An Analysis of Adaptive Recursive Smoothing with Applications to Noise PSD Estimation

Robert Rehr, *Student Member, IEEE*, and Timo Gerkmann, *Senior Member, IEEE*

***Abstract*—First-order recursive smoothing filters using a fixed smoothing constant are in general unbiased estimators of the mean of a random process. Due to their efficiency in terms of memory consumption and computational complexity, they are of high practical relevance and are also often used to track the first-order moment of nonstationary random processes. However, in single-channel speech-enhancement applications, e.g., for the estimation of the noise power spectral density, an adaptively changing smoothing factor is often employed. Here, the adaptivity is used to avoid speech leakage by raising the smoothing factor when speech is likely to be present. In this paper, we investigate the properties of adaptive first-order recursive smoothing factors applied to noise power spectral density estimators. We show that in contrast to a smoothing with fixed smoothing factors, adaptive smoothing is in general biased. We propose different methods to quantify and to compensate for the bias. We demonstrate that the proposed correction methods reduce the estimation error and increases the perceptual evaluation of speech quality scores in a speech enhancement framework.**

***Index Terms*—Adaptive estimation, error correction, IIR filters, smoothing methods, speech enhancement.**

I. Introduction

Speech enhancement algorithms are often used in communication devices, such as mobile telephones and hearing aids, to reduce the detrimental effects of noise on the perceived speech quality and speech intelligibility. In this paper, we consider the case that only a single microphone is available to capture the signal. For this scenario, a common practice is to suppress specific frequency bands when they are dominated by the background noise, e.g., by using the Wiener filter. In general, the attenuation of the frequency bands is determined by the background noise power spectral density (PSD) and the speech PSD. Several algorithms have been proposed to estimate these PSDs from a noisy mixture, e.g., [1]–[4], [5, Section 14] and [6]. The PSDs can be interpreted as the mean of the speech and the noise periodograms, respectively. Because of the low computational complexity and the low memory demand, first-order recursive smoothing is a commonly applied technique to estimate these means. This first-order recursive smoothing is equivalent to a moving average where an exponentially decaying smoothing window is employed. Here, a stronger weight is put on the more recent samples allowing these filters to track changes of the mean value over time.

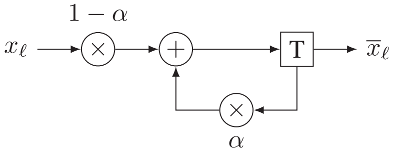

Fig. 1. Block diagram of a first-order recursive filter structure.

In [7], it has been shown that the noise PSD estimators presented in [5, Section 14.1.3] and [6] are implicitly or explicitly based on a first-order recursive structure as shown in Fig. 1. However, in many applications, such as [5], [6], an adaptive smoothing factor $\alpha(x_\ell, \overline{x}_{\ell-1})$ is employed as

$$\overline{x}_\ell = [1-\alpha(x_\ell, \overline{x}_{\ell-1})]x_\ell + \alpha(x_\ell, \overline{x}_{\ell-1})\overline{x}_{\ell-1}, \tag{1}$$

where $\alpha(x_\ell, \overline{x}_{\ell-1})$ is a function of both x_ℓ and $\overline{x}_{\ell-1}$. The quantity x_ℓ is the observation of the random process describing the periodogram of the input signal at time ℓ while $\overline{x}_\ell$ denotes the estimated mean, i.e., the estimated noise PSD. Similar to nonadaptive first-order smoothing, the smoothing factor $0 \leq \alpha(x_\ell, \overline{x}_{\ell-1}) \leq 1$ controls the tracking speed and the variance of the estimate. The noise PSD estimators in [5, Section 14.1.3] and [6] employ adaptive smoothing factors to avoid speech leakage. The algorithm described in [5, Section 14.1.3] switches between two fixed smoothing constants where a larger one is used if the energy of the noisy periodogram is higher than the background noise PSD, i.e., for large *a posteriori* signal-to-noise-ratios (SNRs). In [6], the value of the adaptive smoothing factor is implicitly adapted using the speech presence probability (SPP) and also grows with an increasing *a posteriori* SNR. In contrast to the noise PSD estimator in [5, Section 14.1.3], this results in a soft transition.

A disadvantage of the application of adaptive smoothing is that the estimate of the mean is in general biased as we have previously shown in [7]. In this paper at hand, we analyze this bias and derive an algorithm to compensate for it. The proposed algorithm adds only a low amount of computational complexity to the existing noise PSD estimators as only a computation of

Manuscript received November 9, 2015; revised August 31, 2016 and October 24, 2016; accepted November 18, 2016. Date of publication December 1, 2016; date of current version January 10, 2017. This work was supported by the Ph.D. program "Signals and Cognition." The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yunxin Zhao.

The authors are with the Signal Processing Group, Department of Informatics, University of Hamburg, Hamburg 22527, Germany (e-mail: rehr@informatik.uni-hamburg.de; timo.gerkmann@uni-hamburg.de).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. The material includes Julia and Matlab code for Algorithms 1, 2, and 3. This material is 73.5 kB in size.

Digital Object Identifier 10.1109/TASLP.2016.2633803

a term similar to the Wiener filter and a multiplication with the result is required. We extend the work in [7] and give further insights on the behavior of adaptive smoothing functions. In contrast to [7], we explicitly consider the case of speech presence for the correction which allows to prevent overestimations in high SNR regions. Further, we present a novel method based on the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ between two successive smoothed filter outputs which allows to determine the bias evoked by adaptive smoothing with a higher precision compared to the method used in [7]. Experiments are conducted on real world signals showing that the reduction of the bias leads to a reduced log-error distortion [8] and increases the quality in terms of perceptual evaluation of speech quality (PESQ) scores [9]. Additional experiments are conducted where the influence of signal correlations on the bias is explicitly considered. Throughout the evaluation, we use the noise PSD estimators [5], [6] as examples taken from single-channel noise PSD estimation. In [10], a different way of correcting the bias has been proposed which is not considered here.

This paper is organized as follows: first, we introduce basic properties of adaptive smoothing in Section II. These are used to derive a fixed correction factor to compensate for the bias caused by adaptive smoothing. In Section III and Section IV two different methods are proposed to estimate the fixed correction factor. After that, we apply the bias compensation methods to speech enhancement frameworks. For this, we describe the signal model and explain the relationship between components of the model and the quantities of adaptive smoothing in Section V. In the same section, we also introduce the noise PSD estimators given in [5, Section 14.1.3] and [6] in the context of adaptive smoothing. For the application of noise PSD estimation, we extend the correction method to account for the additional energy of the speech signal in Section VI. The evaluation of the proposed methods follows in Section VII while Section VIII concludes this paper.

II. Basic Properties and Bias Compensation

In this section, we present basic properties of adaptive first-order recursive smoothing: first, adaptive smoothing as defined in (1) does not alter the properties of the input signal x_ℓ in terms of stationarity and ergodicity. Second, the adaptive smoothing functions are scale-invariant if $\alpha(x_\ell, \overline{x}_{\ell-1})$ depends only on the ratio $x_\ell/\overline{x}_{\ell-1}$. Scale-invariance describes the property that if the input x_ℓ is scaled by a factor $r > 0$, the resulting output $\overline{x}_\ell$ is scaled by the same factor r. These two properties allow the bias to be simply compensated by a multiplicative factor.

A. Stationarity and Ergodicity

The propositions 6.6 and 6.31 in [11] state that a process defined by

$$y_\ell = \phi(x_\ell, x_{\ell-1}, \dots) \tag{2}$$

is stationary and ergodic, if the process given by $x_\ell, x_{\ell-1}, \dots$ is stationary and ergodic. Here, $\phi(\cdot)$ is a function of the current and the past elements of the random process, e.g., the adaptive first-order smoothing as in (1). The propositions, however, implicitly assume that the output process y_ℓ exists meaning that the process y_ℓ does not diverge. As the adaptive smoothing factors $\alpha(x_\ell, \overline{x}_{\ell-1})$ are limited to values between zero and one, the filter function in (1) is stable in the sense that a bounded input results in a bounded output. Thus, considering a finite stationary and ergodic input x_ℓ, it follows that also the filter output $\overline{x}_\ell$ is ergodic and stationary.

B. Scale-Invariance

The process of adaptive recursive smoothing (1) is scale-invariant if the adaptive smoothing function depends only on the ratio $x_\ell/\overline{x}_{\ell-1}$. In particular, if the input x_ℓ is scaled by a factor r, the output $\overline{x}_\ell$ is scaled by the same factor r. This property is of particular relevance for the noise PSD estimators considered in Section V as their respective adaptive smoothing function depends only on the ratio $x_\ell/\overline{x}_{\ell-1}$.

The statement can be proven using the method of induction. For linear first-order recursive smoothing filters, it is often assumed that the system is initially at rest, i.e., $\overline{x}_\ell = 0$ for $\ell < 0$. As all of the considered adaptive smoothing functions depend on the ratio $x_\ell/\overline{x}_{\ell-1}$, this assumption is not applicable because of the division by zero. Thus, we assume that the first filter output $\overline{x}_0$ is equal to the first filter input x_0. From the assumption that $\overline{x}_0 = x_0$ it follows that a scaling of x_ℓ by r leads to $r\overline{x}_0 = rx_0$. Hence, it can be shown for the following samples of (1) that

$$\left[1 - \alpha\left(\frac{rx_\ell}{r\overline{x}_{\ell-1}}\right)\right] rx_\ell + \alpha\left(\frac{rx_\ell}{r\overline{x}_{\ell-1}}\right) r\overline{x}_{\ell-1} \tag{3}$$

$$= r\left(\left[1 - \alpha\left(\frac{x_\ell}{\overline{x}_{\ell-1}}\right)\right] x_\ell + \alpha\left(\frac{x_\ell}{\overline{x}_{\ell-1}}\right)\overline{x}_{\ell-1}\right) \tag{4}$$

$$= r\overline{x}_\ell. \tag{5}$$

This shows that the adaptive smoothing procedure is scale-invariant if the smoothing function depends only on the ratio $x_\ell/\overline{x}_{\ell-1}$.

C. Bias Compensation

In this part, we describe how the bias caused by adaptive smoothing can be compensated. For the derivation, we assume that the filter input x_ℓ can be described by a stationary and ergodic random process. Note that the presence of a speech signal in a speech enhancement context will explicitly be taken into account in Section VI. With the stationarity, the ergodicity, and the scale invariance described in the Sections II-A and II-B, the bias can be corrected by multiplying the filter output $\overline{x}_\ell$ by a fixed correction factor c as

$$\check{\overline{x}}_\ell = c\overline{x}_\ell. \tag{6}$$

Here, $\check{\overline{x}}_\ell$ denotes the corrected filter output. For obtaining an unbiased estimate $\mathbb{E}\{\check{\overline{x}}_\ell\} = \mathbb{E}\{x_\ell\}$, the factor has to be set to

$$c = \mathbb{E}\{x_\ell\}/\mathbb{E}\{\overline{x}_\ell\}, \tag{7}$$

where $\mathbb{E}\{\cdot\}$ denotes the statistical expectation operator. As this factor does not depend on the scaling of x_ℓ or $\overline{x}_\ell$, it is sufficient to determine this quantity for a given mean of the input signal, e.g., $\mathbb{E}\{x_\ell\} = 1$. With the assumption of stationarity, the fixed factor c does also not depend on time. Consequently, c can be determined before any processing takes place. The factor c can be considered the bias between filter input and output after convergence. Despite the assumption of stationarity, we show that the bias reflected by c is also applicable to nonstationary signals in the evaluation, i.e., Section VII. Methods that can be employed to determine the fixed correction factor c are presented in Section III and Section IV.

III. Iterative Bias Compensation

In this section, we revise the method for determining the fixed correction factor c that we proposed in [7]. If the adaptive smoothing function depends only on the unsmoothed input x_ℓ but not on the smoothed output $\overline{x}_{\ell-1}$, the bias caused by adaptive smoothing can be determined by analytically deriving the expected value of $\overline{x}_\ell$. Based on the solution obtained for the analytically solvable case, an iterative method has been presented in [7] that can be used to approximately determine the bias for the more complicated case where the adaptive smoothing function also depends on the estimated mean $\overline{x}_{\ell-1}$. The method estimates the fixed correction factor c quite accurately as shown in our evaluations.

First, we consider adaptive smoothing factors $\alpha(x_\ell, \overline{x}_{\ell-1})$ that are independent of the previous filter output $\overline{x}_{\ell-1}$. With this assumption, (1) simplifies to

$$\overline{x}_\ell = [1 - \alpha(x_\ell)]\, x_\ell + \alpha(x_\ell)\overline{x}_{\ell-1}. \tag{8}$$

For the derivations, we assume that all x_ℓ are identically distributed and uncorrelated. Further, using the stationarity property described in Section II-A, we can assume that a stationary input x_ℓ results in a stationary output $\overline{x}_\ell$. From this, it follows that $\mathbb{E}\{\overline{x}_n\} = \mathbb{E}\{\overline{x}_m\}$ where $n \neq m$ are two different time instances. With the first assumption, the expected value $\mathbb{E}\{x_\ell \overline{x}_{\ell-1}\}$ can be written as $\mathbb{E}\{x_\ell\}\mathbb{E}\{\overline{x}_{\ell-1}\}$. Consequently, applying $\mathbb{E}\{\cdot\}$ to (8) and rearranging the terms, results in [7]

$$\mathbb{E}\{\overline{x}_\ell\} = \frac{\mathbb{E}\{x_\ell\} - \mathbb{E}\{x_\ell \alpha(x_\ell)\}}{1 - \mathbb{E}\{\alpha(x_\ell)\}}. \tag{9}$$

The obtained expression depends only on the adaptive function $\alpha(x_\ell)$ and the probability density function (PDF) of x_ℓ.

In the remainder, we consider the case where $\alpha(x_\ell, \overline{x}_{\ell-1})$ depends also on the recursively estimated mean $\overline{x}_{\ell-1}$. This case is more challenging because the quantity $\overline{x}_{\ell-1}$ influences the behavior of the adaptive smoothing factor which, in turn, influences the estimation of $\overline{x}_\ell$. This type of adaptation is, however, the most relevant for noise PSD estimators, e.g., for the approaches [5], [6] considered in Section V.

Deriving $\mathbb{E}\{\overline{x}_\ell\}$ while taking into account the dependence on $\overline{x}_{\ell-1}$ is difficult because $\overline{x}_{\ell-1}$ appears in a generally nonlinear function $\alpha(x_\ell, \overline{x}_{\ell-1})$ and is a random variable itself as it emerges from the combination of all past x_ℓ. Consequently, $\overline{x}_{\ell-1}$ is also correlated with the previous estimates $\overline{x}_{\ell-2}, \overline{x}_{\ell-3}, \cdots$. Hence, the problem was simplified in [7] by replacing $\overline{x}_{\ell-1}$ in the adaptive function by a fixed value ρ. With that, the bias can be determined iteratively based on the result given in (9) as

$$\rho_i = \frac{\mathbb{E}\{x_\ell\} - \mathbb{E}\{x_\ell \alpha(x_\ell, \rho_{i-1})\}}{1 - \mathbb{E}\{\alpha(x_\ell, \rho_{i-1})\}}. \tag{10}$$

Here, ρ_i is the estimate of $\mathbb{E}\{\overline{x}_\ell\}$ obtained for the ith iteration step whereas the initial condition is denoted by ρ_0. This approach is motivated by the recursive update of $\overline{x}_\ell$ in (1), which is performed sample by sample. In each step of (10), however, all samples over an infinite time period are considered. To determine the final estimate of $\mathbb{E}\{\overline{x}_\ell\}$, the iteration is continued until it converges. With the converged ρ_i, the estimated correction factor can be determined as $c = \mathbb{E}\{x_\ell\}/\rho_i$. For the adaptive smoothing factors used in [5], [6], we will show that the parameter ρ_0 does not influence the convergence of the iterative approach. This procedure is summarized in Algorithm 1.

Algorithm 1: Iterative estimation of the fixed correction factor c for adaptive functions depending on $\overline{x}_{\ell-1}$ proposed in Section III. Here, we refer to the solutions for the specific noise PSD estimators [5], [6] where appropriate.

```
$i \leftarrow 0$, $\rho_0 \leftarrow 1$, $\mu \leftarrow 1$.
while convergence criterion for $\rho_i$ is not met do
    Obtain $\rho_{i+1}$ using (10). The solutions for the
    adaptive functions in [5], [6] are given in (31) and
    (32) of Appendix A if $x_\ell$ is exponentially distributed.
    $i \leftarrow i + 1$
end while
Compute compensation factor: $c = \mu/\rho_i$.
```

IV. Estimating the Bias Using Transition Densities

In this section, we propose a novel method for determining the fixed correction factor c. For this, we use the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ which can be considered a description that explains how the smoothing factor $\alpha(x_\ell, \overline{x}_{\ell-1})$ influences the filter output $\overline{x}_\ell$ and vice versa.

If the input samples x_ℓ are assumed to be independent and identically distributed, i.e., stationary and ergodic, the random process of the filter output can be described by the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$. The conditional density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ is a function that depends on the smoothing function $\alpha(x_\ell, \overline{x}_{\ell-1})$ and the distribution of the input variable x_ℓ as we will show later in this section. It can be considered the link that describes how the previous filter output $\overline{x}_{\ell-1}$ affects the behavior of the smoothing function, $\alpha(x_\ell, \overline{x}_{\ell-1})$, and vice versa. In other words, the interaction between $\overline{x}_{\ell-1}$ and $\alpha(x_\ell, \overline{x}_{\ell-1})$ is included in the PDF $f(\overline{x}_\ell|\overline{x}_{\ell-1})$. We use this conditional PDF to optimize the parameters $\boldsymbol{\theta}$ of a known model PDF $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ such that it matches the PDF of the filter output samples, i.e., $f(\overline{x}_\ell)$, as close as possible. To determine the parameters $\boldsymbol{\theta}$, we exploit the stationarity from which it follows that $f(\overline{x}_\ell) = f(\overline{x}_{\ell-1}) = \cdots$. According to that, there is a PDF such that marginalizing $f(\overline{x}_\ell|\overline{x}_{\ell-1})f(\overline{x}_{\ell-1})$ over $\overline{x}_{\ell-1}$ results in the same PDF for $\overline{x}_{\ell-1}$ as for $\overline{x}_\ell$, i.e., $f(\overline{x}_{\ell-1}) = f(\overline{x}_\ell)$. Therefore, we propose to optimize the parameters $\boldsymbol{\theta}$ of a model PDF $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ such that the PDF $\tilde{g}(\overline{x}_\ell|\boldsymbol{\theta})$

obtained by the marginalization

$$\tilde{g}(\overline{x}_\ell|\boldsymbol{\theta}) = \int_{-\infty}^{\infty} f(\overline{x}_\ell|\overline{x}_{\ell-1})\tilde{f}(\overline{x}_{\ell-1}|\boldsymbol{\theta})d\overline{x}_{\ell-1} \tag{11}$$

resembles the originally used model $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ as closely as possible. The similarity between $\tilde{g}(\overline{x}_\ell|\boldsymbol{\theta})$ and $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ is quantified by the Bhattacharyya distance [12]

$$\mathrm{B}\left(\tilde{f},\tilde{g}\right) = -\ln\left[\gamma\left(\tilde{f},\tilde{g}\right)\right]. \tag{12}$$

Here, $\gamma(\cdot)$ is the Bhattacharyya coefficient which is given by

$$\gamma\left(\tilde{f},\tilde{g}\right) = \int_{-\infty}^{\infty}\sqrt{\tilde{f}(x|\boldsymbol{\theta})\tilde{g}(x|\boldsymbol{\theta})}dx \tag{13}$$

for continuous PDFs [13]. The Bhattacharyya coefficient takes values between zero and one where a result of one means that both PDFs are identical. Consequently, the optimal parameters $\hat{\boldsymbol{\theta}}$ are defined as those that minimize the Bhattacharyya distance as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathrm{B}\left(\tilde{f},\tilde{g}\right). \tag{14}$$

As the analytic solution for the integrals in (11) and (13) are unknown, we solve these expressions using numerical integration methods. This also motivates the usage of the Bhattacharyya distance which is numerically easier to handle than other distance measures, e.g., the Kullback-Leibler divergence [14]. After the optimization, the optimal parameters $\hat{\boldsymbol{\theta}}$ are used to determine the expected value $\mathbb{E}\{\overline{x}_\ell\}$. For this, we assume that $m(\boldsymbol{\theta})$ is a function that returns the mean of the model distribution $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ for the parameters $\boldsymbol{\theta}$. With that, the fixed correction factor is determined as $c = \mathbb{E}\{x_\ell\}/m(\hat{\boldsymbol{\theta}})$. This procedure can be described as self-similarity maximization with respect to the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$.

The transition density function $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ can be derived given a model for the PDF of the input $f(x_\ell)$ and (1). As $\overline{x}_{\ell-1}$ is the given variable in the conditional PDF, $\overline{x}_{\ell-1}$ can be thought of as a fixed quantity and (1) can be treated as a function $\overline{x}_\ell = h(x_\ell)$ of the random variable x_ℓ. Then, for a piecewise monotonic function $h(\cdot)$, the conditional density function $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ can be determined using a change of variables as described in [16, Chapter 5]. The solution is given by

$$f(\overline{x}_\ell|\overline{x}_{\ell-1}) = \sum_{m=1}^{M}\frac{f_{x_\ell}(h_m^{-1}(\overline{x}_\ell))}{|h'(h_m^{-1}(\overline{x}_\ell))|} \tag{15}$$

where $h_m^{-1}(\cdot)$ denotes the inverse of the mth monotonic segment of the function $h(\cdot)$ while M denotes the number of monotonic segments of the considered function. Furthermore, $h'(\cdot)$ is the first derivative of the function $h(\cdot)$. In contrast to the iterative method in Section III, the conditional PDF $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ is easily derived for any PDF of the input signal x_ℓ as it is only required to exchange $f_{x_\ell}(\cdot)$ in (15). The whole process of determining the correction factor c is summarized in Algorithm 2.

Algorithm 2: Estimation of the fixed correction factor c by maximizing the self-similarity with respect to the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ as proposed in Section IV. Here, we refer to the solutions of the specific noise PSD estimators [5], [6] where appropriate.

```
Choose a PDF $f(x_\ell)$ that describes the filter input, e.g., (17).
Determine $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ using (15).
For the adaptive functions in [5], [6], the analytical solutions for $h^{-1}(\cdot)$ and $h'(\cdot)$ are given in Appendix B.
Select a model PDF $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$, e.g, (26).
Minimize (14) to obtain $\hat{\boldsymbol{\theta}}$, e.g., using [15].
Compute the correction factor: $c = \mathbb{E}\{x_\ell\}/m(\hat{\boldsymbol{\theta}})$.
```

V. Noise PSD Estimators in the Context of Adaptive Smoothing

In this section, we consider the noise PSD estimators [5], [6] in the context of adaptive smoothing. For this, we introduce the employed signal model in a speech enhancement context and illustrate the relationship between the model components and the quantities of adaptive smoothing in (1). After that, a brief overview over the considered noise PSD estimators is given.

A. Signal Model

The considered smoothing functions are employed in noise PSD estimators that operate in the short-time Fourier transform (STFT) domain. For determining the STFT, the time-discrete input signal is split into overlapping frames and each frame is transformed using the discrete Fourier transform after applying a spectral analysis window. A common window function is, for example, the square-root Hann window. Further, we assume that in the STFT domain the noisy observation is given by a linear superposition of a speech signal and a noise signal

$$X[k,\ell] = S[k,\ell] + D[k,\ell]. \tag{16}$$

The spectrum of the noisy input signal at frame ℓ is denoted by $X[k,\ell]$ while $S[k,\ell]$ and $D[k,\ell]$ represent the speech and noise spectral coefficients, respectively. Additionally, the frequency index is given by k. For better readability, we omit the frequency index k if the dependence on this quantity is not required. We follow the common assumption that the periodogram of the noisy input $|X[\ell]|^2$ follows an exponential distribution which is given by

$$f(|X[\ell]|^2) = \begin{cases}(1/\mu)\exp\left(-|X[\ell]|^2/\mu\right), & \text{if } |X[\ell]|^2 \geq 0,\\ 0, & \text{otherwise,}\end{cases} \tag{17}$$

where $\mu = \mathbb{E}\{|X[\ell]|^2\}$. This model strictly holds if the speech coefficients $S[\ell]$ and the noise coefficients $D[\ell]$ follow circular complex Gaussian distributions. Here, the variance of the speech coefficients $S[\ell]$ and the noise coefficients $D[\ell]$ is denoted by $\sigma_s^2[\ell]$ and $\sigma_d^2[\ell]$, respectively. The considered noise PSD estimators [5], [6] are based on an adaptive recursive smoothing of the noisy periodogram such that the input to

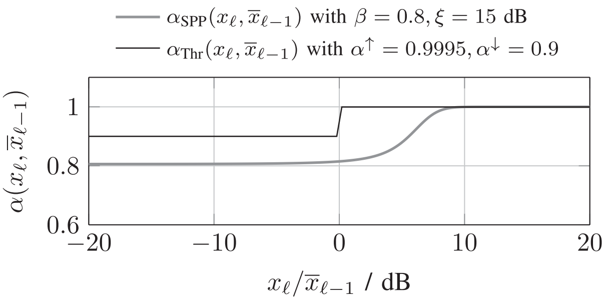

Fig. 2. Adaptive smoothing functions $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$, [5, Section 14.1.3] and $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$, [6] as functions of the *a posteriori* SNR $x_\ell/\overline{x}_{\ell-1} = |X[\ell]|^2/\hat{\sigma}_d^2[\ell-1]$.

the recursive smoother x_ℓ in Fig. 1 is given by $|X[\ell]|^2$ while the output $\overline{x}_\ell$ resembles an estimate of the noise PSD $\hat{\sigma}_d^2[\ell]$.

B. Two Different Smoothing Factors Based on Thresholding

In [5, Section 14.1.3], a simple approach for estimating the background noise PSD from a noisy periodogram has been proposed. Based on a threshold value, one out of two fixed smoothing constants is selected. A larger smoothing constant is used if the input periodogram is larger than the noise PSD which has been estimated for the previous frame. For the other case, a smaller smoothing constant is used. In other words, the tracking speed is reduced if the *a posteriori* SNR $|X[\ell]|^2/\hat{\sigma}_d^2[\ell-1]$ is larger than one. The goal is to reduce the speech leakage if the speech signal is likely to be present. The adaptive smoothing function is given by

$$\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1}) = \begin{cases} \alpha^{\uparrow}, & \text{if } x_\ell/\overline{x}_{\ell-1} > 1 \\ \alpha^{\downarrow}, & \text{otherwise.} \end{cases} \tag{18}$$

Both, $\alpha^{\uparrow}$ and $\alpha^{\downarrow}$ are fixed smoothing constants chosen between zero and one where $\alpha^{\uparrow}$ is chosen larger than $\alpha^{\downarrow}$.

Under the assumption that x_ℓ is exponentially distributed, the analytic solution to (10) is given in Appendix A. For the self-similarity optimization described in Section IV, analytic solutions to the inverse function $h^{-1}(\cdot)$ and the derivative $h'(\cdot)$ are given in Appendix B. A sketch of $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ is given in Fig. 2 for the parameter values $\alpha^{\uparrow} = 0.9995$ and $\alpha^{\downarrow} = 0.9$ proposed in [5].

C. Speech Presence Probability Based Noise PSD Estimation

The noise PSD estimator described in [6] employs an estimate of the SPP to avoid speech leakage. Even though the noise PSD estimator has not been explicitly derived as an adaptive smoothing factor, we show here that the algorithm can be rewritten as such a function. We distinguish between the speech presence hypothesis H_1, i.e., $X[\ell] = S[\ell] + D[\ell]$ and the speech absence hypothesis H_0, i.e., $X[\ell] = D[\ell]$. With Bayes' theorem and using the assumption that the complex Fourier coefficients follow a circular complex Gaussian distribution, the SPP can be derived as (e.g. [6])

$$P(H_1|X[\ell]) = \left(1 + (1+\xi)\exp\left(-\frac{|X[\ell]|^2}{\hat{\sigma}_d^2[\ell-1]}\frac{\xi}{1+\xi}\right)\right)^{-1}. \tag{19}$$

This result is obtained under the assumption that the prior probabilities $P(H_1)$ and $P(H_0)$ are the same. Here, ξ is the SNR expected for time-frequency points where speech is present. In [6], it is regarded as a fixed constant and is not adaptively changed over time. As proposed in [17], the value is optimized such that the Bayesian risk, i.e., the misclassification between speech absence and presence, is minimized. The SPP is used to estimate the noise periodogram as

$$|\hat{D}[\ell]|^2 = \{1 - P(H_1|X[\ell])\}|X[\ell]|^2 + P(H_1|X[\ell])\hat{\sigma}_d^2[\ell]. \tag{20}$$

The background noise PSD is estimated by smoothing the noise periodogram over time using a first-order recursive filter as

$$\hat{\sigma}_d^2[\ell] = (1-\beta)|\hat{D}[\ell]|^2 + \beta\hat{\sigma}_d^2[\ell-1]. \tag{21}$$

Here, $0 \leq \beta \leq 1$ denotes a fixed smoothing constant. The noise PSD is tracked over time by repeating the computations in (19), (20), and (21) for each frame.

By combining (19), (20) and (21), the SPP based noise PSD estimator can be described as an adaptive smoothing function as

$$\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1}) = \beta + \frac{1-\beta}{1 + (1+\xi)e^{-x_\ell\xi/[\overline{x}_{\ell-1}(1+\xi)]}}. \tag{22}$$

The behavior of the adaptive smoothing function is similar to the one proposed by [5, Section 14.1.3] in that the function approaches one for large *a posteriori* SNRs and is close to the fixed smoothing constant β if the *a posteriori* SNR is close to zero.

Also here, analytic solutions are given in Appendix A and Appendix B for the iterative estimation method and the self-similarity optimization, respectively. The function $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ is sketched in Fig. 2 where the parameter values $\beta = 0.8$ and $\xi = 15\,\text{dB}$ proposed in [6] have been employed.

VI. Bias Compensation for Noise PSD Estimation

In Section II-C, a bias compensation method has been presented that can be employed to compensate for the bias caused by adaptive smoothing. However, the composition of the input signal x_ℓ, i.e., whether it contains speech, noise or both, is not taken into consideration. Hence, regarding the application of noise PSD estimation, this correction may overcompensate for the bias in speech presence. To prevent such overcompensations, a time-varying correction factor is derived in this section.

For noise PSD estimation, the input signal comprises two components, namely speech and noise. If speech is present and assumed to be uncorrelated to the noise component, the expected value $\mathbb{E}\{x_\ell\}$ is equal to the sum of the speech PSD $\sigma_s^2[\ell]$ and the noise PSD $\sigma_d^2[\ell]$. If adaptive smoothing is employed on such a noisy signal, with (7) the mean of the filter output converges towards

$$\mathbb{E}\{\overline{x}_\ell\} = \frac{\mathbb{E}\{x_\ell\}}{c} = \frac{\sigma_s^2[\ell] + \sigma_d^2[\ell]}{c}. \tag{23}$$

Applying the fixed factor c removes the bias of the filter output

Algorithm 3: Proposed algorithm for bias compensation.

```
$c$ is obtained using Algorithm 1 or Algorithm 2.
Initialize algorithm: $\overline{x}_0 \leftarrow x_0$.
Compensate bias: $\breve{\overline{x}}_0 \leftarrow G[\ell]\overline{x}_0$.
for all remaining observations $x_\ell$ do
    Perform smoothing:
    $\overline{x}_\ell = [1 - \alpha(x_\ell, \overline{x}_{\ell-1})]x_\ell + \alpha(x_\ell, \overline{x}_{\ell-1})\overline{x}_{\ell-1}$.
    Compensate bias: $\breve{\overline{x}}_\ell = G[\ell]\overline{x}_\ell$.
end for
```

mean $\mathbb{E}\{\overline{x}_\ell\}$ from the input mean $\mathbb{E}\{x_\ell\}$. As a consequence, the output converges towards the noisy PSD, i.e., $\sigma_s^2[\ell] + \sigma_d^2[\ell]$, but not towards the noise PSD. The rate of convergence depends on the additional inertia imposed by the adaptive smoothing factor $\alpha(x_\ell, \overline{x}_{\ell-1})$ which is increased in speech presence by the considered noise PSD estimators [5], [6]. Still, applying c directly may potentially overestimate the noise PSD $\hat{\sigma}_d^2[\ell]$ which, as a consequence, may result in speech distortions in the speech enhancement context. To take the speech energy into account, we propose to modify the correction such that the filter output is corrected towards the noise PSD $\sigma_d^2[\ell]$. For this, a time-varying correction term $G[\ell]$ is introduced which is set such that $G[\ell]\mathbb{E}\{\overline{x}_\ell\} = \sigma_d^2[\ell]$ holds. With (23), $G[\ell]$ can be derived as follows:

$$G[\ell]\mathbb{E}\{\overline{x}_\ell\} = G[\ell]\frac{\sigma_s^2[\ell] + \sigma_d^2[\ell]}{c} \overset{!}{=} \sigma_d^2[\ell] \tag{24}$$

which can be rearranged to

$$G[\ell] = c\frac{\sigma_d^2[\ell]}{\sigma_s^2[\ell] + \sigma_d^2[\ell]}. \tag{25}$$

The time-varying term $G[\ell]$ can be split into the fixed correction factor c and a Wiener-like term $\sigma_d^2[\ell]/(\sigma_s^2[\ell] + \sigma_d^2[\ell])$. Consequently, the fixed correction factor c is reduced such that overestimations in speech presence are avoided. In Section VII, we discuss how the speech and the noise PSD in (25) can be estimated in practical applications.

The proposed bias correction is summarized in Algorithm 3 where $\breve{\overline{x}}_\ell$ denotes the corrected filter output. The additional computational complexity of the proposed correction is given by the computation of the complete correction term $G[\ell]$ and its application. As discussed in Section II-C, it is possible to determine the factor c before the processing starts. Thus, the additional computational cost for the bias correction can be considered low.

VII. Evaluation

In the first part of this section, we verify that the fixed correction factor c estimated with the methods described in Section III and Section IV matches the true underlying bias. For this, we use Monte-Carlo simulations where the input signal consists of artificially generated uncorrelated noise samples that follow an exponential distribution. These experiments also give insights into how large the bias in the considered noise PSD estimators is. Further, we also include an analysis on how signal correlations affect the bias.

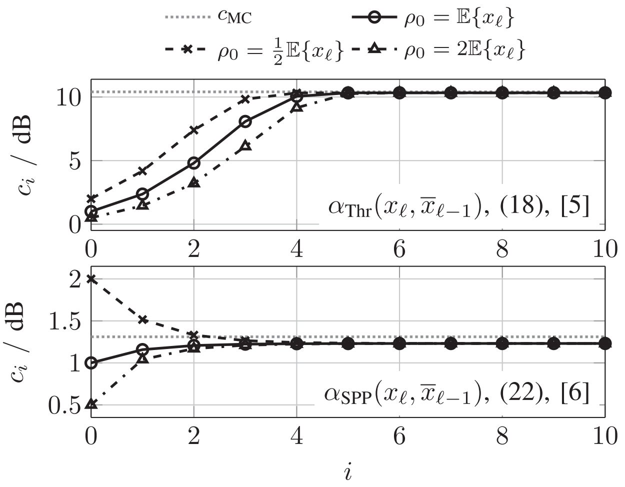

Fig. 3. Bias correction factor $c_i = \mathbb{E}\{x_\ell\}/\rho_i$ computed for each iteration step in Algorithm 1 given the adaptive functions used in [5], [6] and the true bias correction term c_{MC} obtained from Monte-Carlo simulations with 10^6 realizations.

In the second part of this section, the behavior of adaptive smoothing is analyzed in a speech enhancement context using real world signals. We show that correcting the bias leads to an improved estimation of the noise PSD in terms of the log-error distortion measure [8] and also in an improved or similar speech quality as predicted by PESQ [9].

Within our evaluation, the noise PSD estimators given in (18) and (22) are used. In the evaluation, we mainly focus on the default parameters which were proposed in the literature [5], [6]. In accordance with [5, Section 14.1.3], $\alpha^{\uparrow}$ and $\alpha^{\downarrow}$ are set to 0.9995 and 0.9, respectively in (18). In accordance with [6], for the SPP based noise estimator, ξ is set to 15 dB while a value of 0.8 is used for the fixed smoothing constant β in (21).

A. Verification of the Estimation Methods

Here, we analyze how well the proposed methods proposed in Section III and Section IV determine the bias. To obtain the ground-truth, we use Monte-Carlo simulations. For this, 10^6 random numbers x_ℓ are generated that are independently sampled and follow an exponential distribution (17) with fixed parameter $\mu = \mathbb{E}\{x_\ell\}$. The generated random numbers are employed as the input signal of the respective adaptive smoothing filters. As the evaluated algorithms preserve the ergodicity and stationarity of the filter input, the expected value $\mathbb{E}\{\overline{x}_\ell\}$ can be estimated by computing the temporal average of the filter output. With this, a Monte-Carlo estimate of the fixed correction factor $c = \mathbb{E}\{x_\ell\}/\mathbb{E}\{\overline{x}_\ell\}$ is obtained.

First, the iterative procedure described in Section III is covered. Fig. 3 shows the estimated fixed correction factor $c_i = \mathbb{E}\{x_\ell\}/\rho_i$, i.e., the outcome for each iteration step of Algorithm 1. The initial ρ_0 is set to three different values to

TABLE I
CORRECTION FACTOR $c = \mathbb{E}\{x_\ell\}/\mathbb{E}\{\overline{x}_\ell\}$ FOR THE ADAPTIVE SMOOTHING FUNCTIONS IN (18) AND (22) WITHOUT REPLACEMENT OF $\overline{x}_{\ell-1}$

Smoothing factor	Monte-Carlo	Section III/Algorithm 1	Section IV/Algorithm 2
α_{Thr}, (18), [5]	10.18 dB	10.14 dB	10.18 dB
α_{SPP}, (22), [6]	1.17 dB	0.90 dB	1.10 dB

show that the iteration converges to the same value. Additionally, the true correction factor obtained from Monte-Carlo simulations is included. The results show that the iteration converges for all considered smoothing functions after 10 to 15 steps and that the value obtained after convergence is independent of the initial condition ρ_0. For the parameters of the adaptive smoothing functions given in [5], [6], the iteratively determined bias corresponds well with the Monte-Carlo simulations. For the smoothing $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ proposed in [5, Section 14.1.3], the iteratively determined fixed correction factor c is nearly identical to the ground truth obtained from Monte-Carlo simulations while for the SPP based smoothing, i.e., $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$, the bias is underestimated by 0.27 dB (see Table I). This deviation from the correct result is because $\overline{x}_{\ell-1}$ is replaced by a fixed constant ρ, and is not considered as a random variable.

The second method proposed for estimating the bias is described in Section IV. Here, a model PDF $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ is required for the optimization. It is known that after recursive smoothing, an exponentially distributed random process approximately follows a χ^2 distribution with an increased shape parameter [18], [19]. The shape of the resulting PDF can also be approximated by a generalized Gamma distribution or a log-normal distribution. In our experiments, we obtained the best results using the log-normal distribution which is consequently employed in the evaluations. The PDF is given by

$$f(x) = \frac{1}{x\sqrt{2\pi\sigma_{\log}^2}} \exp\left(-\frac{(\log(x) - \mu_{\log})^2}{2\sigma_{\log}^2}\right). \qquad (26)$$

It assumes that the PDF that results after taking the logarithm of the random variable x is a normal distribution. Consequently, $\mu_{\log}$ and $\sigma_{\log}^2$ denote the mean and the variance of the normal distribution in the logarithmic domain, respectively. The mean of this distribution can be computed using its parameters as

$$m(\mu_{\log}, \sigma_{\log}^2) = \exp\left(\mu_{\log} + \frac{\sigma_{\log}^2}{2}\right). \qquad (27)$$

For the minimization of the cost function given in (14), we use the downhill simplex method proposed by [15].

Fig. 4 shows the PDF of the model $\tilde{f}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ with optimized parameters $\hat{\boldsymbol{\theta}}$, which are determined using the method described in Section IV, and the PDF $\tilde{g}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ which is the PDF that results after computing (11) with the optimized parameters $\hat{\boldsymbol{\theta}}$. Finally, the plots also include an estimate of the true PDF of the filter output that has been estimated from Monte-Carlo simulations. Though slight deviations between the true PDF and the optimized log-normal distribution can be observed, the optimized model PDF $\tilde{f}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ approximates the distribution of the filter output reasonably well. Furthermore, Fig. 4 shows that the optimized model distribution $\tilde{f}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ and the marginalized PDF $\tilde{g}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ are nearly identical from which we follow that our approach to finding the bias in Section IV is reasonable.

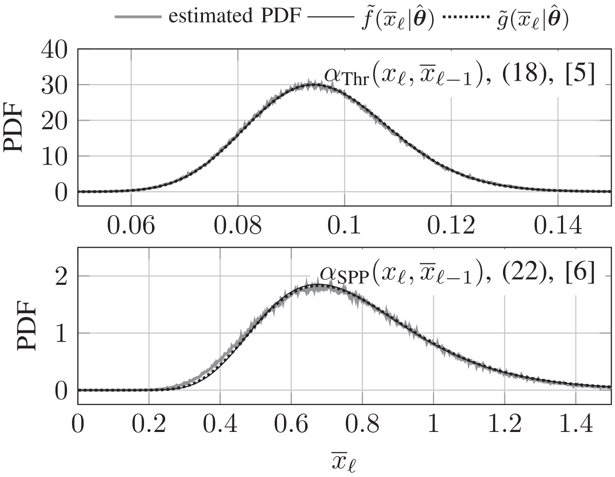

Fig. 4. Shape of the fitted model distribution $\tilde{f}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$, the marginalized distribution $\tilde{g}(\overline{x}_\ell|\hat{\boldsymbol{\theta}})$ obtained by using the optimized model in (11), and the true PDF of the filter output obtained from Monte-Carlo simulations with 10^6 samples for the smoothing factors used in [5], [6].

In Table I, the Monte-Carlo ground-truth of the correction factor c is given along with the estimates of the iterative method of Section III and the self-similarity optimization of Section IV. It can be seen that the self-similarity optimization of Section IV outperforms the iterative method of Section III. Using the self-similarity optimization, for the smoothing with $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ the ground-truth is matched, for the SPP-based smoothing the difference to the ground-truth is only 0.07 dB.

Also note that the bias reported obtained for the SPP based estimation method is only 1.17 dB and, thus, rather small. In contrast, the method in [5, Section 14.1.3] yields a bias of 10.2 dB which is rather large. The reason for this appears to be the choice of the parameter $\alpha^{\uparrow}$. As it is very close to one, the adaptive smoothing is forced to considerably smaller values resulting in the observed bias. Further, this result only covers the case where only noise is present. In the presence of speech, the underestimation is less severe as shown in Section VII-B.

From further experiments we conclude that our proposed Algorithms 1 and 2 work also well for other choices of the parameters $\alpha^{\uparrow}$, $\alpha^{\downarrow}$, β, and ξ. Considering $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ and both algorithms, the deviation of the estimated bias from the true bias is smaller than 1 dB for a wide range of combinations of $\alpha^{\uparrow}$ and $\alpha^{\downarrow}$. Determining the bias for $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ is, however, more challenging. Still, a low deviation of 1 dB from the true mean is obtained for the parameter ranges $0.4 \le \beta \le 0.9$ and $7.5\,\text{dB} \le \xi \le 20\,\text{dB}$ for Algorithm 1 and Algorithm 2. In general, the estimation method proposed in Algorithm 2 has the potential to estimate the bias with very high accuracy as no

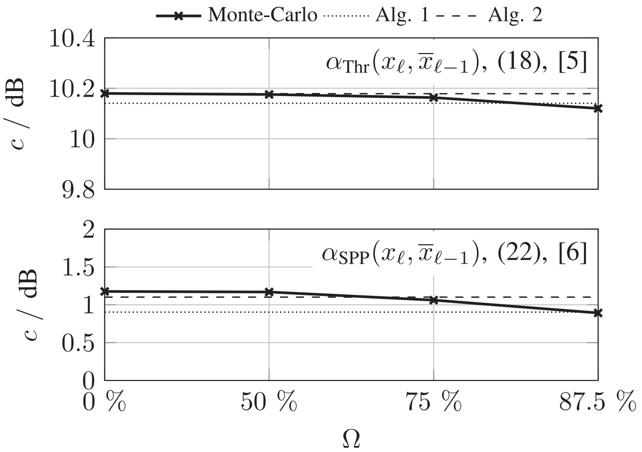

Fig. 5. Correction factor c determined using Monte-Carlo simulations on white noise for different overlaps Ω in the STFT domain with respect to the adaptive smoothing functions used in [5], [6]. Additionally shown: the correction factors reported in Table I.

approximations were used in the derivations. For the practical application, however, an appropriate model PDF $\tilde{f}(\overline{x}_\ell|\boldsymbol{\theta})$ has to be employed and the numerical optimization may converge to local optima leading to unsatisfactory results. In contrast to that, the estimation method in Algorithm 1 is more robust but results only in approximate estimates of the bias due to the used approximations used in the derivation.

Finally, we analyze how the fixed correction factor c is influenced if the samples of the input signal x_ℓ are correlated over time, e.g., due to the overlap in the STFT framework. For this, Monte-Carlo simulations are employed again. Also here, $\mathbb{E}\{\overline{x}_\ell\}$ can be estimated using temporal averaging, as no further restrictions have to be imposed on the random process except for ergodicity which is also fulfilled for correlated input samples. Under the assumption that the sampling rate is 16 kHz, we generate a white Gaussian noise signal with a length of 360 s in the time-domain. After that, we transform the signal to the STFT domain where a Hann-window is employed. The frame and window lengths are set to 32 ms. These STFT parameters are chosen because they allow the results to be easily related to typical single-channel speech enhancement frameworks, e.g., [1], [3], [6]. For this experimental design, the results are also valid if shorter or longer window lengths are used or a different underlying sampling rate is assumed.

To obtain Fourier coefficients with different degrees of correlation, we vary the overlap Ω of the STFT analysis frames, where $\Omega = (\text{frame length} - \text{frame shift})/\text{frame length}$. The adaptive smoothing functions are applied to the magnitude squared coefficients in each frequency band which can be assumed to follow an exponential distribution (17). Finally, the mean over all time-frequency points is computed, where the 0 Hz bin and the Nyquist bin are omitted because the assumption that the coefficients follow an exponential distribution is not fulfilled here. Additionally, we leave out the first 500 frames to account for the adaptation of the adaptive smoothing filters. In Fig. 5, we show the fixed correction factor c as a function of the overlap Ω. In general, it is observed that the bias becomes smaller with increasing overlap — and, thus, also with an increasing amount of correlation. For $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$, the bias is reduced by 0.06 dB in absolute value if the overlap is increased from 0% to 87.5%. Correspondingly, the correlation has a negligible influence on the absolute bias of 10.2 dB. For the SPP based smoothing $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$, the bias is reduced by 0.29 dB for the same increase of correlation. As the absolute bias for $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ is with 1.2 dB much smaller than the bias of $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$, this difference indicates that the influence of the correlation is much stronger here. Thus, the higher overlap leads to a notable reduction of the absolute bias. However, for the typical choice of 50% overlap, the bias hardly changes. As a consequence, the proposed correction methods are directly applicable in practice.

B. Applications to Speech-Enhancement

In this section, we consider the practical implications of the bias caused by adaptive smoothing for noise PSD estimation in a speech enhancement framework. We show that the logarithmic estimation error [8] between the true and the resulting noise PSD is reduced if the bias is corrected. Additionally, we use PESQ scores [9] to give an instrumental prediction of the change in signal quality. Even though PESQ has been developed for the evaluation of speech coding algorithms, it has been shown that it also correlates with the quality of enhanced speech [20]. We show that the log-error distortion and also PESQ scores can be improved for the noise PSD estimators proposed in [5], [6]. For the log-error distortion, we additionally consider a special case where noise only signals are used as input.

For the evaluation, we employ a variety of synthetic and natural noise types. Among these noise types are a pink and a babble noise taken from the Noisex-92 database [21]. Additionally, a traffic noise is employed which comprises an acoustic scene with passing cars. For the experiments that include speech, we use 1120 sentences from the TIMIT corpus [22]. The sentences are corrupted at SNRs ranging from -10 dB to 30 dB in 5 dB steps. Each sentence is embedded in a different segment of the respective background noise. All signals have a sampling rate of 16 kHz.

The speech enhancement framework, in which the considered noise PSD estimators [5], [6] are embedded, operates in the STFT domain. For this, a frame length of 32 ms with 50% overlap is used. This parameter combination is often used for speech enhancement, e.g., [1], [3], [6], as speech signals are assumed to be stationary only for a short time period similar to the chosen frame length [23, Section 5.10]. Further, a square-root Hann window is employed for spectral analysis. For estimating the *a priori* SNR, the decision-directed approach with a smoothing factor of 0.98 is used [1]. The clean speech signal is estimated using the Wiener filter where a lower limit of -12 dB is enforced. For resynthesizing the signal, again, a square-root Hann window is employed.

The time-varying correction term $G[\ell]$ has to be determined at the beginning of a new frame ℓ. At this point, there is no updated estimate of the speech PSD $\hat{\sigma}_s^2[\ell]$ and the noise PSD $\hat{\sigma}_d^2[\ell]$ available. Thus, to determine the time-varying correction term $G[\ell]$, we employ the estimated noise PSD from the previous frame $\hat{\sigma}_d^2[\ell-1]$ while the speech PSD is estimated using the decision-directed approach [1]. Also for the decision-directed approach, the estimated noise PSD of the current frame $\hat{\sigma}_d^2[\ell]$ is replaced by the estimate of the previous one $\hat{\sigma}_d^2[\ell-1]$. We consider only the correction parameters obtained by Algorithm 2 as both methods yield similar values for c such that for the considered practical application very similar outcomes would be obtained. We use the values for c obtained using Algorithm 2 as it performs slightly better than Algorithm 1. Finally, to avoid stagnations of the noise PSD estimation which may be caused by the time-varying correction factor $G[\ell]$, we apply a lower limit to $G[\ell]$ which is set to -20 dB.

Similar to [6], we use a separated version of the log-error distortion which is computed for each speech signal. The measure is split into an overestimation error and an underestimation error such that the equation

$$\text{LogErr} = \text{LogErr}_{\uparrow} + \text{LogErr}_{\downarrow} \qquad (28)$$

is fulfilled. Here, $\text{LogErr}_{\uparrow}$ and $\text{LogErr}_{\downarrow}$ denote the contributions of the overestimation and underestimation to the total log-error distortion, respectively. These quantities are given by

$$\text{LogErr}_{\uparrow} = \frac{10}{KL}\sum_{\ell=0}^{L-1}\sum_{k=0}^{K-1}\max\left(0, \log_{10}\frac{\hat{\sigma}_d^2[k,\ell]}{\sigma_d^2[k,\ell]}\right), \qquad (29)$$

$$\text{LogErr}_{\downarrow} = \frac{-10}{KL}\sum_{\ell=0}^{L-1}\sum_{k=0}^{K-1}\min\left(0, \log_{10}\frac{\hat{\sigma}_d^2[k,\ell]}{\sigma_d^2[k,\ell]}\right). \qquad (30)$$

In this equation, K denotes the number of Fourier coefficients which is equal to 512 in this evaluation due to the sampling frequency of the signals and the chosen analysis window length. Further, L is the number of frames. Only frames after a five seconds initialization period, which only includes noise, are considered in the evaluation. Thus, $\ell = 0$ can be considered the first frame after the initialization phase and L the number of remaining frames after the initialization. During this the initialization phase, the noise PSD estimators can adapt to the background noise. The goal is to exclude initialization artifacts from the evaluation which may result in an erroneous estimate of the performance. Even though in real applications, such an initialization period is not available, this poses only a minor problem as the algorithms recover from an erroneous initializations after a short processing time, e.g., during speech pauses. As the correction factors were determined based on the assumption that the periodogram is exponentially distributed, we exclude the coefficient at 0 Hz and the Nyquist frequency also here. The measure is computed for each speech signal separately and averaged over all speech signals afterwards. For the noise only case, the log-error distortion is computed using a long excerpt of about four minutes from the respective noise signal.

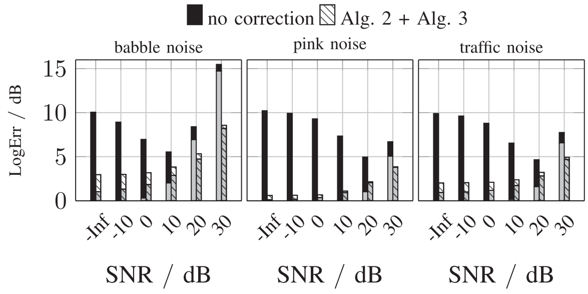

Fig. 6. Log-error distortion of the adaptive smoothing function $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ described in (18), [5, Section 14.1.3] with and without the proposed correction method for speech in noise at different SNRs. The lower part (gray) of the bars represents the overestimation $\text{LogErr}_{\uparrow}$, whereas the upper part (black / white) is the underestimation $\text{LogErr}_{\downarrow}$.

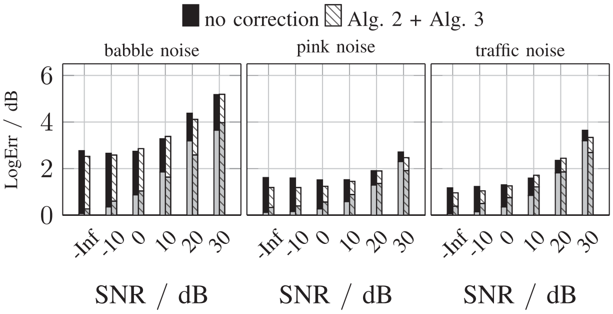

Fig. 7. Same as Fig. 6 for the adaptive smoothing function $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ described in (22), [6].

The reference noise PSD $\sigma_d^2[k,\ell]$ in the log-error distortion is a statistical quantity whose value has to be obtained from the noise signal. For the stationary pink noise, the reference noise PSD is estimated by averaging the periodogram over all frames. This procedure, however, leads to an unsatisfactory result for the nonstationary noise signals as the temporal changes are not captured. Here, a slightly smoothed version of the noise periodogram is used as reference noise PSD $\sigma_d^2[k,\ell]$. On the one hand, the smoothing is applied to reduce the variance in comparison to the direct usage of the noise periodogram. On the other hand, the amount of smoothing is kept at a low level to track changes in the background noise. For this, we employ first-order recursive smoothing with a fixed smoothing constant $\alpha = 0.73$. This choice corresponds to an equivalent moving average smoothing with an rectangular window of 50 ms which yielded a satisfying compromise.

The results for the two noise PSD estimators are shown in Figs. 6 and 7, respectively. Here, an SNR of -Inf denotes the noise only case. For the adaptive smoothing function $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ proposed in [5, Section 14.1.3], the results in Fig. 6 show that the uncorrected version of the noise PSD estimator tends to underestimate the background noise PSD in low SNR regions while it overestimates the noise PSD for high SNRs. The observed overestimation at high SNRs is caused by the fact that this estimator always allows to track the input periodogram albeit slowly even if the *a posteriori* SNR is high. Thus, the speech leakage, which is reflected in the overestimation, increases with increasing SNR. The underestimation at low SNRs is mainly caused by the adaptive smoothing. If the proposed correction is applied, the noise PSD log-error distortion can be considerably reduced for all considered SNRs and noise types. As the fixed correction factor c required for this noise PSD estimator is rather large, the total estimation error is often dominated by the overestimation if the correction is applied. The total log-error distortion, however, is in general smaller.

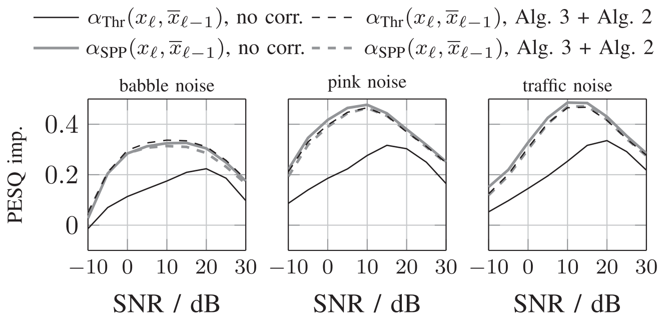

Fig. 8. PESQ improvement scores for a simple speech enhancement framework where the adaptive smoothing functions $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ and $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ are used as noise PSD estimators with and without the correction proposed in Algorithm 3. The fixed correction factor c was estimated using Algorithm 2.

Especially, if either noise or speech is dominant, i.e., for low SNRs and high SNRs, lower estimation errors are obtained.

Similar tendencies are also observed for the SPP based noise estimator $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$ as shown in Fig. 7. For both cases, i.e., with and without correction, the overestimation increases also for this noise PSD estimator with increasing SNR. For an SNR range around 0 dB and 10 dB, the proposed correction increases the log-error distortion slightly. For high SNRs and low SNRs, however, a slight reduction of the log-error distortion is observed. In general, the benefits of the correction are expected to be smaller as the bias of this algorithm is rather low as shown in Table I.

Fig. 8 shows the PESQ improvement scores which are obtained if the considered adaptive smoothing functions are used as noise PSD estimators in a simple enhancement scheme. Again, the adaptive smoothing functions are employed with and without correction to show the change in performance. For $\alpha_{\text{SPP}}(x_\ell, \overline{x}_{\ell-1})$, the corrected and the uncorrected version of the noise PSD lead to nearly the exact same result. In general, the measure indicates a slight reduction of the quality if the proposed correction is applied. Considering the log-error distortions in Fig. 7, the result is not unexpected as the differences between the corrected and uncorrected version are small. Contrarily, the PESQ scores can be considerably improved for the smoothing function $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$, [5, Section 14.1.3]. After applying the correction, the PESQ scores are increased by up to 0.2 points where the largest gains are obtained for SNRs between 0 dB and 10 dB. The predicted quality of the corrected version of $\alpha_{\text{Thr}}(x_\ell, \overline{x}_{\ell-1})$ is comparable to the SPP based noise PSD estimator. These improvements can be attributed to the reduction of the strong underestimation in low SNR regions and the prevention of overestimation in speech presence. These results are also confirmed in informal listening tests.

VIII. Conclusions

In this paper, we analyzed the bias of adaptive first-order recursive smoothing filters which play a central role, e.g., in the noise PSD estimators presented in [5], [6]. From our analysis, it followed that due to the used adaptive smoothing, both algorithms generally underestimate the noise PSD. We could show that the bias is scale-invariant and that the bias from the input signal mean $\mathbb{E}\{x_\ell\}$ caused by adaptive smoothing can be compensated using a single fixed correction factor c. For the application of noise PSD estimation, we extended the correction method which resulted in a time-varying correction factor to avoid overestimation by accounting for the speech energy. This led to the proposed correction method shown in Algorithm 3. The fixed correction factor c can be determined using the proposed Algorithms 1 and 2. Algorithm 1 employs an iterative method which is based on the analytically solvable case where the adaptive smoothing factor does not depend on the previous filter output $\overline{x}_{\ell-1}$. Algorithm 2 determines the factor c by maximizing the self-similarity of a model PDF with respect to the transition density $f(\overline{x}_\ell|\overline{x}_{\ell-1})$. In the evaluation, we could demonstrate that Algorithm 2 estimates the correction factor c with a higher accuracy than the iterative method, i.e., Algorithm 1. If the estimation error of the adaptive smoothing filter is sufficiently large, the proposed correction method yields considerable improvements in terms of the log-error distortion and PESQ.

Appendix A
Analytical Results for the Iterative Bias Estimation

Here, we present the analytic expression of the expected value in (9) that are obtained for the considered adaptive smoothing functions in (18) and (22). Here, we employ the simplification described in Section III again, i.e., $\overline{x}_{\ell-1}$ is replaced by the deterministic ρ. The following equations were derived under the assumption that x_ℓ follows an exponential distribution (17).

For the noise PSD estimator proposed in [5, Section 14.1.3], the expected value $\mathbb{E}\{\overline{x}_\ell\}$, i.e., the solution to (9) given (18), results in

$$\mathbb{E}\{\overline{x}_\ell\} = \mu \frac{(\alpha^{\downarrow} - 1)\exp(\lambda) + (\alpha^{\uparrow} - \alpha^{\downarrow})(1+\lambda)}{(\alpha^{\downarrow} - 1)\exp(\lambda) + \alpha^{\uparrow} - \alpha^{\downarrow}}, \tag{31}$$

with $\lambda = \rho/\mu$.

The expected value $\mathbb{E}\{\overline{x}_\ell\}$ for the expression in (22) can be derived using the property of the geometric series [24, 1.112.1] and the analytic continuation property of the hypergeometric series [24, 9.130]. The result is

$$\mathbb{E}\{\overline{x}_\ell\} = \mu \frac{1 - {}_3F_2\left[1, \zeta, \zeta; \zeta+1, \zeta+1; -(1+\xi)\right]}{1 - {}_2F_1\left[1, \zeta; \zeta+1; -(1+\xi)\right]}, \tag{32}$$

where ${}_pF_q$ is the generalized hypergeometric function with

$$\zeta = \lambda \frac{\xi + 1}{\xi}. \tag{33}$$

Appendix B
Analytic Solutions for the Self-Similarity Optimization

Here, we derive the analytic expressions of the inverse function $h^{-1}(\cdot)$ and the derivative $h'(\cdot)$ for the considered adaptive smoothing functions. Using these results, the conditional PDF $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ can be obtained with (15). For the derivations, we assume that $h(\cdot)$ is given by the expression in (1).

For the adaptive smoothing function $\alpha_{\mathrm{Thr}}(x_\ell, \overline{x}_{\ell-1})$ in (18), [5, Section 14.1.3], the existence of an inverse $h^{-1}(\cdot)$ depends on the relationship between the updated filter output $\overline{x}_\ell$ and the previous filter output $\overline{x}_{\ell-1}$. Under the assumption that $x_\ell \geq 0$, the adaptive smoothing given in (1) can be inverted if $\alpha^{\downarrow}\overline{x}_{\ell-1} \leq \overline{x}_\ell \leq \overline{x}_{\ell-1}$ or if $\overline{x}_\ell > \overline{x}_{\ell-1}$. For the first condition, the inverse is given by

$$h_1^{-1}(\overline{x}_\ell) = \frac{\overline{x}_\ell - \alpha^{\downarrow}\overline{x}_{\ell-1}}{1-\alpha^{\downarrow}} \tag{34}$$

and the denominator of (15) is given by

$$h'(h_1^{-1}(\overline{x}_\ell)) = 1 - \alpha^{\downarrow}. \tag{35}$$

For the case that $\overline{x}_\ell > \overline{x}_{\ell-1}$, the filter function in (1) can be inverted as

$$h_2^{-1}(\overline{x}_\ell) = \frac{\overline{x}_\ell - \alpha^{\uparrow}\overline{x}_{\ell-1}}{1-\alpha^{\uparrow}} \tag{36}$$

where the denominator of (15) is

$$h'(h_2^{-1}(\overline{x}_\ell)) = 1 - \alpha^{\uparrow}. \tag{37}$$

For some values of $\overline{x}_\ell$ none of the conditions applies so that $M = 0$. For these $\overline{x}_\ell$, it follows that also $f(\overline{x}_\ell|\overline{x}_{\ell-1}) = 0$.

If $\alpha_{\mathrm{SPP}}(x_\ell, \overline{x}_{\ell-1})$ from (22), [6] is employed in (1), the filter equation can be inverted if $\overline{x}_{\ell-1}(1+\beta(1+\xi))/(2+\xi) \leq \overline{x}_\ell \leq z\overline{x}_{\ell-1}$ where also the assumption is made that $x_\ell > 0$. In other words, $f(\overline{x}_\ell|\overline{x}_{\ell-1})$ is zero if this condition is not fulfilled. The quantity z is given by

$$z = \tilde{z} + (1-\tilde{z})\left(\beta + \frac{1-\beta}{1+(1+\xi)e^{-\tilde{z}\xi/(1+\xi)}}\right) \tag{38}$$

with

$$\tilde{z} = \frac{\xi+1}{\xi}\left[1 + W_0\left(e^{-1-\xi/(\xi+1)}(\xi+1)\right)\right] + 1. \tag{39}$$

Here, $W_0(\cdot)$ denotes the main branch of the Lambert-W function [25]. This function, together with its second real branch $W_{-1}(\cdot)$, constitutes the inverse of the expression $f(x) = x\exp(x)$ [25]. One inverse function of the filter in (1) with respect to the smoothing function in (22) is

$$h_1^{-1}(\overline{x}_\ell) = -\overline{x}_{\ell-1}\frac{1+\xi}{\xi}W_0(A) + \frac{\overline{x}_\ell - \beta\overline{x}_{\ell-1}}{1-\beta} \tag{40}$$

where A is given by

$$A = \frac{\xi(1-\overline{x}_\ell/\overline{x}_{\ell-1})}{(1-\beta)(1+\xi)^2}\exp\left(\frac{\xi(\overline{x}_\ell - \beta\overline{x}_{\ell-1})}{\overline{x}_{\ell-1}(1-\beta)(1+\xi)}\right). \tag{41}$$

If the first condition holds and, additionally, $\overline{x}_\ell$ fulfills $\overline{x}_\ell > \overline{x}_{\ell-1}$, a second inverse can be found. The result is

$$h_2^{-1}(\overline{x}_\ell) = -\overline{x}_{\ell-1}\frac{1+\xi}{\xi}W_{-1}(A) + \frac{\overline{x}_\ell - \beta\overline{x}_{\ell-1}}{1-\beta}. \tag{42}$$

Note that the conditions for the two inverse functions are not exclusive, i.e., there are values for $\overline{x}_\ell$ where both conditions are fulfilled. For these $\overline{x}_\ell$, the number of piecewise monotonic segments M is two. Finally, the derivative is given by

$$h'(x) = (1-\beta)\left[\frac{\xi B}{(1+(1+\xi)B)^2}\left(1 - \frac{x}{\overline{x}_{\ell-1}}\right) + \left(1 - \frac{1}{1+(1+\xi)B}\right)\right], \tag{43}$$

with

$$B = \exp\left(-\frac{x}{\overline{x}_{\ell-1}}\frac{\xi}{1+\xi}\right). \tag{44}$$

REFERENCES

[1] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, Dec. 1984.

[2] C. Breithaupt, T. Gerkmann, and R. Martin, "A novel a priori SNR estimation approach based on selective cepstro-temporal smoothing," in *Proc. IEEE Int. Acoust., Speech, Signal Process.*, Las Vegas, NV, USA, Apr. 2008, pp. 4897–4900.

[3] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 504–512, Jul. 2001.

[4] F. Heese and P. Vary, "Noise PSD estimation by logarithmic baseline tracing," in *Proc. IEEE Int. . Acoust., Speech, Signal Process.*, Brisbane, Qld, Australia, Apr. 2015, pp. 4405–4409.

[5] E. Hnsler and G. Schmidt, *Acoustic Echo and Noise Control A Practical Approach* (Adaptive and Learning Systems for Signal Processing, Communication and Control). Hoboken, NJ, USA: Wiley, 2004.

[6] T. Gerkmann and R. C. Hendriks, "Noise power estimation based on the probability of speech presence," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoustics.*, New Paltz, NY, USA, 2011, pp. 145–148.

[7] R. Rehr and T. Gerkmann, "On the bias of adaptive First-Order recursive smoothing," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, New Paltz, NY, USA, Oct. 2015, pp. 1–5.

[8] R. C. Hendriks, J. Jensen, and R. Heusdens, "DFT domain subspace based noise tracking for speech enhancement," in *Proc. Conf. Int. Speech Commun. Assoc.*, Antwerp, Belgium, Aug. 2007, pp. 830–833.

[9] "P.862 : Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Jan. 2001. [Online]. Available: http://www.itu.int/rec/T-REC-P.862-200102-I/en

[10] R. Rehr and T. Gerkmann, "Bias correction methods for adaptive recursive smoohting with applications in noise PSD estimation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 206–210.

[11] L. Breiman, *Probability* (Classics in Applied Mathematics). Philadelphia, PA, USA: Soc. Ind. Appl. Math., 1968.

[12] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.

[13] T. Kailath, "The divergence and bhattacharyya distance measures in signal selection," *IEEE Trans. Commun. Technol.*, vol. 15, no. 1, pp. 52–60, Feb. 1967.

[14] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, Mar. 1951.

[15] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, no. 4, pp. 308–313, 1965.

[16] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. (McGraw-Hill Series in Electrical Engineering: Communications and Signal Processing). New York, NY, USA: McGraw-Hill, 2002.

[17] T. Gerkmann, C. Breithaupt, and R. Martin, "Improved a posteriori speech presence probability estimation based on a likelihood ratio with fixed priors," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 5, pp. 910–919, Jul. 2008.

[18] R. Martin, "Bias compensation methods for minimum statistics noise power spectral density estimation," *Signal Process.*, vol. 86, no. 6, pp. 1215–1229, 2006.

[19] T. Gerkmann and R. Martin, “On the statistics of spectral amplitudes after variance reduction by temporal cepstrum smoothing and cepstral nulling,” *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4165–4174, Nov. 2009.

[20] Y. Hu and P. C. Loizou, “Evaluation of objective quality measures for speech enhancement,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 1, pp. 229–238, Jan. 2008.

[21] H. J. M. Steeneken and F. W. M. Geurtsen, “Description of the RSG.10 noise database,” TNO Inst. Perception, Soesterberg, The Netherlands, Tech. Rep. IZF 1988-3, 1988.

[22] J. S. Garofolo *et al.*, “TIMIT Acoustic-Phonetic continuous speech corpus,” Linguistic Data Consortium, Philadelphia, PA, USA, 1993.

[23] P. Vary and R. Martin, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*. Chichester, U.K.: Wiley, 2006.

[24] I. S. Gradshteyn and I. W. Ryzhik, *Table of Integrals, Series, and Products*, 7th ed. D. Zwillinger and V. Moll, Eds. New York, NY, USA: Academic, Feb. 2007.

[25] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, “On the Lambert-W function,” *Adv. Comput. Math.*, vol. 5, no. 1, pp. 329–359, 1996.

Robert Rehr (S’14) received the B.Eng. degree from the Jade Hochschule, Oldenburg, Germany, in 2011, and the M.Sc. degree from the Universität Oldenburg, Oldenburg, in 2013, both in Hearing Technology and Audiology. He is currently working toward the Ph.D. degree. From 2013 to 2016, he was with the Speech Signal Processing Group at the Universität Oldenburg. Since 2017, he has been with the Signal Processing Group at the Universität Hamburg, Hamburg, Germany. His research focuses on speech enhancement using machine-learning-based methods.

Timo Gerkmann (S’08–M’10–SM’15) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering and information sciences from the Ruhr-Universität Bochum, Bochum, Germany, in 2004 and 2010, respectively. In 2005, he spent six months with Siemens Corporate Research, Princeton, NJ, USA. From 2010 to 2011, he was a Postdoctoral Researcher in the Sound and Image Processing Laboratory, Royal Institute of Technology (KTH), Stockholm, Sweden. From 2011 to 2015, he was a Professor of speech signal processing with the Universität Oldenburg, Oldenburg, Germany. From 2015 to 2016, he was the Principal Scientist in Audio & Acoustics, Technicolor Research & Innovation, Hanover, Germany. Since 2016, he has been a Professor of signal processing with the University of Hamburg, Hamburg, Germany. His research interests include digital signal processing algorithms for speech and audio applied to communication devices, hearing instruments, audio–visual media, and human–machine interfaces.