7. Filter Design

7.1.1. FIR versus IIR

IIR: discrete-time version of analog filters

FIR: flagship of digital signal processing

FIR:

- advantages:
  - unconditional stability
  - precise control of phase
  - exact linear phase
  - optimal algorithmic design
  - robust w.r.t. finite numeric precision
- disadvantages:
  - longer input-output delay
  - higher computational cost (compared to IIR)
  - may "sound" "harsh"

IIR:

- advantages:
  - lower computational cost (compared to FIR)
  - shorter input-output delay
  - compact representation
  - good for audio
- disadvantages:
  - stability not guaranteed
  - phase response difficult to control
  - complex desig (in general)
  - sensitive to numerical precision

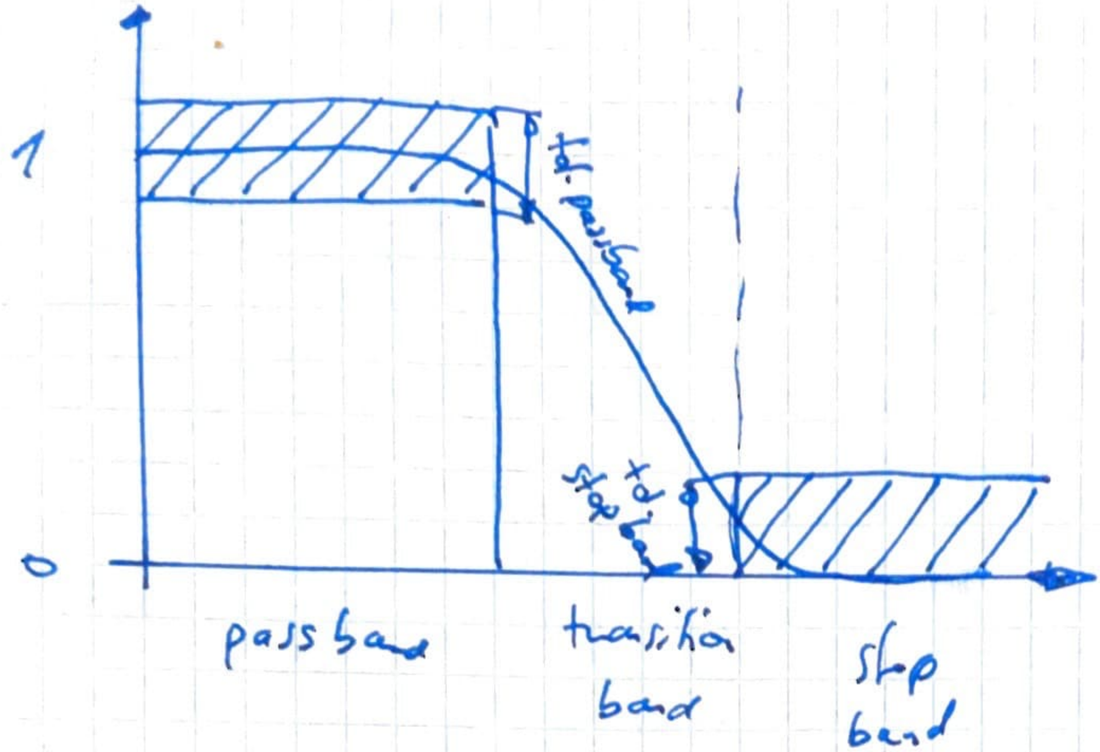
7.1.2. Filter Specifications and Tradeoffs

Generally formulated in frequency domain
- boundaries for magnitude of frequency response
- sometimes also taking phase response into account

Issues:

— Transition band: range of frequencies between passband and stopband
↳ cannot obtain instanious transition
↳ gap between passband and stopband
- multiple pass-/stop-bands possible

— Tolerances: cannot impose strict value of 1 for passband and 0 for stopband
↳ allow tolerances for passband and stopband
- transition band usually not relevant → no tolerances



· Real valued filter coefficients:
sufficient to specify frequency response over $[0, \pi]$

· magnitude response: symmetric

· Filter design problem:

· find minimum size FIR or IIR which fulfills required specification

↳ find N, M, a_k's and b_k's:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_{M-1} z^{-(M-1)}}{a_0 + a_1 z^{-1} + \dots + a_{N-1} z^{-(N-1)}}$$

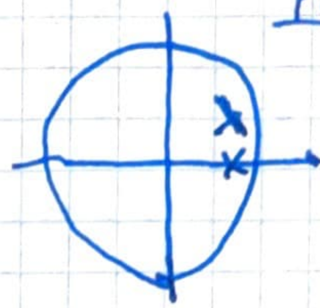
↳ hard nonlinear problem

DSP Coursera II — 17.5.24

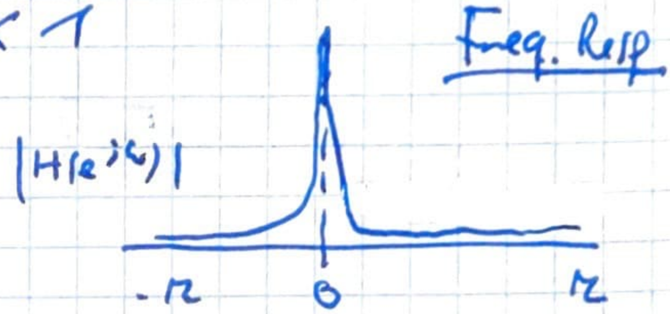
Intuitive IIR Designs (video)

- simple (low order) filters

Leaky integrator

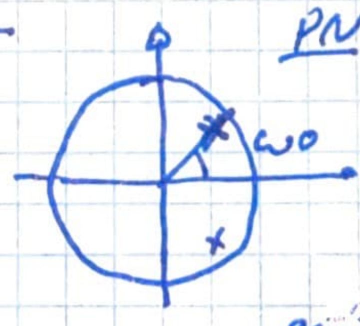


- low pass
- $H(z) = \dfrac{(1-\lambda)}{1-\lambda z^{-1}}$
- $y[n] = (1-\lambda)x[n] + \lambda y[n-1]$
- stable if $\lambda < 1$



Resonator

- narrow band pass, detect sin. of given freq.
- shift passband of leaky integrator
- $H(z) = \dfrac{G_0}{(1-pz^{-1})(1-p^*z^{-1})}$
- complex conjugate poles for real signals!



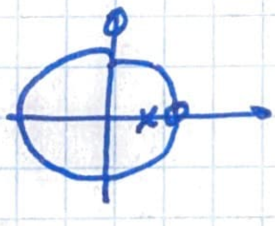
$p = \lambda e^{j\omega_0}$

$y[n] = G_0 x[n] - a_1 y[n-1] - a_2 y[n-2]$ $\quad a_1 = 2\lambda\cos\omega_0$, $a_2 = -|\lambda|^2$

DC Removal

- Remove data at $\omega_0 = 0$
- DTFT of DC balanced signal: 0
- Place zero at $1+0j$

- $H(z) = 1 - z^{-1}$
- $y[n] = x[n] - x[n-1]$
- DC Notch (make passband more narrow)
  - ↳ add leaky integr. (denominator)
  - ↳ $H(z) = \dfrac{1-z^{-1}}{1-\lambda z^{-1}}$
  - $y[n] = \lambda y[n-1] + x[n] - x[n-1]$



DSP Concordia II 17.5.24

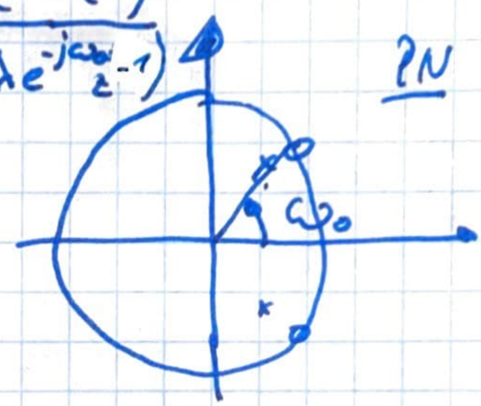
Hum Removal

- similar to DC removal, but at specific freq.
- useful for musicians:

  remove AC freq of mains (· 50Hz Europe
  · 60Hz USA)

$$H(z) = \frac{(1-e^{j\omega_0}z^{-1})(1-e^{-j\omega_0}z^{-1})}{(1-\lambda e^{j\omega_0}z^{-1})(1-\lambda e^{-j\omega_0}z^{-1})}$$

<ins>PN</ins>



DSP Coursera II 17.5.24

IIR Filter design (videos)

- design originates in analogue filters
  → translated to DSP
- Use numerical packages (Matlab, scipy...)
- design process
  - specify parameters
  - run function (from numerical packages)
  - evaluate if specs are fullfilled, else tweak and repeat
- next sections focus on lowpass → similar for other filters

Butterworth lowpass

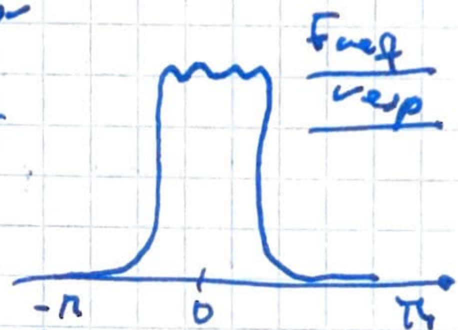
- Magnitude response
  - maximally flat
  - monotonic over $[0, \pi]$
- Design parameters:
  - order: N
  - cutoff frequency

Freq Resp

- Test values:
  - width of transition band
  - passband error

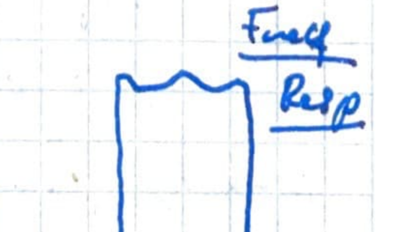
Chebyshev lowpass

- Magnitude response
  - equiripple in passband
  - monotonic in stopband
- Design params
  - order: N
  - passband max error
  - cutoff frequency



- Test values
  - width of transition band
  - stopband error

Elliptic lowpass



- Magnitude response
  - equiripple in passband and stopband
- Design params:
  - Order: N
  - cutoff frequency
  - passband max error
  - stopband min attenuation

Test values:

- width of transition band

DSP Coursera II 17.5.24

FIR: optimal minimax design

- FIR filters are DSP "exclusivity" (they don't exist in analogue form)
  - lineare phase
  - equiripple error in passband and stopband

- Algorithm to design optimal FIR filters
  - Parks and McClellan (70's)
  - minimize maximum error in pass- and stopband

- linear phase:
  - impulse response either symmetric or antisymmetric

impulse resp.

	odd length	even length
symmetric	type I	type II
antisymmetric	type III	type IV

· Linear phase (type I)

$$h[C+n] = h[C-n]$$

$$h'[n] = h[n+C]$$ (centre at 0)

↳ $$h'[n] = h'[-n]$$

$$H(z) = z^{-C} H'(z)$$

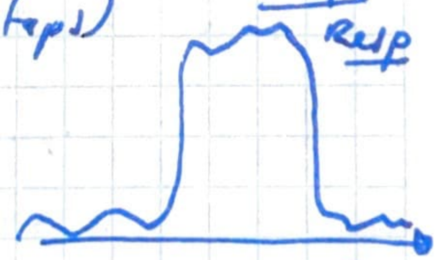
⋮

can be shown that DFT is real → linear phase also for other types

Min max lowpass

- Magnitude response
  - equiripple in passband and stopband
- Design parameters
  - Order: N (number of taps)
  - passband edge ω_p
  - stopband edge ω_s
  - ratio of passband to stopband error $\frac{\delta_p}{\delta_s}$
- Test values
  - passband max error and stopband max error

Freq Resp



DSP Coursera II 17.5.24

Magnitude response in decibels

- G: max passband magnitude
  attenuation expressed in dB

$$A_{dB} = 20\log_{10}\left(|H(e^{j\omega})|/G\right)$$

Life beyond lowpass

- IIR and FIR methods can be used to also design other types than lowpass
  - IIR bandpass and highpass:
    modulate lowpass response
  - optimal FIR bandpass or highpass:
    Parks-McClellan algorithm
  - optimal FIR can be designed with piecewise linear magnitude resp.

DSP Coursera II 24.5.24

7.4. Filter Structures

The cost of a numerical filter is dependent on the number of operations per output sample and on the storage (memory).

Cascade Forms

Transfer function $H(z)$ can always be written as:

$$H(z) = b_0 \frac{\prod_{n=1}^{M-1} (1 - z_n z^{-1})}{\prod_{n=1}^{N-1} (1 - p_n z^{-1})}$$

where:
- z_n : $M-1$ complex zeroes
- p_n : $N-1$ complex poles

↳ complex roots → complex-conj.

A pair of first-order terms with complex-conjugate roots

↳ $(1 - a z^{-1})(1 - a^* z^{-1}) = 1 - 2\mathrm{Re}\{a\} z^{-1} + |a|^2 z^{-2}$

↳ Transfer function can be factored into product of first- and second-order terms with real coefficients

$$H(z) = b_0 \frac{\prod_{n=1}^{M_r} (1 - z_n z^{-1}) \prod_{n=1}^{M_c} (1 - 2\,\mathrm{Re}\{z_n\} z^{-1} + |z_n|^2 z^{-2})}{\prod_{n=1}^{N_r} (1 - p_n z^{-1}) \prod_{n=1}^{N_c} (1 - 2\,\mathrm{Re}\{p_n\} z^{-1} + |p_n|^2 z^{-2})}$$

where: · M_r : number of real zeros

· M_c : number of complex conjugate zeros

↳ $M_r + 2M_c = M-1$

↳ equivalently for poles

$N_r + 2N_c = N-1$

↳ resulting structure : cascade

DSP Coursera II

Parallel Forms

Another possibility to rewrite transfer function (partial fraction expansion):

$$H(z) = \underbrace{\sum_n D_n z^{-n}}^{\text{FIR part}} + \sum_n \frac{A_n}{1 - p_n z^{-1}} + \sum_n \frac{B_n + C_n z^{-1}}{(1 - p_n z^{-1})(1 - p_n^* z^{-1})}$$

↳ parallel structure of filters
- outputs are summed together

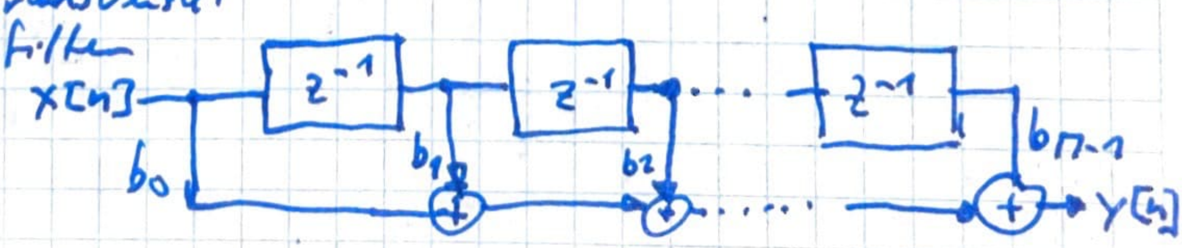
7.4.1. FIR Filter Structures

FIR transferfunction

$$H(z) = b_0 + b_1 z^{-1} + \dots + b_{M-1} z^{-(M-1)}$$

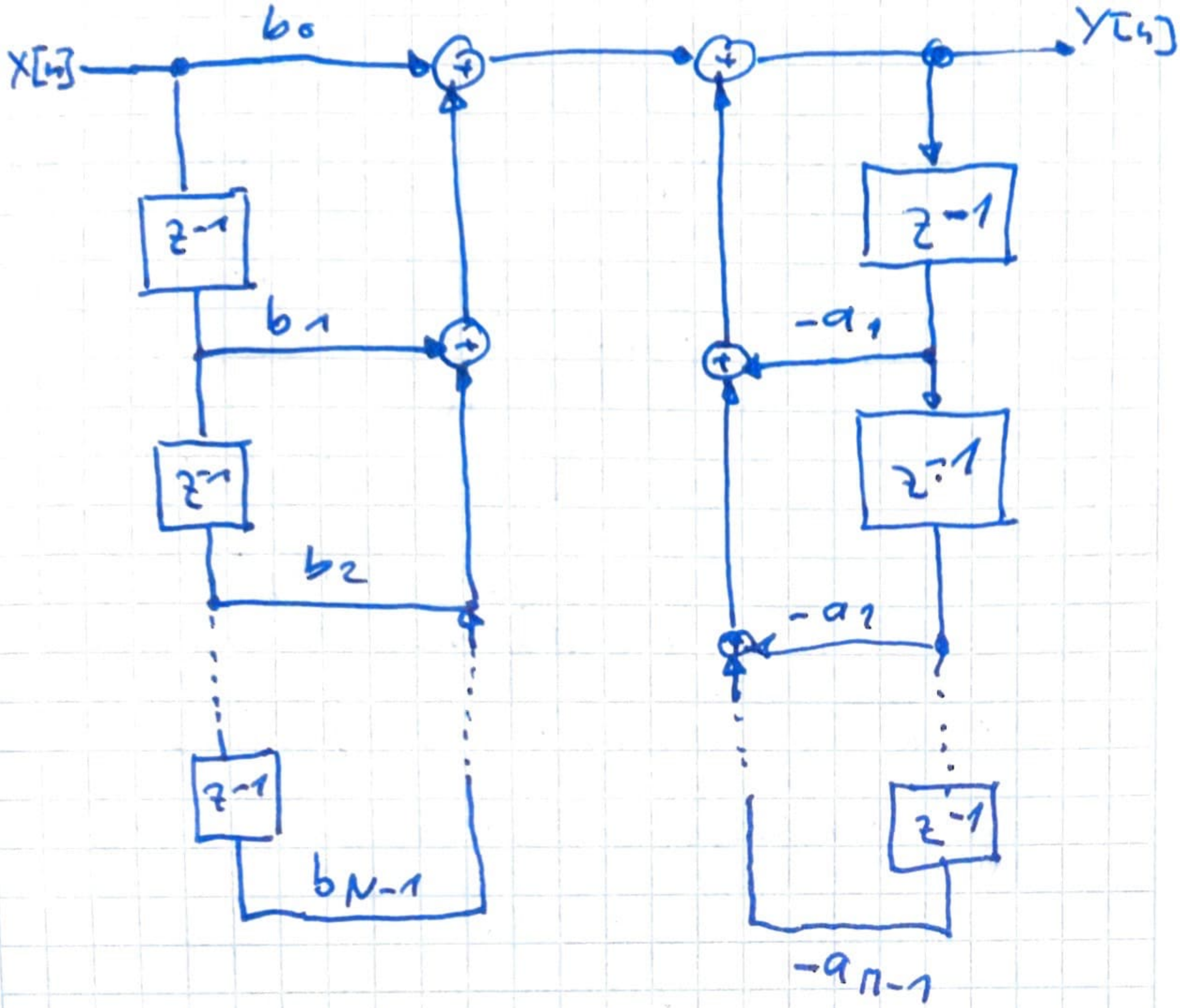
- coefficients: non-zero values of impulse response: $b_n = h[n]$

transversal filter

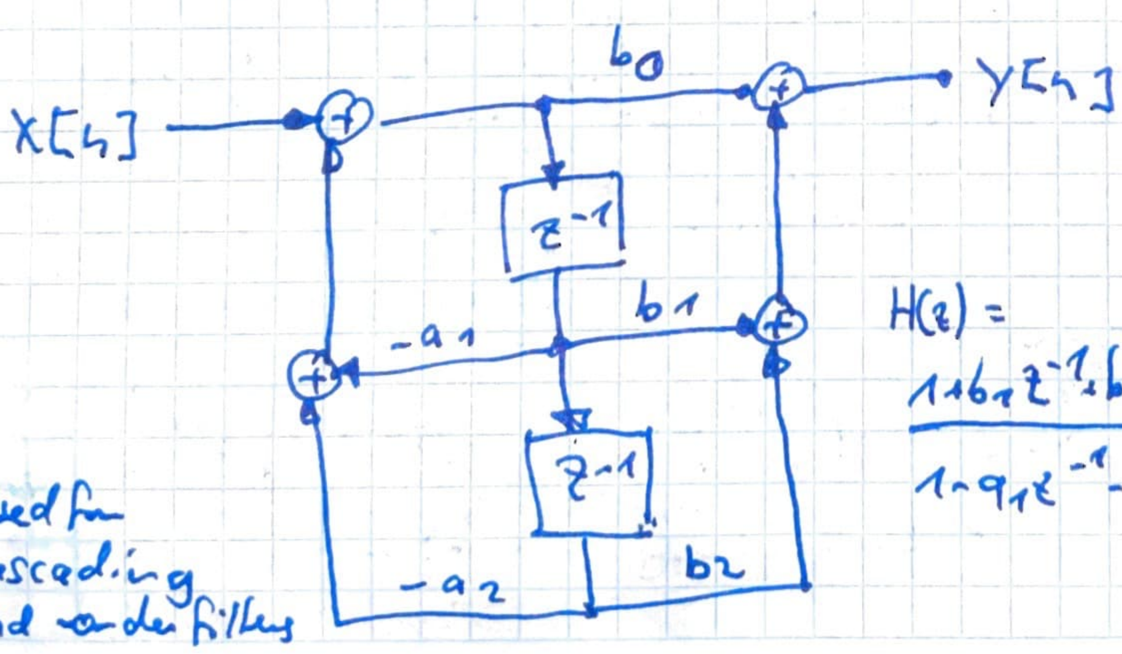


7.4.2 IIR Filter Structures

- Direct form I



- Direct form II (second order)



$$H(z) = \frac{1 + b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}}$$

Used for cascading 2nd order filters