ACOUSTIC ECHO CANCELLATION FOR TWO AND MORE REPRODUCTION CHANNELS

Herbert Buchner and Walter Kellermann

Telecommunications Laboratory,
University of Erlangen-Nuremberg
Cauerstr. 7, D-91058 Erlangen, Germany
{buchner,wk}@LNT.de

ABSTRACT

In this paper we consider an efficient realization of acoustic echo cancellation in the frequency domain for more than two reproduction channels. Simulation results for up to five audio channels with real world data show remarkably good performance. Moreover, we propose some refinements of the original concept to keep the computational complexity moderate.

1. INTRODUCTION

For various applications, such as home entertainment, virtual reality (e.g. games, simulations, training), or advanced teleconferencing there is a growing interest in multimedia terminals with an increased number of audio channels for sound reproduction (e.g., stereo or 5.1 channel - surround systems). In such applications, multi-channel acoustic echo cancellation (M-C AEC) is a key technology whenever hands-free and full-duplex communication is desired (*Fig.* 1). However, even for only two channels (stereo), most adaptive algorithms are very complex and/or show poor convergence behaviour. For these reasons, the choice of algorithms with acceptable properties is very limited in this case and results for real-world conditions have not yet been presented for more than two channels. In the following, we examine a recently proposed frequency-domain adaptive filtering scheme [4] and introduce some extensions for the case of more than two loudspeaker channels. The results using this approach show a remarkable performance at a relatively moderate computational complexity. All considerations in this contribution are referring to only one microphone in the receiving room (*Fig.* 1) but can be easily and efficiently generalized to multichannel sound recording as discussed in [5].

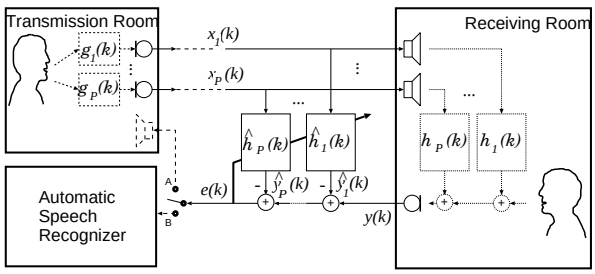

Figure 1: Conventional M-C AEC structure

2. EFFICIENT MULTICHANNEL AEC IN THE FREQUENCY DOMAIN

Frequency-domain adaptive filtering (FDAF) is well known for its very low complexity and increased convergence speed when properly designed. However, only recently a rigorous derivation of FDAF allowing a straightforward and powerful generalization to the multi-channel case has been presented [4].

2.1. Basic concept

In the following we assume a block size equal to the filter length. Generalization to a partitioned version is straightforward.

The two advantages of this structure, low complexity and fast convergence, are based on linear filtering via fast convolution as well as the approximate diagonalization of Toeplitz matrices by the Discrete Fourier Transform (DFT) [7]. For the fast convolution we apply here the overlap-save (OLS) method [1].

In order to compensate the P electro-acoustic echo paths (*Fig.* 1) to one microphone (signal $y(k)$) in the frequency domain, we first transform a block (length $2L$) of each loudspeaker signal x_i, $i = 1, \ldots, P$ to the discrete Fourier domain,

$$\boldsymbol{X}_i(m) = diag\{\boldsymbol{F}[x_i(m\frac{L}{\alpha} - L + 1) \cdots x_i(m\frac{L}{\alpha} + L)]^T\}. \quad (1)$$

$\boldsymbol{F}$ denotes the DFT matrix, L is the modeling filter length, m is the block index over time, and $1 \leq \alpha < L$ denotes a factor capturing the overlap of successive blocks which balances the the computational complexity versus the number of iterations (leading to faster convergence).

The echo replicas $\hat{\boldsymbol{Y}}_i(m)$ in the DFT domain are then generated by multiplying the weights $\hat{\boldsymbol{H}}_i(m)$ with the elements of the diagonal matrix $\boldsymbol{X}_i(m)$. The block of residual errors (including an OLS constraint [4] with the data window $\boldsymbol{W} = diag\{[\boldsymbol{0}_{1\times L} \;\; \boldsymbol{1}_{1\times L}]\}$, where $\boldsymbol{0}_{1\times L}$ and $\boldsymbol{1}_{1\times L}$ denote length L row vectors consisting of zeros and ones, respectively) becomes

$$\begin{aligned}\tilde{\boldsymbol{e}}(m) \;\; &= \;\; [\boldsymbol{0}_{1\times L} \;\; y(m\frac{L}{\alpha} + 1) \cdots y(m\frac{L}{\alpha} + L)]^T - \\ &\quad - \boldsymbol{W}\boldsymbol{F}^{-1} \sum_{i=1}^{P} \boldsymbol{X}_i(m)\hat{\boldsymbol{H}}_i(m). \qquad (2)\end{aligned}$$

This vector serves both as output signal (last $\frac{L}{\alpha}$ elements of $\tilde{\boldsymbol{e}}(m)$) and as feedback for the next adaptation step in order

to identify and track the impulse responses for the models. Using the transformed error vectors $\tilde{\boldsymbol{E}}_b(m) = \boldsymbol{F}\tilde{\boldsymbol{e}}_b(m)$, the filter weights are adapted using a recursive least-squares error criterion in the frequency domain [4]. This leads to the following update equation for $\hat{\boldsymbol{H}} = [\hat{\boldsymbol{H}}_1^T \hat{\boldsymbol{H}}_2^T \cdots \hat{\boldsymbol{H}}_P^T]^T$:

$$\hat{\boldsymbol{H}}(m+1) = \hat{\boldsymbol{H}}(m) + \mu \boldsymbol{S}^{-1}(m)\boldsymbol{X}^H(m)\tilde{\boldsymbol{E}}(m), \quad (3)$$

where

$$\boldsymbol{X} = [\boldsymbol{X}_1 \boldsymbol{X}_2 \cdots \boldsymbol{X}_P] \quad (4)$$

and

$$\boldsymbol{S}(m) = (1-\lambda)\sum_{p=0}^{m} \lambda^{m-p}\boldsymbol{X}^H(p)\boldsymbol{X}(p) \quad (5)$$

in the *unconstrained* version [4] of the algorithm (The constraint removed here is different from the one in [6]). λ is a forgetting factor close to, but less than one. Eqs. (3) and (5) are a good approximation to the well-known recursive least-squares (RLS) algorithm in the time domain (e.g. [3, 1]) which provides optimum convergence speed at very high cost. We call here the product

$$\boldsymbol{K}(m) := \boldsymbol{S}^{-1}(m)\boldsymbol{X}^H(m) \quad (6)$$

in Eq. (3) the *frequency-domain Kalman gain* in analogy to the RLS algorithm [3]. In our case, however the matrices $\boldsymbol{S}$ to be inverted and $\boldsymbol{X}$ are block diagonal (in the unconstrained version), so that each submatrix $\boldsymbol{S}_{i,j}$ $(i,j = 1,\ldots,P)$ representing a cross power spectrum is diagonal. This property follows from the approximate diagonalization of Toeplitz matrices (correlation matrix in the time-domain) by the DFT [7].

Eq. (6) is the solution of a $P \times P$ system of linear equations of block matrices. This allows decomposition of Eq. (3) into P single-channel update equations

$$\hat{\boldsymbol{H}}_i(m+1) = \hat{\boldsymbol{H}}_i(m) + \mu \boldsymbol{K}_i \tilde{\boldsymbol{E}}(m). \quad (7)$$

with modified Kalman gains $\boldsymbol{K}_i(m)$ taking the cross-correlations between the channels into account. These decomposed update equations can then be calculated element-wise and the (cross) power spectra are estimated recursively:

$$\boldsymbol{S}_{i,j}(m) = \lambda \boldsymbol{S}_{i,j}(m-1) + (1-\lambda)\boldsymbol{X}_i^*(m)\boldsymbol{X}_j(m), \quad (8)$$

where $\boldsymbol{S}_{j,i}(\cdot) = \boldsymbol{S}_{i,j}^*(\cdot)$. For the two-channel case, the update equations can be very easily written in an explicit form [1], e.g. for the first channel, we have

$$\begin{aligned} \hat{\boldsymbol{H}}_1(m+1) &= \hat{\boldsymbol{H}}_1(m) + \mu \boldsymbol{S}_1^{-1}[\boldsymbol{X}_1^*(m) - \\ &\quad -\boldsymbol{S}_{1,2}\tilde{\boldsymbol{S}}_{2,2}^{-1}\boldsymbol{X}_2^*(m)]\tilde{\boldsymbol{E}}(m), \quad (9) \\ \boldsymbol{S}_i(m) &= \tilde{\boldsymbol{S}}_{i,i}(m)[\boldsymbol{I}_{2L\times 2L} - \\ &\quad -\boldsymbol{S}_{1,2}^*(m)\boldsymbol{S}_{1,2}(m)\{\tilde{\boldsymbol{S}}_{1,1}(m)\tilde{\boldsymbol{S}}_{2,2}(m)\}^{-1}]. \end{aligned}$$

For robust adaptation the power spectral densities $\boldsymbol{S}_{i,i}$ are regularized according to $\tilde{\boldsymbol{S}}_{i,i} = \boldsymbol{S}_{i,i} + diag\{\boldsymbol{\delta}_i\}$ in these equations. For this purpose we propose here a *bin-selective dynamical regularization vector*

$$\boldsymbol{\delta}_i(m) = \delta_{max}[e^{-S_{i,i}^{(0)}(m)/S_0} \cdots e^{-S_{i,i}^{(2L-1)}(m)/S_0}]^T \quad (10)$$

with two scalar parameters δ_{max} and S_0. $S_{i,i}^{(\nu)}$ denotes the ν-th frequency component $(\nu = 0,\cdots,2L-1)$ on the main diagonal of $S_{i,i}$. This method yields improved results compared to fixed regularization or the popular approach of choosing the maximum out of the respective component $\boldsymbol{S}_{i,i}^{(\nu)}$ and a fixed threshold δ_{th}.

2.2. Efficient calculation of the frequency-domain Kalman gain

The solutions of $\boldsymbol{S}(m)\boldsymbol{K}(m) = \boldsymbol{X}^H(m)$ (Eq. (6)) for more than two channels may be formulated similarly to the corresponding part of the stereo update equations in Eq. (9) (e.g. using Cramer's rule). For three channels, we have (omitting, for simplicity, the time index m of all matrices)

$$\begin{aligned} \boldsymbol{K}_1 &= \boldsymbol{D}^{-1}[\boldsymbol{X}_1^*(\boldsymbol{S}_{2,2}\boldsymbol{S}_{3,3} - \boldsymbol{S}_{3,2}\boldsymbol{S}_{2,3}) - \boldsymbol{X}_2^*(\boldsymbol{S}_{1,2}\boldsymbol{S}_{3,3} - \\ &\quad -\boldsymbol{S}_{1,3}\boldsymbol{S}_{3,1}) - \boldsymbol{X}_3^*(\boldsymbol{S}_{1,3}\boldsymbol{S}_{2,2} - \boldsymbol{S}_{1,2}\boldsymbol{S}_{2,3})], \\ \boldsymbol{D} &:= \boldsymbol{S}_{1,1}(\boldsymbol{S}_{2,2}\boldsymbol{S}_{3,3} - \boldsymbol{S}_{3,2}\boldsymbol{S}_{2,3}) - \boldsymbol{S}_{2,1}(\boldsymbol{S}_{1,2}\boldsymbol{S}_{3,3} - \\ &\quad -\boldsymbol{S}_{1,3}\boldsymbol{S}_{3,1}) - \boldsymbol{S}_{3,1}(\boldsymbol{S}_{1,3}\boldsymbol{S}_{2,2} - \boldsymbol{S}_{1,2}\boldsymbol{S}_{2,3}) \quad (11) \end{aligned}$$

as the first of the three Kalman gain components with the common factor $\boldsymbol{D}$. The representations of Eqs. (9) and (11) allow an intuitive interpretation including a correction of the interchannel-correlations in $\boldsymbol{K}_i$ between $\boldsymbol{X}_i^*$ and the other input signals $\boldsymbol{X}_j^*$, $j \neq i$.

However, for a *practical implementation* of a system with increased number of channels, we propose computationally more efficient methods to calculate Eq. (6).

Due to the block diagonal structure of this equation, it can be trivially decomposed into $2L$ equations

$$\boldsymbol{K}^{(\nu)}(m) = (\boldsymbol{S}^{(\nu)})^{-1}(m)(\boldsymbol{X}^{(\nu)})^H(m) \quad (12)$$

with (small) $P \times P$ unitary and positive definite matrices $\boldsymbol{S}^{(\nu)}$ for the components $\nu = 0,\ldots,2L-1$ on the diagonals. Both $\boldsymbol{K}^{(\nu)}$ and $\boldsymbol{X}^{(\nu)}$ are *vectors* of length P. Note that for real input signals x we need to solve Eq. (12) only for $L+1$ bins.

A well-known and numerically stable method for this type of problems is the Cholesky decomposition of $\boldsymbol{S}^{(\nu)}$ followed by solution via backsubstitution, e.g. [8]. It leads to reduced complexity and a formulation of the algorithm which does not explicitly depend on P. The resulting total complexity for one output value is then

$$O(P \cdot log(2L)) + O(P^3), \quad (13)$$

where in the stereo case the second term $O(P^3)$ is much smaller than the share due to the first term.

For a large number of loudspeakers (e.g. for surround sound or sound field synthesis for virtual reality) we introduce a recursive solution of Eq. (12) that jointly estimates the *inverse* power spectra $(\boldsymbol{S}^{(\nu)})^{-1}$ (Eq. (8)) using the matrix-inversion lemma, e.g. [3]. This lemma relates a matrix

$$\boldsymbol{A} = \boldsymbol{B}^{-1} + \boldsymbol{C}\boldsymbol{D}^{-1}\boldsymbol{C}^H \quad (14)$$

to its inverse according to

$$\boldsymbol{A}^{-1} = \boldsymbol{B} - \boldsymbol{B}\boldsymbol{C}(\boldsymbol{D} + \boldsymbol{C}^H\boldsymbol{B}\boldsymbol{C})^{-1}\boldsymbol{C}^H\boldsymbol{B}, \quad (15)$$

as long as $\boldsymbol{A}$ and $\boldsymbol{B}$ are positive definite. Comparing Eq. (8) to Eq. (14) ($\boldsymbol{A} = \boldsymbol{S}^{(\nu)}(m)$, $\boldsymbol{B}^{-1} = \lambda \boldsymbol{S}^{(\nu)}(m-1)$, $\boldsymbol{C} = (\boldsymbol{X}^{(\nu)})^H$, $\boldsymbol{D}^{-1} = 1-\lambda$), we immediately obtain an update equation for the *inverse* matrices

$$(\boldsymbol{S}^{(\nu)}(m))^{-1} = \lambda^{-1}[(\boldsymbol{S}^{(\nu)}(m-1))^{-1} - \\ - \frac{\lambda^{-1}(\boldsymbol{S}^{(\nu)}(m-1))^{-1}\boldsymbol{X}^{(\nu)H}(m)\boldsymbol{X}^{(\nu)}(m)(\boldsymbol{S}^{(\nu)}(m-1))^{-1}}{(1-\lambda)^{-1}+\lambda^{-1}\boldsymbol{X}^{(\nu)}(m)(\boldsymbol{S}^{(\nu)}(m-1))^{-1}\boldsymbol{X}^{(\nu)H}(m)}]. \tag{16}$$

Here, no explicit matrix inversion is necessary for the calculation of the $L+1$ vectors $\boldsymbol{K}^{(\nu)}(m) = (\boldsymbol{S}^{(\nu)}(m))^{-1}\boldsymbol{X}^{(\nu)H}(m)$.

Note that our approach should not be confused with the classical RLS approach [3] which also makes use of the matrix-inversion lemma. As we apply the lemma independently to small $P \times P$ systems (Eq. (12)) it is numerically much less critical than in the RLS algorithm. Moreover, there is no analogon to a more efficient *fast RLS* due to the different matrix structures. Periodic re-initialization is done for regularization purposes.

2.3. Efficient DFT calculation of overlapping data blocks

In this section we address the first term in Eq. (13) which is mainly determined by the DFTs of the frequency-domain adaptive filtering scheme. The $2L$-point DFT calculation in Eq. (1) has to be carried out for each of the P loudspeaker signals and is therefore most costly. Moreover, as discussed in the simulation section, an increased overlap factor α is often desirable in the multichannel case. Therefore, we aim at exploiting the overlap of the input data blocks by implementing Eq. (1) recursively as well. Note that a similar idea but using a different approach was suggested in [9] for the single-channel case.

Let us consider the ν-th element on the diagonal of $\boldsymbol{X}_i(m)$ in Eq. (1) where $x_i^{(k)}(m) = x_i(m\frac{L}{\alpha} - L + 1 + k)$ denotes the k-th component of the time domain vector (block index m) to be transformed and $w = e^{-j2\pi/2L}$:

$$X_i^{(\nu)}(m) = \sum_{k=0}^{2L-1} x_i^{(k)}(m) w^{\nu k}. \tag{17}$$

Separating the summation into one for previous and one for new input values (*Fig.* 2), followed by the introduction of the previous vector elements $\boldsymbol{x}_i^{(k)}(m-1)$ leads to

$$\begin{aligned} X_i^{(\nu)}(m) &= \sum_{k=0}^{(1-1/\alpha)2L-1} x_i^{(k)}(m) w^{\nu k} + \\ &+ \sum_{k=(1-1/\alpha)2L}^{2L-1} x_i^{(k)}(m) w^{\nu k} \\ &= \sum_{k=2L/\alpha-1}^{2L-1} x_i^{(k)}(m-1) w^{\nu(k-(2L/\alpha-1))} + \\ &+ \sum_{k=(1-1/\alpha)2L}^{2L-1} x_i^{(k)}(m) w^{\nu k}. \end{aligned} \tag{18}$$

Next, we introduce the previous DFT output values $X_i^{(\nu)}(m-1)$ by subtracting the vector elements of $\boldsymbol{x}_i^{(k)}(m-1)$ of the previous data vector shifted out of the DFT length $2L$. In the second sum we shift the index k. The result

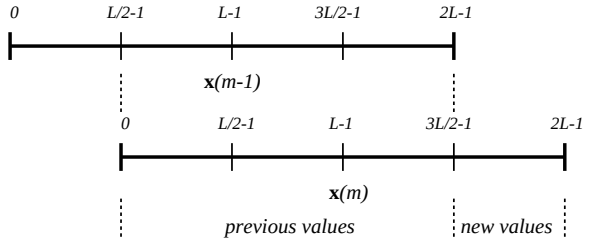

Figure 2: Example: overlapping data blocks, $\alpha = 4$

$$\begin{aligned} X_i^{(\nu)}(m) &= w^{-\nu(2L/\alpha-1)} \left[\sum_{k=0}^{2L-1} x_i^{(k)}(m-1) w^{\nu k} - \right. \\ &\left. - \sum_{k=0}^{2L/\alpha-2} x_i^{(k)}(m-1) w^{\nu k} \right] + \\ &+ w^{\nu(1-1/\alpha)2L} \sum_{\tilde{k}=0}^{2L/\alpha-1} x_i^{((1-1/\alpha)2L+\tilde{k})}(m) w^{\nu\tilde{k}} \end{aligned} \tag{19}$$

can finally be written as

$$\begin{aligned} X_i^{(\nu)}(m) &= w^{\nu(1-2L/\alpha)}[X_i^{(\nu)}(m-1) - \Delta X_i^{(\nu)}(m-\alpha)] + \\ &+ w^{\nu(1-1/\alpha)2L} \Delta X_i^{(\nu)}(m) + x_i^{(0)}(m) \end{aligned} \tag{20}$$

Again, this recursive update needs to be carried out only for $L+1$ bins. Only the update $\Delta X_i^{(\nu)}(m)$ in this equation has to be calculated explicitly using the $2L/\alpha$ new values of the input vector.

With the sparseness of the time-domain input vector for calculating $\Delta X_i^{(\nu)}(m)$ in mind, we consider now the decimation-in-frequency FFT algorithm. *Fig.* 3 shows a very simple example for $2L = 8$ and $\alpha = 4$. $2L - 2L/\alpha$ inputs (thin lines) always carry zero value. As can be seen from the figure, the first $\log_2(\alpha)$ stages do not contain any summations while for the following stages any FFT algorithm (e.g. from highly optimized software libraries) can be employed. Note that for the decimation-in-time approach one would need a special FFT implementation for all stages in order to take advantage from high oberlapping factors α. In summary the recursive DFT approach reduces the first term of the complexity in Eq. (13) to $O(P \cdot log(2L/\alpha))$ for each output point.

3. SIMULATION RESULTS

For the simulations, a speech signal (in the transmission room) was convolved by P different room impulse responses and nonlinearly, but inaudibly preprocessed according to [1] (P different nonlinearities with factor 0.5). The lengths of the receiving room impulse responses were 4096 and the modeling filters were 1024, respectively. A white noise signal for $SNR = 35dB$ was added to the echo on the microphone. *Fig.* 4 shows the misalignment convergence of the described algorithm (solid), $\alpha = 4$ for the multi-channel

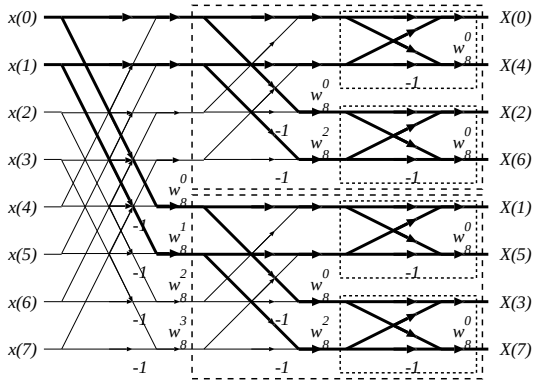

Figure 3: Illustration of decimation-in-frequency FFT with windowed input

cases $P = 2, 3, 4, 5$ (from lowest to uppermost line). In *Fig.* 5 the overlap factor α was adjusted to 8 for $P = 3, 4$ and to 16 for $P = 5$. One can clearly see that these lines are then almost indistinguishable. The dashed lines show the corresponding characteristics for the basic NLMS algorithm [3].

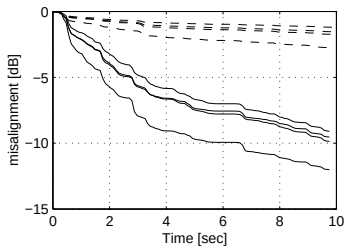

Figure 4: Convergence for P=2,3,4,5 channels, $\alpha = 4$

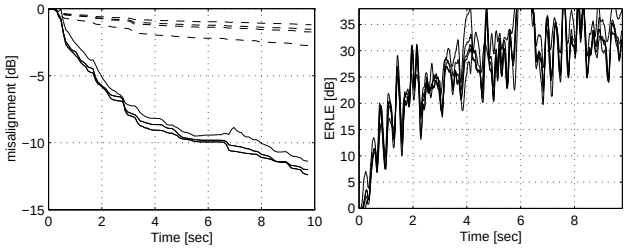

Figure 5: Convergence for for P=2,3,4,5 channels and adjusted alpha

4. CONCLUSIONS

Multichannel acoustic echo cancellation in a computationally efficient frequency-domain framework with results for real-world conditions have been presented for more than two reproduction channels. The convergence curves of the system misalignment and the ERLE clearly indicate that the approach considered here is able to cope very well with scenarios such as 5 channel surround sound with highly correlated loudspeaker signals. In order to improve the computational efficiency further, we examined different ways including a new recursive method to calculate the frequency-domain Kalman gain which plays an important role in this framework. The recursive method is based on a decomposition of block diagonal matrices followed by an application of the matrix-inversion lemma. Moreover, we have shown how the DFTs for the overlapping input data blocks can be calculated recursively as well, while the flexibility of using any FFT implementation is maintained.

The fast convergence and the very weak assumptions on the loudspeaker signals make the described approach very versatile.

5. ACKNOWLEDGEMENT

The authors would like to thank Jacob Benesty of Bell Laboratories, Lucent Technologies for very interesting and stimulating discussions.

6. REFERENCES

[1] S. L. Gay and J. Benesty (eds.), *Acoustic Signal Processing for Telecommunication*, Kluwer Academic Publishers, 2000.

[2] M. M. Sondhi and D. R. Morgan, "Stereophonic Acoustic Echo Cancellation - An Overview of the Fundamental Problem," *IEEE SP Letters*, vol.2, no.8, pp. 148–151, Aug. 1995.

[3] S. Haykin, *Adaptive Filter Theory*, 3rd ed., Prentice Hall Inc., Englewood Cliffs, NJ, 1996

[4] J. Benesty and D. R. Morgan, "Frequency-domain adaptive filtering revisited, generalization to the multichannel case, and application to acoustic echo cancellation," in *Proc. IEEE ICASSP*, Istanbul, Turkey, pp. 789-792, June 2000.

[5] H. Buchner, W. Herbordt, and W. Kellermann, "An Efficient Combination of Multi-Channel Acoustic Echo Cancellation With a Beamforming Microphone Array," Proc. Int. Workshop on Hands-Free Speech Communication, Kyoto, Japan, pp. 55-58, April 2001.

[6] D. Mansour and A. H. Gray, "Unconstrained Frequency-Domain Adaptive Filter," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol.30, no.5, Oct. 1982.

[7] R. M. Gray, "On the Asymptotic Eigenvalue Distribution of Toeplitz Matrices," *IEEE Trans. on Information Theory*, vol.18, no.6, pp. 725–730, Nov. 1972.

[8] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins, Baltimore, MD, 1989.

[9] D. W. E. Schobben, G. P. M. Egelmeers, and P. C. W. Sommen, "Efficient Realization of the Block Frequency Domain Adaptive Filter," in *Proc. IEEE ICASSP*, Munich, Germany, pp. 2257–2260, April 1997.