Digital Signal Processing for Music

Part 15: Digital Filters I

alexander lerch

Georgia Tech Center for Music Technology

filters

introduction 1/2

filter — broad description

system that amplifies or attenuates certain components/aspects of a signal

filter — narrow description

linear time-invariant system for changing the magnitude and phase of specific frequency regions

- example for other type of filters:
  - adaptive and time-variant (e.g., denoising)
- examples for "real-world" filters
  - reverberation
  - absorption
  - echo

filters
introduction 1/2

filter — broad description

system that amplifies or attenuates certain components/aspects of a signal

filter — narrow description

linear time-invariant system for changing the magnitude and phase of specific frequency regions

- example for other type of filters:
  - adaptive and time-variant (e.g., denoising)
- examples for "real-world" filters
  - reverberation
  - absorption
  - echo

filters

introduction 1/2

filter — broad description

system that amplifies or attenuates certain components/aspects of a signal

filter — narrow description

linear time-invariant system for changing the magnitude and phase of specific frequency regions

- example for other type of filters:
  - adaptive and time-variant (e.g., denoising)
- examples for "real-world" filters
  - reverberation
  - absorption
  - echo

filters

introduction 2/2

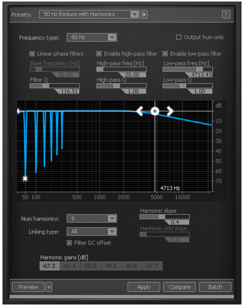
- **audio equalization**
  - parametric and graphic EQs
- **removal** of unwanted components
  - remove DC, rumble
  - remove hum
  - remove hiss
- **pre-emphasis/de-emphasis**
  - Vinyl
  - old Dolby noise reduction systems
- **weighting** function
  - dBA, dBC, ...

en.wikipedia.org/wiki/File:Graphic_equalizer.jpg

filters

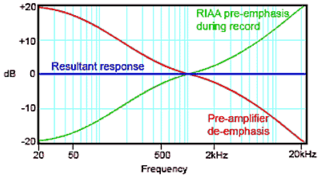
introduction 2/2

- **audio equalization**
  - parametric and graphic EQs
- **removal** of unwanted components
  - remove DC, rumble
  - remove hum
  - remove hiss
- **pre-emphasis/de-emphasis**
  - Vinyl
  - old Dolby noise reduction systems
- **weighting** function
  - dBA, dBC, . . .



filters
introduction 2/2

- **audio equalization**
  - parametric and graphic EQs
- **removal** of unwanted components
  - remove DC, rumble
  - remove hum
  - remove hiss
- **pre-emphasis/de-emphasis**
  - Vinyl
  - old Dolby noise reduction systems
- **weighting** function
  - dBA, dBC, ...

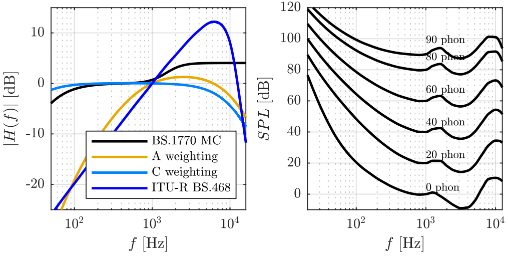


learnabout-electronics.org/Amplifiers/images/RIAA-curve.gif

filters

introduction 2/2

- **audio equalization**
  - parametric and graphic EQs
- **removal** of unwanted components
  - remove DC, rumble
  - remove hum
  - remove hiss
- **pre-emphasis/de-emphasis**
  - Vinyl
  - old Dolby noise reduction systems
- **weighting** function
  - dBA, dBC, . . .

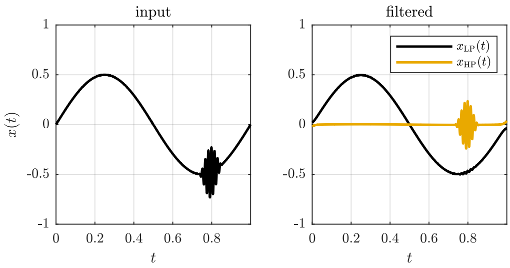


matlab source: plotLoudnessWeighting.m

filters

introduction 2/2

- **audio equalization**
  - parametric and graphic EQs
- **removal** of unwanted components
  - remove DC, rumble
  - remove hum
  - remove hiss
- **pre-emphasis/de-emphasis**
  - Vinyl
  - old Dolby noise reduction systems
- **weighting** function
  - dBA, dBC, ...



matlab source: plotSmoothing.m

filters

reminder: system theory

- output of a system (filter) y computed by **convolution** of input x and impulse response h

$$y(t) = x(t) * h(t)$$

- this is equivalent to a frequency domain multiplication

$$\begin{aligned} Y(\mathrm{j}\omega) &= X(\mathrm{j}\omega) \cdot H(\mathrm{j}\omega) \\ H(\mathrm{j}\omega) &= \frac{Y(\mathrm{j}\omega)}{X(\mathrm{j}\omega)} \end{aligned}$$

- **transfer function** $H(\mathrm{j}\omega)$ is complex, often represented as
    - **magnitude** $|H(\mathrm{j}\omega)|$ and
    - phase $\Phi_{\mathrm{H}}(\mathrm{j}\omega)$

filters

reminder: system theory

- output of a system (filter) y computed by **convolution** of input x and impulse response h
$$y(t) = x(t) * h(t)$$

- this is equivalent to a frequency domain multiplication
$$\begin{aligned} Y(\mathrm{j}\omega) &= X(\mathrm{j}\omega) \cdot H(\mathrm{j}\omega) \\ H(\mathrm{j}\omega) &= \frac{Y(\mathrm{j}\omega)}{X(\mathrm{j}\omega)} \end{aligned}$$

- **transfer function** $H(\mathrm{j}\omega)$ is complex, often represented as
  - **magnitude** $|H(\mathrm{j}\omega)|$ and
  - phase $\Phi_\mathrm{H}(\mathrm{j}\omega)$

filters

reminder: system theory

- output of a system (filter) y computed by **convolution** of input x and impulse response h

$$y(t) = x(t) * h(t)$$

- this is equivalent to a frequency domain multiplication

$$\begin{aligned} Y(\mathrm{j}\omega) &= X(\mathrm{j}\omega) \cdot H(\mathrm{j}\omega) \\ H(\mathrm{j}\omega) &= \frac{Y(\mathrm{j}\omega)}{X(\mathrm{j}\omega)} \end{aligned}$$

- **transfer function** $H(\mathrm{j}\omega)$ is complex, often represented as
  - **magnitude** $|H(\mathrm{j}\omega)|$ and
  - phase $\Phi_\mathrm{H}(\mathrm{j}\omega)$

filters

common transfer function shapes

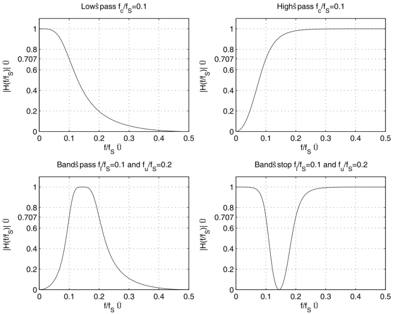
what are typical filters/spectral filter shapes

filters

common transfer function shapes

what are typical filters/spectral filter shapes

- very common:
  - low/high pass



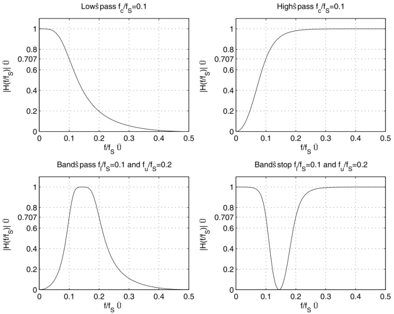
[1] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

common transfer function shapes

what are typical filters/spectral filter shapes

- very common:
  - low/high pass
- common for non-audio/non-parametric:
  - band pass/band stop



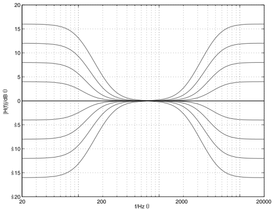
[1] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

common transfer function shapes

what are typical filters/spectral filter shapes

- very common:
  - low/high pass
- common for non-audio/non-parametric:
  - band pass/band stop
- also common in audio apps:
  - low/high shelving



[1] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

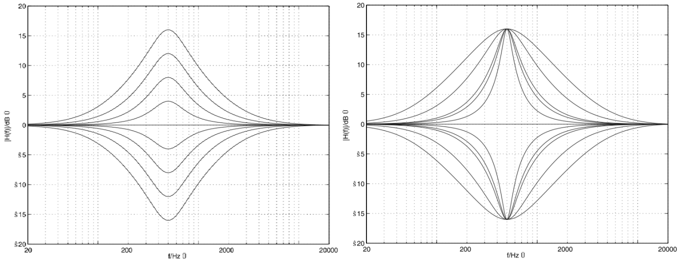
filters

common transfer function shapes

Georgia Tech | Center for Music Technology
College of Design

what are typical filters/spectral filter shapes

- very common:
  - low/high pass
- common for non-audio/non-parametric:
  - band pass/band stop
- also common in audio apps:
  - low/high shelving
  - peak filter



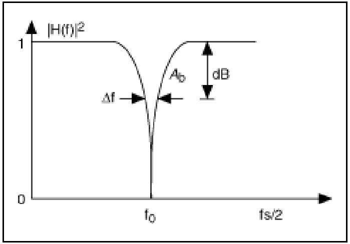
[1] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

common transfer function shapes

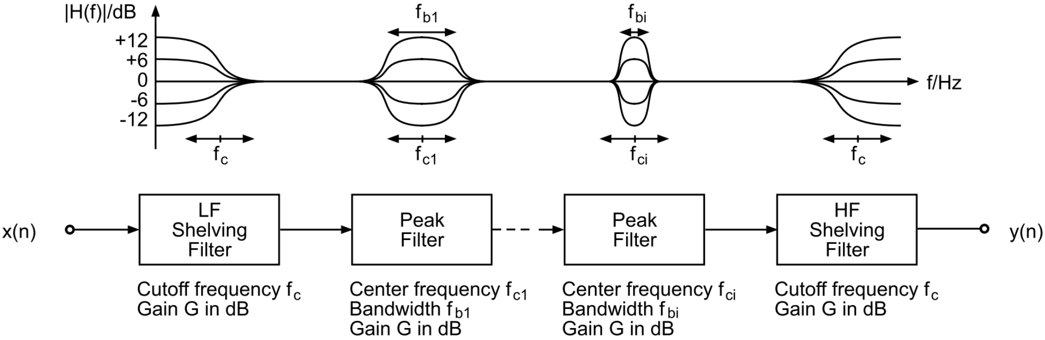
what are typical filters/spectral filter shapes

- very common:
  - low/high pass
- common for non-audio/non-parametric:
  - band pass/band stop
- also common in audio apps:
  - low/high shelving
  - peak filter
  - resonance/notch



filters

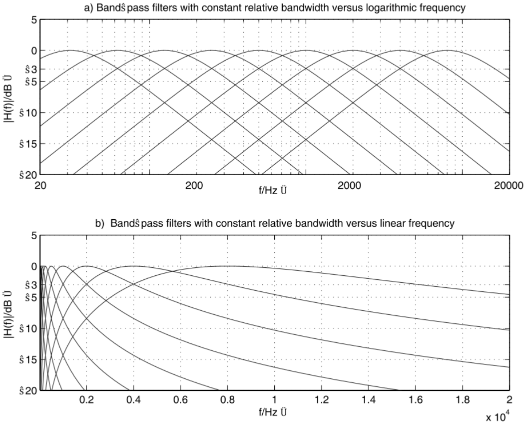
filters in series



[1] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

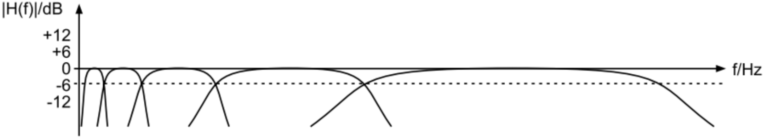
filter banks — parallel connections

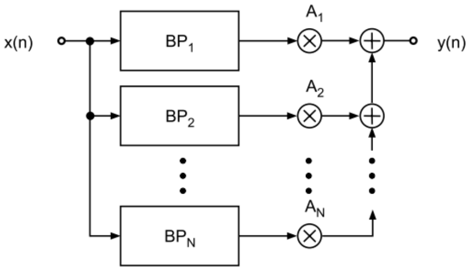


[2] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

filter banks — parallel connections



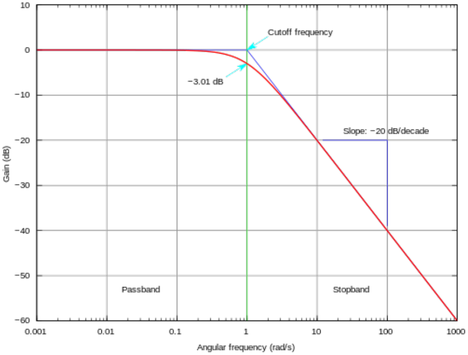


[2] U. Zölzer, *Digital Audio Signal Processing*, 2nd Edition. Stuttgart: John Wiley & Sons Ltd, 2008, ISBN: 978-0-470-99785-7.

filters

filter parameters — lowpass/highpass

- **cut-off** frequency f_c
  - frequency marking the transition of pass to stop band
  - -3 dB of pass band level
- **slope**/steepness
  - measured in dB/Octave or dB/Decade
  - typically directly related to filter order
- sometimes: **resonance**
  - level increase in narrow band around cut-off frequency

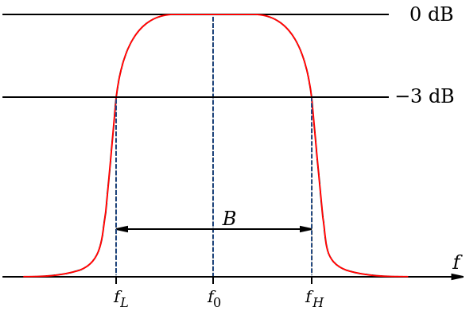


en.wikipedia.org/wiki/File:Butterworth_response.svg

filters

filter parameters — bandpass/bandstop

- **center** frequency f_c
  - frequency marking the center of the pass or stop band
- **bandwidth** ΔB
  - width of the pass band
  - at -3 dB of max pass band level
- possibly: **slope**
  - typically directly related to filter order



en.wikipedia.org/wiki/File:Bandwidth_2.svg

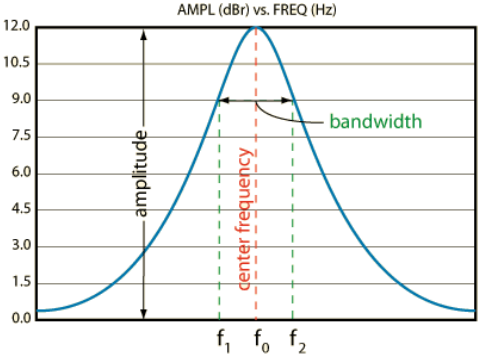
filters

filter parameters — peak

- **center** frequency f_c
  - frequency marking the center of the peak
- **Q factor** or **bandwidth** ΔB
  - width of the bell
  - at -3 dB of max gain

$$Q = \frac{f_c}{\Delta B}$$

- **gain**
  - amplification/attenuation in dB



sengpielaudio.com/FilterBandwidth09B.gif

filters

filter parameters — overview

parameter	**lowpass**	**low shelving**	**band pass**	**peak**	**resonance**
frequency	cut-off	cut-off	center	center	center
bandwidth/Q	res. gain	—	ΔB	Q	—
gain	—	yes	—	yes	—

filters

digital filter description

$$H(\mathrm{j}\omega) = \mathfrak{F}\{h(t)\}$$

filter is defined by its

- complex transfer function $H(\mathrm{j}\omega)$, or its
- impulse response $h(t)$, or its
- *list of pole and zero positions in the Z-plane*

filters

digital filter description

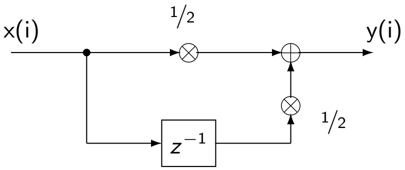
$$H(\mathrm{j}\omega) = \mathfrak{F}\{h(t)\}$$

filter is defined by its

- complex transfer function $H(\mathrm{j}\omega)$, or its
- impulse response $h(t)$, or its
- *list of pole and zero positions in the Z-plane*

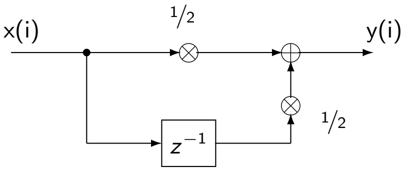
filters

example 1



filters

example 1



$$y(i) = 0.5 \cdot x(i) + 0.5 \cdot x(i-1)$$

filters

example 1: transfer function 1/2

$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) + 0.5 \cdot x(i-1) \\
H(z) &= 0.5 + 0.5 \cdot z^{-1} \\
H(j\omega) &= 0.5 + 0.5 \cdot e^{-j\omega} \\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{\omega}{2}} \cdot \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right| \\
&= 0.5 \cdot \underbrace{\left| e^{-j\frac{\omega}{2}} \right|}_{1} \cdot \underbrace{\left| \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|}_{\left|2\cos\left(\frac{\omega}{2}\right)\right|} \\
&= \left| \cos\left(\frac{\omega}{2}\right) \right|
\end{aligned}
$$

filters

example 1: transfer function 1/2

$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) + 0.5 \cdot x(i-1) \\
H(z) &= 0.5 + 0.5 \cdot z^{-1} \\
H(j\omega) &= 0.5 + 0.5 \cdot e^{-j\omega} \\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{\omega}{2}} \cdot \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right| \\
&= 0.5 \cdot \underbrace{\left| e^{-j\frac{\omega}{2}} \right|}_{1} \cdot \underbrace{\left| \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|}_{\left|2\cos\left(\frac{\omega}{2}\right)\right|} \\
&= \left| \cos\left(\frac{\omega}{2} \right) \right|
\end{aligned}
$$

filters

example 1: transfer function 1/2

$$\begin{aligned}
y(i) &= 0.5 \cdot x(i) + 0.5 \cdot x(i-1)\\
H(z) &= 0.5 + 0.5 \cdot z^{-1}\\
H(j\omega) &= 0.5 + 0.5 \cdot e^{-j\omega}\\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{\omega}{2}} \cdot \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|\\
&= 0.5 \cdot \underbrace{\left| e^{-j\frac{\omega}{2}} \right|}_{1} \cdot \underbrace{\left| \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|}_{\left|2\cos\left(\frac{\omega}{2}\right)\right|}\\
&= \left| \cos\left(\frac{\omega}{2}\right) \right|
\end{aligned}$$

filters

example 1: transfer function 1/2

$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) + 0.5 \cdot x(i-1) \\
H(z) &= 0.5 + 0.5 \cdot z^{-1} \\
H(j\omega) &= 0.5 + 0.5 \cdot e^{-j\omega} \\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{\omega}{2}} \cdot \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right| \\
&= 0.5 \cdot \underbrace{\left| e^{-j\frac{\omega}{2}} \right|}_{1} \cdot \underbrace{\left| \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|}_{\left|2\cos\left(\frac{\omega}{2}\right)\right|} \\
&= \left| \cos\left(\frac{\omega}{2}\right) \right|
\end{aligned}
$$

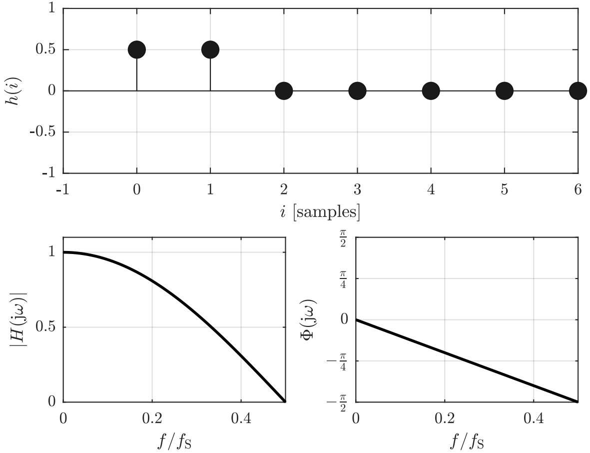
filters

example 1: transfer function 1/2

$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) + 0.5 \cdot x(i-1) \\
H(z) &= 0.5 + 0.5 \cdot z^{-1} \\
H(j\omega) &= 0.5 + 0.5 \cdot e^{-j\omega} \\
\left|H(j\omega)\right| &= 0.5 \cdot \left| e^{-j\frac{\omega}{2}} \cdot \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right| \\
&= 0.5 \cdot \underbrace{\left| e^{-j\frac{\omega}{2}} \right|}_{1} \cdot \underbrace{\left| \left(e^{j\frac{\omega}{2}} + e^{-j\frac{\omega}{2}} \right) \right|}_{\left|2\cos\left(\frac{\omega}{2}\right)\right|} \\
&= \left| \cos\left(\frac{\omega}{2} \right) \right|
\end{aligned}
$$

filters

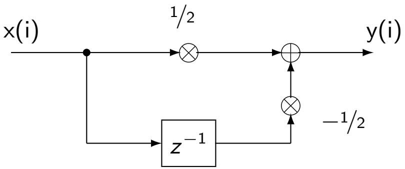
example 1: transfer function 2/2



matlab source: plotSimpleFilter.m

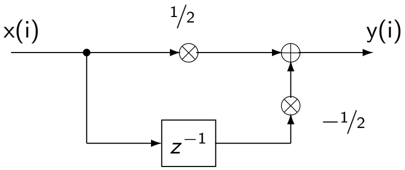
filters

example 2



filters

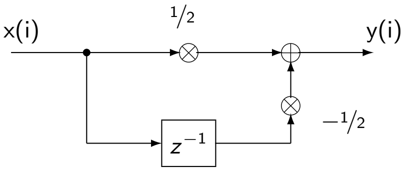
example 2



$$y(i) = 0.5 \cdot x(i) - 0.5 \cdot x(i-1)$$
$$H(z) = 0.5 - 0.5 \cdot z^{-1}$$

filters

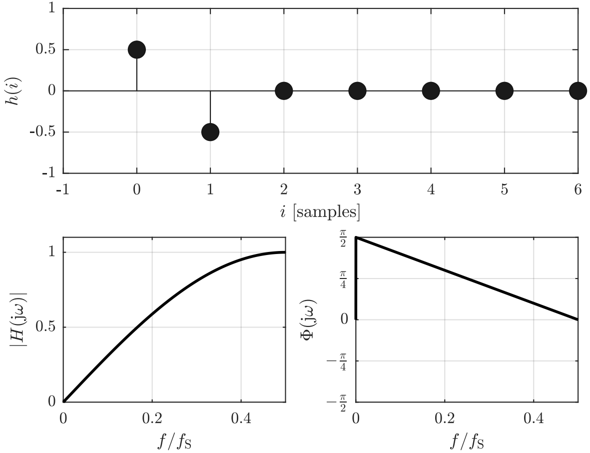
example 2



$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) - 0.5 \cdot x(i-1) \\
H(z) &= 0.5 - 0.5 \cdot z^{-1} \\
|H(j\omega)| &= \left|\sin\left(\frac{\omega}{2}\right)\right|
\end{aligned}
$$

filters

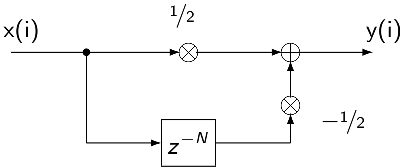
example 2: transfer function



matlab source: plotSimpleFilter.m

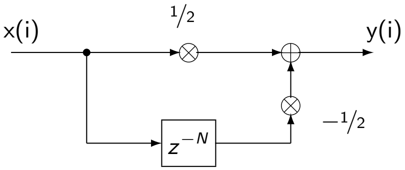
filters

example 3



filters

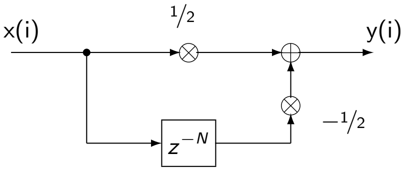
example 3



$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) - 0.5 \cdot x(i-N) \\
H(z) &= 0.5 - 0.5 \cdot z^{-N}
\end{aligned}
$$

filters

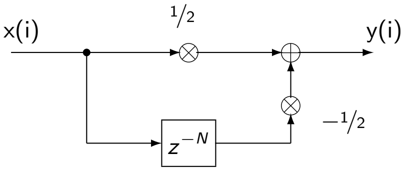
example 3



$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) - 0.5 \cdot x(i-N)\\
H(z) &= 0.5 - 0.5 \cdot z^{-N}\\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{N\omega}{2}} \cdot \left(e^{j\frac{N\omega}{2}} - e^{-j\frac{N\omega}{2}} \right) \right|
\end{aligned}
$$

filters

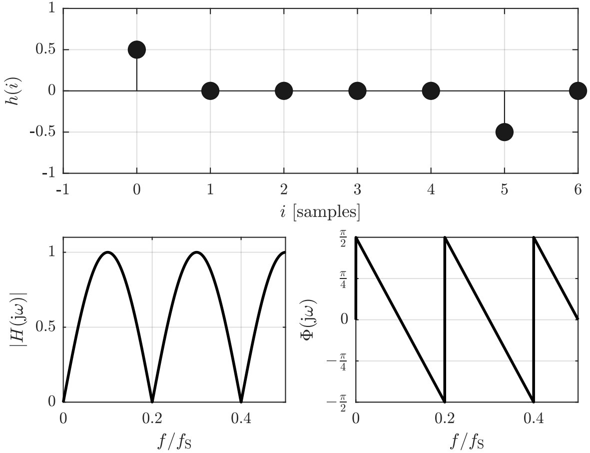
example 3



$$
\begin{aligned}
y(i) &= 0.5 \cdot x(i) - 0.5 \cdot x(i-N) \\
H(z) &= 0.5 - 0.5 \cdot z^{-N} \\
|H(j\omega)| &= 0.5 \cdot \left| e^{-j\frac{N\omega}{2}} \cdot \left(e^{j\frac{N\omega}{2}} - e^{-j\frac{N\omega}{2}} \right) \right| \\
&= \left| \sin\left(\frac{N\omega}{2} \right) \right|
\end{aligned}
$$

filters

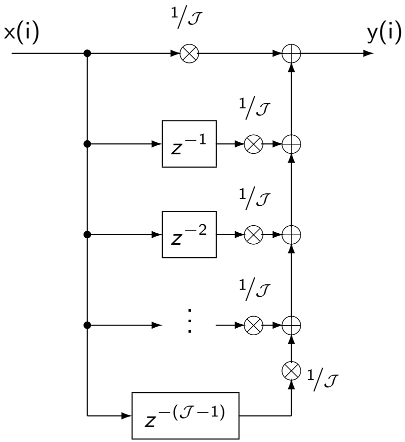
example 3: transfer function



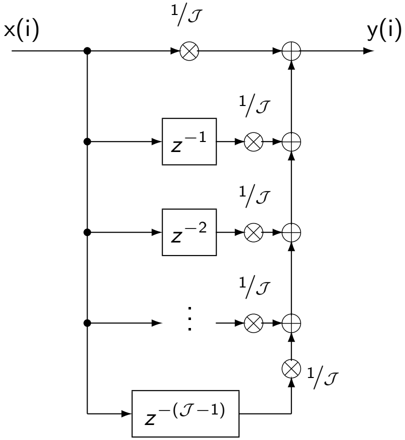
matlab source: plotSimpleFilter.m

filters

example 4



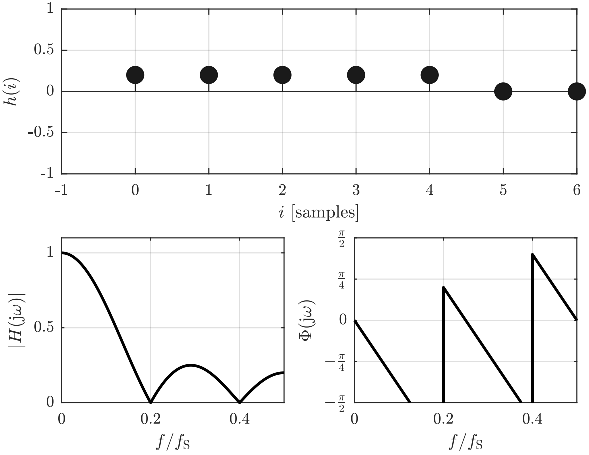
filters
example 4



$$y(i) = \frac{1}{\mathcal{J}} \sum_{j=0}^{\mathcal{J}-1} x(i-j)$$

filters

example 4: transfer function



$$H(j\omega) = e^{-\mathrm{j}\mathcal{J}\frac{\omega}{2}} \frac{\sin\left(\mathcal{J}\cdot\frac{\omega}{2}\right)}{\mathcal{J}\cdot\sin\left(\frac{\omega}{2}\right)}$$

matlab source: plotSimpleFilter.m

filters

example 4: recursive implementation

$$
\begin{aligned}
y(i) &= \sum_{j=0}^{\mathcal{J}-1} \frac{1}{\mathcal{J}} \cdot x(i-j) \\
&= \frac{1}{\mathcal{J}} \cdot \left(x(i) - x(i-\mathcal{J})\right) + \underbrace{\sum_{j=1}^{\mathcal{J}} \frac{1}{\mathcal{J}} \cdot x(i-j)}_{y(i-1)} \\
&= \frac{1}{\mathcal{J}} \cdot \left(x(i) - x(i-\mathcal{J})\right) + y(i-1)
\end{aligned}
$$

not applicable with windowed coefficients!

filters

example 4: recursive implementation

$$
\begin{aligned}
y(i) &= \sum_{j=0}^{\mathcal{J}-1} \frac{1}{\mathcal{J}} \cdot x(i-j) \\
&= \frac{1}{\mathcal{J}} \cdot \left(x(i) - x(i-\mathcal{J})\right) + \underbrace{\sum_{j=1}^{\mathcal{J}} \frac{1}{\mathcal{J}} \cdot x(i-j)}_{y(i-1)} \\
&= \frac{1}{\mathcal{J}} \cdot \left(x(i) - x(i-\mathcal{J})\right) + y(i-1)
\end{aligned}
$$

not applicable with windowed coefficients!

filters

example 4: recursive implementation

$$
\begin{aligned}
y(i) &= \sum_{j=0}^{\mathcal{J}-1} \frac{1}{\mathcal{J}} \cdot x(i-j) \\
&= \frac{1}{\mathcal{J}} \cdot \big(x(i) - x(i-\mathcal{J})\big) + \underbrace{\sum_{j=1}^{\mathcal{J}} \frac{1}{\mathcal{J}} \cdot x(i-j)}_{y(i-1)} \\
&= \frac{1}{\mathcal{J}} \cdot \big(x(i) - x(i-\mathcal{J})\big) + y(i-1)
\end{aligned}
$$

not applicable with windowed coefficients!

filters

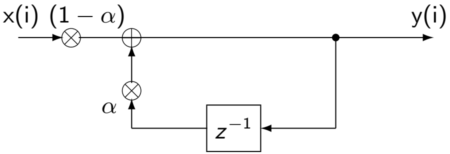
example 4: recursive implementation

$$
\begin{aligned}
y(i) &= \sum_{j=0}^{\mathcal{J}-1} \frac{1}{\mathcal{J}} \cdot x(i-j) \\
&= \frac{1}{\mathcal{J}} \cdot \big(x(i) - x(i-\mathcal{J})\big) + \underbrace{\sum_{j=1}^{\mathcal{J}} \frac{1}{\mathcal{J}} \cdot x(i-j)}_{y(i-1)} \\
&= \frac{1}{\mathcal{J}} \cdot \big(x(i) - x(i-\mathcal{J})\big) + y(i-1)
\end{aligned}
$$

not applicable with windowed coefficients!

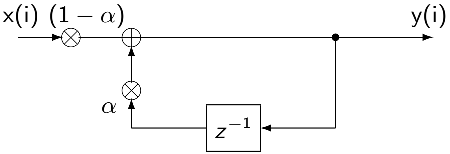
filters

example 5



filters

example 5



$$
\begin{aligned}
y(i) &= (1-\alpha)\cdot x(i) + \alpha \cdot y(i-1)\\
&= x(i) + \alpha \cdot (y(i-1) - x(i))
\end{aligned}
$$

Example 5: transfer function 1/2

$$
\begin{aligned}
y(i) &= (1-\alpha)\cdot x(i) + \alpha \cdot y(i-1)\\
H(z) &= \frac{1-\alpha}{1-\alpha z^{-1}}\\
H(j\omega) &= \frac{1-\alpha}{1-\alpha e^{-j\omega}}\\
|H(j\omega)| &= \left|\frac{1-\alpha}{1-\alpha e^{-j\omega}}\right|\\
&= \frac{1-\alpha}{\sqrt{(1+\alpha^2-2\alpha\cos(\omega))}}
\end{aligned}
$$

Example 5: transfer function 1/2

$$
\begin{aligned}
y(i) &= (1-\alpha)\cdot x(i) + \alpha \cdot y(i-1)\\
H(z) &= \frac{1-\alpha}{1-\alpha z^{-1}}\\
H(j\omega) &= \frac{1-\alpha}{1-\alpha e^{-j\omega}}\\
|H(j\omega)| &= \left|\frac{1-\alpha}{1-\alpha e^{-j\omega}}\right|\\
&= \frac{1-\alpha}{\sqrt{(1+\alpha^2-2\alpha\cos(\omega))}}
\end{aligned}
$$

Example 5: transfer function 1/2

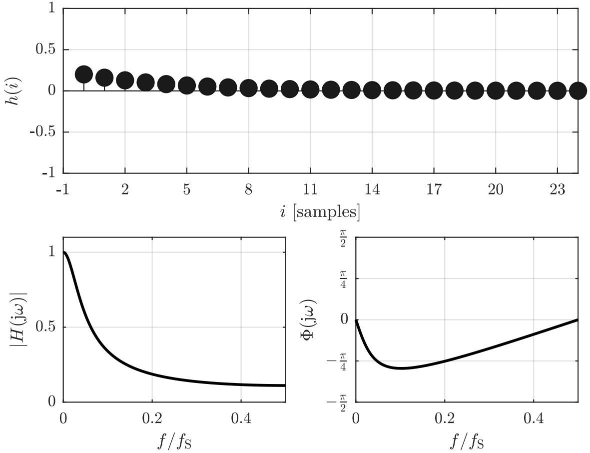
$$
\begin{aligned}
y(i) &= (1-\alpha)\cdot x(i) + \alpha\cdot y(i-1)\\
H(z) &= \frac{1-\alpha}{1-\alpha z^{-1}}\\
H(j\omega) &= \frac{1-\alpha}{1-\alpha e^{-j\omega}}\\
|H(j\omega)| &= \left|\frac{1-\alpha}{1-\alpha e^{-j\omega}}\right|\\
&= \frac{1-\alpha}{\sqrt{(1+\alpha^2-2\alpha\cos(\omega))}}
\end{aligned}
$$

Example 5: transfer function 1/2

$$
\begin{aligned}
y(i) &= (1-\alpha)\cdot x(i) + \alpha \cdot y(i-1) \\
H(z) &= \frac{1-\alpha}{1-\alpha z^{-1}} \\
H(j\omega) &= \frac{1-\alpha}{1-\alpha e^{-j\omega}} \\
|H(j\omega)| &= \left|\frac{1-\alpha}{1-\alpha e^{-j\omega}}\right| \\
&= \frac{1-\alpha}{\sqrt{(1+\alpha^2-2\alpha\cos(\omega))}}
\end{aligned}
$$

filters

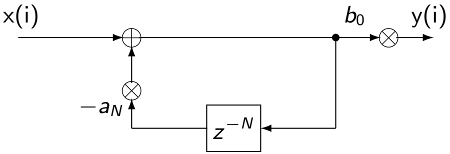
example 5: transfer function 2/2



matlab source: plotSimpleFilter.m

filters

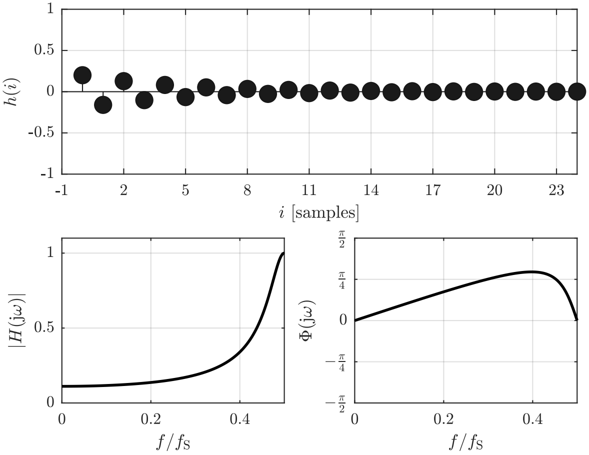
example 6



$$y(i) = b_0 \cdot x(i) - a_N \cdot y(i - N)$$

filters

example 6: transfer function



$$H(j\omega) = \frac{b_0}{1 - a_N \cdot e^{-j\omega N}}$$

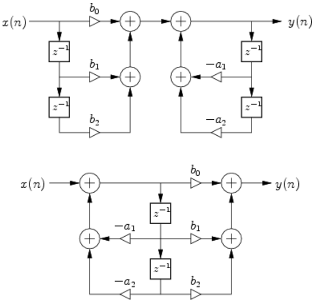
matlab source: plotSimpleFilter.m

filters

biquad: structure

(a)

(b)



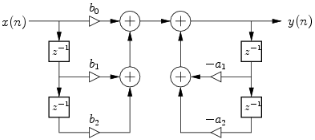
$$\text{diff eq} : y(i) = \sum_{k=0}^{K_1} b_k \cdot x(i-k) + \sum_{k=1}^{K_2} -a_k \cdot y(i-k)$$

$$\text{trans. fct} : H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{K_1} b_k \cdot z^{-k}}{1 + \sum_{k=1}^{K_2} a_k \cdot z^{-k}}$$

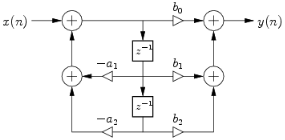
filters

biquad: structure

(a)



(b)



$$\text{diff eq}: y(i) = \sum_{k=0}^{K_1} b_k \cdot x(i-k) + \sum_{k=1}^{K_2} -a_k \cdot y(i-k)$$

$$\text{trans. fct}: H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{K_1} b_k \cdot z^{-k}}{1 + \sum_{k=1}^{K_2} a_k \cdot z^{-k}}$$

filters

summary

- filter (equalization) can be used for various tasks
  - changing the sound quality of a signal
  - hiding unwanted frequency components
  - smoothing
  - processing for measurement and transmission
- most common audio filter types are
  - low/high pass
  - peak
  - shelving
- filter parameters include
  - frequency (mid, cutoff)
  - bandwidth or Q
  - gain
- filter orders
  - typical orders are 1st, 2nd, maybe 4th
  - higher order give more flexibility wrt transfer function
  - higher orders are difficult to design and control

- filter (equalization) can be used for various tasks
  - changing the sound quality of a signal
  - hiding unwanted frequency components
  - smoothing
  - processing for measurement and transmission
- most common audio filter types are
  - low/high pass
  - peak
  - shelving
- filter parameters include
  - frequency (mid, cutoff)
  - bandwidth or Q
  - gain
- filter orders
  - typical orders are 1st, 2nd, maybe 4th
  - higher order give more flexibility wrt transfer function
  - higher orders are difficult to design and control

- filter (equalization) can be used for various tasks
  - changing the sound quality of a signal
  - hiding unwanted frequency components
  - smoothing
  - processing for measurement and transmission
- most common audio filter types are
  - low/high pass
  - peak
  - shelving
- filter parameters include
  - frequency (mid, cutoff)
  - bandwidth or Q
  - gain
- filter orders
  - typical orders are 1st, 2nd, maybe 4th
  - higher order give more flexibility wrt transfer function
  - higher orders are difficult to design and control

- filter (equalization) can be used for various tasks
  - changing the sound quality of a signal
  - hiding unwanted frequency components
  - smoothing
  - processing for measurement and transmission
- most common audio filter types are
  - low/high pass
  - peak
  - shelving
- filter parameters include
  - frequency (mid, cutoff)
  - bandwidth or Q
  - gain
- filter orders
  - typical orders are 1st, 2nd, maybe 4th
  - higher order give more flexibility wrt transfer function
  - higher orders are difficult to design and control