2\right|}{\left|e^{j\omega} - p_1\right|\left|e^{j\omega} - p_2\right|}$$

Example: Simplest lowpass filter

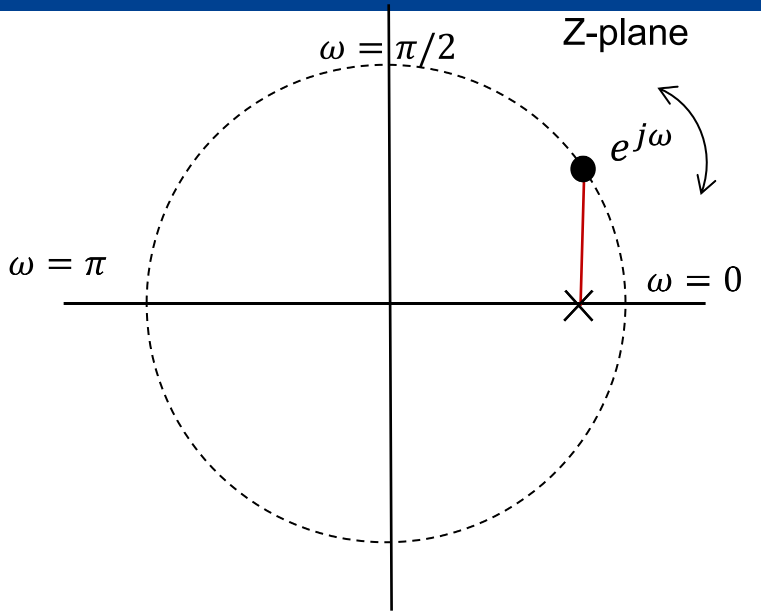
- Transfer function
  - $H(z) = 1 + z^{-1}$
  - Zeros: $q_1 = -1$ (no pole)
- Amplitude Response
  - $G(\omega) = |H(\omega)| = |e^{j\omega} + 1|$

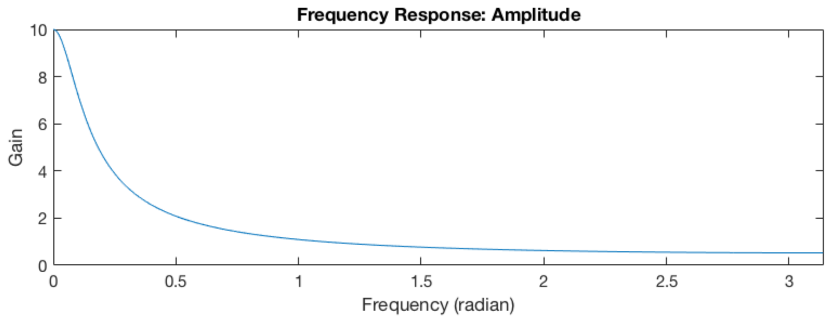




Example: Simple feedback lowpass filter

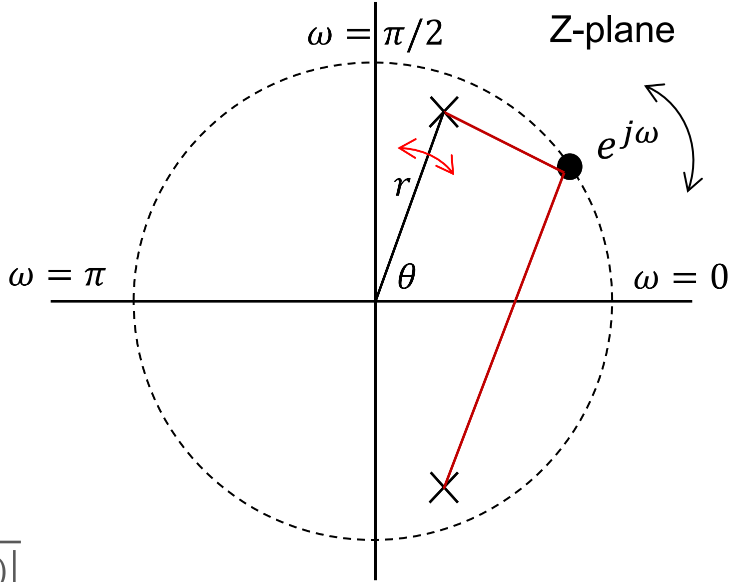
- Transfer function
  - $H(z) = H(z) = \frac{1}{1-0.9 \cdot z^{-1}}$
  - Poles: $p_1 = 0.9$ (no zero)
- Amplitude Response
  - $G(\omega) = |H(\omega)| = \frac{1}{|e^{j\omega}-0.9|}$

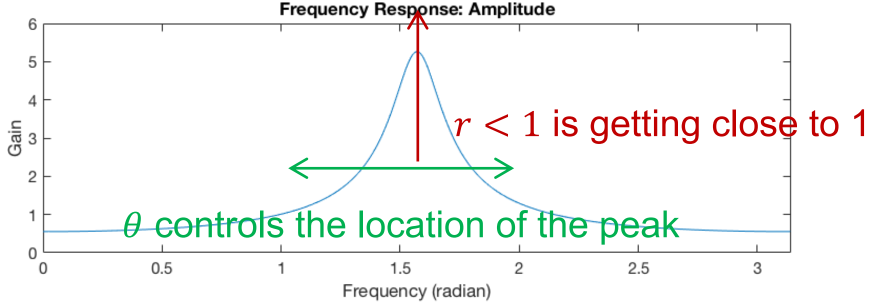




Example: Reson Filter

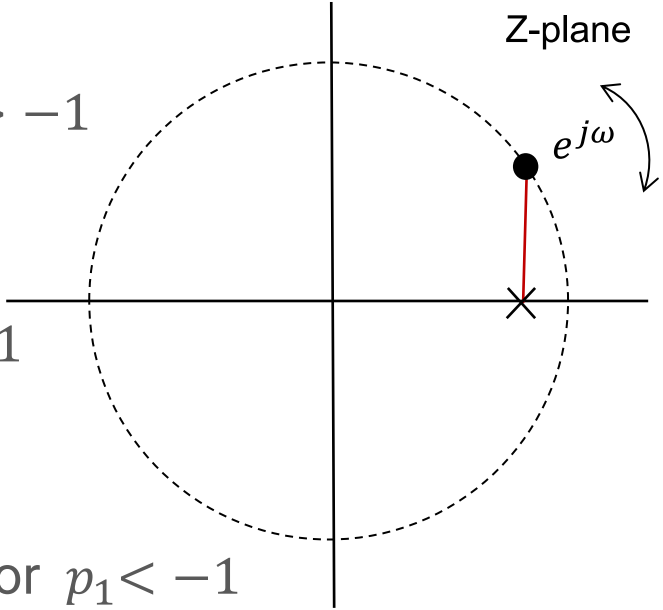
- Transfer function
  - $H(z) = \frac{1}{1-2r\cos\theta \cdot z^{-1}+r^2 \cdot z^{-2}}$
  - Poles: $r(\cos\theta + j\sin\theta), r(\cos\theta - j\sin\theta)$ (no zero)
- Amplitude response
  - $G(\omega) = \frac{1}{\left|e^{j\omega}-(\cos\theta+j\sin\theta)\right| \cdot \left|e^{j\omega}-(\cos\theta-j\sin\theta)\right|}$



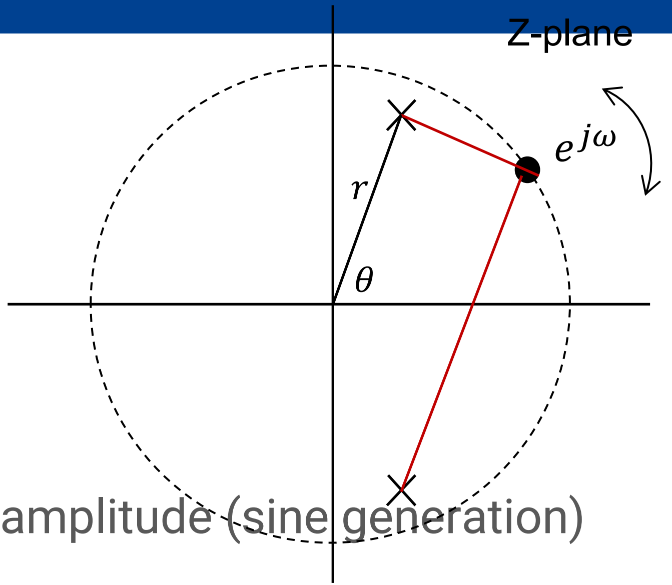


Poles and Stability: one-pole filter

- If poles are inside the unit circle: $p_1 < 1$ or $p_1 > -1$
  - The system is stable
  - The impulse response decays
- If poles are on the unit circle: $p_1 = 1$ or $p_1 = -1$
  - The system is critically-stable
  - The impulse response is constant or oscillates
- If poles are are outside the unit circle : $p_1 > 1$ or $p_1 < -1$
  - The system is unstable
  - The impulse response diverges



Poles and Stability: Reson filter



- If poles are inside the unit circle: $r < 1$
  - The system is stable
  - The impulse response decays with oscillation
- If poles are on the unit circle: $r = 1$
  - The system is critically-stable
  - The impulse response oscillates with constant amplitude (sine generation)
- If poles are outside the unit circle: $r > 1$
  - The system is unstable
  - The impulse response diverges

Bi-quad filter: Pole-Zero Analysis: Phase Response

- Phase Response
  - Computed using angles between poles and unit circles and angles between zeros and units circles on Z-plane

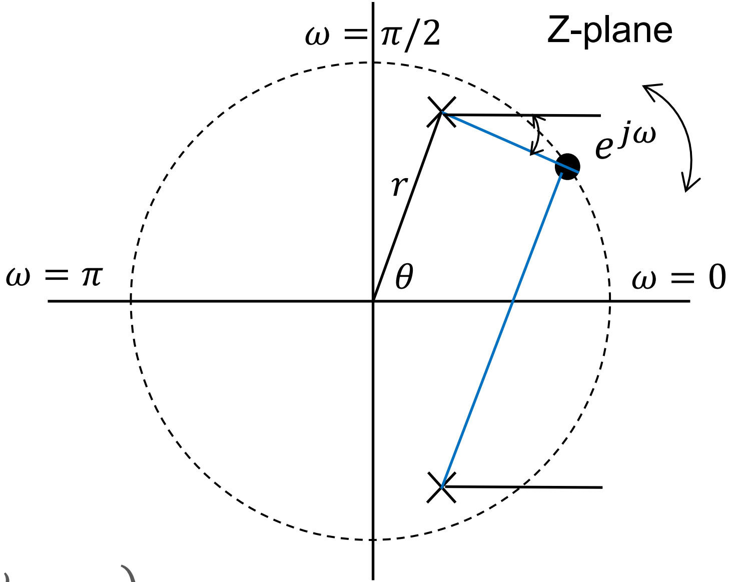
$$\theta(\omega) = \angle H(\omega) = \frac{\angle\left(1 - q_1 e^{-j\omega}\right)\left(1 - q_2 e^{-j\omega}\right)}{\angle\left(1 - p_1 e^{-j\omega}\right)\left(1 - p_2 e^{-j\omega}\right)}$$

$$= \angle\left(1 - q_1 e^{-j\omega}\right) + \angle\left(1 - q_2 e^{-j\omega}\right) - \angle\left(1 - p_1 e^{-j\omega}\right) - \angle\left(1 - p_2 e^{-j\omega}\right)$$

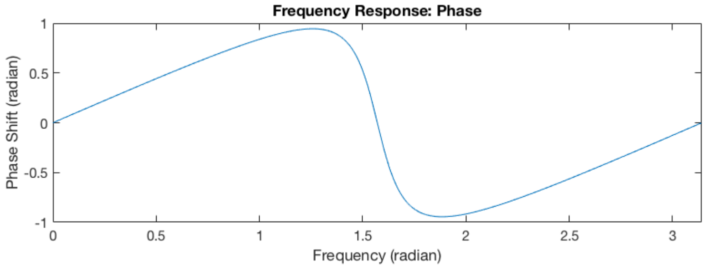
$$= \angle\left(e^{j\omega} - q_1\right) + \angle\left(e^{j\omega} - q_2\right) - \angle\left(e^{j\omega} - p_1\right) - \angle\left(e^{j\omega} - p_2\right)$$

Example: Reson Filter

- Transfer function
  - $H(z) = \frac{1}{1-2r\cos\theta \cdot z^{-1} + r^2 \cdot z^{-2}}$
  - Poles: $p_1 = r(\cos\theta + j\sin\theta)$
    $p_2 = r(\cos\theta - j\sin\theta)$



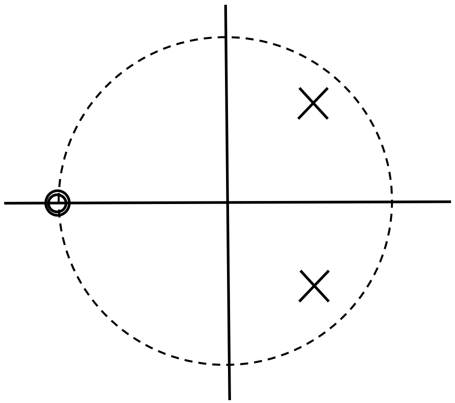
- Phase response
  - $\theta(\omega) = \angle H(\omega) = -\angle(e^{j\omega} - p_1) - \angle(e^{j\omega} - p_2)$



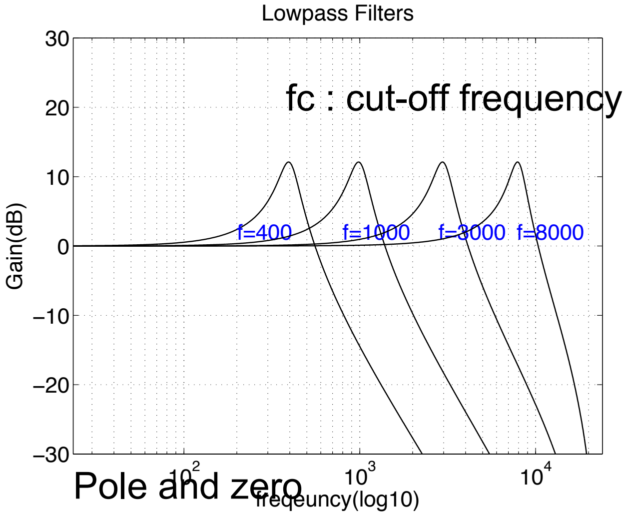
Digitized Resonant Low-pass Filter

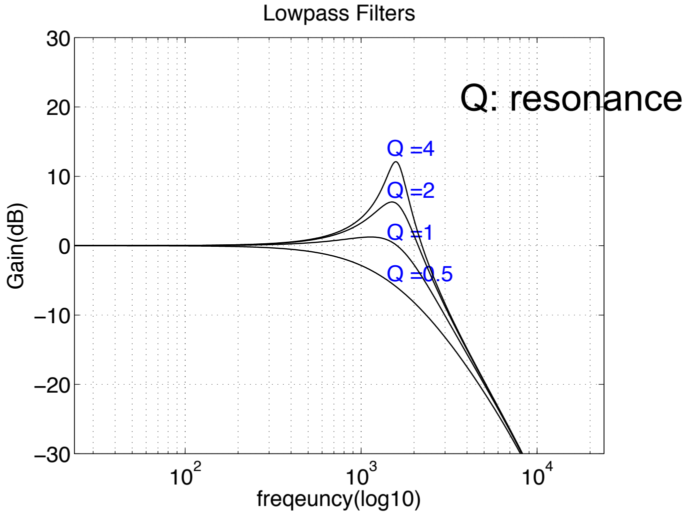
- Transfer Function

$$H(z) = \left(\frac{1-\cos\theta}{2}\right)\frac{1 + 2z^{-1} + z^{-2}}{(1+\alpha) - 2\cos\theta z^{-1} + (1-\alpha)z^{-2}} \qquad \alpha = \frac{\sin\theta}{2Q} \qquad \theta = 2\pi\frac{f_c}{f_s}$$



Pole and zero

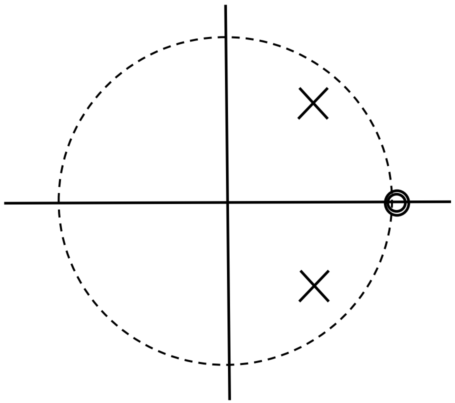


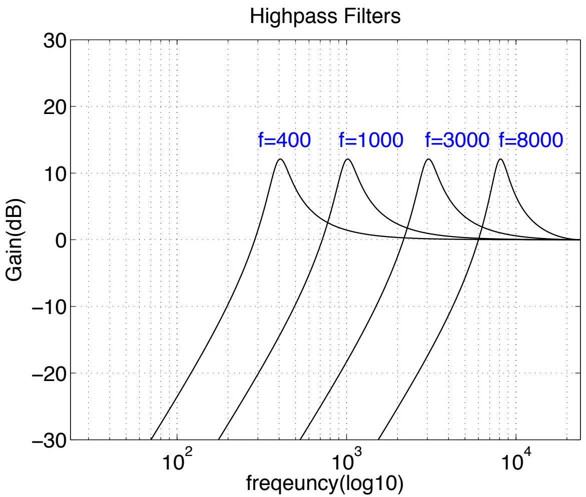


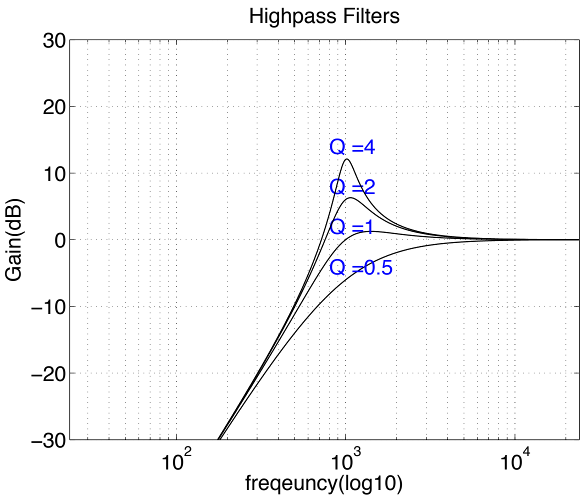
Digitized Resonant High-pass Filter

- Transfer Function

$$H(z) = \left(\frac{1+\cos\theta}{2}\right)\frac{1 - 2z^{-1} + z^{-2}}{(1+\alpha) - 2\cos\theta z^{-1} + (1-\alpha)z^{-2}} \qquad \alpha = \frac{\sin\theta}{2Q} \qquad \theta = 2\pi\frac{f_c}{f_s}$$



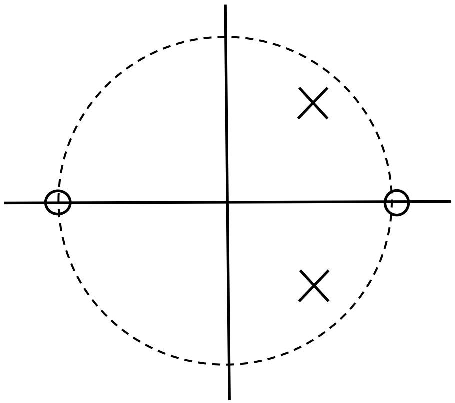


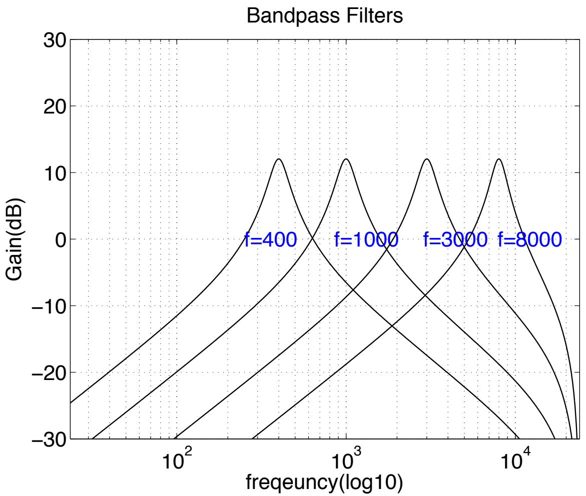


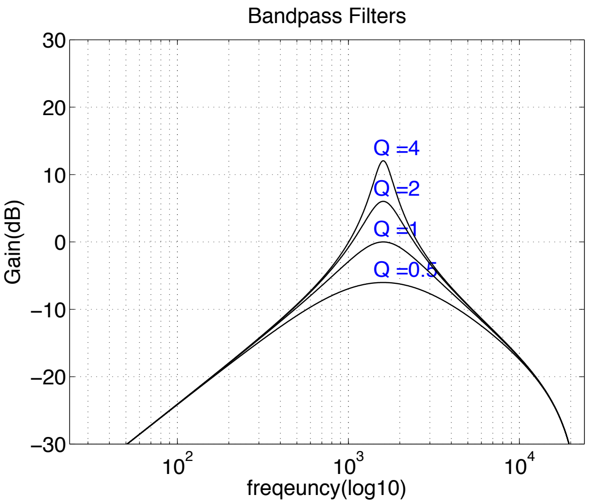
Digitized Band-pass filter

- Transfer Function

$$H(z) = \left(\frac{\sin\theta}{2Q}\right)\frac{1 - z^{-2}}{(1+\alpha) - 2\cos\theta z^{-1} + (1-\alpha)z^{-2}} \qquad \alpha = \frac{\sin\theta}{2Q} \qquad \theta = 2\pi\frac{f_c}{f_s}$$



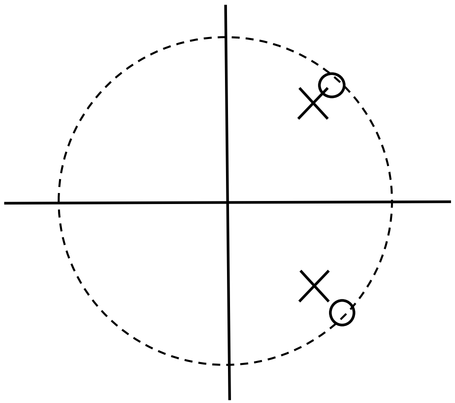


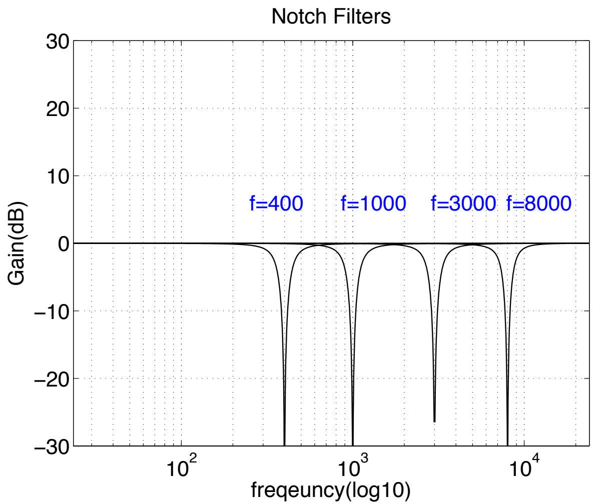


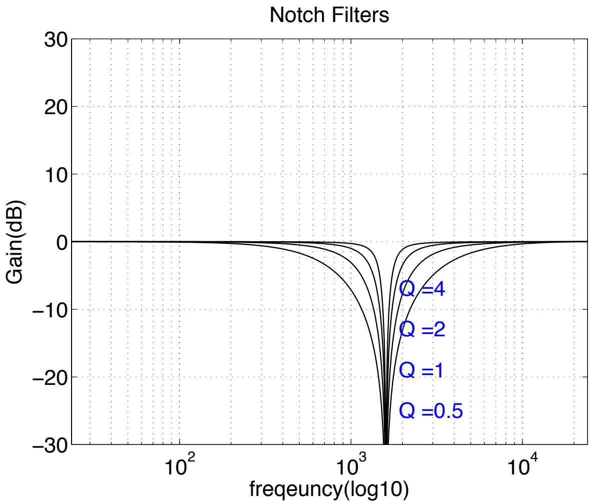
Digitized Notch filter

- Transfer Function

$$H(z) = \frac{1 - 2\cos\theta z^{-1} + z^{-2}}{(1+\alpha) - 2\cos\theta z^{-1} + (1-\alpha) z^{-2}} \qquad \alpha = \frac{\sin\theta}{2Q} \qquad \theta = 2\pi\frac{f_c}{f_s}$$



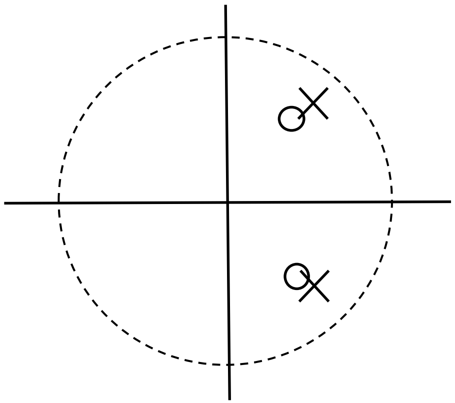


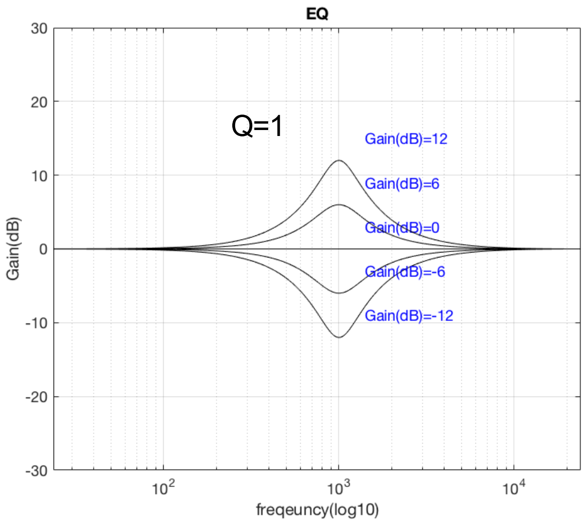


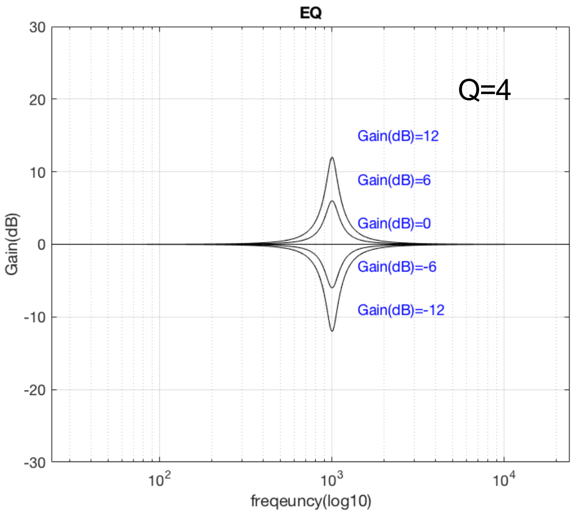
Digitized Equalizer

- Transfer Function

$$H(z) = \frac{(1 + \alpha \cdot A) - 2\cos\theta z^{-1} + (1 - \alpha \cdot A)z^{-2}}{(1 + \alpha / A) - 2\cos\theta z^{-1} + (1 - \alpha / A)z^{-2}} \qquad \alpha = \frac{\sin\theta}{2Q} \qquad \theta = 2\pi\frac{f_c}{f_s} \qquad A = 10^{\left(\frac{\mathrm{Gain(dB)}}{40}\right)}$$







Demo: Pole-Zero Analysis

- https://www.earlevel.com/main/2013/10/28/pole-zero-placement-v2/