Advanced Digital Signal Processing

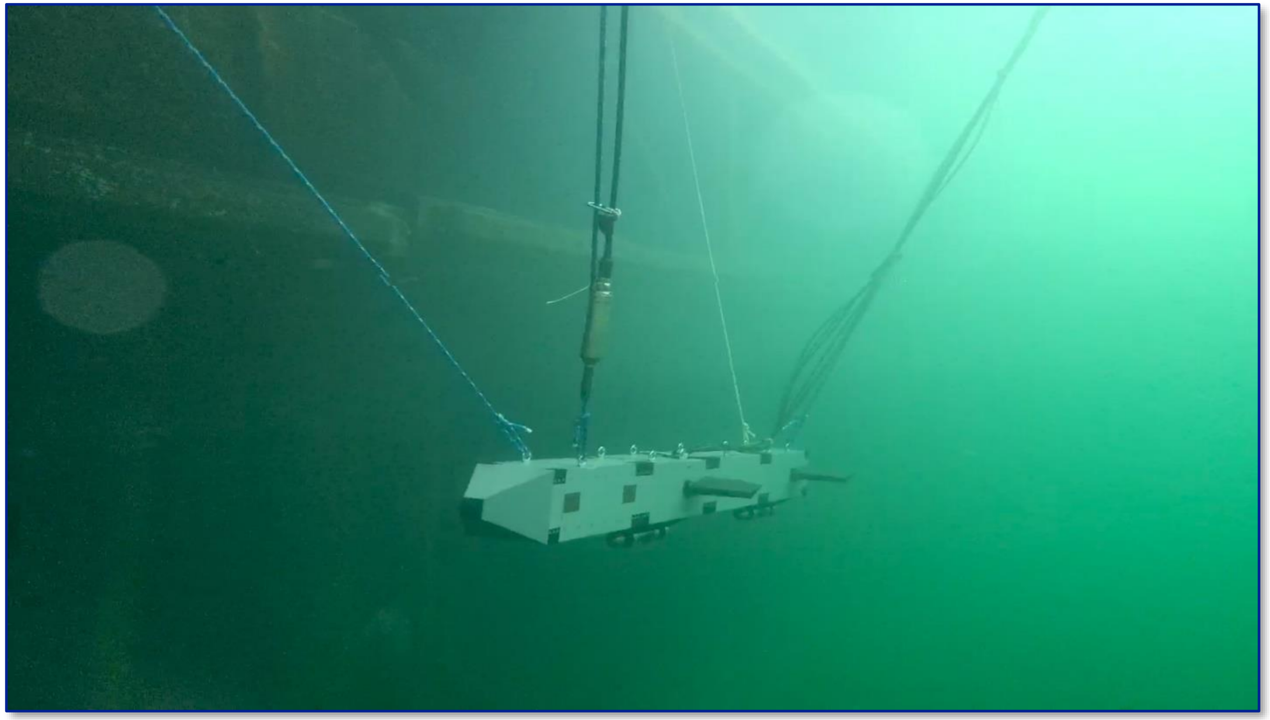
Part 5: Digital Filters

Gerhard Schmidt

Christian-Albrechts-Universität zu Kiel
Faculty of Engineering
Institute of Electrical and Information Engineering
Digital Signal Processing and System Theory

Contents

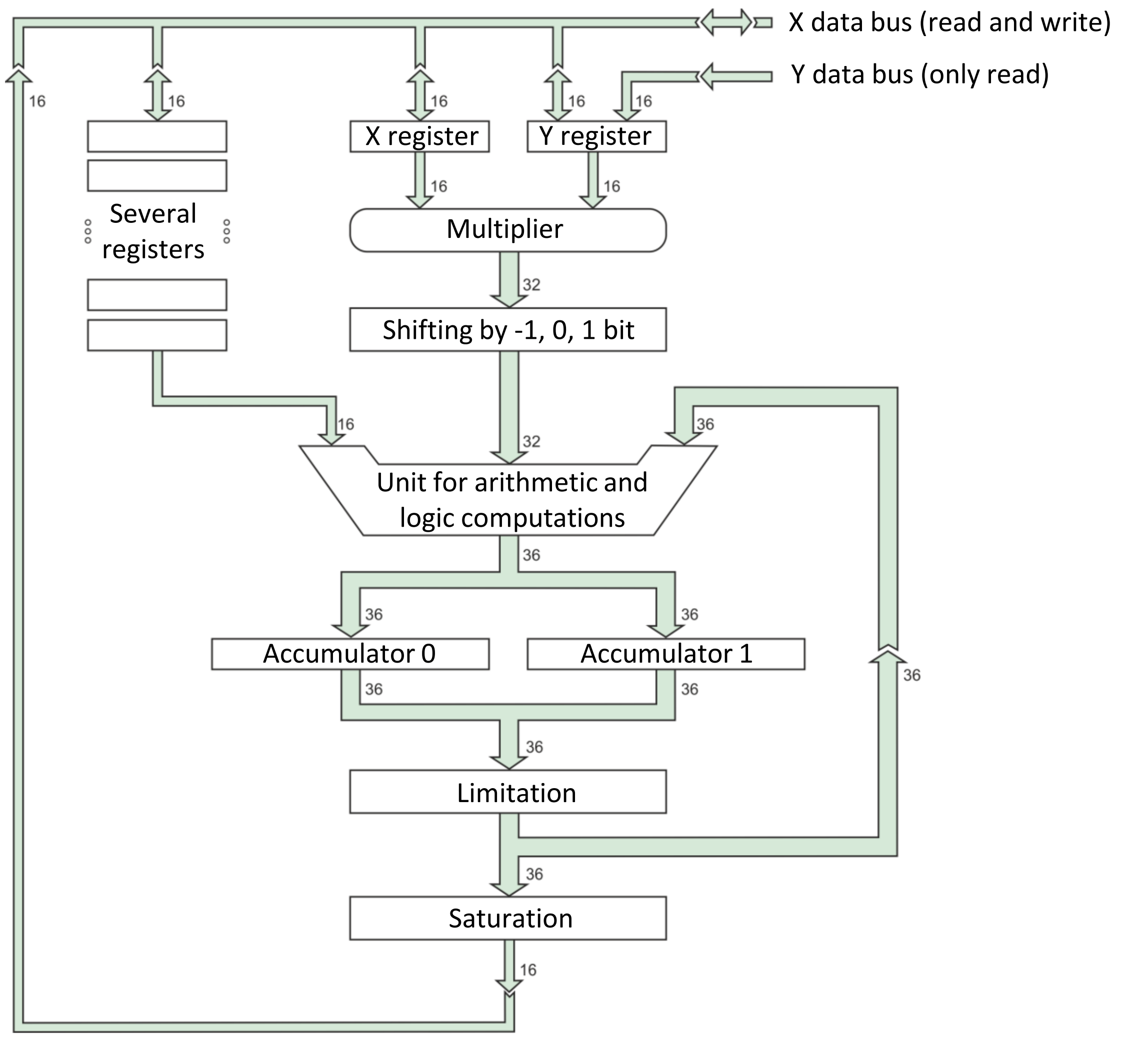
- Introduction
- Digital processing of continuous-time signals
- Efficient FIR structures
- DFT and FFT
- ***Digital filters***
  - ***Structures for FIR systems***
  - ***Structures for IIR systems***
  - ***Coefficient quantization and round-off effects***
  - ***Design of FIR filters***
  - ***Design of IIR filters***
- Multi-rate digital signal processing



Fixed-Point DSP Hardware

Example for a 16-bit fixed-point DSP architecture:

- Architecture with 2 busses
- Only main components are depicted
- Architecture varies among different vendors



General Remarks – Part 1

Digital filters:

- Linear-time-invariant (LTI) causal system with a rational transfer function (without loss of generality: numerator degree = denominator degree = N)

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

with $a_0 = 1$ without loss of generality.
$a_i,\ b_i$: parameters of the LTI system (***coefficients*** of the digital filter)
N : ***filter order***

- Product notation:

$$H(z) = b_0 \frac{\prod_{i=1}^{N} (z - z_{0,i})}{\prod_{i=1}^{N} (z - z_{\infty,i})}$$

where the $z_{0,i}$ are the ***zeros***, and the $z_{\infty,i}$ are the ***poles*** of the transfer function (latter are responsible for stability).

General Remarks – Part 2

- ***Difference equation***:

$$y(n) = \sum_{i=0}^{N} b_i v(n-i) - \sum_{i=1}^{N} a_i y(n-i),$$

with $v(n)$ denoting the input signal and $y(n)$ the resulting signal after filtering.

Remarks:

- Generally the above equation describes a recursive filter with an ***infinite impulse response*** (IIR filter): $y(n)$ is calculated from $v(n),\, v(n-1),\, ...,\, v(n-N)$ and recursively from $y(n-1),\, y(n-2),\, ...,\, y(n-N)$.

- The calculation of $y(n)$ requires some memory elements in order to store $v(n-1),\, ...,\, v(n-N)$ and $y(n-1),\, y(n-2),\, ...,\, y(n-N)$. $\Longrightarrow$ ***Dynamic system***.

- If $b_i \equiv 0 \;\forall i \neq 0$:

$$H(z) = \frac{b_0\, z^N}{\sum\limits_{i=0}^{N} a_{N-i}\, z^i} = \frac{b_0\, z^N}{\prod\limits_{i=1}^{N} (z - z_{\infty,i})}.$$

$\Longrightarrow$ Filter has no zeros $\Longrightarrow$ ***all-pole*** or autoregressive (AR) filter.

General Remarks – Part 3

The difference equation is purely recursive:

$$y(n) = b_0 v(n) - \sum_{i=1}^{N} a_i y(n-i)$$

- If $a_i \equiv 0 \ \forall i \neq 0, \ a_0 = 1$ (causal filter required!):
  The difference equation is purely non-recursive:

$$y(n) = \sum_{i=0}^{N} b_i v(n-i)$$

$\Rightarrow$ Non-recursive filter

Transfer function:

$$H(z) = \frac{1}{z^N} \sum_{i=0}^{N} b_{N-i} \, z^i = \sum_{i=0}^{N} b_i \, z^{-i}.$$

  - Poles $z_{\infty,i} = 0, \ i = 1, \ldots, N,$ but not relevant for stability $\Longrightarrow$ ***all-zero*** filter.
  - According to the difference equation: $y(n)$ is obtained by a weighted average of the last $N+1$ input values $\Longrightarrow$ ***Moving average*** (MA) filter (as opposite to the AR filter).
  - From the transfer function it can be seen that the impulse response has finite length. $\Longrightarrow$ Finite impulse response (FIR) filter of length $L = N+1$ and order N.

Structures for FIR systems – Part 1

The impulse response is equal to the coefficients b_i :

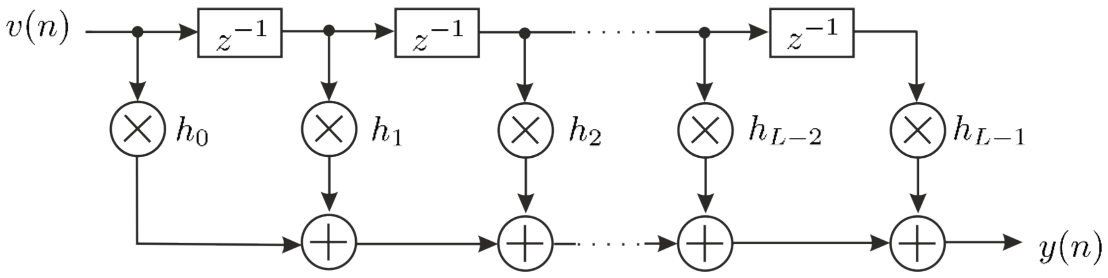
$$h_i = \begin{cases} b_i, & \text{for } 0 \le i \le N, \\ 0, & \text{else.} \end{cases}$$

With the difference equation of the FIR systems and the relation above we get

$$y(n) = \sum_{i=0}^{L-1} h_i \, v(n-i),$$

which is the linear convolution sum (with $L = N - 1$). A possible realization is given in the

Direct form structure: Tapped-delay or transversal filter – Part 1



$\Rightarrow$ ***First direct form***

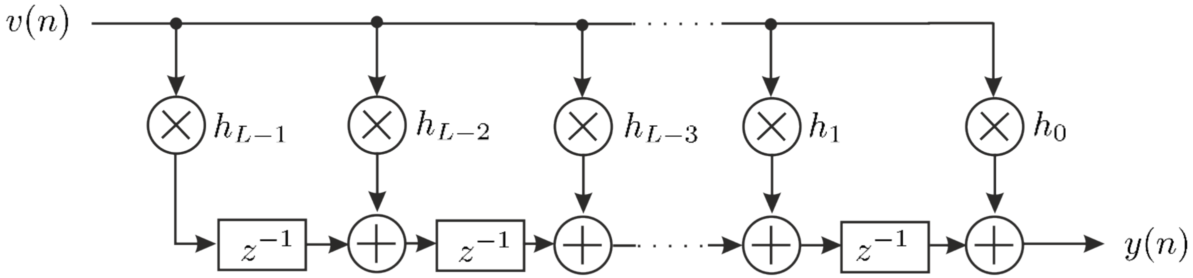
Structures for FIR systems – Part 2

Direct form structure: Tapped-delay or transversal filter – Part 2

By transposing the flow graph, that means

- reversing the direction of all branches,
- exchanging the input and output of the flow graph and
- exchanging summation points with branching points and vice versa,

we get the ***second direct form*** (below redrawn version):



If the unit impulse $v(n) = \gamma_0(n)$ is chosen as the input signal, all samples of the impulse response $h_n = y(n)$ appear successively at the output of the structure.

Structures for FIR systems – Part 3

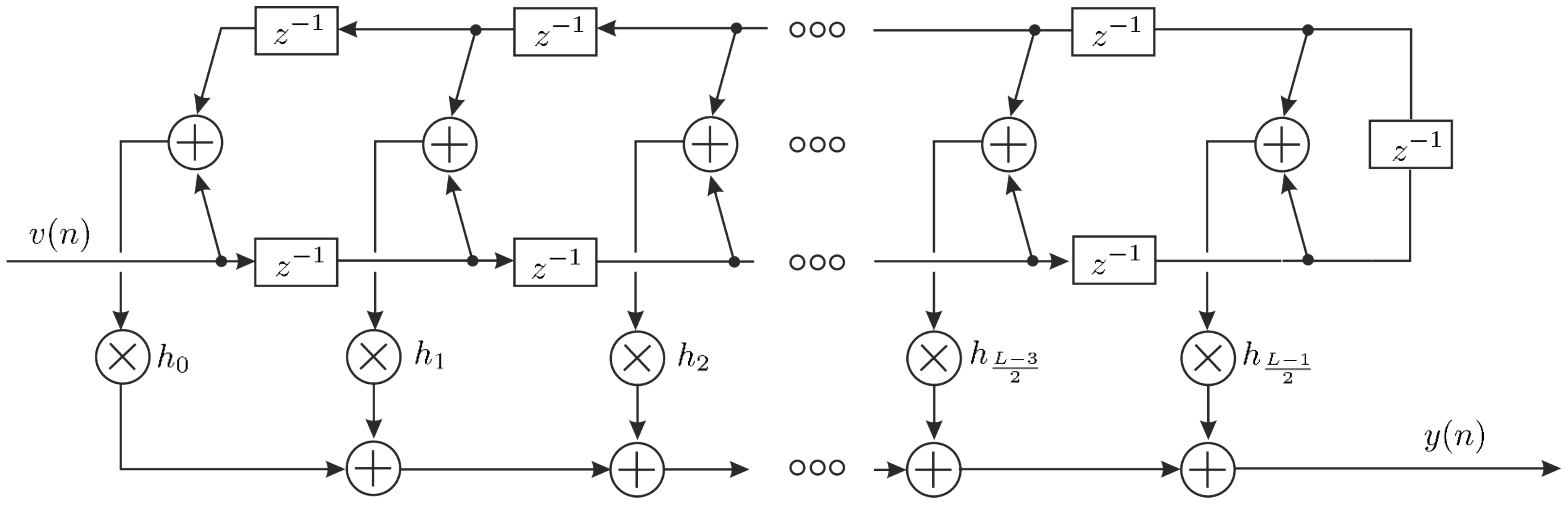
Direct form structure: Tapped-delay or transversal filter – Part 3

The number of multiplications can be reduced if the impulse response of the system is symmetric, e.g. if we have:

$$h_i = h_{L-1-i} \text{ or}$$

$$h_i = -h_{L-1-i}.$$

Then the number of multiplication can be reduced from L to $L/2$ in the even case and to $(L+1)/2$ in the odd case.
Below the flow graph for the odd case is depicted:

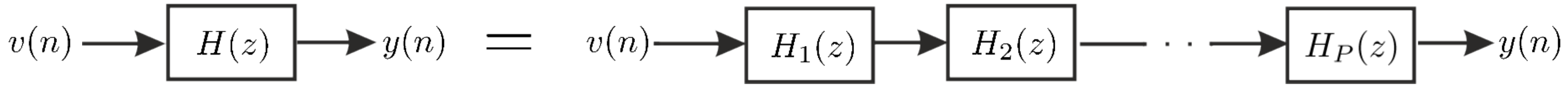


Structures for FIR systems – Part 4

Cascade-form structures

We obtain the cascade realization by factorizing the transfer function into a cascade of shorter length filters:

$$H(z) \; = \; \prod_{p=1}^{P} H_p(z)$$



Remarks:

- The reason for the cascade structure is that the shorter-length filters $H_p(z)$ can be implemented with improved robustness in one of the direct forms than the overall filter.
- The $H_p(z)$ are usually second order filters with real coefficients. Therefore the poles and zeros have to be real or appear in conjugate complex pairs.
- For linear phase filters the zeros have to appear in quadruples.

Structures for IIR systems – Part 1

Direct form structures – Part 1

A rational system function $H(z)$ can be divided into two parts – an ***all-zero part*** $H_1(z)$ and in an ***all-pole part*** $H_2(z)$.

$$H(z) = \frac{N(z)}{D(z)} = H_1(z) \cdot H_2(z)$$

$$H_1(z) = N(z) = \sum_{i=0}^{N} b_i z^{-i},$$

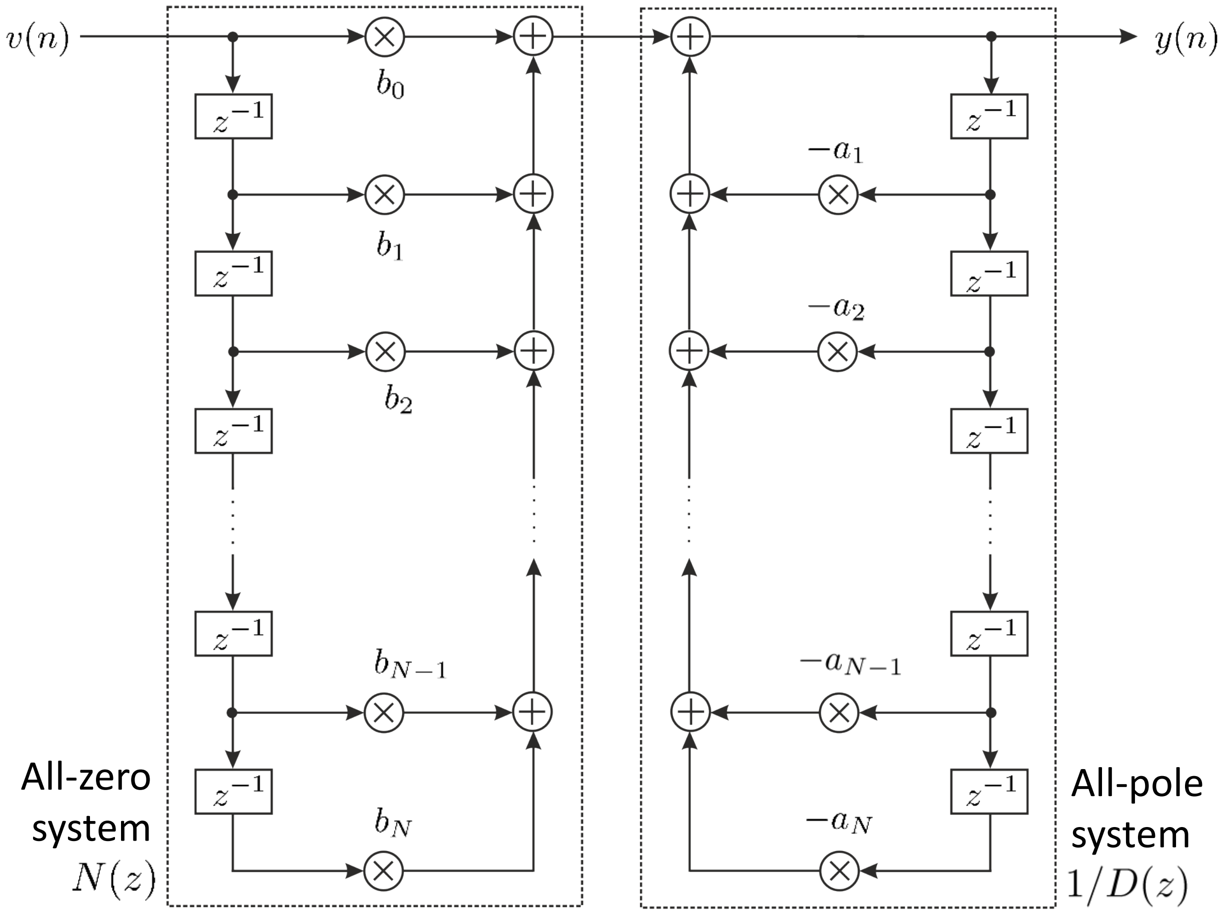
$$H_2(z) = \frac{1}{D(z)} = \frac{1}{1 + \sum_{i=1}^{N} a_i z^{-i}}.$$

The all-zero filter $H_1(z)$ can be realized with the direct form. By attaching the all-pole system $H_2(z)$ in cascade we obtain the ***direct form I realization.***

Structures for IIR systems – Part 2

Direct form structures – Part 2

Signal flow graph of the ***direct form I realization***:



Structures for IIR systems – Part 3

Direct form structures – Part 3

Another realization can be obtained by exchanging the position of the all-pole and the all-zero filter.

Difference equation for the all-pole part:

$$w(n) = -\sum_{i=1}^{N} a_i w(n-i) + v(n)$$

where the sequence $w(n)$ is an intermediate result and is the input of the all-zero part.

Difference equation of the all-zero part:

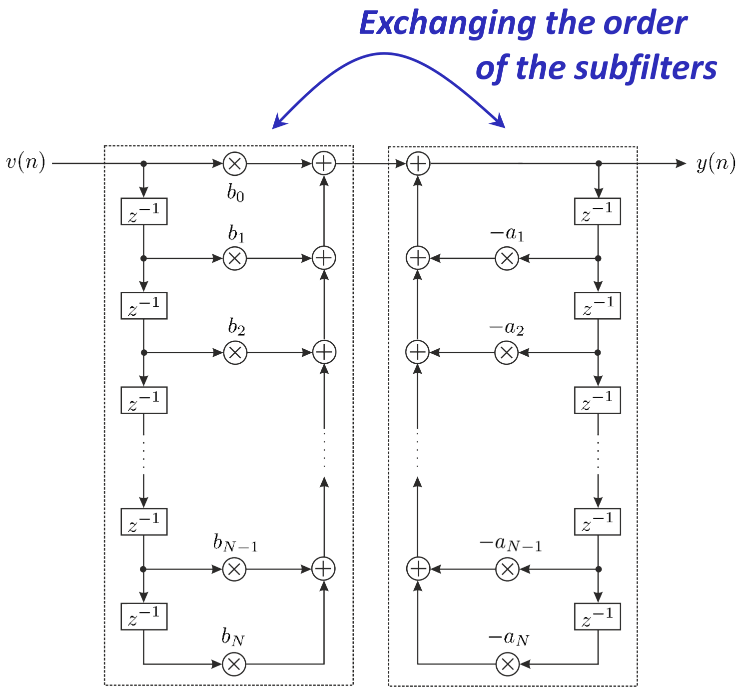
$$y(n) = \sum_{i=0}^{N} b_i w(n-i)$$

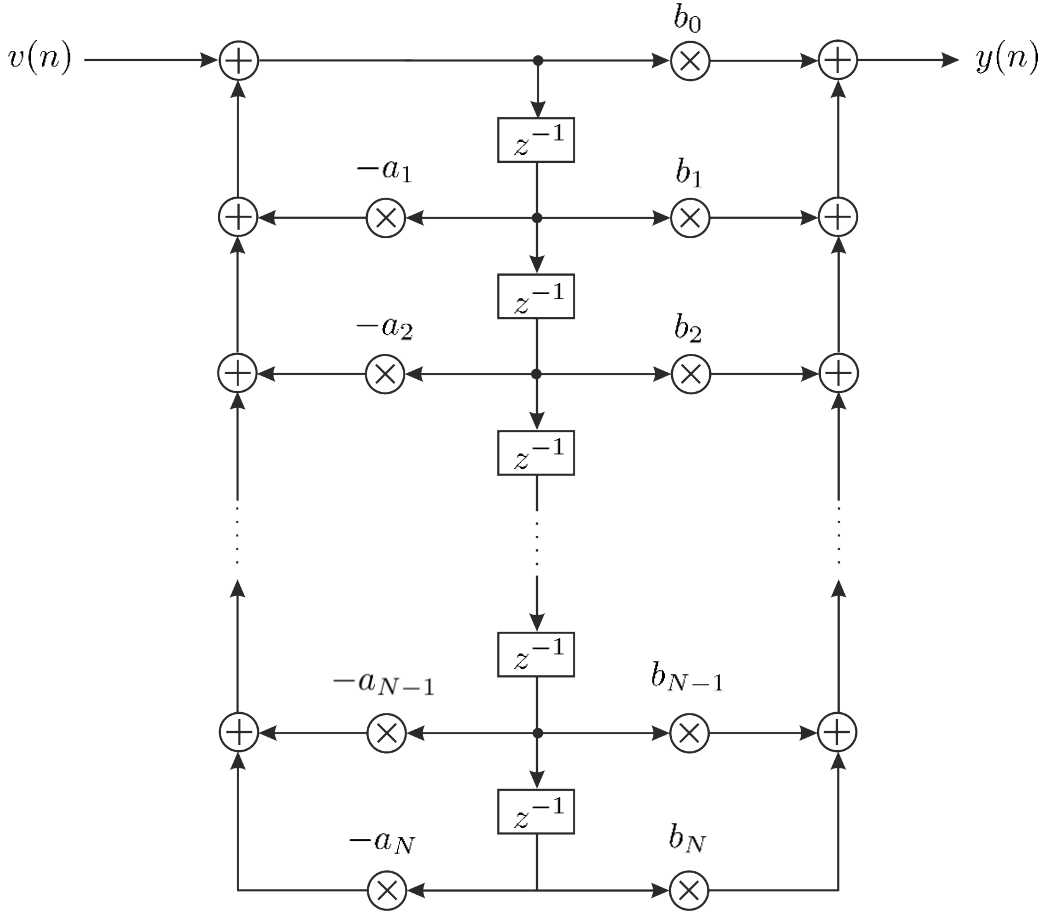
The resulting structure is called ***direct form II realization***. Furthermore, it is said to be ***canonic*** since it minimizes the number of memory elements. Only one single delay line is required for storing the delayed versions of the sequence $w(n)$.

Structures for IIR systems – Part 4

Direct form structures – Part 4

Signal flow graph
of the ***direct form II realization***:

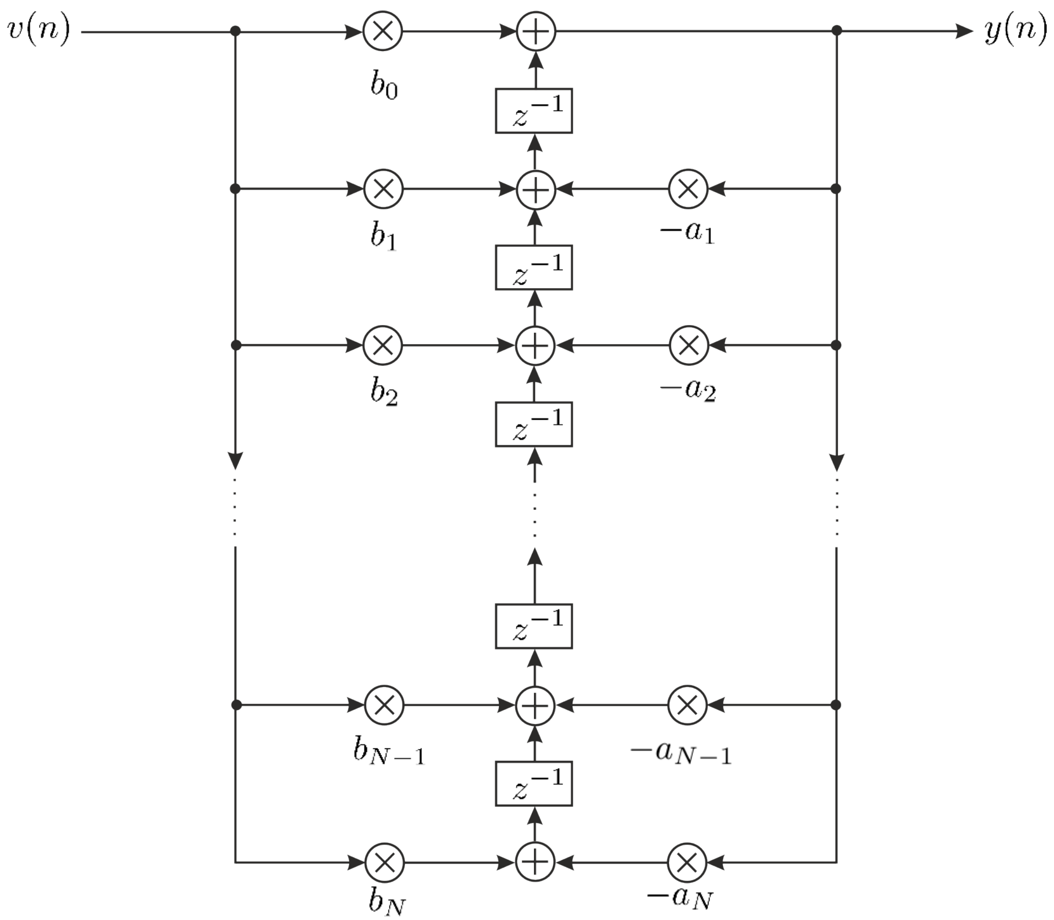




Structures for IIR systems – Part 5

Direct form structures – Part 5

Transposing the ***direct form II realization*** leads to the following structure:



Structures for IIR systems – Part 6

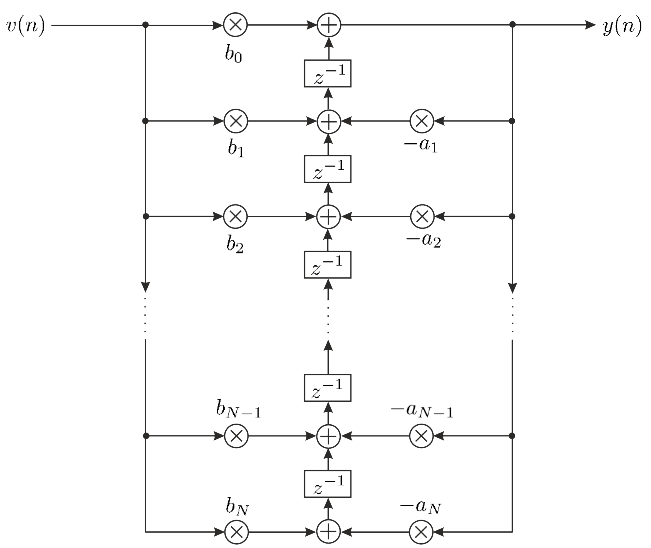
Direct form structures – Part 6

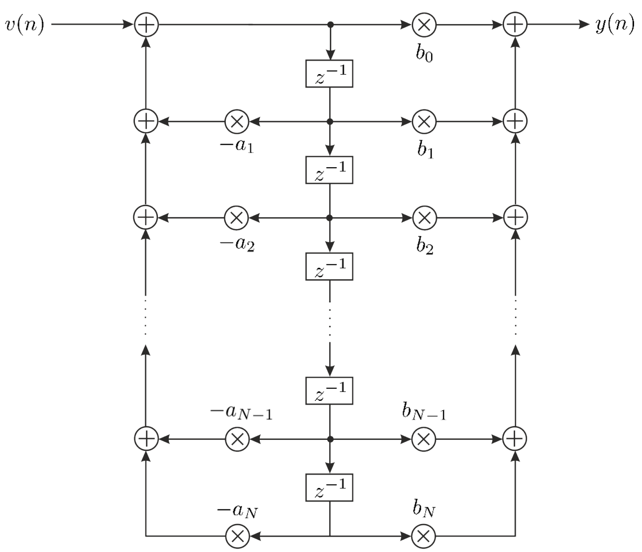
Partner work – Please think about the following question and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What are the advantages and disadvantages of the two structures depicted below?

....

....





Structures for IIR systems – Part 7

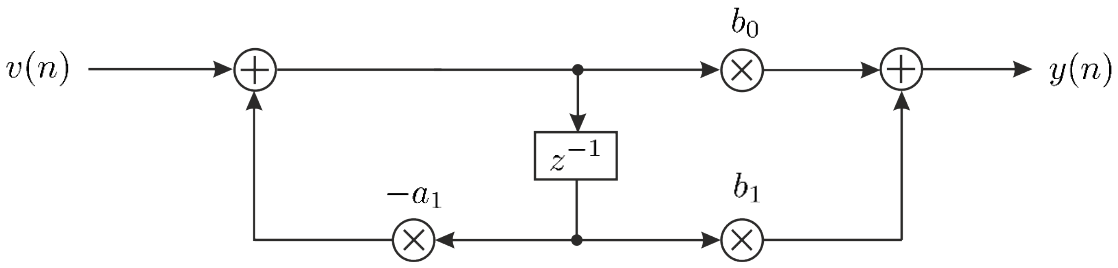
Cascade-form structures – Part 1

As for FIR systems we can cascade Subsystems $H_p(z)$ of first or second order to the desired system $H(z)$:

$$H(z) = \prod_{p=0}^{P-1} H_p(z).$$

First order subsystem:

Canonical direct form for a first order filter ("bi-linear" filter):



Every first order transfer function can be realized with the above flow graph.
Corresponding transfer function:

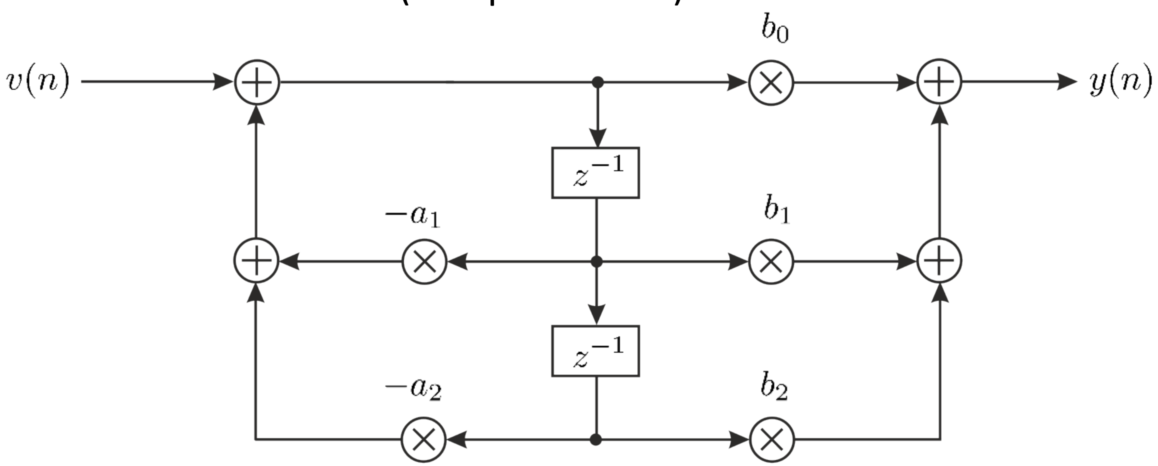
$$H(z) = \frac{Y(z)}{V(z)} = \frac{b'_0 + b'_1 z^{-1}}{a'_0 + a'_1 z^{-1}} = \frac{(b'_0/a'_0) + (b'_1/a'_0) z^{-1}}{1 + (a'_1/a'_0) z^{-1}} = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}.$$

Structures for IIR systems – Part 8

Cascade-form structures – Part 2

Second order subsystem:

Canonical direct form for a second order filter ("bi-quad" filter):



Corresponding transfer function:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_1 z^{-2}}$$

Cascade-form structures – Part 3

Example:

Given is a so-called ***Chebyshev*** lowpass filter of 5th order and the cut-off frequency $f_{\text{co}} = 0.25 f_s$ (f_s is the sampling frequency). A filter design approach yields the transfer function below. The corresponding filter design algorithms will be discussed later on:

$$H(z) \;=\; 0.03217 \cdot \frac{1 + 5z^{-1} + 10z^{-2} + 10z^{-3} + 5z^{-4} + z^{-5}}{1 - 0.782z^{-1} - 1.2872z^{-2} - 0.7822z^{-3} + 0.4297z^{-4} - 0.1234z^{-5}}$$

- The zeros are all at $z = -1$: $z_{0,i} = -1$ for $i = 1, 2, ..., 5$.
  The poles are $z_{\infty,1,2} = -0.0336 \pm j\,0.8821$, $z_{\infty,3,4} = 0.219 \pm j\,0.5804$, $z_{\infty,5} = 0.4113$.
- By grouping the poles $z_{\infty,1,2}$ and $z_{\infty,3,4}$ we get three subsystems – two second order subsystems and one first order subsystem with the pole $z_{\infty,5}$:

$$\hat{H}_{1,2}(z) \;=\; \frac{1 + 2z^{-1} + z^{-2}}{1 + 0.0672z^{-1} + 0.7793z^{-2}}, \qquad \hat{H}_{3,4}(z) \;=\; \frac{1 + 2z^{-1} + z^{-2}}{1 - 0.4379z^{-1} + 0.3849z^{-2}},$$

$$\hat{H}_5(z) \;=\; \frac{1 + z^{-1}}{1 - 0.4113z^{-1}}.$$

Structures for IIR systems – Part 10

Cascade-form structures – Part 4

Example (continued):

- For the implementation on a fixed-point DSP it is advantageous to ensure that all stages have similar amplification in order to avoid numerical problems. Therefore, all subsystems are scaled such that they have approximately the same amplification for low frequencies:

$$
\begin{aligned}
H_1(z) &= \frac{\tilde{H}_5(z)}{\tilde{H}_5(z=1)} = \frac{0.2943 + 0.2943 z^{-1}}{1 - 0.4113 z^{-1}}, \\
H_2(z) &= \frac{\tilde{H}_{3,4}(z)}{\tilde{H}_{3,4}(z=1)} = \frac{0.2367 + 0.4735 z^{-1} + 0.2367 z^{-2}}{1 - 0.4379 z^{-1} + 0.3849 z^{-2}}, \\
H_3(z) &= \frac{\tilde{H}_{1,2}(z)}{\tilde{H}_{1,2}(z=1)} = \frac{0.4616 + 0.9233 z^{-1} + 0.4616 z^{-2}}{1 + 0.0672 z^{-1} + 0.7793 z^{-2}}.
\end{aligned}
$$

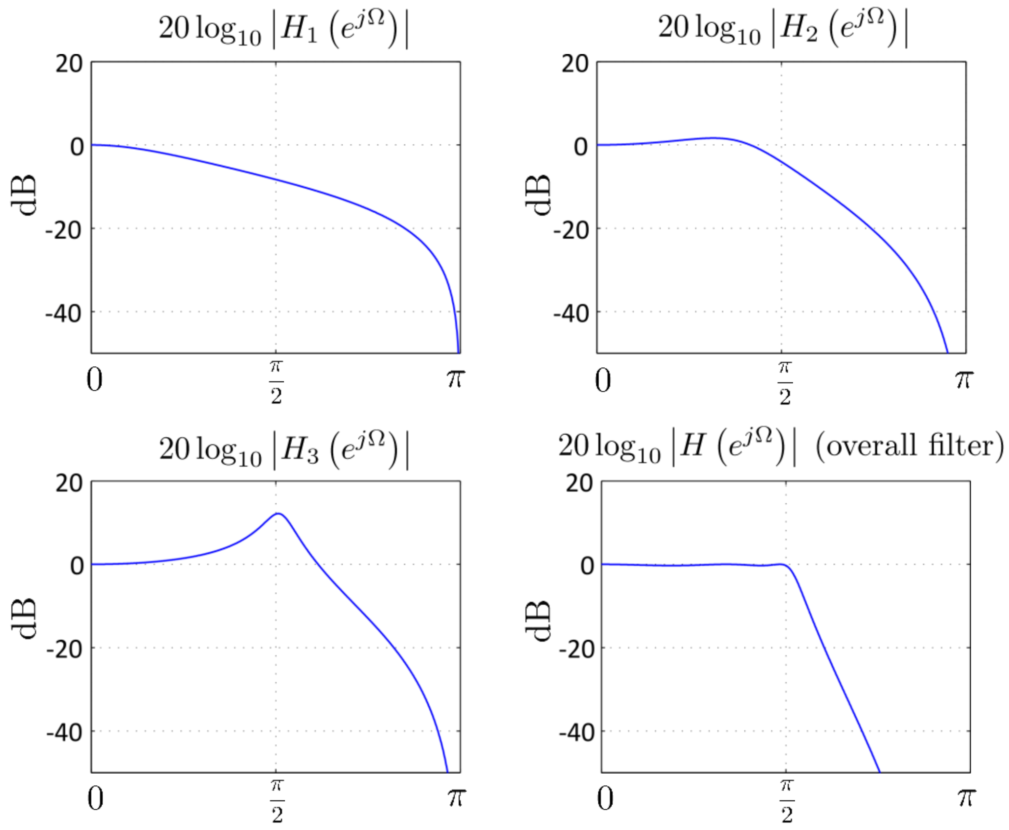
Remark:

The position of the subsystems in the cascade is in principle arbitrary. However, here the poles of $\tilde{H}_{1,2}(z)$ are closest to the unit circle. Thus, using a fixed-point DSP may lead more likely to numerical overflow compared to $\tilde{H}_{3,4}(z)$ and $\tilde{H}_5(z)$. Therefore, it is advisable to realize the most sensible filter as the last subsystem.

Cascade-form structures – Part 5

Example (continued):

- Frequency responses:

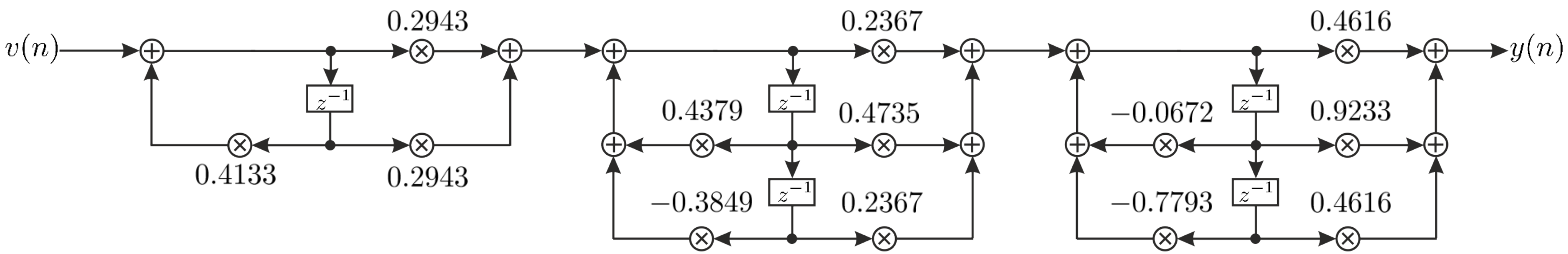


Structures for IIR systems – Part 12

Cascade-form structures – Part 6

Example (continued):

- Resulting signal flow graph:



Structures for IIR systems – Part 13

Parallel-form structures – Part 1

An alternative to the factorization of a general transfer function is to use a partial-fraction expansion, which leads to a ***parallel-form structure***.

- We assume distinct poles (which is quite well satisfied in practice). Then, the ***partial fraction expansion*** of a transfer function $H(z)$ with numerator degree N is given as

$$H(z) \;=\; A_0 + \sum_{i=1}^{N} \frac{A_i}{1 - z_{\infty,i} z^{-1}},$$

where $A_i,\; i \in \{1, \ldots, N\}$ are the coefficients (residues) in the partial fraction expansion and $A_0 = b_N / a_N$.

- We further assume that we have only ***real-valued coefficients***, such that we can combine pairs of complex-conjugate poles to form a second order subsystem:

$$\frac{A_i}{1 - z_{\infty,i} z^{-1}} + \frac{A_i^*}{1 - z_{\infty,i}^* z^{-1}} \;=\; \frac{2\mathrm{Re}\{A_i\} - 2\mathrm{Re}\{A_i z_{\infty,i}^*\} z^{-1}}{1 - 2\mathrm{Re}\{z_{\infty,i}\} z^{-1} + |z_{\infty,i}|^2 \, z^{-2}} \;=\; \frac{b_0 + b_1 z^{-1}}{1 - a_1 z^{-1} + a_2 z^{-2}},$$

with $i \in \{1, \ldots, N\}$.

Structures for IIR systems – Part 14

Parallel-form structures – Part 2

- Two real-valued poles can also be combined to a second order transfer function:

$$\frac{A_i}{1 - z_{\infty,i} z^{-1}} + \frac{A_j}{1 - z_{\infty,j} z^{-1}} = \frac{(A_i + A_j) - (A_i z_{\infty,j} + A_j z_{\infty,i})\, z^{-1}}{1 - (z_{\infty,i} + z_{\infty,j})\, z^{-1} + (z_{\infty,i}\, z_{\infty,j})\, z^{-2}}$$

$$= \frac{b_0 + b_1 z^{-1}}{1 - a_1 z^{-1} + a_2 z^{-2}},$$

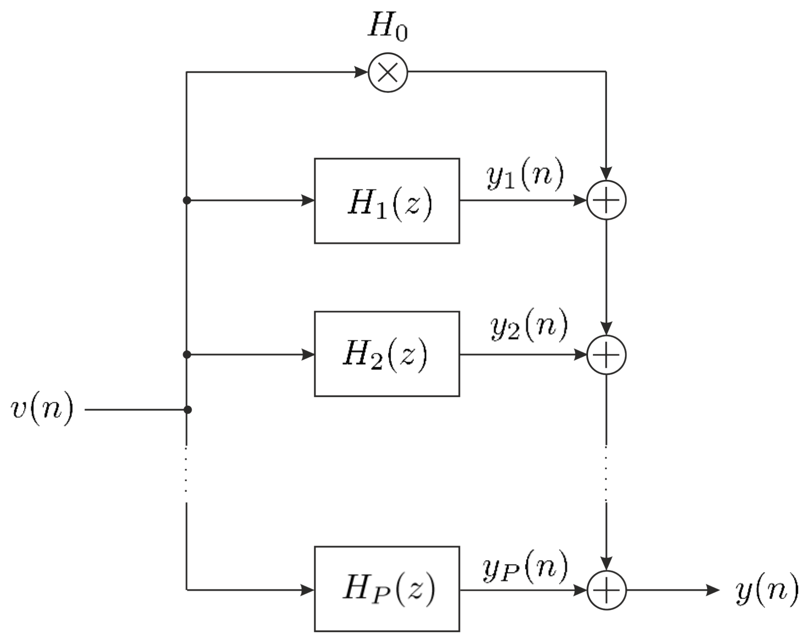
with $i, j \in \{1, \ldots, N\}$.

- If N is odd, there is one real-valued pole left, which leads to one first order partial fraction see example).

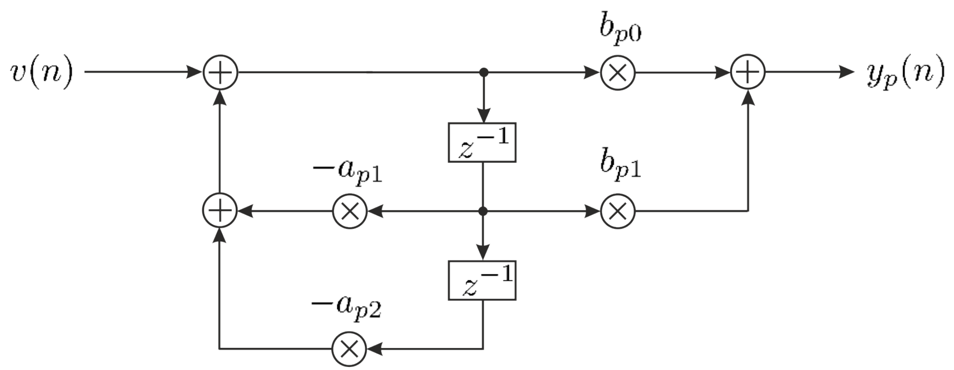
Structures for IIR systems – Part 15

Parallel-form structures – Part 3

Signal flow graph of the ***parallel structure***:



Signal flow graph of a ***second order section***:



$$p = 1, \ldots, P$$

Structures for IIR systems – Part 16

Parallel-form structures – Part 4

Example:

Consider again the 5th order Chebyshev filter with the transfer function

$$H(z) \;=\; 0.03217 \cdot \frac{1 + 5z^{-1} + 10z^{-2} + 10z^{-3} + 5z^{-4} + z^{-5}}{1 - 0.782z^{-1} - 1.2872z^{-2} - 0.7822z^{-3} + 0.4297z^{-4} - 0.1234z^{-5}}.$$

The partial fraction expansion can be given as:

$$H(z) \;=\; -0.2607 + \frac{A_1}{1 - z_{\infty,1}z^{-1}} + \frac{A_1^*}{1 - z_{\infty,1}^*z^{-1}} + \frac{A_3}{1 - z_{\infty,3}z^{-1}} + \frac{A_3^*}{1 - z_{\infty,3}^*z^{-1}} + \frac{A_5}{1 - z_{\infty,5}z^{-1}},$$

with the poles and residues

$$\begin{aligned} z_{\infty,1} &= -0.0336 + j0.8821, & A_1 &= 0.1 + j0.0941, \\ z_{\infty,3} &= 0.219 + j0.5804, & A_3 &= -0.5533 + j0.00926, \\ z_{\infty,5} &= 0.4113, & A_5 &= 1.1996. \end{aligned}$$

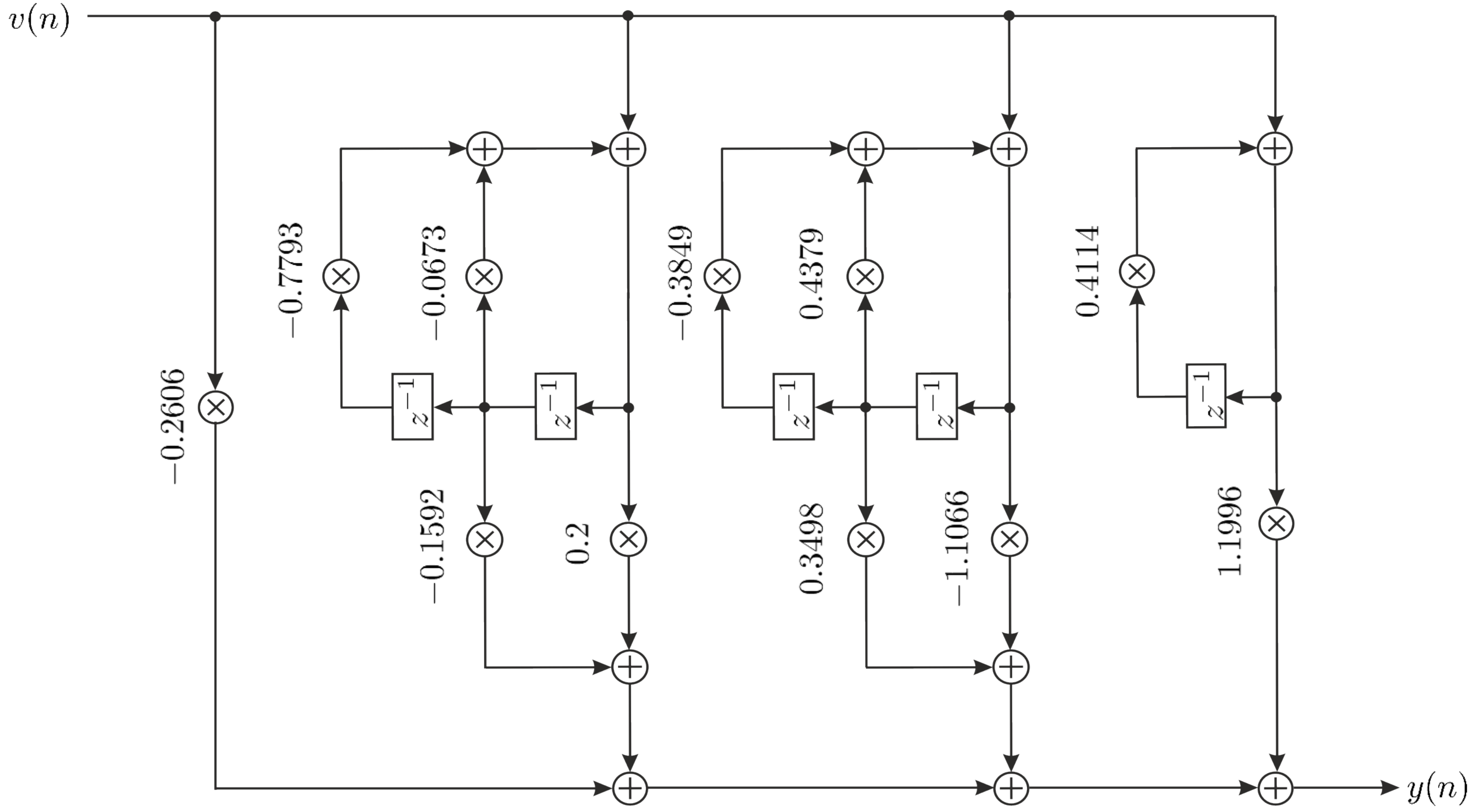
The resulting transfer function is:

$$H(z) \;=\; -0.2607 + \frac{0.2 - 0.1592z^{-1}}{1 + 0.0673z^{-1} + 0.7793z^{-2}} + \frac{-1.1066 + 0.3498z^{-1}}{1 - 0.4379z^{-1} + 0.3849z^{-2}} + \frac{1.1996}{1 - 0.4114z^{-1}}.$$

Structures for IIR systems – Part 17

Parallel-form structures – Part 5

Example (continued): The resulting signal flow graph



Structures for IIR systems – Part 18

Cascaded and Parallel-form structures

Partner work – Please think about the following question and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What are the differences of the cascaded and parallel form structures?

  ……

  ……

- Can you think of applications / hardware architectures where you would prefer on of the structures? What do you need to know about the hardware in order to make such a decision?

  ……

  ……

  ……

  ……

Errors resulting from rounding and truncation – Part 1

In this section we discuss the effects of a fixed-point digital filter implementation on the system performance.

Number representation in fixed-point format:

A real number v can be represented as

$$v = [\beta_{-A}, \ldots, \beta_{-1}, \beta_0, \beta_1, \ldots, \beta_B]_r = \sum_{l=-A}^{B} \beta_l r^{-l},$$

where β_l is the ***digit***, r is the ***radix*** (***base***), A the number of integer digits, and B the number of fractional digits. Example:

$$[101.01]_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2}.$$

Most important in digital signal processing:

- Binary representation with $r = 2$ and $\beta_l \in \{0, 1\}$, β_{-A} most significant bit (MSB) and β_B least significant bit (LSB).
- b-bit fraction format: $A = 0,\ B = b - 1$, binary point between β_0 and β_1 $\rightarrow$ numbers between 0 and $2 - 2^{-b+1}$ are possible.

Errors resulting from rounding and truncation – Part 2

Number representation in fixed-point format (continued):

Positive numbers are represented as

$$v = [0.\beta_1\beta_2\ldots\beta_{b-1}] = \sum_{l=-1}^{b-1} \beta_l 2^{-l}.$$

The negative fraction

$$v = [-0.\beta_1\beta_2\ldots\beta_{b-1}] = -\sum_{l=-1}^{b-1} \beta_l 2^{-l}.$$

can be represented with one of the three following formats:

- ***Signs-magnitude format:***

$$v_{\mathrm{SM}} = [1.\beta_1\beta_2\ldots\beta_{b-1}] \text{ for } v < 0.$$

- ***One's-complement format:***

$$v_{1\mathrm{C}} = [1.\bar{\beta}_1\bar{\beta}_2\ldots\bar{\beta}_{b-1}] \text{ for } v < 0.$$

- ***Two's complement format:***

$$v_{2\mathrm{C}} = [1.\bar{\beta}_1\bar{\beta}_2\ldots\bar{\beta}_{b-1}] \oplus [00\ldots01] \text{ for } v < 0,$$

where $\oplus$ denotes a binary addition.

… with $\bar{\beta} = 1 - \beta$ *…*

Most DSPs use two's-complement arithmetic (because of a good "temporary overflow" handling)

Coefficient Quantization and Rounding Effects – Part 3

Errors resulting from rounding and truncation – Part 3

Number representation in fixed-point format (continued):

Example:

Express the fraction $7/8$ and $-7/8$ in sign-magnitude, two's complement and one's complement.

- $v = 7/8$ can be represented as $2^{-1} + 2^{-2} + 2^{-3}$, such that $v = [0.111]$.
- $v = -7/8$ can be represented
  - in sign-magnitude format as $v_{SM} = [1.111]$,
  - in one's complement format as $v_{1C} = [1.000]$,
  - in two's complement format as $v_{2C} = [1.000] \oplus [0.001] = [1.001]$.

Coefficient Quantization and Rounding Effects – Part 4

Errors resulting from rounding and truncation – Part 4

Truncation and rounding:

Problem: Multiplication of two b-bit numbers yield a result of length $(2b-1)$
$\rightarrow$ truncation/rounding necessary
$\rightarrow$ can again be regarded as ***quantization*** of the (filter) coefficient v

Suppose that we have a fixed-point realization in which a number v is quantized from b_u to b bits.

We first discuss the truncation case. Let the truncation error be defined as $E_\mathrm{t} = Q_\mathrm{t}[v] - v$.

- For positive numbers the error is

$$-2^{-(b-1)} \leq E_\mathrm{t} \leq 0.$$

Truncation leads to a number smaller than the non-quantized number.

- For negative numbers and the sign-magnitude representation the error is

$$0 \leq E_\mathrm{t} \leq 2^{-(b-1)}.$$

Truncation reduces the magnitude of the number.

Coefficient Quantization and Rounding Effects – Part 5

Errors resulting from rounding and truncation – Part 5

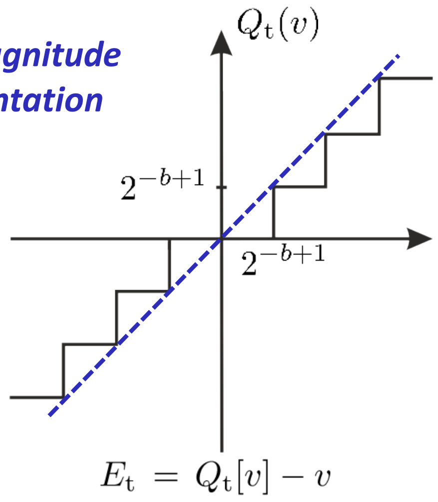
Truncation and rounding (continued):

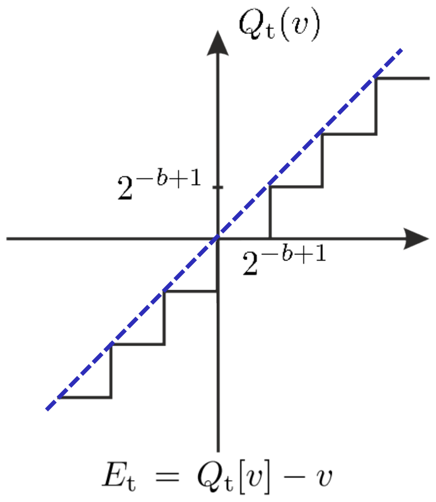
- For negative numbers in the two's complement case the error is

$$-2^{-(b-1)} \leq E_{\mathrm{t}} \leq 0.$$

- Quantization characteristics for a continuous input signal v:

Sign-magnitude representation





Two's-complement representation

Coefficient Quantization and Rounding Effects – Part 6

Errors resulting from rounding and truncation – Part 6

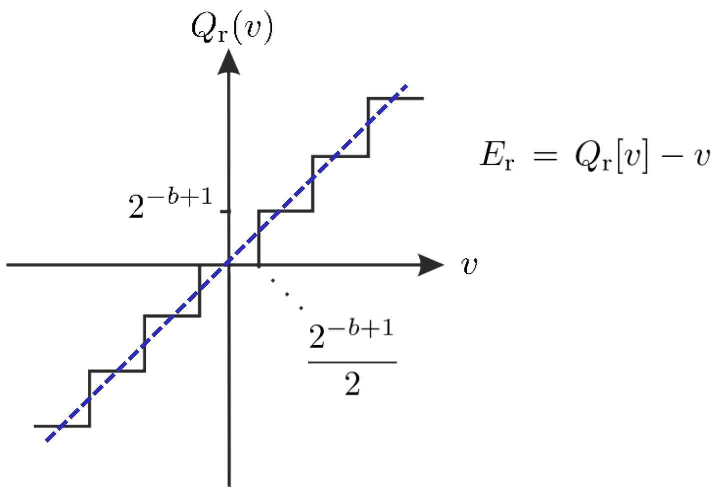
Truncation and rounding (continued):

Rounding case: The Rounding error is defined as $E_\mathrm{r} = Q_\mathrm{r}[v] - v.$

- Rounding affects only the magnitude of the number and is independent from the type of fixed-point realization.
- Rounding error is symmetric around zero and falls in the range

$$-\frac{1}{2}\cdot 2^{-(b-1)} \le E_\mathrm{r} \le \frac{1}{2}\cdot 2^{-(b-1)}.$$

- Quantization characteristic function:



Coefficient Quantization and Rounding Effects – Part 7

Numerical overflow – Part 1:

If a number is larger/smaller than the maximal/minimal possible number representation,

- $\pm\left(1-2^{-b+1}\right)$ for sign magnitude and one's-complement arithmetic,
- -1 and $1-2^{-b+1}$, resp., for two's-complement arithmetic,

we speak of an ***overflow***/***underflow*** condition.

Overflow example in two's-complement arithmetic (range $-8,\ \ldots,\ 7$)

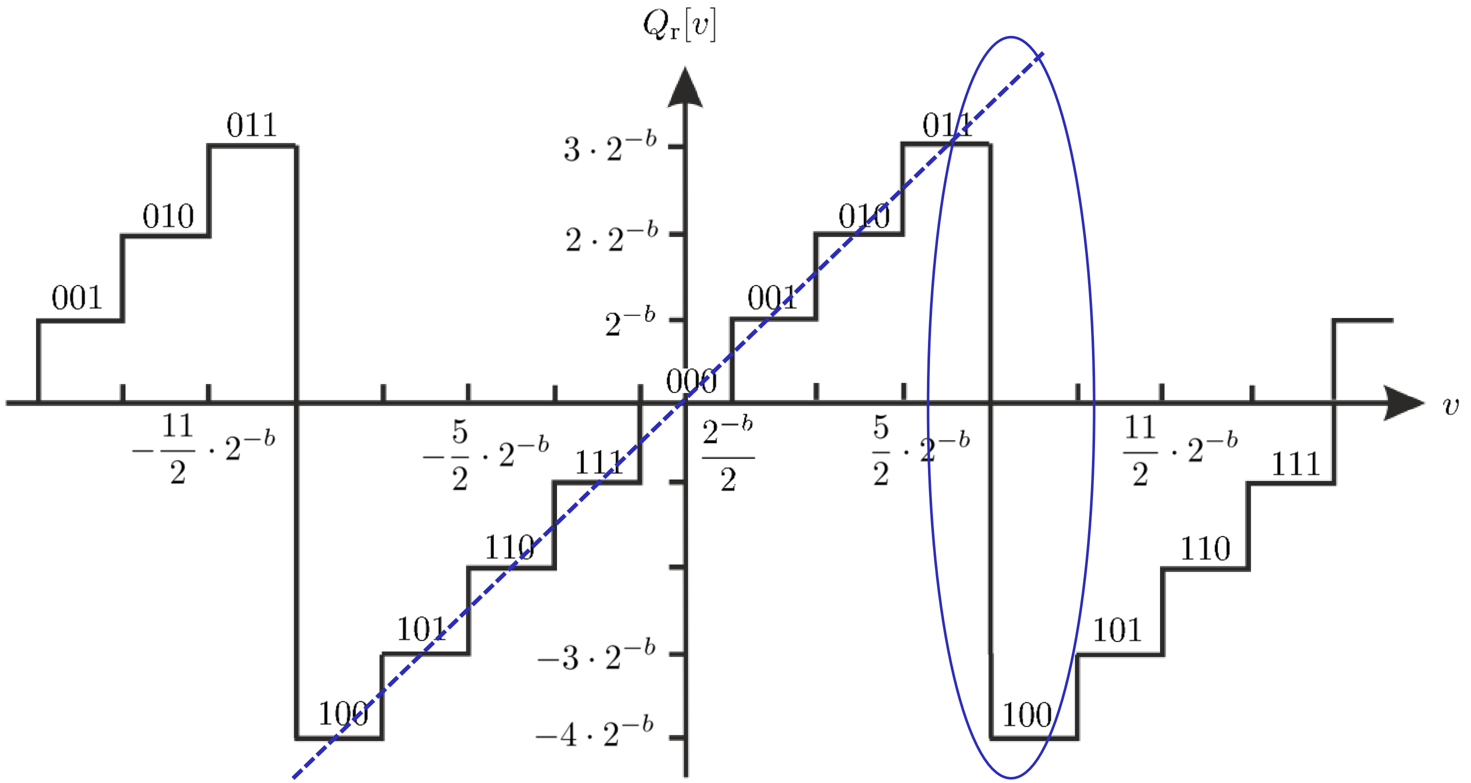
$$\underbrace{[0.111]}_{7} \oplus \underbrace{[0.001]}_{1} \;=\; \underbrace{[1.000]}_{-8}$$

$\Rightarrow$ The resulting error can be very large when overflow/underflow occurs.

Coefficient Quantization and Rounding Effects – Part 8

Numerical overflow – Part 2

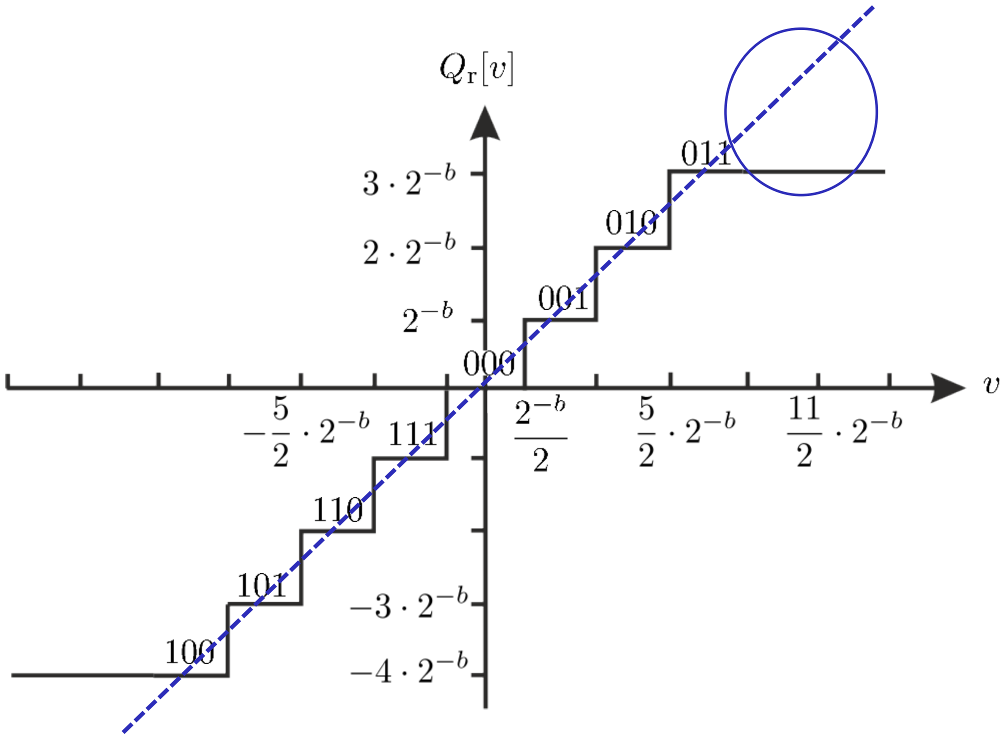
Two's-complement quantizer for $b = 3$:



Coefficient Quantization and Rounding Effects – Part 9

Numerical overflow – Part 3

Alternative: ***saturation*** or ***clipping***. The error does not increase abruptly in magnitude when overflow/underflow occurs:



Disadvantage: "Summation property" of the two's-complement representation is violated.

Coefficient Quantization and Rounding Effects

Partner work – Please think about the following question and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What are the most prominent representations in fixed-point arithmetic?

  ……

  ……

- How large / small can be the result of an addition / multiplication of two fixed-point numbers (e.g. each being represented by a 16 bit value)?

  ……

  ……

- What do you know about number representations in floating-point arithmetic?

  ……

  ……

Coefficient Quantization and Rounding Effects – Part 11

Coefficient quantization errors – Part 1

- In a DSP/hardware realization of an FIR/IIR filter the accuracy is limited by the word length of the computer
  $\Rightarrow$ Coefficients obtained from a design algorithm have to be quantized.
- Word length reduction of the coefficients leads to different poles and zeros to the desired ones. This may lead to
  - modified frequency response with decreased selectivity,
  - stability problems.

Sensitivity to quantization of filter coefficients

Direct form realization, quantized coefficients:

$$\begin{aligned} \bar{a}_i &= a_i + \Delta a_i, \quad i = 1, ..., N, \\ \bar{b}_i &= b_i + \Delta b_i, \quad i = 0, ..., N, \end{aligned}$$

Δa_i and Δb_i represent the quantization errors.

Coefficient Quantization and Rounding Effects – Part 12

Effect of quantization of coefficients:

Matlab example for “robust” filter design …

Coefficient quantization errors – Part 2

Sensitivity to quantization of filter coefficients (continued)

As an example, we are interested in the deviation $\Delta z_{\infty,i} = z_{\infty,i} - \bar{z}_{\infty,i}$, when the denominator coefficients a_i are quantized ($\bar{z}_{\infty,i}$ denotes the resulting pole after quantization). It can be shown that this expression can be expressed as (Proakis, Manolakis, 1996, pp. 569):

$$\Delta z_{\infty,i} = -\sum_{n=1}^{N} \frac{z_{\infty,i}^{N-n}}{\prod\limits_{l=1,l\neq i}^{N} (z_{\infty,i} - z_{\infty,l})} \Delta a_n, \quad i = 1, \ldots, N.$$

Basic derivation on the blackboard!

From this equation we can observe the following:

- By using the direct form, each single pole deviation $\Delta z_{\infty,i}$ depends on all quantized denominator coefficients $\bar{a}_i$.
- The error $\Delta z_{\infty,i}$ can be minimized by maximizing the distance $|z_{\infty,i} - z_{\infty,l}|$ between the poles $z_{\infty,i}$ and $z_{\infty,l}$.

Coefficient Quantization and Rounding Effects – Part 14

Coefficient quantization errors – Part 3

Sensitivity to quantization of filter coefficients (continued)

$\Rightarrow$ Splitting the filter into single or double pole sections (first or second order transfer functions):

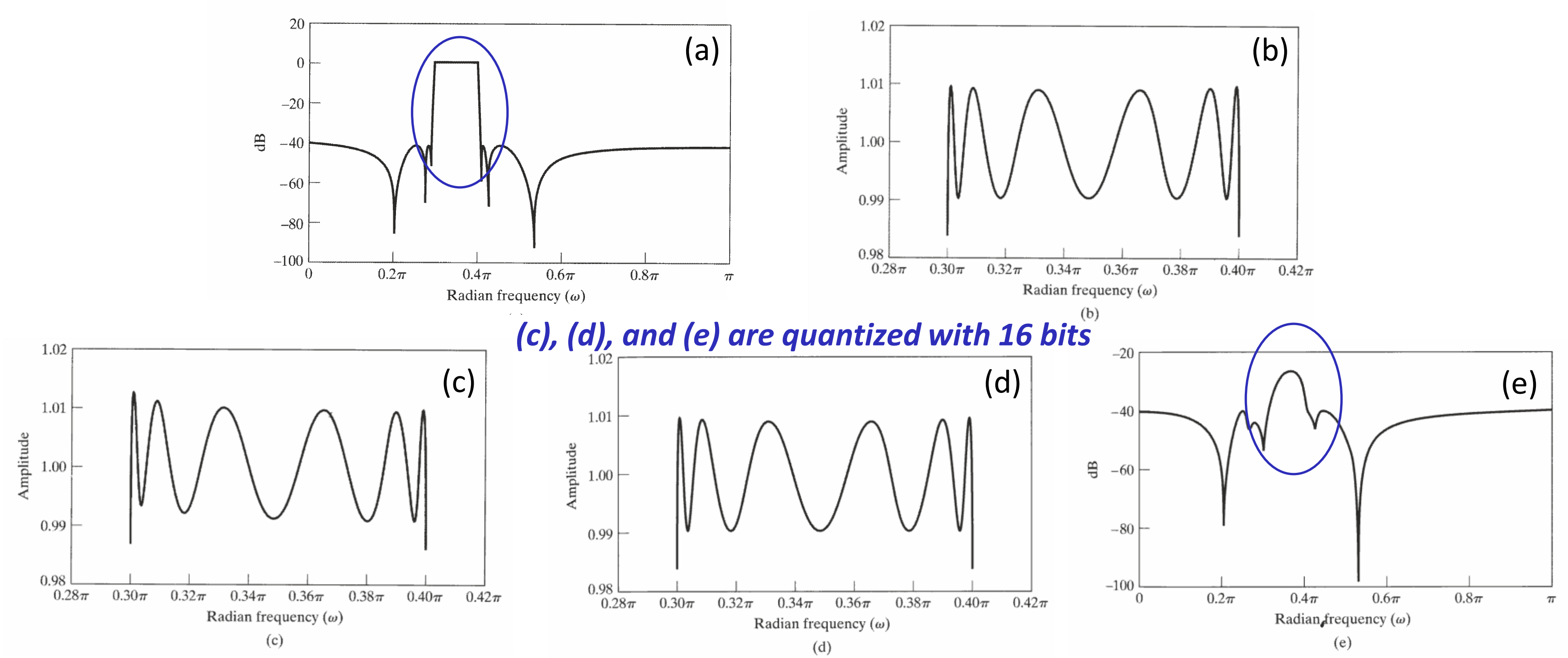
- Combining the poles $z_{\infty,i}$ and $z^*_{\infty,i}$ into a second order section leads to a small perturbation error $\Delta z_{\infty,i}$, since complex conjugate poles are normally sufficient far apart.
- $\Rightarrow$ Realization in ***cascade*** or ***parallel*** form:

  The error of a particular pole pair $z_{\infty,i}$ and $z^*_{\infty,i}$ is independent of its distance from the other poles of the transfer function.

Coefficient Quantization and Rounding Effects – Part 15

Coefficient quantization errors – Part 4

Example: Effects of coefficient quantization



Elliptic filter of order $N = 12$ (Example taken from [Oppenheim, Schafer 1999])

Unquantized: (a) Magnitude frequency response
(b) Passband detail

Quantized: (c) Passband detail for cascade structure
(d) Passband detail for parallel structure
(e) Magnitude frequency response for direct structure

Coefficient Quantization and Rounding Effects – Part 16

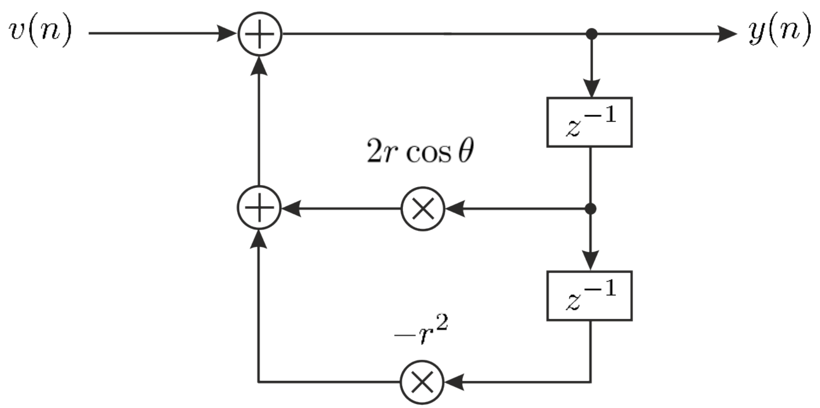
Coefficient quantization errors – Part 5

Pole locations of quantized second order sections

Consider a two-pole filter with the transfer function

$$H(z) \quad = \quad \frac{1}{1-(2r\cos\theta)z^{-1}+r^2z^{-2}}.$$

Poles: $z_{\infty,1,2} = re^{\pm j\theta}$, coefficients: $a_1 = -2r\cos\theta, a_2 = r^2$, stability condition: $|r| \leq 1$

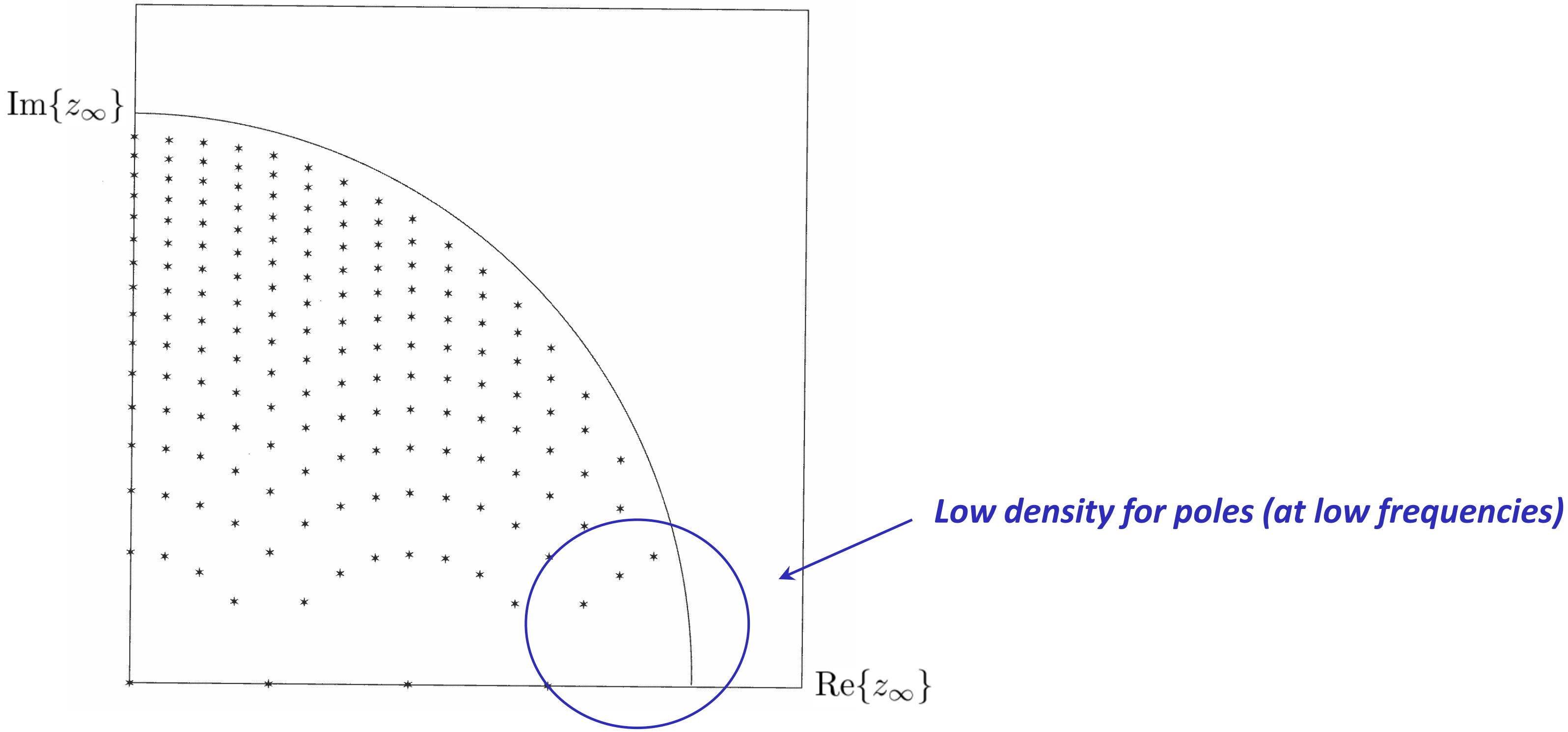


Coefficient Quantization and Rounding Effects – Part 17

Coefficient quantization errors – Part 6

Pole locations of quantized second order sections (continued)

Quantization of a_1 and a_2 with $b = 4$ bits $\rightarrow$ possible pole positions:



Coefficient Quantization and Rounding Effects – Part 18

Coefficient quantization errors – Part 7

Pole locations of quantized second order sections (continued)

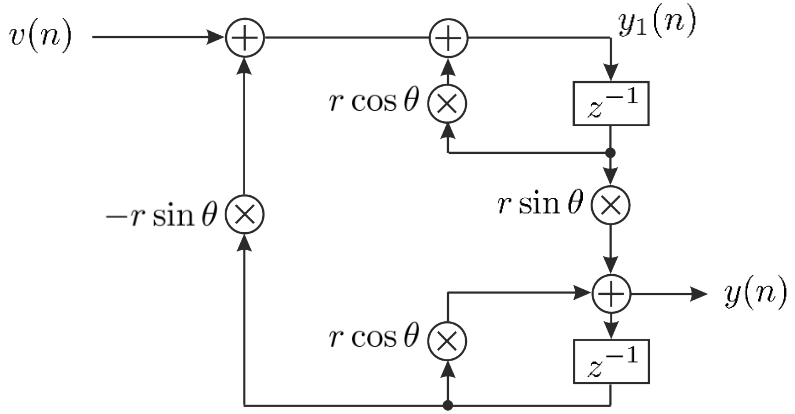
- Non-uniformity of the pole position is due to the fact that $a_2 = r^2$ is quantized, while the pole locations $z_{\infty,1,2} = re^{\pm j\theta}$ are proportional r.
- Sparse set of possible pole locations around $\theta = 0$ and $\theta = \pi$. Disadvantage for realizing lowpass filters where the poles are normally clustered near $\theta = 0$.

Alternative: ***Coupled-form realization***

$$\begin{aligned} y_1(n) &= v(n) + r\cos\theta\ y_1(n-1) - r\sin\theta\ y(n-1), \\ y(n) &= r\sin\theta\ y_1(n-1) + r\cos\theta\ y(n-1), \end{aligned}$$

← ***Analysis on the blackboard***

Which corresponds to the following signal flow graph:



Coefficient Quantization and Rounding Effects – Part 19

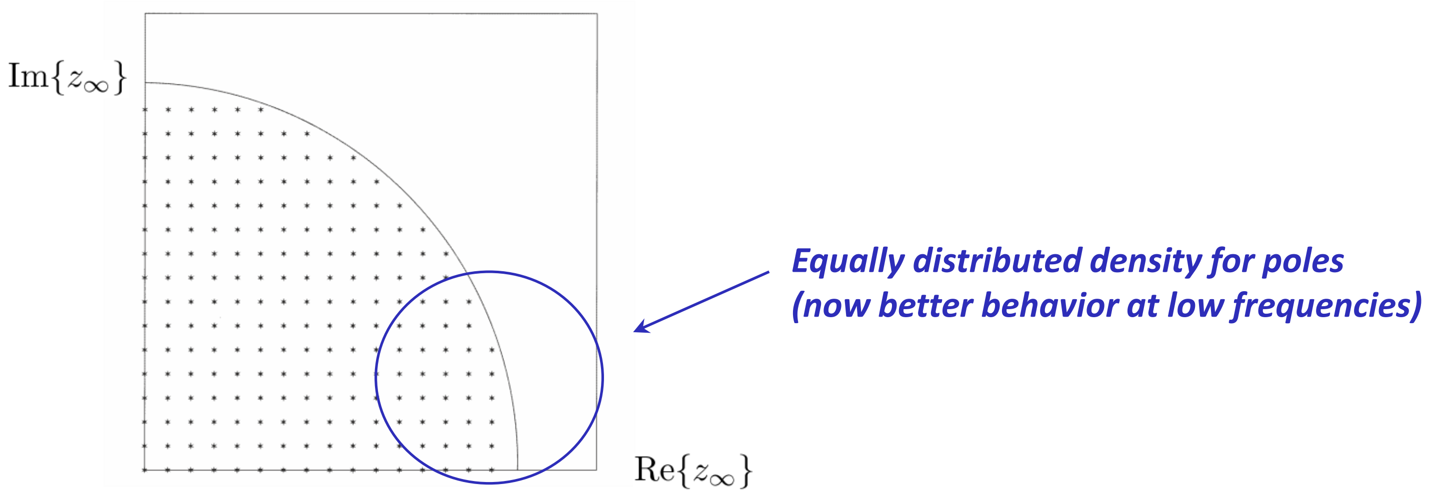
Coefficient quantization errors – Part 8

Pole locations of quantized second order sections (continued)

By transforming the equations into the z-domain, the transfer function of the filter can be obtained as

$$H(z) \;=\; \frac{Y(z)}{V(z)} \;=\; \frac{r\sin\theta\; z^{-1}}{1-2r\cos\theta\; z^{-1}+r^2z^{-2}}$$

- We can see from the signal flow graph that the two coefficients $r\sin\theta$ and $r\cos\theta$ are now linear in r, such that a quantization of these parameters lead to equally spaced pole locations in the z-plane:



- Disadvantage. Increased computational complexity compared to the direct form.

Coefficient Quantization and Rounding Effects – Part 20

Coefficient quantization errors – Part 9

Cascade or parallel form

Cascade form:

$$H(z) = \prod_{p=1}^{P} \frac{b_{p0} + b_{p1} z^{-1} + b_{p2} z^{-2}}{1 + a_{p1} z^{-1} + a_{p2} z^{-2}}$$

Parallel form:

$$H(z) = A_0 + \sum_{p=1}^{P} \frac{c_{p0} + c_{p1} z^{-1}}{1 + a_{p1} z^{-1} + a_{p2} z^{-2}}$$

- ***Cascade form***: Only the numerator coefficients b_{pi} of an individual section determine the perturbation of the corresponding zero locations $\Longrightarrow$ ***direct control over the poles and zeros***
- ***Parallel form***: A particular zero is affected by quantization errors in the numerator and denominator coefficients of all individual sections $\Longrightarrow$ numerator coefficients c_{p0} and c_{p1} do not specify the position of a zero directly, ***direct control over the poles only***.

$\Rightarrow$ ***Cascaded structures are more robust against coefficient quantization and should be used in most cases.***

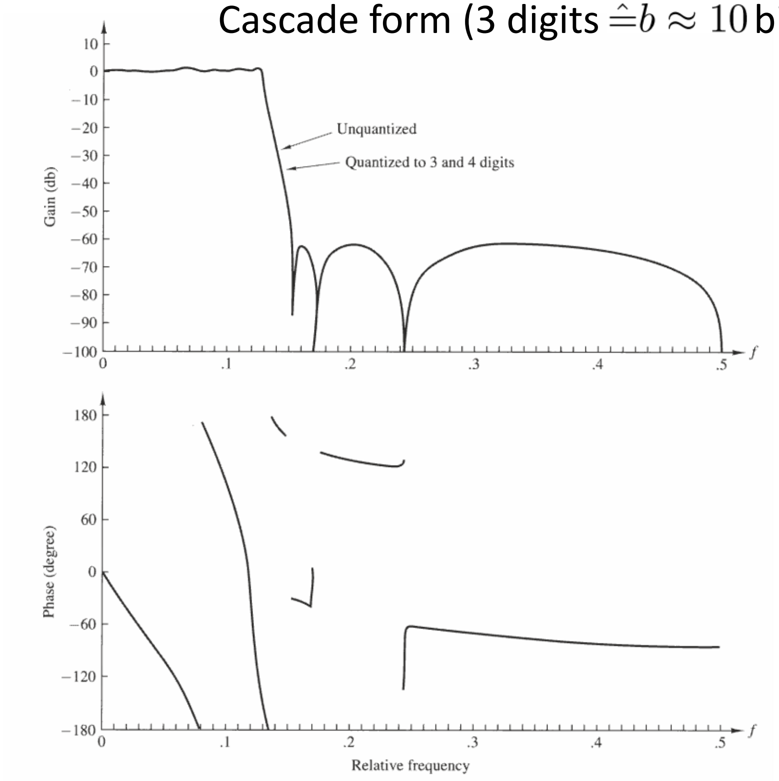
Coefficient Quantization and Rounding Effects – Part 21

Coefficient quantization errors – Part 10

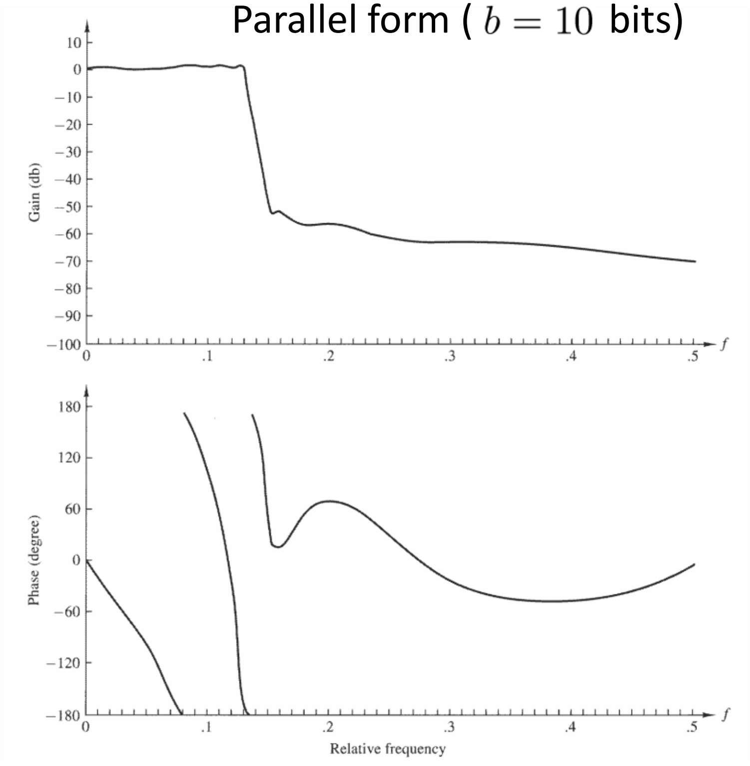
Cascade or parallel form (continued)

Example: Elliptic filter of order $N = 7$, frequency and phase response ([Proakis, Manolakis 96])

Cascade form (3 digits $\hat{=} b \approx 10$ bits)



Parallel form ($b = 10$ bits)



Coefficient Quantization and Rounding Effects – Part 22

Coefficient quantization errors – Part 11

Coefficient quantization in FIR systems

In FIR systems we only have to deal with the locations of the zeros, since for causal filters all poles are at $z = 0$.

Remarks:

- For FIR filters an expression analogous to the deviation and the original and quantized poles can be derived for the zeros.
  $\Longrightarrow$ FIR filters might also be realized in ***cascade form*** according to

$$H(z) \; = \; H_0 \prod_{p=1}^{P} \left(1 + b_{p1} z^{-1} + b_{p2} z^{-2}\right).$$

  with second order subsections, in order to limit the effects of coefficient quantization to zeros of the actual subsection only.
- However, since the ***zeros are more or less uniformly spread*** in the z-plane, in many cases the ***direct form is also used with quantized coefficients***.
- For a ***linear-phase filter*** that has a symmetric or asymmetric impulse response, quantization does ***not affect the phase*** characteristics, but ***only the magnitude***.

Coefficient Quantization and Rounding Effects – Part 23

Coefficient quantization errors – Part 12

Partner work – Please think about the following question and try to find answers
(first group discussions, afterwards broad discussion in the whole group).

- What are the drawbacks of parallel filter structures? Are there also advantages?

  ……

  ……

- Why are FIR filters not as critical in terms of precision compared to IIR filters?

  ……

  ……

- Why are in today's processors sometimes the direct structures better than cascaded structures for FIR filters (answer can not be found in the slides)?

  ……

  ……

Coefficient Quantization and Rounding Effects – Part 24

Zero-input limit cycles – Part 1

- Stable IIR filters implemented with infinite-precision arithmetic: If the excitation becomes zero and remains zero for $n > n_0$ then the output of the filter will decay asymptotically towards zero.
- Same system implemented with fixed-point arithmetic: Output may oscillate indefinitely with a periodic pattern while the input remains equal to zero: ***Zero-input limit cycle behavior***, due to nonlinear quantizers in the feedback loop or overflow of additions.

In the following the effects are shown with two examples:

Limit cycles due to round-off truncation

Given: First-order system with the difference equation

$$y(n) \;=\; ay(n-1) + v(n), \quad |a| < 1.$$

Register length for storing a and the intermediate results: 4 bits (sign bit plus 3 fractional bits)
$\Rightarrow$ product $ay(n-1)$ must be rounded or truncated to 4 bits, before adding to $v(n)$.

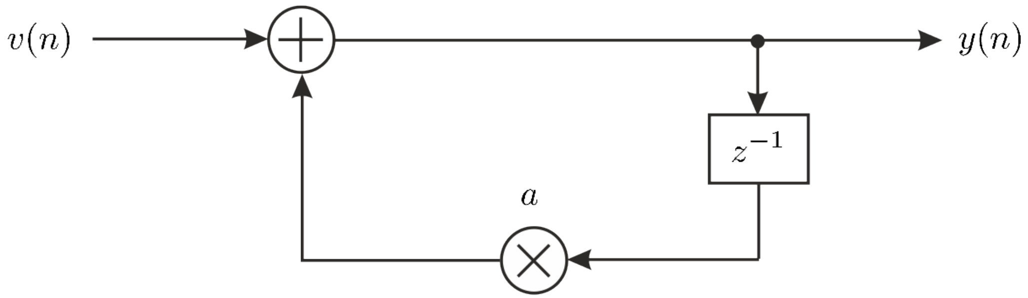
Coefficient Quantization and Rounding Effects – Part 25

Zero-input limit cycles – Part 2

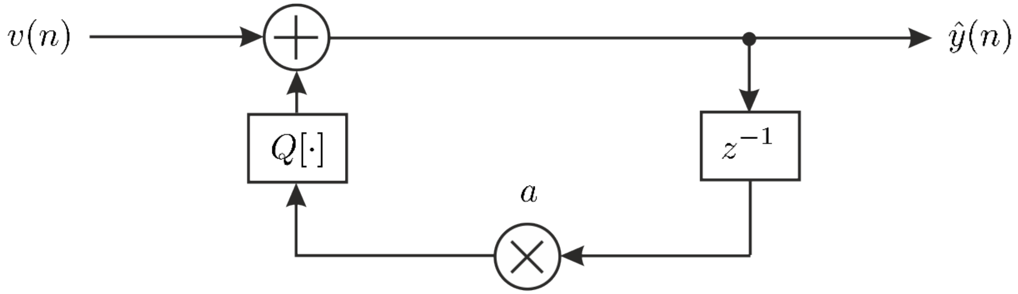
Limit cycles due to round-off truncation (continued)

Signal flow graphs:

Infinite-precision system:



Nonlinear system due to quantization:



Coefficient Quantization and Rounding Effects – Part 26

Zero-input limit cycles – Part 3

Limit cycles due to round-off truncation (continued)

Nonlinear difference equation ($Q[\cdot]$ represents two's-complement rounding):

$$\hat{y}(n) = Q\left[a\hat{y}(n-1)\right] + v(n).$$

Suppose we have

$$a = \frac{1}{2} = [0.100], \quad v(n) = \frac{7}{8}\gamma_0(n) = [0.111]\gamma_0(n).$$

Then:

$$\begin{aligned}
\hat{y}(0) &= 7/8 = [0.111] \\
\hat{y}(1) &= Q\left[a\hat{y}(0)\right] = Q\left[[0.100] \times [0.111]\right] = Q\left[[0.011100]\right] = Q\left[7/16\right] \\
&= [0.100] = 1/2 \\
\hat{y}(2) &= Q\left[a\hat{y}(1)\right] = [0.010] = 1/4 \\
\hat{y}(3) &= Q\left[a\hat{y}(2)\right] = [0.001] = 1/8 \\
\hat{y}(4) &= Q\left[a\hat{y}(3)\right] = Q\left[[0.000100]\right] = [0.001] = 1/8
\end{aligned}$$

Quantization with rounding (+ 0.000100)

⇒ A constant steady value is obtained for $n \geq 3$.

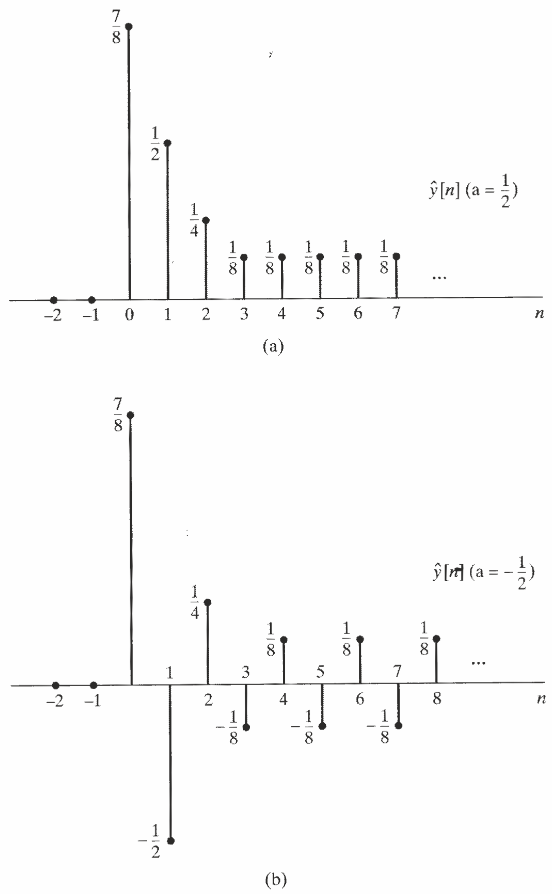
For $a = -1/2$ we have a periodic steady-state oscillation between $-1/8$ and $1/8$.

⇒ Such periodic outputs are called ***limit cycles***.

Coefficient Quantization and Rounding Effects – Part 27

Zero-input limit cycles – Part 4

Limit cycles due to round-off truncation (continued)



(a)

(b)

From [Oppenheim, Schafer, 1999]

Coefficient Quantization and Rounding Effects – Part 28

Zero-input limit cycles – Part 5

Limit cycles due to overflow

Consider a second-order system realized by the difference equation:

$$\hat{y}(n) \;=\; v(n) + Q\left[a_1 \hat{y}(n-1)\right] + Q\left[a_2 \hat{y}(n-2)\right].$$

$Q[\cdot]$ represents two's-complement rounding with one sign and 3 fractional digits.
Overflow can occur with the two's-complement addition of the products.
Suppose that

$$\begin{aligned} a_1 &= 3/4 = [0.110], \quad a_2 = -3/4 = [1.010], \\ \hat{y}(-1) &= 3/4 = [0.110], \quad \hat{y}(-2) = -3/4 = [1.010], \\ v(n) &= 0 \text{ for } n \geq 0. \end{aligned}$$

Then we have:

$$\begin{aligned} \hat{y}(0) &= Q\left[[0.110] \times [0.110]\right] + Q\left[[1.010] \times [1.010]\right] = Q\left[[0.100100]\right] + Q\left[[0.100100]\right] \\ &= [0.101] + [0.101] = [1.010] = -3/4 \\ \hat{y}(1) &= [1.011] + [1.011] = [0.110] = 3/4. \end{aligned}$$

$\Rightarrow \hat{y}(n)$ continues to oscillate unless an input is applied.

Coefficient Quantization and Rounding Effects – Part 29

Zero-input limit cycles – Part 6

Remarks

- Some solutions for avoiding limit cycles:
  - Use of structures which do not support limit-cycle oscillations.
  - Increasing the word length.
  - Use of a double-length accumulator and quantization after the accumulation of products.
- FIR-filters are limit-cycle free since there is no feedback involved in its signal flow graph.

Coefficient Quantization and Rounding Effects – Part 30

Zero-input limit cycles – Part 7

Partner work – Please think about the following question and try to find answers
(first group discussions, afterwards broad discussion in the whole group).

- What kind of limit cycles is more critical? Please, give reasons for your answer!

  ………

  ………

- What can you do to avoid overflow-based limit cycles?

  ………

  ………

- What can you do to avoid truncation-based limit cycles?

  ………

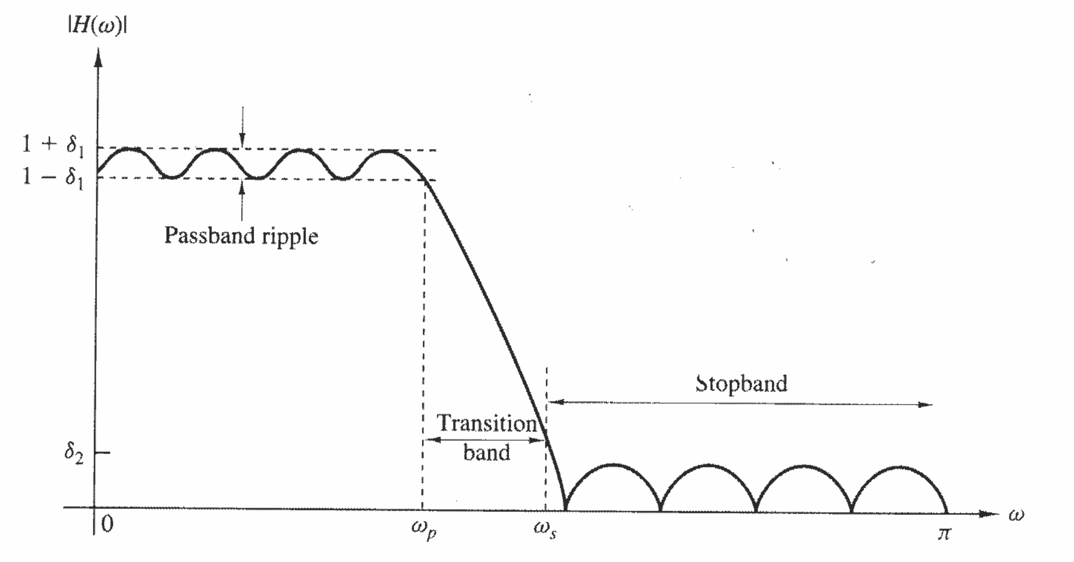
  ………

Design of FIR Filters – Part 1

General remarks (IIR and FIR filters) – Part 1

- Ideal filters are ***non-causal***, and thus physically unrealizable for real-time signal processing applications.
- Causality implies that the filter response $H\left(e^{j\Omega}\right)$ cannot have an infinitely sharp cut-off from passband to stopband, and that the stopband amplification can only be zero for a finite number of frequencies Ω.

Magnitude characteristics of physically realizable filter ($\Omega \rightarrow \omega$):



δ_1: passband ripple,

δ_2 : stopband ripple,

Ω_p : passband edge frequency,

Ω_s : stopband edge frequency

From [Proakis, Manolakis, 1996]

Design of FIR Filters – Part 2

General remarks (IIR and FIR filters) – Part 2

Filter design problem:

- Specify $\delta_1,\ \delta_2,\ \Omega_p$ and Ω_s corresponding to the desired application,
- Select the coefficients a_i and b_i (free parameters), such that the resulting frequency response $H\left(e^{j\Omega}\right)$ best satisfies the requirements for $\delta_1,\ \delta_2,\ \Omega_p$ and Ω_s.
- The degree which $H\left(e^{j\Omega}\right)$ approximates the specifications depends on the criterion for selecting the a_i and the b_i and also on the numerator and denominator degree N (the number of coefficients).

How we will continue:

- Before we will start of “optimal” design procedures, we will first focus on very ***simple design schemes***.
- However, due to their low complexity they are suitable for ***real-time filter design***.
- In addition, we will first focus on ***linear-phase FIR*** filters.

Linear-phase filters – Part 1

Important class of FIR filters, which we will mainly consider in the following.

Definition:

A filter is said to be a ***linear-phase filter***, if its impulse response satisfies the condition $(L = N+1)$:

$$h_i \;=\; \pm h_{L-1-i}.$$

With the definition $S \;=\; (L-1)/2$ and L ***odd***, this leads to a z-transform:

$$H(z) \;=\; \sum_{i=0}^{L-1} h_i z^{-i} \;=\; z^{-S}\left[h_S + \sum_{i=0}^{S-1} h_i \cdot \left(z^{(S-i)} \pm z^{-(S-i)}\right)\right].$$

For L ***even*** we have

$$H(z) \;=\; z^{-S} \sum_{i=0}^{L/2-1} h_i \cdot \left(z^{(S-i)} \pm z^{-(S-i)}\right).$$

Linear-phase filters – Part 2

Result from the last slide for an even filter length $(L-1 = 2S)$ and $h_i = h_{L-1-i}$:

$$H(z) \;=\; z^{-S} \sum_{i=0}^{S-1} h_i \cdot \left(z^{(S-i)} + z^{-(S-i)}\right).$$

When we now substitute z with z^{-1} and multiply both sides both sides by $z^{-(L-1)}$ we obtain with the definition of a linear-phase filter:

$$H(z^{-1}) \;=\; z^{S} \sum_{i=0}^{S-1} h_i \cdot \left(z^{-(S-i)} + z^{(S-i)}\right)$$

… multiplication of both sides with $z^{-(L-1)} = z^{-S}\, z^{-S}$ ***…***

$$z^{-(L-1)}\, H(z^{-1}) \;=\; z^{-S}\, z^{-S}\, z^{S} \sum_{i=0}^{S-1} h_i \cdot \left(z^{-(S-i)} + z^{(S-i)}\right)$$

… simplification and exchanging the order of the addends …

$$z^{-(L-1)}\, H(z^{-1}) \;=\; z^{-S} \sum_{i=0}^{S-1} h_i \cdot \left(z^{(S-i)} + z^{-(S-i)}\right)$$

… inserting the result from above …

$$z^{-(L-1)}\, H(z^{-1}) \;=\; H(z).$$

Linear-phase filters – Part 3

Generalizing the result of the previous slide for all four cases, leads to

$$z^{-(L-1)} H\left(z^{-1}\right) \; = \; \pm H(z),$$

which is the z-transform equivalent to the definition of a linear-phase filter.

Consequences:

- The roots of the polynomial $H(z)$ are identical to the roots of the polynomial $H(z^{-1})$:
  If $z_{0,i}$ is a zero of $H(z)$ then $z_{0,i}^{-1}$ is also a zero.
- If additionally the impulse response h_i is real-valued, the roots must occur in complex-conjugate pairs:
  If $z_{0,i}$ is a zero of $H(z)$ then $z_{0,i}^{*}$ is also a zero.

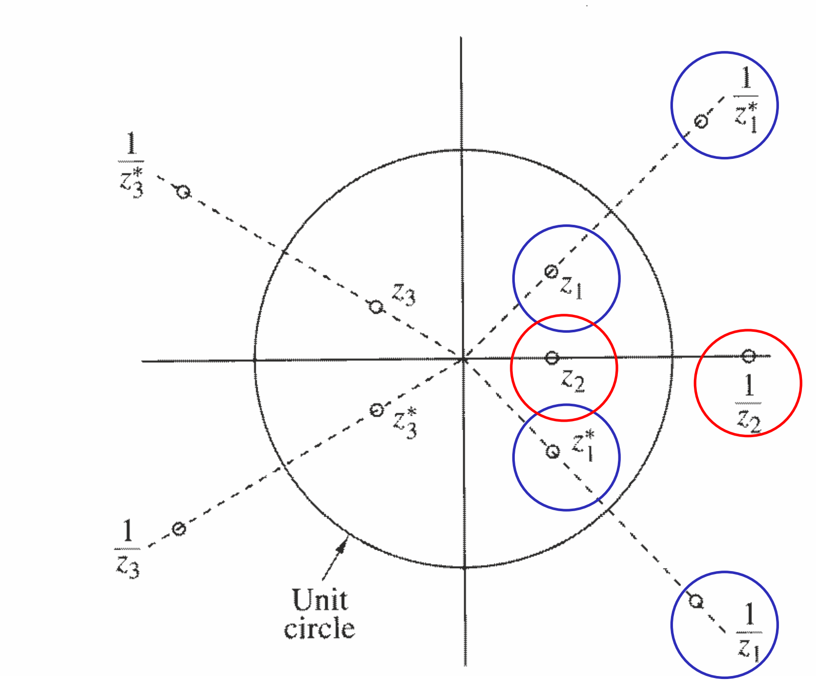
$\Rightarrow$ The zeros of a real-valued linear-phase filter occur in quadruples in the z-plane
(***exception: zeros on the real axis, zeros on the unit circle***).

Design of FIR Filters – Part 6

Linear-phase filters – Part 4

Consequences (continued):

Example: Pole-zero-diagram of a linear-phase filter



Linear-phase filters – Part 5

(a) Type-1 linear-phase system

Definition: ***Odd*** length L, ***even*** symmetry $h_i = h_{L-1-i}$. Frequency response:

$$H(e^{j\Omega}) = e^{-jS\Omega}\left[h_S + \sum_{i=0}^{S-1} h_i \cdot \left[e^{j\Omega\,(S-i)} + e^{-j\Omega\,(S-i)}\right]\right]$$

… using that $e^{jx} + e^{-jx} = 2\cos(x)$ ***…***

$$= e^{-jS\Omega}\left[h_S + 2\sum_{i=0}^{S-1} h_i \cdot \cos\left((S-i)\Omega\right)\right]$$

… abbreviating the term in brackets …

$$= e^{-jS\Omega}\, H_{01}(e^{j\Omega}).$$

Real term, thus we have a linear phase due to $e^{-jS\Omega}$ ***!***

As a result we get for the ***phase*** of that filter type:

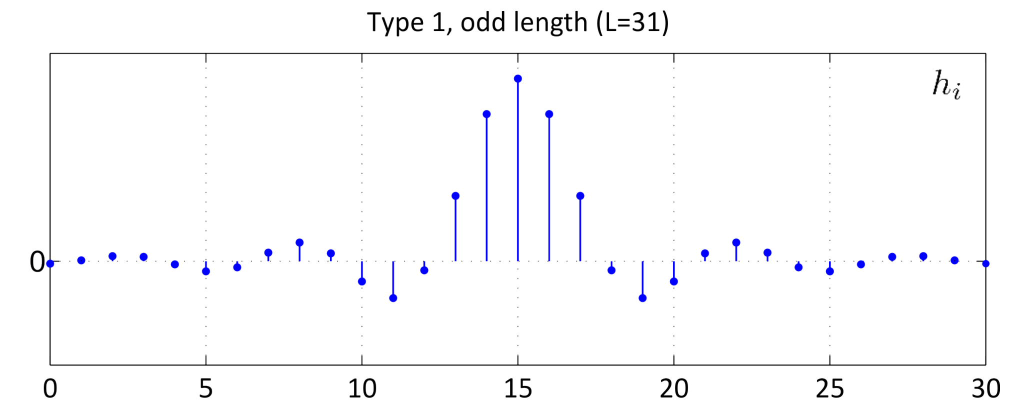
$$\varphi_{01}(\Omega) = -\arg H(e^{j\Omega}) = S\Omega.$$

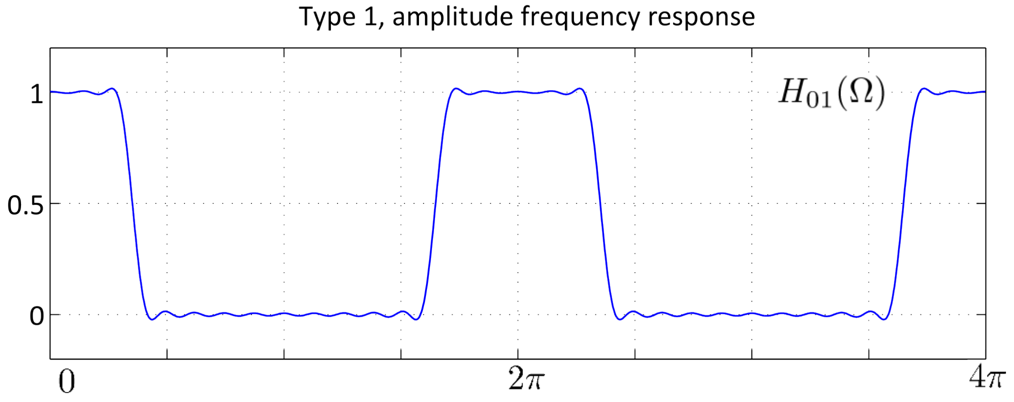
Remember: $H(e^{j\Omega}) = e^{-j\varphi(\Omega)}\,|H(e^{j\Omega})|.$

Linear phase filters – Part 6

(a) Type-1 linear phase system (continued)

Impulse and (amplitude) frequency response:





On the following slides equivalent derivations for the other cases (even/odd, type of symmetry) will be derived! The next seven slides are for reading at home!

Linear-phase filters – Part 7

(b) Type-3 linear-phase system

Odd length L, ***odd*** symmetry $h_i = -h_{L-1-i}$.
Frequency response:

$$H(e^{j\Omega}) = e^{-jS\Omega}\Big[\underbrace{h_S}_{=0} + \sum_{i=0}^{S-1} h_i \cdot \Big(e^{j(S-i)\Omega} - e^{-j(S-i)\Omega}\Big)\Big]$$

… using that $e^{jx} - e^{-jx} = 2\,j\,\sin(x)$ *and* $h_S = 0$ *since* $h_S = -h_{-S}$ *…*

$$= e^{-jS\Omega}\,j\left[2\sum_{i=0}^{S-1} h_i \cdot \sin\Big((S-i)\Omega\Big)\right]$$

… abbreviating the term in brackets with $H_{03}(\Omega)$ *and using* $j = e^{j\pi/2}$ *…*

$$= e^{-jS\Omega}\,j\,H_{03}(\Omega) = e^{-jS\Omega + j\pi/2}\,H_{03}(\Omega).$$

Result:

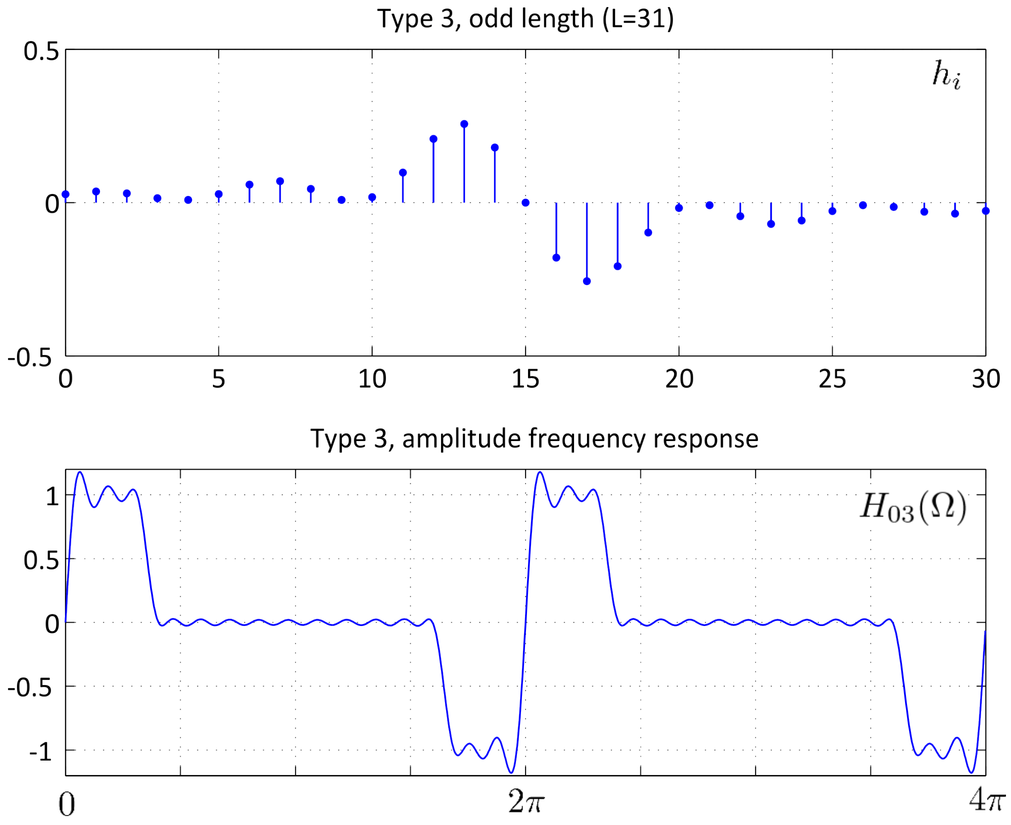
- Linear phase: $\varphi_{03}(\Omega) = -\arg H(e^{j\Omega}) = S\Omega - \pi/2.$
- $H(e^{j0}) = 0, \quad S \in \mathbb{N} \quad \Rightarrow \quad H(e^{j\pi}) = 0.$

Design of FIR Filters – Part 10

Linear-phase filters – Part 8

(b) Type-3 linear-phase system (continued)

Impulse and (amplitude) frequency response:



Linear-phase filters – Part 9

(c) Type-2 linear-phase system

Even length L, ***even*** symmetry $h_i = h_{L-1-i}$.
Frequency response:

$$H(e^{j\Omega}) = e^{-jS\Omega} \sum_{i=0}^{S-1} h_i \cdot \left(e^{j(S-i)\Omega} + e^{-j(S-i)\Omega}\right)$$

… using that $e^{jx} + e^{-jx} = 2\cos(x)$ ***…***

$$= e^{-jS\Omega}\, 2 \sum_{i=0}^{S-1} h_i \cdot \cos\Big((S-i)\Omega\Big)$$

… abbreviating the term in brackets with $H_{02}(\Omega)$ ***…***

$$= e^{-jS\Omega}\, H_{02}(\Omega).$$

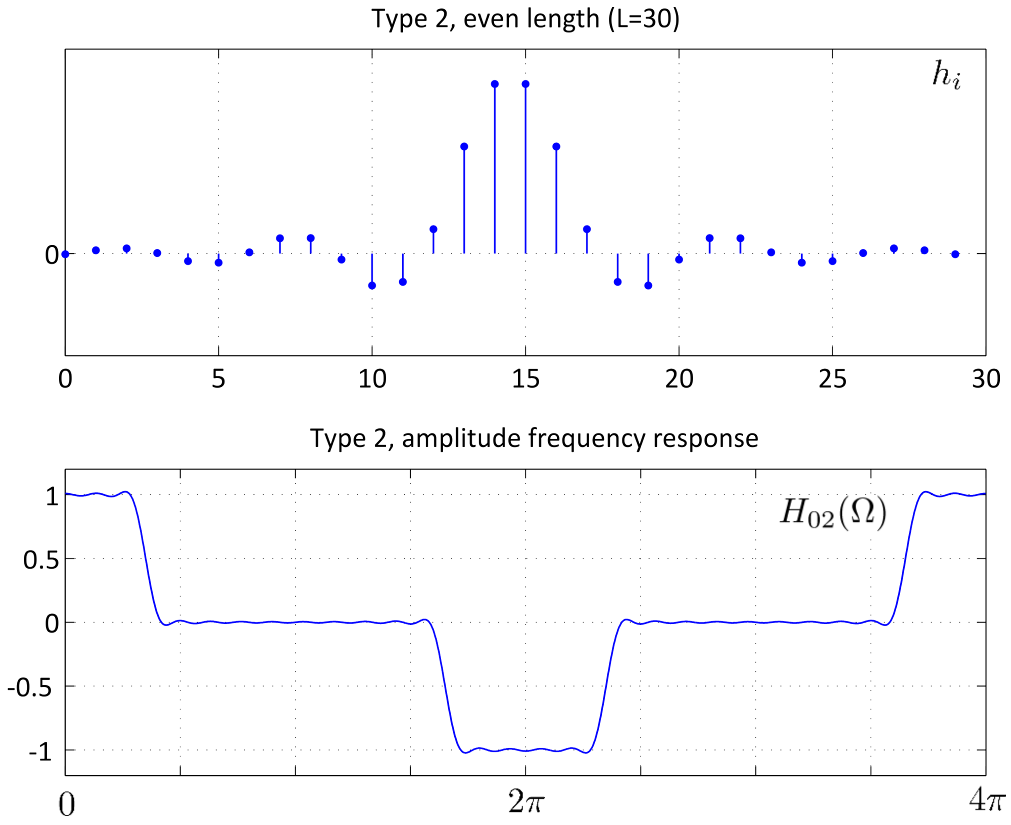
Result:

- Linear phase: $\varphi_{02}(\Omega) = -\arg H(e^{j\Omega}) = S\Omega.$
- $S = (2\lambda - 1)/2, \quad \lambda \in \mathbb{N} \quad \Rightarrow \quad H(e^{j\pi}) = 0.$

Linear-phase filters – Part 10

(c) Type-2 linear-phase system (continued)

Impulse and (amplitude) frequency response:



Note that $H_{02}(\Omega)$ is not periodic with 2π. That's true only for $H(e^{j\Omega})$! The phase term $e^{-jS\Omega}$ makes $e^{-jS\Omega}\,H_{02}(\Omega)$ again periodic with 2π!

Design of FIR Filters – Part 13

Linear-phase filters – Part 11

(d) Type-4 linear-phase system

Even length L, ***odd*** symmetry $h_i = -h_{L-1-i}$.
Frequency response:

$$H(e^{j\Omega}) = e^{-jS\Omega}\left[\sum_{i=0}^{S-1} h_i \cdot \left(e^{j(S-i)\Omega} - e^{-j(S-i)\Omega}\right)\right]$$

… using that $e^{jx} - e^{-jx} = 2\,j\,\sin(x)$ ***…***

$$= e^{-jS\Omega}\,j\left[2\sum_{i=0}^{S-1} h_i \cdot \sin\left((S-i)\Omega\right)\right]$$

… abbreviating the term in brackets with $H_{04}(\Omega)$ ***and using*** $j = e^{j\pi/2}$ ***…***

$$= e^{-jS\Omega}\,j\,H_{04}(\Omega) = e^{-jS\Omega+j\pi/2}\,H_{04}(\Omega).$$

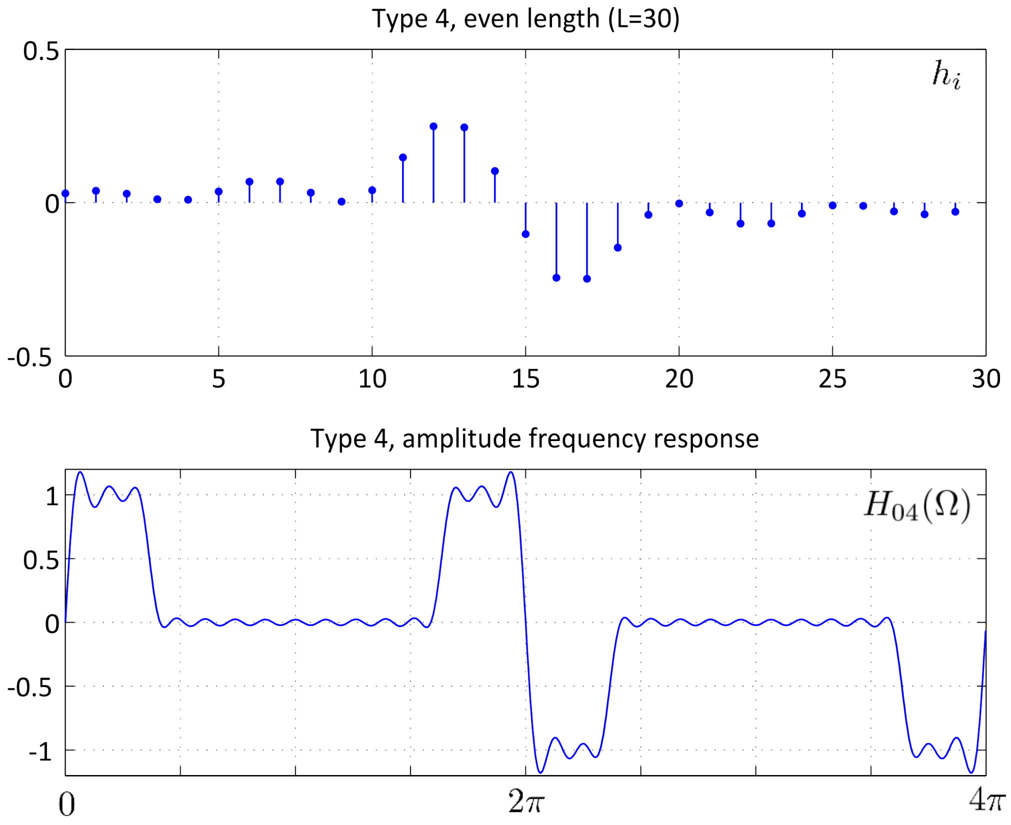
Result:

- Linear phase: $\varphi_{04}(\Omega) = -\arg H(e^{j\Omega}) = S\Omega - \pi/2.$
- $H(e^{j0}) = 0.$

Linear-phase filters – Part 12

(d) Type-4 linear-phase system (continued)

Impulse and (amplitude) frequency response:



Note that also $H_{04}(\Omega)$ is not periodic with 2π. That's true only for $H(e^{j\Omega})$! The phase term $e^{-jS\Omega+j\frac{\pi}{2}}$ makes $e^{-jS\Omega+j\frac{\pi}{2}}\,H_{04}(\Omega)$ again periodic with 2π !

Design of FIR Filters – Part 15

Linear-phase filters – Part 13

Applications:

- Type-1 and type-2 filters are used for "ordinary" filtering, however type-2 filters are not suitable for high-pass filtering.
- Type-3 and type-4 filters for example are used for 90 degree phase shifters and so-called ***Hilbert transformers***.

Design of FIR Filters – Part 16

Linear-phase filters – Part 14

Partner work – Please think about the following question and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What types of linear-phase filters do we have? How do they differ?

  ……

  ……

- Why is the term $e^{-jS\Omega}$ not always periodic with 2π?

  ……

  ……

- Do you know applications where linear-phase filters would be beneficial (compared to other filter types)?

  ……

  ……

FIR Filters

C Code for FIR Filters

Microsoft Visual Studio-Debugging-Konsole

```
Time for linear buffer with direct memory write is:   3.302000 s
Time for linear buffer with local result variable is: 1.126000 s
Time for ringbuffer is:                               1.011000 s
Time for double ringbuffer is:                        0.852000 s
Time for ringbuffer with partitioned convolution is:  0.748000 s
Time for ringbuffer unrolling by hand is:             0.355000 s
Time for SIMD AVX1 256bit instructions is             0.156000 s
Time for double ringbuffer and AVX1 is:               0.152000 s

Summed absolute error for ringbuffer with direct memory writing is:    0.000000
Summed absolute error for ringbuffer is:                               0.000000
Summed absolute error for double ringbuffer is:                        0.000000
Summed absolute error for ringbuffer with partitioned convolution is:  0.000000
Summed absolute error for unrolling by hand is:                        0.410343
Summed absolute error for SIMD AVX1 256bit instructions is:            0.383187
Summed absolute error for double ringbuffer and AVX1 is:               0.388364

D:\dss\projekte\dss_red\sources\fir_filter\vs_project\FIR-Test\Release\Project1.exe (Prozess "22364") wurde mit Code "0"
 beendet.
Um die Konsole beim Beenden des Debuggens automatisch zu schließen, aktivieren Sie "Extras" > "Optionen" > "Debuggen" >
"Konsole beim Beenden des Debuggings automatisch schließen".
Drücken Sie eine beliebige Taste, um dieses Fenster zu schließen.
```

Design of FIR Filters – Part 17

Linear-phase filters – Part 15

Design of linear-phase filters using a window function

Given: ***Desired frequency response***

$$H_{\mathrm{d}}(e^{j\Omega}) \;=\; \sum_{i=-\infty}^{\infty} h_{\mathrm{d},i}\, e^{-j\Omega i}.$$

Thus, the impulse response $h_{\mathrm{d},i}$ can be obtained using the inverse Fourier-transform:

$$h_{\mathrm{d},i} \;=\; \frac{1}{2\pi}\int_{-\pi}^{\pi} H_{\mathrm{d}}(e^{j\Omega})\, e^{j\Omega i}\, d\Omega.$$

Examples for “desired” filters:

- Ideal lowpass, highpass, or bandpass filters
- Delay filters (delaying a signal by a non-integer amount of samples, “fractional delay”)
- Hilbert filters (e.g. for frequency shifting)

Design of FIR Filters – Part 18

Linear phase filters – Part 16

Design of linear-phase filters using a window function (continued)

The impulse response has generally infinite length.
$\Rightarrow$ Truncation to the length L by multiplication with a window function w_i is necessary:

$$h_i = h_{\mathrm{d},i} \cdot w_i.$$

Rectangular window:

$$w_i = \begin{cases} 1, & i = 0, 1, ..., L-1, \\ 0, & \text{otherwise}, \end{cases} \quad \Rightarrow \quad h_i = \begin{cases} h_{\mathrm{d},i}, & i = 0, 1, ..., L-1, \\ 0, & \text{otherwise}. \end{cases}$$

Frequency response of the rectangular window (see section about "Frequency analysis of stationary signals" in the "DFT and FFT" chapter):

$$W(e^{j\Omega}) = e^{-j\Omega\frac{L-1}{2}} \cdot \frac{\sin\left(\frac{\Omega L}{2}\right)}{\sin\left(\frac{\Omega}{2}\right)}.$$

Linear phase filters – Part 17

Design of linear-phase filters using a window function (continued)

Suppose, we want to design a linear-phase filter of length L with the desired frequency response

$$H_{\mathrm{d}}(e^{j\Omega}) \;=\; \begin{cases} e^{-j\Omega\frac{L-1}{2}}, & \text{if } 0 \,\leq\, |\Omega + \lambda\, 2\pi| \,<\, \Omega_{\mathrm{c}}, \\ 0, & \text{else}, \end{cases}$$

where Ω_c is denoting the cut-off frequency. For the corresponding impulse response we get:

$$h_{\mathrm{d},i} \;=\; \frac{1}{2\pi}\int\limits_{-\Omega_{\mathrm{c}}}^{\Omega_{\mathrm{c}}} e^{j\Omega(i-\frac{L-1}{2})}\, d\Omega \;=\; \frac{\sin\left(\Omega_{\mathrm{c}}\left(i-\frac{L-1}{2}\right)\right)}{\pi\left(i-\frac{L-1}{2}\right)}.$$

Multiplication with a rectangular window of length L leads to

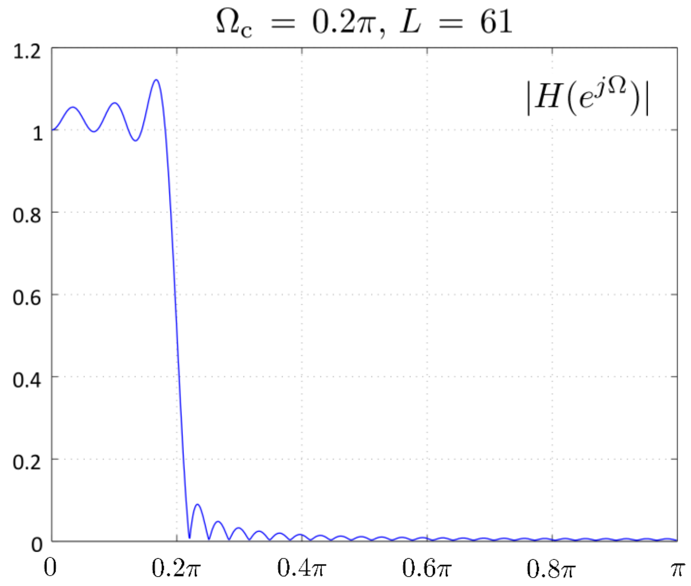
$$h_i \;=\; \begin{cases} \frac{\sin\left(\Omega_{\mathrm{c}}\left(i-\frac{L-1}{2}\right)\right)}{\pi\left(i-\frac{L-1}{2}\right)}, & \text{if } i \,=\, 0,\, 1,\, ...,\, L-1, \\ 0, & \text{else}. \end{cases}$$

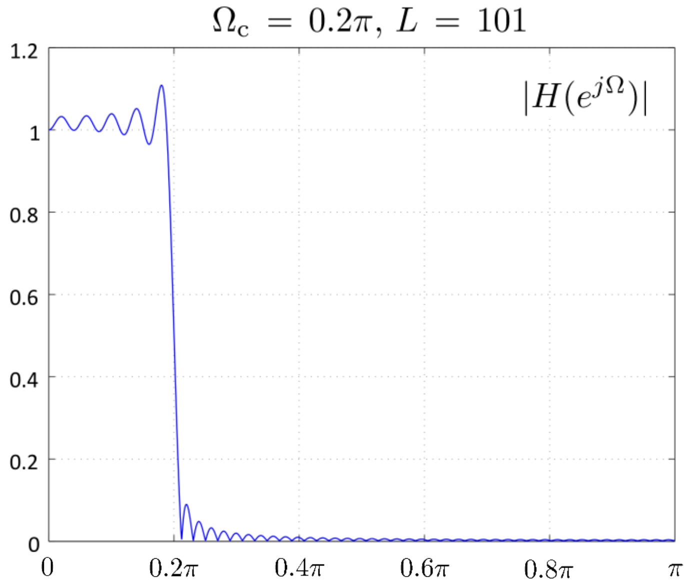
Design of FIR Filters – Part 20

Linear phase filters – Part 18

Design of linear-phase filters using a window function (continued)

Examples for $\Omega_c = 0.2\pi,\ L = 61$ and $L = 101$:





Design of FIR Filters – Part 21

Linear phase filters – Part 19

Design of linear-phase filters using a window function (continued)

Disadvantage of using a rectangular window:

Large sidelobes lead to an undesirable ringing effects (***overshoot at the boundary between pass- and stopband***) in the frequency response of the resulting FIR filter.

$\Rightarrow$ ***Gibbs phenomenon***:

- Result of ***approximating a discontinuity*** in the frequency response with a finite number of filter coefficients and a mean square error criterion
- The relation between $H_\mathrm{d}(e^{j\Omega})$ and $h_{\mathrm{d},i}$ can be interpreted as a ***Fourier series representation*** with the Fourier coefficients $h_{\mathrm{d},i}$ $\rightarrow$ Gibbs phenomenon results from a Fourier series approximation.
- The squared integral ***error***

$$E = \int_{-\pi}^{\pi} \left|H_\mathrm{d}(e^{j\Omega}) - H(e^{j\Omega})\right|^2 \, d\Omega$$

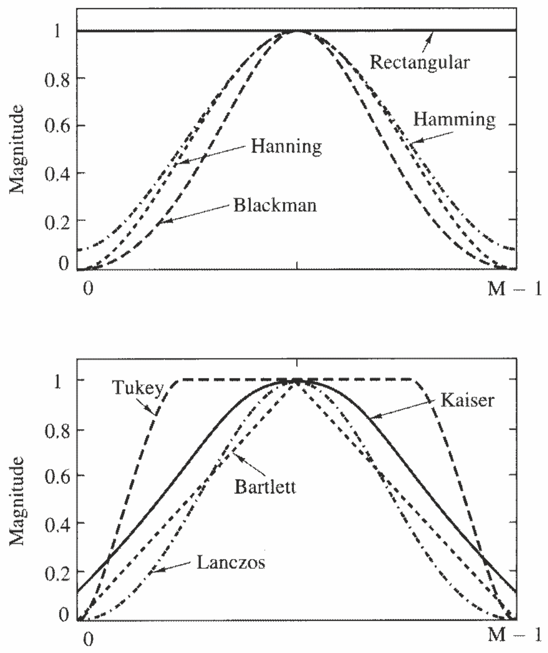
approaches zero with increasing length of $h_{\mathrm{d},i}$. However, the maximum value of the error $\left|H_\mathrm{d}(e^{j\Omega}) - H(e^{j\Omega})\right|$ approaches a constant value (independent of the filter length).

Design of FIR Filters – Part 22

Linear phase filters – Part 20

Design of linear-phase filters using a window function (continued)

$\Rightarrow$ Use of ***other appropriate window functions*** *with* lower sidelobes in their frequency responses.



From [Proakis, Manolakis, 1996]

Name of window	Time-domain sequence, $h(n), 0 \le n \le M-1$
Bartlett (triangular)	$1 - \dfrac{2\left\lvert n - \frac{M-1}{2}\right\rvert}{M-1}$
Blackman	$0.42 - 0.5\cos\dfrac{2\pi n}{M-1} + 0.08\cos\dfrac{4\pi n}{M-1}$
Hamming	$0.54 - 0.46\cos\dfrac{2\pi n}{M-1}$
Hanning	$\dfrac{1}{2}\left(1 - \cos\dfrac{2\pi n}{M-1}\right)$
Kaiser	$\dfrac{I_0\left[\alpha\sqrt{\left(\frac{M-1}{2}\right)^2 - \left(n - \frac{M-1}{2}\right)^2}\right]}{I_0\left[\alpha\left(\frac{M-1}{2}\right)\right]}$
Lanczos	$\left\{\dfrac{\sin\left[2\pi\left(n - \frac{M-1}{2}\right) / (M-1)\right]}{2\pi\left(n - \frac{M-1}{2}\right) / \left(\frac{M-1}{2}\right)}\right\}^L \quad L > 0$
Tukey	$1, \quad \left\lvert n - \frac{M-1}{2}\right\rvert \le \alpha\frac{M-1}{2} \quad 0 < \alpha < 1$ $\dfrac{1}{2}\left[1 + \cos\left(\dfrac{n - (1+a)(M-1)/2}{(1-\alpha)(M-1)/2}\pi\right)\right]$ $\alpha(M-1)/2 \le \left\lvert n - \frac{M-1}{2}\right\rvert \le \frac{M-1}{2}$

Linear phase filters – Part 21

Design of linear-phase filters using a window function (continued)

Frequency-domain characteristics of some window functions [Proakis, Manolakis, 1996]:

Type of window	Approximate transition width of main lobe	Peak sidelobe in dB
Rectangular	$4\pi/L$	-13
Bartlett	$8\pi/L$	-27
Hann	$8\pi/L$	-32
Hamming	$8\pi/L$	-43
Blackman	$12\pi/L$	-58

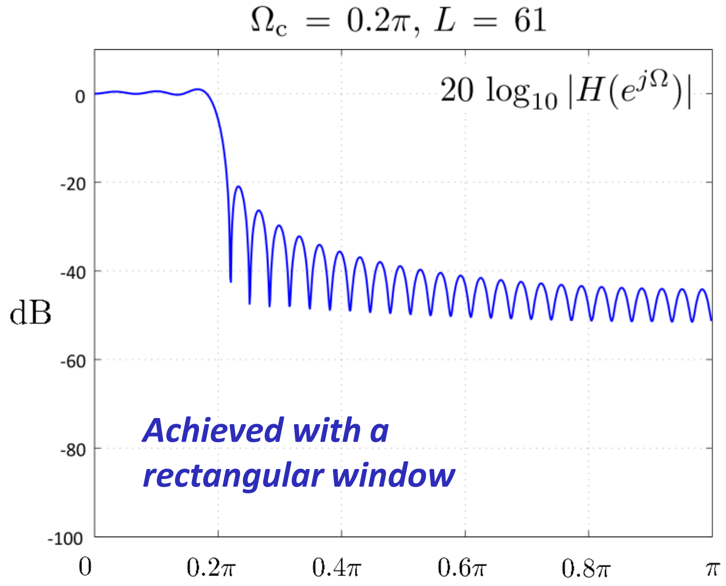
The parameter α in the Kaiser window allows to adjust the width of the main lobe, and thus also to adjust the compromise between overshoot reduction and increased transition bandwidth in the resulting FIR filter. I_0 denotes the Bessel function of the first kind of order zero.

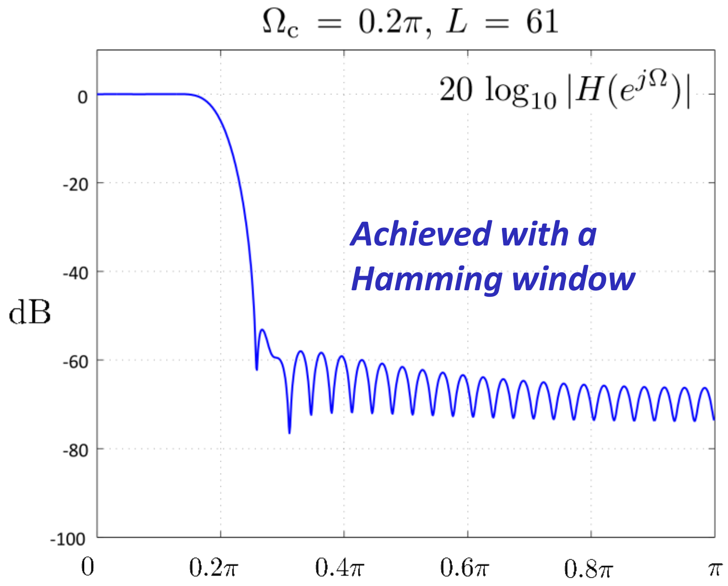
Design of FIR Filters – Part 24

Linear phase filters – Part 22

Design of linear-phase filters using a window function (continued)

Magnitude frequency response $20\log_{10}|H(e^{j\Omega})|$ of the resulting linear-phase FIR filter, when different window functions are used to truncate the infinite-length impulse response $h_{\mathrm{d},i}$ with the desired frequency response $H_\mathrm{d}(e^{j\Omega})$:

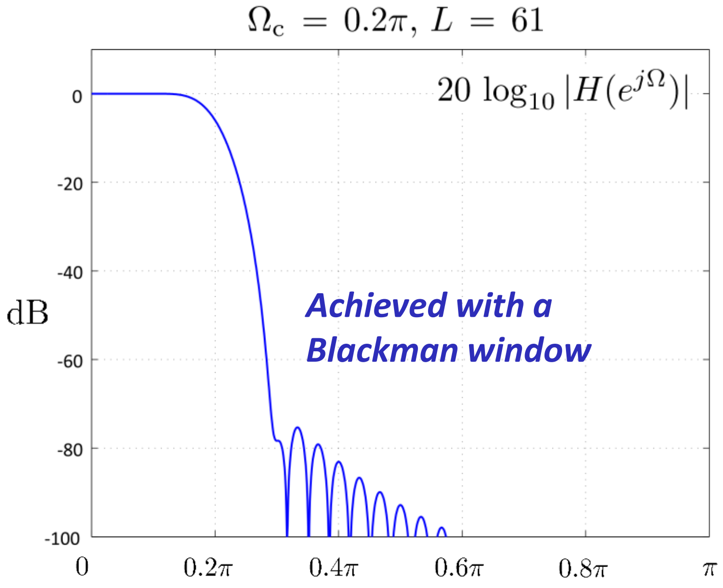


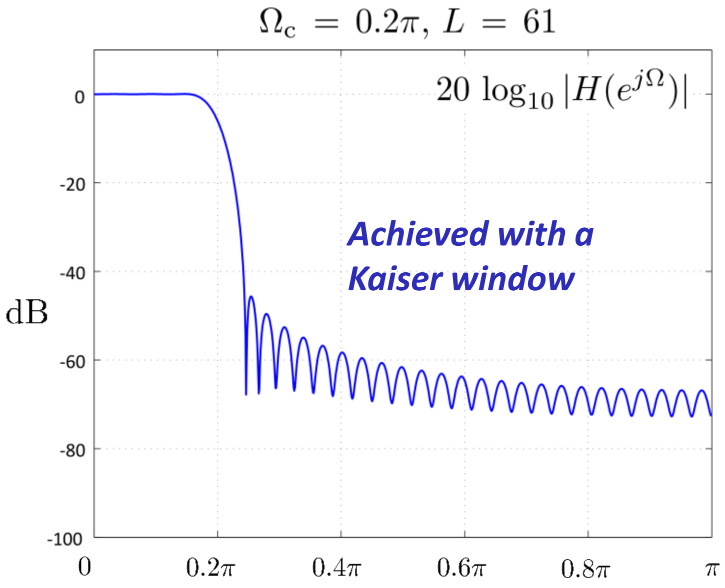


Linear phase filters – Part 23

Design of linear-phase filters using a window function (continued)

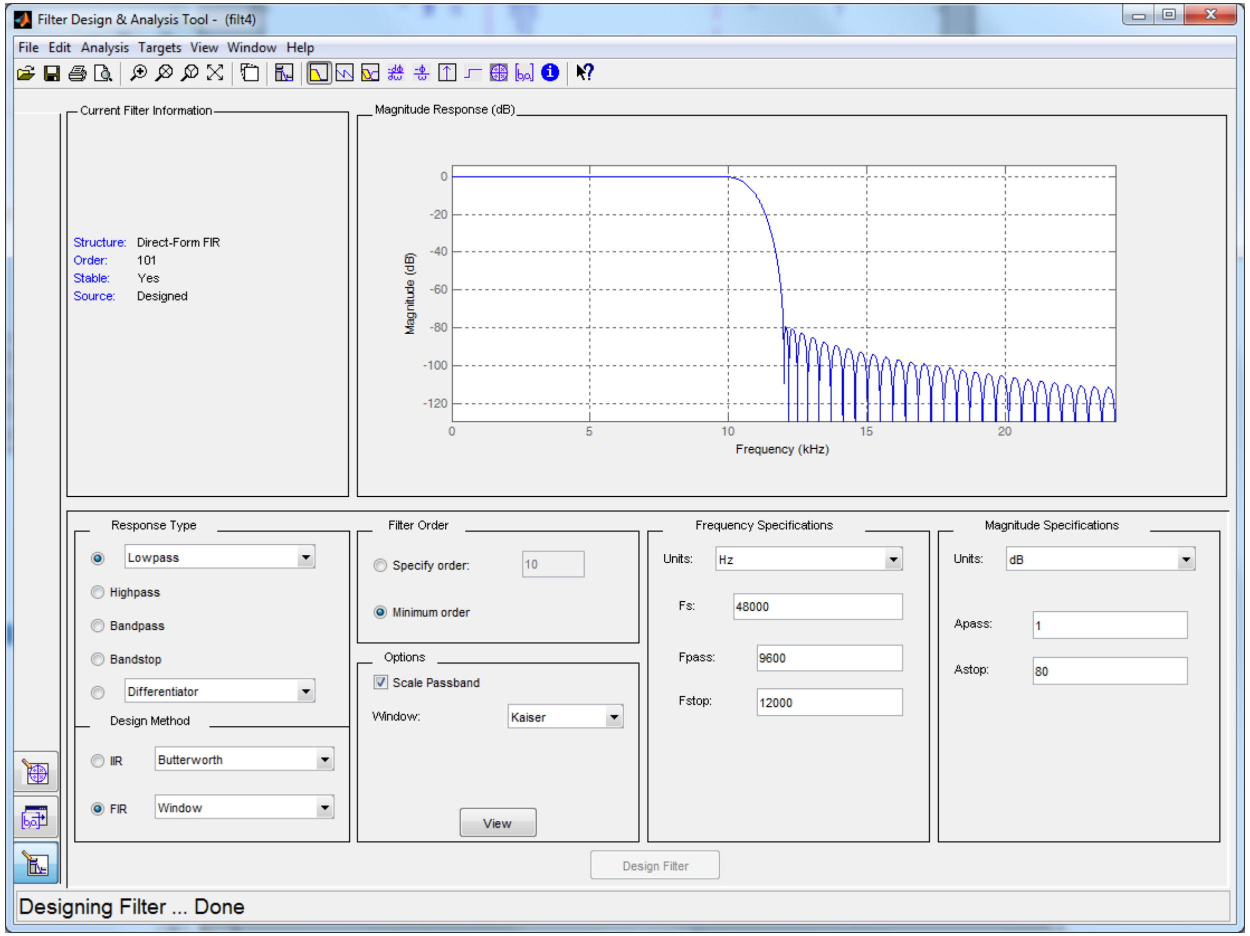
Magnitude frequency response $20\log_{10}|H(e^{j\Omega})|$ of the resulting linear-phase FIR filter, when different window functions are used to truncate the infinite-length impulse response $h_{\mathrm{d},i}$ with the desired frequency response $H_\mathrm{d}(e^{j\Omega})$:





Design of FIR Filters – Part 26

Linear phase filters – Part 24



Matlab example

Design of FIR Filters – Part 27

Linear-phase filters – Part 24

Partner work – Please think about the following questions and try to find answers
(first group discussions, afterwards broad discussion in the whole group).

- What are the basic steps to get a stable, causal, finite, and linear-phase filter from a “desired” filter?

  ………

  ………

- What does the multiplication with a window function corresponds to in the frequency domain?

  How should the spectrum of an “optimal” window function look like?

  ………

  ………

- What are the basic parameters that describe window functions in the frequency domain?

  ………

  ………

Design of FIR Filters – Part 28

Linear phase filters – Part 25

Frequency sampling design

The desired frequency response $H_\mathrm{d}(e^{j\Omega})$ is specified at a set of equally spaced frequencies:

$$\Omega_\mu \;=\; \frac{2\pi}{L}(\mu+\alpha), \quad \mu \;=\; 0,\, 1,\; ...,\, L-1, \quad \alpha \in \left\{0,\, \frac{1}{2}\right\}.$$

We could now design an FIR filter with a frequency response equal to the desired one at the above mentioned frequency supporting points:

$$H_\mathrm{d}(e^{j\Omega_\mu}) \;\stackrel{!}{=}\; H(e^{j\Omega_\mu}) \;=\; \sum_{i=0}^{L-1} h_i\, e^{-j\Omega_\mu i}.$$

By combining both equations we obtain for $\mu = 0, 1, \ldots, L-1$:

$$H_\mathrm{d}(e^{j\frac{2\pi}{L}(\mu+\alpha)}) \;=\; \sum_{i=0}^{L-1} h_i\, e^{-j\frac{2\pi}{L}(\mu+\alpha)i}.$$

Design of FIR Filters – Part 29

Linear phase filters – Part 26

Frequency sampling design (continued)

Multiplication with $e^{j\frac{2\pi}{L}\mu l},\ l = 1,\ ...,\ L-1$ and summation over $\mu = 0,\ ...,\ L-1$ yields to

$$H_\mathrm{d}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; \sum_{i=0}^{L-1} h_i\, e^{-j\frac{2\pi}{L}(\mu+\alpha)i}$$

… multiplication with the exponential term mentioned above and summation …

$$\sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l}\, H_\mathrm{d}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; \sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l} \sum_{i=0}^{L-1} h_i\, e^{-j\frac{2\pi}{L}(\mu+\alpha)i}$$

… exchanging the summation order and rearranging the exponential …

$$\sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l}\, H_\mathrm{d}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; \sum_{i=0}^{L-1} h_i\, e^{-j\frac{2\pi}{L}\alpha i} \sum_{\mu=0}^{L-1} e^{-j\frac{2\pi}{L}(i-l)\mu}$$

… exploiting the properties of sums of exponentials …

$$\sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l}\, H_\mathrm{d}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; h_l\, e^{-j\frac{2\pi}{L}\alpha l}\, L.$$

Linear phase filters – Part 27

Frequency sampling design (continued)

Resolving the result from the last slide to h_l leads to

$$\sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l}\, H_{\mathrm{d}}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; h_l\, e^{-j\frac{2\pi}{L}\alpha l}\, L$$

… dividing by L and multiplication with $e^{j\frac{2\pi}{L}\alpha l}$ …

$$e^{j\frac{2\pi}{L}\alpha l}\,\frac{1}{L}\sum_{\mu=0}^{L-1} e^{j\frac{2\pi}{L}\mu l}\, H_{\mathrm{d}}\big(e^{j\frac{2\pi}{L}(\mu+\alpha)}\big) \;=\; h_l.$$

Some ***remarks***:

- The result can be computed efficiently using an IFFT!
- Note that only frequency supporting point are specified, the filter characteristic in between these supporting points might be “not as expected”.
- This type of design is sometimes used in real-time applications (due to its low complexity)!

Linear phase filters – Part 28

Optimum equiripple design (Chebyshev approximation)

- Window design techniques try to reduce the difference between the desired and the actual frequency response (error function) by choosing suitable windows.
- How far can the maximum error be reduced?
  - ⇒ The theory of ***Chebyshev approximation*** answers this question and provides us with algorithms to find the coefficients of linear-phase FIR filters, where the maximum of the frequency response error is minimized.
- ***Chebyshev approximation***:
  Approximation that minimizes the maximum errors over a set of frequencies.
- The resulting filters exhibit an equiripple behavior in their frequency responses
  - ⇒ ***equiripple filters***.

Design of FIR Filters – Part 32

Linear phase filters – Part 29

Optimum equiripple design (Chebyshev approximation) (continued)

As we have shown before, every linear-phase filter has a frequency response of the form

$$H(e^{j\Omega}) \;=\; (j)^m \cdot A(\Omega) \cdot e^{-j\Omega\frac{L-1}{2}}, \quad m \in \{0,1\},$$

where $A(\Omega)$ is a real-valued positive or negative function (amplitude frequency response).

It can be shown that for all types of linear-phase symmetry $A(\Omega)$ can always be written as a weighted sum of cosines. For example, for type 1 linear-phase filters we have

$$A(\Omega) \;=\; \sum_{n=0}^{(L-1)/2} a_n \cos(n\Omega)$$

with

$$a_0 \;=\; h_{\frac{L-1}{2}}, \quad a_n \;=\; 2h_{-n+\frac{L-1}{2}}.$$

Design of FIR Filters – Part 33

Linear phase filters – Part 30

Optimum equiripple design (Chebyshev approximation) (continued)

Problem definition:

Acceptable frequency response for the FIR filter:

- Linear phase,
- transition bandwidth $\Delta\Omega$ between pass- and stopband,
- passband deviation $\pm\delta_1$ from unity,
- stopband deviation $\pm\delta_2$ from zero.

(Multiple bands are possible as well.)

In the following we will restrict ourselves to lowpass type 1 linear-phase filters.

Design of FIR Filters – Part 34

Linear phase filters – Part 31

Optimum equiripple design (Chebyshev approximation) (continued)

Approximation Problem: Given

- a compact subset F_μ of $[0,\pi]$ in the frequency domain (consisting of pass- and stop-band in the lowpass filter case),
- a desired real-valued frequency response $D(\Omega)$, defined on F_μ,
- a positive weight function $W(\Omega)$, defined on F_μ, and
- the form of $A(\Omega)$, here (type-1 linear phase)

$$A(\Omega) = \sum_{n=0}^{\frac{L-1}{2}=R-1} a_n \cos(n\Omega).$$

This is a so-called "minimax" criterion.

Goal: Minimization of the error

$$E_{\max} = \max_{\Omega \in F_\mu} \left\{ W(\Omega) \cdot \left| D(\Omega) - A(\Omega) \right| \right\} \longrightarrow \min$$

over a_n by the choice of $A(\Omega)$.

Design of FIR Filters – Part 35

Linear phase filters – Part 32

Optimum equiripple design (Chebyshev approximation) (continued)

Alternation theorem (without proof):

If $A(\Omega)$ is a linear combination of R cosine functions,

$$A(\Omega) = \sum_{n=0}^{R-1} a_n \cos(n\Omega),$$

then a necessary and sufficient condition is that $A(\Omega)$ is the unique and best weighted Chebyshev approximation to a given continuous function $D(\Omega)$ on F_μ is:

The weighted error function $E(\Omega) = W(\Omega) \cdot \big[D(\Omega) - A(\Omega)\big]$ *exhibits at least* $R+1$ *extremal frequencies in* F_μ*.*

These frequencies are supporting points for which hold:

$$\Omega_0 < \cdots < \Omega_{R-1} < \Omega_R,$$

$$E(\Omega_\nu) = -E(\Omega_{\nu+1}), \quad \nu = 0, 1, ..., R-1,$$

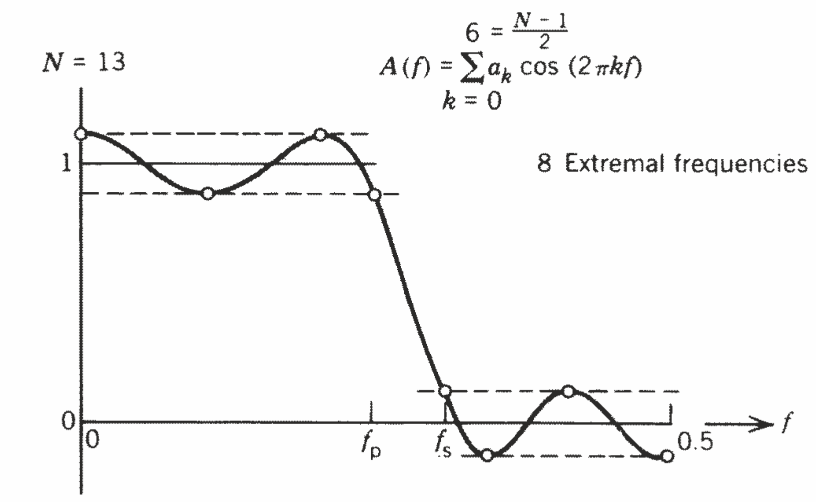
$$\big|E(\Omega_\nu)\big| = \max_{\Omega \in F_\mu} \big\{E(\Omega)\big\}, \quad \nu = 0, 1, ..., R.$$

Design of FIR Filters – Part 36

Linear phase filters – Part 33

Optimum equiripple design (Chebyshev approximation) (continued)

- Consequences from the alternation theorem:
  Best Chebyshev approximation must have an equiripple error function $E(\Omega)$ and is ***unique***.
- Example: Amplitude frequency response of an optimum type 1 linear-phase filter with $L = 13 \rightarrow R = 7 \; (\Omega \rightarrow f, n \rightarrow k)$



[Parks, Burrus: Digital Filter Design, 1987]

Design of FIR Filters – Part 37

Linear phase filters – Part 34

Optimum equiripple design (Chebyshev approximation) (continued)

- If the $R+1$ extremal frequencies were known, we could use the frequency-sampling design from above to specify the desired values $1 \pm \delta_1$ at the extremal frequencies in the passband, and $\pm\delta_2$ in the stopband, respectively.

How to find the set of extremal frequencies?

Remez exchange algorithm (Parks, McLellan, 1972)

- It can be shown that the error function

$$E(\Omega) \;=\; D(\Omega) - \sum_{n=0}^{R-1} a_n \cos(n\Omega)$$

can be forced to take on some values $\pm\delta$ for any given set of $R+1$ frequency points $\Omega_i,\ i = 0, ..., R.$

Simplification

Restriction to $W(\Omega) = 1$ and $\delta_1 = \delta_2 = \delta.$

Design of FIR Filters – Part 38

Linear phase filters – Part 35

Optimum equiripple design (Chebyshev approximation) (continued)

Remez exchange algorithm (continued)

This can be written as a set of linear equations according to

$$D(\Omega_i) = \sum_{n=0}^{R-1} a_n \cos(n\Omega_i) + (-1)^i \cdot \delta, \quad i = 0, \ldots, R.$$

R+1 equations!

We obtain a unique solution for the coefficients $a_n, \; n = 0, \ldots, R-1$ and the error magnitude δ.

R unknowns! *1 unknown!*

Finding the new set of extremal frequencies can be obtained using an FFT with zero padding:

- The frequency point are usually chosen in an equally spaced grid.
  The number of the frequency points is approximately $10 \cdot L$.
- The algorithm is initialized with a trial set of arbitrarily chosen frequencies

$$T = \{\Omega_0, \Omega_1, \ldots, \Omega_R\}.$$

Linear phase filters – Part 36

Optimum equiripple design (Chebyshev approximation) (continued)

Remez exchange algorithm (continued)

The steps of the Remez algorithm:

1. Solve the linear equation for the desired frequency response $D(\Omega)$, yielding an error magnitude δ_n in the n-th iteration.
2. Interpolate to find the frequency response on the entire grid of frequencies.
3. Search over the entire grid of frequencies for a larger magnitude error than δ_n obtained in step 1.
4. Stop, if no larger magnitude error can be found.
   Otherwise, take the $R+1$ frequencies, where the error attains its maximum magnitude as a new trial set of extremal frequencies and go to step 1.

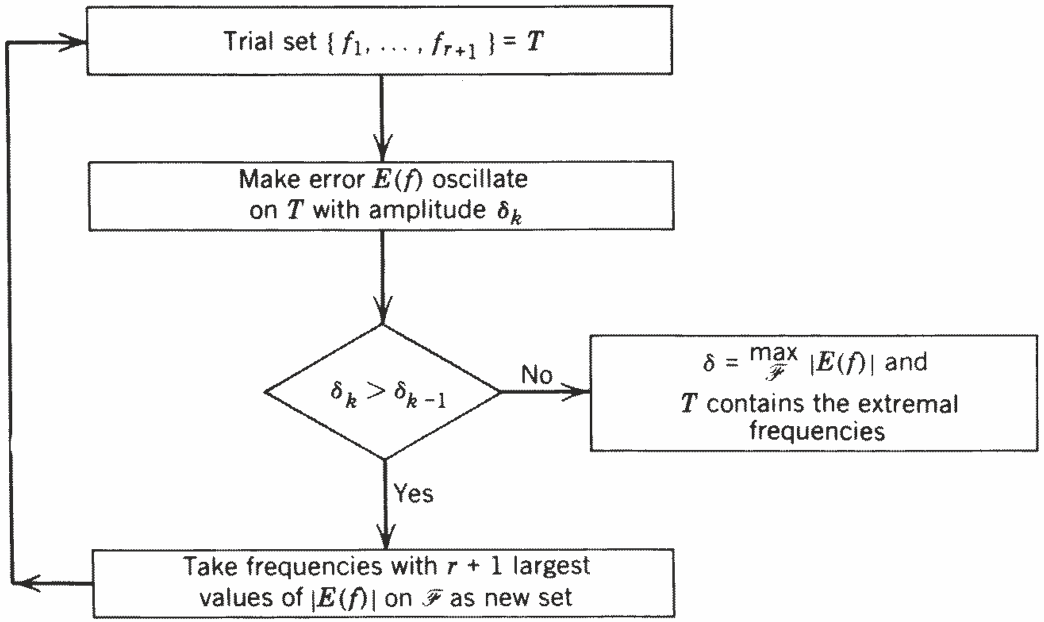
Design of FIR Filters – Part 40

Linear phase filters – Part 37

Optimum equiripple design (Chebyshev approximation) (continued)

Remez exchange algorithm (continued)

Remes exchange



δ_k increases on each iteration. The iteration stops when δ_k stops increasing. At this point, $\delta_k = \max_{\mathcal{F}} |E(f)|$ and T contains the $r + 1$|extremal frequencies.

[From: Parks, Burrus: Digital Filter Design, 1987]

Linear phase filters – Part 38

Remez exchange algorithm (continued)

Example:

Desired: $D(x) = x^2$

Problem: Choose the two coefficients d_0 and d_1 such that they minimize the Chebyshev error

$$\max_{x\in[0,1]} \left| x^2 - (d_0 + d_1 x) \right| \longrightarrow \min$$

(approximation of a parabola by a straight line).

Approach/ solution:

$\rightarrow$ three extremal points

$\rightarrow$ the resulting equations to be solved:

$$x_i^2 \;=\; d_0 + d_1\, x_i + (-1)^i \cdot \delta, \quad i \in \{0,\,1,\,2\}.$$

Linear phase filters – Part 39

Remez exchange algorithm (continued)

Example:

1. Arbitrarily chosen trial set: $T_0 = [0.25, 0.5, 1.0]$
   Matrix version of the linear equations:

$$x_i^2 = d_0 + d_1\,x_i + (-1)^i \cdot \delta$$

$$\begin{bmatrix} 1 & 0.25 & 1 \\ 1 & 0.5 & -1 \\ 1 & 1.0 & 1 \end{bmatrix} \begin{bmatrix} d_0 \\ d_1 \\ \delta_0 \end{bmatrix} = \begin{bmatrix} 0.0625 \\ 0.25 \\ 1.0 \end{bmatrix} \quad \rightarrow \delta_0 = 0.0625$$

2. Next trial set chosen as those three points, where the error

$$E(x) = x^2 - (d_0 + d_1 x)$$

   achieves its maximum magnitude $\rightarrow T_1 = [0, 0.625, 1]$
   Linear equations to solve:

$$\begin{bmatrix} 1 & 0.0 & 1 \\ 1 & 0.625 & -1 \\ 1 & 1.0 & 1 \end{bmatrix} \begin{bmatrix} d_0 \\ d_1 \\ \delta_0 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 0.390625 \\ 1.0 \end{bmatrix} \quad \rightarrow \delta_1 = 0.1171975$$

Linear phase filters – Part 40

Remez exchange algorithm (continued)

Example:

3. Next trial set: $T_2 = [0.0, 0.5, 1.0]$
   Linear equations to solve:

$$\begin{bmatrix} 1 & 0.0 & 1 \\ 1 & 0.5 & -1 \\ 1 & 1.0 & 1 \end{bmatrix} \begin{bmatrix} d_0 \\ d_1 \\ \delta_0 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 0.25 \\ 1.0 \end{bmatrix} \quad \rightarrow \delta_2 = 0.125 \mathrel{\hat{=}} \text{maximum error}$$

   $\rightarrow T_2$ is the extremal point set.

After the third step the parameter δ_i does not change any more. Now the coefficients d_0 and d_1 are used for the final solution.

Design of FIR Filters – Part 44

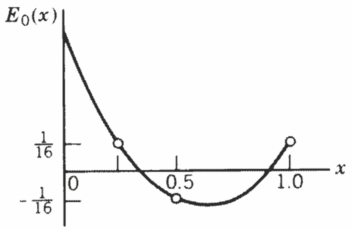
Linear phase filters – Part 41

Remez exchange algorithm (continued)

Example:

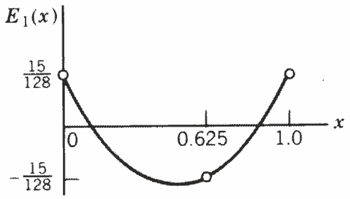
Choose d_0, d_1 to minimize $\max_{x \in [0,1]} |D(x) - (d_0 + d_1 x)|$

$D(x) = x^2$.



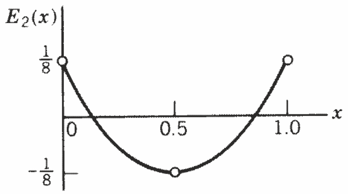
$T_0 = \left\{\frac{1}{4}, \frac{1}{2}, 1\right\}$ $\delta_0 = \frac{1}{16}$

$E_0 = x^2 - \frac{5}{4}x + \frac{5}{16}$, $||E_0|| = \frac{5}{16}$



$T_1 = \left\{0, \frac{5}{8}, 1\right\}$ $\delta_1 = \frac{15}{128}$

$E_1 = x^2 - x + \frac{15}{128}$ $||E_1|| = \frac{17}{128}$



$T_2 = \left\{0, \frac{1}{2}, 1\right\}$ $\delta_2 = \frac{1}{8}$

$E_2 = x^2 - x + \frac{1}{8}$ $||E_2|| = \frac{1}{8}$

[From: Parks, Burrus: Digital Filter Design, 1987]

Design of FIR Filters – Part 45

Linear phase filters – Part 42

Remez exchange algorithm (continued)

Estimation of the filter length:

Given the stop-/ passband ripple $\delta_1,\ \delta_2$ and the transition bandwidth $\Delta\Omega = \Omega_s - \Omega_p$ the necessary filter order N can be estimated as (Kaiser, 1974)

$$N = \frac{-10\log_{10}(\delta_1\,\delta_2) - 13}{2.324\Delta\Omega}.$$

Design example:

Design a linear-phase lowpass filter with the specifications

$$\delta_1 = 0.01, \quad \delta_2 = 0.001, \quad \Omega_p = 0.4\pi, \quad \Omega_s = 0.6\pi.$$

$\rightarrow$ weighting $\delta_2/\delta_1 = 10$ in the stopband.

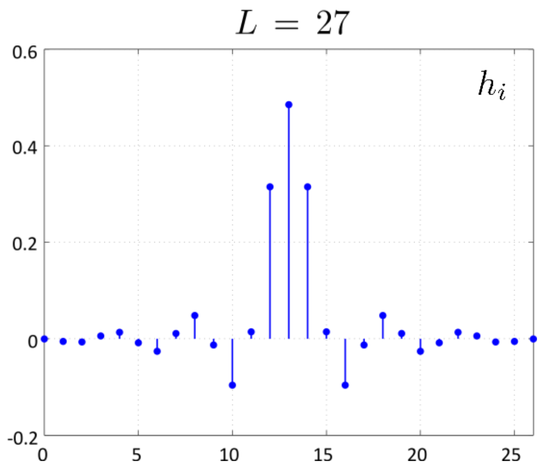
The filter order estimate gives $N \approx 25.34$. Rounding up yields a filter length of $L = N + 1 = 27$.

Design of FIR Filters – Part 46

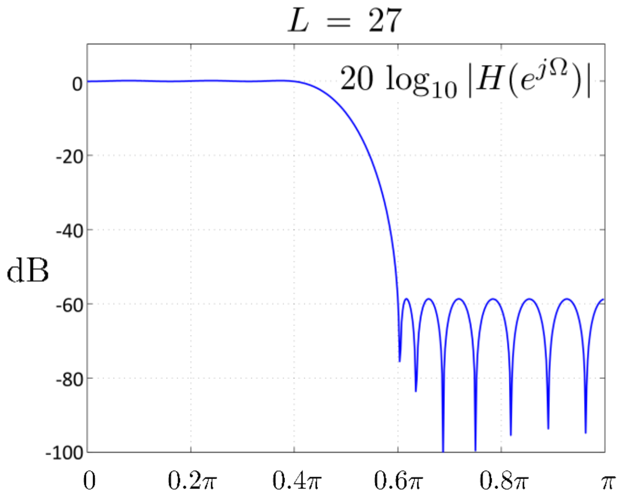
Linear phase filters – Part 43

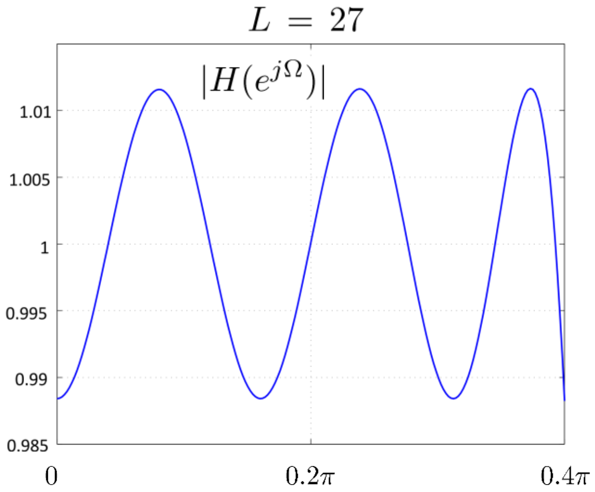
Remez exchange algorithm (continued)

Design example (continued):



In the passband the specifications are not satisfied.
$\rightarrow$ Increasing the filter length by one, $L = 28$.



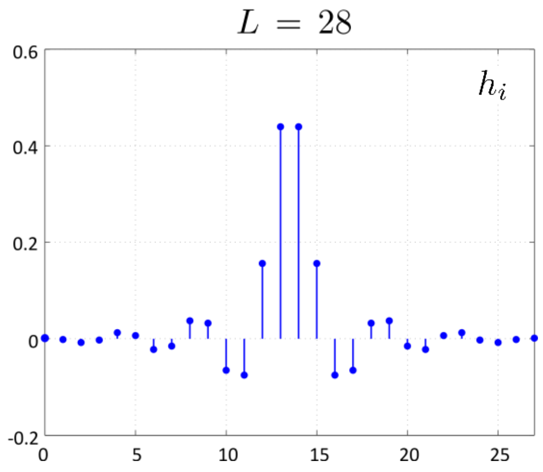


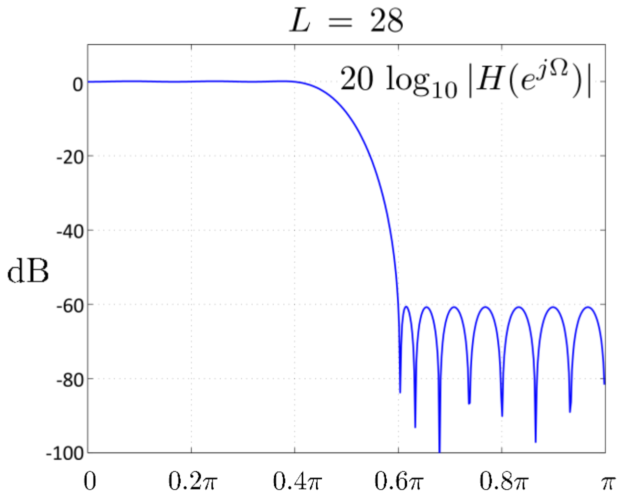
Design of FIR Filters – Part 47

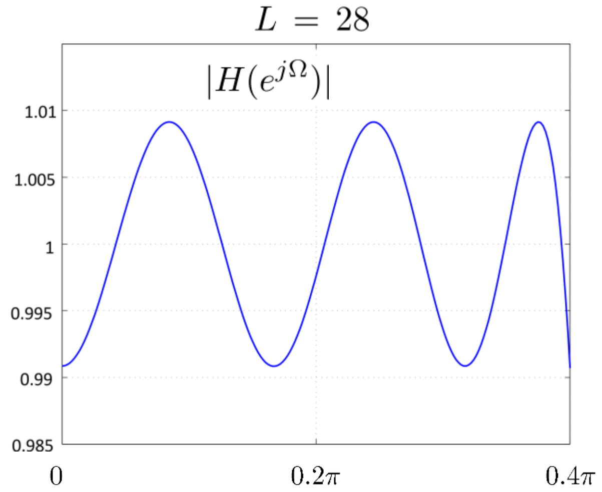
Linear phase filters – Part 44

Remez exchange algorithm (continued)

Design example (continued):







Design of FIR Filters – Part 48

Linear-phase filters – Part 45

Partner work – Please think about the following questions and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What are the problems when designing an FIR filter using only frequency supporting points?

  ………

  ………

- What is optimized with a "minimax" criterion? What other criteria do you know?

  ………

  ………

- What are the basic steps of the Remez exchange algorithm?

  ………

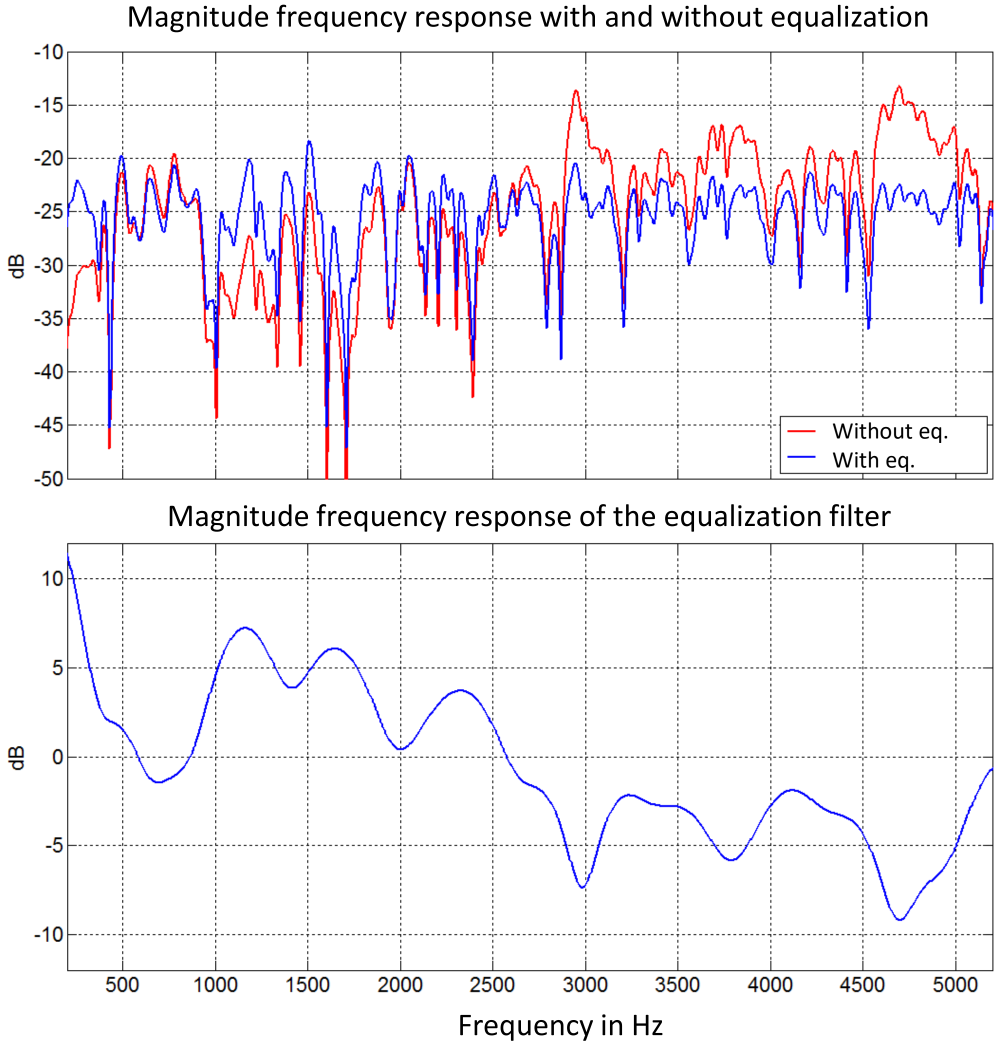
  ………

Christian-Albrechts-Universität zu Kiel

Predictor-based Filter Design

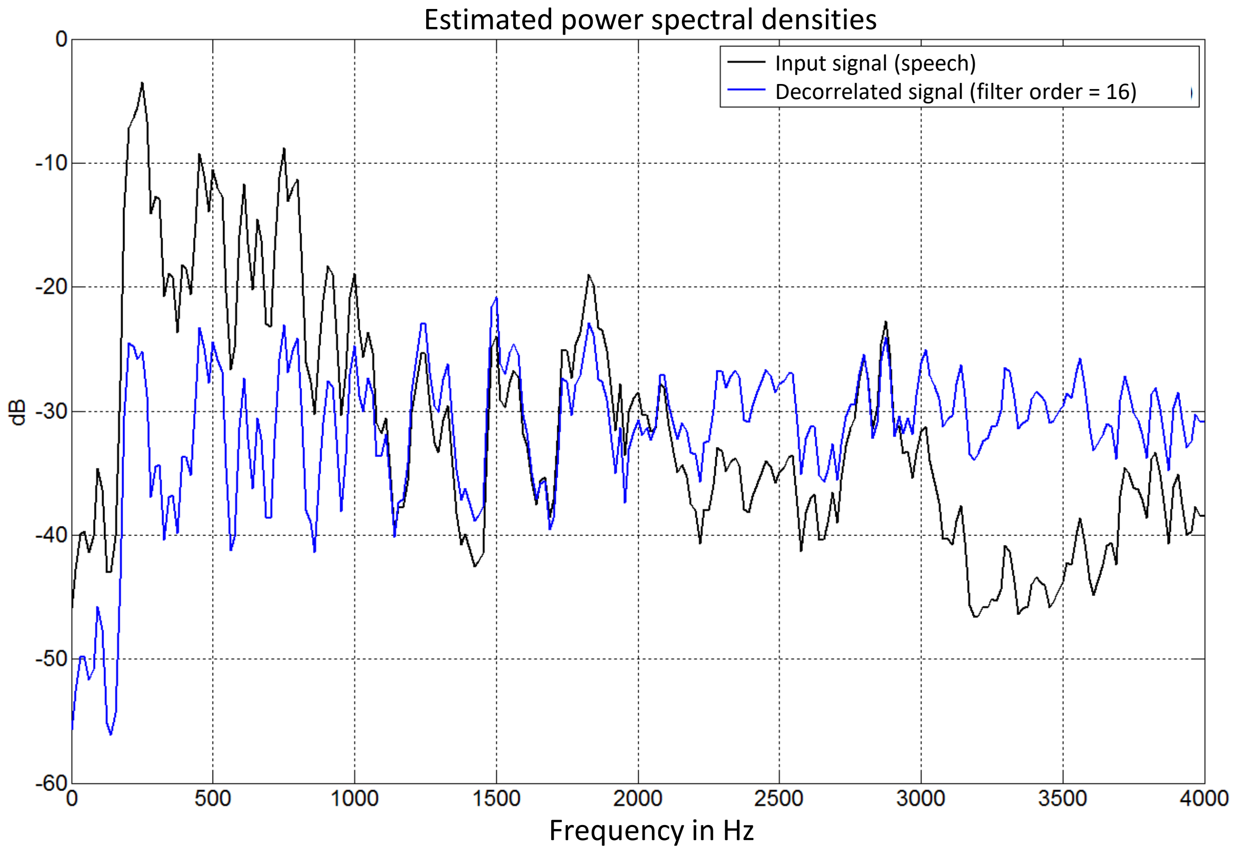
Equalization with FIR and IIR filters

- Predictor-based filter design
- Linear-phase extension



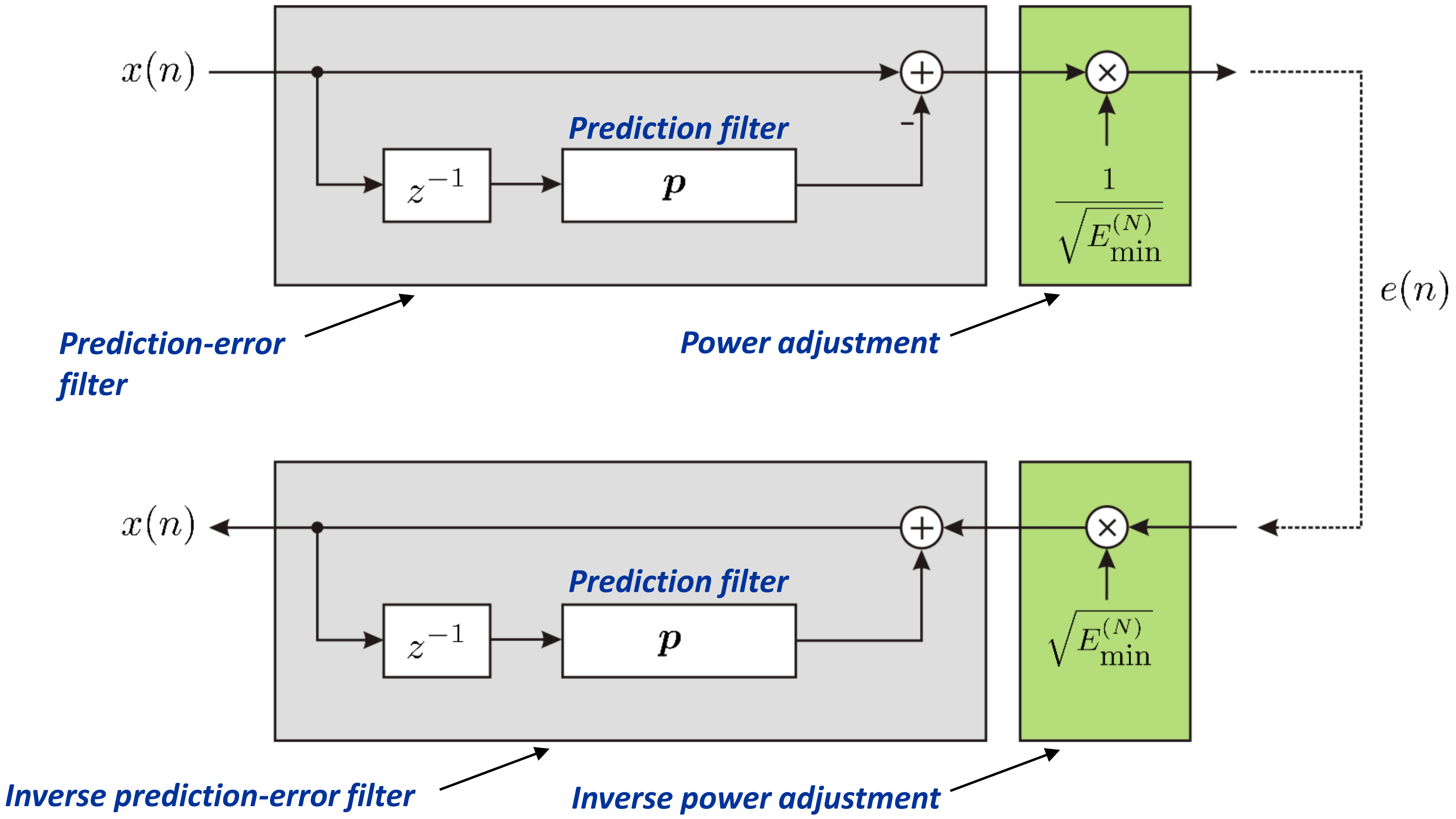
Predictor-based Filter Design

Effect of a Prediction-Error Filter in the Frequency Domain



Predictor-based Filter Design

Effect of a Prediction-Error Filter in the Frequency Domain



Predictor-based Filter Design

Minimization without side conditions (they are included in the filter structure)

- Cost function:

$$\mathrm{E}\{e^2(n)\} \;=\; \mathrm{E}\left\{\left(x(n) - \sum_{i=0}^{N-1} p_i\, x(n-i-1)\right)^2\right\} \;\rightarrow\; \min$$

The resulting prediction-error filter is minimum phase:

- An FIR-filter is computed, whose zeros are all inside the unit circle.
- Signals can pass the filter "maximally fast".
- The inverse prediction filter (an IIR filter) is therefore automatically stable because all zeros turn into poles which are now inside the unit circle as well.

The resulting filters are normalized:

- Frequency response of the prediction-error filter: $P_{\mathrm{PF}}\big(e^{j\Omega}\big) = 1 - \sum_{i=1}^{N} p_{i-1}\, e^{-j\Omega i}$
- Frequency response of the inverse filter: $P_{\mathrm{inv.\,PF}}\big(e^{j\Omega}\big) = \dfrac{1}{1 - \sum_{i=1}^{N} p_{i-1}\, e^{-j\Omega i}}$

Predictor-based Filter Design

Cost function:

- Minimization of the average error power

$$\mathrm{E}\{e^2(n)\} \;\rightarrow\; \min$$

$$r(k) = \mathrm{E}\{x(n)\,x(n+k)\}$$

- Yule-Walker equation system

$$\underbrace{\begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(N) \end{bmatrix}}_{\boldsymbol{r}_{ss}(1)} = \underbrace{\begin{bmatrix} r(0) & r(1) & \dots & r(N-1) \\ r(1) & r(0) & \dots & r(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(N-1) & r(N-2) & \dots & r(0) \end{bmatrix}}_{\boldsymbol{R}_{ss}} \underbrace{\begin{bmatrix} p_{\mathrm{opt},0} \\ p_{\mathrm{opt},1} \\ \vdots \\ p_{\mathrm{opt},N-1} \end{bmatrix}}_{\boldsymbol{p}_{\mathrm{opt}}}$$

Robust and computationally efficient solution:

- Levinson-Durbin recursion

Predictor-based Filter Design

Interpretation of the impulse response as a signal:

With this point of view, the autocorrelation function can be estimated directly out of the impulse response

$$r(k) = \frac{1}{M} \sum_{i=0}^{M-k} h_i \, h_{i+k}.$$

Magnitude frequency response can be interpreted as a power density spectrum:

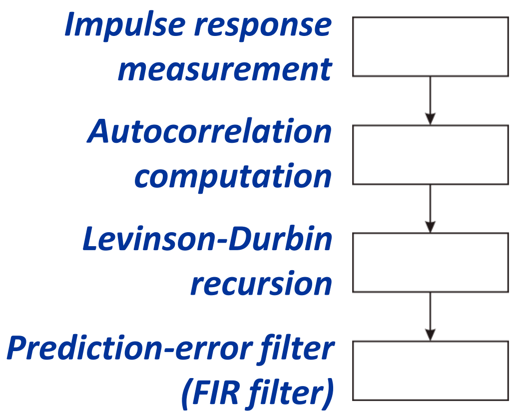
With this point of view, the autocorrelation function can be estimated directly out of the magnitude frequency response by

$$r(k) = \frac{1}{M} \sum_{i=0}^{M} \left| H(e^{j\frac{2\pi}{M}i}) \right|^2 e^{j\frac{2\pi}{M}ik}.$$

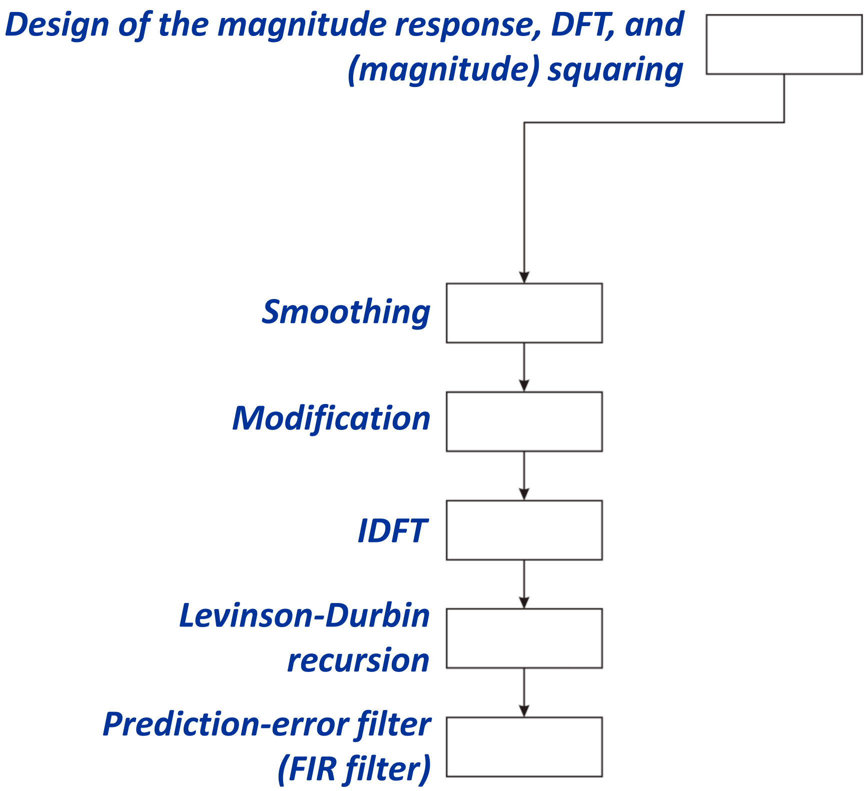
This gives the option to modify the frequency response prior to computing the IDFT.

Christian-Albrechts-Universität zu Kiel

Predictor-based Filter Design

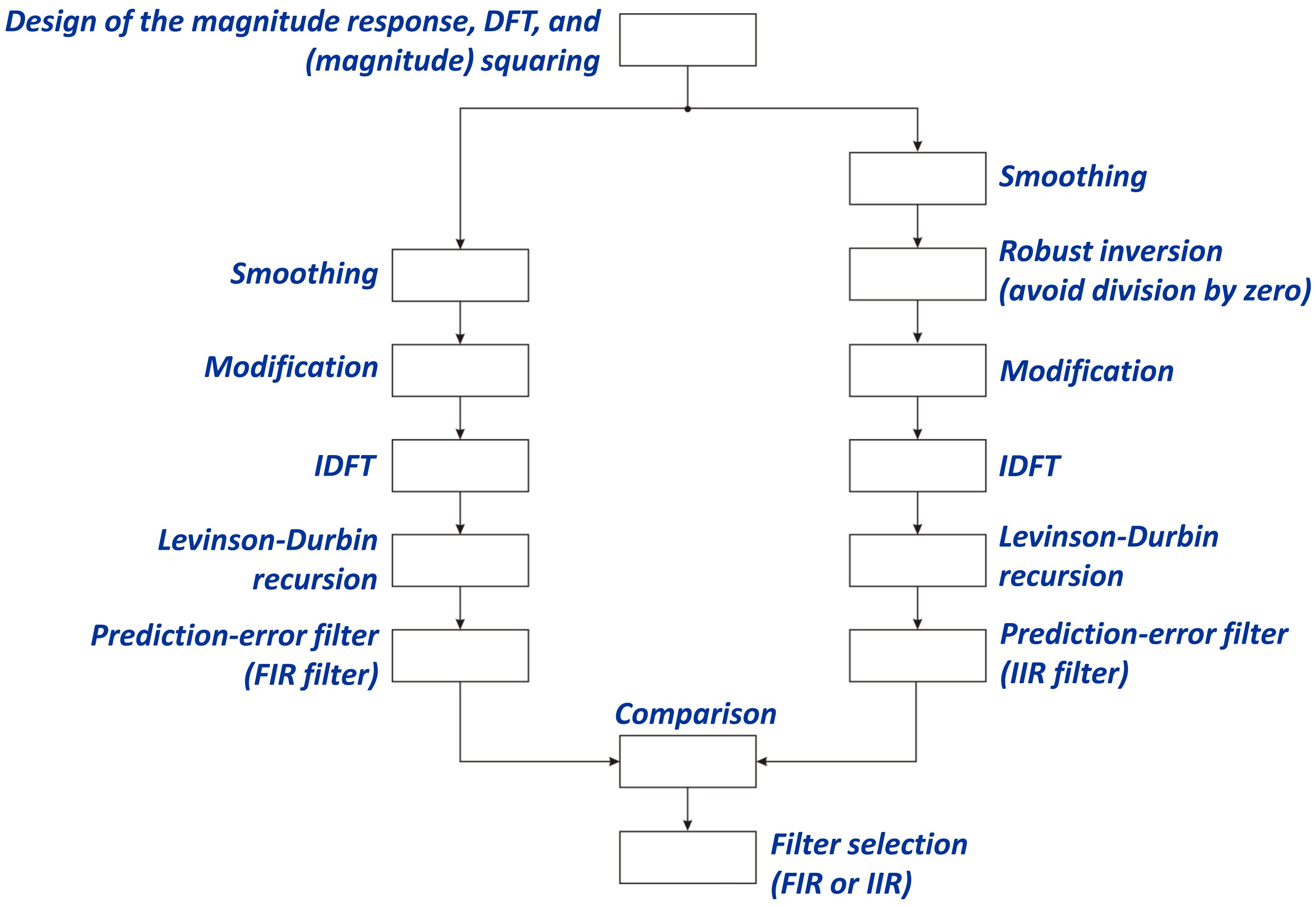


Predictor-based Filter Design

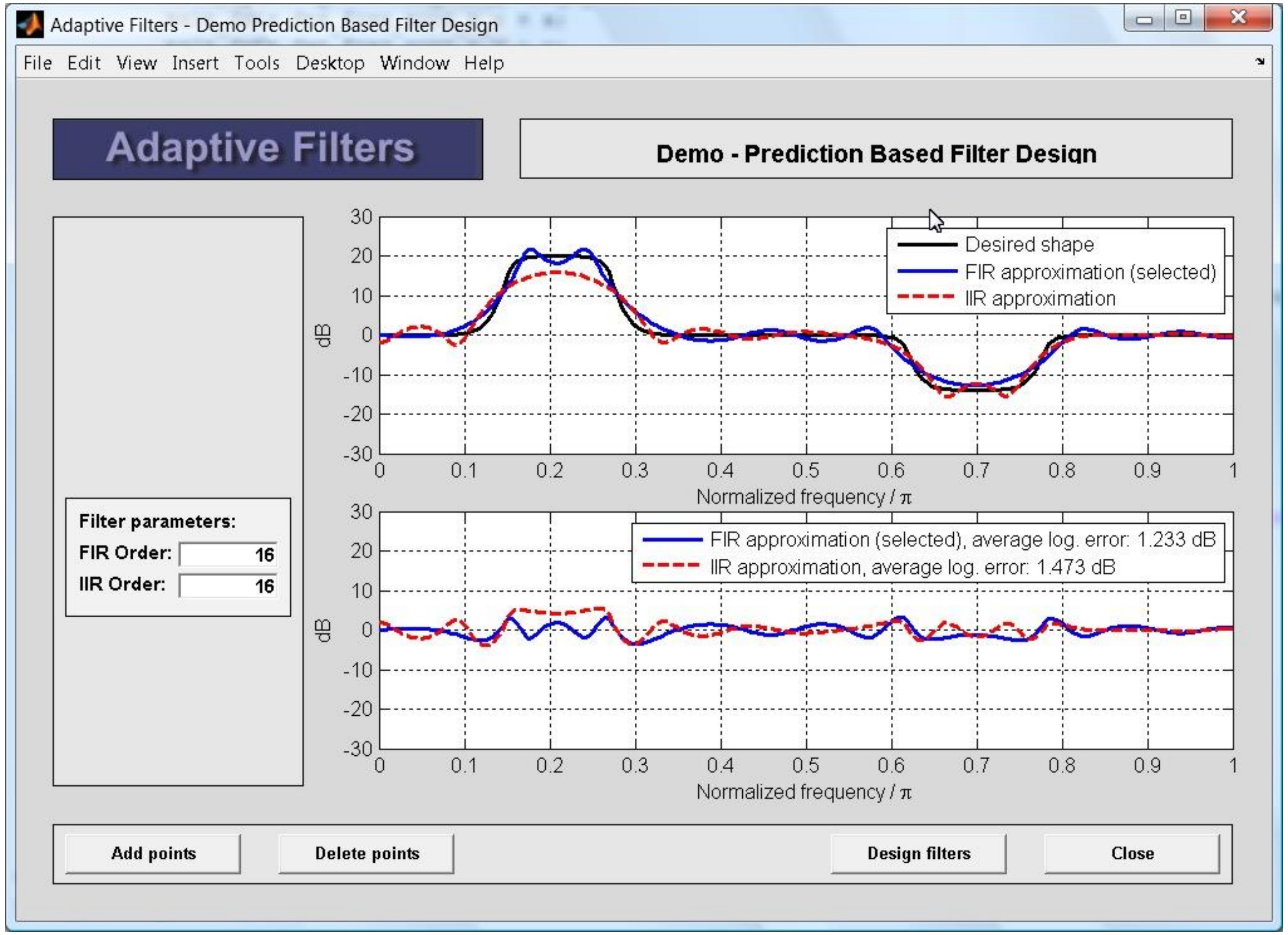


Christian-Albrechts-Universität zu Kiel

Predictor-based Filter Design



Predictor-based Filter Design



Linear-Phase Filter Structures – Part 1

The system properties of FIR filters can be used to design a linear-phase equalization filter. If an FIR filter with frequency response

$$P\left(e^{j\Omega}\right) = \sum_{i=0}^{N-1} p_i\, e^{-j\Omega i}$$

is mirrored with respect to time, i.e.,

$$\bar{p}_i = p_{N-i},$$

the frequency response of the mirrored filter is

$$\begin{aligned}\bar{P}\left(e^{j\Omega}\right) &= \sum_{i=-\infty}^{\infty} \bar{p}_i\, e^{-j\Omega i} = \sum_{i=-\infty}^{\infty} p_{N-i}\, e^{-j\Omega i} \\ &= \sum_{i=-\infty}^{\infty} p_i\, e^{-j\Omega(N-i)} = e^{-j\Omega N} \sum_{i=0}^{N-1} p_i\, e^{j\Omega i} \\ &= e^{-j\Omega N}\, P^*\left(e^{j\Omega}\right).\end{aligned}$$

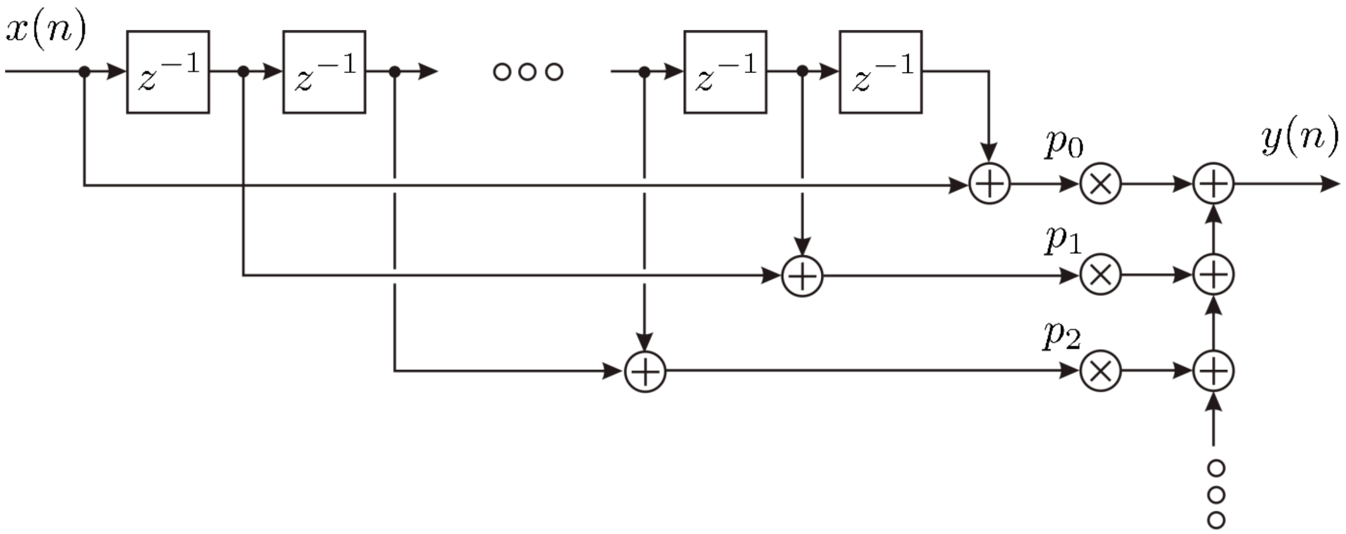
Both filters connected in series result therefore in a linear-phase system. The attenuation (or gain) properties of the entire filter is doubled compared to a single filter.

Predictor-based Filter Design

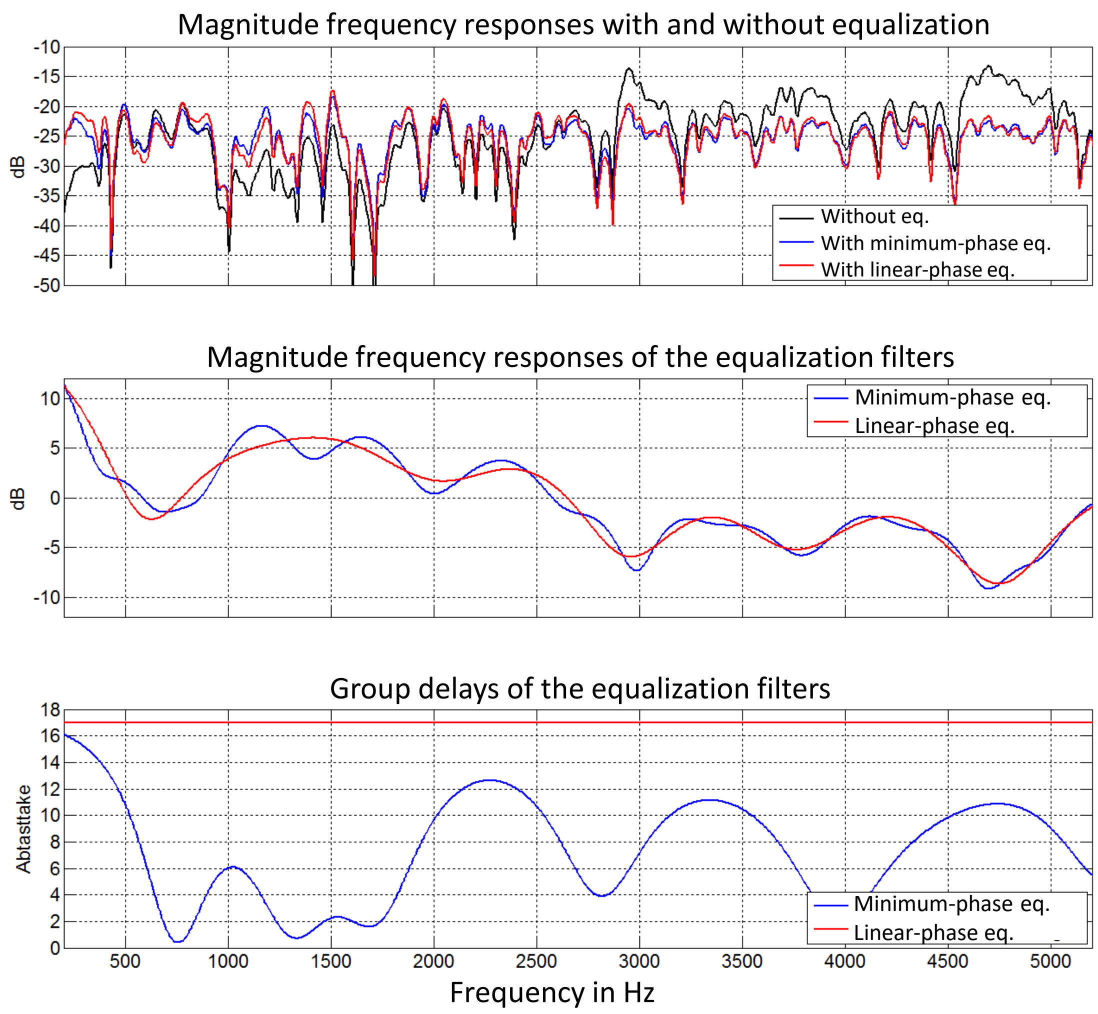
Linear-Phase Filter Structures – Part 2

The property described on the previous slide can be exploited. In order not to do a “double” equalization, the filter is designed only with the magnitude spectrum (instead of the power density spectrum of the filter). This is a simple way to achieve halving (in the logarithmic domain) or taking the square root (in the linear domain) of the desired frequency response

The block diagram of the filter structure looks like this:



Predictor-based Filter Design



Example for FIR-based equalization filters

- 32 coefficients
- No additional modifications in the frequency domain
- Minimum-phase and linear-phase approach

Basics of IIR-Filter Design – Part 1

- In the following only design algorithms are discussed which ***convert an analog into a digital filter***. However, there are also numerous algorithms for directly designing an IIR filter in the z-domain (frequency sampling method, least-squares design).
- Why starting with an analog filter?
  Analog filter design is a well developed field (lots of existing design catalogs).
- The problem can be defined in the z-domain, transformed into the s-domain and solved there, and finally transformed back into the z-domain.
- Analog filter: Transfer function

$$H_{\mathrm{a}}(s) = \frac{N(s)}{D(s)} = \frac{\sum\limits_{n=0}^{M} \beta_n \, s^n}{\sum\limits_{n=0}^{N} \alpha_n \, s^n}$$

with the filter coefficients $\alpha_n, \; \beta_n$ and the filter order $N \geq M$.

Basics of IIR-Filter Design – Part 2

Furthermore: Definition of the Laplace Transform

$$H_{\mathrm{a}}(s) \;=\; \int_{-\infty}^{\infty} h(t)\, e^{-st}\, dt.$$

- Note that linear-phase designs are not possible for causal and stable IIR Filters, since the condition

$$H(z) \;=\; \pm z^{-N} H(z^{-1})$$

has to be satisfied.

$\rightarrow$ Mirror-image pole outside the unit-circle for every pole inside the unit circle.

$\rightarrow$ Unstable filter.

Design of IIR Filters – Part 3

Filter design by impulse invariance – Part 1

Goal:

Design an IIR filter with an impulse response h_n being the sampled version of the impulse response $h_\mathrm{a}(t)$ of a given analog filter

$$h_n = h_\mathrm{a}(nT), \quad n = 0, 1, 2, \ldots$$

with T being the sampling interval. For the frequency response (ideal sampling assumed) we obtain:

$$H(e^{j\Omega})\Big|_{\Omega=\omega T} = \frac{1}{T} \sum_{\mu=-\infty}^{\infty} H_\mathrm{a}\Big(j\big(\omega - \frac{2\pi}{T}\mu\big)\Big).$$

Remarks:

- T should be selected sufficiently small to avoid aliasing.
- The method is not suitable to design highpass filters due to the large amount of possible aliasing.

Filter design by impulse invariance – Part 2

Suppose that the poles of the analog filter are distinct. In that case we can transform the transfer function into a ***partial-fraction expansion*** of $H_{\mathrm{a}}(s)$:

$$H_{\mathrm{a}}(s) = \sum_{i=0}^{N-1} \frac{A_i}{s - s_{\infty,i}},$$

with A_i : coefficients of the partial-fraction expansion,

$s_{\infty,i}$: poles of the analog filter.

The ***inverse Laplace transform*** yields

$$h_{\mathrm{a}}(t) = \begin{cases} \sum\limits_{i=0}^{N-1} A_i\, e^{s_{\infty,i} t}, & \text{if } t \geq 0, \\ 0, & \text{else.} \end{cases}$$

Sampling of $h_{\mathrm{a}}(t)$ yields

$$h_n = h_{\mathrm{a}}(nT) = \sum_{i=0}^{N-1} A_i\, e^{s_{\infty,i} nT}.$$

Filter design by impulse invariance – Part 3

We obtain for the transfer function of h_n:

$$H(z) = \sum_{n=0}^{\infty} h_n\, z^{-n}$$

… inserting the computation of h_n (see last slide) …

$$= \sum_{n=0}^{\infty} \left[\sum_{i=0}^{N-1} A_i\, e^{s_{\infty,i} nT} \right] z^{-n}$$

… changing the summation order …

$$= \sum_{i=0}^{N-1} A_i \sum_{n=0}^{\infty} \left[e^{s_{\infty,i} T}\, z^{-1} \right]^n$$

… using the summation $\sum_{k=0}^{\infty} a^k = \frac{1}{1-a}$ *for* $|a| < 1$ *…*

$$= \sum_{i=0}^{N-1} \frac{A_i}{1 - e^{s_{\infty,i} T}\, z^{-1}}.$$

Filter design by impulse invariance – Part 4

Thus, given an analog filter $H_\mathrm{a}(s)$ with poles $s_{\infty,i}$ the transfer function of the corresponding digital filter using the impulse invariant transform is:

$$H(z) = \sum_{i=0}^{N-1} \frac{A_i}{1 - e^{s_{\infty,i}T}\, z^{-1}},$$

with poles at $z_{\infty,i} = e^{s_{\infty,i}T},\ i = 0,\ 1,\ ...,\ N-1\,.$

Note: This result holds only for ***distinct poles***. The generalization to multiple-order poles is possible.

Filter design by impulse invariance – Part 5

Example:

Problem:

Convert the analog filter with the transfer function

$$H_\mathrm{a}(s) \;=\; \frac{s+0.1}{(s+0.1)^2+9}$$

into a digital filter using the impulse invariant method.

Solution:

The poles of $H_\mathrm{a}(s):\; s_{\infty,0,1} \;=\; -0.1 \pm j3$. Partial fraction expansion yields:

$$H_\mathrm{a}(s) \;=\; \frac{0.5}{s-(-0.1+j3)} + \frac{0.5}{s-(-0.1-j3)}.$$

$$\frac{1}{s-s_\infty} \;\Leftrightarrow\; \frac{1}{1-e^{s_\infty T}\,z^{-1}}$$

We finally have:

$$H(z) \;=\; \frac{0.5}{1-e^{-(0.1+j3)T}\,z^{-1}} + \frac{0.5}{1-e^{-(0.1-j3)T}\,z^{-1}}$$

$$=\; \frac{1-\left[e^{-0.1T}\cos(3T)\right]z^{-1}}{1-\left[2\,e^{-0.1T}\cos(3T)\right]z^{-1}+e^{-0.2T}\,z^{-2}}.$$

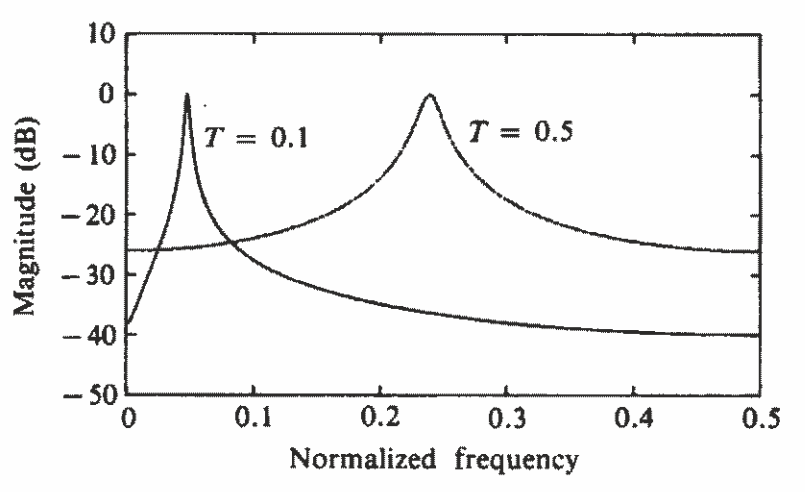
Design of IIR Filters – Part 8

Filter design by impulse invariance – Part 6

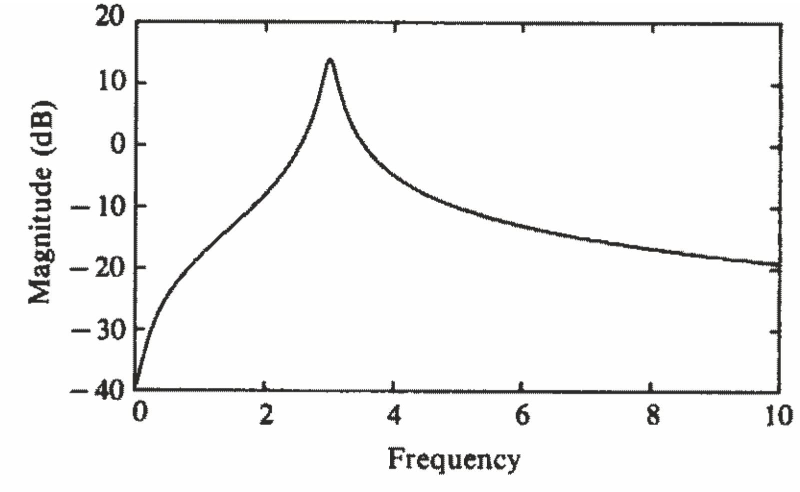
Example (continued):

Magnitude frequency responses:

Digital filter: $20\log_{10}|H(e^{j\Omega})|$



Analog filter: $20\log_{10}|H_{\mathrm{a}}(j\omega)|$



[Proakis, Manolakis, 1996]

Design of IIR Filters – Part 9

Bilinear transform – Part 1

Algebraic transform between the variables s and z.

$\rightarrow$ Mapping of the entire $j\omega$-axis of the s-plane to one revolution of the unit circle in the z-plane.

Definition:

$$s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right),$$

T denoting the sampling interval.

The transfer function of the corresponding digital filter can be obtained from the transfer function of the analog filter $H_{\mathrm{a}}(s)$ according to

$$H(z) = H_{\mathrm{a}}\left(\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)\right).$$

Design of IIR Filters – Part 10

Bilinear transform – Part 2

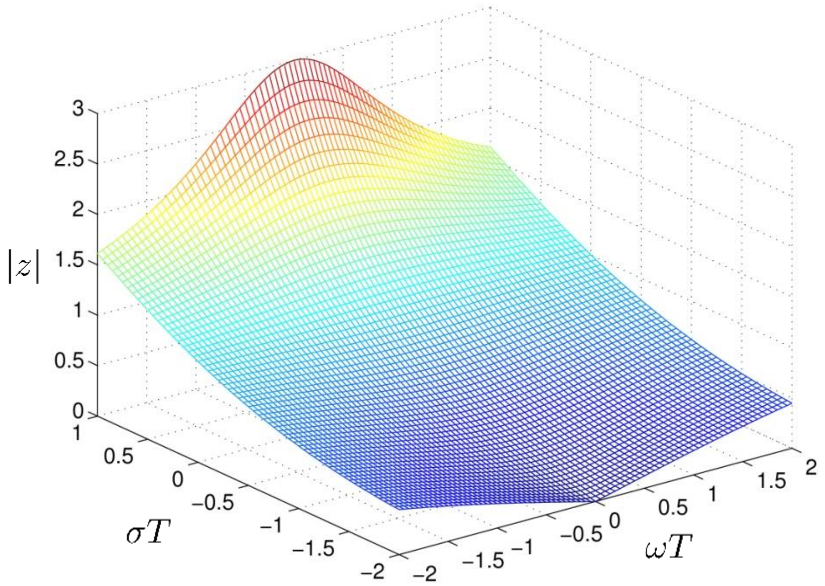
Properties:

- Rearranging the definition for z yields

$$z = \frac{1+(T/2)s}{1-(T/2)s}.$$

By substituting $s = \sigma + j\omega$ we obtain

$$z = \frac{1+\sigma T/2 + j\omega T/2}{1-\sigma T/2 - j\omega T/2}.$$



If $\sigma < 0 \rightarrow |z| < 1,$ and if $\sigma > 0 \rightarrow |z| > 1 \quad \forall\omega$
$\Rightarrow$ causal, stable continuous-time filters map into causal stable discrete-time filters.

- By inserting $s = j\omega$ into the above expression, it can be seen that $|z| = 1$ for all values of s on the $j\omega$-axis.
$\Rightarrow$ ***The $j\omega$-axis maps onto the unit circle (meaning that the bilinear transform is unique).***

Bilinear transform – Part 3

Properties (continued):

- Relationship between Ω and ω:
  Inserting $s = j\omega$ and $z = e^{j\Omega}$ into the definition

$$
\begin{aligned}
j\omega &= \frac{2}{T}\frac{1-e^{-j\Omega}}{1+e^{-j\Omega}} = \frac{2}{T}\frac{j\sin(\frac{\Omega}{2})}{\cos(\frac{\Omega}{2})} \\
&= \frac{2j}{T}\tan\left(\frac{\Omega}{2}\right).
\end{aligned}
$$

$\Rightarrow$ Nonlinear mapping between Ω and ω (warping of the frequency axis) according to

$$
\begin{aligned}
\omega &= \frac{2}{T}\tan\left(\frac{\Omega}{2}\right), \\
\Omega &= 2\arctan\left(\frac{\omega T}{2}\right).
\end{aligned}
$$

Design of IIR Filters – Part 12

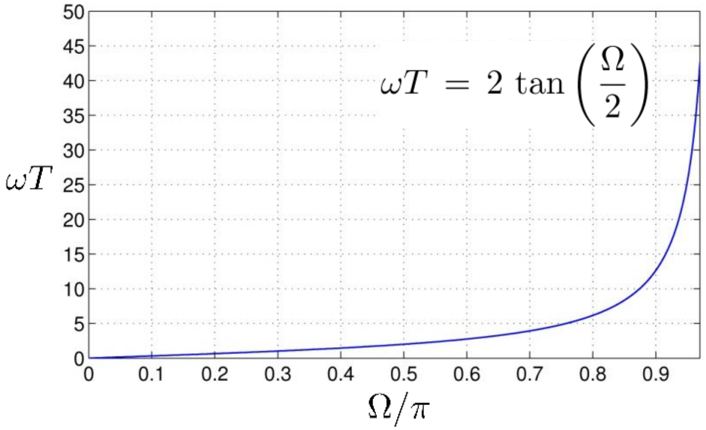
Bilinear transform – Part 4

Properties (continued):

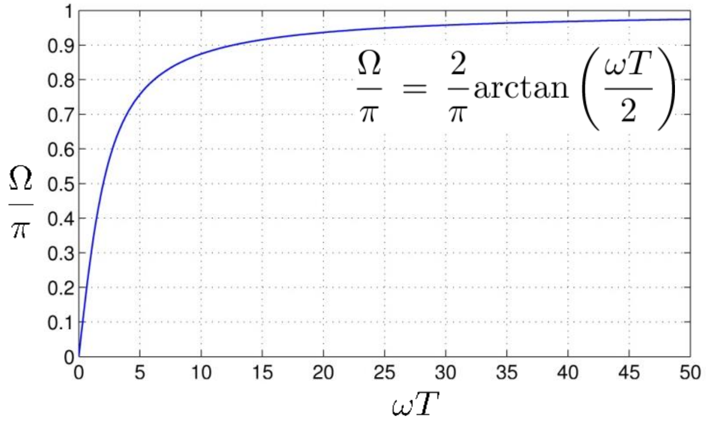
As a result we obtain for the nonlinear mapping between Ω and ω:

$$\omega = \frac{2}{T}\tan\left(\frac{\Omega}{2}\right),$$

$$\Omega = 2\arctan\left(\frac{\omega T}{2}\right).$$

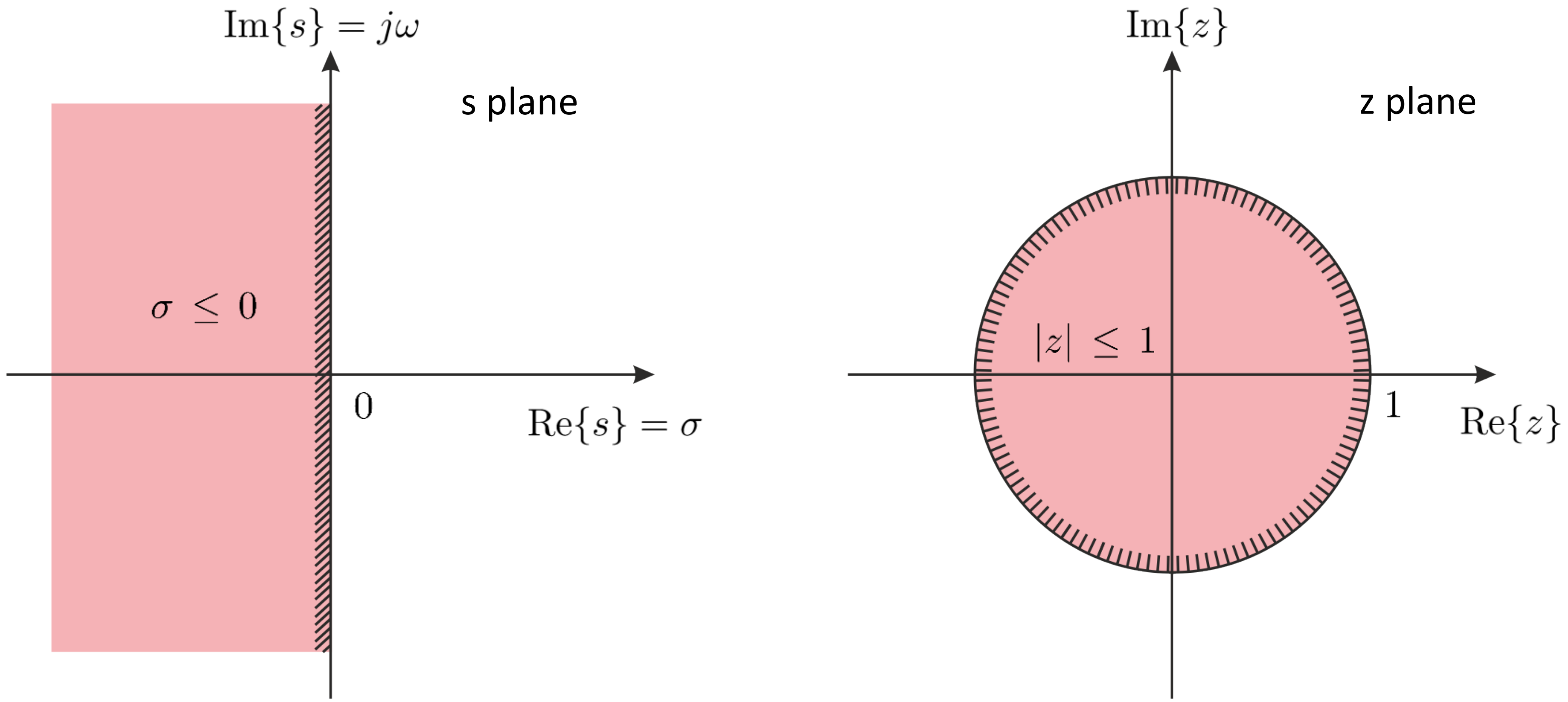


This could be interpreted as a warping of the frequency axis.



Design of IIR Filters – Part 13

Bilinear transform – Part 5



Bilinear transform – Part 6

Remarks:

- The design of a digital filter often begins with ***frequency specifications in the digital domain***, which are ***converted to the analog domain***. The ***analog filter is then designed*** considering these specifications (i.e. using the classical approaches from the following section) and ***converted back into the digital domain using the bilinear transform***.
- When using this procedure, the parameter T cancels out and thus can be set to an arbitrary value ($T = 1$).

Example:

Problem:

Design a digital single-pole lowpass filter with –3 dB frequency (cutoff frequency) of $\Omega_\mathrm{c} = 0.2\pi$, using the bilinear transform applied to the analog filter with the transfer function

$$H_\mathrm{a}(s) = \frac{\omega_\mathrm{c}}{s + \omega_\mathrm{c}},$$

with ω_c denoting the analog cut-off frequency.

Bilinear transform – Part 7

Example (continued):

Solution:

ω_c is obtained from $\Omega_c = \frac{\pi}{5}$ with

$$\omega_c = \frac{2}{T}\tan\left(\frac{\Omega_c}{2}\right)\Big|_{\Omega_c=\frac{\pi}{5}} = \frac{2}{T}\tan\left(\frac{\pi}{10}\right) \approx \frac{0.65}{T}.$$

The analog filter has now the transfer function

$$H_a(s) = \frac{0.65/T}{s+0.65/T},$$

which is transformed back into the digital domain by using the bilinear transform

$$s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right).$$

Design of IIR Filters – Part 16

Bilinear transform – Part 8

Example (continued):

Solution:

The transfer function of the digital filter is

$$H(z) \;=\; \frac{0.245\,(1+z^{-1})}{1-0.509\,z^{-1}}.$$

Note that the parameter T has been divided out.

The frequency response is

$$H(e^{j\Omega}) \;=\; \frac{0.245\,(1+e^{-j\Omega})}{1-0.509\,e^{-j\Omega}}.$$

Especially we have $H(e^{j0}) = 1$, and $\left|H(e^{j0.2\pi})\right| \;=\; 0.707$, which is the desired response.

Design of IIR Filters – Part 17

Questions:

Partner work – Please think about the following question and try to find answers (first group discussions, afterwards broad discussion in the whole group).

- What can you set/adjust when using the impulse invariance method?

  ………

  ………

- If you would have to specify properties of a method that maps the Laplace domain into the z-domain, what would you mention?

  ………

  ………

- When using the bilinear transform, where is the imaginary axis mapped in the z-domain?

  ………

  ………

Design of IIR Filters – Part 18

Characteristics of commonly used analog filters – Part 1

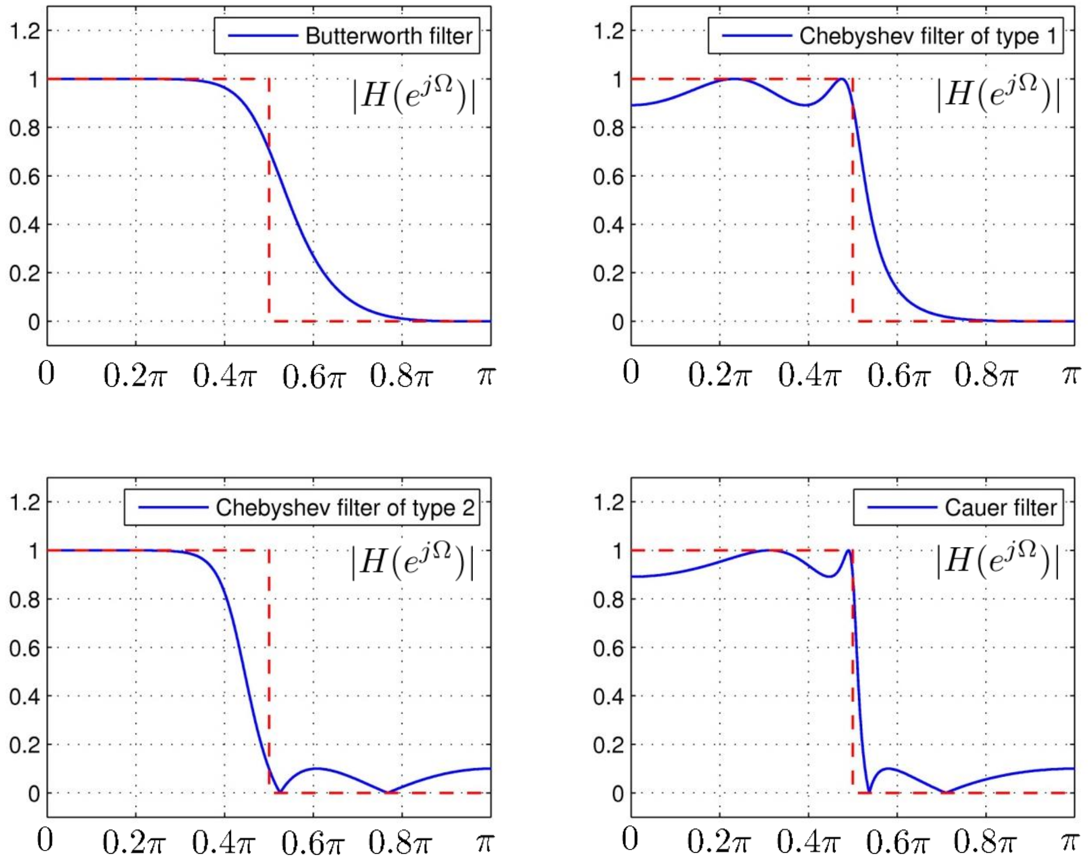
- The design of a digital filter can be reduced to the ***design of an appropriate analog filter*** and then performing the ***conversion*** from $H(s)$ to $H(z)$.
- In the following we briefly discuss the characteristics of commonly used analog (lowpass) filters. We will focus here on ***four different types of IIR filters***:
  - Butterworth filters
  - Type 1 Chebyshev filters
  - Type 2 Chebyshev filters
  - Cauer filters

Characteristics of commonly used analog filters – Part 2

Examples for the different filter types:

All filter are of order 4.

The bilinear transform has been used to create discrete filters.



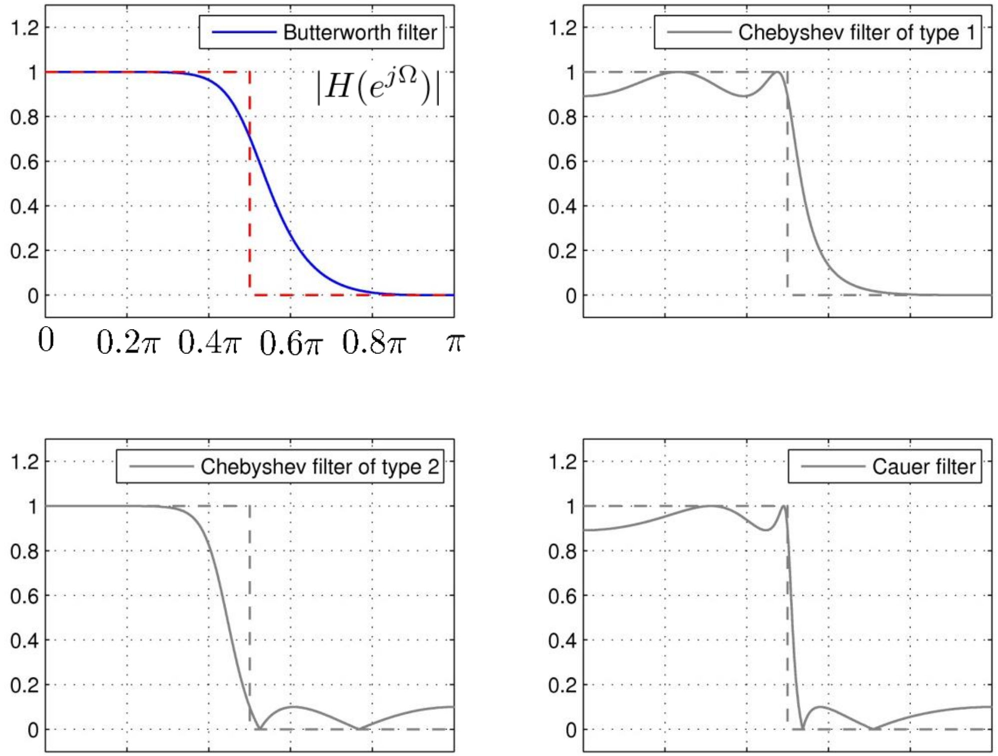
Characteristics of commonly used analog filters – Part 3

Butterworth filters

Lowpass Butterworth filters are allpole-filters characterized by the squared magnitude frequency response

$$\left|H(\omega)\right|^2 = \frac{1}{1+\left(\frac{\omega}{\omega_{\text{cut}}}\right)^{2N}},$$

N is the order of the filter, ω_{cut} is the -3 dB frequency (cut-off frequency).



Design of IIR Filters – Part 21

Characteristics of commonly used analog filters – Part 4

Butterworth filters (continued)

Since $H(s) \cdot H(-s)\big|_{s=j\omega} = \big|H(j\omega)\big|^2$, we get by analytic continuation into the whole s-plane

$$H(s) \cdot H(-s) = \frac{1}{1 + \left(\frac{-s^2}{\omega_{\text{cut}}^2}\right)^N}.$$

Poles of $H(s) \cdot H(-s)$:

$$\frac{-s^2}{\omega_{\text{cut}}^2} = (-1)^{1/N} = e^{j(2i+1)\pi/N}$$

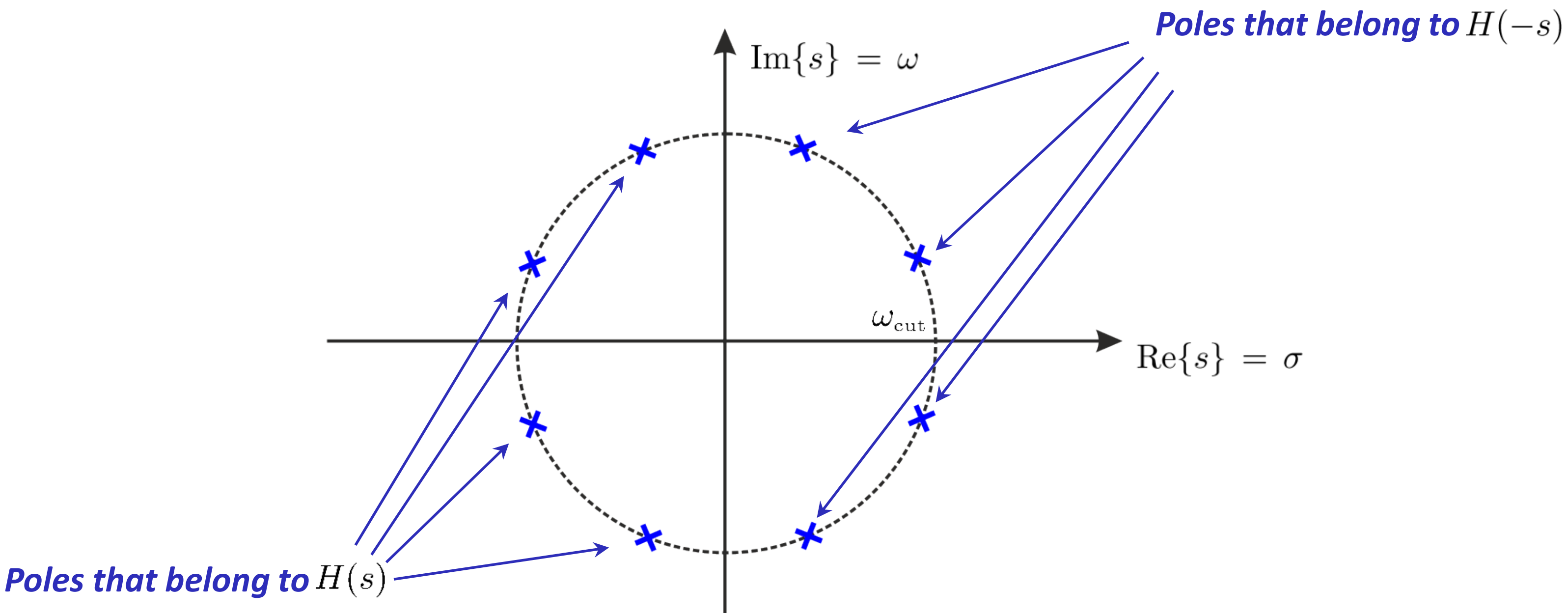
$$\rightarrow \quad s_{\infty,i} = \omega_{\text{cut}}\, e^{j\frac{\pi}{2}}\, e^{j\frac{2i+1}{2N}\pi}, \quad i = 0,\, 1,\, \ldots,\, 2N-1.$$

- The $2N$ poles of $H(s)H(-s)$ occur on a circle of radius ω_{cut} at equally spaced points in the s-plane.
- N poles are located in the left half of the s-plane and belong to $H(s)$.
- The N remaining poles lie in the right half of the s-plane and belong to $H(-s)$ (stability!).

Characteristics of commonly used analog filters – Part 5

Butterworth filters (continued)

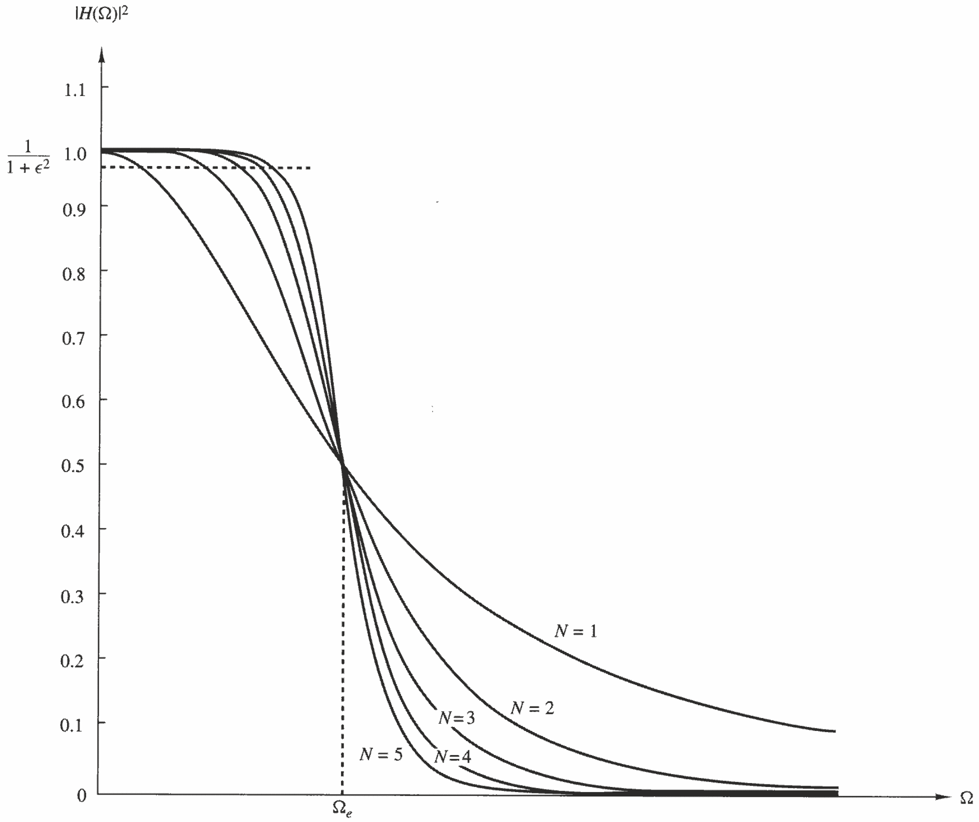
Pole locations in the s-plane for $N = 4$:



Characteristics of commonly used analog filters – Part 6

Butterworth filters (continued)

Frequency responses ($\Omega \rightarrow \omega,\ |H(\omega_p)|^2 = 1/(1+\epsilon^2)$):



[Proakis, Manolakis, 1996]

Characteristics of commonly used analog filters – Part 7

Butterworth filters (continued)

Estimation of the required filter order:

At the stopband edge frequency ω_{stop} the squared magnitude frequency response can be written as

$$\frac{1}{1+\left(\frac{\omega_{\text{stop}}}{\omega_{\text{cut}}}\right)^{2N}} = \delta_2^2,$$

which leads to

$$N = \frac{\log_{10}\left([1/\delta_2^2]-1\right)}{2\log_{10}(\omega_{\text{stop}}/\omega_{\text{cut}})}.$$

Characteristics of commonly used analog filters – Part 8

Butterworth filters (continued)

Example:

Problem:

Determine the order and the poles of a lowpass Butterworth filter that has a -3 dB bandwidth of 500 Hz and an attenuation of 40 dB at 1000 Hz,

- -3 dB frequency $\omega_{\text{cut}} = 2\pi \cdot f_{\text{cut}} = 1000\,\pi\,\text{Hz},$
- stopband frequency $\omega_{\text{stop}} = 2\pi \cdot f_{\text{stop}} = 2000\,\pi\,\text{Hz},$
- attenuation of 40 dB $\Rightarrow \delta_2 = 0.01.$

Solution:

For the order we obtain

$$N = \frac{\log_{10}(10^4 - 1)}{2\log_{10} 2} = 6.64.$$

In order to be "on the safe side" we choose $N = 7.$

Design of IIR Filters – Part 26

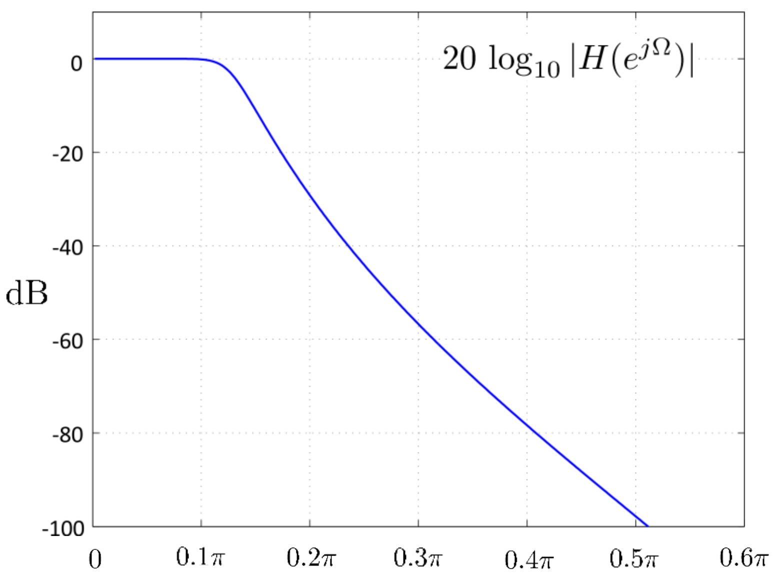
Characteristics of commonly used analog filters – Part 9

Butterworth filters (continued)

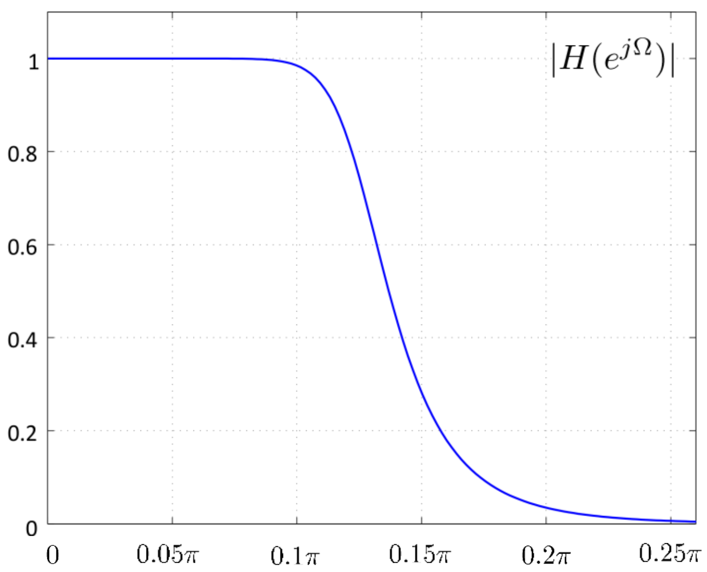
Example (continued):

Properties of the resulting digital filter (transformation by the bilinear transform, $f_s = 8000\,\text{Hz}$)

Magnitude frequency response



Transition band



Design of IIR Filters – Part 27

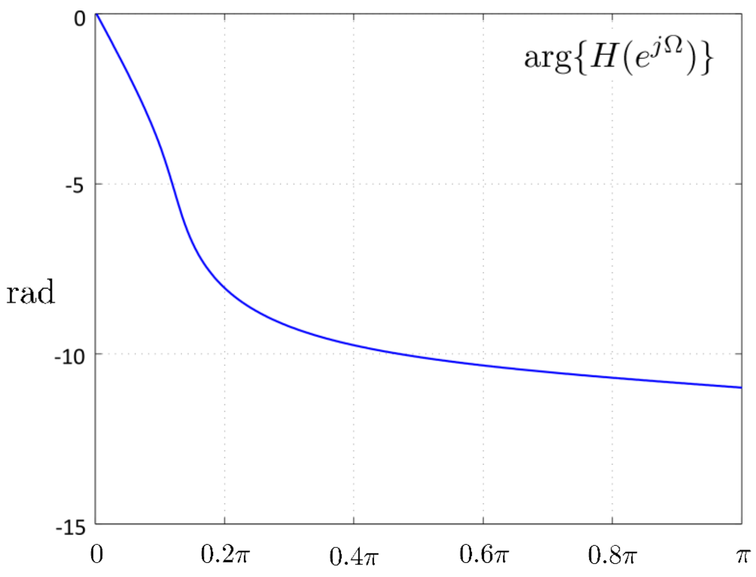
Characteristics of commonly used analog filters – Part 10

Butterworth filters (continued)

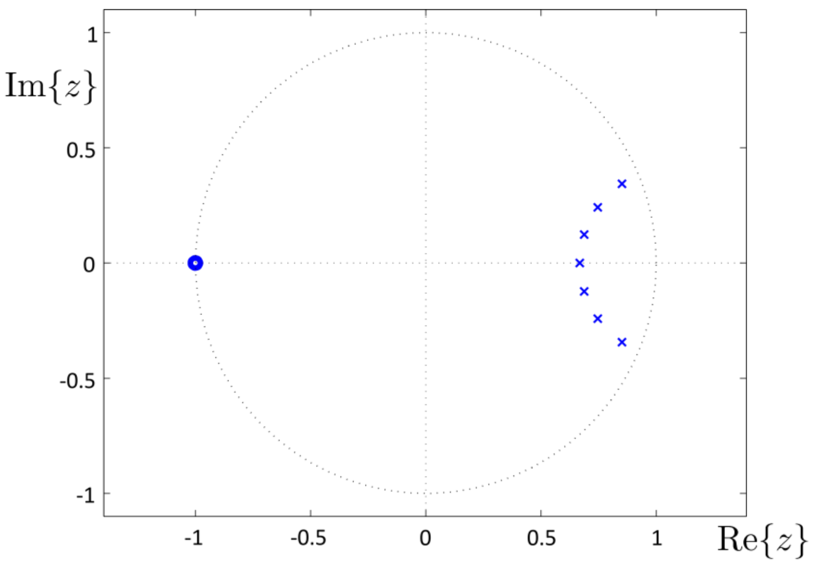
Example (continued):

Properties of the resulting digital filter (transformation by the bilinear transform, $f_s = 8000\,\mathrm{Hz}$)

Phase response



Pole/zero locations



Design of IIR Filters – Part 28

Characteristics of commonly used analog filters – Part 11

Chebyshev filters

Two types of Chebyshev filters:

- Type 1 filters are all-pole filters with equiripple behavior in the passband and monotonic characteristic (similar to a Butterworth filter) in the stopband.
- Type 2 filters have poles and zeros (for finite s), and equiripple behavior in the stopband, but a monotonic characteristic in the passband.

Design of IIR Filters – Part 29

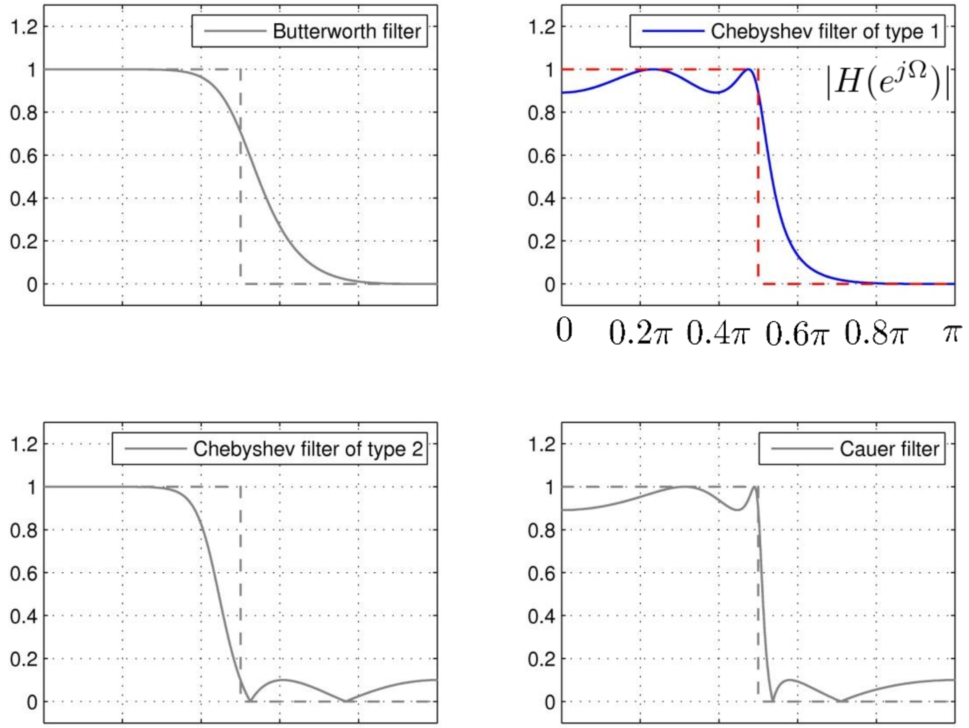
Characteristics of commonly used analog filters – Part 12

Type 1 Chebyshev filters

Squared magnitude frequency response:

$$\left|H(\omega)\right|^2 = \frac{1}{1+\epsilon^2\, T_N^2\left(\frac{\omega}{\omega_{\text{pass}}}\right)},$$

where ϵ is a parameter related to the passband ripple, and $T_N(x)$ is the N-th order ***Chebyshev polynomial*** (see next slide).



Characteristics of commonly used analog filters – Part 13

Type 1 Chebyshev filter (continued)

The Chebyshev polynomial is defined as

$$T_N(x) = \begin{cases} \cos\big(N\operatorname{arccos}(x)\big), & \text{for } |x| \leq 1, \\ \cosh\big(N\operatorname{arccosh}(x)\big), & \text{for } |x| > 1, \end{cases}$$

and can be obtained by the recursive equation

$$T_{N+1}(x) = 2\,x\,T_N(x) - T_{N-1}(x), \quad N = 1,\, 2,\, \dots$$

Examples:

- $T_0(x) = 1,\; T_1(x) = \cos\big(\arccos(x)\big) = x$
- $T_2(x) = \cos\big(2\arccos(x)\big) = 2\cos^2\big(\arccos(x)\big) - 1 = 2\,x^2 - 1$
- $T_3(x) = \cos\big(3\arccos(x)\big) = 4\cos^3\big(\arccos(x)\big) - 3\cos\big(\arccos(x)\big) = 4\,x^3 - 3\,x$

$T_N(x)$ represents a polynom of degree N in x.
$\Rightarrow$ Chebyshev behavior (minimizing the maximal error) in the passband (or in the stopband for 2 type filters)

Characteristics of commonly used analog filters – Part 14

Type 1 Chebyshev filter (continued)

The filter parameter ϵ is related to the passband ripple:

For N odd:

$$T_N(0) = 0 \quad \rightarrow \quad |H(0)|^2 = 1,$$

For N even:

$$T_N(0) = 1 \quad \rightarrow \quad |H(0)|^2 = \frac{1}{1+\epsilon^2}.$$

At the passband edge frequency $\omega = \omega_{\text{pass}}$ we have $T_N(1) = 1,$ such that

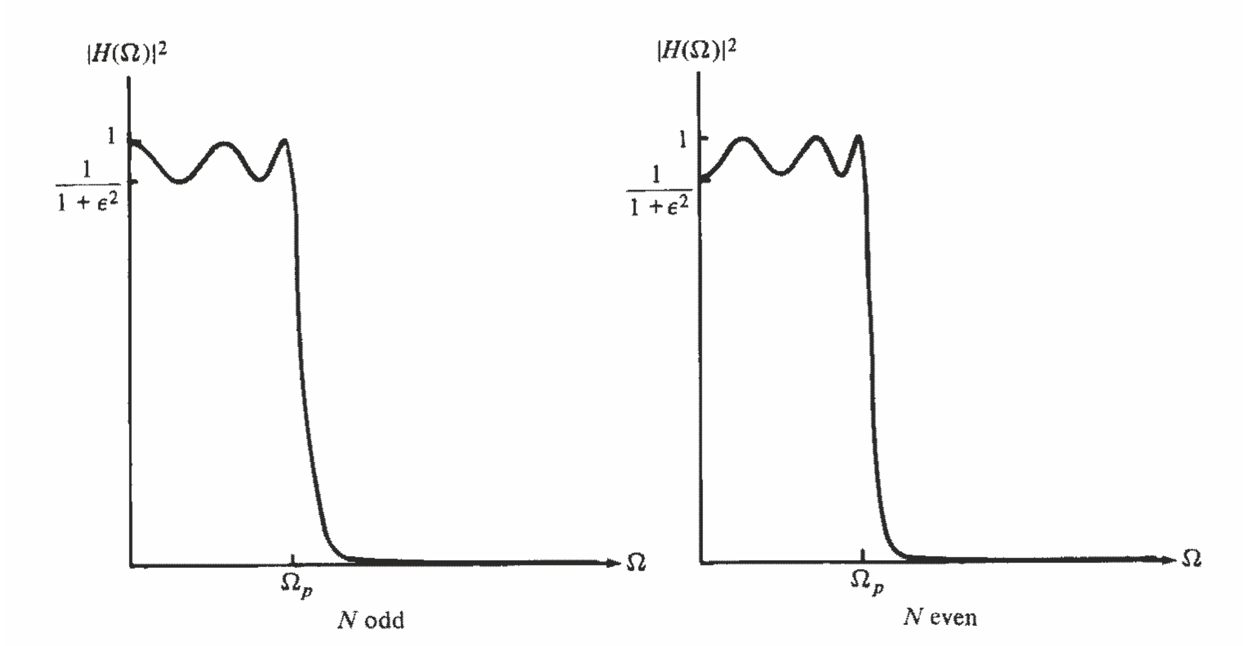
$$\frac{1}{\sqrt{1+\epsilon^2}} = 1 - \delta_1 \quad \Longleftrightarrow \quad \epsilon = \sqrt{\frac{1}{(1-\delta_1)^2} - 1},$$

which establishes a relation between the passband ripple δ_1 and the parameter ϵ.

Characteristics of commonly used analog filters – Part 15

Type 1 Chebyshev filter (continued)

Typical squared magnitude frequency responses for a Chebyshev type 1 filter ($\Omega \to \omega$):



[Proakis, Manolakis, 1996]

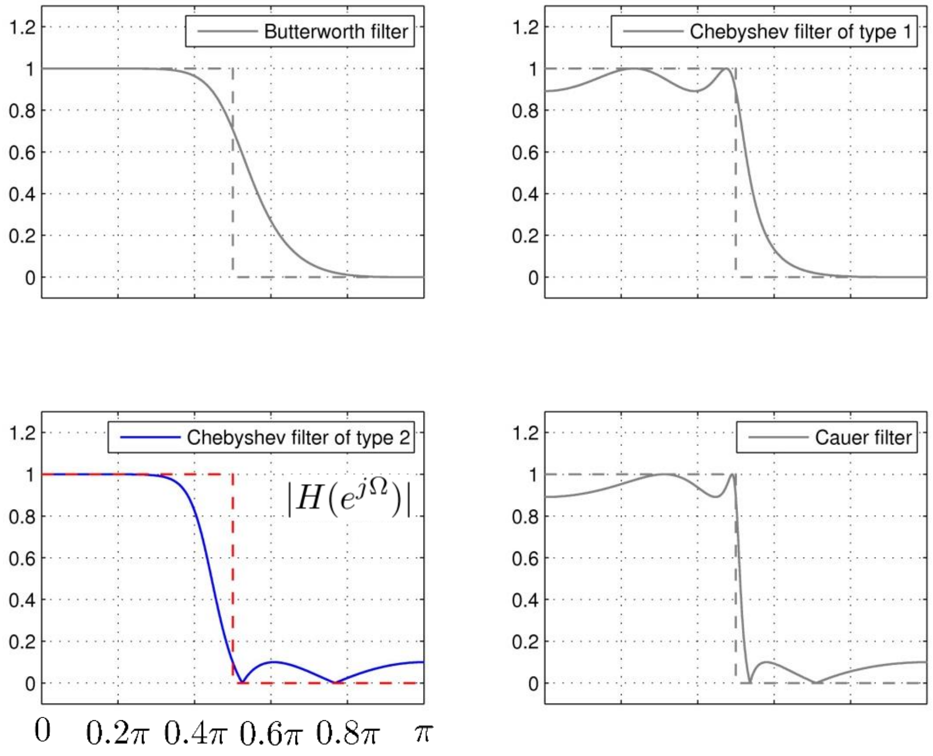
Design of IIR Filters – Part 33

Characteristics of commonly used analog filters – Part 16

Type 2 Chebyshev filter

Squared magnitude frequency response:

$$\left|H(\omega)\right|^2 = \frac{1}{1+\epsilon^2 \dfrac{T_N^2\left(\frac{\omega_{\text{stop}}}{\omega_{\text{pass}}}\right)}{T_N^2\left(\frac{\omega_{\text{stop}}}{\omega}\right)}}.$$



Design of IIR Filters – Part 34

Characteristics of commonly used analog filters – Part 17

Type 2 Chebyshev filter

Estimation of the ***filter order***:

Chebyshev filters only depend on the parameters $N,\ \epsilon,\ \delta_2$ and the ratio $\omega_{\text{stop}}/\omega_{\text{pass}}$.
Using these values, it can be shown that the required order can be estimated as

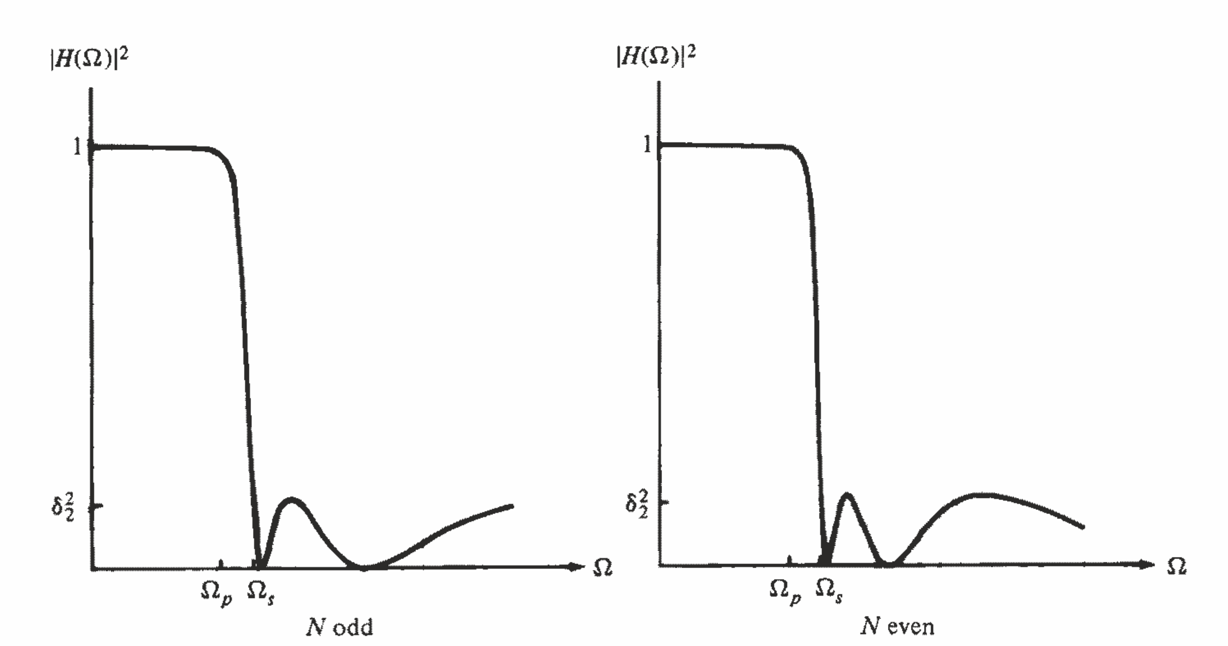
$$N = \frac{\log\left[\left(\sqrt{1-\delta_2^2}+\sqrt{1-\delta_2^2(1+\epsilon^2)}\right)/(\epsilon\,\delta_2)\right]}{\log\left[\frac{\omega_{\text{stop}}}{\omega_{\text{pass}}}+\sqrt{\left(\frac{\omega_{\text{stop}}}{\omega_{\text{pass}}}\right)^2-1}\right]}.$$

Design of IIR Filters – Part 35

Characteristics of commonly used analog filters – Part 18

Type 2 Chebyshev filter

Typical squared magnitude frequency responses for a Chebyshev type 2 filter ($\Omega \to \omega$):



[*Proakis, Manolakis, 1996*]

Design of IIR Filters – Part 36

Characteristics of commonly used analog filters – Part 19

Elliptic (Cauer) filters

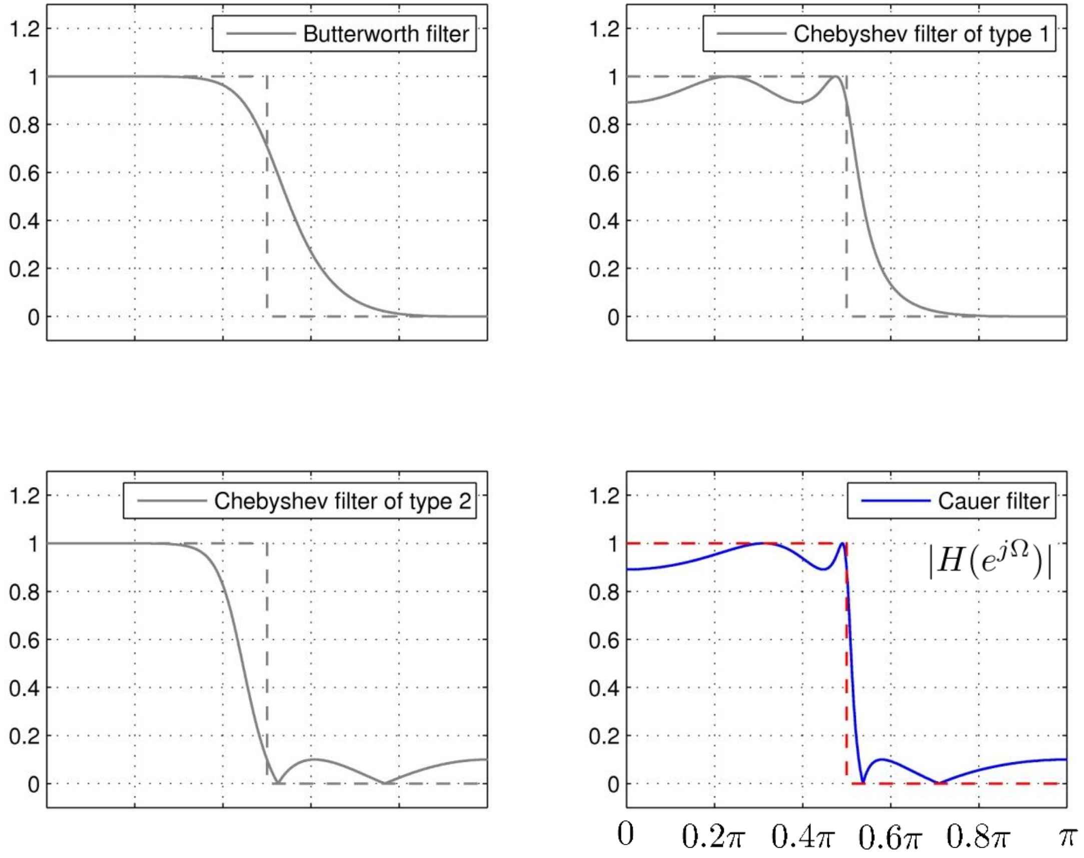
- Elliptic filters have equiripple (Chebyshev) behavior in both pass- and stopband.
- The transfer function contains both poles and zeros, where the zeros are located on the $j\omega$ -axis.
- The squared magnitude frequency response

$$\big|H(\omega)\big|^2 = \frac{1}{1+\epsilon^2\, U_N\left(\frac{\omega_{\text{stop}}}{\omega_{\text{pass}}}, \frac{\omega}{\omega_{\text{pass}}}\right)},$$

  where $U_N(x,y)$ denotes the Jacobian elliptic function of order N, and the parameter ϵ controls the passband ripple.
- The filter design is optimal in pass- and stopband in the equiripple sense:
  However, other types of filters may be preferred due to their better phase response characteristics (i.e. approximately linear-phase), for example the Butterworth filter.

Characteristics of commonly used analog filters – Part 20

Elliptic (Cauer) filters

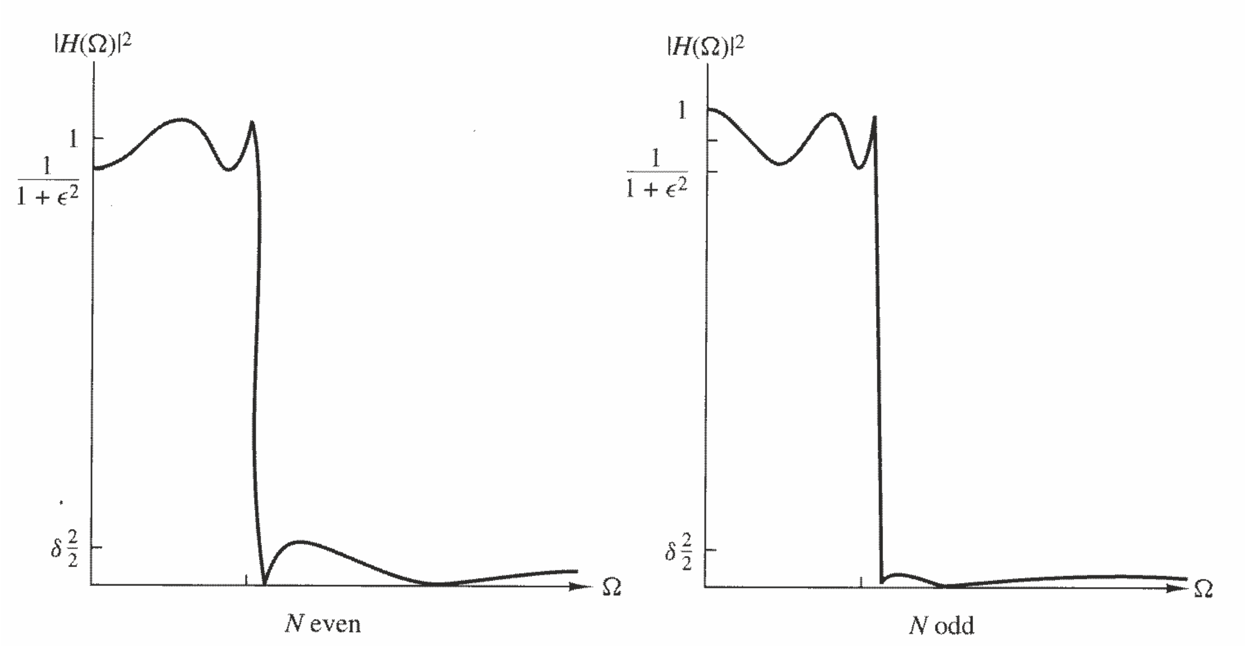


Design of IIR Filters – Part 38

Characteristics of commonly used analog filters – Part 21

Elliptic (Cauer) filters (continued)

Characteristic squared magnitude frequency responses for a elliptic filter ($\Omega \rightarrow \omega$):



[Proakis, Manolakis, 1996]

Design of IIR Filters – Part 39

Characteristics of commonly used analog filters – Part 22

Elliptic (Cauer) filters (continued)

Estimation of the filter order:

Required order to achieve the specifications with the parameters $\delta_1,\ \delta_2$ and $\omega_{\text{pass}}/\omega_{\text{stop}}$ ($1-\delta_1 \;=\; 1/\sqrt{1+\epsilon^2},\ 1-\delta_2 \;=\; 1/\sqrt{1+\delta^2}$):

$$N \;=\; \frac{K\left(\frac{\omega_{\text{pass}}}{\omega_{\text{stop}}}\right)\ K\left(\sqrt{1-\left(\frac{\epsilon}{\delta}\right)^2}\right)}{K\left(\frac{\epsilon}{\delta}\right)\ K\left(\sqrt{1-\left(\frac{\omega_{\text{pass}}}{\omega_{\text{stop}}}\right)^2}\right)},$$

where $K(x)$ denotes the complete elliptic integral of the first kind (tabulated)

$$K(x) \;=\; \int\limits_0^{\pi/2} \frac{d\theta}{\sqrt{1-x^2\sin^2\theta}}.$$

Summary

- Introduction
- Digital processing of continuous-time signals
- DFT and FFT
- ***Digital filters***
  - ***Structures for FIR systems***
  - ***Structures for IIR systems***
  - ***Coefficient quantization and round-off effects***
  - ***Design of FIR filters***
  - ***Design of IIR filters***
- Multi-rate digital signal processing