Hindawi Publishing Corporation
EURASIP Journal on Audio, Speech, and Music Processing
Volume 2007, Article ID 51831, 19 pages
doi:10.1155/2007/51831

Research Article

Multimicrophone Speech Dereverberation: Experimental Validation

Koen Eneman[1, 2] and Marc Moonen[3]

[1] *ExpORL, Department of Neurosciences, Katholieke Universiteit Leuven, O & N 2, Herestraat 49 bus 721, 3000 Leuven , Belgium*
[2] *GroupT Leuven Engineering School, Vesaliusstraat 13, 3000 Leuven, Belgium*
[3] *SCD, Department of Electrical Engineering (ESAT), Faculty of Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium*

Received 6 September 2006; Revised 9 January 2007; Accepted 10 April 2007

Recommended by James Kates

Dereverberation is required in various speech processing applications such as handsfree telephony and voice-controlled systems, especially when signals are applied that are recorded in a moderately or highly reverberant environment. In this paper, we compare a number of classical and more recently developed multimicrophone dereverberation algorithms, and validate the different algorithmic settings by means of two performance indices and a speech recognition system. It is found that some of the classical solutions obtain a moderate signal enhancement. More advanced subspace-based dereverberation techniques, on the other hand, fail to enhance the signals despite their high-computational load.

1. INTRODUCTION

In various speech communication applications such as teleconferencing, handsfree telephony, and voice-controlled systems, the signal quality is degraded in many ways. Apart from acoustic echoes and background noise, reverberation is added to the signal of interest as the signal propagates through the recording room and reflects off walls, objects, and people. Of the different types of signal deterioration that occur in speech processing applications such as teleconferencing and handsfree telephony, reverberation is probably least disturbing at first sight. However, rooms with a moderate to high reflectivity reverberation can have a clearly negative impact on the intelligibility of the recorded speech, and can hence significantly complicate conversation. *Dereverberation* techniques are then called for to enhance the recorded speech. Performance losses are also observed in voice-controlled systems whenever signals are applied that are recorded in a moderately or highly reverberant environment. Such systems rely on automatic speech recognition software, which is typically trained under more or less anechoic conditions. Recognition rates therefore drop, unless adequate dereverberation is applied to the input signals.

Many speech dereverberation algorithms have been developed over the last decades. However, the solutions available today appear to be, in general, not very satisfactory, as will be illustrated in this paper. In the literature, different classes of dereverberation algorithms have been described. Here, we will focus on multimicrophone dereverberation algorithms, as these appear to be most promising. Cepstrum-based techniques were reported first [1–4]. They rely on the separability of speech and acoustics in the cepstral domain. Coherence-based dereverberation algorithms [5, 6] on the other hand, can be applied to increase listening comfort and speech intelligibility in reverberating environments and in diffuse background noise. Inverse filtering-based methods attempt to invert the acoustic impulse response, and have been reported in [7, 8]. However, as the impulse responses are known to be typically nonminimum phase they have an unstable (causal) inverse. Nevertheless, a noncausal stable inverse may exist. Whether the impulse responses are minimum phase depends on the reverberation level. Acoustic beamforming solutions have been proposed in [9–11]. Beamformers were mainly designed to suppress background noise, but are known to partially dereverberate the signals as well. A promising matched filtering-based speech dereverberation scheme has been proposed in [12]. The algorithm relies on subspace tracking and shows improved dereverberation capabilities with respect to classical solutions. However, as some environmental parameters are assumed to be known in advance, this approach may be less suitable in practical applications. Finally, over the last years, many blind subspace-based system identification techniques have been developed for channel equalization in digital communications [13, 14]. These techniques can be applied to speech enhancement applications as well [15], be it with limited success so far.

In this paper, we give an overview of existing dereverberation techniques and discuss more recently developed subspace and frequency-domain solutions. The presented algorithms are compared based on two performance indices and are evaluated with respect to their ability to enhance the word recognition rate of a speech recognition system. In Section 2, a problem statement is given and a general framework is presented in which the different dereverberation algorithms can be cast. The dereverberation techniques that have been selected for the evaluation are discussed in Section 3. The speech recognition system and the performance indices that are used for the evaluation are defined in Section 4. Section 5 describes the experiments based on which dereverberation algorithms have been evaluated and discusses the experimental results. The conclusions are formulated in Section 6.

2. SPEECH DEREVERBERATION

The signal quality in various speech communication applications such as teleconferencing, handsfree telephony, and voice-controlled systems is compromised in many ways. A first type of disturbance are the so-called acoustic echoes, which arise whenever a loudspeaker signal is picked up by the microphone(s). A second source of signal deterioration is noise and disturbances that are added to the signal of interest. Finally, additional signal degradation occurs when reverberation is added to the signal as it propagates through the recording room reflecting off walls, objects, and people. This propagation results in a signal attenuation and spectral distortion that can be modeled well by a linear filter. Nonlinear effects are typically of second-order and mainly stem from the nonlinear characteristics of the loudspeakers. The linear filter that relates the emitted signal to the received signal is called the acoustic impulse response [16] and plays an important role in many signal enhancement techniques. Often, the acoustic impulse response is a nonminimum phase system, and can therefore not be causally inverted as this would lead to an unstable realization. Nevertheless, a noncausal stable inverse may exist. Whether the impulse response is a minimum phase system depends on the reverberation level.

Acoustic impulse responses are characterized by a dead time followed by a large number of reflections. The dead time is the time needed for the acoustic wave to propagate from source to listener via the shortest, direct acoustic path. After the direct path impulse a set of early reflections are encountered, whose amplitude and delay are strongly determined by the shape of the recording room and the position of source and listener. Next come a set of late reflections, also called reverberation, which decay exponentially in time. These impulses stem from multipath propagation as acoustic waves reflect off walls and objects in the recording room. As objects in the recording room can move, acoustic impulse responses are typically highly time-varying.

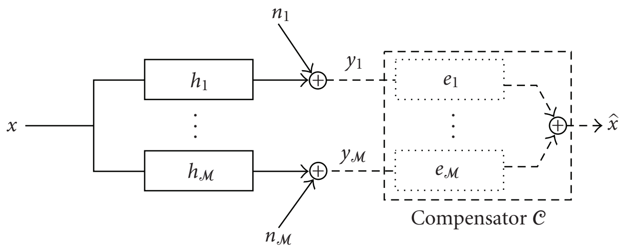


FIGURE 1: Multichannel speech dereverberation setup: a speech signal x is filtered by acoustic impulse responses $h_1 \cdots h_{\mathcal{M}}$, resulting in $\mathcal{M}$ microphone signals $y_1 \cdots y_{\mathcal{M}}$. Typically, also some background noises $n_1 \cdots n_{\mathcal{M}}$ are picked up by the microphones. Dereverberation is aimed at finding the appropriate compensator $\mathcal{C}$ to retrieve the original speech signal x and to undo the filtering by the impulse responses h_m.

Although signals (music, e.g.) may sound more pleasant when reverberation is added, (especially for speech signals), the intelligibility is typically reduced. In order to cope with this kind of deformation, dereverberation or deconvolution techniques are called for. Whereas enhancement techniques for acoustic echo and noise reduction are well known in the literature, high-quality, computationally efficient dereverberation algorithms are, to the best of our knowledge, not yet available.

A general $\mathcal{M}$-channel speech dereverberation system is shown in Figure 1. An unknown speech signal x is filtered by unknown acoustic impulse responses $h_1 \cdots h_{\mathcal{M}}$, resulting in $\mathcal{M}$ microphone signals $y_1 \cdots y_{\mathcal{M}}$. In the most general case, also noises $n_1 \cdots n_{\mathcal{M}}$ are added to the filtered speech signals. The noises can be spatially correlated, or uncorrelated. Spatially correlated noises typically stem from a noise source positioned somewhere in the room.

Dereverberation is aimed at finding the appropriate compensator $\mathcal{C}$ such that the output $\hat{x}$ is close to the unknown signal x. If $\hat{x}$ approaches x, the added reverberation and noises are removed, leading to an enhanced, dereverberated output signal. In many cases, the compensator $\mathcal{C}$ is linear, hence $\mathcal{C}$ reduces to a set of linear dereverberation filters $e_1 \cdots e_{\mathcal{M}}$ such that

$$\hat{x} = \left(\sum_{m=1}^{\mathcal{M}} e_m \star h_m \right) \star x. \tag{1}$$

In the following section, a number of representative dereverberation algorithms are presented that can be cast in the framework of Figure 1. All of these approaches, except the cepstrum-based techniques discussed in Section 3.3, are linear, and can hence be described by linear dereverberation filters $e_1 \cdots e_{\mathcal{M}}$.

3. DEREVERBERATION ALGORITHMS

In this section, a number of representative, wellknown dereverberation techniques are reviewed and some more recently developed algorithmic solutions are presented. The different algorithms are described and references to the literature are given. Furthermore, it is pointed out which parameter settings are applied for the simulations and comparison tests.

3.1. Beamforming

By appropriately filtering and combining different microphone signals a spatially dependent amplification is obtained, leading to so-called acoustic beamforming techniques [11]. Beamforming is primarily employed to suppress background noise, but can be applied for dereverberation purposes as well: as beamforming algorithms spatially focus on the signal source of interest (speaker), waves coming from other directions (e.g., higher-order reflections) are suppressed. In this way, a part of the reverberation can be reduced.

A basic but, nevertheless, very popular beamforming scheme is the *delay-and-sum beamformer* [17]. The microphones are typically placed on a linear, equidistant array and the different microphone signals are appropriately delayed and summed. Referring to Figure 1, the output of the delay-and-sum beamformer is given by

$$\hat{x}[k] = \sum_{m=1}^{\mathcal{M}} y_m[k - \Delta_m]. \tag{2}$$

The inserted delays are chosen in such a way that signals arriving from a specific direction in space (steering direction) are amplified, and signals coming from other directions are suppressed. In a digital implementation, however, Δ_m are integers, and hence the number of feasible steering directions is limited. This problem can be overcome by replacing the delays by non-integer-delay (interpolation) filters at the expense of a higher implementation cost. The interpolation filters can be implemented as well in the time as in the frequency domain.

The spatial selectivity that is obtained with (2) is strongly dependent on the frequency content of the incoming acoustic wave. Introducing frequency-dependent microphone weights may offer more constant beam patterns over the frequency range of interest. This leads to the so-called *"filter-and-sum beamformer"* [10, 18]. Whereas the form of the beam pattern and its uniformity over the frequency range of interest can be fairly well controlled, the frequency selectivity, and hence the expected dereverberation capabilities, mainly depend on the number of microphones that is used. In many practical systems, however, the number of microphones is strongly limited, and therefore also the spatial selectivity and dereverberation capabilities of the approach.

Extra noise suppression can be obtained with *adaptive beamforming* structures [9, 11], which combine classical beamforming with adaptive filtering techniques. They outperform classical beamforming solutions in terms of achievable noise suppression, and show, thanks to the adaptivity, increased robustness with respect to nonstatic, that is, time-varying environments. On the other hand, adaptive beamforming techniques are known to suffer from signal leakage, leading to significant distortion of the signal of interest. This effect is clearly noticeable in highly reverberating environments, where the signal of interest arrives at the microphone array basically from all directions in space. This makes adaptive beamforming techniques less attractive to be used as dereverberation algorithms in highly acoustically reverberating environments.

For the dereverberation experiments discussed in Section 5, we rely on the basic scheme, the delay-and-sum beamformer, which serves as a very cheap reference algorithm. During our simulations, it is assumed that the signal of interest (speaker) is in front of the array, in the far field, that is, not too close to the array. Under this realistic assumption all Δ_m can be set to zero. More advanced beamforming structures have also been considered, but showed only marginal improvements over the reference algorithm under realistic parameters settings.

3.2. Unnormalized matched filtering

Unnormalized matched filtering is a popular technique used in digital communications to retrieve signals after transmission amidst additive noise. It forms the basis of more advanced deconvolution techniques that are discussed in Sections 3.4.2 and 3.6, and has been included in this paper mainly to serve as a reference.

The underlying idea of unnormalized matched filtering is to convolve the transmitted (microphone) signal with the inverse of the transmission path. Assuming that the transmission paths h_m are known (see Figure 1), an enhanced system output can indeed be obtained by setting $e_m[k] = h_m[-k]$ [17]. In order to reduce complexity the dereverberation filters $e_m[k]$ have to be truncated, that is, the l_e most significant (typically, the last l_e) coefficients of $h_m[-k]$ are retained. In our experiments, we choose $l_e = 1000$, irrespective of the length of the transmission paths. Observe that even if $l_e \rightarrow \infty$, significant frequency distortion is introduced, as $|\sum_m \underline{h}_m(f)^* \underline{h}_m(f)|$ is typically strongly frequency-dependent. It is hence not guaranteed that the resulting signal will sound better than the original reverberated speech signal. Another disadvantage of this approach is that the filters h_m have to be known in advance. On the other hand, it is known that matched filtering techniques are quite robust against additive noise [17]. During the simulations we provide the true impulse responses h_m as an extra input to the algorithm to evaluate the algorithm under ideal circumstances. In the case of experiments with real-life data the impulse responses are estimated with an NLMS adaptive filter based on white noise data.

3.3. Cepstrum-based dereverberation

Reverberation can be considered as a convolutional noise source, as it adds an unwanted convolutional factor h, the acoustic impulse response, to the clean speech signal x.

By transforming signals to the cepstral domain, convolutional noise sources can be turned into additive disturbances:

$$y[k] = x[k] \star \underbrace{h[k]}_{\text{unwanted}} \iff y_{\mathrm{rc}}[m] = x_{\mathrm{rc}}[m] + \underbrace{h_{\mathrm{rc}}[m]}_{\text{unwanted}}, \tag{3}$$

where

$$z_{\mathrm{rc}}[m] = \mathcal{F}^{-1}\{\log|\mathcal{F}\{z[k]\}|\} \tag{4}$$

is the real cepstrum of signal $z[k]$ and $\mathcal{F}$ is the Fourier transform. Speech can be considered as a "low quefrent" signal as $x_{\mathrm{rc}}[m]$ is typically concentrated around small values of m. The room reverberation $h_{\mathrm{rc}}[m]$, on the other hand, is expected to contain higher "quefrent" information. The amount of reverberation can hence be reduced by appropriate lowpass "liftering" of $y_{\mathrm{rc}}[m]$, that is, suppressing high "quefrent" information, or through peak picking in the low "quefrent" domain [1, 3].

Extra signal enhancement can be obtained by combining the cepstrum-based approach with multimicrophone beamforming techniques [11] as described in [2, 4]. The algorithm described in [2], for instance, factors the input signals into a minimum-phase and an allpass component. As the minimum-phase components appear to be least affected by the reverberation, the minimum-phase cepstra of the different microphone signals are averaged and the resulting signal is further enhanced with a lowpass "lifter." On the allpass components, on the other hand, a spatial filtering (beamforming) operation is performed. The beamformer reduces the effect of the reverberation, which acts as uncorrelated additive noise to the allpass components.

Cepstrum-based dereverberation assumes that the speech and the acoustics can be clearly separated in the cepstral domain, which is not a valid assumption in many realistic applications. Hence, the proposed algorithms can only be successfully applied in simple reverberation scenarios, that is, scenarios for which the speech is degraded by simple echoes. Furthermore, cepstrum-based dereverberation is an inherently nonlinear technique, and can hence not be described by linear dereverberation filters $e_1 \cdots e_{\mathcal{M}}$, as shown in Figure 1.

The algorithm that is used in our experiments is based on [2]. The two key algorithmic parameters are the frame length L and the number of low "quefrent" cepstral coefficients n_c that are retained. We found that $L = 128$ and $n_c = 30$ lead to good perceptual results. Making n_c too small leads to unacceptable speech distortion. With too large values of n_c, the reverberation cannot be reduced sufficiently.

3.4. Blind subspace-based system identification and dereverberation

Over the last years, many blind subspace-based system identification techniques have been developed for channel equalization in digital communications [13, 14]. These techniques are also applied to speech dereverberation, as shown in this section.

3.4.1. Data model

Consider the $\mathcal{M}$-channel speech dereverberation setup of Figure 1. Assume that $h_1 \cdots h_{\mathcal{M}}$ are FIR filters of length N and that $e_1 \cdots e_{\mathcal{M}}$ are FIR filters of length L. Then,

$$\begin{aligned}&\hat{x}[k]\\ &\quad = \underbrace{[e_1[0] \ \cdots \ e_1[L-1] \mid \cdots \mid e_{\mathcal{M}}[0] \ \cdots \ e_{\mathcal{M}}[L-1]]}_{\mathbf{e}^T}\,\mathbf{y}[k],\end{aligned} \tag{5}$$

with

$$\mathbf{y}[k] = \mathbf{H} \cdot \mathbf{x}[k], \tag{6}$$

$$\begin{aligned}\mathbf{y}[k] = [&y_1[k] \ \cdots \ y_1[k-L+1] \mid \cdots \mid y_{\mathcal{M}}[k]\\ &\cdots \ y_{\mathcal{M}}[k-L+1]]^T,\end{aligned} \tag{7}$$

$$\begin{aligned}\mathbf{x}[k] &= [x[k]\,x[k-1] \ \cdots \ x[k-L-N+2]]^T,\\ \mathbf{H} &= \left[\mathbf{H}_1^T \ \cdots \ \mathbf{H}_{\mathcal{M}}^T\right]^T,\end{aligned} \tag{8}$$

$$\mathbf{H}_m \overset{\forall m}{=} \begin{bmatrix} \boxed{\mathbf{h}_m^T} & & & \\ & \boxed{\mathbf{h}_m^T} & & \\ & & \ddots & \\ & & & \boxed{\mathbf{h}_m^T} \end{bmatrix}, \qquad \mathbf{h}_m \overset{\forall m}{=} \begin{bmatrix} h_m[0] \\ \vdots \\ h_m[N-1] \end{bmatrix}. \tag{9}$$

3.4.2. Zero-forcing algorithm

Perfect dereverberation, that is, $\hat{x}[k] = x[k-n]$ can be achieved if

$$\mathbf{e}_{\mathrm{ZF}}^T \cdot \mathbf{H} = \left[\mathbf{0}_{1\times n} \ \ 1 \ \ \mathbf{0}_{1\times(L+N-2-n)}\right] \tag{10}$$

or

$$\mathbf{e}_{\mathrm{ZF}}^T = \left[\mathbf{0}_{1\times n} \ \ 1 \ \ \mathbf{0}_{1\times(L+N-2-n)}\right]\mathbf{H}^{\dagger}, \tag{11}$$

where $\mathbf{H}^{\dagger}$ is the pseudoinverse of $\mathbf{H}$. From (11) the filter coefficients $e_m[l]$ can be computed if $\mathbf{H}$ is known. Observe that (10) defines a set of $L + N - 1$ equations in $\mathcal{M}L$ unknowns. Hence, only if

$$L \geq \frac{N-1}{\mathcal{M}-1} \tag{12}$$

and $h_1 \cdots h_{\mathcal{M}}$ are known exactly, perfect dereverberation can be obtained. Under this assumption (11) can be written as [19]

$$\mathbf{e}_{\mathrm{ZF}}^T = \left[\mathbf{0}_{1\times n} \ \ 1 \ \ \mathbf{0}_{1\times(L+N-2-n)}\right]\left(\mathbf{H}^H\mathbf{H}\right)^{-1}\mathbf{H}^H. \tag{13}$$

If $\mathbf{y}[k]$ is multiplied by $\mathbf{e}_{\text{ZF}}^T$, one can view the multiplication with the right-most $\mathbf{H}^H$ in (13) as a time-reversed filtering with h_m, which is a kind of matched filtering operation (see Section 3.2). It is known that matched filtering is mainly effective against noise. The matrix inverse $(\mathbf{H}^H\mathbf{H})^{-1}$, on the other hand, performs a normalization and compensates for the spectral shaping and hence reduces reverberation.

In order to compute $\mathbf{e}_{\text{ZF}}$ the transmission matrix $\mathbf{H}$ has to be known. If $\mathbf{H}$ is known only within a certain accuracy, small deviations on $\|\mathbf{H}\|$ can lead to large deviations on $\|\mathbf{H}^\dagger\|$ if the condition number of $\mathbf{H}$ is large. This affects the robustness of the zero-forcing (ZF) approach in noisy environments.

3.4.3. Minimum mean-squared error algorithm

When both reverberation and noise are added to the signal, minimum mean-squared error (MMSE) equalization may be more appropriate. If noise is present on the sensor signals the data model of (6) can be extended to

$$\mathbf{y}[k] = \mathbf{H} \cdot \mathbf{x}[k] + \mathbf{n}[k] \tag{14}$$

with

$$\begin{aligned}\mathbf{n}[k] = [&n_1[k] \ \cdots \ n_1[k-L+1] \mid \cdots \mid n_{\mathcal{M}}[k] \\ &\cdots \ n_{\mathcal{M}}[k-L+1]]^T.\end{aligned} \tag{15}$$

A noise robust dereverberation algorithm is then obtained by minimizing the following MMSE criterion:

$$J = \min_{\mathbf{e}} \mathcal{E}\{|\hat{x}[k] - x[k-n]|^2\}, \tag{16}$$

where $\mathcal{E}\{\cdot\}$ is the expectation operator. Inserting (5) and setting ∇J to $\mathbf{0}$ leads to [19]

$$\mathbf{e}_{\text{MMSE}}^T = \mathcal{E}\{x[k-n]\mathbf{y}[k]^H\}\mathcal{E}\{\mathbf{y}[k]\mathbf{y}[k]^H\}^{-1}. \tag{17}$$

If it is assumed that the noises n_m and the signal of interest x are uncorrelated, it follows from (14) that (17) can be written as

$$\begin{aligned}\mathbf{e}_{\text{MMSE}}^T = \left[\mathbf{0}_{1\times n} \mid 1 \mid \mathbf{0}\right] \mathbf{H}^\dagger (&\mathcal{E}\{\mathbf{y}[k]\mathbf{y}[k]^H\} \\ &- \mathcal{E}\{\mathbf{n}[k]\mathbf{n}[k]^H\})\mathcal{E}\{\mathbf{y}[k]\mathbf{y}[k]^H\}^{-1}\end{aligned} \tag{18}$$

if $(\mathcal{M}-1)L \geq N-1$ (see (12)).

Matrix $\mathcal{E}\{\mathbf{y}[k]\mathbf{y}[k]^H\}$ can be easily computed based on the recorded microphone signals, whereas $\mathcal{E}\{\mathbf{n}[k]\mathbf{n}[k]^H\}$ has to be estimated during noise-only periods, when $y_m[k]=n_m[k]$. Observe that the MMSE algorithm approaches the zero-forcing algorithm in the absence of noise, that is, (18) reduces to (11), provided that $\mathcal{E}\{\|\mathbf{y}[k]\mathbf{y}[k]^H\|\} \gg \mathcal{E}\{\|\mathbf{n}[k]\mathbf{n}[k]^H\|\}$. Whereas the MMSE algorithm is more robust to noise, in general it achieves less dereverberation than the zero-forcing algorithm. Compared to (11), extra computational power is required for the updating of the correlation matrices and the computation of the right-hand part of (18).

3.4.4. Multichannel subspace identification

So far it was assumed that the transmission matrix $\mathbf{H}$ is known. In practice, however, $\mathbf{H}$ has to be estimated. To this aim $L \times K$ Toeplitz matrices

$$\begin{aligned}&\mathbf{Y}_m[k] \\ &\overset{\forall m}{=} \begin{bmatrix} y_m[k-K+1] & y_m[k-K+2] & \cdots & y_m[k] \\ y_m[k-K] & y_m[k-K+1] & \cdots & y_m[k-1] \\ \vdots & \ddots & \ddots & \vdots \\ y_m[k-K-L+2] & y_m[k-K-L+3] & \cdots & y_m[k-L+1] \end{bmatrix}\end{aligned} \tag{19}$$

are defined. If we leave out the noise contribution for the time being, it follows from (5)–(8) that

$$\begin{aligned}\mathbf{Y}[k] &= \left[\mathbf{Y}_1^T[k] \ \cdots \ \mathbf{Y}_{\mathcal{M}}^T[k]\right]^T \\ &= \mathbf{H} \underbrace{\left[\mathbf{x}[k-K+1] \ \cdots \ \mathbf{x}[k]\right]}_{\mathbf{X}[k]}.\end{aligned} \tag{20}$$

If $L \geq N$,

$$\begin{aligned}\mathbf{v}_{mn} = [&\mathbf{0}_{1\times(n-1)L} \mid \mathbf{h}_m^T \, \mathbf{0}_{1\times(L-N)} \mid \mathbf{0}_{1\times(m-n-1)L} \mid \\ &- \mathbf{h}_n^T \, \mathbf{0}_{1\times(L-N)} \mid \mathbf{0}_{1\times(\mathcal{M}-m)L}]^T\end{aligned} \tag{21}$$

can be defined. Then, for each pair (n, m) for which $1 \leq n < m \leq \mathcal{M}$, it is seen that

$$\mathbf{v}_{mn}^T\mathbf{H}\mathbf{X}[k] = \mathbf{v}_{mn}^T\mathbf{Y}[k] = \mathbf{0}, \tag{22}$$

as $\mathbf{v}_{mn}^T\mathbf{H} = [w_{mn}[0] \ \cdots \ w_{mn}[2N-2] \ 0 \ \cdots \ 0]$, where $w_{mn} = h_m \star h_n - h_n \star h_m$ is equal to zero. Hence, $\mathbf{v}_{mn}$ and therefore also the transmission paths can be found in the left null space of $\mathbf{Y}[k]$, which has dimension

$$\nu = \mathcal{M}L - \underbrace{\text{rank}\{\mathbf{Y}[k]\}}_{r}. \tag{23}$$

By appropriately combining the ν basis vectors[1] $\mathbf{v}_\rho$, $\rho = r+1 \cdots \mathcal{M}L$, which span the left null space of $\mathbf{Y}[k]$, the filter h_m can be computed up to within a constant ambiguity factor α_m. This can, for instance, be done by solving the following set of equations:

$$\left[\mathbf{v}_{r+1} \ \cdots \ \mathbf{v}_{\mathcal{M}L}\right] \begin{bmatrix} \beta_{r+1}^{(m)} \\ \vdots \\ \beta_{\mathcal{M}L-1}^{(m)} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_m\mathbf{h}_m \\ \mathbf{0}_{(L-N)\times 1} \\ \hline \mathbf{0}_{(m-2)L\times 1} \\ \hline -\alpha_m\mathbf{h}_1 \\ \mathbf{0}_{(L-N)\times 1} \\ \hline \mathbf{0}_{(\mathcal{M}-m)L\times 1} \end{bmatrix} \quad \forall m : 1 < m \leq \mathcal{M}. \tag{24}$$

[1] Assuming $\mathbf{Y}^T[k] \overset{\text{SVD}}{=} \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$, $\mathbf{V} = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_r \ \ \mathbf{v}_{r+1} \ \cdots \ \mathbf{v}_{\mathcal{M}L}]$ is the singular value decomposition of $\mathbf{Y}^T[k]$.

It can be proven [20] that an exact solution to (24) exists in the noise-free case if $\mathcal{M}L \geq L+N-1$. If noise is present, (24) has to be solved in a least-square sense. In order to eliminate the different ambiguity factors α_m, it is sufficient to compare the coefficients of, for example, $\alpha_2\mathbf{h}_1$ with $\alpha_m\mathbf{h}_1$ for $m > 2$. In this way, the different scaling factors α_m can be compensated for, such that only a single overall ambiguity factor α remains.

3.4.5. *Channel-order estimation*

From (24) the transmission paths h_m can be computed [13], provided that the length of the transmission paths (channel order) N is known. It can be proven [20] that for generic systems for which $K \geq L+N-1$ and $L \geq (N-1)/(\mathcal{M}-1)$ (see (12)) the channel order can be found from

$$N = \text{rank}\,\{\mathbf{Y}[k]\} - L + 1, \tag{25}$$

provided that there is no noise added to the system. Furthermore, once N is known, the transmission paths can be found based on (24) if $L \geq N$ and $K \geq L+N-1$, as shown in [20].

If there is noise in the system one typically attempts to identify a "gap" in the singular value spectrum to determine the rank of $\mathbf{Y}[k]$. This gap is due to a difference in amplitude between the large singular values, which are assumed to correspond to the desired signal, and the smaller, noise-related singular values. Finding the correct system order is typically the Achilles heel, as any system order mismatch usually leads to an important decrease in the overall performance of the dereverberation algorithm. Whereas for adaptive filtering applications, for example, small errors on the system order typically lead to a limited and controllable performance decrease, in the case of subspace identification unacceptable performance drops are easily encountered, even if the error on the system order is small.

This is illustrated by the following example: consider a 2-channel system (cf. Figure 1) with transmission paths h_1 and h_2 being random 10-taps FIR filters with exponentially decaying coefficients. To the system white noise is input. Filter h_1 was adjusted such that the DC response equals 1. With this example the robustness of blind subspace identification against order mismatches is assessed under noiseless conditions. Thereto, h_1 and h_2 are identified with the subspace identification method described in Section 3.4.4, compensating for the ambiguity to allow a fair comparison. Additionally, the transmission paths are estimated with an NLMS adaptive filter. In order to check the robustness of both approaches against order estimate errors, the length of the estimation filters N is changed from 4, 8, and 9 (underestimates) to 12 (overestimate). The results are plotted in Figure 2. The solid line corresponds to the frequency response of the 10-taps filter h_1. The dashed line shows the frequency response of the N-taps subspace estimate. The dashed-dotted line represents the frequency response of the N-taps NLMS estimate. It was verified that for $N = 10$ both methods identify the correct transmission paths h_1 and h_2, as predicted by theory. In the case of a channel-order overestimate (subplot 4), it is observed that h_1 and h_2 are correctly estimated by the NLMS approach. Also the subspace algorithm provides correct estimates, be it up to a common (filter) factor. This common factor can be removed using (24). In the case of a channel order underestimate (subplots 1–3) the NLMS estimates are clearly superior to those of the subspace method. Whereas the performance of the adaptive filter gradually deteriorates with decreasing values of N, the behavior of the subspace identification method more rapidly deviates from the theoretical response.

In a second example, a white noise signal x is filtered by two impulse responses h_1 and h_2 of 10 filter taps each. Additionally, uncorrelated white noise is added to $h_1 \star x$ and $h_2 \star x$ at different signal-to-noise ratios. The system order is estimated based on the singular value spectrum of $\mathbf{Y}$. For this experiment $L = 20$ and $K = 40$. In Figure 3, the 10-logarithm of the singular value spectrum is shown for different signal-to-noise ratios. From (25) it follows that rank$\{\mathbf{Y}[k]\} = 29$. In each subplot therefore the 29th singular value is encircled. Remark that for low, yet realistic signal-to-noise ratios such as 0 dB and 20 dB, there is no clear gap between the signal-related singular values and the noise-related singular values.

Even when the system order is estimated correctly the system estimates $\widehat{h}_1$ and $\widehat{h}_2$ differ from the true filters h_1 and h_2. To illustrate this a white noise signal x is filtered by two random impulse responses h_1 and h_2 of 20 filter taps each. White noise is added to $h_1 \star x$ and $h_2 \star x$ at different signal-to-noise ratios, leading to y_1 and y_2. Based on y_1 and y_2 the impulse responses $\widehat{h}_1$ and $\widehat{h}_2$ are estimated following (24) and setting L equal to N. In Figure 4, the angle between h_1 and $\widehat{h}_1$ is plotted in degrees as a function of the signal-to-noise ratio. The angle has been projected onto the first quadrant $(0 \rightarrow 90^\circ)$ as due to the inherent ambiguity, blind subspace algorithms can solely estimate the orientation of the impulse response vector, and not the exact amplitude or sign. Observe that the angle between h_1 and $\widehat{h}_1$ is small only at high signal-to-noise ratios. Remark furthermore that for low signal-to-noise ratios the angle approaches 90°.

3.4.6. *Implementation and cost*

The dereverberation and the channel estimation procedures discussed in Sections 3.4.2, 3.4.3, and 3.4.4 tend to give rise to a high algorithmic cost for parameter settings that are typically used for speech dereverberation. Advanced matrix operations are required, which result in a computational cost of the order of $\mathcal{O}(N^3)$, where N is the length of the unknown transmission paths, and a memory storage capacity that is $\mathcal{O}(N^2)$. This leads to computational and memory requirements that exceed the capabilities of many modern computer systems.

In our simulations the length of the impulse response filters, that is, N, is computed following (25) with $K = 2N_{\text{max}}$ and $L = N_{\text{max}}$, where rank$\{\mathbf{Y}[k]\}$ is determined by looking for a gap in the singular value spectrum. In this way, the impulse response filter length N is restricted to N_{max}.

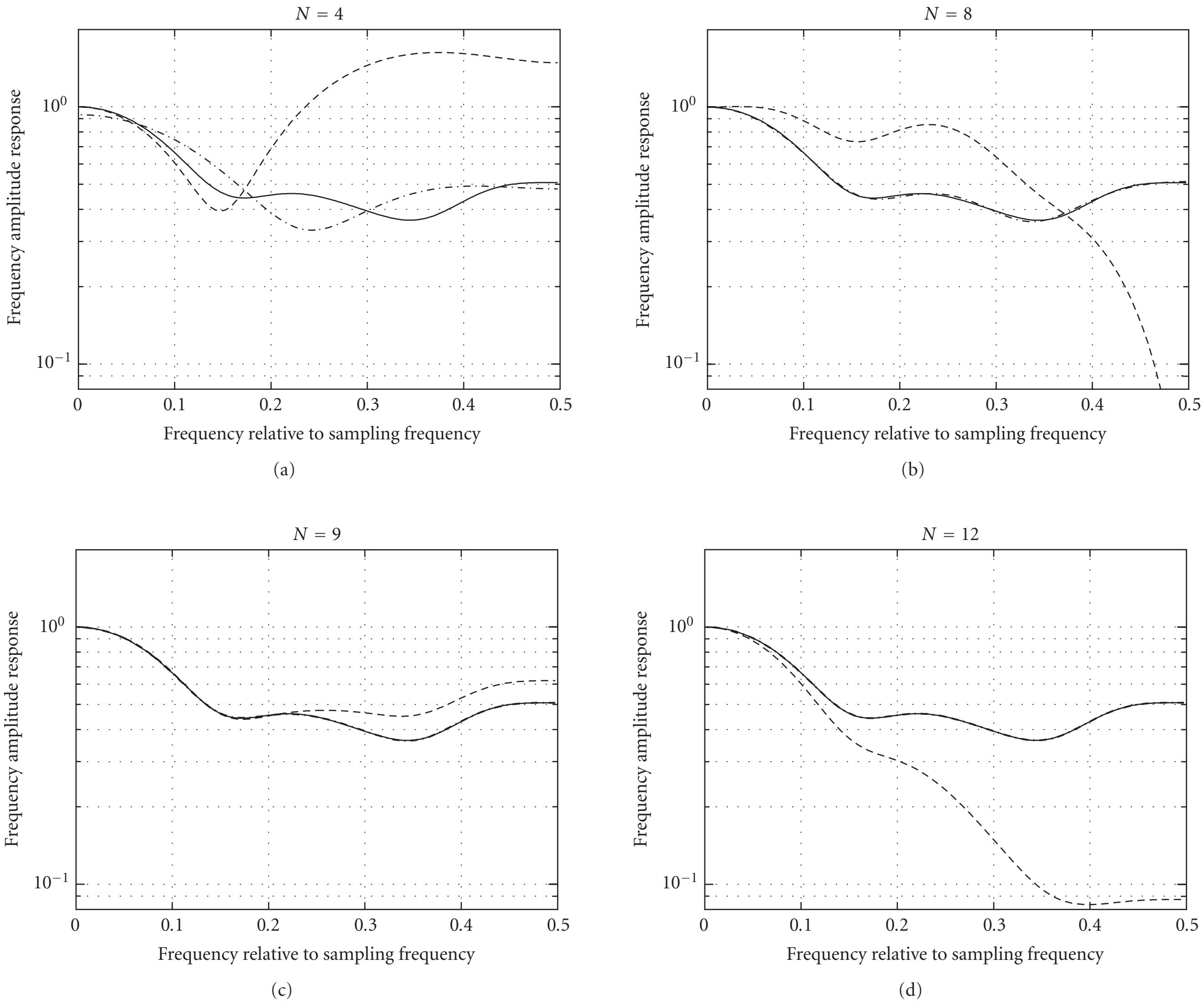


Figure 2: Robustness of 2-channel system identification against order estimate errors: 10-taps filters h_1 and h_2 are identified with a blind subspace identification method and an NLMS adaptive filter. The length of the estimation filters N was changed from 4, 8, and 9 (underestimates) to 12 (overestimate). The solid line corresponds to the frequency response of the 10-taps filter h_1. The dashed line shows the frequency response of the N-taps subspace estimate. The dashed-dotted line represents the frequency response of the N-taps NLMS estimate. Whereas the performance of the adaptive filter gradually deteriorates with decreasing values of N, the behavior of the subspace identification method more rapidly deviates from the theoretical response.

The impulse responses are computed with the algorithm of Section 3.4.4, with $K = 5N_{\max}$ and $L = N$. For the computation of the dereverberation filters, we rely on the zero-forcing algorithm of Section 3.4.2 with $n = 1$ and $L = \lceil N/(\mathcal{M} - 1)\rceil$. Several values have been tried for n, but changing this parameter hardly affected the performance of the algorithms. Most experiments have been done with $N_{\max} = 100$, restricting the impulse response filter length N to 100. This leads to fairly small matrix sizes, which however already demand considerable memory consumption and simulation time. To investigate the effect of larger matrix sizes and hence longer impulse responses, additional simulations have been done with $N_{\max} = 300$. Values of $N_{\max}$ larger than 300 will quickly lead to a huge memory consumption and unacceptable simulation times without additionally enhancing the signal (see also Section 5.1).

3.5. Subband-domain subspace-based dereverberation

3.5.1. Subband implementation scheme

To overcome the high computational and memory requirements of the time-domain subspace approach of Section 3.4, subband processing can be put forward as an alternative. In a subband implementation all microphone signals $y_m[k]$

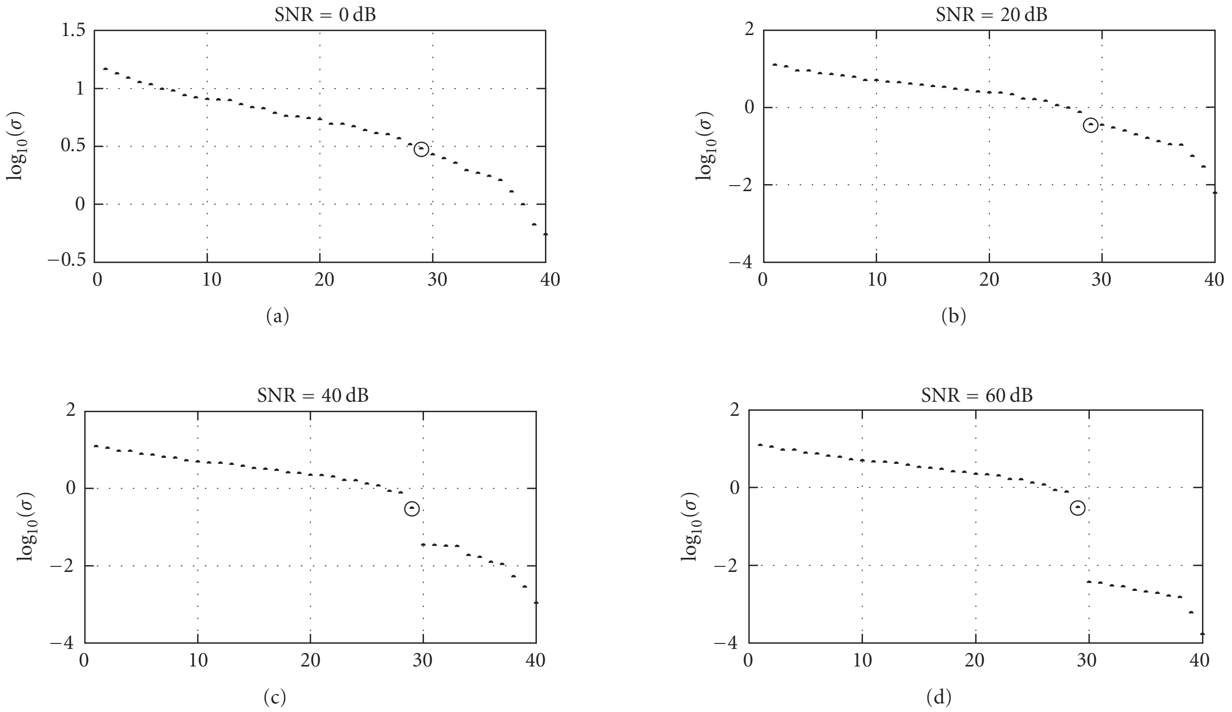


Figure 3: Subspace-based system identification: singular value spectrum of the block-Toeplitz data matrix $\mathbf{Y}$ at different signal-to-noise ratios. The system under test is a 9th-order, 2-channel FIR system ($N = 10$, $\mathcal{M} = 2$) with white noise input. Additionally, uncorrelated white noise is added to the microphone signals at different signal-to-noise ratios. Remark that for low, yet realistic signal-to-noise ratios such as 0 dB and 20 dB, there is no clear gap between the signal-related singular values and the noise-related singular values.

are fed into identical analysis filter banks $\{a_0, \ldots, a_{P-1}\}$, as shown in Figure 5. All subband signals are subsequently D-fold subsampled. The processed subband signals are upsampled and recombined in the synthesis filter bank $\{s_0, \ldots, s_{P-1}\}$, leading to the system output $\hat{x}$. As the channel estimation and equalization procedure are performed in the subband domain at a reduced sampling rate, a substantial cost reduction is expected.

3.5.2. *Filter banks*

To reduce the amount of overall signal distortion that is introduced by the filter banks and the subsampling, perfect or nearly perfect reconstruction filter banks are employed [21, 22]. Oversampled filter banks ($P > D$) are used to minimize the amount of aliasing distortion that is added to the subband signals during the downsampling. DFT modulated filter bank schemes are then typically preferred. In many applications very simple so-called DFT filter banks are used [22].

3.5.3. *Ambiguity elimination*

With blind system identification techniques the transmission paths can only be estimated up to a constant factor. Contrary to the fullband approach where a global uncertainty factor α is encountered (see Section 3.4.4), in a subband implementation there is an ambiguity factor $\alpha^{(p)}$ in each subband. This leads to significant signal distortion if the ambiguity factors $\alpha^{(p)}$ are not compensated for.

Rahbar et al. [23] proposed a noise robust method to compensate for the subband-dependent ambiguity that occurs in frequency-domain subspace dereverberation with 1-tap compensation filters. An alternative method is proposed in [20], which can also handle higher-order frequency-domain compensation filters. These ambiguity elimination algorithms are quite computationally demanding, as the eigenvalue or the singular value decomposition has to be computed of a large matrix. It further appears that the ambiguity elimination methods are sensitive to system order mismatches.

In the simulations, we apply a frequency-domain subspace dereverberation scheme with the DFT-IDFT as analysis/synthesis filter bank and 1-tap subband models. Further, $P = 512$ and $D = 256$, so that effectively 256-tap time-domain filters are estimated in the frequency domain. For the subband channel estimation the blind subspace-based channel estimation algorithm of Section 3.4.4 is used with $N = 1$, $L = 1$, and $K = 5$. For the dereverberation the zero-forcing algorithm of Section 3.4.2 is employed with $L = 1$ and $n = 1$. The ambiguity problem that arises in the subband approach is compensated for based on the technique that is described in [20] with $N = 256$ and $P = 512$.

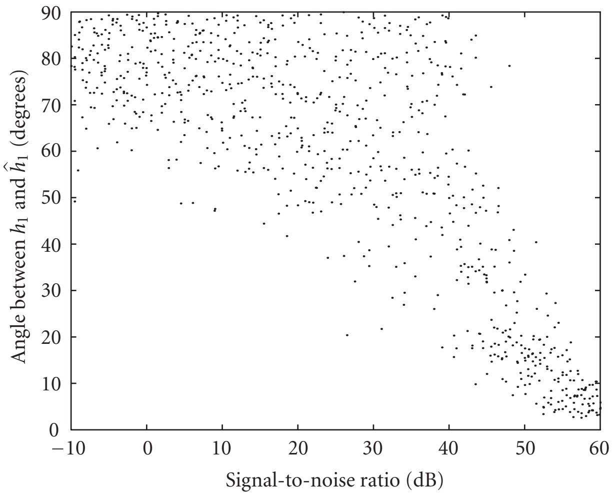


Figure 4: Subspace-based system identification: angle between h_1 and $\widehat{h}_1$ as a function of the signal-to-noise ratio for a random 19th-order, 2-channel system with white noise input (141 realizations are shown). Uncorrelated white noise is added to the microphone signals at different signal-to-noise ratios. The angle between h_1 and $\widehat{h}_1$ has been projected onto the first quadrant ($0 \rightarrow 90^\circ$) as due to the inherent ambiguity, blind subspace algorithms can solely estimate the orientation of the impulse response vector, and not the exact amplitude or sign. Observe that the angle between h_1 and $\widehat{h}_1$ is small only at high signal-to-noise ratios. Remark furthermore that for low signal-to-noise ratios the angle approaches 90°.

3.5.4. Cost reduction

If there are P subbands that are D-fold subsampled, one may expect that the transmission path length reduces to N/D in each subband, lowering the memory storage requirements from $\mathcal{O}(N^2)$ (see Section 3.4.6) to $\mathcal{O}(P(N^2/D^2))$. As typically $P \approx D$, it follows that $\mathcal{O}(P(N^2/D^2)) \approx \mathcal{O}(N^2/D)$. As far as the computational cost is concerned not only the matrix dimensions are reduced, also the updating frequency is lowered by a factor D, leading to a huge cost reduction from $\mathcal{O}(N^3)$ to $\mathcal{O}(P(N^3/D^4)) \approx \mathcal{O}(N^3/D^3)$. In practice, however, the cost reduction is less spectacular, as the transmission path length will often have to be larger than N/D to appropriately model the acoustics [24]. Secondly, so far we have neglected the filter bank cost, which will further reduce the complexity gain that can be reached with the subband approach. Nevertheless, a significant overall cost reduction can be obtained, given the $\mathcal{O}(N^3)$ dependency of the algorithm.

Summarizing, the advantages of a subband implementation are the substantial cost reduction and the decoupled subband processing, which is expected to give rise to improved performance. The disadvantages are the frequency-dependent ambiguity, the extra processing delay, as well as possible signal distortion and aliasing effects caused by the subsampling [24].

3.6. Frequency-domain subspace-based matched filtering

In [12] a promising dereverberation algorithm was presented that relies on 1-dimensional frequency-domain subspace tracking. An LMS-type updating scheme was proposed that offers a low-cost alternative to the matrix-based algorithms of Section 3.4.

The 1-dimensional frequency-domain subspace tracking algorithm builds upon the following frequency-dependent data model (compare with (14)) for each frequency f and each frame n:

$$\begin{aligned}\underline{\mathbf{y}}^{[n]}(f) &= \underbrace{\left[\underline{h}_1^{[n]}(f) \ \cdots \ \underline{h}_{\mathcal{M}}^{[n]}(f)\right]^T}_{\underline{\mathbf{h}}^{[n]}(f)} \underline{x}^{[n]}(f) \\ &\quad + \underbrace{\left[\underline{n}_1^{[n]}(f) \ \cdots \ \underline{n}_{\mathcal{M}}^{[n]}(f)\right]^T}_{\underline{\mathbf{n}}^{[n]}(f)},\end{aligned} \tag{26}$$

where, for example (similar formulas hold for $\underline{\mathbf{y}}^{[n]}(f)$ and $\underline{\mathbf{n}}^{[n]}(f)$),

$$\underline{x}^{[n]}(f) = \sum_{p=0}^{P-1} x[nP+p]e^{-j2\pi(nP+p)f} \tag{27}$$

if there is no overlap between frames. If it is assumed that the transfer functions $h_m[k] \leftrightarrow \underline{h}_m(f)$ slowly vary as a function of time, $\underline{\mathbf{h}}^{[n]}(f) \approx \underline{\mathbf{h}}(f)$.

To dereverberate the microphone signals, equalization filters $\underline{\mathbf{e}}(f)$ have to be computed such that

$$\underline{r}_t(f) = \underline{\mathbf{e}}^H(f)\underline{\mathbf{h}}(f) = 1. \tag{28}$$

Observe that the matched filter $\underline{\mathbf{e}}(f) = \underline{\mathbf{h}}(f)/\|\underline{\mathbf{h}}(f)\|^2$ is a solution to (28).

For the computation of $\underline{\mathbf{h}}(f)$ and $\underline{\mathbf{e}}(f)$ the $\mathcal{M} \times \mathcal{M}$ correlation matrix $\mathbf{R}_{yy}(f)$ has to be calculated:

$$\begin{aligned}\mathbf{R}_{yy}(f) &= \mathcal{E}\left\{\underline{\mathbf{y}}^{[\mathbf{n}]}(f)\left(\underline{\mathbf{y}}^{[n]}(f)\right)^H\right\} \\ &= \underbrace{\underline{\mathbf{h}}(f)\mathcal{E}\{|\underline{x}^{[n]}(f)|^2\}\underline{\mathbf{h}}^H(f)}_{\mathbf{R}_{xx}(f)} \\ &\quad + \underbrace{\mathcal{E}\left\{\underline{\mathbf{n}}^{[n]}(f)\left(\underline{\mathbf{n}}^{[n]}(f)\right)^H\right\}}_{\mathbf{R}_{nn}(f)},\end{aligned} \tag{29}$$

where it is assumed that the speech and noise components are uncorrelated. It is seen from (29) that the speech correlation matrix $\mathbf{R}_{xx}(f)$ is a rank-1 matrix. The noise correlation matrix $\mathbf{R}_{nn}(f)$ can be measured during speech pauses.

The transfer function vector $\underline{\mathbf{h}}(f)$ can be estimated using the generalized eigenvalue decomposition (GEVD) of the correlation matrices $\mathbf{R}_{yy}(f)$ and $\mathbf{R}_{nn}(f)$,

$$\begin{aligned}\mathbf{R}_{yy}(f) &= \mathbf{Q}(f)\boldsymbol{\Sigma}_y(f)\mathbf{Q}^H(f) \\ \mathbf{R}_{nn}(f) &= \mathbf{Q}(f)\boldsymbol{\Sigma}_n(f)\mathbf{Q}^H(f)\end{aligned} \tag{30}$$

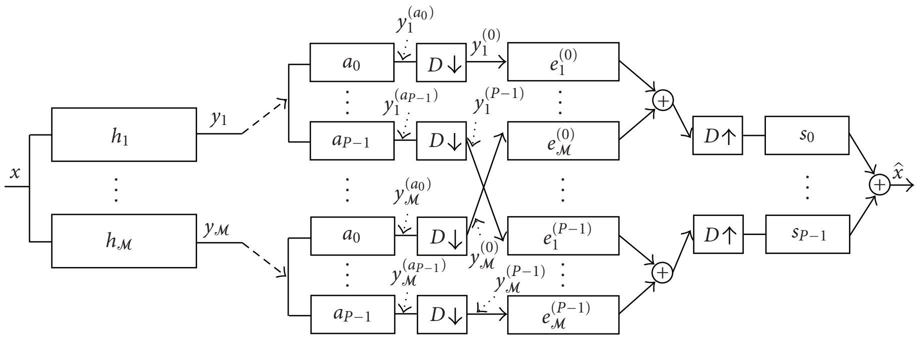


Figure 5: Multi-channel subband dereverberation system: the microphone signals y_m are fed into identical analysis filter banks $\{a_0, \dots, a_{P-1}\}$, and are subsequently D-fold subsampled. After processing the subband signals are upsampled and recombined in the synthesis filter bank $\{s_0, \dots, s_{P-1}\}$, leading to the system output $\hat{x}$.

with $\mathbf{Q}(f)$ an invertible, but not necessarily orthogonal matrix [25]. As the speech correlation matrix

$$\mathbf{R}_{xx}(f) = \mathbf{R}_{yy}(f) - \mathbf{R}_{nn}(f) = \mathbf{Q}(f)(\boldsymbol{\Sigma}_y(f) - \boldsymbol{\Sigma}_n(f))\mathbf{Q}^H(f) \tag{31}$$

has rank 1, it is equal to $\mathbf{R}_{xx}(f) = \sigma_x^2(f)\underline{\mathbf{q}}_1(f)\underline{\mathbf{q}}_1^H(f)$ with $\underline{\mathbf{q}}_1(f)$ the principal generalized eigenvector corresponding to the largest generalized eigenvalue. Since

$$\mathbf{R}_{xx}(f) = \sigma_x^2(f)\underline{\mathbf{q}}_1(f)\underline{\mathbf{q}}_1^H(f) = \mathcal{E}\{|\underline{x}^{[n]}(f)|^2\}\underline{\mathbf{h}}(f)\underline{\mathbf{h}}^H(f), \tag{32}$$

$\underline{\mathbf{h}}(f)$ can be estimated up to a phase shift $e^{j\theta(f)}$ as

$$\hat{\underline{\mathbf{h}}}(f) = e^{j\theta(f)}\underline{\mathbf{h}}(f) = \frac{||\underline{\mathbf{h}}(f)||}{||\underline{\mathbf{q}}_1(f)||}\underline{\mathbf{q}}_1(f)e^{j\theta(f)} \tag{33}$$

if $||\underline{\mathbf{h}}(f)||$ is known. It is assumed that the human auditory system is not very sensitive to this phase shift.

If the additive noise is spatially white, $\mathbf{R}_{nn}(f) = \sigma_n^2\mathbf{I}_{\mathcal{M}}$ and then $\underline{\mathbf{h}}(f)$ can be estimated as the principal eigenvector corresponding to the largest eigenvalue of $\mathbf{R}_{yy}(f)$. It is this algorithmic variant, which assumes spatially white additive noise, that was originally proposed in [12].

Using the matched filter

$$\underline{\mathbf{e}}(f) = \frac{\hat{\underline{\mathbf{h}}}(f)}{||\underline{\mathbf{h}}(f)||^2} = \frac{\underline{\mathbf{q}}_1(f)}{||\underline{\mathbf{q}}_1(f)||||\underline{\mathbf{h}}(f)||}, \tag{34}$$

the dereverberated speech signal $\hat{\underline{x}}^{[n]}(f)$ is found as

$$\begin{aligned}\hat{\underline{x}}^{[n]}(f) &= \underline{\mathbf{e}}^H(f)\underline{\mathbf{y}}^{[n]}(f)\\ &= e^{-j\theta(f)}\underline{x}^{[n]}(f) + \frac{\underline{\mathbf{q}}_1^H(f)}{||\underline{\mathbf{q}}_1(f)||||\underline{\mathbf{h}}(f)||}\underline{\mathbf{n}}^{[n]}(f),\end{aligned} \tag{35}$$

from which the time-domain signal $\hat{x}[k]$ can be computed.

As can be seen from (34), the norm $\boldsymbol{\beta} = ||\underline{\mathbf{h}}(f)||$ has to be known in order to compute $\underline{\mathbf{e}}(f)$. Hence, $\boldsymbol{\beta}$ has to be measured beforehand, which is unpractical, or has to be fixed to an environment-independent constant, for example, $\boldsymbol{\beta} = \mathbf{1}$, as proposed in [12].

The algorithm is expected to fail to dereverberate the speech signal if $\boldsymbol{\beta}$ is not known or is wrongly estimated, as in a matched filtering approach mainly the filtering with the inverse of $||\underline{\mathbf{h}}(f)||^2$ is responsible for the dereverberation (see also Section 3.4.2). Hence, we could claim that the method proposed in [12] is primarily a noise reduction algorithm and that the dereverberation problem is not truly solved.

If the frequency-domain subspace estimation algorithm is combined with the ambiguity elimination algorithm presented in Section 3.5.3, the transmission paths $\underline{h}_m(f)$ can be determined up to within a global scaling factor. Hence, $\boldsymbol{\beta} = ||\underline{\mathbf{h}}(f)||$ can be computed and does not have to be known in advance. Uncertainties on $\boldsymbol{\beta}$, however, which are due to the limited precision of the channel estimation procedure and the "lag error" of the algorithm during tracking of time-varying transmission paths, affect the performance of the subspace tracking algorithm.

In our simulations, we compare two versions of the subspace-based matched filtering approach, both relying on the eigenvalue decomposition of $\mathbf{R}_{yy}(f)$. One variant uses $\boldsymbol{\beta} = \mathbf{1}$ and the other computes $\boldsymbol{\beta}$ as described in Section 3.5.3. For all implementations the block length is set equal to 64, $N = 256$, and the FFT size $P = 512$. To evaluate the algorithm under ideal conditions we simulate a batch version instead of the LMS-like tracking variant of the algorithm proposed in [12].

4. EVALUATION CRITERIA

The performance of the dereverberation algorithms presented in Sections 3.1 to 3.6 has been assessed through a number of experiments that are described in Section 5. For the evaluation, two performance indices have been applied and the ability of the algorithms to enhance the word recognition rate of a speech recognition system has been determined. In this section, the automatic speech recognition system is described and the performance indices are defined that have been used throughout the evaluation.

4.1. Performance indices

For a proper comparison between the different dereverberation procedures, we consider two performance indices, which will be referred to as δ_1 and δ_2. They can be derived from the total response filter

$$r_t = \sum_{m=1}^{\mathcal{M}} e_m \star h_m, \tag{36}$$

where r_t describes the total response from the source signal x to the output $\hat{x}$ if the compensator $\mathcal{C}$ is linear (see Figure 1). Let $\underline{r}_t(f)$ be the frequency response of r_t, then δ_1 is defined as

$$\delta_1 = \frac{\mu_{|\underline{r}_t|}}{\sigma_{|\underline{r}_t|}} \tag{37}$$

with

$$\begin{aligned} \mu_{|\underline{r}_t|} &= \int_{-(1/2)}^{1/2} |\underline{r}_t(f)|\, df, \\ \sigma^2_{|\underline{r}_t|} &= \int_{-(1/2)}^{1/2} \left(|\underline{r}_t(f)| - \mu_{|\underline{r}_t|}\right)^2 df. \end{aligned} \tag{38}$$

In the case of perfect dereverberation, the total response filter r_t is a delay, and hence $|\underline{r}_t(f)|$ is flat. Therefore, with a larger δ_1, more dereverberation is expected. This relative standard deviation measure only takes into account the amplitude of the frequency response of r_t and neglects the phase response.

A more exact measure can be defined in the time domain. If r_t can be represented as an Lth-order FIR filter

$$\mathbf{r}_t = \left[r_t[0] \;\cdots\; r_t[L] \right]^T, \tag{39}$$

performance index δ_2 is defined as

$$\delta_2 = \frac{r_t^{\max}}{||\mathbf{r}_t||}, \tag{40}$$

where

$$r_t^{\max} = \max_{n=0:L}\{|r_t[n]|\}. \tag{41}$$

Here, a unique maximum is assumed, for conciseness. Hence, δ_2^2 corresponds to the energy in the dominant impulse of r_t divided by the total energy in r_t. Again, with a larger δ_2, more dereverberation is expected. It is easily verified that $0 < \delta_2 \leq 1$.

The first part of the evaluation that is presented in this paper relies on simulated impulse responses h_m [26]. Hence, the total response filter can be computed following (36). The second part of the evaluation is based on experiments with recorded real-life data. In that case, the transmission paths $h_1 \cdots h_{\mathcal{M}}$, and so r_t, are unknown, hence the proposed performance indices cannot be applied. However, in the absence of any knowledge about the transmission paths, the total response filter can still be computed based on x and $\hat{x}$, provided that x is known. The impulse responses then are measured offline by inputting white noise to the system and then applying an NLMS adaptive filter.

Note that in the definition of the performance indices δ_1 and δ_2, it is implicitly assumed that the dereverberation algorithm is linear, and therefore can be described by linear dereverberation filters $e_1 \cdots e_{\mathcal{M}}$, as shown in Figure 1. Cepstrum-based dereverberation techniques are inherently nonlinear. They can hence not be described by linear dereverberation filters. Performance indices δ_1 and δ_2 are therefore not defined for the cepstrum-based approach.

4.2. Automatic speech recognition

Objective quality measures to check dereverberation performance are difficult to identify. Apart from the two performance indices defined in Section 4.1, in this paper we rely on the recognition rate of an automatic speech recognizer to compare different algorithms. One of the possible target applications of dereverberation software is indeed speech recognition. Automatic speech recognition systems are typically trained under more or less anechoic conditions. Recognition rates therefore drop whenever signals are applied that are recorded in a moderately or highly reverberant environment. In order to enhance the speech recognition rate, dereverberation software can be used as a preprocessing step to reduce the amount of reverberation that is input to the speech recognition system. In this way, increased recognition rates are hoped for. In this paper, the effect of reverberation on the performance of the speech recognizer is measured and several dereverberation algorithms are evaluated as a means to enhance the recognition rate.

For the recognition experiments [27], a speaker-independent large vocabulary continuous speech recognition system was used that has been developed at the ESAT-PSI research group of Katholieke Universiteit Leuven, Belgium. In this system, the data is sampled at 16 kHz and is first pre-emphasized. Then, every 10 milliseconds, the power spectrum is computed using a window with a time horizon of 30 milliseconds. By means of a nonlinear mel-scaled triangular filterbank, 24 mel-spectrum coefficients are computed and transformed to the log domain. By subtracting the average, the coefficients are mean normalized. In this way, robustness is added against differences in the recording channel. A feature vector with 72 parameters is then constructed by combining the 24 coefficients with their first and second time derivatives. The feature vector is reduced in size and decorrelated, as explained in [28, 29]. A more detailed overview of the acoustic modeling can be found in [27, 30]. The search module is described in [31].

The data set that was used for the speech recognition experiments is the Wall Street Journal November 92 speech recognition evaluation test set [27]. It consists of 330 sentences, amounting to about 33 minutes of speech, uttered by eight different speakers, both male and female. The (clean) data set is recorded at 16 kHz and contains almost no additive noise, nor reverberation. With the recognition system described in the previous paragraph a word error rate (WER)

Table 1: A list of the dereverberation algorithms that have been experimentally evaluated, as presented in Section 5. References are given to previous sections and to the literature, as well as indicative relative complexity numbers for each of the algorithms.

no.	Algorithm	Used graphical symbol	Discussed in section	Reference to the literature	Relative algorithmic complexity
	Unprocessed microphone signal	□	—	—	—
(1)	Delay-and-sum beamforming	◁	Section 3.1	[11, 17]	1.0
(2)	Unnormalized matched filtering	⋆	Section 3.2	[17]	2.7
(3)	Cepstrum-based dereverberation	∗	Section 3.3	[2]	52.7
(4)	Zero-forcing time-domain subspace-based dereverberation	▽	Sections 3.4.2, 3.4.4, 3.4.5	[20]	121.6
(5)	Zero-forcing frequency-domain subspace-based dereverberation	△	Section 3.5	[20]	192.3
(6)	Matched filtering subspace-based dereverberation, $\boldsymbol{\beta} = \mathbf{1}$	○	Section 3.6	[12]	14.8
(7)	Matched filtering subspace-based dereverberation, $\boldsymbol{\beta}$ computed	×	Sections 3.6, 3.5.3	[12, 20]	223.1

of 1.9% can be obtained on the clean Wall Street Journal benchmark test set.

It is important to note that the speech recognizer is trained on the clean, noise-free, unreverberated data, and that the recognition system does not dispose of any special features to combat additive noise or reverberation. Hence, the improvements in word error rate that are observed during simulation are entirely due to the preprocessing signal enhancement tools that are added to the system. Better word error rates may possibly be obtained if the recognizer was trained on noisy or reverberated data. However, the noise and reverberation that are added during training may not correspond to the actual noise and reverberation that are observed when the recognizer is used afterwards to recognize unknown speech fragments in real environments. It is not clear whether a system trained on typical noises and reverberation would do better than a recognizer trained on clean speech. Furthermore, most practical speech recognition systems, for which the recognizer used in this paper serves as a reference, are trained on clean data. If they are used in voice-controlled systems operating in noisy and reverberated environments, performance decreases are expected similar to those observed in our experiments.

5. EXPERIMENTAL RESULTS

For the evaluation the three criteria are taken into account that have been presented in Sections 4.1 and 4.2. Most of the experiments have been carried out under stationary acoustics and are based on data that was generated in a simulated acoustic environment using the method described in [26]. In all the experiments omnidirectional microphones were used, placed on a linear array, 5 cm apart. The speaker was in front of the array in the broadside direction (making an angle of 90° with the array). The sampling frequency used throughout the simulations is 16 kHz.

In total 7 dereverberation algorithms have been compared, as summarized in Table 1. The table gives references to the literature and to previous sections in the text, and an indication of the relative complexity of the algorithms. The exact parameter setting can be found at the end of the sections mentioned in the third column of the table. The complexity numbers are based on the execution time of the current implementation of the algorithms. So far, the algorithms have been mainly evaluated and optimized towards dereverberation performance. The implementation schemes still need to be improved.

5.1. Reverberation time

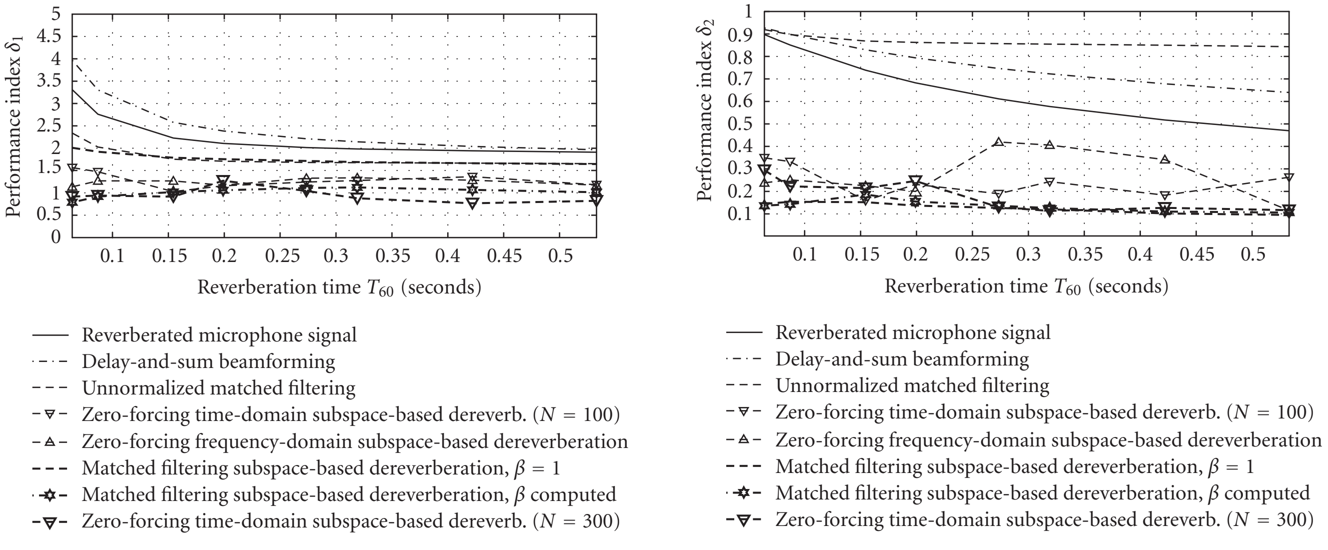
A first parameter that has a strong influence on the performance of the algorithms is the reverberation time. The reverberation time T_{60} is defined as the time needed for the sound energy to fall by 60 dB [16]. Typical reverberation times are of the order of hundreds or even thousands of milliseconds. For a typical office room T_{60} is between 100 and 400 milliseconds, while for a church T_{60} can be several seconds.

The simulated recording room is rectangular (36 m^3) and empty, with all walls having the same energy reflection coefficient ρ. Hence, the reverberation time can be computed using Eyring's formula [16]:

$$T_{60} = \frac{0.163V}{-S\log_e \rho}, \tag{42}$$

where S is the total surface of the room and V is the volume of the room.

The results corresponding to the first experiment are presented in Figures 6 and 7, showing performance indices δ_1 or δ_2 and the word error rate as a function of the reverberation time T_{60}. Recall that higher δ_1 or δ_2 values, or lower word error rates correspond to smaller expected residual reverberation. A number of T_{60}-values were considered that are representative for low-reverberant up to office room environments, ranging from 64, 87, 155, 199, 274, 319, 422 to 533 milliseconds. All room environments have been generated using the method described in [26].



(a)

(b)

FIGURE 6: Performance indices δ_1 and δ_2 as a function of the reverberation time: only the beamforming algorithm and the unnormalized matched filter succeed in enhancing the signal quality. The subspace-based approaches fail to improve the signal quality.

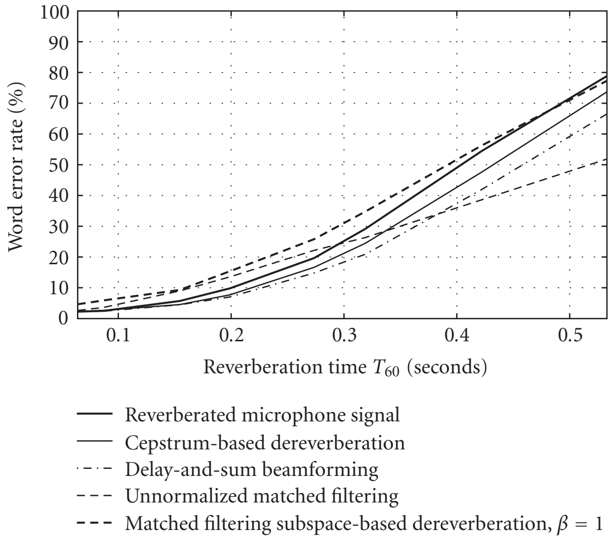


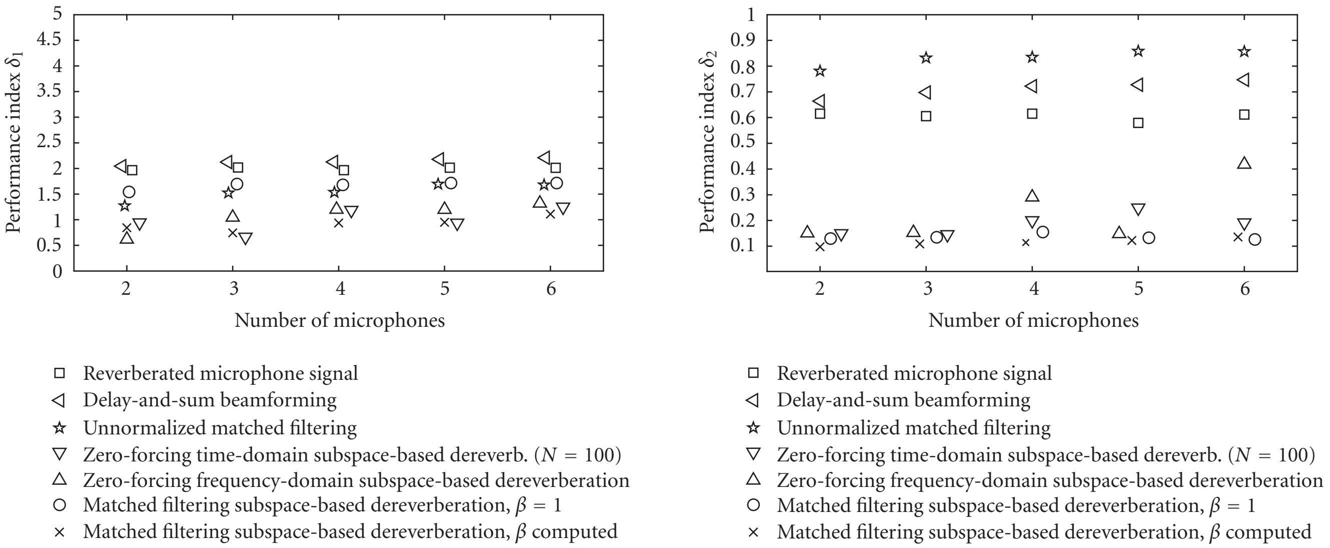
FIGURE 7: Word error rate as a function of the reverberation time: apparently, the speech recognition system shows a significant performance loss in highly reverberant rooms. Only the beamforming algorithm, the cepstrum-based dereverberation and the unnormalized matched filter succeed in enhancing the signal quality. The subspace-based approaches fail to improve the signal quality.

The reverberation times were computed following (42). Also the direct-to-reverberant ratios were calculated corresponding to the considered T_{60} values, resulting in drr $=$ $6.16, 4.19, 0.8, -0.59, -2.24, -3.01, -4.37$, and -5.48 dB, respectively. The direct-to-reverberant ratio was computed following [32] as the energy in the direct path impulse to the third microphone divided by the energy in the rest of the acoustic impulse response to the third microphone. During all simulations the distance between the speaker and the center of the 6-microphone array was 94 cm.

The reference curve (thick full line) corresponds to the nonenhanced, reverberated signals. It is seen from Figure 7 that the speech recognition system shows a significant performance loss in more highly reverberant rooms. This justifies the need for adequate dereverberation of the input signals. It can furthermore be observed that only the beamforming algorithm, the cepstrum-based dereverberation, and the unnormalized matched filter are able to enhance the signals and show better performance than the unprocessed reference, especially for large reverberation times.

Apparently, the subspace-based dereverberation algorithms fail to enhance the signals, unfortunately. Increasing $N_{\max}$ (compare the time-domain subspace approach with $N_{\max} = N = 100$ and $N_{\max} = N = 300$) will typically not improve the signal enhancement, and only put higher demands on the computational and memory capabilities of the computer system. The reason why subspace algorithms fail to enhance the signals is that they are typically "blind" and hence estimate the transmission paths based on the microphone signals only. The first algorithmic step consists in estimating the order of the transmission path filters. Given the fact that speech signals are nonpersistently exciting and that the system order is typically high (even infinite in fact), the order of the transmission path filters is always underestimated. This results in large errors as subspace algorithms are highly sensitive to system-order mismatches (see Figure 2).

The above observations that follow from Figures 6 and 7 have also been confirmed through subjective listening tests: the cepstrum and the beamforming approach clearly reduce



(a)

(b)

Figure 8: Performance indices δ_1 and δ_2 as a function of the number of microphones: the performance of the algorithms only marginally improves if the number of microphones is increased. Beamforming and standard unnormalized matched filtering are able to remove some of the reverberation.

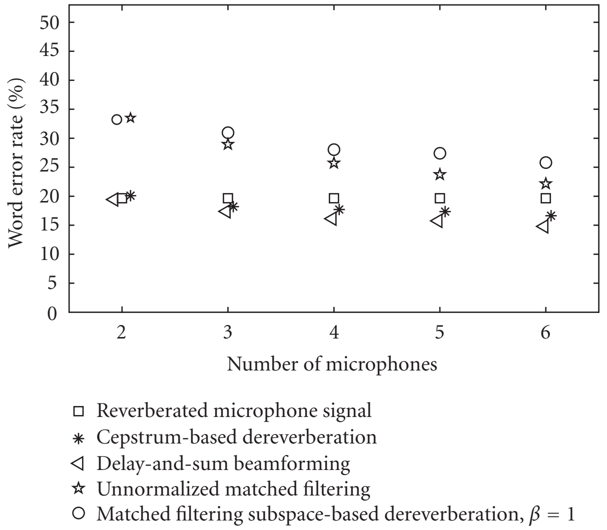


Figure 9: Word error rate as a function of the number of microphones: the performance of the algorithms only marginally improves if the number of microphones is increased. Beamforming, cepstrum-based dereverberation, and standard unnormalized matched filtering succeed in removing some of the reverberation.

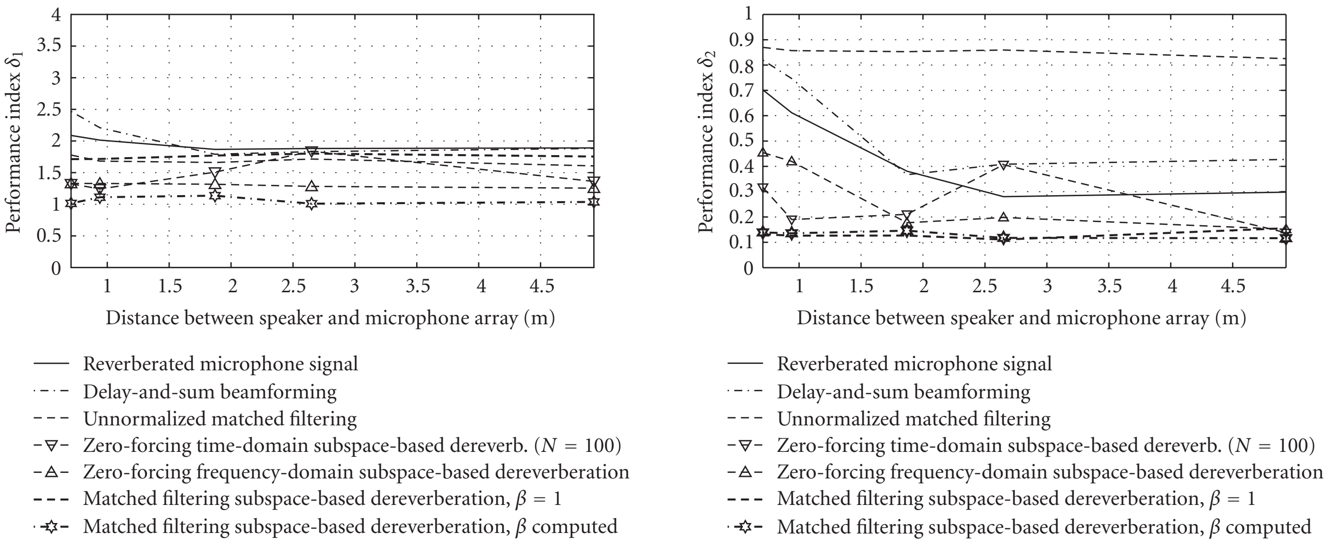
that amount of dereverberation, whereas the subspace algorithms do not improve or sometime worsen the signal quality. The unnormalized matched filtering algorithm leaves a considerable amount of residual reverberation, which is confirmed by the word error rate score and by performance index δ_1, but contradicted by the high score on the δ_2 index.

5.2. Number of microphones

The number of microphones was gradually increased from 2 to 6. The distance between the speaker and the center of the microphone array was again 94 cm. The reflection coefficient was chosen such that the reverberation time (see (42)) corresponded to 274 milliseconds. The other characteristics of the room were left unchanged. The results are shown in Figures 8 and 9. It appears that beamforming, cepstrum-based dereverberation, and standard unnormalized matched filtering are able to remove some of the reverberation. Unfortunately, subspace algorithms do not seem to be able to enhance the reverberated signals. It is furthermore observed that if the number of microphones is increased, the performance of the algorithms only marginally improves. This performance increase is possibly due to the increased number of degrees of freedom and the increased spatial sampling that is obtained when more microphones are involved.

5.3. Distance between speaker and microphone array

The distance d between the speaker and the center of the 6-microphone array was changed between 70.7 cm to 4.93 m. The other characteristics of the room were left unchanged. Hence, the reverberation time T_{60} (see (42)) remained fixed to 274 milliseconds. Based on the results that are presented in Figures 10 and 11 it can be concluded that, again, beamforming, standard unnormalized matched filtering, and cepstrum-based dereverberation outperform the unprocessed reverberated signal. We should keep in mind, however, that the unnormalized matched filtering algorithm uses a priori knowledge. Hence, a performance loss



(a)

(b)

Figure 10: Performance indices δ_1 and δ_2 as a function of the distance between speaker and microphone array: beamforming and standard unnormalized matched filtering outperform the unprocessed reverberated signal and the subspace technique. Performance decreases with increasing speaker-to-array distance.

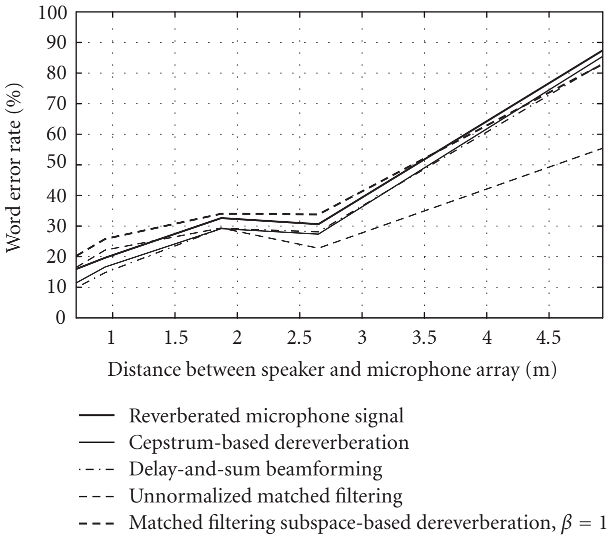


Figure 11: Word error rate as a function of the distance between speaker and microphone array: beamforming, standard unnormalized matched filtering and cepstrum-based dereverberation outperform the unprocessed reverberated signal and the subspace technique. Performance decreases with increasing speaker-to-array distance.

is expected if the transmission paths h_m (see Figure 1) are unknown and have to be estimated.

Note that the dereverberation performance decreases with increasing speaker-to-array distance. A reason for this could be that less direct sound is captured when the speaker is far from the array. The relative amount of late reflections, also called reverberation, then increases and a more complex reverberation scenario is obtained. To quantify this, the direct-to-reverberant ratio was calculated following [32] for each of the scenarios we considered, that is, for $d = 0.707, 0.94, 1.87, 2.65, 4.93$ meter, resulting in drr $= -0.12, -2.24, -7.70, -10.69, -10.11$ dB, respectively.

5.4. Additive noise

Finally, noise has been added to the multichannel speech recordings at different signal-to-noise ratios (SNR). For the computation of the SNR, first speech and noise periods are determined. Then the unbiased SNR can be computed as the mean variance of the speech signal during speech periods divided by the mean variance of the noise. White noise as well as speech-weighted noise have been added to the reverberated speech signals. In order to obtain a desired SNR level, the noise amplitude was adjusted on the first data base signal. For simplicity, equal-noise amplitudes were then applied to the other data base signals. In order to validate the spatial selectivity properties of the dereverberation algorithms both uncorrelated noise and spatially correlated noise have been considered. In all cases, the number of microphones was fixed to 6. The speaker was right in front of the array at a distance of 94 cm from the center of the microphone array. The reflection coefficients of the room were chosen such that the reverberation time (see (42)) corresponded to 274 milliseconds. The other characteristics of the room were left unchanged with respect to the previous experiments. To generate spatially correlated noise a noise source was positioned in the recording room at 146 cm from the center of

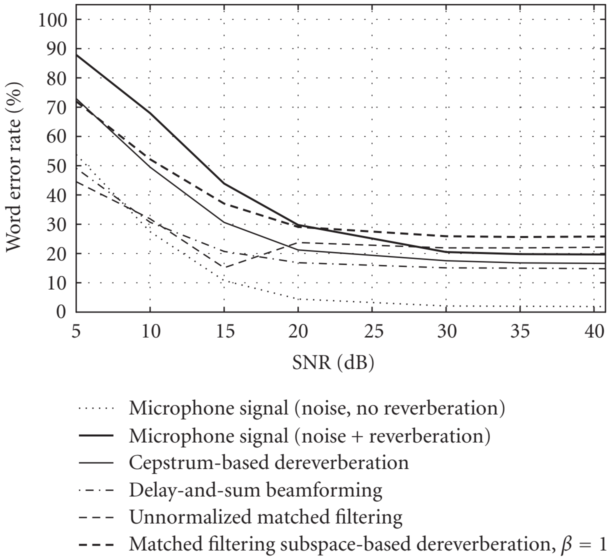


Figure 12: Word error rate as a function of the SNR for uncorrelated additive white noise: the speech recognizer shows a significant performance decrease in noisy, reverberated environments. Observe that the effect of the noise on the overall performance of the algorithms is significant for SNR levels lower than 20 dB. Best performance is obtained with beamforming, standard unnormalized matched filtering, and cepstrum-based dereverberation. Finally, a performance increase is observed for the subspace techniques at low SNR levels.

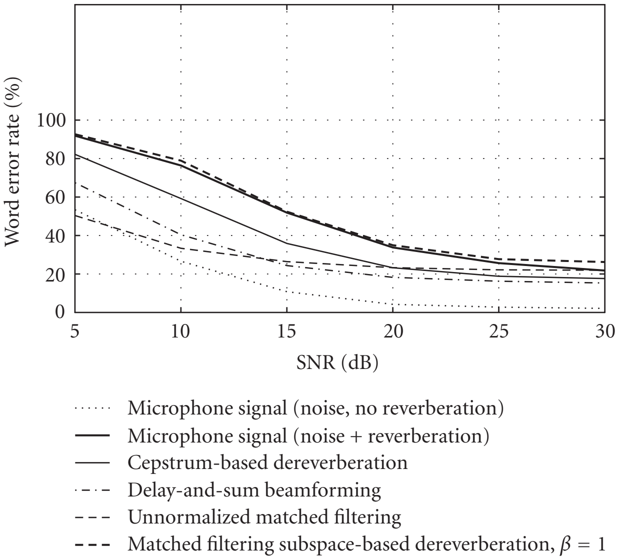


Figure 13: Word error rate as a function of the SNR for spatially correlated additive white noise: the speech recognizer shows a significant performance decrease in noisy, reverberated environments. Observe that the effect of the noise on the overall performance of the algorithms is significant for SNR levels lower than 20 dB. Best performance is obtained with beamforming, standard unnormalized matched filtering, and cepstrum-based dereverberation.

the microphone, at an angle of 38 degrees with respect to the main axis of the array. In this way, the distance between the noise source and the speaker was 175 cm. The impulse responses were computed to each of the microphones using the method described in [26]. The noise signal was then convolved with these impulse responses and added to the speech components at each microphone.

The word error rate is shown as a function of the SNR in Figures 12, 13, 14 for each noise configuration. It can be concluded that the effect of the noise on the overall performance of the algorithms is significant for SNR levels lower than 20 dB. Apparently, the type of noise (white, speech-weighted) as such does not play an important role. Best performance is obtained with beamforming, standard unnormalized matched filtering, and cepstrum-based dereverberation. Finally, it is observed that subspace techniques show a performance increase at low SNR levels. Maybe this is due to the fact that at low SNR levels we see the effect of the noise reduction rather than that of the dereverberation capabilities of the subspace algorithms.

Remark that we also added noise to the clean data base signals (i.e., signals without reverberation) under the same signal-to-noise ratio as in the case of the reverberated multichannel data. The corresponding results are labeled “noise, no reverberation.” The main objective was to determine the performance of the speech recognizer on the additive noise-only data and to compare it with data to which additive noise as well as reverberation were added. It can be concluded that adding only additive noise only mildly increases the word error rate. Hence, we conclude that reverberation has a more negative impact on the overall recognition rate than additive noise.

5.5. *Real-life recordings*

Next, also some real-life experiments were performed. The corresponding results can be found in Table 2. The real-life data was recorded in the speech laboratory (68.5 m^3) of the Electrotechnical Department of the Katholieke Universiteit Leuven, Belgium. The speaker was imitated by a loudspeaker in order to preserve stationary acoustics. The acoustic impulse responses from the speaker to each of the 6 microphones was determined (for experiment 2 and 4) with an NLMS adaptive filter based on white noise data. The reflectivity of the room was changed over the four experiments and the reverberation time T_{60} was computed using Schroeder’s method, a reference to which can be found in [26]. Observe that in the fourth experiment spatially correlated speech-weighted noise was added at about 8 dB SNR. The position of the noise source was similar to that of the simulated environment of Section 5.4 and Figure 14.

Proper dereverberation is only obtained with beamforming and cepstrum-based dereverberation. Unnormalized matched filtering appears to be effective only if both reverberation and noise are added. The effectiveness of the dereverberation algorithms also appears to be smaller than

TABLE 2: Dereverberation performance on real-life experiments. For each of the algorithms the word error rate (WER) is shown. Proper dereverberation is only obtained with beamforming and cepstrum-based dereverberation. Unnormalized matched filtering appears to be effective only if both reverberation and noise are added. The effectiveness of the dereverberation algorithms also appears to be smaller than with simulated data.

Experiment	1	2	3	4
Speaker-to-array distance (m)	1.87	1.85	1.87	1.32
Reverberation time T_{60} (ms)	121	244	275	293
Signal-to-noise ratio (dB)	—	—	—	7.87
	WER	WER	WER	WER
Unprocessed microphone signal	6.35%	14.09%	16.79%	49.97%
Cepstrum-based dereverberation	5.98%	13.64%	13.95%	42.39%
Delay-and-sum beamforming	6.20%	14.63%	15.34%	36.99%
Unnormalized matched filtering	—	24.92%	—	44.70%
Zero-forcing frequency-domain subspace	—	20.53%	—	75.70%
Matched filtering subspace, $\boldsymbol{\beta} = \mathbf{1}$	10.01%	21.37%	25.42%	56.55%

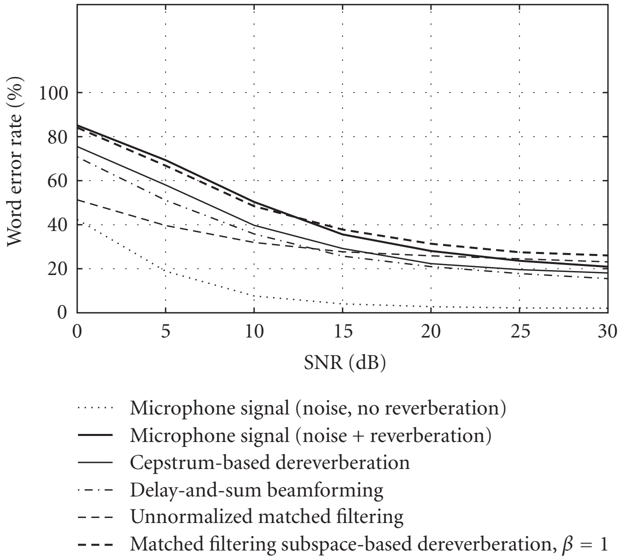


FIGURE 14: Word error rate as a function of the SNR for spatially correlated additive speech-weighted noise: the speech recognizer shows a significant performance decrease in noisy, reverberated environments. Observe that the effect of the noise on the overall performance of the algorithms is significant for SNR levels lower than 20 dB. Best performance is obtained with beamforming, standard unnormalized matched filtering, and cepstrum-based dereverberation.

with simulated data. It is observed from Table 2 that 26% relative improvement can be obtained (see experiment 4) if both reverberation and noise are added. In the cases where no noise was present, the relative improvement is limited to 17%.

Of all algorithms that have been evaluated through the experiments presented in Section 5, standard techniques such as beamforming and the cepstrum-based approach seem most effective in combating reverberation distortion. More recently proposed techniques such as the subspace methods discussed in Sections 3.4, 3.5, and 3.6 fail to enhance the signal quality, despite the larger computational complexity and memory requirements. Apart from limitations due to the undermodeling discussed in Section 3.4.5, the algorithms also suffer from the time variations of the acoustics, which are common in practice. This furthermore complicates the identification task. More simple, classical solutions such as beamforming and cepstrum-based techniques do not require a modeling and tracking of the acoustics, and are hence much more robust against these time variations.

Hence, most signal enhancement systems that are currently used in speech communication applications such as teleconferencing, handsfree telephony, and voice-controlled systems mainly concentrate on the suppression of background noise and acoustic echoes, and simply deal with reverberation by assuming relatively small speaker-to-microphone distances or by applying simple, classical dereverberation algorithms such as beamforming and cepstrum-based techniques. As long as these systems are used in lowly to moderately reverberating environments with the speaker at a close distance to the microphones, good performance is expected. Whenever the environment is highly reverberating and the speaker-to-microphone distance is large, severely reverberated speech is expected, which can considerably compromise speech intelligibility.

6. CONCLUSIONS

Dereverberation algorithms are employed to preserve the signal quality in speech communication systems. Several classical and more recently developed multichannel dereverberation algorithms have been compared. It was shown that classical solutions, such as beamforming, cepstral dereverberation, and unnormalized matched filtering show moderate performance increases with respect to the processing of nonenhanced, reverberated speech signals. More advanced subspace-based dereverberation techniques, on the other hand, did not provide any signal enhancement despite their high-computational load. There are mainly three impediments that explain this poor performance. First of all, blind subspace methods are highly sensitive to model order mismatches. The effect of this is most prominent for higher-order systems, which are commonly encountered in speech enhancement applications. Secondly, the quality of the model is compromised by the additive noise that is superimposed on the signals. Subspace techniques are known to be quite sensitive to small amounts of additive noise. Thirdly, in a real-life situations, the acoustics are time varying, and need to be tracked, which makes it even more difficult to obtain a reliable system model. Subspace-based identification procedures furthermore tend to give rise to a high algorithmic cost and large memory consumption for parameter settings that are typically used in speech applications. In practice, very often the model order needs to be limited for computational reasons. Better performance might be reached if higher, and hence more realistic model orders can be applied. This would be feasible if cheaper implementation schemes were available.

List of symbols

Lower case bold-faced letters are used to denote vectors and upper case bold-faced letters to denote matrices. In addition the following notation is used throughout the paper:

$\star$	Convolution operation
$\mathcal{F}$	Fourier transform
$x[k]$	Discrete-time-domain signal x
$\underline{x}(f)$	Frequency-domain representation of $x[k]$
$x_{\text{rc}}[m]$	Real cepstrum of $x[k]$
$\mathbf{A}^T$	Transpose of $\mathbf{A}$
$\mathbf{A}^*$	Complex conjugate of $\mathbf{A}$
$\mathbf{A}^\dagger$	Pseudo-inverse of $\mathbf{A}$
$\mathbf{A}^H$	Hermitian transpose of $\mathbf{A}$
$\mathbf{1}$	Vector with all elements equal to 1
$\mathbf{0}$	Zero matrix
$\mathbf{0}_N$	$N \times N$ zero matrix
$\mathbf{0}_{M\times N}$	$M \times N$ zero matrix
$\mathbf{I}_N$	$N \times N$ identity matrix
$\|\cdot\|$	2-norm of a vector or a matrix

Other notation is explained in the text.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Jacques Duchateau for making available the automatic speech recognition system of the ESAT-PSI research group. This research work was carried out at the ESAT Laboratory of the Katholieke Universiteit Leuven, in the frame of the Belgian State Interuniversity Poles of Attraction Programmes P5/22 and P5/11. The scientific responsibility is assumed by its authors.

REFERENCES

[1] D. Bees, M. Blostein, and P. Kabal, “Reverberant speech enhancement using cepstral processing,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’91)*, vol. 2, pp. 977–980, Toronto, Ontario, Canada, May 1991.

[2] Q.-G. Liu, B. Champagne, and P. Kabal, “A microphone array processing technique for speech enhancement in a reverberant space,” *Speech Communication*, vol. 18, no. 4, pp. 317–334, 1996.

[3] A. Oppenheim and R. Schafer, *Digital Signal Processing*, chapter 10, Prentice-Hall, Englewood Cliffs, NJ, USA, 1975.

[4] A. P. Petropulu and S. Subramaniam, “Cepstrum based deconvolution for speech dereverberation,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’94)*, vol. 1, pp. 9–12, Adelaide, Australia, April 1994.

[5] J. B. Allen, D. A. Berkley, and J. Blauert, “Multimicrophone signal-processing technique to remove room reverberation from speech signals,” *The Journal of the Acoustical Society of America*, vol. 62, no. 4, pp. 912–915, 1977.

[6] T. Wittkop and V. Hohmann, “Strategy-selective noise reduction for binaural digital hearing aids,” *Speech Communication*, vol. 39, no. 1-2, pp. 111–138, 2003.

[7] 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.

[8] P. A. Nelson, F. Orduna-Bustamante, and H. Hamada, “Inverse filter design and equalization zones in multichannel sound reproduction,” *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 3, pp. 185–192, 1995.

[9] Y. Grenier, “A microphone array for car environments,” *Speech Communication*, vol. 12, no. 1, pp. 25–39, 1993.

[10] C. Sydow, “Broadband beamforming for a microphone array,” *The Journal of the Acoustical Society of America*, vol. 96, no. 2, pp. 845–849, 1994.

[11] B. D. van Veen and K. M. Buckley, “Beamforming: a versatile approach to spatial filtering,” *IEEE ASSP Magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[12] S. Affes and Y. Grenier, “A signal subspace tracking algorithm for microphone array processing of speech,” *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 5, pp. 425–437, 1997.

[13] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, “Subspace methods for the blind identification of multichannel FIR filters,” *IEEE Transactions on Signal Processing*, vol. 43, no. 2, pp. 516–525, 1995.

[14] A.-J. van der Veen, S. Talwar, and A. Paulraj, “Blind identification of FIR channels carrying multiple finite alphabet signals,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’95)*, vol. 2, pp. 1213–1216, Detroit, Mich, USA, May 1995.

[15] S. Gannot and M. Moonen, “Subspace methods for multimicrophone speech dereverberation,” in *Proceedings of the 7th IEEE/EURASIP International Workshop on Acoustic Echo and Noise Control (IWAENC ’01)*, pp. 47–50, Darmstadt, Germany, September 2001.

[16] H. Kuttruff, *Room Acoustics*, Applied Science Publishers, Essex, England, 2nd edition, 1979.

[17] D. Johnson and D. Dudgeon, *Array Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[18] D. van Gerven, S. van Compernolle, P. Wauters, W. Verstraeten, K. Eneman, and K. Delaet, “Multiple beam broadband beamforming: filter design and real-time implementation,” in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA ’95)*, pp. 173–176, New Paltz, NY, USA, October 1995.

[19] G. B. Giannakis and S. D. Halford, “Blind fractionally spaced equalization of noisy FIR channels: direct and adaptive solutions,” *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2277–2292, 1997.

[20] K. Eneman and M. Moonen, “Ambiguity elimination in frequency-domain subspace identification,” Internal Report ESAT-SCD 06.151, p. 12, Katholieke Universiteit Leuven, Leuven, Belgium, 2007, https://gilbert.med.kuleuven.be/~koen/reports/06-151.pdf.

[21] K. Eneman and M. Moonen, “DFT modulated filter bank design for oversampled subband systems,” *Signal Processing*, vol. 81, no. 9, pp. 1947–1973, 2001.

[22] P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[23] K. Rahbar, J. P. Reilly, and J. H. Manton, “A frequency domain approach to blind identification of MIMO FIR systems driven by quasi-stationary signals,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’02)*, vol. 2, pp. 1717–1720, Orlando, Fla, USA, May 2002.

[24] K. Eneman and M. Moonen, “Hybrid subband/frequency-domain adaptive systems,” *Signal Processing*, vol. 81, no. 1, pp. 117–136, 2001.

[25] S. Doclo and M. Moonen, “Combined frequency-domain dereverberation and noise reduction technique for multi-microphone speech enhancement,” in *Proceedings of the 7th IEEE/EURASIP International Workshop on Acoustic Echo and Noise Control (IWAENC ’01)*, pp. 31–34, Darmstadt, Germany, September 2001.

[26] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small-room acoustics,” *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[27] K. Eneman, J. Duchateau, M. Moonen, D. van Compernolle, and H. van Hamme, “Assessment of dereverberation algorithms for large vocabulary speech recognition systems,” in *Proceedings of the 8th European Conference on Speech Communication and Technology (Eurospeech ’03)*, pp. 2689–2692, Geneva, Switzerland, September 2003.

[28] J. Duchateau, K. Demuynck, D. van Compernolle, and P. Wambacq, “Class definition in discriminant feature analysis,” in *Proceedings of the 7th European Conference on Speech Communication and Technology (Eurospeech ’01)*, vol. 3, pp. 1621–1624, Aalborg, Denmark, September 2001.

[29] K. Demuynck, J. Duchateau, D. van Compernolle, and P. Wambacq, “Improved feature decorrelation for HMM-based speech recognition,” in *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP ’98)*, vol. 7, pp. 2907–2910, Sydney, Australia, November-December 1998.

[30] J. Duchateau, K. Demuynck, and D. van Compernolle, “Fast and accurate acoustic modelling with semi-continuous HMMs,” *Speech Communication*, vol. 24, no. 1, pp. 5–17, 1998.

[31] K. Demuynck, J. Duchateau, D. van Compernolle, and P. Wambacq, “An efficient search space representation for large vocabulary continuous speech recognition,” *Speech Communication*, vol. 30, no. 1, pp. 37–53, 2000.

[32] D. Bees, *Enhancement of acoustically reverberant speech using cepstral methods*, Ph.D. thesis, McGill University, Montreal, Canada, July 1990.