ACOUSTIC SYSTEM EQUALIZATION USING CHANNEL SHORTENING TECHNIQUES FOR SPEECH DEREVERBERATION

†*Wancheng Zhang*, ‡*Andy W. H. Khong, and* †*Patrick A. Naylor*

†Department of EEE, Imperial College London, London, SW7 2AZ
{wancheng.zhang07, p.naylor}@imperial.ac.uk
‡School of EEE, Nanyang Technological University, Singapore
andykhong@ntu.edu.sg

ABSTRACT

The use of channel shortening techniques for speech dereverberation is discussed in this paper. This approach is motivated by the observation that early reverberation caused by the early reflections in room acoustics is not perceived as a separate sound to the direct sound but is perceived to reinforce the direct sound and is therefore considered useful with regards to speech intelligibility. Compared with inverse filtering, the convergence rate of iterative channel shortening is much higher, which is significant in real-time speech dereverberation. Two iterative channel shortening techniques are presented in this paper and they are shown to outperform standard inverse filtering approaches in the comparative tests described.

1. INTRODUCTION

In hands-free communications, the speech signal can be distorted by room reverberation, resulting in reduced intelligibility to listeners. One method to achieve dereverberation is to perform identification and inverse filtering of the room impulse responses (RIRs). The methodology is illustrated in Fig. 1. Consider a clean speech

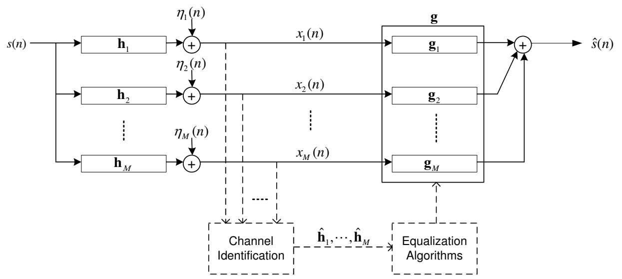

Figure 1: Illustration of identification and inverse filtering of acoustic systems.

signal $s(n)$ propagating through an M-channel acoustic system, the channels of which are characterized by their impulse responses $\mathbf{h}_m = [h_m(0)\ h_m(1)\ \cdots\ h_m(L-1)]^T$, $m = 1, \cdots, M$, where $\{\cdot\}^T$ denotes the transpose operation. Using the noisy reverberant speech signals

$$x_m(n) = s(n) * h_m(n) + \eta_m(n), \tag{1}$$

estimates of the RIRs $\mathbf{h}_m$ can be obtained with blind system identification techniques such as in [1], where $*$ denotes linear convolution, and $\eta_m(n)$ is the channel noise of the mth channel. Then, with the estimates $\hat{\mathbf{h}}_m = [\hat{h}_m(0)\ \hat{h}_m(1)\ \cdots\ \hat{h}_m(L-1)]^T$, an inverse filtering system $\mathbf{g} = [\mathbf{g}_1^T\ \mathbf{g}_2^T\ \ldots\ \mathbf{g}_M^T]^T$, which is formed by stacking column vectors of the components $\mathbf{g}_m = [g_m(0)\ g_m(1)\ \ldots\ g_m(L_i-1)]^T$, can be designed with some equalization algorithm. Equalization algorithms are generally designed so that

$$\sum_{m=1}^{M} \hat{h}_m(n) * g_m(n) = d(n), \tag{2}$$

where $d(n)$ is the delta function. By summing up the output of $\mathbf{g}_m$ with input $x_m(n)$, we expect a good estimate, $\hat{s}(n)$, of $s(n)$ can be obtained. In this paper, we do not consider the channel noise or the errors that may possibly be introduced into $\hat{\mathbf{h}}_m$ by the system identification. In this case, $\eta_m(n) = 0$ and $\hat{\mathbf{h}}_m = \mathbf{h}_m$.

Traditionally, inverse filtering systems can be obtained, for single channel cases, by using the method of least squares (LS), or employing multiple-input/output inverse theorem (MINT) when multiple microphones are deployed [2]. Generally, the LS method only gives an approximate inverse system, which is usually of limited effectiveness in the context of speech dereverberation [3]. On the other hand, RIRs can, in theory, be exactly inverse filtered for the multichannel case using MINT providing that the multichannel RIRs do not share any common zeros [2]. MINT has been generalized to a multichannel least squares (MCLS) method [4]. The MCLS can be shown to invert those parts of the channels with factors which are not common in the multichannel RIRs and to perform the LS inverse of the parts with common zeros [5]. Using MINT or MCLS, an inverse filtering system $\mathbf{g}$ can be obtained by

$$\mathbf{g} = \mathbf{H}^{+}\mathbf{d}, \tag{3}$$

where $\mathbf{H} = [\mathbf{H}_1\ \cdots\ \mathbf{H}_M]$ is defined as the system matrix formed by the convolution matrices $\mathbf{H}_m$, $\{\cdot\}^+$ denotes pseudo inverse, and

$$\mathbf{d} = [1\ 0\ \ldots\ 0]^T \tag{4}$$

is an $(L+L_i-1)\times 1$ vector. $\mathbf{H}_m$ is an $(L+L_i-1)\times L_i$

convolution matrix of $\mathbf{h}_m$

$$\mathbf{H}_m = \begin{bmatrix} h_m(0) & 0 & \cdots & 0 \\ h_m(1) & h_m(0) & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ h_m(L-1) & \cdots & \vdots & \vdots \\ 0 & h_m(L-1) & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & h_m(L-1) \end{bmatrix}.$$

Although, compared with single channel LS, better performance can be achieved by using MINT or MCLS, MINT and MCLS are still computationally expensive [4]. This motivates the use of subband and iterative algorithms [4][6] to reduce the computational complexity. However, the system matrix $\mathbf{H}$ is usually ill-conditioned, which limits the convergence rate of the iterative algorithms.

In this paper, we address the performance degradation in iterative inverse filtering algorithms due to the ill-conditioning of the system matrix $\mathbf{H}$. An additional advantage of the proposed approach is that its computational complexity is much lower than MINT. We achieve this by a process known as channel shortening which has been extensively developed in the context of digital communications to mitigate the inter-symbol and inter-carrier interference. These techniques were firstly developed for the single-input/single-output (SISO) cases and later extended for the multiple-input/multiple-output (MIMO) cases [7][8]. In addition, both closed form [9] and adaptive [10][11] methods have been studied. A common frame work and an overview of the design techniques for channel shortening can be found in [12]. For the shortening of acoustic systems, the closed form channel shortening techniques have been studied in [13]. Since the closed form channel shortening techniques need to compute the inverse of large scale matrix, they are also computationally complex. The motivation behind employing channel shortening techniques for our acoustic system equalization application is based on the fact that early reverberation caused by the early reflections in room acoustics is not perceived as a separate sound to the direct sound but is perceived to reinforce the direct sound and is therefore considered useful with regards to speech intelligibility [14]. Therefore, it can be argued that it is not necessary to use the delta function as the target impulse response (TIR) in RIRs equalization for the purpose of dereverberation. Shortening the RIRs is indeed satisfactory for enhancing the intelligibility of reverberant speech. By relaxing the TIR to be less constrained than the delta function, we expect that fast and high suppression of the tail of RIRs is correspondingly achieved.

This paper is organized as follows: firstly, two iterative algorithms for RIRs shortening will be presented in Section 2. Then, the efficiency of inverse filtering and shortening will be compared by simulations in Section 3, and the two proposed channel shortening algorithms will also be compared and computational complexity will be analyzed in this section. Finally, we will draw some conclusions in Section 4.

2. ITERATIVE APPROACHES TO RIRs SHORTENING

The closed form MINT, or the iterative algorithm [6] based on it, usually aims to force the equalized impulse response

$$\begin{aligned} \mathbf{y} &= [y(0)\ y(1)\ \cdots\ y(L+L_i-2)]^T \\ &= \sum_{m=1}^{M} h_m(n) * g_m(n) \end{aligned} \tag{5}$$

to be a TIR of the delta function (4). Their aim is to minimize the cost function

$$J = \|\mathbf{d} - \mathbf{y}\|^2, \tag{6}$$

where $\|\cdot\|$ denotes the Euclidean norm.

As stated above, forcing $\mathbf{y}$ to be $\mathbf{d}$ is not always necessary for dereverberation. In many cases, the characteristics of the early part of the TIR are not important with regard to the intelligibility of the speech. Therefore, in this work, the aim is to minimize the energy of the late part of the equalized impulse response, sometimes referred to as the equalization tail; at the same time, the early part of the TIR is left unconstrained. We propose to achieve this by using a weighting function in the cost function

$$J = \|\mathbf{w} \circ (\mathbf{d} - \mathbf{y})\|^2, \tag{7}$$

where

$$\begin{aligned} \mathbf{w} &= [w(0)\ w(1)\ \cdots\ w(L+L_i-2)]^T \\ &= [\underbrace{1\ 0\ \cdots\ 0}_{L_r}\ 1\ \cdots\ 1]^T \end{aligned} \tag{8}$$

is the weighting function and $\circ$ denotes the Hadamard product. Here L_r is the length of the 'relaxing' window. We use $w(0) = 1$ to avoid the trivial solution.

The steepest descent (SD) method [15] has been used in [5] for shortening the RIRs. Here we will firstly review it and then apply the conjugate gradient (CG) method [16] to compute the shortening systems of the RIRs. We will compare the performance of these two algorithms in Section 3.2.

2.1 Steepest descent method for RIRs shortening

In matrix form, (7) can be written as

$$J = \|\mathbf{W}(\mathbf{d} - \mathbf{H}\mathbf{g})\|^2, \tag{9}$$

where $\mathbf{W} = \mathrm{diag}\{\mathbf{w}\}$ and $\mathbf{g} = [\mathbf{g}_1^T\ \mathbf{g}_2^T\ \ldots\ \mathbf{g}_M^T]^T$ is the shortening system. The gradient of J can be written as

$$\nabla J = -2(\mathbf{W}\mathbf{H})^T\mathbf{W}\mathbf{d} + 2(\mathbf{W}\mathbf{H})^T(\mathbf{W}\mathbf{H})\mathbf{g}. \tag{10}$$

The shortening system $\mathbf{g}$ can then be iteratively obtained by

$$\mathbf{g}(k+1) = \mathbf{g}(k) - \mu\nabla J, \tag{11}$$

where k denotes the index of iteration, and μ is the step-size. The proposed steepest descent channel shortening ($\mathrm{SD_{CS}}$) algorithm is given in Algorithm 1.

Algorithm 1 Proposed $\mathrm{SD_{CS}}$ for computing $\mathbf{g}$.

$\mathbf{g}(0) = \mathbf{0}_{ML_i}$
$\mathbf{b} = (\mathbf{WH})^T\mathbf{Wd},\ \mathbf{A} = (\mathbf{WH})^T(\mathbf{WH})$
for $k = 0, 1, 2, \ldots$ **do**
  $\nabla J = -2\mathbf{b} + 2\mathbf{A}\mathbf{g}(k)$
  $\mathbf{g}(k+1) = \mathbf{g}(k) - \mu\nabla J$
end for

2.2 Conjugate gradient method for RIRs shortening

The CG method chooses A-conjugate search directions in searching the optimal solution in order to avoid the gradient directions that are possibly not different enough during the iteration in SD [16]. The proposed conjugate gradient channel shortening ($\mathrm{CG_{CS}}$) algorithm using CG method is given in Algorithm 2.

Algorithm 2 Proposed $\mathrm{CG_{CS}}$ for computing $\mathbf{g}$.

$\mathbf{g}(0) = \mathbf{0}_{ML_i}$
$\mathbf{b} = (\mathbf{WH})^T\mathbf{Wd},\ \mathbf{A} = (\mathbf{WH})^T(\mathbf{WH})$
$\mathbf{r}_a = \mathbf{b} - \mathbf{A}\mathbf{g}(0),\ \mathbf{p}_a = \mathbf{r}_a,\ \mu = (\mathbf{r}_a^T\mathbf{r}_a)/(\mathbf{p}_a^T\mathbf{A}\mathbf{p}_a)$
$\mathbf{g}(1) = \mathbf{g}(0) + \mu\mathbf{p}_a,\ \mathbf{r}_b = \mathbf{r}_a - \mu\mathbf{A}\mathbf{p}_a$
for $k = 1, 2, \ldots$ **do**
  $\beta = (\mathbf{r}_b^T\mathbf{r}_b)/(\mathbf{r}_a^T\mathbf{r}_a)$
  $\mathbf{p}_b = \mathbf{r}_b + \beta\mathbf{p}_a$
  $\mathbf{q} = \mathbf{A}\mathbf{p}_b$
  $\mu = (\mathbf{r}_b^T\mathbf{r}_b)/(\mathbf{p}_b^T\mathbf{q})$
  $\mathbf{g}(k+1) = \mathbf{g}(k) + \mu\mathbf{p}_b$
  $\mathbf{r}_a = \mathbf{r}_b$
  $\mathbf{r}_b = \mathbf{r}_b - \mu\mathbf{q}$
  $\mathbf{p}_a = \mathbf{p}_b$
end for

3. SIMULATION RESULTS

3.1 Comparison of inverse filtering and shortening

A comparison of the outputs obtained by inverse filtering and channel shortening will now be given on the basis of the results obtained with the $\mathrm{CG_{CS}}$ algorithm.

In simulations, an $M = 2$ channel acoustic system was used and the channel RIRs were taken from the MARDY database [17]. The length of the RIRs is $L = 2000$, with a sampling frequency of 8 kHz. Both the length of L_i used for inverse filtering and L_s for shortening are $L_i = L_s = L_c$, where $L_c = \lceil \frac{L-1}{M-1} \rceil = 1999$ is the critical length. This length is the minimum length to obtain an inverse filtering system [2] when channels do not share any common zeros.

The inverse filtering approach can be seen equivalent to using a weighting function of $\mathbf{w} = [1\ 1\ \ldots\ 1]^T$ in (7). For shortening, since reflections arriving within 20 ms of the direct sound cause little or no disturbance in hearing even when the amplitude of the reflections is greater than the direct sound [14], we aim to shorten the channel to less than 20 ms (160 taps). Accordingly, the window length L_r in (8) is set as $L_r = 160$.

The squared coefficients of $\mathbf{y}$ after 1000 iterations are shown in Fig. 2 for the cases of inverse filtering and shortening. The energy of the equalization tail

$$E_t = \sum_{n=160}^{L+L_s-2} y(n) \tag{12}$$

against iterations is shown in Fig. 3. We can see from Fig. 2 that for shortening, the tail energy was highly suppressed after $L_r = 160$, whereas the inverse filtering result shows a tail remaining relatively strong.

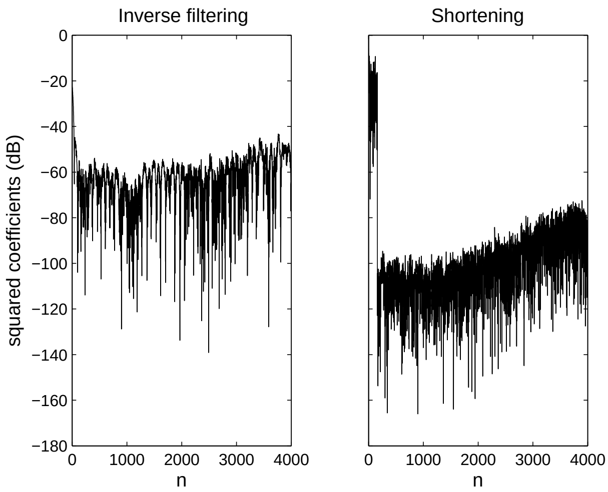

Figure 2: Equalization results of inverse filtering and shortening.

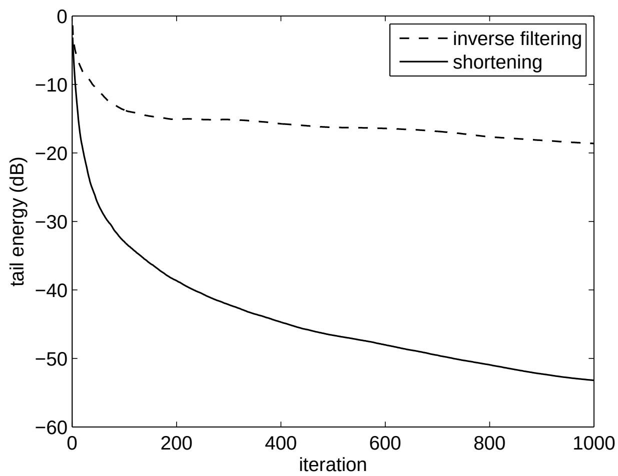

Figure 3: Comparison of the convergence of the tail energy.

3.2 Comparison of $\mathrm{SD_{CS}}$ and $\mathrm{CG_{CS}}$

In this experiment, the efficiency of $\mathrm{SD_{CS}}$ and $\mathrm{CG_{CS}}$ will be compared. In the simulation, $L_r = 160$, $L_s = L_c$, are used. For comparison, a step-size μ equal to the largest eigenvalue of the matrix A in Algorithm 1 is used. This corresponds a step-size capable of achieving the highest rate of convergence. Comparison of the convergence of E_t using the $\mathrm{SD_{CS}}$ and $\mathrm{CG_{CS}}$ is given in Fig. 4.

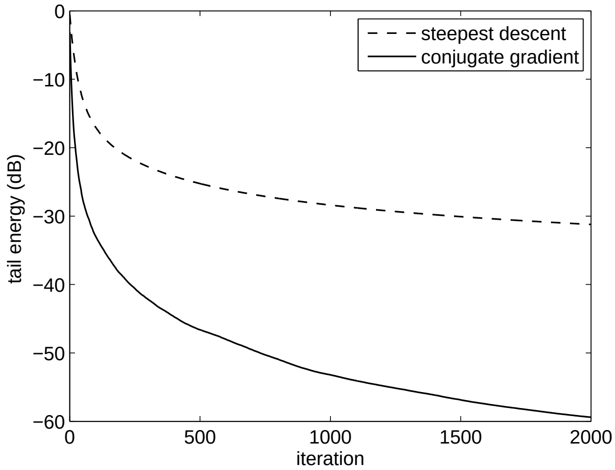

Figure 4: Comparison of E_t between SD_{CS} and CG_{CS}.

It shows that E_t descends much faster using the CG_{CS} than SD_{CS}. Listening tests have indicated that the tail is negligible when its energy E_t is less than -30 dB. In all the experiments for the remainder of this paper, we terminate iteration when $E_t < -30$ dB. For E_t to descend to -30 dB, SD_{CS} requires 1469 iterations, whereas CG_{CS} only needs 69. Since for each iteration, both SD_{CS} and CG_{CS} execute about $2(ML_s)^2$ floating point operations (flops) [16], the CG_{CS} shows significant computational complexity saving.

3.3 Effect of L_r and L_s

In this section, the effect of L_r and L_s on the performance of the proposed CG_{CS} algorithm will be investigated. Firstly, a summary of the effect of L_r and L_s will be given. Then, the simulation results will be shown and analyzed.

a. window length L_r. The window length L_r in the TIR can be chosen as required for the target application. For inverse filtering, it corresponds to $L_r = 1$. In the above experiments, we chose it to be $L_r = 160$. Smaller L_r may sometimes be preferred by applications such as in speech recognition and speaker verification. Using smaller value of L_r however can reduce the convergence rate of the algorithm.

b. L_s, length of the components of shortening systems. Since we only try to shorten, rather than inverse filter the RIRs, a length of $L_s < L_c$ may be sufficient for the tail energy to reduce below some preferred level, for instance, -30 dB. The length L_s can be much smaller than L_c for large L_r. However, a small value of L_s can cause the algorithm to converge slowly beyond which it will limit the lower bound of the tail energy.

The iterations needed for the CG_{CS} to make the tail energy to descend to -30 dB for different combinations of L_r and L_s are given in Table 1. It can be seen that for the same L_r, more iterations will be needed to achieve -30 dB when using smaller L_s. However, as stated above, the flops needed for each iteration is about $2(ML_s)^2$. When L_s is reduced, the flops needed for each iteration will be reduced. Therefore, though more iterations are needed for smaller L_s, the overall computational complexity

L_s	L_r					
	8	32	64	96	128	160
L_c	198	123	99	95	85	69
L_c-100	207	133	106	105	90	75
L_c-200	246	151	118	118	98	83
L_c-300	423	198	146	137	110	92
L_c-400	×	321	185	165	128	106
L_c-500	×	×	419	267	172	139
L_c-600	×	×	×	×	441	240
where × means that -30 dB is not achievable.						

Table 1: Iterations used for the tail energy to descend to -30 dB for different combinations of L_r and L_s.

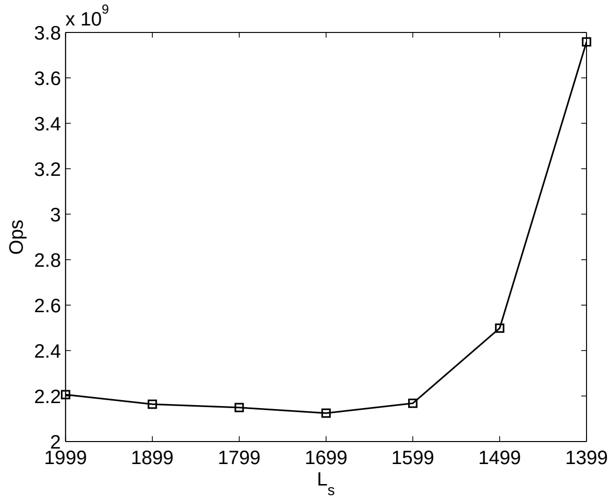

Figure 5: Total operations used for different L_s.

$$Ops = 2(ML_s)^2 \times iterations \tag{13}$$

may be reduced. The Ops against L_s for $L_r = 160$ is plotted in Fig. 5. We can see that the lowest computational complexity is given by $L_s = L_c - 300$. Compared with $(L + L_s - 1) \times (ML_s)^2 + (ML_s)^3/3 \approx 6 \times 10^{10}$ [16][4], which is needed for the computation of matrix inverse when using MINT, we can see a reduction by a factor of about 30.

It can also be seen from Table 1 that, for the same L_s, fewer iterations are needed when using larger L_r. For some combinations of L_r and L_s, the tail energy converges above -30 dB.

4. CONCLUSION

In this paper, two algorithms for shortening of multichannel RIRs have been introduced. It has been shown that, compared with inverse filtering, channel shortening is more efficient in suppression of the tail of the RIRs and in computational complexity. The CG_{CS} algorithm is more effective than the SD_{CS}.

REFERENCES

[1] S. Gannot and M. Moonen, "Subspace methods for multimicrophone speech dereverberation," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 11, pp. 1074–1090, 2003.

[2] M. Miyoshi and K. Kaneda, "Inverse filtering of room acoustics," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 145–152, 1988.

[3] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2005.

[4] N. D. Gaubitch and P. A. Naylor, "Equalization of multichannel acoustic systems in oversampled subbands," *to appear in IEEE Trans. Audio, Speech, Language Processing*, 2009.

[5] W. Zhang, A. W. H. Khong, and P. A. Naylor, "Adaptive inverse filtering of room acoustics," in *Asilomar Conf. Signals, Systems, and Computers*, 2008.

[6] K. Furuya and A. Kataoka, "Robust speech dereverberation using multichannel blind deconvolution with spectral subtraction," *IEEE Trans. Speech Audio Processing*, vol. 15, pp. 1579–1591, 2007.

[7] N. Al-Dhahir, "FIR channel shortening equalizers for MIMO ISI channels," *IEEE Trans. Commun.*, vol. 49, pp. 213–218, 2001.

[8] R. K. Martin, J. M. Walsh, and C. R. Johnson Jr., "Low-complexity MIMO blind, adaptive channel shortening," *IEEE Trans. Signal Processing*, vol. 53, pp. 1324–1334, 2005.

[9] P. J. W. Melsa, R. C. Younce, and C. E. Rohrs, "Impulse response shortening for discrete multitone transceivers," *IEEE Trans. Commun.*, vol. 44, pp. 1662–1672, 1996.

[10] M. Nafie and A. Gatherer, "Time-domain equalizer training for ADSL," in *IEEE Int. Conf. Commun.*, 1997.

[11] R. K. Martin, J. Balakrishnan, W. A. Sethares, and C. R. Johnson Jr., "A blind, adaptive TEQ for multicarrier systems," *IEEE Signal Processing Lett.*, vol. 9, pp. 341–343, 2002.

[12] R. K. Martin, K. Vanbleu, M. Ding, G. Ysebaert, M. Milosevic, B. L. Evans, M. Moonen, and C. R. Johnson Jr., "Unification and evaluation of equalization structures and design algorithms for discrete multitone modulation systems," *IEEE Trans. Signal Processing*, vol. 53, pp. 3880–3894, 2005.

[13] M. Kallinger and A. Mertins, "Multi-channel room impulse response shaping - a study," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2006.

[14] H. Kuttruff, *Room Acoustics, 4th edition*. Taylor & Frances, 2000.

[15] S. S. Haykin, *Adaptive filter theory, 4th edition*. Prentice Hall, 2002.

[16] G. H. Golub and C. F. van Loan, *Matrix computations, 3rd edition*. London: John Hopkins University Press, 1996.

[17] J. Y. C. Wen, N. D. Gaubitch, E. A. P. Habets, T. Myatt, and P. A. Naylor, "Evaluation of speech dereverberation algorithms using the MARDY database," in *Proc. Int. Workshop Acoust. Echo Noise Control*, 2006.