Available online at www.sciencedirect.com
SCIENCE DIRECT®

Measurement

Measurement 39 (2006) 120–129

www.elsevier.com/locate/measurement

Improved spectrum estimation from digitized time series on a logarithmic frequency axis

Michael Tröbs [a,*], Gerhard Heinzel [b]

[a] *Laser Zentrum Hannover e.V., Laser Development, Hollerithallee 8, D-30419 Hannover, Germany*
[b] *Max-Planck-Institut für Gravitationsphysik (Albert-Einstein-Institut), Institut Hannover, Callinstrasse 38, D-30167 Hannover, Germany*

Received 11 October 2004; accepted 24 October 2005
Available online 5 December 2005

Abstract

We present a practical technique for spectrum and spectral density estimation from long time series by Fourier transforms. We apply Welch's popular technique of "averaging over modified periodograms" which uses Fourier transforms of fixed length with time-domain windows and overlap. Our technique retains the basic properties of this method, but computes the optimal frequency resolution individually for each Fourier frequency on a logarithmic frequency axis, thus yielding results that are more useful than those of the standard techniques.

Keywords: Spectrum estimation; Windowed overlapped segmented average

1. Introduction

Two experimental methods commonly used to measure the spectral distribution of an experimental quantity are (a) using analog bandpass filters or (b) digitizing the quantity with an A/D converter and applying digital algorithms to the time series. While (a) is used for high-frequency signals, (b) is a general method that finds widespread application, i.e. in commercial spectrum analyzers. We only consider (b) here.

It is often useful to plot spectral estimates on a logarithmically uniform frequency axis. Special algorithms have been developed in the context of acoustical research [1–3] and their spectral estimates are known as "constant-Q" estimates. A survey of other spectral estimation methods with logarithmically uniform frequency resolution not specifically tailored to musical research can be found in [4].

A popular method for spectral estimation with uniform frequency resolution for the case of abundant equidistant samples is the so-called 'overlapped segmented averaging of modified periodograms'. Here a *periodogram* means the discrete Fourier transform (DFT) of one *segment* of the time series, while *modified* refers to the application of a time-domain window function and *averaging* is used to reduce the variance of the spectral estimates. This

[*] Corresponding author. Tel.: +49 511 762 19841; fax: +49 511 762 2784.
E-mail addresses: michael.troebs@aei.mpg.de (M. Tröbs), gerhard.heinzel@aei.mpg.de (G. Heinzel).

0263-2241/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.measurement.2005.10.010

method is attributed to Welch [5] and is also known under various acronyms such as WOSA (for 'windowed overlapped segmented average'). It is widely used in commercial signal analyzers, oscilloscopes and other equipment. Specialized methods for the case of non-equidistant or very few samples [6,7] or other methods such as the Blackman–Tukey method [8] and the multitaper method [9] exist, but are not considered here.

Modern data acquisition equipment easily allows the collection of very long time series. This leads to spectra covering several decades of frequency which are often plotted on a logarithmic frequency axis in order to display a maximum of information.

The results from a direct application of the WOSA method are, however, equidistant in the frequency domain. A trade-off is necessary between frequency resolution of the spectrum and the number of averages. Especially with a logarithmically scaled frequency axis spanning several decades the problem becomes obvious, as can be seen in Fig. 1: while at low frequencies a narrow frequency resolution is necessary, this resolution leads to an inconvenient density of data-points at higher frequencies, where instead better averaging is desirable.

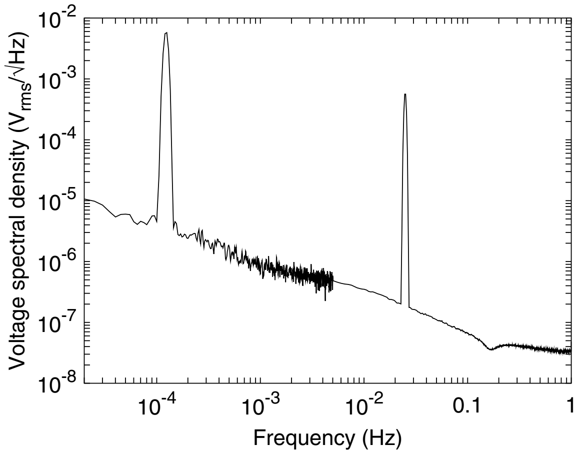

Fig. 2. Spectral density estimate obtained by combining the results of the WOSA method with two different frequency resolutions (10 averages and 1266 averages). The noise floor is preserved, and the peak heights depend on the resolution bandwidth.

The situation can be improved by arbitrarily splitting the frequency axis and using a coarser frequency resolution for the higher Fourier frequency part of the spectral estimate as is shown in Fig. 2: the coarser frequency resolution allows more averages for Fourier frequencies above 5 mHz which results in a smoother curve.

For Figs. 1–3 and 8–10 we have used the measured input voltage noise of an operational amplifier sampled with a digital voltmeter. For illustrative purposes, we have additionally introduced two sinusoidal signals, both with 20 μV_{rms} amplitude and frequencies of 1.23×10^{-4} Hz and 2.5×10^{-2} Hz, and a 3.5 dB notch at 0.16 Hz by digital processing of the measured data.

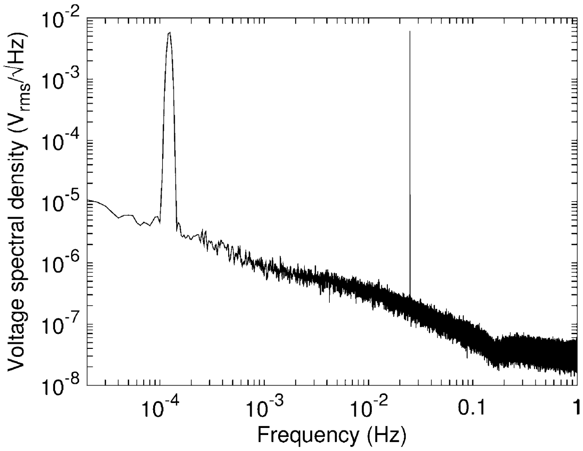

Fig. 1. Spectral density estimate obtained by Welch's method of windowed, overlapped segmented average (WOSA). The noise floor is preserved and the peak heights depend on the resolution bandwidth. Ten estimates were averaged.

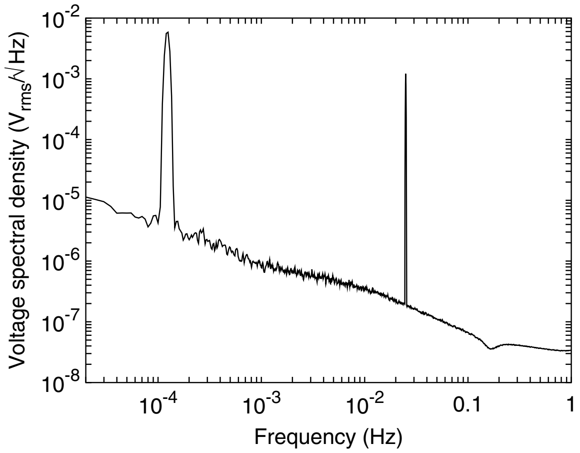

Fig. 3. Spectral density estimate obtained by our new LPSD method. The resolution bandwidth is adjusted to the frequency to obtain more averages at higher frequencies (10–12,982 averages). The peak heights depend on the resolution bandwidth.

In this paper we describe an extension of the WOSA method that is especially suited for the case of very long time series (at least a few thousand samples) and a logarithmic frequency axis. A typical result is shown in Fig. 3. While the WOSA method uses the same frequency resolution for every Fourier frequency, our new method named LPSD (for Logarithmic frequency axis Power Spectral Density) adjusts the frequency resolution for every Fourier frequency in the estimate. The property of the WOSA method that the distance between adjacent Fourier frequencies equals the resolution bandwidth, is approximately maintained.

As will be described below, the computational cost of the LPSD method scales as $\mathcal{O}(N \cdot J)$ with input data size N and number of Fourier frequencies J. For $J = 419$ our non-optimized implementation takes less than 9 s for $N = 10^5$ and less than 2 min for $N = 10^6$ on a typical 2 GHz PC. For the many cases where the authors and their colleagues have used LPSD, the time required for a program run was always negligible compared with the time necessary to collect the input data.

This paper is organized as follows: Table 1 contains a list of symbols commonly used in this paper, in Section 2 we review segmentation of the time ser-

Table 1
Commonly used symbols

Symbols	Comment
α	Parameter for Kaiser window
$a(j,k)$	Average value of segment k
$A(j,k,m(j))$	Discrete Fourier transform of $G(j,k,l)$
C	Normalization factor for spectral estimate
$C_{\mathrm{PS}}(j)$	Calibration coefficient for power spectrum estimation
$C_{\mathrm{PSD}}(j)$	Calibration coefficient for power spectral density estimation
$D(j)$	Segment offset
ENBW	Equivalent noise bandwidth
$f(j)$	Fourier frequencies in spectral estimate, $j = 0,\ldots,J-1$
$f_{\max}$	Largest frequency in spectral estimate
$f_{\min}$	Smallest frequency in spectral estimate
f_{s}	Sampling frequency of the time-series
g	Abbreviation, $g = \log(f_{\max}) - \log(f_{\min})$
$G(j,k,l)$	Data segment minus segment average multiplied by window function
J	Actual number of Fourier frequencies $f(j)$ in spectral estimate
J_{des}	Desired number of Fourier frequencies $f(j)$ in spectral estimate
j	Index $j = 0,\ldots,J-1$ of Fourier frequencies $f(j)$
$K(j)$	Number of averages for Fourier frequency $f(j)$
K_{des}	Desired number of averages
$K_{\min}$	Minimum number of averages
L	Constant number of data in one segment
$L(j)$	Number of data in one segment
l	Data index within one segment $l = 0,\ldots,L(j)-1$
$m(j)$	Frequency index, $m(j) = f(j)/r(j)$
N	Total number of data
n	Index of data $x(n)$, $n = 0,\ldots,N-1$
$P(f(j))$	Spectral estimate for Fourier frequency $f(j)$
r	Constant resolution bandwidth
$r(j)$	Frequency resolutions in spectral estimate, $j = 0,\ldots,J-1$
$r'(j)$, $r''(j)$, $r'''(j)$	Preliminary frequency resolution
r_{avg}	Smallest frequency resolution with K_{des} averages
$r_{\min}$	Frequency resolution corresponding to $K_{\min}$ averages
S_1	Constant sum of all window values
$S_1(j)$	Sum of all window values
S_2	Constant sum of all squared window values
$S_2(j)$	Sum of all squared window values
$w(j,l)$	Window function for Fourier frequency $f(j)$
$x(n)$	Time series of equidistant samples $n = 0,\ldots,N-1$
$x_k(j,l)$	Data segment k, $k = 0,\ldots,K(j)-1$, $l = 0,\ldots,L(j)-1$
ξ	Fractional segment overlap, $0 \leqslant \xi \leqslant 1$
$z(j,l)$	Abbreviation for definition of Hanning window

ies, subtraction of the average value of each segment, multiplication with a window function, computation of the discrete Fourier transforms, and calculation and averaging of their squared magnitudes, all of which are common to both the classical WOSA and our new method. In Section 3 we describe our novel method of selecting equally spaced Fourier frequencies on a logarithmically spaced axis. The main difference between the WOSA method and the LPSD method is that the WOSA method computes scalar products of the modified time series with a complete set of complex exponentials using integer frequency bins while our new method uses individual non-integer bins in the frequency domain. Section 4 discusses the effects of segment overlapping and in Section 5 we study the effects of non-integer bins on the estimate. Section 6 contains the important topic of proper calibration of spectral estimates. Section 7 discusses the effect of aliasing for small frequency bins and in Section 8 we give a step-by-step list for the implementation of the LPSD algorithm.

2. The WOSA method

Consider the N equidistant samples $x(n)$, $n = 0,\ldots,N-1$ of a signal x that has been sampled with sampling frequency f_s. Furthermore, take a set of J Fourier frequencies $f(j)$, $j = 0,\ldots,J-1$ at which the periodogram of x will be evaluated. The bandwidth at each of the frequencies is denoted $r(j)$ and is not necessarily constant (see Section 3). Then, for each j (i.e. for each Fourier frequency $f(j)$) we do the following.

We divide the data into overlapping segments of length $L(j)$ with the starting points of these segments $D(j)$ samples apart as is illustrated in Fig. 4. Here, the segment length $L(j)$ depends on sampling frequency f_s and resolution bandwidth $r(j)$

$$L(j) = f_s/r(j). \tag{1}$$

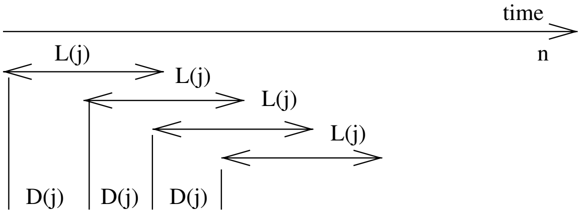

Fig. 4. Segmentation of the data stream.

Note that the $r(j)$ must be chosen in such a way that $L(j)$ are integers. The $D(j)$ depend on the fractional overlap ξ $(0 \leqslant \xi < 1)$ of the segments

$$D(j) = (1-\xi)\cdot L(j). \tag{2}$$

The data of segment k is given by $\{x(l + kD(j)), l = 0,\ldots,L(j)-1\}$. We suppose we have $K(j)$ such segments $(k = 0,\ldots,K(j)-1)$. It is easily shown that $K(j)$ is given by

$$K(j) = \mathrm{floor}\left(\frac{N-L(j)}{L(j)(1-\xi)} + 1\right), \tag{3}$$

where floor( ) takes the largest integer smaller or equal to its argument. For each segment we calculate the mean value

$$a(j,k) = \frac{1}{K(j)}\sum_{l=0}^{L(j)-1} x(l + kD(j)), \quad k = 0,\ldots,K(j)-1 \tag{4}$$

that will be subtracted from each segment to reduce spectral leakage (see also Section 4).

In order to calculate a spectral estimate, we select a data window $w(j,l)$, $l = 0,\ldots,L(j)-1$, and form the windowed sequences $(k = 0,\ldots,K(j)-1)$

$$G(j,k,l) = [x(l + kD(j)) - a(j,k)]w(j,l), \quad l = 0,\ldots,L(j)-1, \tag{5}$$

where the mean has been subtracted separately for each segment.

A brief discussion of windowing and segment overlapping can be found in Section 4 and a detailed discussion is given in [10]. We then compute the scalar products $A(j,k)$ of these sequences with the complex exponential of the appropriate Fourier frequency $f(j)$

$$A(j,k) = \sum_{l=0}^{L(j)-1} G(j,k,l)\exp\left(-2\pi \mathrm{i}\frac{m(j)l}{L(j)}\right), \quad k = 0,\ldots,K(j)-1 \tag{6}$$

with

$$m(j) = f(j)/r(j) \tag{7}$$

being the bin number [11], further discussed in Section 5 below.

We obtain the squared magnitudes of the complex scalar products $|A(j,k)|^2$. Finally, the spectral estimate at the Fourier frequency $f(j)$ is the average of these squared magnitudes, i.e.

$$P(f(j)) = \frac{C}{K(j)} \sum_{k=0}^{K(j)-1} |A(j,k)|^2, \tag{8}$$

where C is a normalization factor that will be discussed in Section 6, and the relation between Fourier frequency $f(j)$ and frequency bin number $m(j)$ is given by

$$f(j) = m(j) \cdot \frac{f_s}{L(j)}. \tag{9}$$

3. Fourier frequencies, frequency resolutions, and bins on a logarithmic axis

If J_{des} Fourier frequencies $f(j)$ are to be spaced equally on a logarithmic axis ranging from $f_{\min}$ to $f_{\max}$, the logarithms of the frequencies obey the linear equation

$$\log(f(j)) = \log(f_{\min}) + \frac{j}{J_{\text{des}}-1}(\log(f_{\max}) - \log(f_{\min})) \tag{10}$$

with $j = 0, \ldots, J_{\text{des}} - 1$, and the frequency limits being bounded by

$$\frac{f_s}{N} \leqslant f_{\min} < f_{\max} \leqslant \frac{f_s}{2} \tag{11}$$

according to the total length of the time series and the Nyquist criterion, respectively. Using the abbreviation

$$g := \log(f_{\max}) - \log(f_{\min}). \tag{12}$$

Eq. (10) can be rewritten to

$$f(j) = f_{\min} \cdot \exp\left(\frac{jg}{J_{\text{des}}-1}\right). \tag{13}$$

In analogy to the WOSA method we aim for a frequency resolution $r(j)$ at Fourier frequency $f(j)$ that equals the difference to the next higher Fourier frequency $f(j+1)$ [12]

$$r'(j) = f(j+1) - f(j), \tag{14}$$

which we solve for our preliminary estimate $r'(j)$

$$\begin{aligned} r'(j) &= f_{\min} \exp\left(\frac{jg}{J_{\text{des}}-1}\right)\left(\exp\left(\frac{g}{J_{\text{des}}-1}\right) - 1\right) \\ &\approx f(j)\frac{g}{J_{\text{des}}-1}. \end{aligned} \tag{15}$$

As will be shown, for common values of N and J_{des}, it is typically not possible to produce spectral estimates at Fourier frequencies $f(j)$ according to Eq. (13) and frequency resolutions according to Eq. (15). In the following we will use $N = 10^6$, $J_{\text{des}} = 1000$, $f_s = 2$ Hz, $f_{\min} = f_s/N = 2$ μHz, and $f_{\max} = f_s/2 = 1$ Hz as example. This leads to a desired frequency spacing of $r'(0) = 26$ nHz between the first two Fourier frequencies which is much smaller than the smallest possible frequency resolution $f_s/N = 2$ μHz.

It turns out that typically for the higher Fourier frequencies, suitable parameters $r(j)$ and $K(j)$ can be found without restrictions, while in the lower end of the spectrum we need to abandon our aim of equally spaced Fourier frequencies on the logarithmically scaled axis. To allow a trade-off between the number of averages and the uniformity of the Fourier frequencies on the logarithmic axis, we introduce an additional parameter in our algorithm, the desired number of averages K_{des}, with typical values of $K_{\text{des}} \approx 100$. This corresponds to a frequency resolution

$$r_{\text{avg}} = \frac{f_s}{N} \cdot (1 + (1-\xi)(K_{\text{des}} - 1)) \tag{16}$$

that also depends on the overlap of the segments and is a constant for each run of the full algorithm. With an overlap of $\xi = 0.3$, $K_{\text{des}} = 100$, and the example values mentioned above, we get $r_{\text{avg}} = 141$ μHz. This means that only for frequency resolutions (spacing between adjacent Fourier frequencies) above 141 μHz is it possible to obtain the desired number of averages.

The smallest possible frequency resolution $r_{\min}$ is given by f_s/N. If it is desired to use at least $K_{\min}$ averages for each Fourier frequency ($K_{\min} \geqslant 1$), then one should use

$$r_{\min} = \frac{f_s}{N} \cdot (1 + (1-\xi)(K_{\min} - 1)) \tag{17}$$

instead.

The procedure to determine suitable Fourier frequencies and corresponding frequency resolutions is as follows:

(1) Start at $f_{\min}$, let $j = 0$, $f(j=0) = f_{\min}$.
(2) Determine the frequency resolution $r'(j)$ according to Eq. (15). Then compute

$$r''(j) = \begin{cases} r'(j), & r'(j) \geqslant r_{\text{avg}}, \\ \sqrt{r_{\text{avg}} \cdot r'(j)}, & r_{\min} \leqslant r'(j) < r_{\text{avg}}, \\ r_{\min}, & r'(j) < r_{\min}. \end{cases} \tag{18}$$

For $r'(j) \geqslant r_{\text{avg}}$, we can obtain both the desired spacing on the logarithmic axis and reach or exceed the desired number of averages. If, on the other hand, $r'(j) < r_{\min}$, the minimum number of averages cannot be achieved, and we have no choice but to use $r_{\min}$ as frequency resolution. Finally, if $r_{\min} \leqslant r'(j) < r_{\text{avg}}$ we choose a heuristic compromise between an equally spaced logarithmic frequency axis and the desired number of averages.

(3) From $r''(j)$ we calculate the closest integer segment length $L(j)$ by

$$L(j) = \text{round}\left(\frac{f_{\text{s}}}{r''(j)}\right). \tag{19}$$

(4) We obtain our final frequency resolution $r(j)$ from the integer segment length $L(j)$ by

$$r(j) = \frac{f_{\text{s}}}{L(j)} \tag{20}$$

and the frequency bin number $m(j)$ from Eq. (7). Note that $m(j)$ can be non-integer (see Section 5).

(5) The next Fourier frequency is given by

$$f(j+1) = f(j) + r(j). \tag{21}$$

If $f(j+1) < f_{\max}$ we increase j by 1 and go to step 2.

(6) Finally, the actual number of Fourier frequencies obtained is given by $J = j + 1$. Since usually for the lower frequencies we have $r'(j) < r_{\text{avg}}$ in step 2, this will be less than were aimed for. The spacing of Fourier frequencies at the upper end of the spectrum, however, corresponds to $J = J_{\text{des}}$.

4. Windowing and overlap of segments

Each segment of length $L(j)$ is multiplied with a window function before being subjected to the DFT. One common feature of the variety of window functions (see e.g. [13]) is that they have a maximum in the middle and tend to zero near the beginning and the end. The effect on the data is twofold: the benefit of a window function is to make the modified data quasi-continuous. This reduces spectral leakage, i.e. the transfer of power from one frequency bin to neighboring frequencies. The other effect is that the data in the middle have a stronger weight than the data at the beginning and at the end, which is undesirable in principle, because all data intrinsically have the same importance. The remedy is to use overlapping segments as is illustrated below.

As an example of a window function we use the Hanning window defined by (see [13])

$$w(j,l) = \frac{1-\cos(z(j,l))}{2} = \cos^2\left(\frac{z(j,l)-\pi}{2}\right) \tag{22}$$

with

$$z(j,l) = \frac{2\pi \cdot l}{N} \quad \text{and} \quad l = 0,\ldots,L(j)-1.$$

For normalization purposes we define two sums, specific to a window function, that are needed for calibrating the spectral estimates (see Section 6)

$$S_1(j) = \sum_{l=0}^{L(j)-1} w(j,l), \tag{23}$$

$$S_2(j) = \sum_{l=0}^{L(j)-1} w^2(j,l). \tag{24}$$

When the time series of data is divided into non-overlapping segments that are multiplied with a window function, we have a situation as illustrated in Fig. 5. Due to the fact that the window function is typically very small or zero near its boundaries, a significant portion of the data is effectively ignored in the analysis. This is clearly not optimal in cases where the data has been produced at great expense and maximal information is to be extracted from it. The situation can be improved by letting the segments overlap as illustrated in Fig. 6. The amount of overlap is a trade-off between computational effort and flatness of the data weighting. A detailed discussion of segment overlap, along with

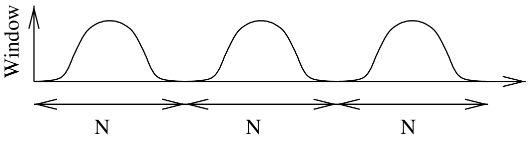

Fig. 5. Non-overlapping segmented data stream with window.

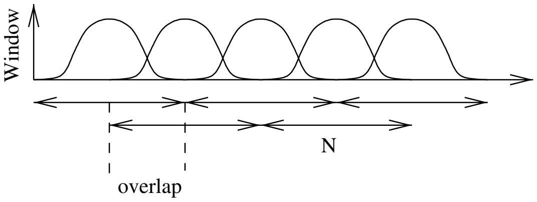

Fig. 6. Data stream with overlapping segments and window.

a comprehensive list of window functions and some new flat-top windows can be found in [10].

5. Non-integer frequency bin numbers

Both the WOSA and the LPSD method for spectral estimation calculate scalar products of windowed data segments with complex exponentials. The main difference between WOSA and LPSD is that the former uses a complete set of integer bin numbers while the latter uses individual non-integer bin numbers in the frequency domain. We have investigated the transfer function of several windows for integer and non-integer bin numbers. Since we discard the phase information in our spectral estimate, we ignore it in this discussion. We have calculated the transfer function magnitude $|H(f)|$ using

$$H(f) = \sum_{l=0}^{L-1} \sin\left(\frac{2\pi f l}{L}\right) w(l) \exp\left(\frac{-2\pi \mathrm{i} l m}{L}\right), \qquad (25)$$

where f is the frequency of the sinusoidal test function in units of bins, $w(l)$ the window function under investigation, and m the frequency bin number of interest. Fig. 7 shows the magnitude of the frequency response of a Hanning window to a sinusoidal input signal for frequencies ranging from 0 to 15, for fixed frequency bin numbers of 6 and 6.23.

At the respective bin number, the window-function responses equal 0 dB, as is expected. Further away, the responses drop and side-lobes are visible. The drop in frequency response between the main lobe and the highest side lobes is called peak side lobe level (PSLL). For the Hanning window this is −31.5 dB for both the integer and the non-integer case.

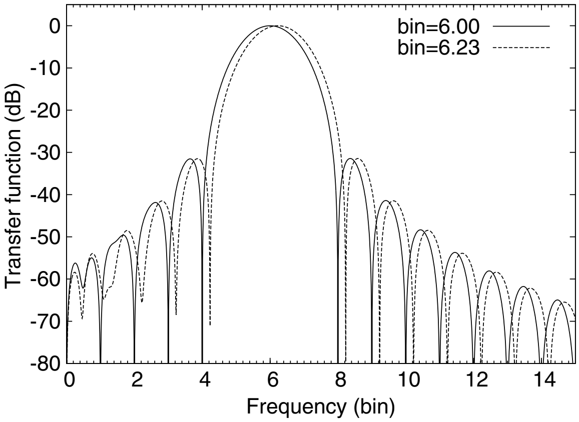

Fig. 7. Magnitude of the Hanning window frequency response for sinusoidal signals.

The PSLL is an important characteristic of a window function: it describes the amount of suppression of spectral leakage of peaks to neighboring frequencies. To obtain spectra without artifacts, a window function with an appropriate PSLL must be chosen. For Figs. 1–3 and 8–10 we have used a Kaiser window [13] with its parameter $\alpha = 3.826$ corresponding to a PSLL of −90 dB.

Fig. 7 shows that the frequency response for non-integer bin numbers slightly differs from the integer

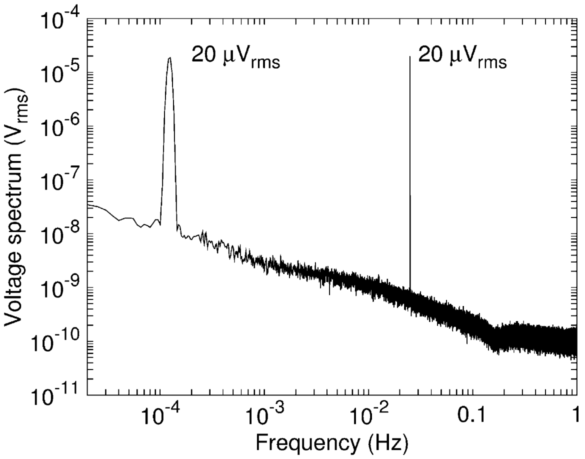

Fig. 8. Spectrum estimate obtained by Welch's method of windowed, overlapped segments averaging (WOSA). The level of the noise floor depends on the resolution bandwidth.

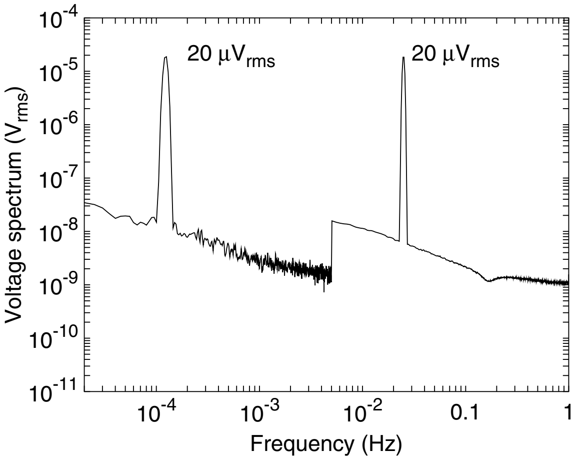

Fig. 9. Spectrum estimate obtained by Welch's method of windowed, overlapped segments averaging (WOSA). The WOSA method has been applied to the data with two different resolution bandwidths to obtain more averages for higher frequencies. Peak heights are preserved, but the noise floor shows a discontinuity due to the change of resolution bandwidth.

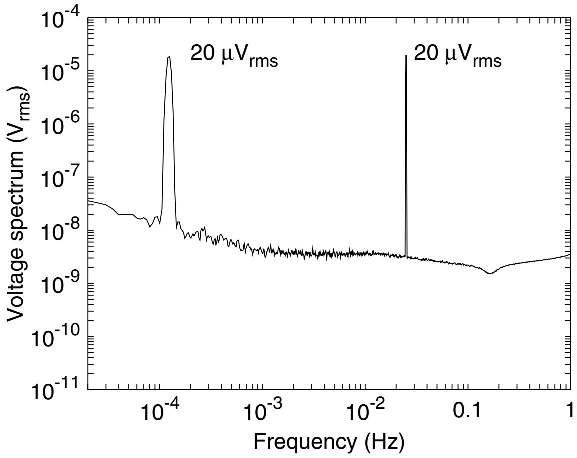

Fig. 10. Spectrum estimate obtained our new LPSD method. Peak heights are preserved and the noise floor is continuous due to the continuous adjustment of resolution bandwidth. The level of the noise floor depends on the resolution bandwidth.

case, but since these differences only occur below the PSLL of the window, they have no effect on the spectrum estimate and can be neglected.

We have done this comparison for many window functions and bin numbers and consistently found that no significant differences exist between integer and non-integer bins, provided that the bin number is higher than the effective half-width of the window transfer function in the frequency domain, measured at the PSLL.

6. Calibration of spectral estimates

The result of any spectral estimation algorithm can be scaled as spectral density or spectrum. Spectral densities are commonly used to estimate the incoherent content of a measured quantity, e.g. a noise floor or a wide-band signal. Spectra, on the other hand, are used to measure the amplitude of sinusoidal peaks.

Table 2 shows four commonly used calibrations for spectral estimates and their relationships.

Spectral densities and spectra with constant frequency resolution can be converted to each other when the equivalent noise bandwidth ENBW is known. It is computed as

$$\mathrm{ENBW} = r \cdot \mathrm{NENBW} = \mathrm{NENBW} \cdot \frac{f_\mathrm{s}}{L} = f_\mathrm{s} \frac{S_2}{(S_1)^2}, \tag{26}$$

where

$$\mathrm{NENBW} = \frac{L S_2}{(S_1)^2} \tag{27}$$

is the normalized equivalent noise bandwidth [13], r the frequency resolution used to calculate the estimate, f_s the sampling frequency, $L = f_\mathrm{s}/r$ the length of the Fourier transforms and S_1 and S_2 window-function-specific sums defined by Eqs. (23) and (24) in Section 4.

The calibration coefficient for power spectrum estimation $C_\mathrm{PS}(j)$ is hence given by

$$C_\mathrm{PS}(j) = \frac{2}{(S_1(j))^2} \tag{28}$$

and the corresponding coefficient $C_\mathrm{PSD}(j)$ for power spectral density is defined as

$$C_\mathrm{PSD}(j) = \frac{2}{f_\mathrm{s} S_2(j)}, \tag{29}$$

where the sums defined in Eqs. (23) and (24) have been used. The linear quantities LSD and LS can be found by taking the square root of PSD or PS, respectively, after averaging has been completed.

Fig. 8 shows a linear spectrum estimate of our time series by the WOSA method. The peak height is 20 μV$_{rms}$ at the correct frequencies.

Fig. 9 shows a linear spectrum estimate of the same time series by the WOSA method, but this time two different frequency resolutions have been used. As a result, the noise floor shows a discontinuity at 5 mHz. The peak heights have however been preserved.

Table 2
Naming convention for spectra

Abbreviation	Name	Relation for constant ENBW	Unit
PSD	Power spectral density		V^2/Hz
PS	Power spectrum	$\mathrm{PS} = \mathrm{PSD} \cdot \mathrm{ENBW}$	V^2
LSD	Linear spectral density	$\mathrm{LSD} = \sqrt{\mathrm{PSD}}$	$V/\sqrt{\mathrm{Hz}}$
LS	Linear spectrum	$\mathrm{LS} = \sqrt{\mathrm{PS}} = \mathrm{LSD} \cdot \sqrt{\mathrm{ENBW}}$	V

ENBW is the equivalent noise bandwidth, defined in Eq. (26).

Fig. 10 shows the result of our new LPSD method. Peak heights have been preserved and due to the continuous adjustment of the frequency resolution, the noise floor is continuous.

7. Aliasing for small frequency bins

It is well known that windowing of data segments is necessary in the WOSA method to reduce the bias of the spectral estimate [14]. When calculating one-sided spectral estimates containing only positive Fourier frequencies windowing causes a bias at low frequency bins—a fact that is also well known: one cannot trust the lowest frequency bins on the spectrum analyzer. The bias stems from aliasing of power from negative bins and bin zero to the lowest positive frequency bins. Aliasing from bin zero can be eliminated by subtracting the mean data value from the segment. Aliasing from negative bins however, cannot be reduced that way. Hence we propose not to use the first few frequency bins. The first frequency bin that yields unbiased spectral estimates depends on the window function used. The bin is given by the effective half-width of the window transfer function. Values for a variety of windows are tabulated in [10].

8. Algorithm summary

In this section we summarize how we have implemented the LPSD algorithm for spectrum and spectral density estimation. We have programmed it in C and compiled executable programs for Linux and DOS/Windows. The source code is available upon request from the authors. Readers requiring a different implementation (e.g. in Matlab, Mathcad, etc.) may use this summary as a guideline.

Input data. We assume to have a long stream $x(n)$, $n = 0,\ldots,N-1$ of equally spaced input data sampled with frequency f_s. Typical values for N range from 10^4 to $>10^6$.

Window function. Choose a window function $w(j,l)$ to reduce spectral leakage within the estimate. Lists of window functions can be found in [13,10]. The computations of the window function will be performed when the segment lengths $L(j)$ have been determined.

Fourier frequencies, frequency resolutions, and bins. Calculate Fourier frequencies $f(j)$ ranging from $f_{\min}$ to $f_{\max}$, frequency resolutions $r(j)$, and frequency bins $m(j)$ according to steps 1 to 5 in Section 3 with j ranging from 0 to $J-1$. A typical value for J is 1000.

For every Fourier frequency, $f(j)$ do:

Splitting of the data stream. The segment length $L(j)$ is determined from the $r(j)$ calculated in the previous step by using Eq. (20). Split the data into segments of length $L(j)$, overlapping as desired. Remove the mean of each segment if desired.

DFT. Compute the window function $w(j,l)$ with l running from 0 to $L(j)-1$. Multiply each data segment with the window function (Eq. (5)), and compute the discrete Fourier transforms (Eq. (6)).

Averaging and calibration. Average the squared magnitudes of Eq. (6) and apply the desired calibration factor according to Eq. (8). Table 2 and Eqs. (28), (29) can be used to determine the desired calibration factor.

Acknowledgements

We thank Oliver Jennrich, Albrecht Rüdiger, and Roland Schilling for useful discussions.

References

[1] J.C. Brown, Calculation of a constant Q spectral transform, J. Acoust. Soc. Am. 89 (1991) 425–434.

[2] J.C. Brown, M.S. Puckette, An efficient algorithm for the calculation of a constant Q transform, J. Acoust. Soc. Am. 92 (1992) 2698–2701.

[3] D.T. Teaney, V.L. Moruzzi, F.C. Mintzer, The tempered Fourier transform, J. Acoust. Soc. Am. 67 (1980) 2063–2067.

[4] C.L. Carnal, R.W. Rochelle, Power spectrum estimation with logarithmically uniform frequency resolution: a survey, in: Conference Proceedings of IEEE Southeastcon, vol. 84, 1984, pp. 418–421.

[5] P.D. Welch, The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms, IEEE Trans. Audio Electroacoust. AU-15 (June) (1967) 70–73.

[6] R.O.R.Y. Thompson, Spectral estimation from irregularly spaced data, IEEE Trans. Geosci. Electr. GE-9 (1971) 220–233.

[7] J.-F. Giovannelli, G. Demoment, A. Herment, A Bayesian method for long AR spectral estimation: a comparative study, IEEE Trans. Ultrason. Ferroelectr. Freq. Control 43 (March) (1996) 220–233.

[8] R.B. Blackman, J.W. Tukey, The Measurement of Power Spectra, Dover Publications, New York, 1959.

[9] D.J. Thomson, Spectrum estimation and harmonic analysis, Proc. IEEE 70 (1982) 1055–1096.

[10] G. Heinzel, A. Rüdiger, R. Schilling, Spectrum and spectral density estimation by the Discrete Fourier transform (DFT), including a comprehensive list of window functions ans some new flat-top windows, Internal Report, Max-Planck-Institut für Gravitationsphysik, Hannover, 2002.

[11] Note that for each Fourier frequency $f(j)$ the data is segmented differently, such that a Fast Fourier Transform (which provides results for all frequency bins with a given segmentation) is usually inefficient, and direct computation of Eq. (6) is often faster.

[12] For any reasonable number of Fourier frequencies on the x-axis it makes no practical difference whether the distance to the next higher frequency, the distance to the next lower frequency or their average is used.

[13] F.J. Harris, On the use of windows for harmonic analysis with the discrete Fourier transform, Proc. IEEE 66 (1978) 51–83.

[14] D.J. Manolakis, V.K. Ingle, S.M. Kogon, Statistical and Adaptive Signal Processing: Spectral Estimation, Signal Modeling, Adaptive Filtering and Array Processing, McGraw-Hill, 2000.