Noisy Audio Signal Filtering

Description

In this example, we analyze an audio signal which is corrupted by noise. The audio signal is mainly a speech signal which is corrupted by three different types of noise: an additive white Gaussian noise (AWGN), a wideband noise and a street background noise. The signal is analyzed by the frequency domain, in order to investigate its content. Based on the original uncorrupted signal, the filtering requirements are determined to process any noisy signal. All processing, including frequency domain analysis, filter design, and filtering, is performed by using MatDeck software with appropriate Toolkits.

Input signal

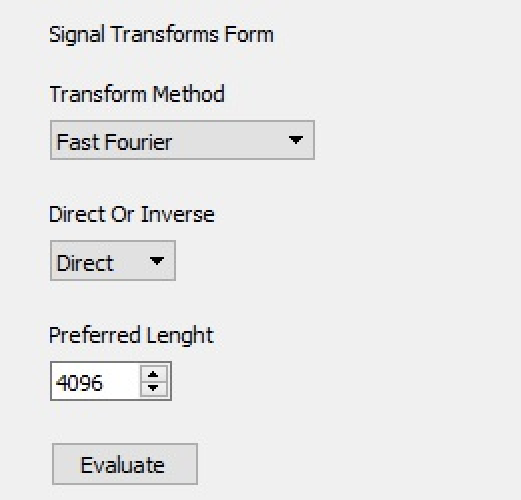
The input signal consists of several .wav files sampled at 8000Hz, with 16 bits per sample. The .wav files contain recordings of a speech signal, original, as well as, recordings of a speech signal corrupted by three different types of noises: AWGN, wideband noise, and street noise. The first signal is stored in the vector h1 contains a original speech signal which has a length of 4s, and the second signal stored in h1n is the same speech signal corrupted with AWGN.

```
props := wave properties("h_orig.wav")
Fs := props[0]      Hz, sampling frequency
h1 := wave read("h_orig.wav")
h1n := wave read("h_noise.wav")
```

Analysis in the frequency domain

The frequency domain analysis is performed by using FFT. For convenient processing, we use Signal Transform Form. We generate two forms for two input signals.

```
f1 := sigtransform(0 , "FormST12")      Create form
H1 := sigtransresult(f1 , h1)      Determine FFT
```



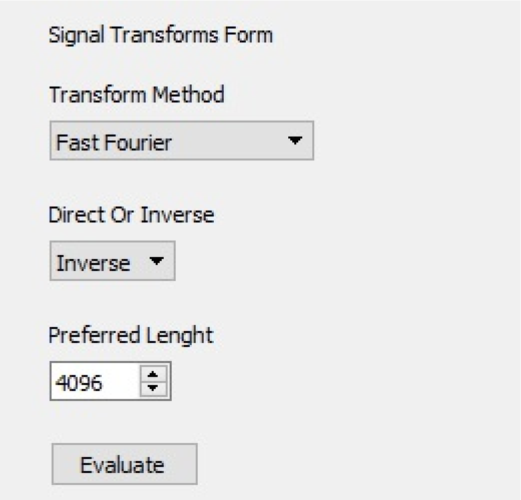
Signal Transforms Form

Transform Method: Fast Fourier

Direct Or Inverse: Direct

Preferred Lenght: 4096

```
f2 := sigtransform(0 , "FormST22")      Create form
H1N := sigtransresult(f2 , h1n)      Determine FFT
```



Signal Transforms Form

Transform Method: Fast Fourier

Direct Or Inverse: Inverse

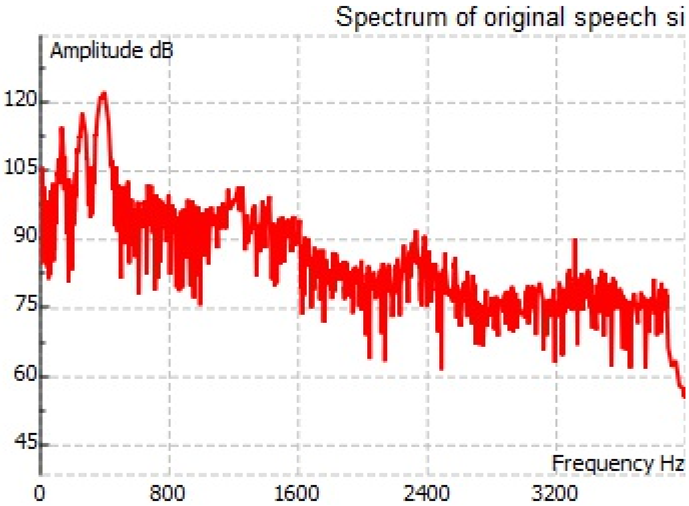
Preferred Lenght: 4096

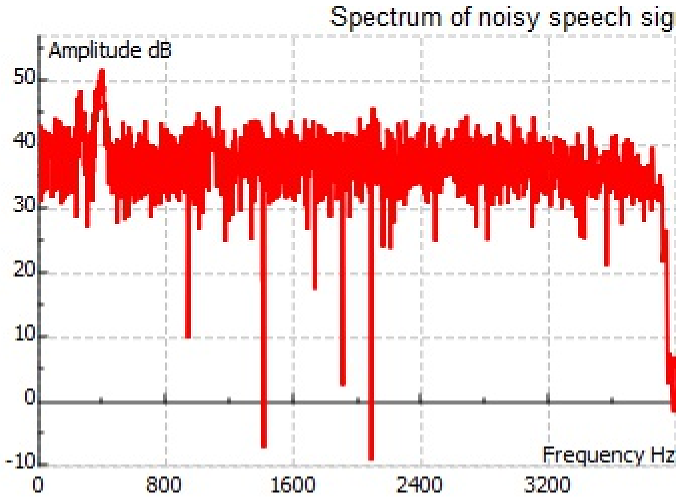
```
Fr := ynodes(x , 0 , Fs/2 , size(H1)/2)      Frequency axis for spectrum plot
```

Next, we plot both spectra graphically in a dB scale.

$$Sph1 := \text{join mat cols}\left(Fr\ ,\ 20\ \log10\left(\left|\text{subset}\left(H1\ ,0\ ,0\ ,\text{size}(H1)/2-1\ ,0\right)\right|\right)\right)$$

$$Sph2 := \text{join mat cols}\left(Fr\ ,\ 20\ \log10\left(\left|\text{subset}\left(H1N\ ,0\ ,0\ ,\text{size}(H1N)/2-1\ ,0\right)\right|\right)\right)$$





Filter Design and Data Filtering

By comparing the spectrum of the original speech signal on the left graph, and speech of the speech signal corrupted with noise, one can see that speech signal occupies the spectrum up to 2000Hz. The noise level is 40 dB above signal in a frequency band above 2000Hz. The noise spectrum is approximately flat which corresponds to the spectrum of the AWGN. The signal below 2000Hz should be preserved, therefore the distortions are restricted to 0.5dB. For speech signal, it is recommended to use a linear phase filter. FIR filters can be designed with linear phase property. Therefore, a linear phase low-pass FIR filter will be designed with following parameters>: the number of bands is two (passband and stopband), passband edge at 2000Hz, stopband edge at 2400,the passband ripple is 0.5dB, and the stopband attenuation is 60dB. In order to use the correct order of the FIR filter, firoptord is used to estimate it.

$numbands := 2$

$fp := 2000$ Hz, passband edge

$fs := 2400$ Hz, stopband edge

$Ap := 0.5$ dB, passband ripple

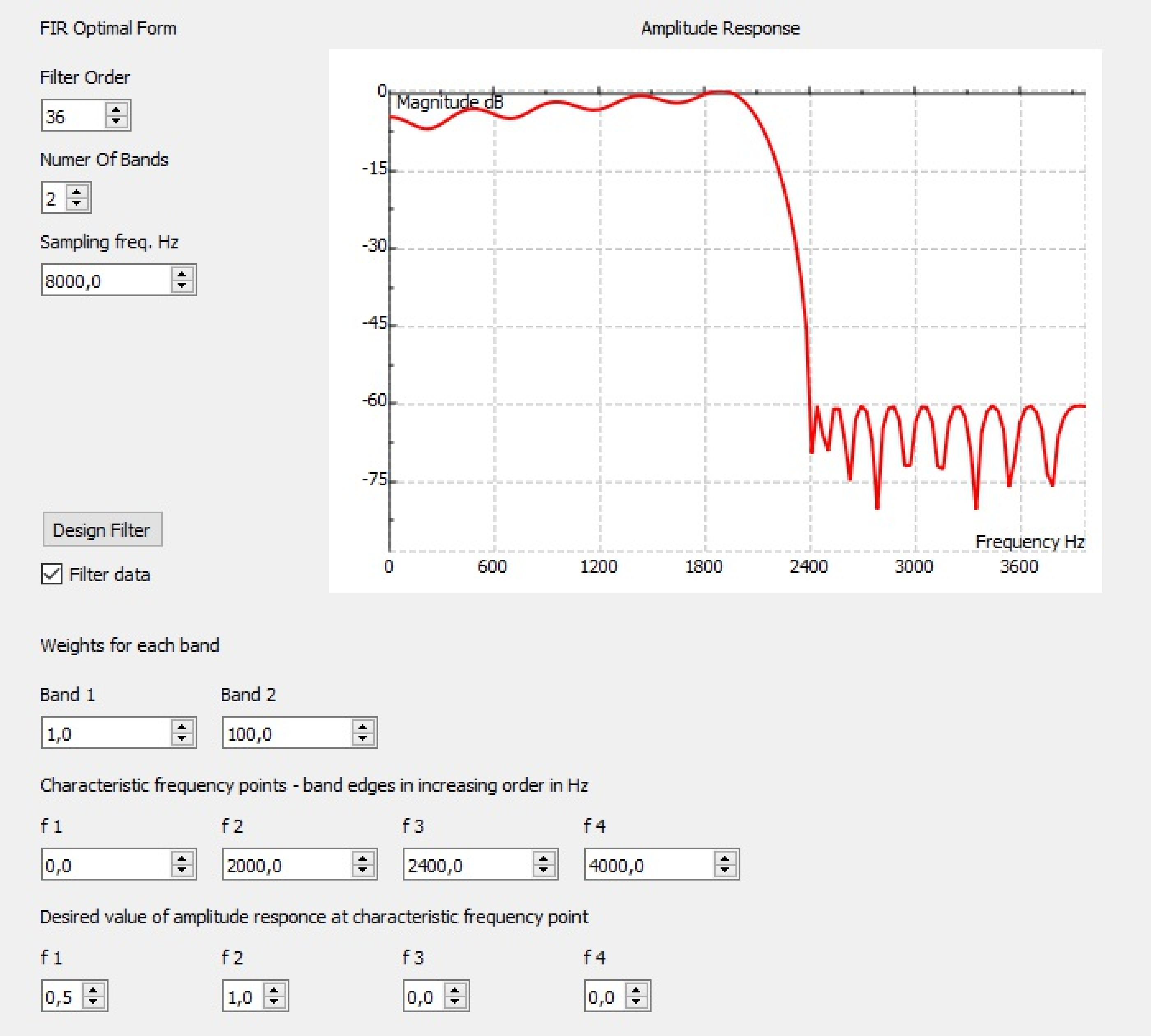
$As := 60$ dB, stopband attenuation

$$n := \text{firoptord}\left(numbands\ ,\begin{bmatrix}0\\2000\\2400\\Fs/2\end{bmatrix},\begin{bmatrix}1\\1\\0\\0\end{bmatrix},\begin{bmatrix}0.1\\0.001\end{bmatrix},Fs\right)$$ Estimate filter order

The next step is to design a filter, and to apply a filter to a noisy signal. In order, to design a filter, firoptimalform is used, with parameters defined above.

```
f3:=firoptimalform(0 , "FIRF1")       Create filtering form

h1f:=firoptimalresult(f3 , h1n)      Filter data using designed FIR filter
```



FIR Optimal Form

Filter Order

36

Numer Of Bands

2

Sampling freq. Hz

8000,0

Design Filter

☑ Filter data

Weights for each band

Band 1	Band 2
1,0	100,0

Characteristic frequency points - band edges in increasing order in Hz

f 1	f 2	f 3	f 4
0,0	2000,0	2400,0	4000,0

Desired value of amplitude responce at characteristic frequency point

f 1	f 2	f 3	f 4
0,5	1,0	0,0	0,0

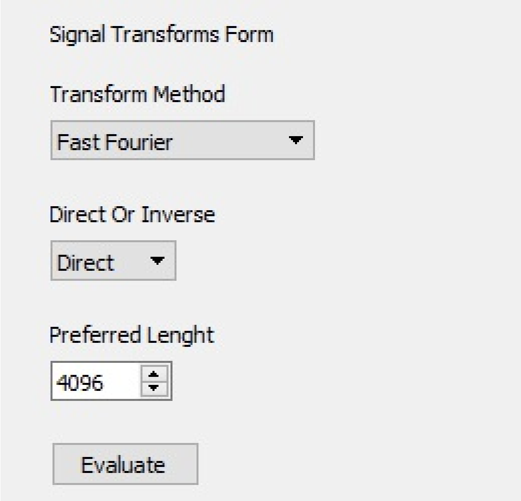
Spectrum of the filtered signal

What is left is to check spectrum of the filtered signal. For that purpose, we design one additional signal transform form in order to calculate signal spectrum. **Conclusions**

It has been shown how .wav signal can be analyzed in the frequency domain. The noisy speech signal is processed by optimal linear phase FIR filter. The filtered signal is clearly without noise. MatDeck proves to be very useful package for this type of digital signal processing.

$f4 := \mathrm{sigtransform}(0\ ,\ \text{"FormST33"})$ Create form

$H3 := \mathrm{sigtransresult}(f4\ ,\ h1f)$ Determine FFT



Signal Transforms Form

Transform Method

Fast Fourier

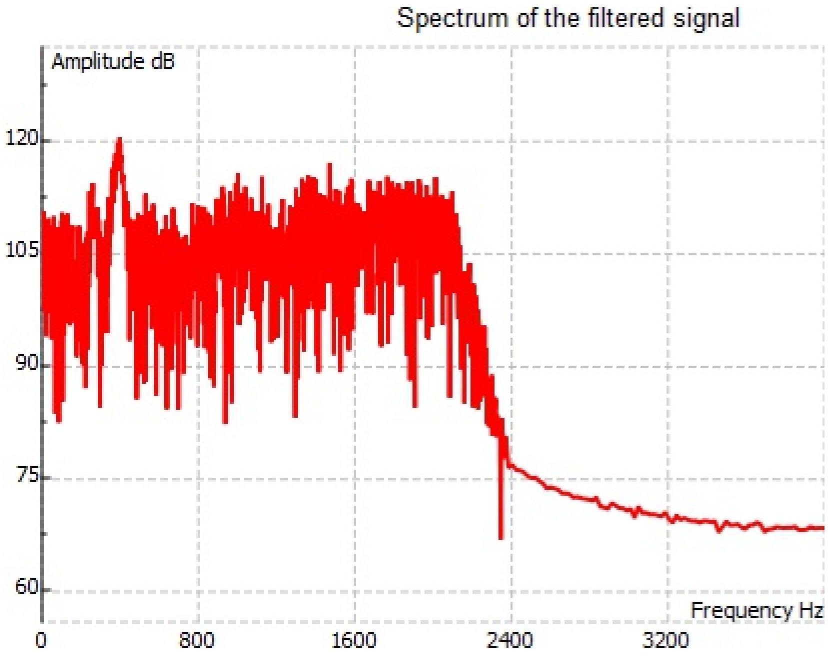
Direct Or Inverse

Direct

Preferred Lenght

4096

$$Sph3 := \mathrm{join\ mat\ cols}\left(Fr\ ,\ 20\ \log10\left(\left|\mathrm{subset}\left(H3\ ,\ 0\ ,\ 0\ ,\ \mathrm{size}(H3)/2-1\ ,\ 0\right)\right|\right)\right)$$


Spectrum of the filtered signal
Amplitude dB
Frequency Hz