Wave Digital Filter Designer
Linear Phase Design Method

User Documentation

Hans-Georg Brachtendorf
Christian Zeintl

February 22, 2021

Contents

1 General

The Wave Digital Filter Designer (WDF Designer) is an extension to MATLAB™ that was originally developed by the Delft University of Technology [1]. In the course of a thesis, a newly developed method to design Wave Digital Filters with an approximately linear phase response has been added to this filter design toolbox. Therefore some changes were made to the existing toolbox to suit the new design method as well as the implementation of the design method itself.

The Lattice Wave Digital Filter (LWDF) is an implementation variant of the infinite impulse response (IIR) filter. This type generally causes a lower implementation effort for a given specification than a finite impulse response filter. The LWDF is a particularly favourable variation due to its minimized number of multipliers and its insensitivity w.r.t. the effects of parameter quantization. However infinite impulse response filters induce a non-constant group delay to the processed signal. This means that there is a variation of the signal processing delay for different frequencies, which leads to dispersion. Dispersion is a critical phenomenon if the designated application relies on the shape of the signal envelope. In audio applications sufficiently large group delay variations can lead to audible signal distortion as well as to effects of inter-symbol interference in digital modulation applications. This adverse effect is inherent to this implementation variant and cannot be eliminated completely. The Lattice Wave Digital Filter, which generates the required filter characteristic through a series of all-pass filters however can be designed in a way to minimize the group delay variation. Applying a novel design method, yields LWDFs with approximately constant group delay.

This document explains the updated user interface as a guideline for new users and presents some example designs for the linear phase design method. Since this document will only discuss the extension to the toolbox, it is highly recommended to also read the original documentation that can be found elsewhere [1].

This extension to the WDF Designer from Delft University of Technology is free for use. The University of Applied Sciences of Upper Austria does not overtake any warranty. For terms of use please check also the web-page of TU Delft [1].

2 Toolbox Overview

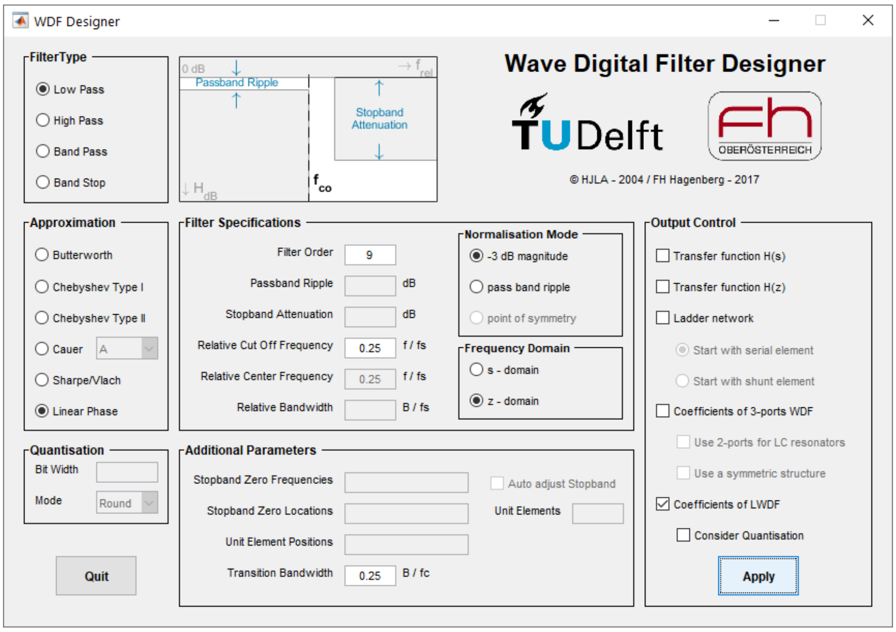


Figure 1: Settings for the design of a 9 th order half-band filter.

To start the LWDF Designer toolbox, run the WDF_GUI with MATLAB. Getting started, choose as FilterType LOW PASS, as Approximation LINEAR PHASE, as Filter Specifications FILTER ORDER 9, RELATIVE CUT OFF FREQUENCY 0.25 and as Additional Parameters the TRANSITION BANDWIDTH 0.25. Moreover, select Z-DOMAIN (very important!) and in the Output Control unit COEFFICIENTS OF LWDF as depicted in Figure 1 and press the Apply button. One obtains the magnitude response and group delay of the low pass and corresponding high pass filters, and the block diagram of the filter. The lower all pass filter is a simple delay line of order 8, the upper all pass comprises 4 second order and one first order sections referred to as adaptors, as depicted in Figures 4 and 5, respectively. The adaptor coefficients, referred to as reflector coefficients, can be found in the command line of MATLAB.

Figure 2 shows the general design of the LWDF toolbox as well as the implemented changes to integrate the new design method. The toolbox allows the user to design filters of the four characteristic magnitude behaviours: low-pass, high-pass, band-pass and band-stop. For each characteristic behaviour a number of approximation methods including Butterworth, Chebyshev, Cauer and Sharpe/Vlach can be selected. Depending on the approximation method different parameters like cut-off frequency, bandwidth, passband and stopband ripples and the zero locations of the filter transfer function have to be provided to further specify the filter behaviour. Unlike other toolboxes the design specifications have to be input in relative frequency values, which makes it

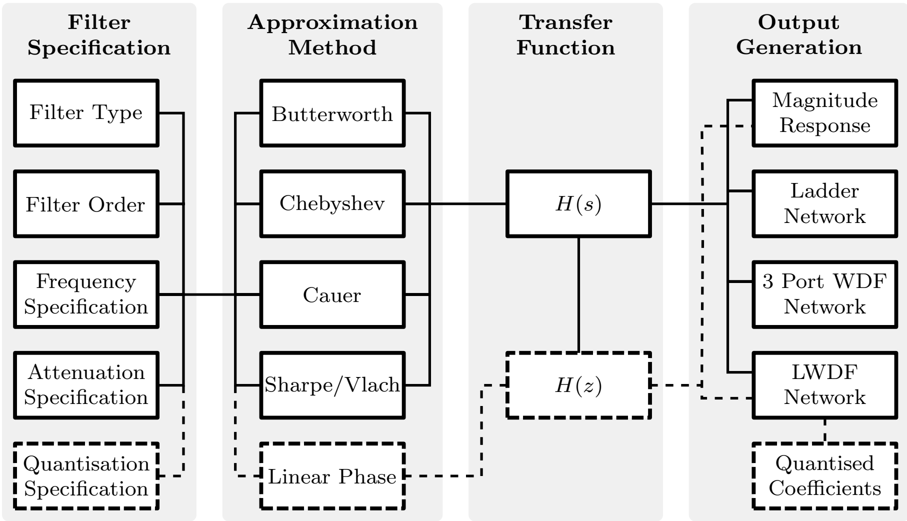


Figure 2: Current design flow (solid) and implemented changes (dashed)

easier to compare different filter designs and reuse existing filters for different application scenarios. The filter parameters can be translated into concrete values at the end of the design process though. The toolbox provides the possibility to design the filters in both the continuous-time and discrete-time domains. The transformation of the two domains is done using the bilinear transformation. The results of the design process include the general transfer functions in continuous-time and discrete-time domains as well as the actual Wave Digital Filter parameters for two and three port adaptors. For purely analogue applications a ladder network containing resistors, capacitors and inductances can also be created.

3 Software Architecture

The toolbox is implemented as a plug-in for Matlab, using the built in GUI framework. Therefore the software consists of two parts: a layout description and a complementing script file including all user interface related functionality and callback methods as well as a set of calculation methods with which the actual approximation methods are performed and results are calculated. In this Section the most commonly used data structures are explained in detail.

3.1 Data Structures

3.1.1 Transfer Function Data Structure

The interfaces for the different filter approximation functions vary in number and types of specification parameters due to their different methodical approaches. The return values of the approximation functions however share the same format, a data structure (`Hs`) representing the transfer function of the filter in the continuous-time domain $H(s)$. The following code examples show the resulting structures of a normalized 5^{th} order Butterworth low-pass filter at the cut-off frequency $\omega_{co} = 1$.

```
Hs =

  struct with fields

     poly_fs: 1
     poly_gs: [1 3.2361 5.2361 5.2361 3.2361 1.0000]
       ident: 'LP PROTOTYPE: 'butter',5,1'
    roots_fs: []
    roots_gs: [5x1 double]
```

The individual fields of the structure representing the transfer function are as follows:

- `poly_fs`: coefficients of the numerator polynomial $[b_n \, b_{n-1} \ldots b_0]$
- `poly_gs`: coefficients of the denominator polynomial $[a_m \, a_{m-1} \ldots a_1 \, 1]$
- `ident`: textual representation of the filter characteristics
- `roots_fs`: roots of the numerator polynomial
- `roots_gs`: roots of the denominator polynomial

This data structure is the outcome of every approximation algorithm and serves as the main exchange format between the approximation step and the subsequent Wave Digital Filter calculation algorithms. Another data structure (`Hz`) exists for representing the transfer function in discrete time domain $H(z)$, but it is rather used as a final result of the design process than as an exchange format.

3.1.2 LWDF Parameter Structure

A result of the whole design process is another data structure (`LWDF`) which contains the filter parameters for the Lattice Wave Digital Filter type as well as additional information for plotting the lattice structure. These parameters can be directly used as coefficients of the LWDF adaptor structures in hardware implementation.

```
LWDF =

  struct with fields:

    wdaCodes: [2x2 char]
       gamma: [2x2x2 double]
```

The fields of the structure contain data as follows:

- `wdaCodes`: encoded types of LWDF structure elements for top and bottom all pass branches

- `gamma`: γ-coefficients of each two-port adaptor in the top and bottom LWDF branches

The γ-coefficients are output in a 3-dimensional matrix with $2 \times N \times 2$ fields where the 1st dimension represents the maximum number of adaptors per all-pass section which is always 2, N represents the maximum number of all-pass sections in the top and bottom branches and the 3rd dimension represents the two branches of the LWDF structure and is always 2 as well. Not used coefficients are marked as NaN. Concerning above mentioned example, the corresponding gamma matrix looks as follows:

$$\gamma_{top} = \begin{bmatrix} 0.00 & -0.5279 \\ NaN & 0.00 \end{bmatrix}, \gamma_{bottom} = \begin{bmatrix} -0.1056 & NaN \\ 0.00 & NaN \end{bmatrix}. \tag{1}$$

3.2 Functions

The existing functions of the toolbox were changed as little as possible. Even though a lot of functionality of the original toolbox is already built into the newer versions of the Matlab framework – especially the standard filter design and transformation functions – the original ones were kept due to compatibility reasons.

3.2.1 LWDF Coefficient Calculation

The central functionality to obtain a Lattice Wave Digital Filter is implemented in the `Hs2LWDF` function. As a parameter it requires the data structure `Hs` containing the general filter transfer function and converts it to the LWDF parameter structure mentioned before. Therefore the roots of the denominator polynomial are used to calculate the coefficients for the two all-pass branches. This method aims to assign neighbouring roots of the polynomial to different branches of the LWDF and calculating the filter coefficients directly from the continuous time transfer function.

Due to the design process of the linear phase method, the structure of the linear phase filters does not comply with this approach because it designs dedicated all-pass filters for the upper and lower all-pass branches of the filter directly in the discrete-time domain. When combined with the above mentioned method this would lead to a much bigger resulting filter in terms of coefficients and thus multipliers. Hence the functionality has to be extended to support the conversion of transfer functions originating from the linear phase method.

3.2.2 LWDF Structural Plot

Another key functionality – for plotting the resulting LWDF structure – resides in the `ShowLWDF` function. It is used to visualize the generated lattice structure of two-port adaptors and delays. Among other parameters, this function requires the calculated `LWDF` structure and draws the structure in a Matlab plot window. The function is by design not able to display the special structure of the linear phase filters and therefore requires some adaptions as well.

3.2.3 Frequency Response Plot

To visualize the characteristic of the designed filter the toolbox offers a function (`PlotHz`) to plot the magnitude response of a given discrete time transfer function. While initially only the magnitude response was of relevance, with the implementation of the linear phase method the phase response and the group delay are relevant results of the design process as well.

4 User Interface

The new interface (see Figure 3) shows all available controls at all times, where controls are disabled if they are not needed with the current settings though. The redesigned layout now also works for different operating systems. To simplify the handling, not all optional parameters of the linear phase design functions are made available in the user interface.

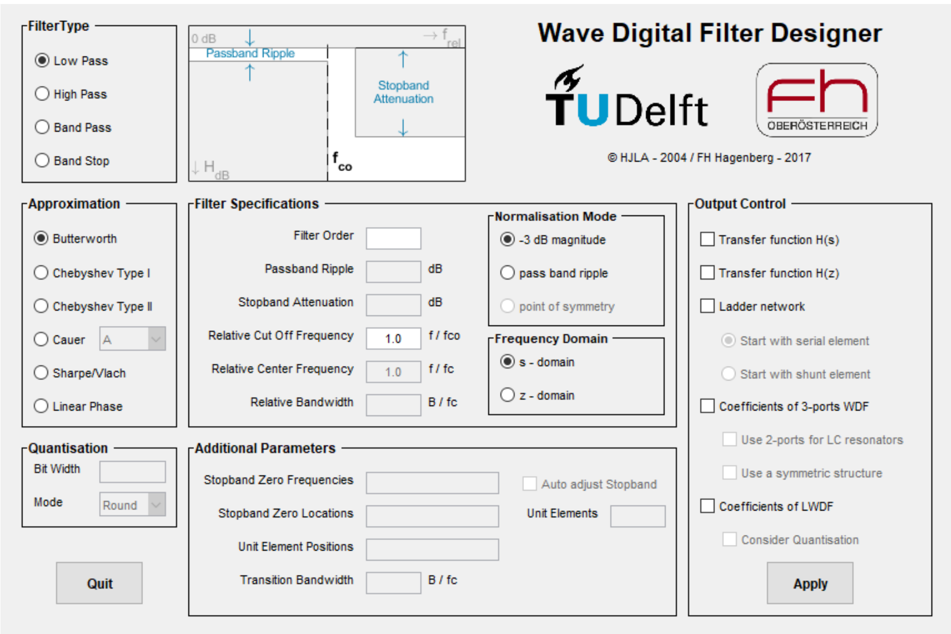


Figure 3: Toolbox user interface.

4.1 Filter Type

Selection of the four common filter characteristics. The plot area to the right of this box shows the selected filter characteristic along with a description of the key parameters to specify the filter.

4.2 Approximation

The supported filter approximation methods now include the *Linear Phase* option. The approximation method implicitly defines characteristic properties of the transfer function, such as the existence of passband ripples, the minimal stopband attenuation or the location of zeros in the transfer function. In the case of the linear phase design method it ensures a very low group delay variation in the passband region up to the cut-off frequency.

4.3 Filter Specifications

The filter specification and additional parameter sections contains all possible configuration options for the filter design. Depending on the selection of the first two options, the required filter parameters are enabled. Using the linear phase method, only requires some basic parameters like the filter order and the cut-off frequency, as well as a bandwidth for band-pass and band-stop filters. Designing low-pass or high-pass filters, an odd filter order is required, while at band-pass and band-stop designs, an even filter order is mandatory.

It is possible to specify all design methods either in continuous time (s-domain) or discrete time (z-domain). However, for the linear phase method, the z-domain is recommended due to the fact that the design process is performed in the z-domain. It has to be noted that the linear phase approximation method, unlike others, does not specify a fixed 3 dB corner frequency but rather a transition band, the actual 3 dB frequency can be anywhere within this region.

4.4 Quantisation

A new feature of this toolbox is the assessment of the effects of parameter quantisation. In practical applications the precision of the filter coefficients is limited due to the implementation efforts and costs, therefore it is necessary to know whether or not the filter performance meets the specifications under real life conditions. This option provides the possibility to evaluate the frequency response for a given quantisation bit width and a number of quantisation methods. For this option to be enabled, the *Consider Quantisation* checkbox in the *Output Control* section needs to be activated.

4.5 Output Control

In the *Output Control* section, different output plots and data structures can be enabled. From plots of the magnitude response for both continuous and discrete time domains up to structural graphics and coefficient values of the LWDF filter implementation, a large number of documentary or implementation details can be generated.

5 Toolbox Functionality

To incorporate the new design process into the toolbox, changes were made to several functionalities, especially for handling the intermediate data structures holding the transfer function. Additionally, adaptions of the structural plots of the LWDFs and transfer function plots were introduced. The most apparent changes were made to the user interface, incorporating the new functionality and reorganizing the controls to get a more intuitive look and feel.

5.1 Linear Phase Approximation Functions

The actual implementation of the linear phase design method was done in separate Matlab functions to be usable also without the toolbox. They incorporate the whole set of mandatory and optional parameters to fine-tune the approximation method. This set of functions contains the two approximation methods, Galerkin approach and Collocation method as well as the actual design functions for retrieving filters of lowpass and bandpass characteristic. Other characteristics were not implemented due to the fact that the inverse characteristic can be achieved by subtracting the two branches from each other instead of adding them, which is a well-known advantage of LWDFs.

The functions to access the linear phase design method require only the filter order and the transition bandwidth for the lowpass, and the bandwidth and transition bandwidths for the bandpass filter. The implementation of the bandpass design function makes use of the frequency transformations. There are optional parameters to control the approximation method as well as further optimisation algorithms. The functions return the transfer functions in both continuous-time and discrete-time domain and can be accessed as follows:

```
[Hs, Hz] = Hs_linear(filterOrder, fp_rel, transBw_rel, varargin)
```

```
[Hs, Hz] = Hs_linearBP(filterOrder, bw_rel, transBw_rel, varargin)
```

The first three parameters of these functions are mandatory. The additional optional parameters have no fixed order and have to be entered as key-value pair. The frequencies and bandwidths are normalized with respect to the sampling frequency and have to be inside the interval [0, 0.5[.

- `filterOrder`: filter order of the linear phase filter; This parameter actually represents the number of multipliers needed, which does not necessarily coincide with the order of the transfer function.
- `fp_rel`: normalized passband frequency of lowpass filter
- `bw_rel`: normalized passband bandwidth for symmetrical bandpass filter with respect to frequency $f = \frac{f_s}{4}$
- `transBw_rel`: normalized transition bandwidth

- `varargin`: optional argument specifier, arguments can be one or more of the following key-value pairs
  - `'DesignMethod' : ('galerkin' | 'collocation')`
    underlying approximation method (default `'collocation'`)
  - `'OrderReduction' : (0 | 1)`
    attempt to reduce the filter order (default 0)
  - `'IterativeOptimisation' : (0 | 1)`
    perform an iterative optimization (default 0)

Through the filter transformation functions in theory every other kind of filter characteristic is realizable. Due to the efficient implementation structure of the linear phase design, arbitrary transformations – especially for bandpass and bandstop characteristics – are not possible, because they would lead to transfer functions with which this particular structure is not realizable anymore. Arbitrary transformations would lead to linear phase designs with a significantly higher implementation effort. It is possible though to design a bandpass characteristic which is symmetric with respect to frequency $f = \frac{f_s}{4}$. Therefore this design function only needs two specification parameters.

5.2 LWDF Creation Functions

As shown in Figure 2, the data exchange format of the toolbox equals the continuous-time domain transfer function. Hence retrieving the LWDF structure and coefficients require some additional transformation functions. Since the linear phase design function yields the transfer function in the discrete-time domain, the subsequent LWDF transformation functions were upgraded so that they are able to calculate the LWDF coefficients. Similarly, the function for plotting the LWDF adaptor structure was upgraded to be able to plot structures of linear phase filters. Usually the adaptors and delay elements are evenly distributed among the upper and lower branches. In a linear phase design, the adaptors are all located in the upper branch, leaving only delay elements in the lower branch.

5.3 Coefficient Quantisation

To be able to analyse the quantisation effects on the magnitude and phase response, a possibility to quantise the γ-coefficients of the LWDF was implemented as well. This feature can be accessed through the user interface of the toolbox and requires the bit-width and the quantisation method as parameters. The quantisation method can either be a simple rounding to the nearest discrete value (*round*), always rounding up (*ceil*) or down (*floor*) or rounding towards zero (*trunc*). When selected, the quantisation functionality yields a separate data structure containing the quantised filter coefficients as well as plots comparing the ideal and quantised frequency responses.

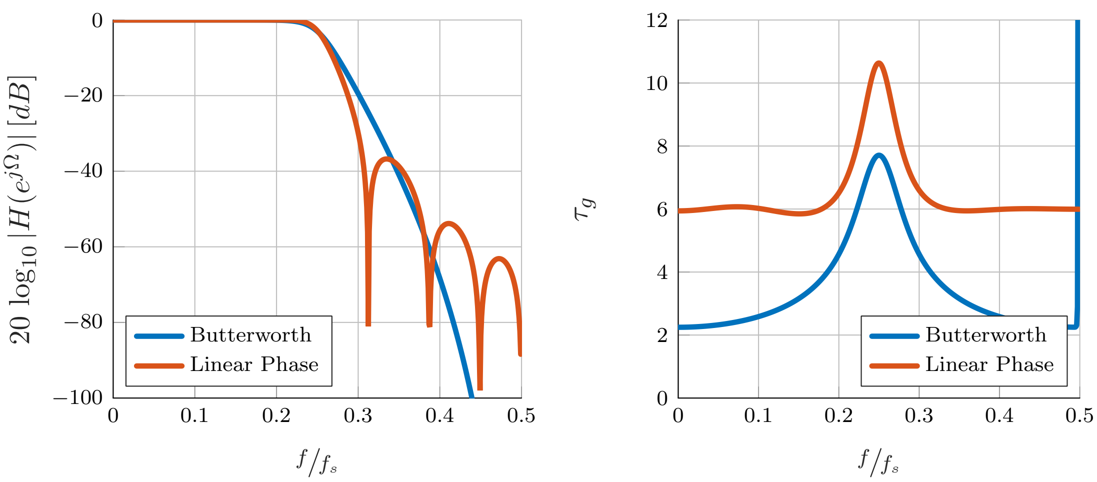
6 Example Filter Designs

This section shows some example results of the implementation of the linear phase design method. As there are numerous design specification variations, not all of these are addressed and compared in detail to reference implementations. For comparison of the magnitude and phase responses a Butterworth LWDF implementation is chosen. Addressing the mismatch between filter order and order of the transfer function between a linear phase and a Butterworth implementation, the Butterworth reference implementation shall have the same number of multipliers, which better reflects the practical advantages of the linear phase design method.

6.1 Lowpass Design

The first example design, depicted in Figure 4, shows a 7^{th} order lowpass filter. Both filters are designed using the same passband frequency $\frac{f}{f_s} = 0.25$. The linear phase design parameters also include a transition bandwidth of $\frac{B_t}{f_s} = 0.05$ and the Collocation approximation method.

Regarding the magnitude response, the linear phase design initially exhibits a steeper transition to the stopband, but a lower maximum attenuation than the reference design. Depending on the implementation bit-width and the quantisation noise floor, this might have to be considered for the desired application. The group delay variation on shows the strengths of this approximation method, and is approximately constant for the pass band at $\tau_g \approx 6$.



(a) Magnitude response

(b) Group delay

Figure 4: Frequency responses for lowpass designs of Butterworth reference and linear phase design of 7^{th} order

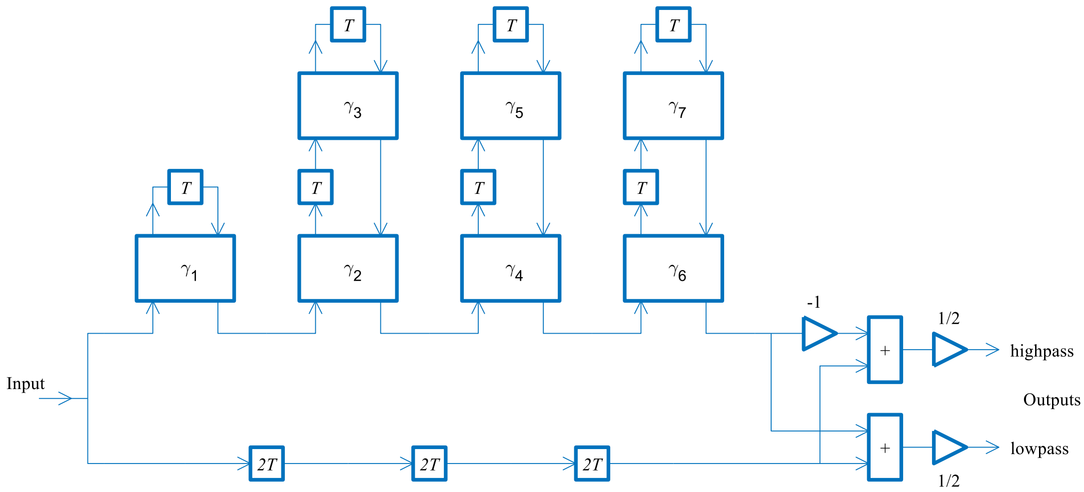


Figure 5: LWDF topology for lowpass filter, using the linear phase design.

The group delay baseline is noticeably higher than the minimal group delay of the reference filter, which is characteristic to this design method. The group delay variation is very low in the passband and considerably lower than the reference design near the transition band. Hence there is a trade off between passband attenuation and group delay variation, which has to be considered.

The LWDF topology for the example lowpass filter, suing the linear phase design can be seen in Figure 5. The LWDF topology can be characterised by the two branches of allpass filters, with sequences of adaptors and delay elements, which are combined at the end by either subtraction or addition. Considering the linear phase design, the lower branch consists only of delay elements, which leads to an ideal constant group delay. The upper branch contains the actual allpass filter cascade, which creates the desired phase response. In this case it consists of 7 adaptors, resulting in a 7^{th} order filter. Each of these adaptors has a single coefficient and requires exactly one multiplier. Even though this is a lowpass design, it can be seen that the inverse highpass characteristic can be easily derived from this topology by subtracting the upper branch from the lower one.

Figure 6 shows the weighting function for the evaluated collocation points. Every sampling point represents one frequency at which the desired phase response is optimised. The magnitudes of the sampling points indicate how severely the desired phase response is enforced at this frequency. The filter coefficients, which resemble the desired phase response as best as possible, are obtained using an optimisation algorithm with least Euclidean norm.

Another option to enforce the weight in a specific frequency region is to choose a different spacing for the sampling points. In Figure 6 it can be seen, that the phase response in

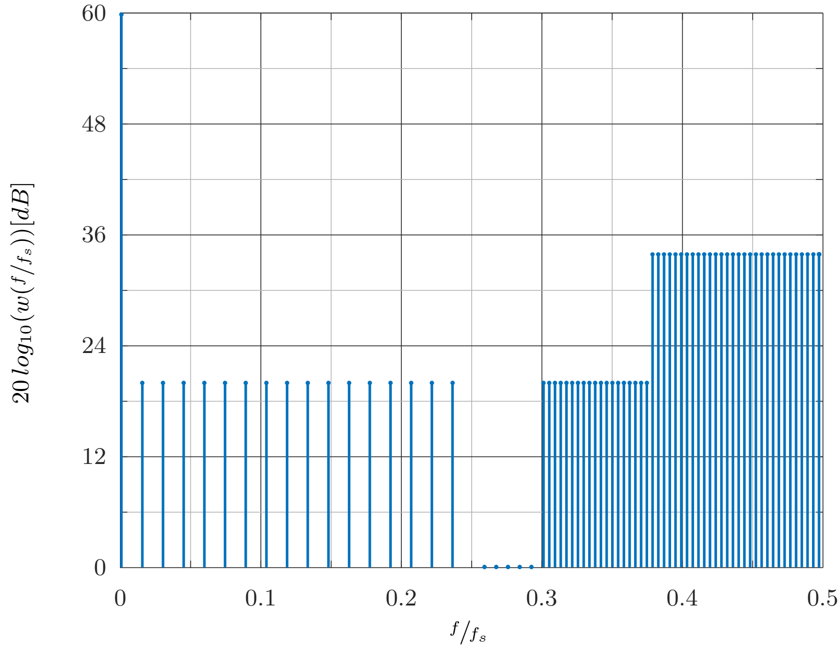


Figure 6: Weighting function for the example linear phase design of a 7^{th} order lowpass filter

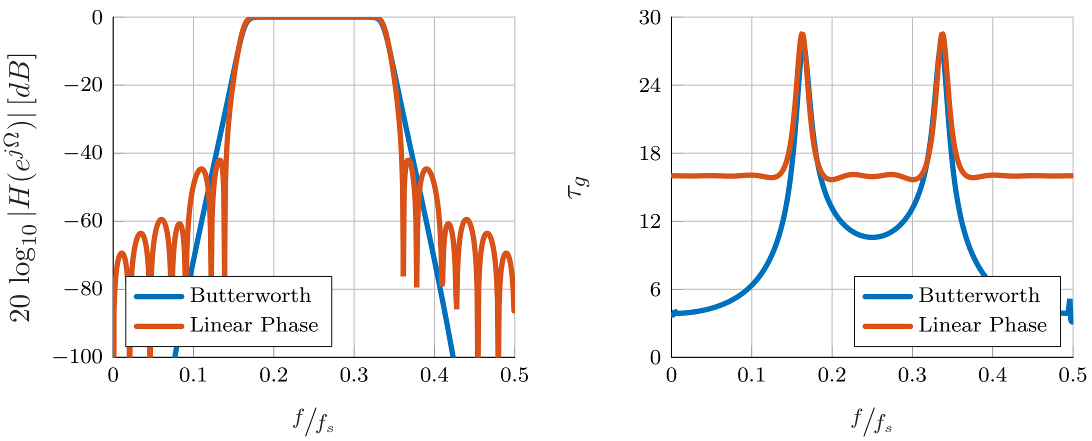
the transition band is not enforced at all. The stopband on the other hand is enforced strongly by using a dense spacing and high magnitudes. Another sampling point with a high weight is at DC frequency $f = 0$.

6.2 Bandpass Design

For the bandpass example a 10^{th} order linear phase and Butterworth filters were chosen. The bandwidth for both filters was set at $\frac{B}{f_s} = 0.2$, the remaining parameters for the linear phase design are the same as in the lowpass example.

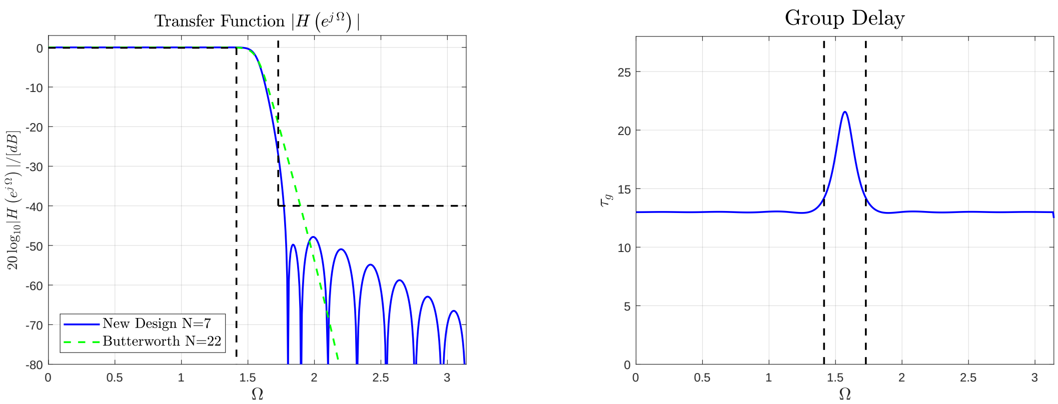
The magnitude response of Figure 7 shows similar relations between the reference and linear phase design. In the passband region and near the cut-off frequencies the magnitude responses are nearly the same. The reference design however exhibits a higher total stopband attenuation.

The group delay plot again shows a significantly reduced group delay variation over the whole frequency range. From the lower to the upper corner frequency the group delay variation is almost constant at $\tau_g \approx 16$.



(a) Magnitude response (b) Group delay

Figure 7: Frequency responses for bandpass designs of Butterworth reference and linear phase design of 10^{th} order



(a) Magnitude response. (b) Group delay.

Figure 8: Half-band filter design: magnitude response and group delay, $N = 15$.

6.3 Half-band filter design

The Fig. 8 depicts the magnitude response and group delay of half-band filter of order $N = 15$. The group delay is nearly constant within the passband. Due to symmetry, the number of non-zero coefficients is only $K = 8$. The exact symmetry is achieved by a symmetric choice of the weights as depicted in Fig. 9. In the transition regime, the weights are intentionally small, whereas in the pass- and stop-bands weights of 40 dB were chosen. To emphasis the phases at DC and at the Nyquist rate, weights of 80 dB were chosen for these frequencies.

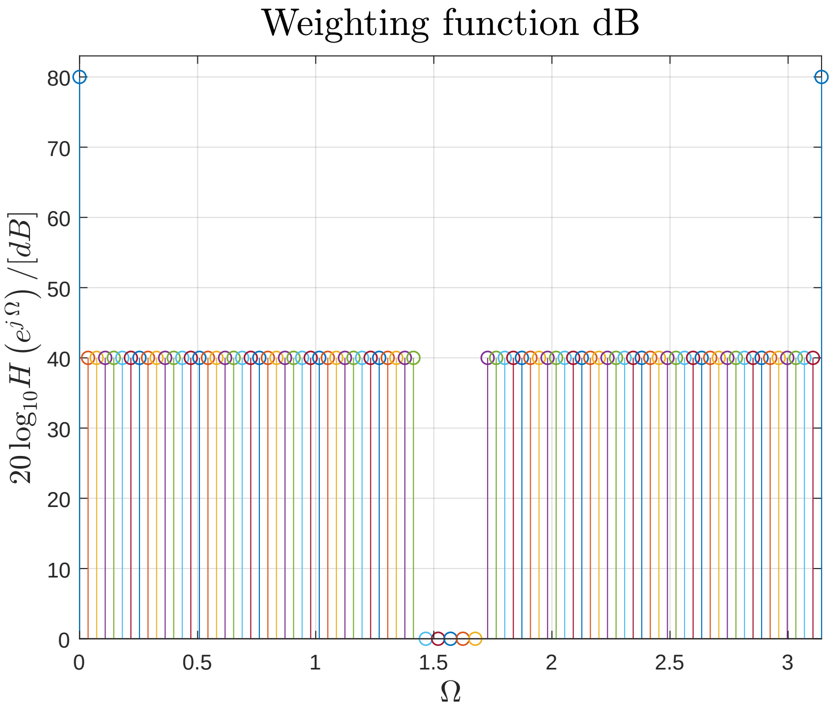


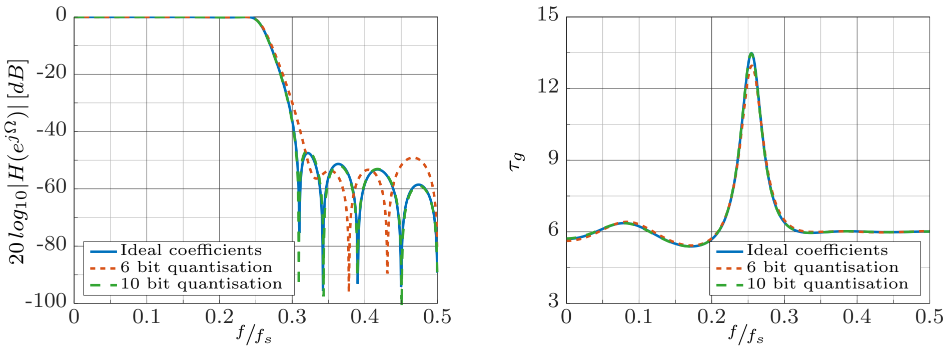
Figure 9: Weights for the half-band filter design in dB.

6.4 Quantisation Examples

To analyse the quantisation effects, a 7^{th} order lowpass filter with a passband frequency ($\frac{f}{f_s} = 0.25$) and a transition bandwidth ($\frac{B_t}{f_s} = 0.05$) was designed. For this design the Galerkin approximation method was used. The quantisation effects are shown for quantised coefficients at two different bit-widths and compared to the ideal coefficients in Figure 10. The quantisation methods were chosen to be *round* and *ceil* for word-lengths of 6 and 10 bit respectively.

The quantisation effects are clearly visible in the magnitude response of Figure 10, especially with small bit-widths. Choosing a slightly greater bit-width and varying the quantisation mode lead to a very good resemblance of the ideal magnitude response, as can be seen with the 10 bit quantisation. The effects on the group delay however are almost not noticeable. Expectedly, the 10 bit quantisation exhibits no difference at all. The 6 bit variation shows small deviations from the ideal group response, but the differences – especially in the transition band – are even beneficial.

Compared to the design in Figure 4, the Galerkin method achieves a noticeably higher stopband attenuation, however this is achieved at the cost of a higher group delay variation.



(a) Magnitude response (b) Group delay

Figure 10: Frequency responses for linear phase lowpass designs of 7^{th} order, with different quantisation bit-widths.

References

[1] Huib J. Lincklaen Arriëns. *(L)WDF Toolbox*. Aug. 2006. URL: `http://latech.nl/mtbx`

[2] C. Zeintl and H. G. Brachtendorf. *Linear phase design of lattice wave digital filters*, 2018 28th International Conference Radioelektronika (RADIOELEKTRONIKA), Prague, 2018