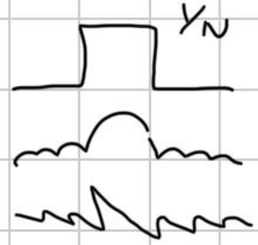
DSP : IIR Filters (Dan's PPT slide Deck) (02/15/2023, Wednesday)

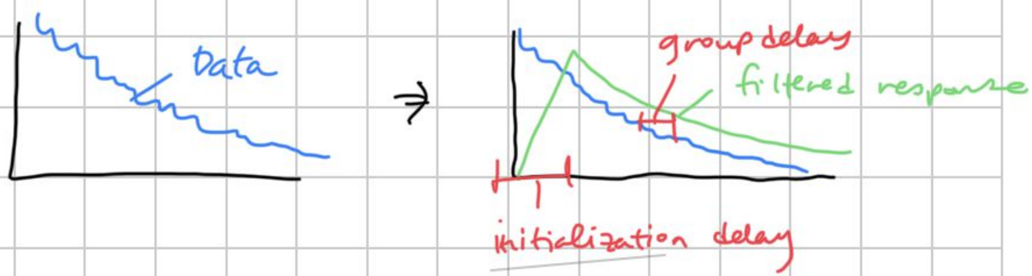
Start w/ Discrete Time (D.T.) Moving Average (FIR)

rectangular-like in time

sinc-like in frequency (low pass filter)



moving average smooths out your data curve



some problems : ① group delays (larger filter order N

$\Rightarrow$ longer group delay $= \frac{N-1}{2}$)

② initialization delay (also dependent on filter order N)

③ all values weighted equally (bad if your data changes frequently)

in a time sensitive system, large group delay can be disastrous

(ie. landing a plane,

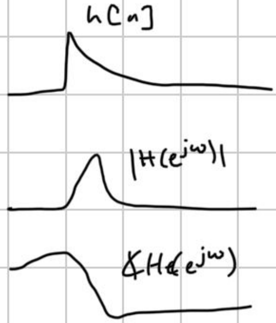
procedure to shut down refinery,

safety mechanisms ...)

what about alternatives?

Exponential Moving Avg.

system exponential in time



⇒ no longer linear group delay!!

also smoothes out data (not as well)
but has SHORTER delay!

but! is this feasible? ⇒ non-linear phase

$$h[n] = 0.9^n u[n]$$
$$y[n] = x[n] + 0.9\, y[n-1]$$

$$Y(z) = X(z) + 0.9\, Y(z) z^{-1} \Rightarrow H(z) = \frac{1}{1-0.9z^{-1}}$$

can we go $H(z) \to H(e^{j\omega})$?
Yes! (check if $|z| < 1$)

{so we can obtain $H(z) \to H(e^{j\omega})$

so, this is a valid filter!

★ but, it is IIR filter (b/c $H(z)$, $H(e^{j\omega})$, $h[n]$ depends on current / previous INPUT & previous OUTPUT!!)

(FIR only depend only current / previous INPUTS)

General Form of IIR Filter

$$y[n] = \sum_{i=0}^{N} b_i \cdot x[n-i] - \sum_{i=1}^{N} a_i \cdot y[n-i]$$

coefficients for current & previous inputs

current & previous inputs

coefficients for previous outputs

previous outputs

$$H(z) = \frac{1 + \sum_{i=0}^{N} b_i \cdot z^{-i}}{1 - \sum_{i=1}^{N} a_i \cdot z^{-i}}$$

check slides for exact/correct notation!!

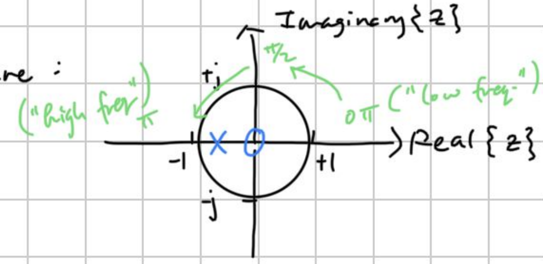
1st Order IIR Filters (demonstration @ 10:50 is AM)

$$H(z) = \frac{1}{1+0.9z^{-1}}$$

zero : @ $z_0 = 0$

pole : @ $p_0 = -0.9$

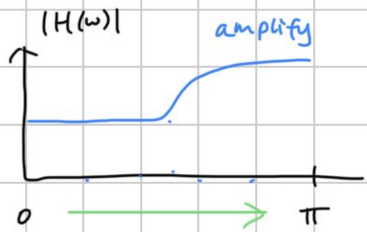
z-plane :



[0π or 2π ⇒ low frequencies

π ⇒ high frequencies

magnitude $|H(\omega)|$



⇒ no need to plot 2nd half ($\pi \to 2\pi$ freqs.)

if z-plane is $\boxed{\text{symmetric}}$ about x-axis.

$\boxed{\text{zeros:}}$ will attenuate nearby freqs.

$\boxed{\text{poles:}}$ will amplify nearby freqs.

⇒ ★ the closer a pole / zero is to the unit circle, the more effect the pole / zero has on nearby frequencies (pole amplify more, zero attenuate more)

⇒ ★ the further away a pole/zero is to unit circle, the less effect it has on nearby frequencies (pole amplify less, zero attenuate less)

what if we have some H(z) that is NOT or MARGINALLY stable?

⇒ we would NOT call H(z) a filter, but we might say, for example, this H(z) has behaviour of low pass filter,

⇒ @ 10:50 - 11 AM, highly recommend to watch demonstration of Designing different kinds of IIR filters. (1st order)

☆ (note: 1st order IIR filter ⇒ you have 1 pole, 1 zero to design your filters)

⇒ @ 11 AM, highly recommend to watch demo on 2nd Order Filter (IIR)

Second Order Biquad

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_N z^{-N}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} = \cdots$$ } Direct Form

$$H(z) = \underbrace{H_1(z) \cdot H_2(z) \cdot \cdots \cdot H_{N/2}(z)}_{\text{cascading biquads}}$$ ⇒ cascade of biquads form

complex conjugates

$$\text{where } H_i(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} = C \frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)}$$ } a biquad

complex conjugates

same filter, same filter order. why do Direct Form or Cascade of Biquads?

⇒ Direct form is easier to implement

⇒ Biquads have better stability in terms of magnitude response as IIR filter order $N \to \infty$

↳ we can leverage complex conjugate symmetry in each biquad.

⇒ in each biquad, it is made of complex conjugate pairs of zeros & poles

↳ nice because your z-plane is symmetric. (easy to see/analyze design of filter)

(break @ 11:12 AM)

⇒ return @ 11:18 AM

@ 11:18) demo of IIR filter implementation (vocoder — "voice encoder")

@ 11:27) discussion of Chromagram application of IIR