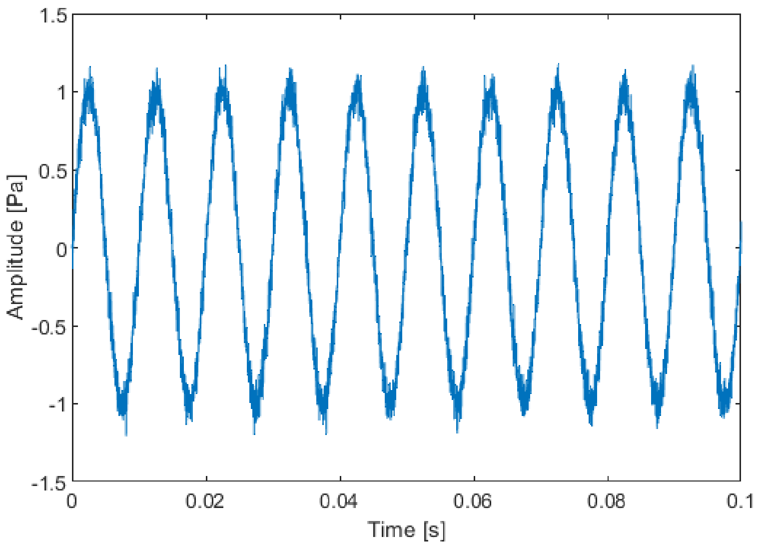
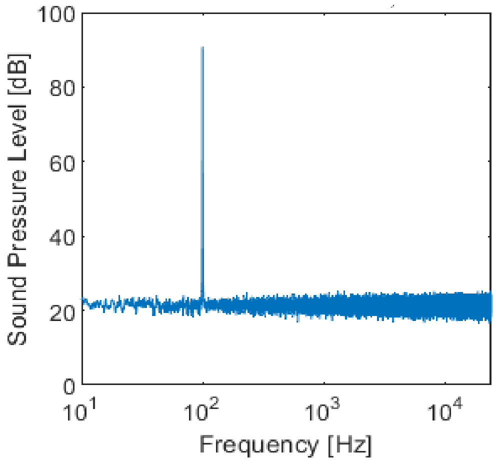
1/3 octave analogue filters in accordance with digital technology – a discontinued model?

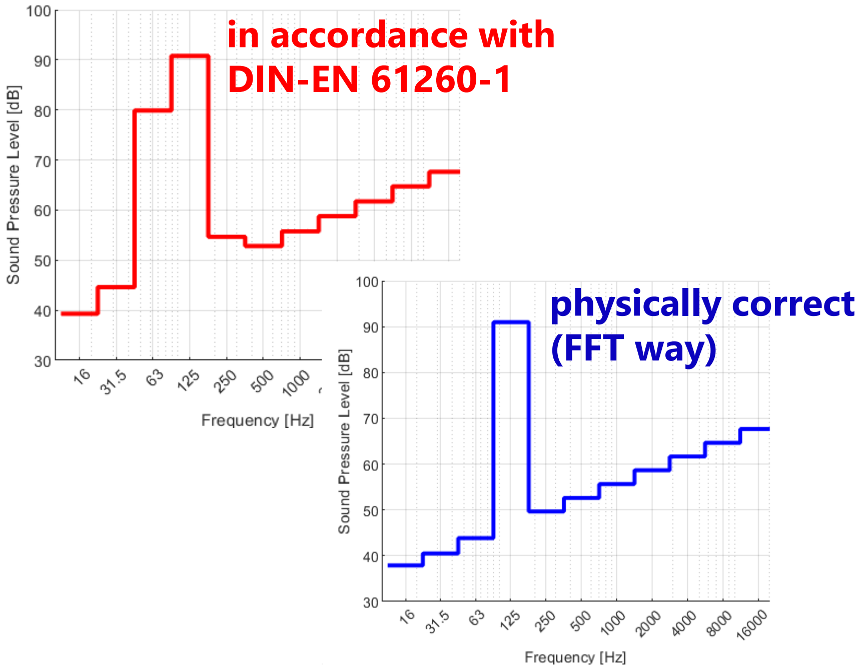
- Implementation using the MATLAB Signal Processing Toolbox
- Verification and evaluation of octave spectra

Introduction: two methods to calculate octave spectra

- Sine-tone (100 Hz) with white noise (small amplitude)







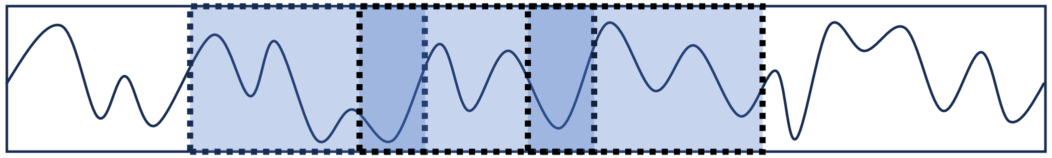
- **Current state** of DIN-EN 61260-1: analogue filters are imitated by digital filters
- Up-to-date method with FFT-calculations is not mentioned explicitly

- Topic: **implementation** of both methods and **systematic comparison**
  - Parseval's theorem, evaluation of octave spectra
  - Discussion for low frequencies (DIN 45680)

Method with FFT-calculations

Implementation using the MATLAB Signal Processing Toolbox

- **Amplitude spectrum** with spectrogram(…) – command
  - Window functions
  - Blockwise average (squared values)



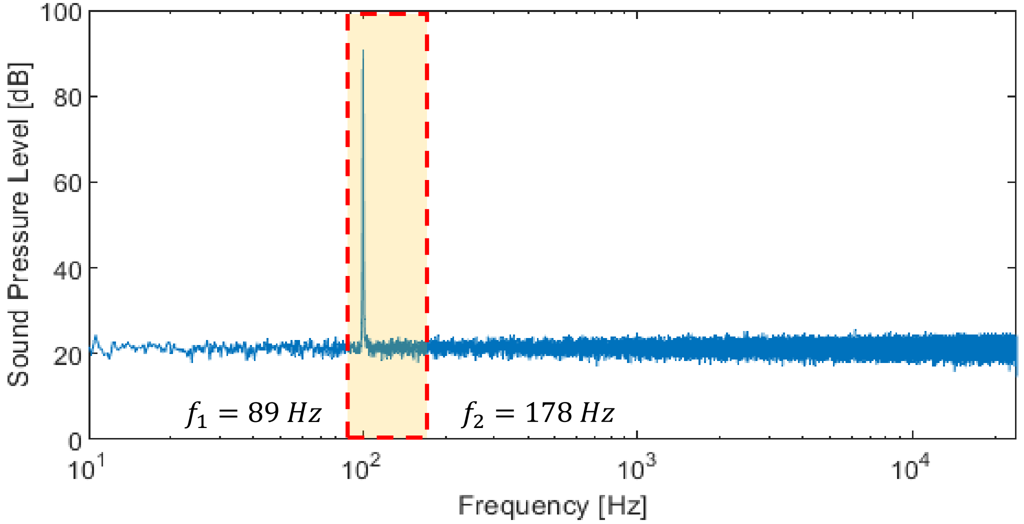
e.g., subdivision of a signal into 3 blocks

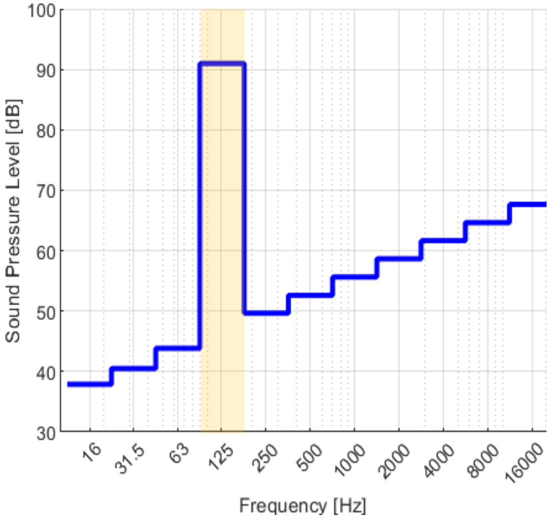
- **Octave**: sum of N spectral components A_i within the limit frequencies $f_{1,2}$
  - Energy-correction ε for non-normalized window functions

- With enough block-averaging: OAL (frequency domain) ≙ RMS (time domain)

Parseval

$$OAL = \sqrt{\frac{1}{\varepsilon}\sum_{i=1}^{N} A_i}$$





OAL of an octave band (midband frequency $f_m = 125\ Hz$)

	ε
Rectangle	**1**
Hanning	**1.5**

Energy-correction for window functions

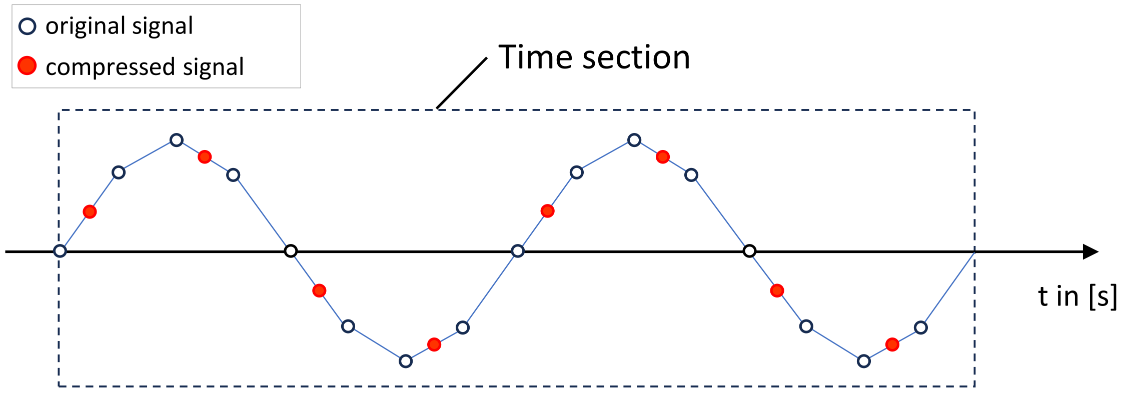
Imitation of analogue filters with digital filters

Implementation using the MATLAB Signal Processing Toolbox

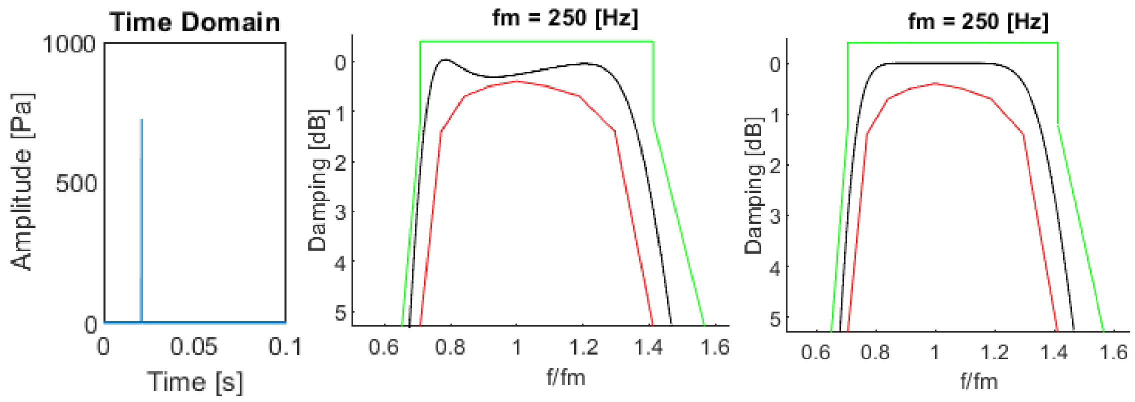
- Bandpass filters (*Butterworth*) are applied with a filter passage range $f_{1,2}$
- **Octaves** follow from RMS over the complete time span

- Compliance with DIN-EN 61260-1 is checked with damping curves
- Impulse response gets „**unstable**" with increasing filter order and decreasing bandwidth

- Countermeasure: **resampling** – sampling rate is reduced by compressing time data
  - A lowpass filter is necessary to prevent aliasing



Reducing the sampling rate of existing time data by a factor 2

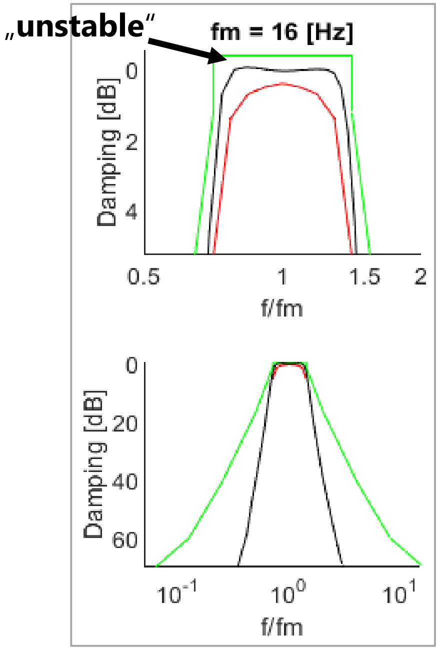


Damping curves – impulse response before and after one resampling, filter order 4

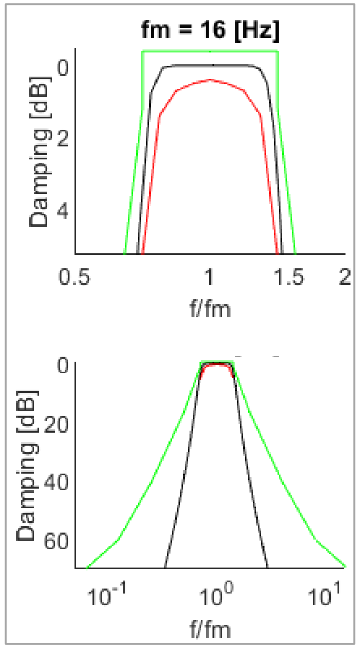
Imitation of analogue filters with digital filters (cont.)

Implementation using the MATLAB Signal Processing Toolbox

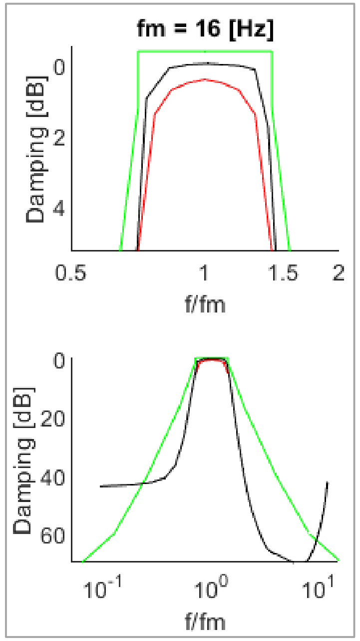
- A „stable“ damping curve requires:
  - adaptation of sampling on bandwidth and filter order
  - a certain number of remaining samples N within the blocklength



d = 64, N = 512



d = 128, N = 512



d = 128, N = 256

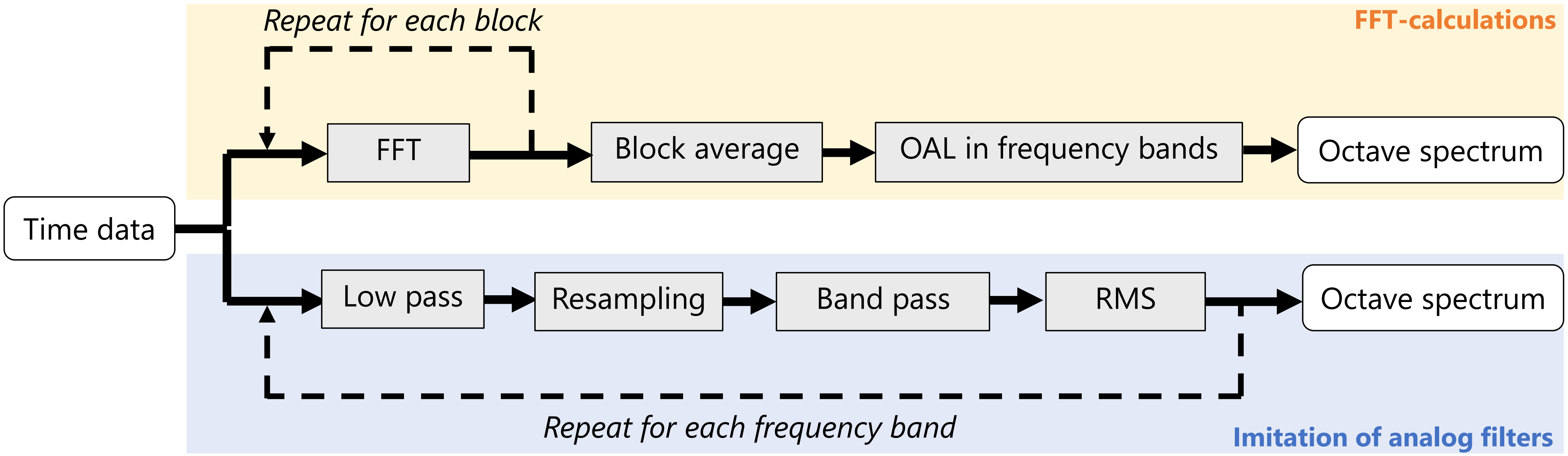
- N = 512 after resampling means 65536 samples in the original signal

Octave band	$f_m\ [Hz]$	16	31.5	63	125	250	500	1000	2000	4000	8000	16000
Subdivisions	d	128	64	32	16	8	4	2	1	1	1	1

Chosen number of subdivisions d of the sampling rate → „stable“ impulse response for filters of 6th order

Overview

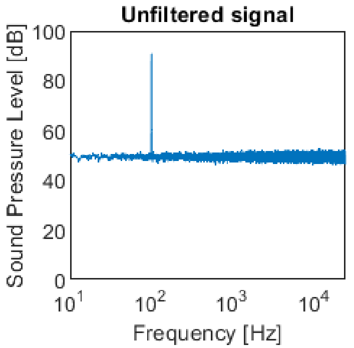
Implementation using the MATLAB Signal Processing Toolbox



Parseval‘s theorem with changing signal-frequency

Verification and evaluation of octave spectra

- **FFT-method**: error is independent of the signal-frequency
  - Parseval‘s theorem is satisfied with the third decimal place in dB
- **Calculation with filters**: error depends on signal and filter order
  - Parseval‘s theorem is satisfied with the first decimal place in dB



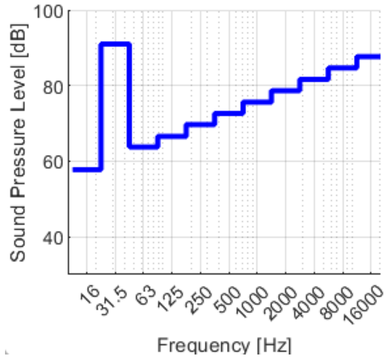
$$\text{Error [dB]} = \text{OAL [dB]} - \text{RMS [dB]}$$

Sum of all octaves → OAL

RMS in the time domain → RMS

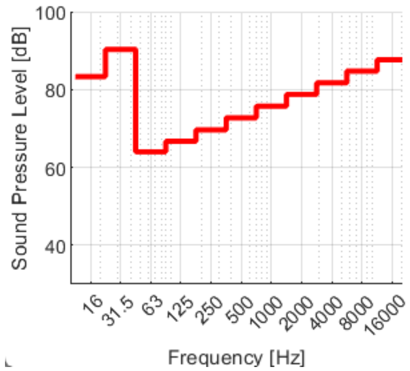
FFT-method

Signal type	SN-ratio [dB]	RMS [dB]	OAL [dB]	**Error [dB]**
Noise	-	90.7	90.7	**0.00**
24 Hz sine	40	99.3	99.3	**0.00**
24 Hz sine	50	93.8	93.8	**0.00**
31.5 Hz sine	40	99.3	99.3	**0.00**
31.5 Hz sine	50	93.8	93.8	**0.00**



Filter 4. order

Signal type	SR-ratio [dB]	RMS [dB]	OAL [dB]	**Error [dB]**
Noise	-	90.7	90.7	**0.05**
24 Hz sine	40	99.3	99.4	**0.06**
24 Hz sine	50	93.8	93.9	**0.09**
31.5 Hz sine	40	99.3	99.4	**0.04**
31.5 Hz sine	50	93.8	93.9	**0.02**



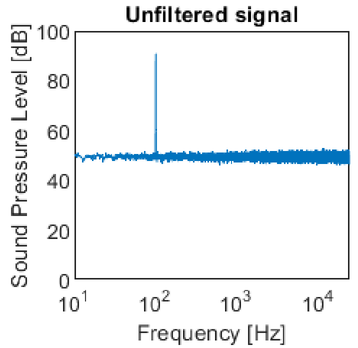
Parseval's theorem with changing signal-frequency (cont.)

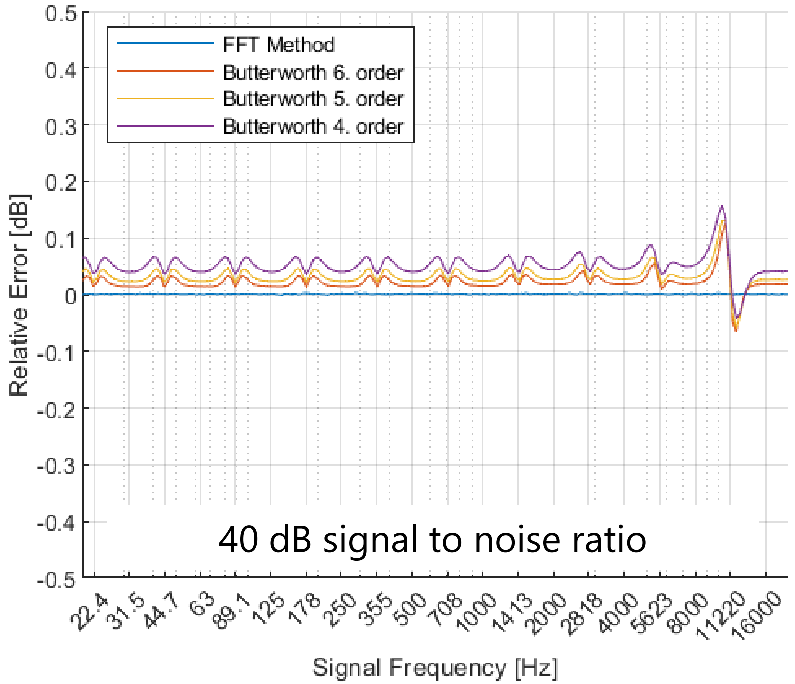
Verification and evaluation of octave spectra

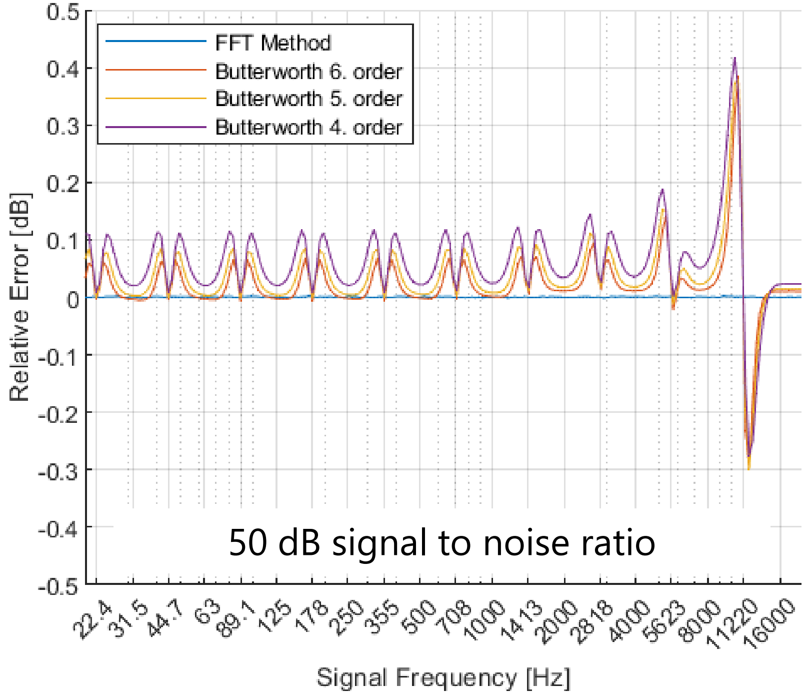
- **FFT-method**: error is independent of the signal-frequency
  - Parseval's theorem is satisfied with the third decimal place in dB
- **Calculation with filters**: error depends on signal and filter order
  - Parseval's theorem is satisfied with the first decimal place in dB

$$\mathrm{Error}\ [\mathrm{dB}] = \mathrm{OAL}\ [\mathrm{dB}] - \mathrm{RMS}\ [\mathrm{dB}]$$

  - Higher precision if signal-frequency ≈ midband frequency f_m (z.B. 31.5 Hz)



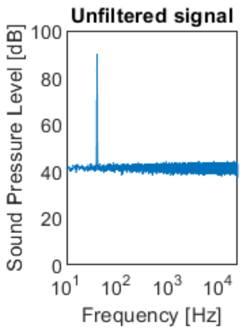


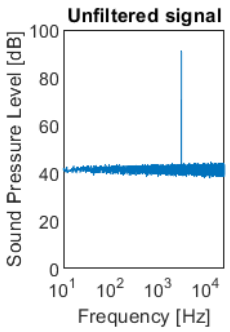


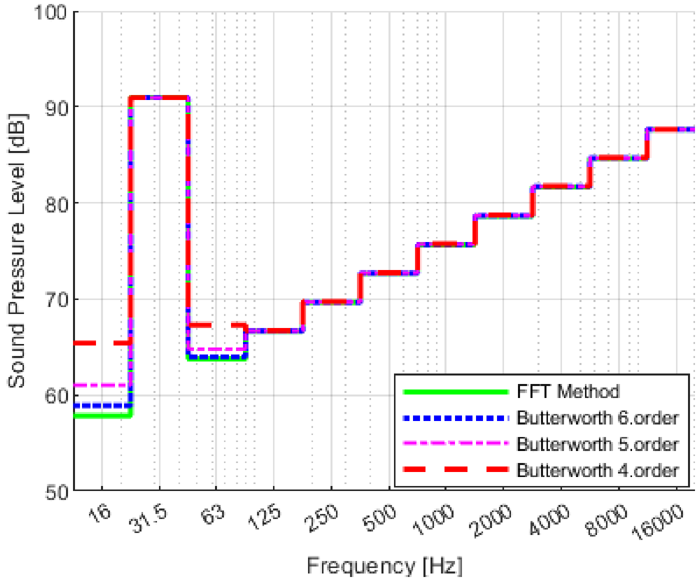
Octave spectra in the usual frequency range

Verification and evaluation of octave spectra

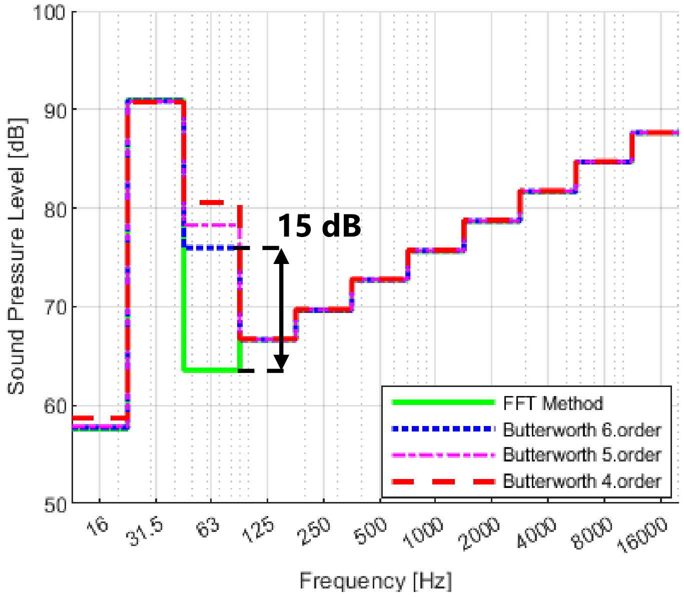
- **FFT-method**: sine-tone has an impact on its corresponding octave band
  - Leakage effect can have a minor impact on adjacent octaves
- **Calculation with filters**: sine-tone has an influence on several octaves
  - High discrepancy if sine-frequency is located between two octaves
  - Higher filter order leads slightly reduces discrepancy



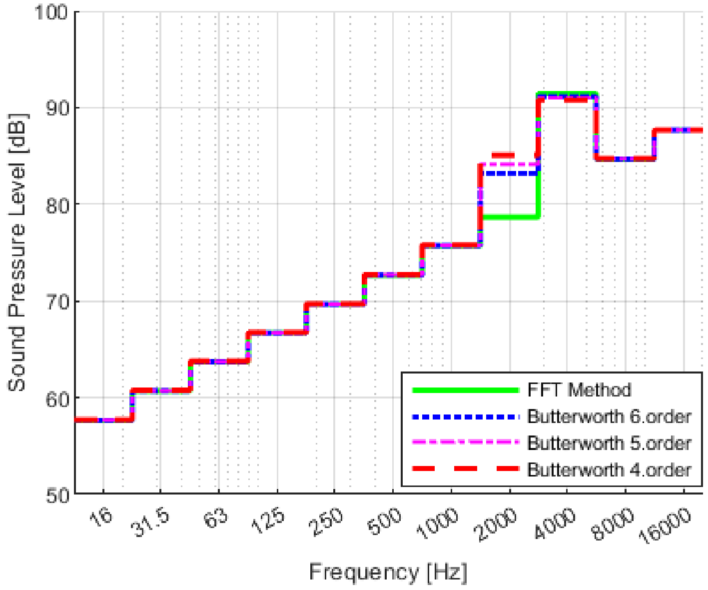




Signal-frequency = 31.5 Hz



Signal-frequency = 40 Hz



Signal-frequency = 3000 Hz

Octave spectra with low frequencies

Verification and evaluation of octave spectra

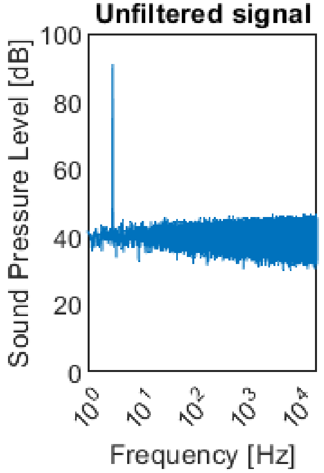
Settings

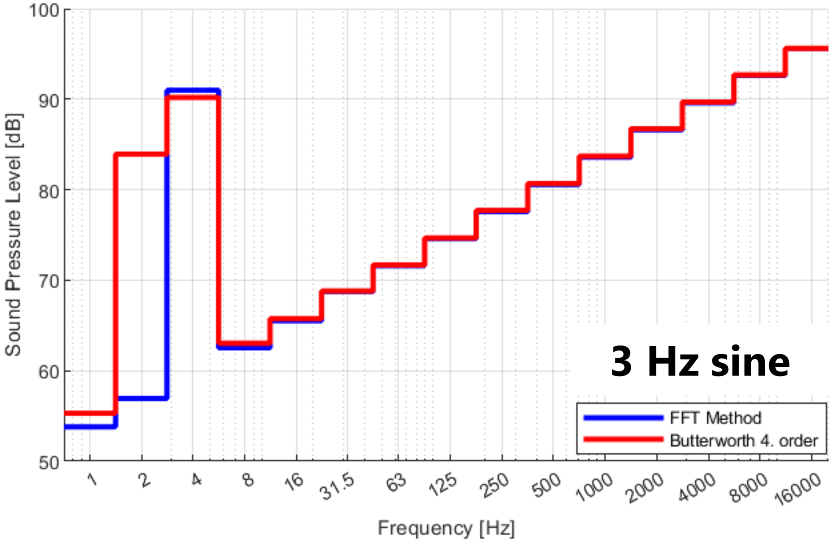
- 1024 necessary subdivisions of the sampling rate for a „stable“ Impulse response at 1 Hz octave
- Number of samples after resampling N = 2048 (→ 2^{21} in the original signal)

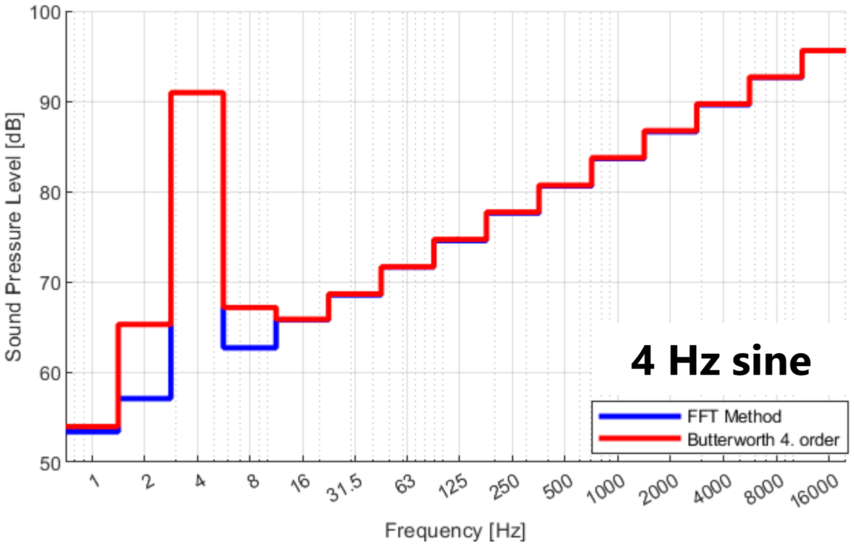
Octave band	$f_m\ [Hz]$	1	2	4	8
Subdivisions	d	1024	512	256	128

d = 1024 → new sampling rate of 46.9 Hz instead of 48000 Hz

- Extension for **1/3 octaves** is possible in analogy to this







Signal type	FFT method [dB]	Filter 4. Ord. [dB]
Noise	0.00	0.05
3 Hz sine	0.00	0.06
4 Hz sine	0.00	0.05

Error of OAL in dB

Results

- **FFT-calculations**:
  - Physically correct – Parseval‘s theorem is satisfied with the third decimal place in dB
  - Full octave spectrum with infrasound range can be easily calculated

- **Imitation of analogue filters**:
  - Precision depends on the signal – Parseval‘s theorem is satisfied with the first decimal place in dB
  - Signal-frequency between two octaves falsifies adjacent octave bands
  - Long recording times and multiple resampling are necessary to depict octave spectrum with infrasound range

- **The FFT-method satisfies the specifications of DIN EN 61260-1. This raises the question whether it should be explicitly mentioned in the standard**
- **Filtering in the time domain requires an adjusted sampling rate to the desired filter order while considering the frequency span**
- **This investigation has shown that the imitation of analog filters requires a higher effort in terms of implementation in comparison to the FFT method**
- **The original aim of DIN EN 61260-1 was to allow old hand-held sound level meters with analogue technology to continue to be used. Doesn't the standard now leave too much room for frequencies between two octaves in the 21st century?**

Thanks to Prof. J. Becker-Schweitzer for his background information of DIN 61260-1.