The Fundamental Problem of the Spectral Subtraction

B. Runow[1], J. D. Ziegler[2], H. Paukert[3], A. Schilling[4], O. Curdt[5]

[1] *Wilhelm-Schickard Institut, Eberhard Karls University, Tübingen, Email: bernfried@runow.info*
[2] *Wilhelm-Schickard Institut, Eberhard Karls University, Tübingen, Email: zieglerj@hdm-stuttgart.de*
[3] *Hochschule der Medien Stuttgart, Email: paukert@hdm-stuttgart.de*
[4] *Wilhelm-Schickard Institut, Eberhard Karls University, Tübingen, Email: schilling@uni-tuebingen.de*
[5] *Hochschule der Medien Stuttgart, Email: curdt@hdm-stuttgart.de*

Abstract

Spectral Subtraction is often used for noise reduction and speech enhancement. It is an important tool of digital audio signal processing. Since its introduction in 1979, several problems like Phase Errors, Cross-time Errors and Magnitude Errors cause rather disappointing results. Beyond these errors, there is a fundamental problem within the basic principles of Spectral Subtraction, which is documented in this publication.

1. Introduction

Spectral Subtraction is a widespread method to dynamically process the spectrum of a digital audio signal. It gives you the possibility to edit a signal in a specific spectral range. The basis for this procedure is the discrete Fourier transform (DFT), which converts a time-series signal into the frequency domain and makes frequency analysis possible. In the spectral domain it is possible to edit individual spectral components, the so-called spectral coefficients. This makes it possible to subtract information from a specific frequency component. Finally, the processed signal can be resynthesized by means of an inverse discrete Fourier transform (iDFT). Therefore, the edited signal is available in the time domain once again.

The crucial advantage of the Spectral Subtraction is given by the short-time Fourier transform (STFT). With the STFT, it is possible to decompose a continuous stochastic signal and transform each time segment into the spectral domain. There, the time segments can be edited one after another. After the inverse transformation, the time segments can be recomposed into a continuous signal.

Because of the segmental processing, it is possible to edit each segment individually. This means, we can create an adaptive, real-time signal processing algorithm with a short latency. This is the reason for the importance of the Spectral Subtraction in the last decades. A multitude of applications use this technique, like noise reduction and speech enhancement.

2. Fundamentals

2.1. Windowing of a Signal

The segmentation of a continuous input signal $x(n)$ can be achieved with a window function $w(n)$, as we can see in Fig. 1.

Each segment is multiplied with the window function $w(n)$:

$$x_{win}(n) = x(\eta_{win} + n) \cdot w(n)\,, \tag{1}$$

where $n = 0,1,2,\dots,N-1$ is the discrete time index and N the length of the segments. The variable η_{win} defines the first sample of the current segment.

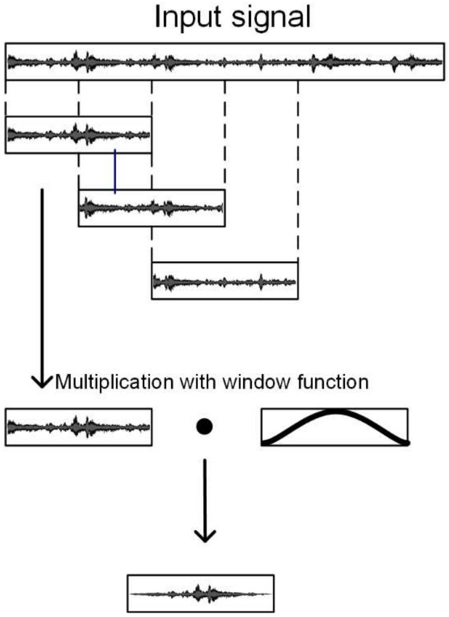

Fig. 1: Windowing of a continuous input signal using the von-Hann window function with an overlap of 50%.

An overlap of the segments is possible. Depending on the length of overlap, a compatible window function has to be chosen. The sum of the successive window functions always has to be one. This restriction is given in order to prevent a distortion of the signal within the resynthesis process, more precisely through the multiplication with the window

ISBN
978-3-9812830-9-9

function. This means that the windowing must result in a constant amplification of 1.

If we don't want an overlap of segments, we can choose the rectangular window:

$$w_{rect}(n) = 1\,, \tag{2}$$

with $n = 0,1,2,\ldots,N-1$.

If we want an overlap of 50%, we can, for example, choose the von-Hann window function:

$$w_{hann}(n) = \frac{1}{2} - \frac{1}{2}\cos\left(2\pi\frac{n}{N-1}\right), \tag{3}$$

with $n = 0,1,2,\ldots,N-1$.

In Fig. 2 you can see the von-Hann window functions for the segmentation of an input signal. Any window function can be used as long as the constraint of constant amplification is met.

Because of the use of a window function, a segment is also called window.

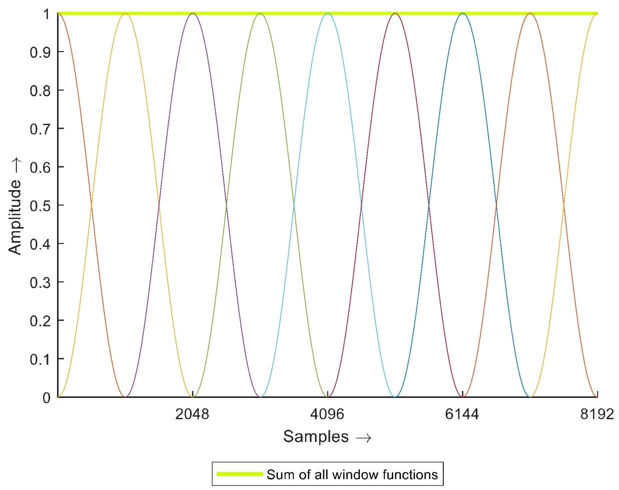

Fig. 2: Von-Hann window functions with a length of 2048 samples and their sum.

2.2. Short-Time Fourier Transform

After the segmentation of the input signal, the short-time Fourier transform (STFT) uses the Discrete Fourier transform to transport each window into the frequency domain. We obtain the DFT-coefficients using [4][8]:

$$X_{win}(k) = \sum_{n=0}^{N-1} x_{win}(n) \cdot e^{-j2\pi k\frac{n}{N}}\,, \tag{4}$$

where $k = 0,1,2,\ldots,N-1$ is the discrete frequency index.

Each DFT coefficient represents a constant oscillation with the dedicated frequency f_k:

$$f_k = f_s \cdot \frac{k}{N}\,, \tag{5}$$

where f_s represents the sampling frequency which was used for the sampling during the digitalisation of the input signal. The absolute value of the DFT coefficient is the amplitude $|X_{win}(k)|$ of the oscillation and $\angle X_{win}(k)$ describes the corresponding phase angle.

By means of the inverse discrete Fourier transform we can transport the spectral signal $X_{win}(k)$ back into the time domain [4][8]:

$$x_{win}(n) = \frac{1}{N}\sum_{k=0}^{N-1} X_{win}(k) \cdot e^{j2\pi n\frac{k}{N}}\,, \tag{6}$$

with $n = 0,1,2,\ldots,N-1$. Thus, the two signal sequences $x_{win}(n)$ and $X_{win}(k)$ are a transform pair.

Finally, the processed signal segments can be recombined according to the defined overlap.

2.3. Characteristics of the STFT

The Short-time Fourier transform has a number of characteristics which are accurately described in the relevant literature [2][4][8][9]. Two of these characteristics are especially important for Spectral Subtraction: the periodicity and the resolution of time and frequency.

2.3.1. Periodicity

The exponential function $e^{-j2\pi kn/N}$ behaves in a periodic fashion depending on N. This results the periodicity of the DFT and consequently of the STFT [4][8]:

$$X_{win}(k) = X_{win}(k+N) \tag{7}$$

and

$$x_{win}(n) = x_{win}(n+N)\,. \tag{8}$$

2.3.2. Time Resolution and Frequency Resolution

By using a clever analogy to the Heisenberg uncertainty principle, Küpfmüller points out that it is not possible to simultaneously achieve both a high resolution in time and in frequency within the spectral domain [7].

The background of this principle is the identical length N of the transform pair consisting of the time-domain signal $x_{win}(n)$ and the signal in the frequency domain $X_{win}(k)$. To get a high frequency resolution, we need a preferably long signal length. Contrarily we achieve a high time resolution using a short window in the time domain as this enables us to compute an individual spectrum for each short time segment.

Fig. 3-5 make this uncertainty principle clear. We can see several spectra over time. The test signal, which is a sine wave changing its frequency every second, was transformed into the spectral domain by means of STFT. The charts differ in the window lengths which were used for the STFT.

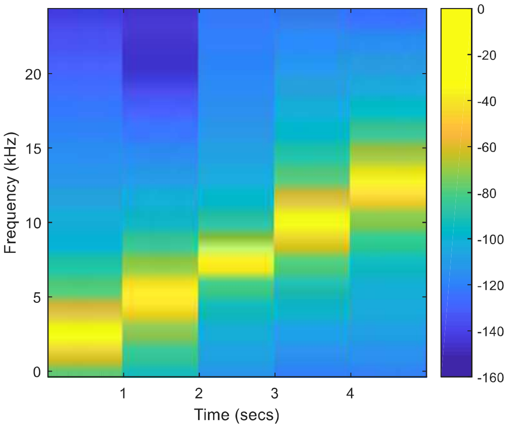

Fig. 3: Spectrogram of a sine wave changing its frequency every second. Analysed using STFT with a window length of 64 samples and a sampling frequency of 48kHz.

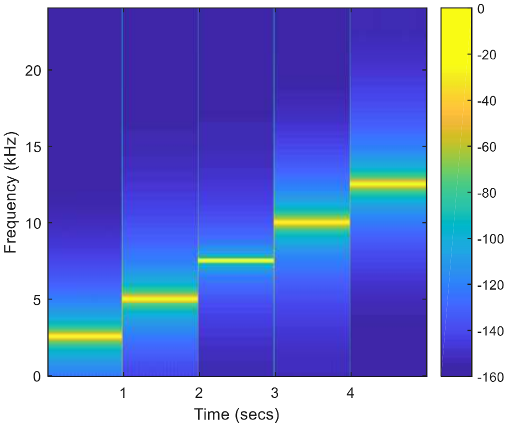

Fig. 4: Spectrogram of a sine wave changing its frequency every second. Analysed using STFT with a window length of 512 samples and a sampling frequency of 48kHz.

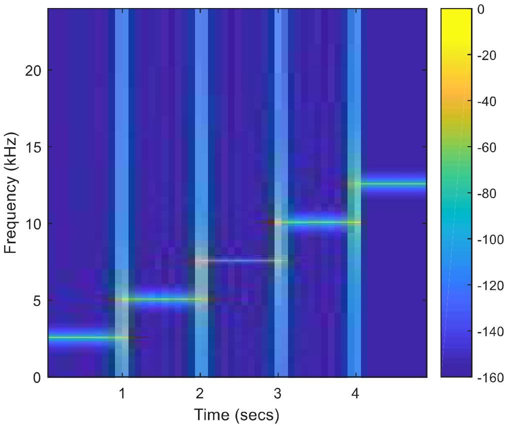

Fig. 5: Spectrogram of a sine wave changing its frequency every second. Analysed using STFT with a window length of 8192 samples and a sampling frequency of 48kHz.

We can solve this conflict with the help of a process called ‘Zero Padding’. To get a high frequency resolution for a short time segment we can add a number of zeros at the end of the windowed time signal:

$$\tilde{x}_{win}(n) = \begin{cases} x_{win}(n) & for\ 0 \leq n \leq N-1 \\ 0 & for\ N \leq\ n \leq N+L-1\,, \end{cases} \qquad (9)$$

where $n = 0, 1, 2, \ldots, N+L-1$ and L represents the number of the added zeros. Thus, it is possible to simultaneously achieve a high time resolution and a high frequency resolution within the STFT.

2.4. Spectral Subtraction

During Spectral Subtraction the amplitudes of two spectral signals are subtracted from each other. If $|X_{win}(k)|$ is the minuend and $|U_{win}(k)|$ is the subtrahend, we obtain the difference [3]:

$$|Y_{win}(k)| = |X_{win}(k)| - |U_{win}(k)| \cdot \upsilon(k, p = 1)\,, \qquad (10)$$

where $\upsilon(k,p)$ is a real weighting factor which regulates the subtrahend $|U_{win}(k)|$, so that $|Y_{win}(k)|$ cannot assume negative values:

$$\upsilon(k,p) = \begin{cases} 1 \cdot \iota & for\ |U_{win}(k)| \leq |X_{win}(k)| \\ \dfrac{|X_{win}(k)|^p}{|U_{win}(k)|^p} \cdot \iota & for\ |U_{win}(k)| > |X_{win}(k)| \end{cases}. \qquad (11)$$

The real factor $0 \leq \iota \leq 1$ defines the intensity of the Spectral Subtraction. If $\iota = 0$, there is no subtraction. If $\iota = 1$, the subtraction is maximal. The quotient of $|X_{win}(k)|$ and $|U_{win}(k)|$ prevents that $|Y_{win}(k)|$ can become negative if the absolute value of $U_{win}(k)$ is larger than the absolute value of $X_{win}(k)$.

If we don’t want to subtract the amplitudes, but the power, equation (10) is modified to produce $|Y_{win}(k)|$:

$$|Y_{win}(k)| = \sqrt{|X_{win}(k)|^2 - |U_{win}(k)|^2 \cdot \upsilon(k, p = 2)}\,. \qquad (12)$$

A more general form can be written as:

$$|Y_{win}(k)| = \left(|X_{win}(k)|^p - |U_{win}(k)|^p \cdot \upsilon(k,p)\right)^{\frac{1}{p}}. \qquad (13)$$

This is often named parametric spectral subtraction [5] and sets a variable exponent. With $p = 1$ we obtain the spectral subtraction from (10) and with $p = 2$ we obtain the spectral subtraction of the power from (12).

Combined with the phase $\angle X_{win}(k)$ of the input signal $X_{win}(k)$, the output signal can be computed with:

$$Y_{win}(k) = |Y_{win}(k)| \cdot e^{j\angle X_{win}(k)}\,. \qquad (14)$$

To an extent, this operating sequence is a makeshift method. It is to be expected that after the subtraction, the correct phase of $Y_{win}(k)$ is not identical to the phase of the input signal $X_{win}(k)$. Jens Groh asserts that the correct phase often cannot be derived [6]. Thus, in many cases, the correct phase of the output signal is simply unknown. Studies have shown, that phase corruption in the spectral domain is considerably less perceptible than a corruption of the amplitude in this domain [10].

Finally, the output signal can be transformed back into the time domain by using the iDFT:

$$y_{win}(n) = \frac{1}{N}\sum_{k=0}^{N-1} Y_{win}(k)\cdot e^{j2\pi n\frac{k}{N}}\,. \qquad (15)$$

Thereby the output signal is as long as the input signal and consists of N samples.

3. Spectral Subtraction as a Time-Variant System

The Spectral Subtraction can be considered as a time-variant system with a varying processing and parameters that can change from window to window.

Hence, we are able to write the subtraction in the spectral domain from (13) as a multiplication:

$$\begin{aligned} |Y_{win}(k)| &= \left(|X_{win}(k)|^p - |U_{win}(k)|^p \cdot v(k,p)\right)^{\frac{1}{p}} \\ &= |X_{win}(k)| \cdot \left(1 - \frac{|U_{win}(k)|^p}{|X_{win}(k)|^p}\cdot v(k,p)\right)^{\frac{1}{p}}. \end{aligned} \qquad (16)$$

Then the amplitude response of this system is:

$$H_0(win,k) = \left(1 - \frac{|U_{win}(k)|^p}{|X_{win}(k)|^p}\cdot v(k,p)\right)^{\frac{1}{p}}, \qquad (17)$$

and the frequency response of each window is:

$$H_{win}(k) = H_0(win,k)\cdot e^{j\angle H_{win}(k)}\,. \qquad (18)$$

Assuming $\angle H_{win}(k) = \angle X_{win}(k)$, the equations (17) and (18) leads us to:

$$H_{win}(k) = \left(1 - \frac{|U_{win}(k)|^p}{|X_{win}(k)|^p}\cdot v(k,p)\right)^{\frac{1}{p}} \cdot e^{j\angle X_{win}(k)}. \qquad (19)$$

Like the input signal $X_{win}(k)$, the frequency response consists of N DFT-coefficients. Thus, the spectral output signal $Y_{win}(k)$ can be computed as a product of the spectral input signal and the frequency response $H_{win}(k)$:

$$Y_{win}(k) = X_{win}(k)\cdot H_{win}(k) \qquad (20)$$

A multiplication in the spectral domain corresponds to a convolution of the equivalent signals in the time domain [2]:

$$\begin{aligned} y_{win}(n) &= x_{win}(n) * h_{win}(n) \\ &= \sum_{m=0}^{N_{IR}-1} x_{win}(n)\cdot h_{win}(n-m)\,, \end{aligned} \qquad (21)$$

where $h_{win}(n)$ describes the impulse response of the system and N_{IR} is the length of this impulse response.

4. The Fundamental Problem

4.1. The Length of the Output Signal

The length of the output signal of a convolution is [2]:

$$N_{conv} = N_{input} + N_{IR} - 1\,, \qquad (22)$$

where N_{input} is the length of the input signal, N_{IR} is the length of the impulse response and N_{conv} is the length of the convolved signal.

Considering the convolution in (21), both the input signal and the impulse response are of length N. Therefore, the output signal consists of $2N - 1$ samples.

This means, that the output signal computed using convolution in the time domain is nearly twice as long as the output signal which is computed using Spectral Subtraction in the spectral domain and which has N samples. Thus, the output signal $y_{win}(n)$ in (21) cannot be the same as the output signal in (15) with (13) and (14), as we can see in Fig. 6.

The reason for this is the static signal length in the spectral domain and the periodicity of the DFT. The periodicity presupposes a continuous repetition of the finite output signal. The modifications of the DFT coefficients cause an extension of the signal when transformed back into the time domain. The part of the processed signal after the Nth sample will be continued at the beginning of the window. Since the STFT does not take this repetition at the recombination of the windows into account, an error inevitably occurs. We receive an overlap with a signal part, which is inserted at the wrong time position. This error becomes apparent when the signal is compared directly with the output signal, which is computed by convolution in the time domain. In Fig. 6 we can see the differences between the output signal of the Spectral Subtraction and the output signal of the convolution.

4.2. Zero Padding is no Solution

By using zero padding, we can reduce the effective length of the input signal in relation to the length of the window $N_{input} + L$. Consequently, the length of the frequency response $H_{win}(k)$ increases and for this reason the length of the impulse response $h_{win}(n)$ will increase up to the extended window length of $N_{input} + L$ samples.

The constraint that the output signal fits into the window without an overlap is only fulfilled in the case of $N_{input} = 1$:

$$\begin{aligned} N_{input} + N_{IR} - 1 &\le N_{input} + L \\ 2N_{input} + L - 1 &\le N_{input} + L \\ N_{input} &\le 1\,. \end{aligned} \qquad (23)$$

This case is unusable for Fourier analysis.

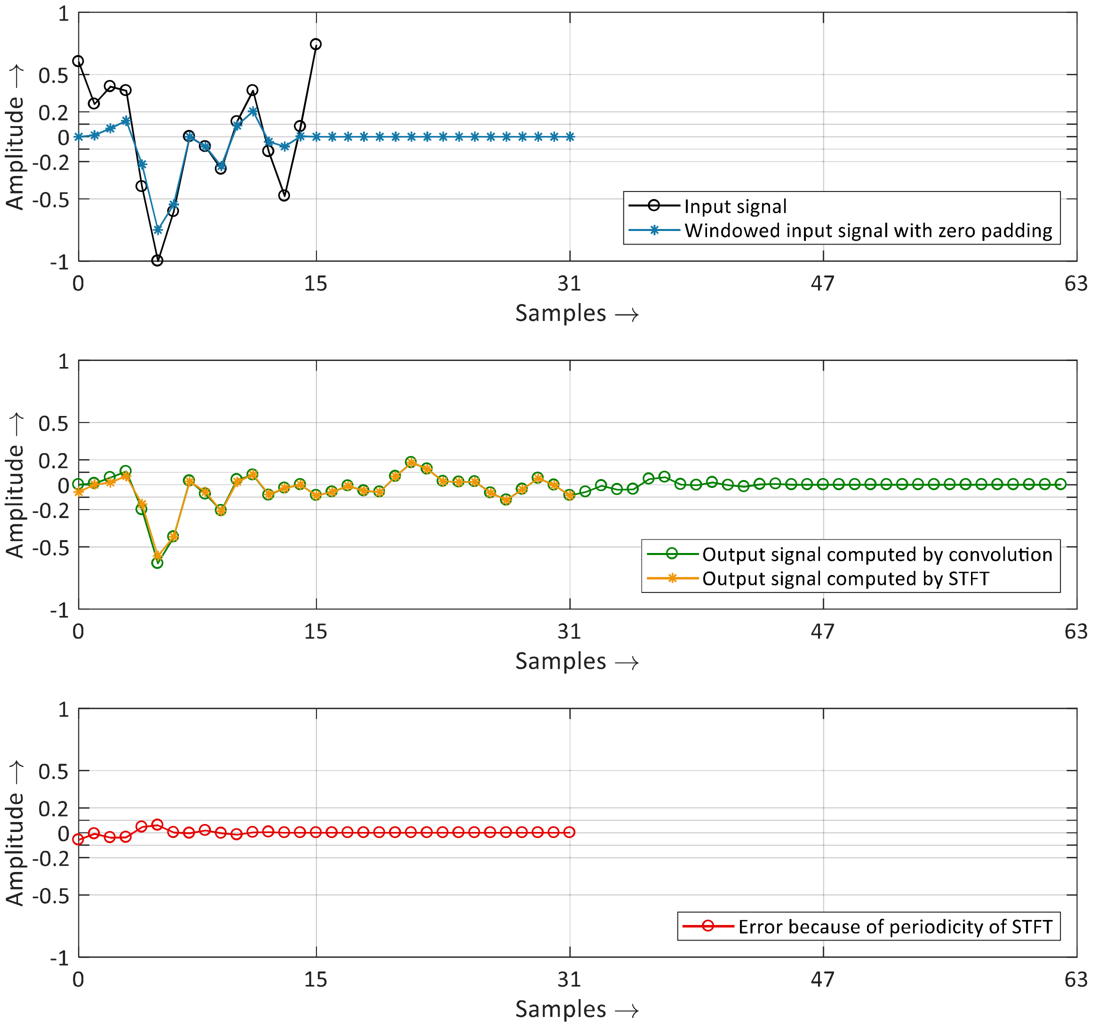

Fig. 6: Comparison of Spectral Subtraction using STFT and the equivalent processing with a convolution in the time domain. A windowed test signal of 16 samples is processed with Spectral Subtraction in the spectral domain and with a convolution in the time domain. The last diagram shows the signal part which is at the wrong position in the output signal when processed with the Spectral Subtraction.

4.3. An Example to Illustrate

To illustrate the behaviour of the DFT in combination with Spectral Subtraction we generate a window of a synthetic input signal:

$$x(n) = \cos\left(2\pi\frac{n}{N}\right) + \sin\left(4\pi\frac{n}{N}\right) + \cos\left(8\pi\frac{n}{N}\right), \qquad (24)$$

with $N = 16$ and $n = 0,1,2,\dots,N-1$. The result is the black graph in Fig. 6. We multiply this input signal with the von-Hann window function from (3):

$$x_{win}(n) = x(n) \cdot w_{hann}(n)\,. \qquad (25)$$

To get a better frequency resolution we add 16 zeros:

$$\tilde{x}_{win}(n) = \begin{cases} x_{win}(n) & for\ 0 \le n \le 15 \\ 0 & for\ 16 \le n \le 31\,. \end{cases} \qquad (26)$$

We receive the windowed input signal with zero padding, as illustrated by the blue graph of Fig. 6.
As an example, we reduce the third, fifth and ninth DFT coefficients by about 70%, using Spectral Subtraction and (4), (10) and (14). The result is the output signal of the Spectral Subtraction, shown as the orange graph. Now we compare this result with the equivalent processing using convolution in the time domain. By means of (19) with $p = 1$ and (21), we receive the green graph. The difference of these two output signals (red graph) shows the wrongly inserted part of the signal, occurring due to the periodicity of the DFT.

5. Analysis of the Impulse Response

If we look to the impulse response $h_{win}(n)$ of the Spectral Subtraction, which is the inverse Fourier transform of $H_{win}(k)$:

$$h_{win}(n) = \frac{1}{N}\sum_{k=0}^{N-1} H_{win}(k) \cdot e^{j2\pi n\frac{k}{N}}\,, \qquad (27)$$

it becomes apparent, that the maximum of the impulse response is located at the first sample $n = 0$, as we can see in Fig. 7.

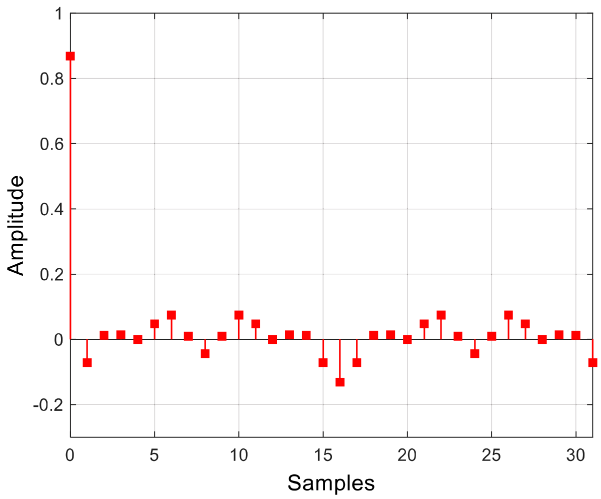

Fig.7: Impulse response of the Spectral Subtraction, computed with (10) and (14). The third, fifth and ninth DFT coefficients are reduced by ~70%.

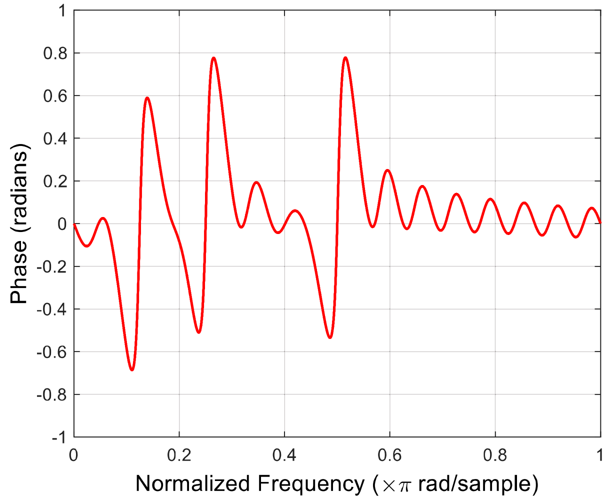

Fig. 8: Phase response of the Spectral Subtraction, computed with (10) and (14). The third, fifth and ninth DFT coefficients are reduced by ~70%.

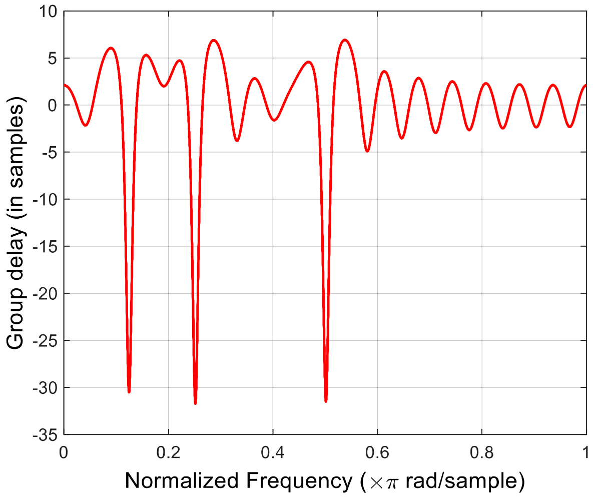

Fig. 9: Group delay response of the Spectral Subtraction, computed with (10) and (14). The third, fifth and ninth DFT coefficients are reduced by ~70%.

Furthermore, the samples $n = 1$ to $n = 31$ are axis-symmetric to $n = 16$. This impulse response behaves as if multiplied with the Heaviside step function:

$$\mathfrak{H}(n) = \begin{cases} 0 & for\ n < 0 \\ 1 & for\ n \geq 0\,, \end{cases} \tag{28}$$

and shows a nonlinear phase shift, as we can see in Fig. 8 and a strong varying group delay depending on frequency, as we can see in Fig. 9. We obtain the strongest group delay at the three processed DFT coefficients.

To prevent nonlinear phase shifting and an inconstant group delay, we must shift the phase within the processing in the spectral domain, depending on frequency. The phase of the DFT coefficients representing high frequencies with a short wavelength have to be shifted more than the phase of DFT coefficients representing low frequencies. For an impulse response with an even length and an even symmetry we obtain the phase difference [2][9]:

$$\theta(k) = -\frac{N-1}{2}\Omega\,, \tag{29}$$

where $\Omega = 2\pi k/N$ is the normalized complex angular frequency. If we include this phase difference in (14), we receive:

$$Y_{win}(k,\theta) = |Y_{win}(k)| \cdot e^{j\angle X_{win}(k)} \cdot e^{j\theta}\,. \tag{30}$$

We can call this enhanced algorithm 'Advanced' Spectral Subtraction.

In Fig. 10–12 we can see the symmetric impulse response, the linear phase response and the constant group delay of the Advanced Spectral Subtraction using (10) and (30).

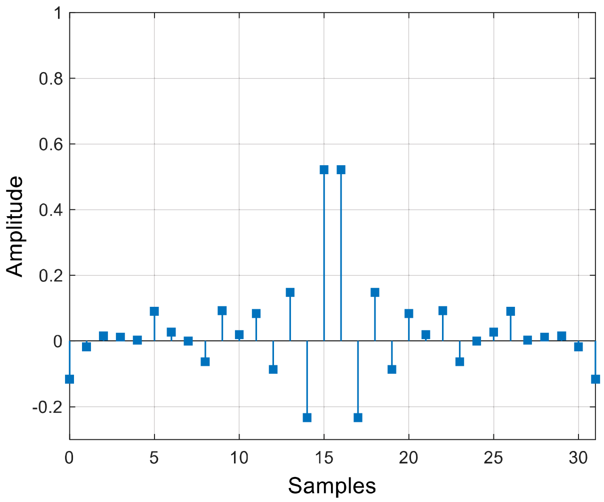

Fig. 10: Impulse response of the Spectral Subtraction, computed with (10) and (30). The third, fifth and ninth DFT coefficients are reduced by ~70%.

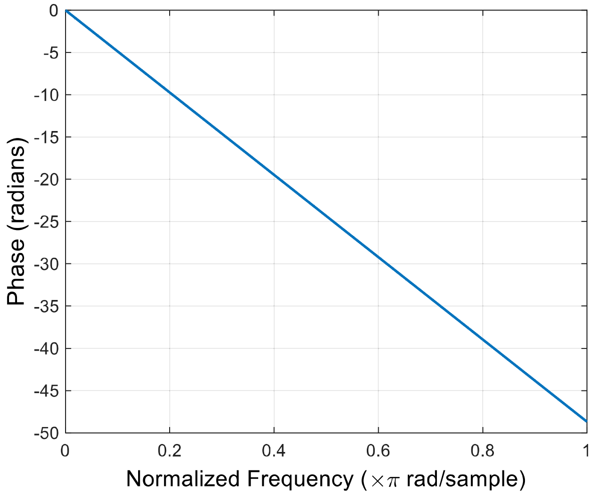

Fig. 11: Phase response of the Spectral Subtraction, computed with (10) and (30). The third, fifth and ninth DFT coefficients are reduced by ~70%.

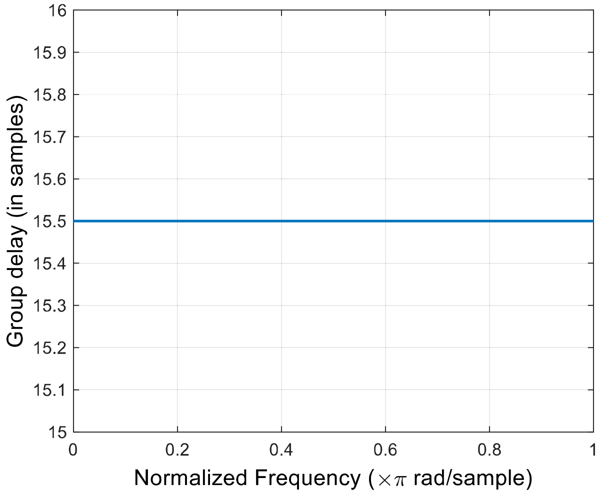

Fig. 12: Group delay response of the Spectral Subtraction, computed with (10) and (30). The third, fifth and ninth DFT coefficients are reduced by ~70%.

As we can see in Fig. 12, the Advanced Spectral Subtraction results in a constant group delay, which also means that the processing has a latency of one half window length.

Finally, we can take a look at the two magnitude responses, computed by Spectral Subtraction using (10) and (14) and by the Advanced Spectral Subtraction using (10) and (30).

The two magnitude responses show strong similarity. We can see the three attenuations, with the red one providing a slightly narrower band width. It also becomes apparent that the Spectral Subtraction with linear phase has a low-pass behaviour at very high frequencies. This is the result of an impulse response with an even length and an even symmetry [2][9]. In the vast majority of cases, this behaviour is of little to no consequence. For example, in digital audio signal processing with a sampling frequency of $f_s = 48kHz$, the cut off is located above the upper limit of human perception.

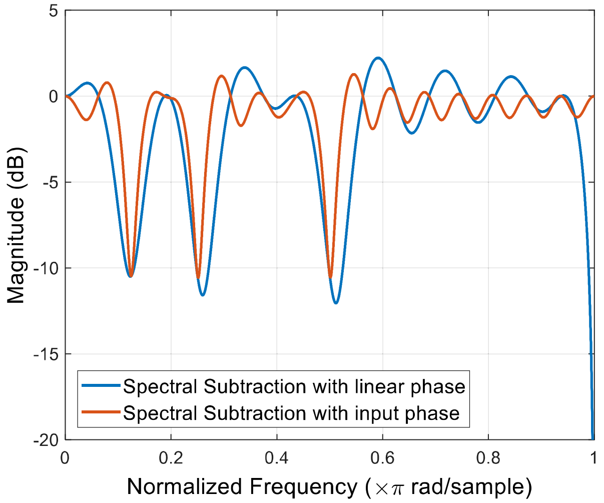

Fig. 13: Magnitude response of the Spectral Subtraction. The blue graph is computed with (10) and (30) and the red graph is computed with (10) and (14). The third, fifth and ninth DFT coefficients are reduced by ~70% within the processing.

6. Conclusion

We can state that processing in the frequency domain makes the signal longer. The signal part by which the output signal is longer than the input signal corresponds to the transient effect and decay process of the impulse response. The crucial point is to arrange the transient and decay parts at the correct time position in the output signal.
If we do the processing in the spectral domain via STFT, because of the periodicity, we receive an overlap in the output signal during resynthesis. This means, that we have a signal part at the wrong time position. Since the STFT does not take this repetition into account, an error inevitably occurs.

The subjective perception of this error is relatively small. Furthermore, it is not the reason of the artefact called 'musical noise'. Presumably, the resulting error is covered by stronger artefacts like the aforementioned 'musical noise', which can occur because of a dynamic processing in the spectral domain, too.

Irrespective of this, it is recommended to work around this error. For example, the resulting amplitude response can be smoothed. This approach minimizes the error, but it does not completely prevent it. To obtain the correct output signal, the frequency response can be generated. By means of the iFFT, we receive the impulse response of the processing. Now it is possible to compute the output signal with convolution of the windowed input signal and the impulse response in the time domain. This means, that the algorithm has more calculating steps and needs more time for the processing. However, with the fast convolution we have a fast-acting tool, which uses the fast Fourier Transform FFT.

The question arises as to why the fast convolution can compute the output signal without an error while still using the DFT. When we use the fast convolution, we have the

windowed input signal and the complete processing information within the impulse response. We don't have to generate the frequency response in the spectral domain. The fast convolution fills up the windowed input signal and the impulse response with enough zeros to fit the entire output signal into the window.
This is still not possible if we generate the frequency response of the Fourier transformed window with the input signal in the spectral domain, like the Spectral Subtraction does. In this case, the frequency response and for this reason the impulse response are always as long as the transformed window. Therefore, the output signal does never fit into the window.

We can conclude that Spectral Subtraction has a fundamental problem within its approach. But it is possible to work around this weak spot and prevent the occurring error. Furthermore, we can use a phase shift within the processing, so that the 'Advanced' Spectral Subtraction does not have any nonlinear phase response or inconstant group delay.

7. References

[1] Benesty, J.; Chen, J.; Habets, E.A.P.: Speech Enhancement in the STFT Domain, Springer Briefs in Electrical and Computer Engineering. Springer, Berlin, 2011.

[2] Bellanger, M.G.: Digital Processing of Signals - Theory and Practice. John Wiley and Sons Ltd, Chichester, 2000.

[3] Boll, S. F.: Suppression of Acoustic Noise in Speech Using Spectral Subtraction. IEEE Tran. on Acoustics, Speech and Signal Processing ASSP-27, 2, 1979, S. 113-120.

[4] Briggs, W. L.; Van Emden, H.: The DFT - An Owners' Manual for the Discrete Fourier Transform. Society for Industrial and Applied Mathematics, Philadelphia, 1995.

[5] Etter, W., Moschytz, G. S.: Noise reduction by noise-adaptive spectral magnitude expansion. Journal of the Audio Engineering Society 42 (1994), S. 341-349.

[6] Groh, J.: Verringerung von Kammfilterverzerrungen bei Multimikrofonaufnahmen. Tagungsbericht 26. Tonmeistertagung (2010), S. 616-625.

[7] Küpfmüller, K.; Kohn, G.: Theoretische Elektrotechnik und Elektronik. Springer-Verlag, Berlin, Heidelberg, 2000.

[8] Neubauer, A.: DFT – Diskrete Fourier-Transformation. Springer Vieweg, Wiesbaden, 2012.

[9] Oppenheim, A.V.; Schafer, R.W.: Digital Signalprocessing. Prentice Hall, Englewood Cliffs, 1975.

[10] Vary, P.: Noise suppression by spectral magnitude estimation-mechanism and theoretical limits. Signal Processing 8 (1985), S. 387-400.