Multichannel Online Dereverberation based on Spectral Magnitude Inverse Filtering

Xiaofei Li, Laurent Girin, Sharon Gannot, Radu Horaud

Multichannel Online Dereverberation Based on Spectral Magnitude Inverse Filtering

Xiaofei Li, Laurent Girin, Sharon Gannot and Radu Horaud

***Abstract*—This paper addresses the problem of multichannel online dereverberation. The proposed method is carried out in the short-time Fourier transform (STFT) domain, and for each frequency band independently. In the STFT domain, the time-domain room impulse response is approximately represented by the convolutive transfer function (CTF). The multichannel CTFs are adaptively identified based on the cross-relation method, and using the recursive least square criterion. Instead of the complex-valued CTF convolution model, we use a nonnegative convolution model between the STFT magnitude of the source signal and the CTF magnitude, which is just a coarse approximation of the former model, but is shown to be more robust against the CTF perturbations. Based on this nonnegative model, we propose an online STFT magnitude inverse filtering method. The inverse filters of the CTF magnitude are formulated based on the multiple-input/output inverse theorem (MINT), and adaptively estimated based on the gradient descent criterion. Finally, the inverse filtering is applied to the STFT magnitude of the microphone signals, obtaining an estimate of the STFT magnitude of the source signal. Experiments regarding both speech enhancement and automatic speech recognition are conducted, which demonstrate that the proposed method can effectively suppress reverberation, even for the difficult case of a moving speaker.**

I. Introduction

This paper addresses the problem of multichannel online dereverberation of speech signals, emitted by either a static or a moving speaker. The objective of dereverberation is to improve speech quality/intelligibility for human listening or for automatic speech recognition (ASR). In the REVERB challenge [1], a number of dereverberation methods were benchmarked, which showed that both speech quality (naturalness, distortion, perceived reverberation, etc.) and ASR performance can be improved by dereverberation, and that larger the number of microphones better the improvement. As for ASR, [2], [3], [4], [5] show that, even for an advanced ASR back-end with multi-condition training to account for the reverberation effect, a standalone dereverberation front-end is still helpful. The influence of reverberation on speech intelligibility was analyzed in [6], [7], [8], [9] for both normal- and hearing-impaired listeners. It was shown that, in office rooms, reverberation alone does not severely degrade speech intelligibility for normal-hearing listeners, while it does for hearing-impaired listeners. Under noisy conditions, reverberation significantly degrades speech intelligibility for both normal- and hearing-impaired listeners. It was shown in [10] that, for normal-hearing listeners, dereverberation indeed improves the tolerance of listeners to noise. Compared to normal-hearing listeners, [11] showed that speech intelligibility for hearing-impaired listeners can be prominently improved by dereverberation . The output of a dereverberation system may include some early reflections, since they deteriorate neither speech quality nor speech intelligibility [12].

Multichannel dereverberation includes the following different techniques. Spectral enhancement techniques [13], [14], [15], which are performed in the short-time Fourier transform (STFT) domain, remove late reverberation by spectral subtraction. To iteratively estimate the room filters and the speech source signal, other techniques minimize a cost function between the microphone signal(s) and a generative model thereof (or equivalently maximize an objective function). The generative model here mainly indicates the convolutive model between the room filters and the source signal, and sometimes the source signal is assumed to be generated by a random process. These techniques are also usually applied in the STFT domain, where the time-domain RIR is represented by a subband convolutive transfer function (CTF). An expectation-maximization (EM) algorithm is used in [16] to maximize the likelihood of the microphone signals. The idea is extended to joint dereverberation and source separation in [17]. In [18], [19], [20], a nonnegative convolution approximation is assumed, namely the STFT magnitude of the microphone signal is approximated by the convolution between the STFT magnitude of the source signal and the CTF magnitude. Based on this nonnegative model, tensor factorization [18], iterative auxiliary functions [19] and iterative multiplicative update [20] are used to minimize the fit cost between the STFT magnitude of the microphone signal and its nonnegative generative model. Inverse filtering techniques aim at inverting the room convolution process and recovering the source signal. Depending on the way inverse filters are estimated, inverse filtering techniques can be classified into two groups:

- Linear prediction based techniques model the convolution with the RIR as an auto-regressive (AR) process. This AR process can be carried out either in the time domain or in the STFT domain. In the linear-predictive multi-input equalization (LIME) algorithm [21], the speech source signal is estimated as the multichannel linear prediction residual, which however is excessively whitened. The whitening effect is then compensated by estimating the average speech characteristics. To avoid such whitening effect, a prediction delay is used in the delayed linear prediction techniques [22], [23]. These techniques only model late reverberation

X. Li and R. Horaud are with INRIA Grenoble Rhône-Alpes, Montbonnot Saint-Martin, France.

L. Girin is with GIPSA-lab and with Univ. Grenoble Alpes, Saint-Martin d'Hères, France.

Sharon Gannot is with Bar Ilan University, Faculty of Engineering, Israel.

This work was supported by the ERC Advanced Grant VHIA #340113.

into the AR process and leave early reflections of the speech signal in the prediction residual. To account for the time-varying characteristics of speech, the statistical model-based approach [23] iteratively estimates the time-varying speech variance and normalizes the linear prediction with this speech variance. This variance-normalized delayed linear prediciton method is also called weighted prediction error (WPE);
- Techniques based on system identification first blindly identify the room filters. Then, the corresponding inverse filters are estimated and applied on the microphone signals to recover the source signal. The cross-relation method [24] is a widely-used system identification method. Inverse filter estimation techniques include the multiple-input/output inverse theorem (MINT) method [25] and some of its variants, such as channel shortening [26] and partial MINT [27]. In [28], [29], the cross-relation method was applied in the STFT domain for CTF estimation. Several variants of subband MINT were proposed based on filter banks [30], [31] or CTF model [32], [33].

For dynamic scenarios with moving speakers or speech turns among speakers, an online dereverberation method is required. Based on the CTF model, an online likelihood maximization method was proposed in [34], [35] using a Kalman filter and an EM algorithm. An online extension of LIME was proposed in [36] using several different adaptive estimation criteria, such as normalized least mean squares (LMS), steepest descent, conjugate gradient and recursive least square (RLS). RLS-based adaptive WPE (AWPE) [3], [37], [38], [39] became a popular online dereverberation method. For example, it is used by the Google Home smart loudspeaker device [2]. In AWPE, the anechoic speech variance is estimated using a spectral subtraction method in [38], and is simply approximated by the microphone speech variance in [37], [3], [39]. In [40], [41], a probabilistic model and a Kalman filter were used to implement the delayed linear prediction method, which can be seen as a generalization of the RLS-based AWPE. A class of adaptive cross-relation methods were proposed in [42] for online system identification, with the adaptive estimation criteria of normalized LMS and multichannel Newton method. Adaptive multichannel equalization methods were proposed in [43], [44] based on time-domain MINT and gradient descent update. These methods reduce the computational complexity of the original MINT, however they were only used for offline multichannel equalization in static scenarios.

In our previous work [29], a blind dereverberation method was proposed in batch mode for static scenarios. This method consists of a blind CTF identification algorithm and a sparse source recovery algorithm. The CTF identification algorithm is based on the cross-relation method. For source recovery, instead of the complex-valued CTF convolution model, we used its nonnegative convolution approximation [18], [19], [20], since the latter was shown to be less sensitive to the CTF perturbations than the former. More precisely, the STFT magnitude of the source signal is recovered by solving a basis pursuit problem that minimizes the ℓ_1-norm of the STFT magnitude of the source signal while constraining the fit cost, between the STFT magnitude of the microphone signals and the nonnegative convolution model, to be below a tolerance.

In the present work, we propose an online dereverberation method. First, we extend the batch formulation of CTF identification in [29] to an adaptive method based on an RLS-like recursive update. The RLS-like method has a better convergence rate than the normalized LMS method used in [42], which is crucial for its application in dynamic scenarios. This adaptive CTF identification is carried out in the complex domain, then the magnitude of the identified CTF is used for online inverse filtering, based on the nonnegative convolution model: the inverse filters of the CTF magnitudes are adaptively estimated and applied to the STFT magnitude of the microphone signals to obtain an estimate of the STFT magnitude of the source signal. The inverse filters estimation is based on the MINT theorem [25]. Due to the use of the nonnegative CTF convolution model, the proposed magnitude MINT is different from the conventional MINT methods, such as [26], [27], [32], mainly in aspect to that multichannel fusion and target response. Following the spirit of normalized LMS, we propose to adaptively update the inverse filters based on a gradient descent method. In summary, the proposed method consists of two novelties i) an online RLS-like CTF identification technique, and ii) an online STFT-magnitude inverse filtering technique. To the best of our knowledge this is the first time such procedures are proposed for online speech dereverberation. Experimental comparison with AWPE shows that the proposed method performs better for the moving speaker case, mainly due to the use of the less sensitive magnitude convolution model.

The remainder of this paper is organized as follows. The adaptive CTF identification is presented in Section II. The online STFT magnitude inverse filtering method is presented in Section III. Experiments with two datasets are presented in Section IV. Section V concludes the work.

II. Online CTF Identification

We consider a system with I channels and one speech source. In the time domain, the i-th microphone signal $x_i(n)$ is

$$x_i(n) = s(n) \star a_i(n) + e_i(n), \quad i = 1, \dots, I \tag{1}$$

where n is the time index, $\star$ denotes convolution, $s(n)$ is the speech source signal, and $a_i(n)$ is the RIR from the speech source to the i-th microphone. The additive noise term $e_i(n)$ will be discarded in the following, since we do not consider noise in this work. In the STFT domain, based on the CTF approximation, we have

$$x_{i,p,k} \approx s_{p,k} \star a_{i,p,k}, \quad i = 1, \dots, I \tag{2}$$

where $x_{i,p,k}$ and $s_{p,k}$ are the STFT coefficients of the corresponding signals, and the CTF $a_{i,p,k}$ is the subband representation of the RIR $a_i(n)$. $p = 1, \dots, P$ denotes the STFT frame index and $k = 0, \dots, N-1$ denotes the frequency index, P is the number of signal frames in a given processed

speech sequence, and N is the STFT frame (window) length. The convolution is executed along the frame index p. The length of the CTF, denoted as Q, is assumed to be identical for all frequency bins and is approximately equal to the length of the corresponding RIR divided by L, where L denotes the STFT frame shift.

A. Batch CTF Identification

In [29], we proposed a batch mode CTF identification method in the complex domain. This method is based on the following cross-relation between channels [24]:

$$x_{i,p,k} \star a_{j,p,k} = s_{p,k} \star a_{i,p,k} \star a_{j,p,k} = x_{j,p,k} \star a_{i,p,k}. \quad (3)$$

However, this equation cannot be directly used. The reason is that, for the oversampling case (i.e. $L < N$), there is a common region with magnitude close to zero in the frequency response of the CTFs for all channels, caused by the non-flat frequency response of the STFT window. This common zero frequency region is problematic for the cross-relation method. It can be alleviated by using critical sampling (i.e. $L = N$), which however leads to a severe frequency aliasing of the signals. To achieve a good trade-off, it was proposed in [29] that the signal STFT coefficients are oversampled to avoid frequency aliasing, but the multichannel CTF coefficients are forced to be critically sampled to avoid the common zero problem. More precisely, the Hamming window[1] is used, and we set $L = N/4$ and $L_f = N$, where L_f denotes the frame step of CTF. Since the channel identification algorithm presented in this section and the inverse filtering algorithm presented in the next section are both applied frequency-wise, hereafter the frequency index k will be omitted for clarity of presentation.

Based on the oversampled CTF $a_{i,p}$, the critically sampled CTF is defined in vector form as $\tilde{\mathbf{a}}_i = [a_{i,0}, a_{i,4}, \ldots, a_{i,4(\tilde{Q}-1)}]^\top$, where $^\top$ denotes matrix/vector transpose and $\tilde{Q} = \lceil Q/4 \rceil$ ($\lceil \cdot \rceil$ is the ceiling function). In accordance with this critically sampled CTF, (2) should be reformulated with critically sampled source STFT coefficients. However, such reformulation of (2) is actually not used. Instead, in the following CTF identification and inverse filtering methods, the filtering process is applied to the microphone signals, thence the STFT coefficients of microphone signals will be critically sampled. From the oversampled STFT coefficients of microphone signals, we define the convolution vector as $\tilde{\mathbf{x}}_{i,p} = [x_{i,p}, x_{i,p-4}, \ldots, x_{i,p-4(\tilde{Q}-1)}]^\top,\ p = 1, \ldots, P$. Note that, when $p < 4(\tilde{Q}-1)+1$, the vector $\tilde{\mathbf{x}}_{i,p}$ is constructed by padding zeros. Then, the cross-relation can be recast as

$$\tilde{\mathbf{x}}_{i,p}^\top \tilde{\mathbf{a}}_j = \tilde{\mathbf{x}}_{j,p}^\top \tilde{\mathbf{a}}_i. \quad (4)$$

This convolution formulation can be interpreted as that $3/4$ of the original oversampled CTF coefficients are forced to be zero. This cross-relation is defined for each microphone pair.

[1]Other commonly used windows, such as Hanning and Sine windows, are also applicable.

To present the cross-relation equation in terms of the CTF of all channels, i.e.

$$\tilde{\mathbf{a}} = [\tilde{\mathbf{a}}_1^\top, \tilde{\mathbf{a}}_2^\top, \ldots, \tilde{\mathbf{a}}_I^\top]^\top, \quad (5)$$

we define:

$$\tilde{\mathbf{x}}_{ij,p} = [\underbrace{0, \ldots, 0}_{(i-1)\tilde{Q}}, \tilde{\mathbf{x}}_{j,p}^\top, \underbrace{0, \ldots, 0}_{(j-i-1)\tilde{Q}}, -\tilde{\mathbf{x}}_{i,p}^\top, \underbrace{0, \ldots, 0}_{(I-j)\tilde{Q}}]^\top,\ j > i. \quad (6)$$

Then the cross-relation (4) can be written as:

$$\tilde{\mathbf{x}}_{ij,p}^\top \tilde{\mathbf{a}} = 0. \quad (7)$$

There is a total of $M = I(I-1)/2$ distinct microphone pairs, indexed by (i, j) with $j > i$. For notational convenience, let $m = 1, \ldots, M$ denote the microphone-pair index. Then let the subscript $_{ij}$ be replaced with $_m$. For the static speaker case, the CTF $\tilde{\mathbf{a}}$ is time-invariant, and can be estimated by solving the following constrained least square problem in batch mode:

$$\min \sum_{p=1}^{P} \sum_{m=1}^{M} |\tilde{\mathbf{x}}_{m,p}^\top \tilde{\mathbf{a}}|^2 \quad \text{s.t. } \mathbf{g}^\top \tilde{\mathbf{a}} = 1, \quad (8)$$

where $|\cdot|$ denotes the (entry-wise) absolute value, and $\mathbf{g}$ is a constant vector

$$\mathbf{g} = [1, \underbrace{0, \ldots, 0}_{\tilde{Q}-1}, 1, \underbrace{0, \ldots, 0}_{\tilde{Q}-1}, \ldots, 1, \underbrace{0, \ldots, 0}_{\tilde{Q}-1}]^\top. \quad (9)$$

Here we constrain the sum of the first entries of the I CTFs to be equal to 1, i.e. $\sum_{i=1}^{I} a_0^i = 1$. As discussed in [29], in contrast to the eigendecomposition method proposed in [24], this contrained least square method is robust against noise interference. The solution to (8) is

$$\breve{\mathbf{a}} = \frac{\mathbf{R}^{-1}\mathbf{g}}{\mathbf{g}^\top \mathbf{R}^{-1} \mathbf{g}}, \quad (10)$$

where $\mathbf{R}$ is the sample covariance matrix of the microphone signals, i.e. $\mathbf{R} = \sum_{p=1}^{P} \sum_{m=1}^{M} \tilde{\mathbf{x}}_{m,p}^* \tilde{\mathbf{x}}_{m,p}^\top$.

B. Recursive CTF Identification

In dynamic scenarios, the CTF vector $\tilde{\mathbf{a}}$ is time-varying, is thus rewritten as $\tilde{\mathbf{a}}^{(p)}$ to specify the frame-dependency. Note that we need to distinguish the superscript $^{(p)}$, which represents the time index with respect to the online update, from the subscript $_p$, which represents the frame index of the signals and filters. At frame p, $\tilde{\mathbf{a}}^{(p)}$ can be caculated by (10) using the microphone signals at frame p and recent frames. However, this requires a large amount of inverse matrix calculations, which is computationally expensive. In this work, we adopt the RLS-like algorithm for recursive CTF identification. At the current frame p, RLS aims to solve the minimization problem

$$\min \sum_{p'=1}^{p} \lambda^{p-p'} \Big(\sum_{m=1}^{M} |\tilde{\mathbf{x}}_{m,p'}^\top \tilde{\mathbf{a}}^{(p)}|^2 \Big) \quad \text{s.t. } \mathbf{g}^\top \tilde{\mathbf{a}} = 1. \quad (11)$$

The forgetting factor $\lambda^{p-p'}$ with $\lambda \in (0, 1]$ gives exponentially decaying weight to older frames. This time-weighted minimization problem can be solved using (10) with $\mathbf{R}$ replaced with a frame-dependent sample covariance matrix $\mathbf{R}^{(p)} = \sum_{p'=1}^{p} \lambda^{p-p'} (\sum_{m=1}^{M} \tilde{\mathbf{x}}_{m,p'}^{*} \tilde{\mathbf{x}}_{m,p'}^{\top})$, namely

$$\breve{\mathbf{a}}^{(p)} = \frac{(\mathbf{R}^{(p)})^{-1}\mathbf{g}}{\mathbf{g}^{\top}(\mathbf{R}^{(p)})^{-1}\mathbf{g}}. \tag{12}$$

$\mathbf{R}^{(p)}$ can be recursively updated as

$$\mathbf{R}^{(p)} = \lambda \mathbf{R}^{(p-1)} + \sum_{m=1}^{M} \tilde{\mathbf{x}}_{m,p}^{*} \tilde{\mathbf{x}}_{m,p}^{\top}. \tag{13}$$

The covariance matrix is updated in M steps, where each step modifies the covariance matrix by adding a rank-one matrix $\tilde{\mathbf{x}}_{m,p}^{*} \tilde{\mathbf{x}}_{m,p}^{\top}$, $m = 1, \ldots, M$. To avoid explicit inverse matrix computation, instead of $\mathbf{R}^{(p)}$ itself, we recursively estimate its inverse $(\mathbf{R}^{(p)})^{-1}$ based on the Sherman-Morrison formula (14). This procedure is summarized in Algorithm 1, where the Sherman-Morrison formula is applied in each of M loops. As an initialization, we set $(\mathbf{R}^{(0)})^{-1}$ to $1,000\mathbf{I}$, where $\mathbf{I}$ denotes identity matrix. The computational complexity of Algorithm 1 is proportional to the squared number of microphones. It is found by experiments that the microphone pairs are actually highly redundant for CTF estimation. Therefore, in practice, only the $I - 1$ microphone pairs that involve one specific microphone, e.g. the first microphone, are used. This achieves similar performance with using all microphone pairs.

Algorithm 1 Recursive estimation of $(\mathbf{R}^{(p)})^{-1}$ at frame p

Inputs: $\tilde{\mathbf{x}}_{m,p}$, $m = 1, \ldots, M$; $(\mathbf{R}^{(p-1)})^{-1}$
Initialization: $\mathbf{P} \leftarrow \lambda^{-1}(\mathbf{R}^{(p-1)})^{-1}$
for each microphone pair $m = 1$ to M **do**

$$\mathbf{P} \leftarrow \mathbf{P} - (\mathbf{P}\tilde{\mathbf{x}}_{m,p}^{*}\tilde{\mathbf{x}}_{m,p}^{\top}\mathbf{P})/(1 + \tilde{\mathbf{x}}_{m,p}^{\top}\mathbf{P}\tilde{\mathbf{x}}_{m,p}^{*}) \tag{14}$$

end for
Output: $(\mathbf{R}^{(p)})^{-1} \leftarrow \mathbf{P}$

The number of frames used to estimate $\tilde{\mathbf{a}}^{(p)}$ should be proportional to the length of the critically sampled CTF, i.e. $\tilde{Q}$, and is thus denoted with $\tilde{P} = \rho\tilde{Q}$. On the one hand, a large $\tilde{P}$ is required to ensure estimation accuracy. On the other hand, $\tilde{P}$ should be set as small as possible to reduce the dependency of the estimation on the past frames, namely to reduce the latency of the estimation, which is especially important for the moving speaker case. Similar to the RIR samples, the critically sampled CTF coefficients can be assumed to be temporally uncorrelated. However, the microphone signals STFT coefficients are highly correlated due to the temporal correlation of time-domain speech samples and to the oversampling of signals STFT coefficients (i.e. large overlapping of STFT frames). Empirically, we set $\rho = 2.5 \times 4 = 10$, where the factor 4 is used to compensate the signal oversampling effect. To approximately have a memory of $\tilde{P}$ frames, we can set $\lambda = \frac{\tilde{P}-1}{\tilde{P}+1}$.

III. Adaptive STFT Magnitude Inverse Filtering

In [29], it was found that the estimated complex-valued CTF is not accurate enough for effective inverse filtering, due to the influence of noise interference and the frequency aliasing caused by critical sampling. To reduce the sensitivity of the inverse filtering procedure to the CTF perturbations, instead of the complex-valued CTF convolution (2), its magnitude approximation was used, i.e.

$$|x_{i,p}| \approx |s_p| \star |a_{i,p}|, \quad i = 1, \ldots, I. \tag{15}$$

This magnitude convolution model is widely used in the context of dereverberation, e.g. [18], [19], [20]. In [32], [33], we proposed a MINT method based on the complex-valued CTF convolution for multisource separation and dereverberation. In the present work, we adapt this MINT method to the magnitude domain, and develop its adaptive version for online dereverberation.

A. Adaptive MINT in the Magnitude Domain

The CTF estimate of each channel, denoted by $\breve{\mathbf{a}}_i^{(p)}$, $i = 1, \ldots, I$, can be extracted from $\breve{\mathbf{a}}^{(p)}$. Let $\bar{\mathbf{a}}_i^{(p)} = |\breve{\mathbf{a}}_i^{(p)}|$ denote the CTF magnitude vector, and $\bar{a}_{i,0}^{(p)}, \ldots, \bar{a}_{i,\tilde{Q}-1}^{(p)}$ its elements. Define the inverse filters of $\bar{\mathbf{a}}_i^{(p)}$ in vector form as $\mathbf{h}_i^{(p)} \in \mathbb{R}^{\tilde{O}\times 1}$, $i = 1, \ldots, I$, where $\tilde{O}$ is the length of the inverse filters, which is assumed to be identical for all channels. Note that both $\bar{\mathbf{a}}_i^{(p)}$ and $\mathbf{h}_i^{(p)}$ are critically sampled. To apply the magnitude inverse filtering using $\mathbf{h}_i^{(p)}$, we construct the STFT magnitude vector of microphone signals as $\bar{\mathbf{x}}_{i,p} = [|x_{i,p}|, |x_{i,p-4}|, \ldots, |x_{i,p-4(\tilde{O}-1)}|]^{\top}$. The output of the multichannel inverse filtering is given by

$$\bar{s}_p = \sum_{i=1}^{I} \mathbf{h}_i^{(p)\top} \bar{\mathbf{x}}_{i,p}. \tag{16}$$

This output should target the STFT magnitude of the source signal, i.e. $|s_p|$.

To this aim, the multichannel equalization, i.e. MINT, should target an impulse function, namely

$$\sum_{i=1}^{I} \bar{\mathbf{A}}_i^{(p)} \mathbf{h}_i^{(p)} = \mathbf{d}, \tag{17}$$

where the impulse function $\mathbf{d}$ is defined by $\mathbf{d} = [1, 0, \ldots, 0]^{\top} \in \mathbb{R}^{(\tilde{Q}+\tilde{O}-1)\times 1}$, and the convolution matrix $\bar{\mathbf{A}}_i^{(p)}$ is defined by

$$\bar{\mathbf{A}}_i^{(p)} = \begin{bmatrix} \bar{a}_{i,0}^{(p)} & 0 & \cdots & 0 \\ \bar{a}_{i,1}^{(p)} & \bar{a}_{i,0}^{(p)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \bar{a}_{i,\tilde{Q}-1}^{(p)} & \vdots & \ddots & 0 \\ 0 & \bar{a}_{i,\tilde{Q}-1}^{(p)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \bar{a}_{i,\tilde{Q}-1}^{(p)} \end{bmatrix} \in \mathbb{R}_{\geq 0}^{(\tilde{Q}+\tilde{O}-1)\times\tilde{O}}. \tag{18}$$

In a more compact form, we can write

$$\bar{\mathbf{A}}^{(p)}\mathbf{h}^{(p)} = \mathbf{d}, \tag{19}$$

where $\bar{\mathbf{A}}^{(p)} = [\bar{\mathbf{A}}_1^{(p)}, \dots, \bar{\mathbf{A}}_I^{(p)}] \in \mathbb{R}_{\geq 0}^{(\tilde{Q}+\tilde{O}-1)\times I\tilde{O}}$ and $\mathbf{h}^{(p)} = [\mathbf{h}_1^{(p)\top}, \dots, \mathbf{h}_I^{(p)\top}]^\top \in \mathbb{R}^{I\tilde{O}\times 1}$. The inverse filter estimation amounts to solving problem (19), or equivalently minimizing the squared error

$$J^{(p)} = \| \bar{\mathbf{A}}^{(p)}\mathbf{h}^{(p)} - \mathbf{d} \|^2, \tag{20}$$

where $\| \cdot \|$ denotes ℓ_2-norm. The size of $\bar{\mathbf{A}}^{(p)}$ can be adjusted by tuning the length of the inverse filter, i.e. $\tilde{O}$. If $\bar{\mathbf{A}}^{(p)}$ is square or wide, i.e. $(\tilde{Q}+\tilde{O}-1) \leq I\tilde{O}$ and thus $\tilde{O} \geq \frac{\tilde{Q}-1}{I-1}$, (19) has an exact solution and (20) can reach zero. Otherwise, (19) is a least square problem, and only an approximate solution can be achieved.

The minimization of (20) has a closed-form solution. However, this needs the computation of an inverse matrix for each frame and frequency, which is computationally expensive. In this work, we propose to adaptively estimate $\mathbf{h}^{(p)}$ following the principle of normalized LMS. For a summary of normalized LMS design and analysis, please refer to Chapter 10.4 of [45]. The proposed LMS-like adaptive estimation method presented in the following is based on a stationary filtering system, but can be directly used for the nonstationary case due to its natural adaptive characteristic. In a stationary system, the filter to be estimated, i.e. the inverse filter $\mathbf{h}$ in the present work, is assumed to be time-invariant. The aim of LMS is to adaptively minimize the mean squared error $\mathbb{E}[J]$, where $\mathbb{E}[\cdot]$ denotes expectation. Note that with the superscript $^{(p)}$ removed, $\mathbf{h}$ and J denote the stationary filter and the (stationary) random variable for the squared error, respectively. At frame p, the instantaneous filtering process in (19) and the squared error (20) are a random instance of the stationary system. At frame p, the adaptive update uses the gradient of the instantaneous error $J^{(p)}$ at the previous estimation point $\mathbf{h}^{(p-1)}$, i.e.

$$\Delta J^{(p)}|_{\mathbf{h}^{(p-1)}} = 2\bar{\mathbf{A}}^{(p)\top}(\bar{\mathbf{A}}^{(p)}\mathbf{h}^{(p-1)} - \mathbf{d}). \tag{21}$$

An estimate of $\mathbf{h}^{(p)}$ based on the gradient descent update is

$$\mathbf{h}^{(p)} = \mathbf{h}^{(p-1)} - \frac{\mu}{\mathrm{Tr}(\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)})}\Delta J^{(p)}|_{\mathbf{h}^{(p-1)}}, \tag{22}$$

where $\mathrm{Tr}(\cdot)$ denotes the matrix trace, and $\frac{\mu}{\mathrm{Tr}(\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)})}$ is the *step-size* for gradient descent. The normalization term $\frac{1}{\mathrm{Tr}(\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)})}$ is set to make the gradient descent update converge to an optimal solution, namely to ensure the update stability. It is proven in [45] that, to guarantee the stability, the *step-size* should be set to be lower than $\frac{1}{\mathrm{Tr}(\mathbb{E}[\bar{\mathbf{A}}^\top\bar{\mathbf{A}}])}$, where $\bar{\mathbf{A}}$ denotes the (stationary) random variable for the CTF convolution matrix. Following the principle of normalized LMS, we replace the expectation $\mathbb{E}[\bar{\mathbf{A}}^\top\bar{\mathbf{A}}]$ with the instantaneous matrix $\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)}$. The matrix trace can be computed as $\mathrm{Tr}(\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)}) = \tilde{Q}\sum_{i=1}^{I}\bar{\mathbf{a}}_i^{(p)\top}\bar{\mathbf{a}}_i^{(p)}$. The constant step factor μ $(0 < \mu \leq 1)$ should be empirically set to achieve a good tradeoff between convergence rate (and tracking ability in a dynamic scenarios with time-varying CTFs) and update stability.

Algorithm 2 Adaptive STFT magnitude inverse filtering at frame p

Input: $\breve{\mathbf{a}}^{(p)}$ computed by (12) and $\mathbf{h}^{(p-1)}$.
1 Construct $\bar{\mathbf{A}}^{(p)}$ using (18),
2 Compute gradient $\Delta J^{(p)}|_{\mathbf{h}^{(p-1)}}$ using (21),
3 Update inverse filter $\mathbf{h}^{(p)}$ using (22),
4 Estimate the speech signal STFT magnitude $\bar{s}_p$ with inverse filtering (16).
Output: $\bar{s}_p$ and $\mathbf{h}^{(p)}$.

The proposed magnitude inverse filtering method is summarized in Algorithm 2, which is recursively executed frame by frame. As an initialization, we set $\mathbf{h}^{(0)}$ to a vector with all entries being zero.

B. Multichannel Processing

In the time-domain and complex-valued CTF MINT methods, e.g., [27], [32], [46], the optimal inverse filtering performance is achieved by setting the length of the inverse filter to the smallest value that makes $\bar{\mathbf{A}}^{(p)}$ be square or slightly wide, i.e. $\tilde{O} = \lceil \frac{\tilde{Q}-1}{I-1} \rceil$. This means $\tilde{O}$ becomes smaller with the increase of the number of channels. However, for the present magnitude inverse filtering method, this configuration is only suitable for the two-channel case. For the two-channel case, the length of the inverse filters $\tilde{O} = \tilde{Q} - 1$ is close to the CTF length $\tilde{Q}$, and in our experiments we actually set $\tilde{O} = \tilde{Q}$. The STFT magnitude of the microphone signals for the current frame includes the information of the past $\tilde{Q} - 1$ frames of the speech source signal due to the CTF convolution. Therefore, it is reasonable that the magnitude inverse filtering at the current frame uses the past $\tilde{Q} - 1$ frames of the microphone signals to remove the reflections.. When the number of channels is larger than two, the configuration $\tilde{O} = \lceil \frac{\tilde{Q}-1}{I-1} \rceil$ leads to a very small $\tilde{O}$, since the length of the critically sampled CTF, i.e. $\tilde{Q}$, is already relatively small. As will be shown in the experiments section, $\tilde{Q}$ is related to both the STFT setting and the reverberation time, and is set to 4 in this work. For the time-domain and complex-valued CTF MINT methods [27], [32], [46], dereverberation is guaranteed by solving the multichannel MINT equation, regardless of the length of the inverse filter, since the time-domain and CTF convolutions are exactly evaluated. By contrast, the magnitude convolution (15) is a rough approximation. Even if the magnitude MINT (19) can be exactly solved with a very small $\tilde{O}$, experiments show that the resulting magnitude inverse filtering is not able to efficiently suppress reverberation.

As detailed below, we propose two multichannel processing schemes suitable for the present magnitude inverse filtering method. They are both evaluated in Section IV.

1) Multichannel magnitude MINT with $\tilde{O} = \tilde{Q}$ regardless of the number of channels: This exactly follows the formulations presented in Section III-A. The setting $\tilde{O} = \tilde{Q}$ is motivated by the principle that, as is done for the two-channel case, the reflection magnitude of the past $\tilde{Q} - 1$ frames should be subtracted from the magnitude of the current frame.

2) Pairwise magnitude MINT: First, the adaptive MINT (and inverse filtering) presented in Section III-A is separately applied for each microphone pair. Then the estimates of the source magnitude obtained by all the M microphone pairs are averaged as a new source magnitude estimate, which is still denoted by $\bar{s}_p$ for brevity. The source magnitude estimates provided by the different microphone pairs are assumed to be independent, thence the average of them is hopefully suffering from lower interferences and distortions than each of them.

C. Postprocessing

The above STFT magnitude inverse filtering does not automatically guarantee the non-negativity of $\bar{s}_p$, which is infeasible solution for the STFT magnitude of the source signal. Negative values generally appear for the microphone signal frames with a magnitude that is considerably smaller than the magnitude in the preceding frames. Indeed, in that case, applying negative inverse filter coefficients to the preceding frames produces a negative magnitude estimate. Such frames are normally following a high-energy speech region, but themselves include very low source energy or purely reverberations. To overcome this, one way is to add the non-negativity constraint of the inverse filtering output to (20), which however leads to a larger complexity for both algorithm design and computation. Instead, we constrain the lower limit of the STFT magnitude of source signal according to the (averaged) STFT magnitude of microphone signals. Formally, the final estimate of the STFT magnitude of source signal is

$$\check{s}_p = \max(\bar{s}_p, G_{\min} \frac{1}{I} \sum_{i=1}^{I} |x_{i,p}|), \tag{23}$$

where $G_{\min}$ is a constant lower limit gain factor. This type of lower limit is widely used in single-channel speech enhancement methods, e.g. in [47], mainly to keep the noise naturalness. In the experiments described below, about 20% of TF bins are modified by this constraint.

Finally, the STFT phase of one of the microphone signals, e.g. the first microphone is used in this work, is taken as the phase of the estimated STFT coefficient of source signal, i.e. we have $\hat{s}_p = \check{s}_p e^{j \arg[x_p^1]}$, where $\arg[\cdot]$ is the phase of complex number. The time-domain source signal $\hat{s}(n)$ is obtained by applying the inverse STFT. Note that the MINT formulation (19) implies that the proposed inverse filtering method aims at recovering the signal corresponding to the first CTF frame, which not only includes the direct-path impulse response, but also the early reflections within the duration of one STFT frame. As a result, the estimated source signal $\hat{s}(n)$ includes both the direct-path source signal and its early reflections within N/f_s seconds following the direct-path propagation, where f_s is the signal sampling rate.

D. Difference from Conventional MINT Methods

Due to the use of i) the magnitude convolution model, ii) the critically sampled CTFs and inverse filters, and iii) the adaptive update of the inverse filters, the present adaptive MINT method is largely different from the complex-valued CTF MINT [32], [33] and the time-domain MINT, such as [26], [27], [46], [48], [49]. Besides the pairwise processing scheme, the two main differences are the following.

1) Desired Response of MINT: In many time-domain methods, to improve the robustness of MINT to microphone noise and filter perturbations, the target function (desired response) is designed to have multiple non-zero taps. This can be done either by explicitly filling the target function with multiple non-zero taps, such as the partial MINT in [27], or by relaxing the constraint for some taps, such as the relaxed multichannel least-squares in [46]. This way, the desired response with multiple non-zero taps includes both the direct-path propagation and some early reflections. In the present work, the impulse function $\mathbf{d}$ is used as the desired response of MINT in the CTF domain, namely only one non-zero tap is sufficient, since one tap of CTF corresponds to a segment of RIR that includes both direct-path propagation and early reflections.

It was shown in [32], [33] that, due to the effect of short time STFT windows, the oversampled CTF of multiple channels have common zeros, which is problematic for MINT. A target function incorporating the information of the STFT windows was proposed to compensate the common zeros. In the present work, the critically sampled CTFs do not suffer from this problem.

A modeling delay is always used in the time-domain MINT and complex-valued CTF MINT methods, i.e., in the target function, a number of zeros are inserted prior to the first non-zero tap. It is shown in [32], [48] that the optimal length of the modeling delay is related to the direct-path tap and the length of the room filters. In the present method, the room filters, i.e. CTFs, are blindly estimated, with the direct-path lying in the first tap. In addition, the CTF length is very small as mentioned above. Therefore, the modeling delay is set to 0, which achieved the best performance in our experiments.

2) Energy Regularization: An energy regularization is used in [27], [32], [48] to limit the energy of the inverse filters derived by MINT, since high energy inverse filters will amplify microphone noise and filter perturbations. For example, in the present problem, the optimal MINT solution could have a very large energy, especially when the matrix $\bar{\mathbf{A}}^{(p)\top}\bar{\mathbf{A}}^{(p)}$ is ill-conditioned. However, for the proposed method, the inverse filters are adaptively updated based on the previous estimation. The step size is set with guaranteed update stability. Thence, the energy of the inverse filters will not be boosted once the inverse filters are properly initialized.

IV. Experiments

A. Experimental Configuration

1) Dataset: We evaluate the proposed method using two datasets.

a) The REVERB challenge dataset [1]: We used the evaluation set of SimData-room3 and RealData datasets. SimData-room3 was generated by convolving clean signals from the WSJCAM0 dataset with RIRs measured in a room with reverberation time $T_{60} = 0.7$ s, and adding pre-recorded stationary ambient noise with an SNR of 20 dB. The microphone-to-speaker distances are 1 m (*near*) and 2 m (*far*). For these two distances, the *direct-to-reverberation ratios* (DRRs) are 10.6 dB and 1.0 dB, respectively, and the *early-to-late reverberation ratios* (C_{50}) are 14.9 dB and 6.3 dB, respectively. RealData was recorded in a noisy room with $T_{60} = 0.7$ s (different room than *SimData-room3*) and where humans pronounce MC-WSJ-AV utterances [50] microphone-to-speaker distances of 1 m (*near*) and 2.5 m (*far*). We used the data captured with two microphones (2-ch) or an eight-channel circular microphone array (8-ch).

We tested the automatic speech recognition (ASR) performance obtained with the enhanced signals, in addition to the speech enhancement performance. The ASR system provided by [51], [52], with the Kaldi recipe,[2] is taken as the baseline system. This system uses Mel-frequency cepstral coefficients (MFCC) and iVector [53] features, time-delay neural network (TDNN) acoustic model, and the WSJ 5k vocabulary and trigram language model. TDNN is capable to learn the long-term temporal dynamics of speech signals including the effects of reverberation. TDNN is trained using the multi-condition WSJCAM0 training dataset. The eight-channel multi-condition data are generated by convolving the 7,861 utterances of clean WSJCAM0 training signals with real recorded RIRs, and adding pre-recorded stationary ambient noise with an SNR of 20 dB. The eight-channel multi-condition data are then speed-perturbed with speed factors of 0.9, 1 and 1.1. In total, $7,861 \times 8 \times 3 = 188,664$ reverberant and speed-perturbed multi-condition utterances are used for TDNN training, which represents a total speech signal duration of about 373 hours. To account for the online nature of the proposed method, the online ASR decoding provided in the REVERB Kaldi recipe is used.

b) The Dynamic dataset [35]: This dataset was recorded by an eight-channel linear microphone array and a close-talk microphone in a room with $T_{60} = 0.75$ s. The average DRR and C_{50} values for this dataset are -5.5 dB and 3.0 dB, respectively. The recording SNR is about 20 dB. Four human speakers read an article from the New-York Times. Speakers could be static, or moving slightly, such as when standing up, sitting down and turning their head, or moving largely such as moving from one point to another. Speakers could be facing or not facing the microphone array. The total length of the dataset is 48 minutes. We split the data into three subsets: i) A subset with speakers being static and facing the microphone array (Static-FA). Note that some slight movements are inevitable even if human speakers are asked to be static; ii) Static and not facing the array (Static-NFA), and iii) Moving from one point to another. We used the central two channels (2-ch) or all the eight channels (8-ch). As for ASR, some pilot experiments show that the REVERB recognizer performs poorly for this dataset, since a number of words in this dataset are not in the WSJ 5k vocabulary. Instead, we used Google Cloud Speech-to-Text[3] to conduct the ASR experiment on this dataset.

2) Parameter Settings: The following parameter settings are used for both datasets, and all the experimental conditions. The sampling rate is 16 kHz. The STFT uses a Hamming window with length of $N = 768$ (48 ms) and frame step $L = N/4 = 192$ (16 ms). As a result, the 48 ms early reflections will be preserved in the dereverberated signal. It is shown in [54] that, to achieve a better ASR performance, early reflections should be removed as much as possible when late reverberation is perfectly removed. However, when the remaining late reverberation is not low, ASR performance benefits from preserving more early reflections up to 50 ms. Therefore, as we are dealing with adverse acoustic conditions, such as intense reverberation/noise or moving speakers, where late reverberation cannot be perfectly suppressed, we have decided to preserve the early reflections in the first 48 ms. The CTF length Q (and $\tilde{Q}$) is related to the reverberation time, and is the only prior knowledge that the proposed method requires. It is set to $Q = 16$ (and $\tilde{Q} = 4$), which covers the major part of the RIRs, and also excludes a heavy tail. According to the CTF length, the forgetting factor λ is set to $\frac{40-1}{40+1} \approx 0.95$. The constant step factor μ is set to 0.025. The constant lower limit gain factor G_{min} is set to correspond to -15 dB. These parameters are set to achieve the best ASR performance for the RealData subset of the REVERB challenge dataset, and are directly used for other experimental conditions.

3) Comparison Method: We compare the proposed method with the adaptive weighted prediction error (AWPE) method presented in [3]. The STFT uses a Hanning window with a length of 512 (32 ms) and frame step of 128 (8 ms). For the 2-ch and 8-ch cases, the length of the linear prediction filters is set to 16 and 8, respectively. The prediction delay is set to 6 to also involve 48 ms of early reflections in the dereverberated signal. In RLS, the length of the prediction filter vector to be estimated is equal to the length of the filters times the number of channels. Some pilot experiments show that, to obtain the optimal performance, the number of frames used to estimate the prediction filter vector should be set to be twice the vector length. Accordingly, the forgetting factor in RLS is set to 0.97 and 0.985 for the 2-ch and 8-ch cases, respectively. The first channel is taken as the target channel. Note that these parameters are also set to achieve the best ASR performance for RealData of REVERB challenge dataset, and are directly used for other experimental conditions.

To evaluate the effectiveness of the online realization of AWPE and the proposed method, we also conducted experiments using these methods implemented in offline (batch) mode. **i)** For the REVERB challenge dataset, the offline WPE is tested. We used the Python software package [4], which is integrated in the REVERB kaldi recipe. We adopted the WPE parameters as set by the authors of REVERB kaldi recipe, which are supposed to have been optimally tuned. The STFT

[2]https://github.com/kaldi-asr/kaldi/tree/master/egs/reverb

[3]https://cloud.google.com/speech-to-text/

TABLE I: SRMR, PESQ and STOI metrics (larger the better) for the REVERB challenge dataset.

		SRMR						PESQ			STOI		
		SimData-room3			RealData			SimData-room3			SimData-room3		
ch		*near*	*far*	Average	*near*	*far*	Average	*near*	*far*	Average	*near*	*far*	Average
	unproc.	2.35	2.29	2.32	2.29	2.20	2.24	1.89	1.55	1.72	0.89	0.71	0.80
2-ch	BWPE	2.44	2.42	2.43	2.55	2.54	2.55	2.06	1.67	1.87	0.92	0.78	0.85
	AWPE	2.61	2.84	2.73	2.99	2.98	2.99	2.32	1.77	2.05	0.78	0.76	0.77
	SMIF (ours)	2.51	2.63	2.57	2.83	2.76	2.80	2.25	1.74	2.00	0.77	0.73	0.75
8-ch	BWPE	2.49	2.59	2.54	2.79	2.83	2.81	2.38	2.10	2.24	0.94	0.87	0.91
	AWPE	2.60	2.89	2.75	3.04	3.01	3.03	2.48	1.90	2.19	0.80	0.79	0.80
	SMIF-MC (ours)	2.50	2.64	2.57	2.88	2.80	2.84	2.35	1.78	2.07	0.76	0.74	0.75
	SMIF-PW (ours)	2.51	2.72	2.62	2.94	2.87	2.91	2.40	1.84	2.12	0.78	0.75	0.77

configuration was the same as our AWPE implementation, namely using Hanning window with a length of 512 and a frame step of 128. The prediction delay is set to 3. The length of the linear prediction filters was set to 10 for both the 2-ch and 8-ch cases. The number of iterations for speech variance estimation was set to 5. We refer to this offline WPE as BWPE (batch WPE). **ii)** For the Dynamic dataset, the batch mode counterpart of the proposed method was tested. The CTF identification was conducted in batch mode using (10). Since the magnitude MINT in batch mode has not been investigated, we used the adaptive magnitude MINT presented in Section III for inverse filtering, where the inverse filter $\mathbf{h}^{(p)}$ quickly converged to a constant due to the use of the constant offline estimated CTF.

4) Performance Metrics: To evaluate the speech enhancement performance, three measures are used, i) a non-intrusive metric, i.e. normalized speech-to-reverberation modulation energy ratio (SRMR) [9], which mainly measures the amount of reverberation and noise, and also reflects the speech intelligibility; and two intrusive metrics ii) perceptual evaluation of speech quality (PESQ) [55] evaluates the quality of the enhanced signal in terms of both reverberation reduction and speech distortion; iii) short-time objective intelligibility (STOI) [56] is a metric that highly correlates with speech intelligibility. To measure PESQ and STOI, the clean source signal is taken as the reference signal. For the Dynamic dataset, the close-talk recording is taken as the source signal. For RealData of the REVERB challenge dataset, the clean signals are not available, thus neither PESQ nor STOI metrics are reported in this case. For these three metrics, the larger the better. The ASR performance is measured with the percentage of word error rate (WER): the lower the better. Note that all the tested methods do not perform noise reduction, thence the outputs used to calculate the metrics may contain some amount of noise.

B. Results for the REVERB Challenge Dataset

In the REVERB challenge dataset, each subset involves several hundreds of individual signals, with each signal being one utterance spoken by one static speaker. The relative speaker-microphone position changes from utterance to utterance. To simulate a realistic turn-taking scenario, for each subset, all the individual signals are first concatenated as a long signal, which is then processed by the online dereverberation methods, i.e. AWPE and the proposed method. The long enhanced signal is finally separated corresponding to the original individual signals. For BWPE, the individual signals are separately processed. The perfomance measures are computed using the individual enhanced signals.

1) Speech Enhancement Results: We refer to the proposed method as **SMIF** (Spectral Magnitude Inverse Filtering). For the multichannel case, the two schemes proposed in Section III-B, i.e. multichannel processing and pairwise processing, are refered to SMIF-MC and SMIF-PW, respectively. Table I presents the speech enhancement results. As for the proposed method, compared to the 2-ch case, the 8-ch SMIF-MC method improves the SRMR and PESQ metrics on RealData, and achieves identical SRMR and STOI metrics on the SimData-room3 data. The 8-ch SMIF-PW method systematically outperforms the 2-ch case and the 8-ch SMIF-MC method. This indicates that, for the SMIF-MC method, the magnitude inverse filtering accuracy can be improved by using more microphones, however the improvement is not always significant in terms of speech enhancement metrics. In the 8-ch SMIF-PW method, the average of pairwise source estimates successfully suppress the interferences and distortions of the one-pair source estimates. Informal listening tests show that the residual late reverberation can be sometimes noticeably perceived for the 2-ch case, while it is not clearly audible for the 8-ch case.

For all conditions and for all metrics, AWPE outperforms the proposed method, especially the gaps between SRMR metrics are noticeable, see Table I. The proposed method is based on the STFT-magnitude convolution and inverse filtering, which is a coarse approximation of the real filtering process. By contrast, AWPE is based on a more accurate complex-valued inverse filtering. As a result, the dereverberated signals obtained with the proposed method are likely to have more late reverberation, extra noise and speech distortions, especially for the 2-ch case. Relative to the unprocessed signal, AWPE and the proposed method slightly improve the STOI metrics for the *far* case, but reduce the STOI metrics for the *near* case. This is possibly because the parameters are set based on the RealData data, and in particular the length of the (inverse) filters may be too large for the *near* simulation data.

Compared to AWPE, BWPE achieves worse SRMR and 2-ch PESQ metrics, and better 8-ch PESQ and STOI metrics.

TABLE II: WER (%) for the REVERB challenge dataset.

ch		SimData-room3 *near*	SimData-room3 *far*	SimData-room3 Average	RealData *near*	RealData *far*	RealData Average
	unproc.	5.08	8.08	6.58	20.95	21.27	21.11
2-ch	BWPE	4.55	6.95	5.75	15.65	15.77	15.71
	AWPE	5.37	7.28	6.33	15.36	16.21	15.79
	SMIF (ours)	5.01	7.16	6.09	15.30	16.04	15.67
8-ch	BWPE	4.04	4.96	4.50	12.20	13.17	12.69
	AWPE	4.65	6.07	5.36	12.26	13.54	12.90
	SMIF-MC (ours)	4.53	6.34	5.44	13.09	14.11	13.60
	SMIF-PW (ours)	4.53	5.98	5.26	13.00	14.48	13.74

Generally speaking, BWPE would outperform AWPE if the same parameters were set for both methods, since the speech variance estimate of BWPE is more accurate than the one for AWPE, where the former is iteratively estimated while the latter is approximated by the microphone speech variance. The performance difference between BWPE and AWPE is mainly due to their different prediction delays, i.e. 3 and 6 respectively. A larger prediction delay preserves more early reverberation, which promotes the SRMR metrics, but leads to a larger difference with the clean direct-path signal.

2) ASR Results: Table II presents the WER. It is seen that the present baseline WERs are already very advanced compared with the REVERB challenge WERs reported in [1]. The baseline WERs are noticably reduced by all the tested methods. For instance, as for RealData, the proposed method achieves 25.8%, 35.6% and 34.9% relative WER improvement with 2-ch, 8-ch SMIF-MC and SMIF-PW schemes, respectively. In contrast to the speech enhancement metrics presented in Table I, the ASR performance of the proposed 8-ch SMIF-MC method is noticeably better than the one of the 2-ch case, and is comparable to the one of the 8-ch SMIF-PW method. This means the speech quality improvement caused by the 8-ch SMIF-MC method over the 2-ch case can be well recognized by the ASR system.

Approximately, the proposed method achieves comparable ASR performance with AWPE. Compared with AWPE, the remaining late reverberation and extra noise caused by the proposed method degrades the speech enhancement metrics as shown in Table I, but can be tackled by the well-trained TDNN acoustic model.

AWPE does not perform as well as BWPE for SimData-room3, but is comparable to BWPE for RealData. As mentioned above, AWPE preserves more early reflections, which is beneficial for the more challenging RealData, since the late reverberation cannot be well suppressed. Concerning the RealData, it is possible to further improve the BWPE parameters. However, the parameter tuning for BWPE is out of the scope of this work.

3) Dereverberation Performance under Noisy Conditions: To evaluate the sensitivity of the proposed method to noise, experiments for the SimData-room3 *far* data are conducted with various SNRs. Fig. 1 shows the results. As expected, the performance of the proposed method decreases with the decrease of SNR, and it has a similar decrease rate with the performance of AWPE. In terms of SRMR, the performance of the two methods have a similar decrease rate with the one of the unprocessed signals, and the performance improvement of the two methods over the unprocessed signals is still significant when SNR is low, e.g. 0 dB. For PESQ and STOI, the performance metrics of the two methods gradually approach the metrics of the unprocessed signals with the decrease of SNR. This means these two metrics are dominated by the intense noise for the low SNR cases. The WER improvement of the two methods over the unprocessed signals are even larger for the low SNR cases than for the high SNR cases. This indicates that reverberation degrades the ASR performance more significantly when it is combined with noise than itself alone, and the two methods are able to efficiently suppress reverberation under intense noise condition.

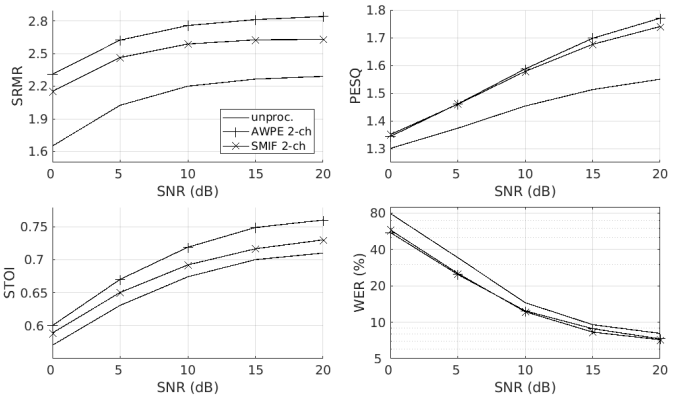

Fig. 1: Dereverberation performance as a function of SNR, for the SimData-room3 *far* data.

C. Results for Dynamic Dataset

Fig. 2 presents the dereverberation results for the three subsets in the Dynamic dataset. For the unprocessed data, all the performance measures are bad due to the intense reverberation. The Static-NFA set has the lowest SRMR and PESQ metrics. When speakers do not face the microphones, the direct-path speech signal received by microphones becomes smaller relative to the reverberation and ambient noise, in other words the microphone signals are more reverberated and noisy. The Moving case has the lowest STOI metrics. The WER clearly increases from the Static-FA set to the Static-NFA and Moving sets.

For all conditions and performance metrics, the proposed 8-ch SMIF-MC and SMIF-PW methods perform similarly, thence we will not distinguish them in the following. For both AWPE and the proposed method, the SRMR performance slightly degrades from the Static-FA set to the Static-NFA set, and further noticeably degrade for the Moving set. AWPE achieves larger PESQ metrics than the proposed method for the static cases, but has a large performance degradation for the Moving set. By contrast, the proposed method achieves even larger PESQ metrics for the Moving set. In terms of STOI, the two methods perform similarly for the static cases, and the proposed method outperforms AWPE for the Moving set. As

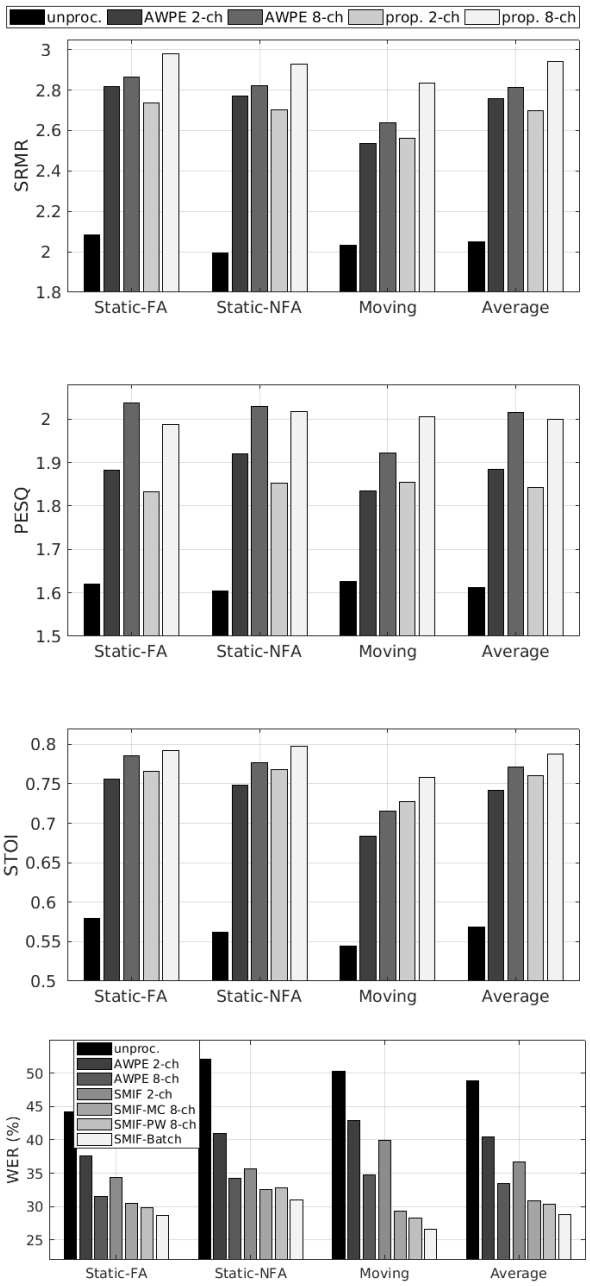

Fig. 2: Dereverberation performance, i.e. SRMR, PESQ, STOI metrics and WER (from top to bottom), for the Dynamic dataset. The WER of close-talk signals for the three subsets are 22.1%, 24.2% and 14.4%, respectively.

for ASR, the proposed method outperforms AWPE, especially for the Moving set. Overall, the performance measures show the comparable dereverberation capability of AWPE and the proposed method for the static speaker cases, and show the superiority of the proposed method for the moving speaker case. The Dynamic dataset is more challenging than the REVERB dataset in terms of adaptive (inverse) filter estimation mainly due to its lower DRR and C_{50}. In addition, the moving speaker case suffers a larger filter estimation error compared to the static speaker case, due to the imperfect tracking ability. Compared to the complex-valued inverse filtering in AWPE, the proposed STFT magnitude inverse filtering is less sensitive to additive noise, filter perturbations and other unexpected distortions [1].

The batch mode counterpart of the proposed method, referred to as SMIF-Batch in Fig. 2, uses eight microphones and the pairwise scheme of magnitude inverse filtering. The speech enhancement and ASR performance measures of the batch method are not consistent. Compared to the online method, on the one hand, the batch method achieves worse speech enhancement metrics, even for the static speaker cases. On the other hand, it performs slightly better for ASR, even for the moving speaker case. The reason for this inconsistency is not very clear. The present work uses the critically sampled CTF convolution and the magnitude CTF convolution, which are rough approximations. As a result, for the static speaker case, the CTF and inverse filter that optimize the approximations are actually time-varying, and thus the online method could sometimes outperform the batch method.

Fig. 3 shows the STOI metrics computed with a 1-s sliding window for one audio recording. This result is consistent with Fig. 2 depicting that the two methods have comparable STOI metrics when the speaker is static before 11 s, and the proposed method achieves higher STOI metrics when the speaker is moving after 11 s. When the speaker starts speaking after a silent period, the two methods adapt from background noise to speech, and quickly converge. It is observed from Fig. 3 that the two methods have a similar convergence speed, i.e. less than 1 s. Fig. 4 depicts the spectrograms of the middle part (around the point where the speaker starts moving) of the recording in Fig. 3. It can be seen that reverberation is largely removed by both methods. However, the difference between the two methods and the difference between the static and moving cases cannot be clearly observed from the spectrograms. Informal listening tests show that, the proposed method is not perceived to have more residual reverberation for the moving speaker case compared to the static speaker case. Audio examples for all experiments presented in this paper are available in our website.[4]

D. Computational Complexity Analysis

Both the proposed method and AWPE are frame-wise online methods. We analyze their computational complexity for one frame. The proposed method consists of CTF identification and magnitude inverse filtering. The computation of CTF identification is mainly composed of Algorithm 1, which executes (14) $I-1$ times. The computation of (14) includes three matrix-vector multiplications. The matrix/vector size is $I\tilde{Q}$. We remind that $I=2$ or 8 and $\tilde{Q}=4$ are the number of channels and the length of the critically sampled CTF, respectively. CTF identification is performed for each of the $N/2+1$ positive-valued frequency bins. Overall, the computational complexity of CTF identification is approximately $\mathcal{O}(NI^3\tilde{Q}^2)$. The computation of inverse filtering is mainly composed of the gradient calculation (21), which includes two

[4]https://team.inria.fr/perception/research/ctf-dereverberation

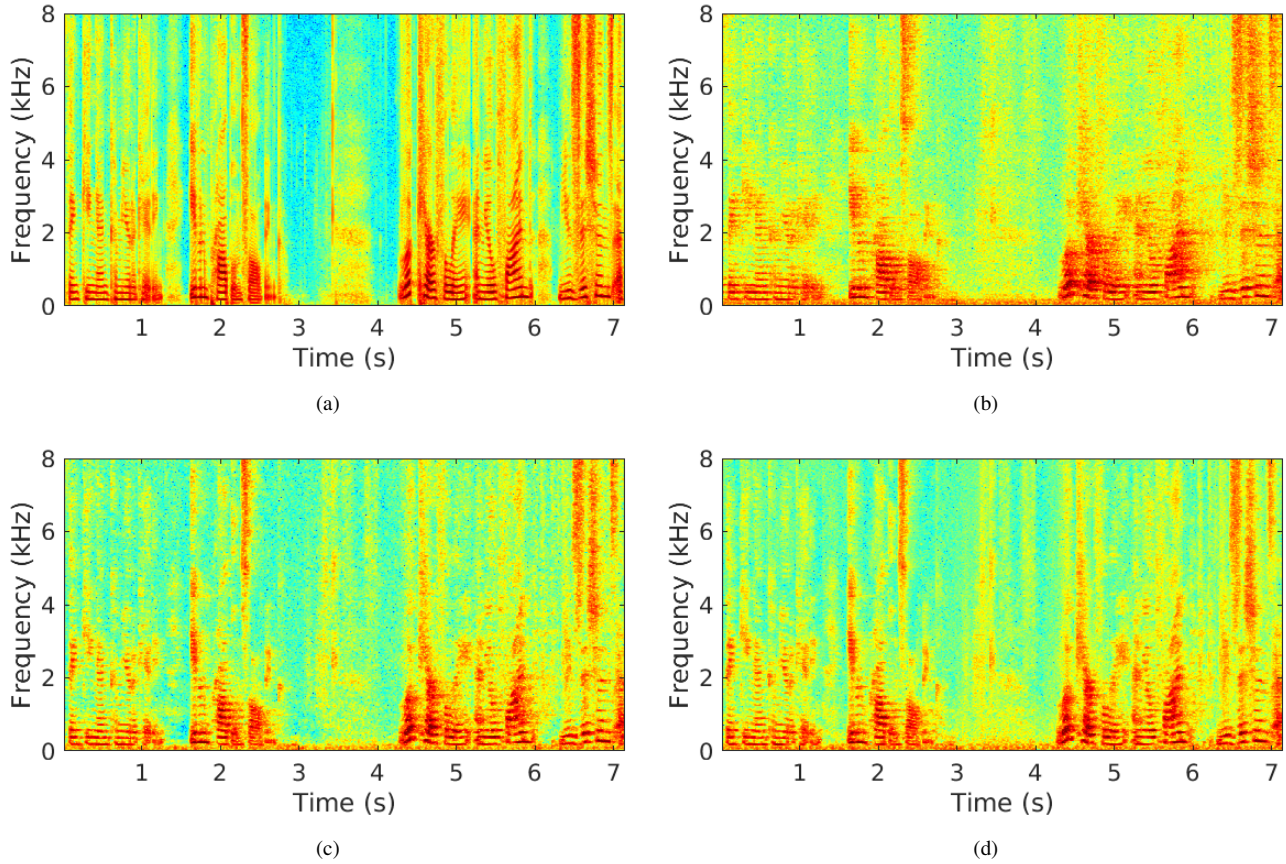

Fig. 4: Example of spectrogram for a signal from the Dynamic dataset. (a) close-talk clean signal, (b) microphone signal, (c) enhanced signal by 8-ch AWPE and (d) the proposed 8-ch SMIF-PW method. The speaker was static with in 0-4 s, and started walking from one point to another from 4 s.

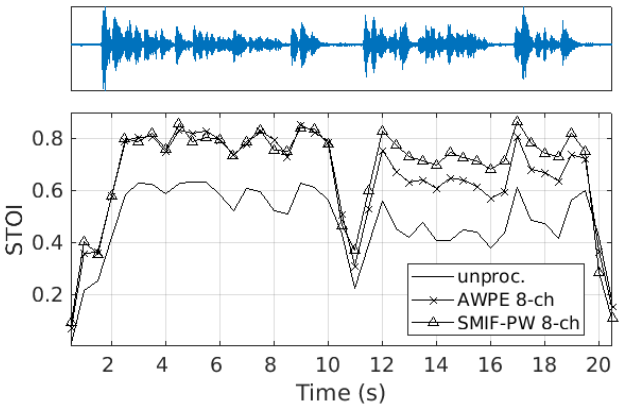

Fig. 3: The short-time STOI metrics computed with a 1-s sliding window and 0.5 s sliding step. One speaker was standing at one point within 0-11 s, and started walking to another point from 11 s.

matrix-vector multiplications. However, each of these multiplications actually represents I one-dimensional convolutions. In practice, we implement the convolution using an FFT (fast Fourier transform) with $N_{\text{fft}} = 2\tilde{Q}+\tilde{O}-2$ points, where $\tilde{O} = 4$ is the length of the inverse filter. Overall, the computational complexity of multichannel inverse filtering is approximately $\mathcal{O}(NIN_{\text{fft}}\log(N_{\text{fft}}))$. For the pairwise processing scheme, the two-channel inverse filtering is executed $I(I-1)/2$ times, thence the computational complexity is $\mathcal{O}(NI^2N_{\text{fft}}\log(N_{\text{fft}}))$.

TABLE III: Real-time factor for AWPE and each step of the proposed method.

Method		2-ch	8-ch
AWPE		0.54	2.45
SMIF (ours)	CTF identification	0.11	2.73
	Inverse filtering (SMIF-MC)	0.09	0.52
	Inverse filtering (SMIF-PW)	0.09	1.35
	Overall (SMIF-MC)	0.20	3.25
	Overall (SMIF-PW)	0.20	4.08

Similar to the proposed CTF identification method, the computation of RLS-based AWPE is also composed of matrix-vector multiplications. The matrix/vector size is IQ_{wpe}, where Q_{wpe} denotes the length of the prediction filter, i.e. 16 and 8 for the 2-ch and 8-ch cases, respectively. The computational complexity of AWPE is $\mathcal{O}(N_{\text{wpe}}I^2Q_{\text{wpe}}^2)$, where N_{wpe} denotes the STFT frame length for AWPE, i.e. 512 in this experiment.

The computation time is measured with the real-time factor (RF), which is the processing time of a method divided by the length of the processed signal. Both AWPE and the proposed method are implemented in MATLAB. RF for WPE and each step of the proposed method are shown in Table III. For the 2-ch case, all processes have an RF smaller than 1, and thus

can be run in real-time. The proposed method is less time-consuming than AWPE, since the critically sampled CTF and inverse filter of the proposed method are shorter than the predicition filter of AWPE, i.e. 4 versus 16. For the 8-ch case, AWPE is faster than the proposed method. As analyzed above, the computational complexity of the proposed CTF identification is cubic of the number of channels, while the one of AWPE is square of the number of channels.

V. Conclusions

In this paper, a blind multichannel online dereverberation method has been proposed. The batch algorithm for multi-channel CTF identification proposed in our previous work [33] was extended to an online method based on the RLS criterion. Then, a gradient descent-based adaptive magnitude MINT was proposed to estimate the inverse filters of the identified CTF magnitude. Finally, an estimate of the STFT magnitude of the source signal can be obtained by applying the inverse filtering onto the STFT magnitude of the microphone signals. Experiments were conducted in terms of both speech quality and intelligibility. Compared to the AWPE method, the proposed method achieves comparable ASR performance on the REVERB challenge dataset. Experiments with the Dynamic dataset show that the proposed method performs better than AWPE for the moving speaker case due to the robustness of the STFT magnitude-based scheme. Even though the proposed method does not account for noise reduction at all, the dereverberation experiments were performed on data including additive noise. The experimental results indicate that the dereverberation capability of the proposed method is not significantly deteriorated by the additive noise. However, the noise in the dereverberated signal still has some influence on both human listening and ASR metrics. A noise reduction method that fits well the proposed dereverberation method will be investigated in the future.

References

[1] K. Kinoshita, M. Delcroix, S. Gannot, E. A. Habets, R. Haeb-Umbach, W. Kellermann, V. Leutnant, R. Maas, T. Nakatani, B. Raj, *et al.*, "A summary of the REVERB challenge: state-of-the-art and remaining challenges in reverberant speech processing research," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 1–19, 2016.

[2] B. Li, T. Sainath, A. Narayanan, J. Caroselli, M. Bacchiani, A. Misra, I. Shafran, H. Sak, G. Pundak, K. Chin, *et al.*, "Acoustic modeling for google home," *Proc. Interspeech*, pp. 399–403, 2017.

[3] J. Caroselli, I. Shafran, A. Narayanan, and R. Rose, "Adaptive multichannel dereverberation for automatic speech recognition," in *Proc. Interspeech*, 2017.

[4] L. Drude, J. Heymann, C. Boeddeker, and R. Haeb-Umbach, "Narawpe: A python package for weighted prediction error dereverberation in numpy and tensorflow for online and offline processing," in *13th ITG-Symposium Speech Communication*, pp. 1–5, VDE, 2018.

[5] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, and T. Nakatani, "Frame-online dnn-wpe dereverberation," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 466–470, IEEE, 2018.

[6] A. A. Kressner, A. Westermann, and J. M. Buchholz, "The impact of reverberation on speech intelligibility in cochlear implant recipients," *The Journal of the Acoustical Society of America*, vol. 144, no. 2, pp. 1113–1122, 2018.

[7] J. Xia, B. Xu, S. Pentony, J. Xu, and J. Swaminathan, "Effects of reverberation and noise on speech intelligibility in normal-hearing and aided hearing-impaired listeners," *The Journal of the Acoustical Society of America*, vol. 143, no. 3, pp. 1523–1533, 2018.

[8] J. F. Santos and T. H. Falk, "Updating the SRMR-CI metric for improved intelligibility prediction for cochlear implant users," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 2197–2206, 2014.

[9] J. F. Santos, M. Senoussaoui, and T. H. Falk, "An improved non-intrusive intelligibility metric for noisy and reverberant speech," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 55–59, IEEE, 2014.

[10] A. Warzybok, I. Kodrasi, J. O. Jungmann, E. Habets, T. Gerkmann, A. Mertins, S. Doclo, B. Kollmeier, and S. Goetze, "Subjective speech quality and speech intelligibility evaluation of single-channel dereverberation algorithms," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 332–336, IEEE, 2014.

[11] Y. Zhao, D. Wang, E. M. Johnson, and E. W. Healy, "A deep learning based segregation algorithm to increase speech intelligibility for hearing-impaired listeners in reverberant-noisy conditions," *The Journal of the Acoustical Society of America*, vol. 144, no. 3, pp. 1627–1637, 2018.

[12] I. Arweiler and J. M. Buchholz, "The influence of spectral characteristics of early reflections on speech intelligibility," *The Journal of the Acoustical Society of America*, vol. 130, no. 2, pp. 996–1005, 2011.

[13] E. A. Habets, S. Gannot, and I. Cohen, "Late reverberant spectral variance estimation based on a statistical model," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 770–773, 2009.

[14] A. Schwarz and W. Kellermann, "Coherent-to-diffuse power ratio estimation for dereverberation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 6, pp. 1006–1018, 2015.

[15] A. Kuklasiński, S. Doclo, S. H. Jensen, and J. Jensen, "Maximum likelihood PSD estimation for speech enhancement in reverberation and noise," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 9, pp. 1595–1608, 2016.

[16] O. Schwartz, S. Gannot, E. Habets, *et al.*, "Multi-microphone speech dereverberation and noise reduction using relative early transfer functions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 240–251, 2015.

[17] X. Li, L. Girin, and R. Horaud, "An EM algorithm for audio source separation based on the convolutive transfer function," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2017.

[18] S. Mirsamadi and J. H. Hansen, "Multichannel speech dereverberation based on convolutive nonnegative tensor factorization for ASR applications," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[19] N. Mohammadiha and S. Doclo, "Speech dereverberation using non-negative convolutive transfer function and spectro-temporal modeling," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 2, pp. 276–289, 2016.

[20] D. Baby and H. Van Hamme, "Joint denoising and dereverberation using exemplar-based sparse representations and decaying norm constraint," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 2024–2035, 2017.

[21] M. Delcroix, T. Hikichi, and M. Miyoshi, "Precise dereverberation using multichannel linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 2, pp. 430–440, 2007.

[22] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 534–545, 2009.

[23] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717–1731, 2010.

[24] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Transactions on signal processing*, vol. 43, no. 12, pp. 2982–2993, 1995.

[25] M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 36, no. 2, pp. 145–152, 1988.

[26] M. Kallinger and A. Mertins, "Multi-channel room impulse response shaping-a study," in *IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 5, pp. V101–V104, 2006.

[27] I. Kodrasi, S. Goetze, and S. Doclo, "Regularization for partial multichannel equalization for speech dereverberation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 9, pp. 1879–1890, 2013.

[28] X. Li, L. Girin, R. Horaud, and S. Gannot, "Estimation of the direct-path relative transfer function for supervised sound-source localization," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 24, no. 11, pp. 2171–2186, 2016.

[29] X. Li, S. Gannot, L. Girin, and R. Horaud, "Multichannel identification and nonnegative equalization for dereverberation and noise reduction based on convolutive transfer function," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1755–1768, 2018.

[30] S. Weiss, G. W. Rice, and R. W. Stewart, "Multichannel equalization in subbands," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 203–206, 1999.

[31] N. D. Gaubitch and P. A. Naylor, "Equalization of multichannel acoustic systems in oversampled subbands," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1061–1070, 2009.

[32] X. Li, S. Gannot, L. Girin, and R. Horaud, "Multisource MINT using the convolutive transfer function," in *IEEE International Conference on Acoustic, Speech and Signal Processing*, 2018.

[33] X. Li, L. Girin, S. Gannot, and R. Horaud, "Multichannel speech separation and enhancement using the convolutive transfer function," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 645–659, 2018.

[34] B. Schwartz, S. Gannot, and E. A. Habets, "An online dereverberation algorithm for hearing aids with binaural cues preservation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 1–5, 2015.

[35] B. Schwartz, S. Gannot, and E. A. Habets, "Online speech dereverberation using Kalman filter and EM algorithm," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 2, pp. 394–406, 2015.

[36] J.-M. Yang and H.-G. Kang, "Online speech dereverberation algorithm based on adaptive multichannel linear prediction," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 3, pp. 608–619, 2014.

[37] T. Yoshioka, H. Tachibana, T. Nakatani, and M. Miyoshi, "Adaptive dereverberation of speech signals with speaker-position change detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3733–3736, 2009.

[38] T. Yoshioka and T. Nakatani, "Dereverberation for reverberation-robust microphone arrays," in *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, pp. 1–5, 2013.

[39] T. Xiang, J. Lu, and K. Chen, "RLS-based adaptive dereverberation tracing abrupt position change of target speaker," in *IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 336–340, 2018.

[40] S. Braun and E. A. Habets, "Online dereverberation for dynamic scenarios using a Kalman filter with an autoregressive model," *IEEE Signal Processing Letters*, vol. 23, no. 12, pp. 1741–1745, 2016.

[41] S. Braun and E. A. Habets, "Linear prediction based online dereverberation and noise reduction using alternating Kalman filters," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1119–1129, 2018.

[42] Y. Huang and J. Benesty, "A class of frequency-domain adaptive approaches to blind multichannel identification," *IEEE Transactions on Signal Processing*, vol. 51, no. 1, pp. 11–24, 2003.

[43] W. Zhang, A. W. Khong, and P. A. Naylor, "Adaptive inverse filtering of room acoustics," in *Asilomar Conference on Signals, Systems and Computers*, pp. 788–792, IEEE, 2008.

[44] D. Liu, R. S. Rashobh, A. W. Khong, and M. Yukawa, "A subspace-based adaptive approach for multichannel equalization of room acoustics," in *Proc. Asia-Pacific Signal and Info. Process. Assoc. Annual Summit and Conf.*, 2011.

[45] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and adaptive signal processing: spectral estimation, signal modeling, adaptive filtering, and array processing*. McGraw-Hill Boston, 2000.

[46] F. Lim, W. Zhang, E. A. Habets, and P. A. Naylor, "Robust multichannel dereverberation using relaxed multichannel least squares," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 9, pp. 1379–1390, 2014.

[47] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal processing*, vol. 81, no. 11, pp. 2403–2418, 2001.

[48] T. Hikichi, M. Delcroix, and M. Miyoshi, "Inverse filtering for speech dereverberation less sensitive to noise and room transfer function fluctuations," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 1–12, 2007.

[49] A. Mertins, T. Mei, and M. Kallinger, "Room impulse response shortening/reshaping with infinity-and-norm optimization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 249–259, 2010.

[50] M. Lincoln, I. McCowan, J. Vepa, and H. K. Maganti, "The multi-channel wall street journal audio visual corpus (mc-wsj-av): Specification and initial experiments," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 357–362, 2005.

[51] F. Weninger, S. Watanabe, J. Le Roux, J. Hershey, Y. Tachioka, J. Geiger, B. Schuller, and G. Rigoll, "The merl/melco/tum system for the reverb challenge using deep recurrent neural network feature enhancement," in *Proc. REVERB Workshop*, 2014.

[52] V. Peddinti, G. Chen, V. Manohar, T. Ko, D. Povey, and S. Khudanpur, "Jhu aspire system: Robust lvcsr with tdnns, ivector adaptation and rnn-lms," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 539–546, IEEE, 2015.

[53] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[54] A. Sehr, E. A. Habets, R. Maas, and W. Kellermann, "Towards a better understanding of the effect of reverberation on speech recognition performance," in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2010.

[55] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 749–752, 2001.

[56] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.