



Generalized multichannel frequency-domain adaptive filtering: efficient realization and application to hands-free speech communication

Herbert Buchner^{a,*}, Jacob Benesty^b, Walter Kellermann^a

^a*Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, Cauerstr. 7, 91058 Erlangen, Germany*

^b*Université du Québec, INRS-EMT, 800 de la Gauchetière Ouest, Suite 6900, Montréal, Québec, Canada H5A 1K6*

Received 15 September 2003

Abstract

In unknown environments where we need to identify, model, or track unknown and time-varying channels, adaptive filtering has been proven to be an effective tool. In this contribution, we focus on multichannel algorithms in the frequency domain that are especially well suited for input signals which are not only auto-correlated but also highly cross-correlated among the channels. These properties are particularly important for applications like multichannel acoustic echo cancellation. Most frequency-domain algorithms, as they are well known from the single-channel case, are derived from existing time-domain algorithms and are based on different heuristic strategies, e.g. for stepsize normalization. Here, we present a new rigorous derivation of a whole class of multichannel adaptive filtering algorithms in the frequency domain based on a recursive least-squares criterion. Then, from the normal equation, we derive a generic adaptive algorithm in the frequency domain. Due to the rigorous approach, the proposed framework inherently takes the coherence between all input signal channels into account. An analysis of this multichannel algorithm shows that the mean-squared error convergence is independent of the input signal statistics (i.e., both auto-correlation and cross-correlation). A useful approximation provides interesting links between some well-known algorithms for the single-channel case and the general multichannel framework. We also give design rules for important parameters to optimize the performance in practice. The computational complexity is kept low by introducing several new techniques, such as a robust recursive Kalman gain computation in the frequency domain and efficient fast Fourier transform (FFT) computation tailored to overlapping data blocks. Simulation results and real-time performance for applications such as multichannel acoustic echo cancellation show the high efficiency of the approach.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Adaptive filtering; Multichannel; Frequency domain; Echo cancellation; Beamforming

*Corresponding author. Tel.: +499131 85 27106; fax: +499131 85 28849.

E-mail addresses: buchner@LNT.de (H. Buchner), benesty@inrs-emt.quebec.ca (J. Benesty), wk@LNT.de (W. Kellermann).

1. Introduction

The ability of adaptive filters to operate satisfactorily in unknown environments and to track time variations of input statistics make it a powerful tool of statistical signal processing in such diverse fields as communications, acoustics, radar, sonar, seismology, and biomedical engineering. Despite the large variety of applications, four basic classes of adaptive filtering applications may be distinguished [19]: system identification, inverse modelling, prediction, and interference cancelling.

In speech and acoustics, where all those basic classes of adaptive filtering can be found, we often have to deal with very long filters (sometimes several thousand taps), unpredictably time-varying environments, and highly non-stationary and auto-correlated signals. In addition, the simultaneous processing of multiple input streams, i.e., multichannel adaptive filtering (MC-ADF) is becoming more and more desirable for future applications. Typical examples are multichannel acoustic echo cancellation (system identification) or adaptive beamforming microphone arrays (interference cancelling).

In this article, we investigate adaptive MIMO (multiple input and multiple output) systems that are updated in the frequency domain and we show its high efficiency for the above-mentioned applications. A major difficulty for such applications has been the practically important case of highly cross-correlated input signals to the adaptive filter [7,29,30] which is the main motivation for this contribution. The resulting generalized multichannel frequency-domain adaptive filtering accounts for the cross-correlations and has already led to efficient real-time implementations of multichannel acoustic echo cancellers on standard personal computers [10,14]. Moreover, the approach shown in this contribution also provides a basis for new improved single-channel algorithms.

Generally, we distinguish two classes of adaptive algorithms. One class includes filters that are updated in the time domain, usually on a sample-by-sample basis, like the classical least-mean-square (LMS) [33] and recursive least-squares (RLS) [1] algorithms. The other class may be defined as filters that are updated in the discrete

Fourier transform (DFT) domain ('frequency domain'), block-by-block in general, exploiting the efficiency of the fast Fourier transform (FFT) so that the arithmetic complexity of this latter category is significantly reduced compared to time-domain adaptive algorithms.

As preliminaries to our generalization of frequency-domain adaptive filtering, we briefly summarize already known concepts and algorithms. Single-channel frequency-domain adaptive filtering was first introduced by Dentino et al., based on the LMS algorithm in the time-domain [12]. Ferrara [13] was the first to present an efficient frequency-domain adaptive filter algorithm (FLMS) that converges to the Wiener solution using the classical overlap-save (OLS) method. An even more efficient algorithm, the *unconstrained* FLMS (UFLMS) was derived by Mansour and Gray [24], using only three FFT operations per block instead of five for the FLMS, with comparable performance [23]. However, in some applications, a major drawback of these structures is the delay introduced between input and output. Indeed, for efficient implementations, this delay is equal to the length L of the adaptive filter, which is considerable for applications like acoustic echo cancellation. A new structure called *multidelay filter* (MDF), using the OLS method, was proposed in [3,31] and generalized in [4] where the new block size N is independent of the filter length L ; N can be chosen as small as desired, with a delay equal to N . Although from a complexity point of view, the optimum choice is $N = L$, using smaller block sizes ($N < L$) to reduce the delay is still more efficient than time-domain algorithms. A more general scheme, the *generalized multi-delay filter* (GMDF α) was proposed in [27,28], where α denotes an overlap factor of input data blocks. The settings $\alpha > 1$ appear to be very attractive, since the filter coefficients can be adapted more frequently (every N/α samples instead of every N samples in the conventional OLS scheme) and the delay can be (further) reduced as well. Thus, GMDF α includes the previous ones, and introduces one more degree of freedom, but the complexity is increased roughly by a factor α (which is most often still much lower than that of time-domain algorithms). Taking the block size in

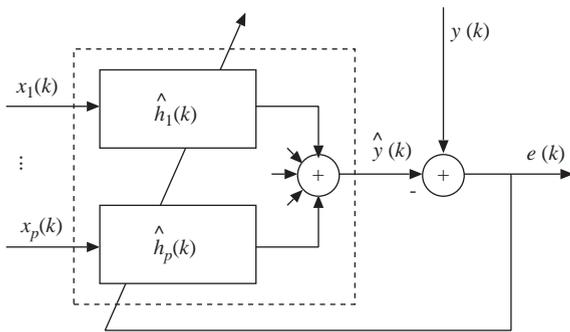


Fig. 1. Multichannel adaptive filtering.

the GMDF α as large as the delay permits will increase the convergence rate of the algorithm, while choosing the overlap factor greater than one will increase the tracking ability of the algorithm.

The case of multichannel adaptive filtering, as shown in Fig. 1, has been found to be structurally more difficult in general. In typical scenarios, the input signals to the adaptive filter $x_p(n)$, $p = 1, \dots, P$, are not only auto-correlated but also highly cross-correlated which often results in very slow convergence of the LP filter coefficients $\hat{h}_{p,\kappa}(n)$, where $\kappa = 0, \dots, L - 1$. This problem becomes particularly severe in applications like multichannel acoustic echo cancellation [7,29,30], where the signals $x_p(n)$ represent loudspeaker signals that may originate from common sources. Signal $y(n)$ represents an echo received by a microphone.

Applying common low-complexity algorithms, such as the LMS algorithm, or conventional frequency-domain adaptive filtering according to [27,28], to the multichannel case usually leads to disappointing results as the cross-correlations between the input channels are not taken into account [15]. In contrast to this, high-order affine projection algorithms and RLS algorithms do take the cross-correlations into account [2,15]. Indeed, it can be shown that the RLS provides optimum convergence speed even in the multichannel case [15], but its complexity is prohibitively high and will, e.g., not allow real-time implementation of multichannel acoustic echo cancellation on standard hardware any time soon.

Exploiting the efficiency of FFT algorithms is possible because of the Toeplitz structure of the

matrices involved, which results from the transversal structure of the adaptive filters: The Toeplitz matrices can be expressed by circulant matrices which are diagonalizable by the DFT. Consequently, the key for deriving the frequency-domain adaptive algorithms is to reformulate the time-domain error criterion so that Toeplitz and circulant matrices are explicitly shown. This idea was already followed in [24] for an LMS-like derivation (i.e., based on gradient descent with stochastic approximation), and the result has been used for the subsequent algorithms mentioned above. Unfortunately, this procedure does not lead to a completely systematic solution, e.g., to the stepsize normalization (which is crucial in the multichannel case), and does not directly lead to the *constrained FLMS*.

Two-channel frequency-domain adaptive filtering was first introduced in [6] in the context of stereophonic acoustic echo cancellation and derived from the extended least-mean-squares (ELMS) algorithm [2] in the time domain using similar heuristic considerations as for the single-channel case outlined above.

Unlike the above at least partially heuristic derivations, the rigorous derivation of frequency-domain adaptive filtering presented in the next section leads to a generic algorithm with RLS-like properties. An efficient approximation of this algorithm can then be studied in a way that is independent of the signals. While this approximation provides a direct link to the existing and interesting new frequency-domain algorithms in the single-channel case, the property of a tight coupling of the input channels in the multichannel case is efficiently preserved. For clarity, we will confine the detailed derivation to a block size $N = L$. A generalization to $N \leq L$ is straightforward; we will point to interesting new results along the lines.

The organization of this paper is as follows. In Section 2, we introduce the notation and a frequency-domain recursive least-squares criterion from which the so-called normal equation is derived. Then, from the normal equation, we deduce a generic multichannel adaptive algorithm and we introduce the so-called frequency-domain Kalman gain. In Section 3, we study the convergence of this novel class of multichannel

algorithms. In Section 4, we consider the general MIMO case and, in Section 5, we give a very useful approximation that can be studied independently of the signals, deduce some well-known single-channel algorithms as special cases, and explicitly show how the cross-correlations are taken into account in the multichannel case. We also give design rules for some important parameters such as the exponential window, regularization, and adaptation stepsize. A useful dynamical regularization method is discussed in more detail in Section 6. Section 7 introduces several methods for increasing computational efficiency in the multi-input and MIMO cases, such as a robust recursive Kalman gain computation and FFT computation tailored for overlapping data blocks. Section 8 presents some simulations and multichannel real-world implementations for hands-free speech communications. Finally, our results are summarized in Section 9.

2. General derivation of multichannel frequency-domain algorithms

In the first part of this section we formulate a block recursive least-squares criterion in the frequency domain. Once the criterion is rigorously defined, the adaptive multichannel algorithm follows immediately.

2.1. Optimization criterion

From Fig. 1, it can be seen that the error signal at time k between the output of the multichannel adaptive filter $\hat{y}(k)$ and the desired output signal $y(k)$ is given by

$$e(k) = y(k) - \hat{y}(k), \tag{1}$$

with

$$\hat{y}(k) = \sum_{p=1}^P \mathbf{x}_p^T(k) \hat{\mathbf{h}}_p = \mathbf{x}^T(k) \hat{\mathbf{h}}, \tag{2}$$

where

$$\mathbf{x}_p(k) = [x_p(k), x_p(k-1), \dots, x_p(k-L+1)]^T \tag{3}$$

is a vector containing the latest L samples of the input signal x_p of the p th channel, and where

$$\hat{\mathbf{h}}_p = [\hat{h}_{p,0}, \hat{h}_{p,1}, \dots, \hat{h}_{p,L-1}]^T \tag{4}$$

contains the current weights of the adaptive FIR filter taps for the p th input channel. The vectors

$$\mathbf{x}(k) = [\mathbf{x}_1^T(k), \mathbf{x}_2^T(k), \dots, \mathbf{x}_P^T(k)]^T \tag{5}$$

and

$$\hat{\mathbf{h}} = [\hat{\mathbf{h}}_1^T, \hat{\mathbf{h}}_2^T, \dots, \hat{\mathbf{h}}_P^T]^T \tag{6}$$

allow a convenient notation of the multichannel algorithms. Superscript T denotes transposition of a vector or a matrix.

We now define the block error signal of length L . Based on (1) and (2) we write

$$\mathbf{e}(m) = \mathbf{y}(m) - \hat{\mathbf{y}}(m), \tag{7}$$

with m being the block time index, and

$$\hat{\mathbf{y}}(m) = \sum_{p=1}^P \mathbf{U}_p^T(m) \hat{\mathbf{h}}_p = \mathbf{U}^T(m) \hat{\mathbf{h}}, \tag{8}$$

where

$$\mathbf{e}(m) = [e(mL), \dots, e(mL+L-1)]^T, \tag{9}$$

$$\mathbf{y}(m) = [y(mL), \dots, y(mL+L-1)]^T, \tag{10}$$

$$\hat{\mathbf{y}}(m) = [\hat{y}(mL), \dots, \hat{y}(mL+L-1)]^T, \tag{11}$$

$$\mathbf{U}_p(m) = [\mathbf{x}_p(mL), \dots, \mathbf{x}_p(mL+L-1)], \tag{12}$$

$$\mathbf{U}(m) = [\mathbf{U}_1^T(m), \dots, \mathbf{U}_P^T(m)]^T. \tag{13}$$

It can easily be verified that \mathbf{U}_p , $p = 1, \dots, P$ are Toeplitz matrices of size $(L \times L)$:

$$\mathbf{U}_p^T(m) = \begin{bmatrix} x_p(mL) & \cdots & x_p(mL-L+1) \\ x_p(mL+1) & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x_p(mL+L-1) & \cdots & x_p(mL) \end{bmatrix}.$$

These Toeplitz matrices are now diagonalized in two steps:

Step 1: Transformation of Toeplitz matrices into circulant matrices.

Any Toeplitz matrix \mathbf{U}_p can be transformed, by doubling its size, to a circulant matrix

$$\mathbf{C}_p(m) = \begin{bmatrix} \mathbf{U}'_p{}^T(m) & \mathbf{U}_p{}^T(m) \\ \mathbf{U}_p{}^T(m) & \mathbf{U}'_p{}^T(m) \end{bmatrix}, \quad (14)$$

where the \mathbf{U}'_p are also Toeplitz matrices and can be expressed in terms of the elements of $\mathbf{U}_p{}^T(m)$, except for an arbitrary diagonal, e.g.,

$$\mathbf{U}'_p{}^T(m) = \begin{bmatrix} x_p(mL - L) & \cdots & x_p(mL + 1) \\ x_p(mL - L + 1) & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ x_p(mL - 1) & \cdots & x_p(mL - L) \end{bmatrix}.$$

It follows

$$\mathbf{U}_p{}^T(m) = \mathbf{W}_{L \times 2L}^{01} \mathbf{C}_p(m) \mathbf{W}_{2L \times L}^{10}, \quad (15)$$

where we introduced the windowing matrices

$$\mathbf{W}_{L \times 2L}^{01} = [\mathbf{0}_{L \times L}, \mathbf{I}_{L \times L}],$$

$$\mathbf{W}_{2L \times L}^{10} = [\mathbf{I}_{L \times L}, \mathbf{0}_{L \times L}]^T.$$

Step 2: Transformation of the circulant matrices into diagonal matrices.

Using the $2L \times 2L$ DFT matrix $\mathbf{F}_{2L \times 2L}$ with elements $e^{-j2\pi vk/(2L)}$, where $v, k = 0, \dots, 2L - 1$, the circulant matrices are diagonalized as follows:

$$\mathbf{C}_p(m) = \mathbf{F}_{2L \times 2L}^{-1} \mathbf{X}_p(m) \mathbf{F}_{2L \times 2L}, \quad (16)$$

where the diagonal matrices $\mathbf{X}_p(m)$ can be expressed by the first columns of $\mathbf{C}_p(m)$,

$$\mathbf{X}_p(m) = \text{diag} \left\{ \mathbf{F}_{2L \times 2L} \begin{bmatrix} x_p(mL - L) \\ \vdots \\ x_p(mL + L - 1) \end{bmatrix} \right\}. \quad (17)$$

Now, (15) can be rewritten equivalently as

$$\mathbf{U}_p{}^T(m) = \mathbf{W}_{L \times 2L}^{01} \mathbf{F}_{2L \times 2L}^{-1} \mathbf{X}_p(m) \mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times L}^{10}. \quad (18)$$

Since

$$\begin{aligned} & [\mathbf{A}\mathbf{X}_1\mathbf{B}, \dots, \mathbf{A}\mathbf{X}_p\mathbf{B}] \\ &= \mathbf{A}[\mathbf{X}_1, \dots, \mathbf{X}_p] \text{diag}\{\mathbf{B}, \dots, \mathbf{B}\} \end{aligned}$$

for any matrices \mathbf{A} , \mathbf{B} , \mathbf{X}_p with compatible dimensions, it follows for the error vector using (13) and (18):

$$\begin{aligned} \mathbf{e}(m) &= \mathbf{y}(m) - \mathbf{W}_{L \times 2L}^{01} \mathbf{F}_{2L \times 2L}^{-1} [\mathbf{X}_1(m), \dots, \mathbf{X}_p(m)] \\ &\quad \cdot \text{diag}\{\mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times L}^{10}, \dots, \mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times L}^{10}\} \hat{\mathbf{h}}. \end{aligned} \quad (19)$$

If we multiply (19) by the $L \times L$ DFT matrix $\mathbf{F}_{L \times L}$, we obtain the error signal in the frequency domain¹:

$$\underline{\mathbf{e}}(m) = \underline{\mathbf{y}}(m) - \mathbf{G}_{L \times 2L}^{01} \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10} \hat{\mathbf{h}}, \quad (20)$$

where

$$\underline{\mathbf{e}}(m) = \mathbf{F}_{L \times L} \mathbf{e}(m), \quad (21)$$

$$\underline{\mathbf{y}}(m) = \mathbf{F}_{L \times L} \mathbf{y}(m), \quad (22)$$

$$\mathbf{G}_{L \times 2L}^{01} = \mathbf{F}_{L \times L} \mathbf{W}_{L \times 2L}^{01} \mathbf{F}_{2L \times 2L}^{-1}, \quad (23)$$

$$\mathbf{G}_{2LP \times LP}^{10} = \text{diag}\{\mathbf{G}_{2L \times L}^{10}, \dots, \mathbf{G}_{2L \times L}^{10}\}, \quad (24)$$

$$\mathbf{G}_{2L \times L}^{10} = \mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times L}^{10} \mathbf{F}_{L \times L}^{-1}, \quad (25)$$

$$\mathbf{X}(m) = [\mathbf{X}_1(m), \mathbf{X}_2(m), \dots, \mathbf{X}_p(m)], \quad (26)$$

$$\hat{\mathbf{h}}_p = \mathbf{F}_{L \times L} \hat{\mathbf{h}}_p, \quad (27)$$

$$\hat{\mathbf{h}} = [\hat{\mathbf{h}}_1^T, \hat{\mathbf{h}}_2^T, \dots, \hat{\mathbf{h}}_p^T]^T. \quad (28)$$

Optimization criterion: Having derived a frequency-domain error signal, we now define a frequency-domain criterion for optimizing the coefficient vector $\hat{\mathbf{h}} = \hat{\mathbf{h}}(m)$:

$$J_f(m) = (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \underline{\mathbf{e}}^H(i) \underline{\mathbf{e}}(i), \quad (29)$$

where \mathbf{H} denotes conjugate transposition and λ ($0 < \lambda < 1$) is an exponential forgetting factor.

¹As we will see, first using a length L DFT here, instead of a length $2L$ DFT as in the classical overlap-save technique [24] allows us to obtain the more general *constrained* adaptive filter.

The criterion (29) is very similar² to the one leading to the well-known RLS algorithm [1]. The main advantage of using (29) is to take advantage of the FFT in order to have low-complexity adaptive filters.

2.2. Normal equation

Let $\nabla_{\hat{\mathbf{h}}}$ be the gradient operator with respect to $\hat{\mathbf{h}}$. Applying the operator $\nabla_{\hat{\mathbf{h}}}$ to the cost function J_f , we obtain [8,19] the complex gradient vector:

$$\begin{aligned} \nabla_{\hat{\mathbf{h}}} J_f(m) &= 2 \frac{\partial J_f(m)}{\partial \hat{\mathbf{h}}^*} = -2(1-\lambda) \sum_{i=0}^m \lambda^{m-i} \\ &\cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) (\mathbf{G}_{L \times 2L}^{01})^H \underline{\mathbf{y}}(i) \\ &+ 2(1-\lambda) \left[\sum_{i=0}^m \lambda^{m-i} (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) \right. \\ &\left. \cdot (\mathbf{G}_{2L \times 2L}^{01}) \mathbf{X}(i) (\mathbf{G}_{2LP \times LP}^{10}) \right] \hat{\mathbf{h}}(m), \end{aligned} \quad (30)$$

where * denotes complex conjugation,

$$\begin{aligned} \mathbf{G}_{2L \times 2L}^{01} &= (\mathbf{G}_{L \times 2L}^{01})^H \mathbf{G}_{L \times 2L}^{01} \\ &= \mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times 2L}^{01} \mathbf{F}_{2L \times 2L}^{-1}, \end{aligned} \quad (31)$$

and

$$\mathbf{W}_{2L \times 2L}^{01} = \begin{bmatrix} \mathbf{0}_{L \times L} & \mathbf{0}_{L \times L} \\ \mathbf{0}_{L \times L} & \mathbf{I}_{L \times L} \end{bmatrix}. \quad (32)$$

By setting the gradient of the cost function equal to zero and defining

$$\begin{aligned} \underline{\mathbf{y}}_{2L}(m) &= (\mathbf{G}_{L \times 2L}^{01})^H \underline{\mathbf{y}}(m) \\ &= \mathbf{F}_{2L \times 2L} \begin{bmatrix} \mathbf{0}_{L \times 1} \\ \underline{\mathbf{y}}(m) \end{bmatrix}, \end{aligned} \quad (33)$$

we obtain the so-called *normal equation*

$$\mathbf{S}(m) \hat{\mathbf{h}}(m) = \mathbf{s}(m), \quad (34)$$

where

$$\begin{aligned} \mathbf{S}(m) &= (1-\lambda) \sum_{i=0}^m \lambda^{m-i} (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) \\ &\cdot \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(i) \mathbf{G}_{2LP \times LP}^{10} \\ &= \lambda \mathbf{S}(m-1) + (1-\lambda) (\mathbf{G}_{2LP \times LP}^{10})^H \\ &\cdot \mathbf{X}^H(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10} \end{aligned} \quad (35)$$

and

$$\begin{aligned} \mathbf{s}(m) &= (1-\lambda) \sum_{i=0}^m \lambda^{m-i} \cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) \underline{\mathbf{y}}_{2L}(i) \\ &= \lambda \mathbf{s}(m-1) + (1-\lambda) (\mathbf{G}_{2LP \times LP}^{10})^H \\ &\cdot \mathbf{X}^H(m) \underline{\mathbf{y}}_{2L}(m) \\ &= \lambda \mathbf{s}(m-1) + (1-\lambda) (\mathbf{G}_{2LP \times LP}^{10})^H \\ &\cdot \mathbf{X}^H(m) (\mathbf{G}_{L \times 2L}^{01})^H \underline{\mathbf{y}}(m). \end{aligned} \quad (36)$$

If the input signal is well-conditioned, matrix $\mathbf{S}(m)$ is non-singular. In this case, the normal equation has a unique solution which is the optimum Wiener solution.

2.3. Adaptive algorithm

The different formulations for filter adaptation discussed below, i.e., recursive updates of $\hat{\mathbf{h}}(m)$, are all derived directly from the normal equation (34) and associated equations (35) and (36).

Here, we replace $\mathbf{s}(m)$ and $\mathbf{s}(m-1)$ in the recursive equation (36) by formulating (34) in terms of block time indices m and $m-1$, respectively. We then eliminate $\mathbf{S}(m-1)$ from the resulting equation using (35). Reintroducing the error signal vector (20), we obtain an exact recursive solution of (34) by the following adaptation algorithm:

$$\underline{\mathbf{e}}(m) = \underline{\mathbf{y}}(m) - \mathbf{G}_{L \times 2L}^{01} \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10} \hat{\mathbf{h}}(m-1), \quad (37)$$

$$\begin{aligned} \hat{\mathbf{h}}(m) &= \hat{\mathbf{h}}(m-1) + (1-\lambda) \mathbf{S}^{-1}(m) \\ &\cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m) (\mathbf{G}_{L \times 2L}^{01})^H \underline{\mathbf{e}}(m). \end{aligned} \quad (38)$$

For practical purposes, it is useful to reformulate this algorithm. First, we multiply (37)

²Note that the time-frequency equivalence is assured by Parseval's theorem.

by $(\mathbf{G}_{L \times 2L}^{01})^H$,

$$\mathbf{e}_{2L}(m) = \mathbf{y}_{2L}(m) - \mathbf{G}_{2L \times 2L}^{01} \cdot \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10} \hat{\mathbf{h}}(m-1) \quad (39)$$

$$\hat{\mathbf{h}}(m) = \hat{\mathbf{h}}(m-1) + (1-\lambda) \mathbf{S}^{-1}(m) \cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m) \mathbf{e}_{2L}(m), \quad (40)$$

where we defined analogously to (33)

$$\mathbf{e}_{2L}(m) = (\mathbf{G}_{L \times 2L}^{01})^H \mathbf{e}(m) = \mathbf{F}_{2L \times 2L} \begin{bmatrix} \mathbf{0}_{L \times 1} \\ \mathbf{e}(m) \end{bmatrix}. \quad (41)$$

If we multiply (40) by $\mathbf{G}_{2LP \times LP}^{10}$, we obtain the algorithm (39) and (40) in a slightly different form:

$$\mathbf{e}_{2L}(m) = \mathbf{y}_{2L}(m) - \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \hat{\mathbf{h}}_{2LP}(m-1) \quad (42)$$

$$\hat{\mathbf{h}}_{2LP}(m) = \hat{\mathbf{h}}_{2LP}(m-1) + (1-\lambda) \mathbf{G}_{2LP \times LP}^{10} \cdot \mathbf{S}^{-1}(m) (\mathbf{G}_{2LP \times LP}^{10})^H \cdot \mathbf{X}^H(m) \mathbf{e}_{2L}(m), \quad (43)$$

where $\mathbf{S}(m)$ is given by (35), and

$$\hat{\mathbf{h}}_{2LP}(m) = \mathbf{G}_{2LP \times LP}^{10} \hat{\mathbf{h}}(m) = [\hat{\mathbf{h}}_{2LP,1}^T(m), \dots, \hat{\mathbf{h}}_{2LP,p}^T(m)]^T,$$

$$\hat{\mathbf{h}}_{2LP,p}(m) = \mathbf{F}_{2L \times 2L} \begin{bmatrix} \hat{\mathbf{h}}_p(m) \\ \mathbf{0}_{L \times 1} \end{bmatrix}. \quad (44)$$

The rank of the matrix $\mathbf{G}_{2LP \times LP}^{10}$ is equal to LP . Since we have to adapt LP unknowns, in principle, (43) is equivalent to (40). Indeed, if we multiply (43) by $(\mathbf{G}_{2LP \times LP}^{10})^H$, we obtain exactly (40) since $(\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{G}_{2LP \times LP}^{10} = \mathbf{I}_{LP \times LP}$. It is interesting to see how naturally we have ended up using blocks of length $2L$ (especially for the error signal) even though we have used an error criterion with blocks of length L . We can do even better than that and rewrite the algorithm exclusively using FFTs of size $2L$. This formulation is by far the most interesting one because an explicit link with existing frequency-domain algorithms can be

established. Let us first define the $(2LP \times 2LP)$ matrix

$$\begin{aligned} \mathbf{S}_d(m) &= (1-\lambda) \sum_{i=0}^m \lambda^{m-i} \mathbf{X}^H(i) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(i) \\ &= \lambda \mathbf{S}_d(m-1) \\ &\quad + (1-\lambda) \mathbf{X}^H(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m). \end{aligned} \quad (45)$$

The relation of $\mathbf{S}_d(m)$ to $\mathbf{S}(m)$ is obviously given by

$$\mathbf{S}(m) = (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{S}_d(m) \mathbf{G}_{2LP \times LP}^{10}. \quad (46)$$

Next, we define

$$\begin{aligned} \mathbf{G}_{2L \times 2L}^{10} &= \mathbf{G}_{2L \times L}^{10} (\mathbf{G}_{2L \times L}^{10})^H \\ &= \mathbf{F}_{2L \times 2L} \mathbf{W}_{2L \times 2L}^{10} \mathbf{F}_{2L \times 2L}^{-1} \end{aligned} \quad (47)$$

and

$$\mathbf{G}_{2LP \times 2LP}^{10} = \text{diag}\{\mathbf{G}_{2L \times 2L}^{10} \cdots \mathbf{G}_{2L \times 2L}^{10}\}, \quad (47)$$

where

$$\mathbf{W}_{2L \times 2L}^{10} = \begin{bmatrix} \mathbf{I}_{L \times L} & \mathbf{0}_{L \times L} \\ \mathbf{0}_{L \times L} & \mathbf{0}_{L \times L} \end{bmatrix}. \quad (48)$$

Now, we have a relation between the inverse of the two matrices \mathbf{S} (as it appears in (43)) and \mathbf{S}_d :

$$\begin{aligned} \mathbf{G}_{2LP \times 2LP}^{10} \mathbf{S}_d^{-1}(m) &= \mathbf{G}_{2LP \times LP}^{10} \mathbf{S}^{-1}(m) (\mathbf{G}_{2LP \times LP}^{10})^H. \end{aligned} \quad (49)$$

This can be verified by post-multiplying both sides of (49) by $\mathbf{S}_d(m) \mathbf{G}_{2LP \times LP}^{10}$ and noting that $\mathbf{G}_{2LP \times 2LP}^{10} \mathbf{G}_{2LP \times LP}^{10} = \mathbf{G}_{2LP \times LP}^{10}$. Using (49), the adaptive algorithm (35), (42), and (43) can now be written more conveniently:

$$\begin{aligned} \mathbf{S}_d(m) &= \lambda \mathbf{S}_d(m-1) \\ &\quad + (1-\lambda) \mathbf{X}^H(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \end{aligned} \quad (50)$$

$$\begin{aligned} \mathbf{e}_{2L}(m) &= \mathbf{y}_{2L}(m) \\ &\quad - \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \hat{\mathbf{h}}_{2LP}(m-1) \end{aligned} \quad (51)$$

$$\begin{aligned} \hat{\mathbf{h}}_{2LP}(m) &= \hat{\mathbf{h}}_{2LP}(m-1) + (1-\lambda) \mathbf{G}_{2LP \times 2LP}^{10} \\ &\quad \cdot \mathbf{S}_d^{-1}(m) \mathbf{X}^H(m) \mathbf{e}_{2L}(m). \end{aligned} \quad (52)$$

Due to the structure of the update equations, we introduce a frequency-domain Kalman gain matrix in analogy to the RLS algorithm [19]:

$$\mathbf{K}(m) = (1-\lambda) \mathbf{S}_d^{-1}(m) \mathbf{X}^H(m). \quad (53)$$

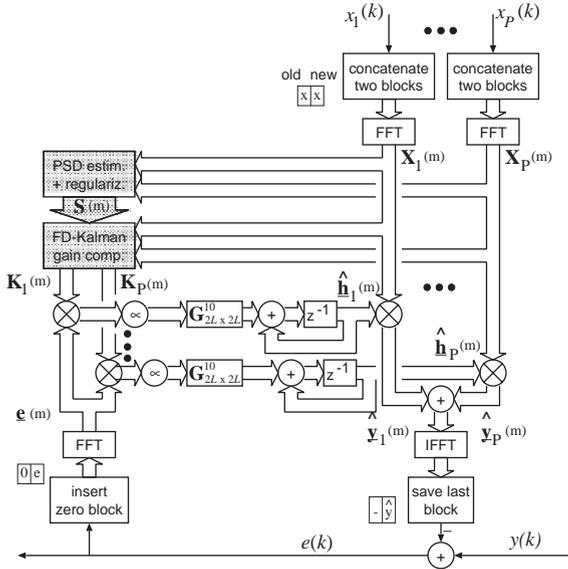


Fig. 2. Multichannel frequency-domain adaptive filtering.

This $2LP \times 2L$ matrix includes the inverse in (52) and plays an important role in practical realizations, including a tight coupling between the multiple input channels by coherence terms, as shown in detail in subsequent sections. Fig. 2 summarizes the general steps in multichannel frequency-domain adaptive filtering. The two shaded blocks represent the calculation of the Kalman gain using (50) and (53), or efficient realizations thereof.

A note concerning block sizes $N < L$: Analogously to (50)–(52), a generic algorithm can be derived straightforwardly using $K = L/N$ sub-filters per channel and block convolution. As shown in Section 5, this leads to a new class of algorithms for $N < L$ with improved convergence characteristics compared to the classical multi-delay filter [31]. Using DFTs of length N and length $2N$, respectively, the error criterion (29) is then applied to length- N error vectors. In the resulting algorithm, the matrices $\mathbf{G}_{2L \times 2L}^{01}$ in (50) and (51) are replaced by the corresponding $2N \times 2N$ -matrices $\mathbf{G}_{2N \times 2N}^{01}$. Equation (52) remains in the same form, where $\mathbf{G}_{2LP \times 2LP}^{10}$, defined in (47), is then composed from $K \cdot P$ sub-matrices of size $2N \times 2N$; $\hat{\mathbf{h}}_{2LP}(m)$ and $\mathbf{X}(m)$ are the concatena-

tions of the sub-filters and diagonal sub-matrices, respectively.

3. Convergence analysis

In this section, we analyze the convergence behavior of the algorithm for stationary signals $x_p(k)$ and $y(k)$ based on (37) and (38).

Due to the assumed stationarity of the filter input signals, we obtain, after taking the expected value of (35):

$$E\{\mathbf{S}(m)\} = (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \mathbf{S}_e, \quad (54)$$

where

$$\mathbf{S}_e = E\{(\mathbf{G}_{2LP \times LP}^{10})^H \cdot \mathbf{X}^H(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10}\} \quad (55)$$

denotes the time-independent ensemble average. Noting that in (54) we have a sum of a finite geometric series, it can be simplified to

$$E\{\mathbf{S}(m)\} = (1 - \lambda^{m+1}) \mathbf{S}_e. \quad (56)$$

For a single realization of the stochastic process $\mathbf{S}(m)$, we assume that

$$\mathbf{S}(m) \approx (1 - \lambda^{m+1}) \mathbf{S}_e, \quad (57)$$

and for the steady state we see with $0 < \lambda < 1$ that

$$\mathbf{S}(m) \approx \mathbf{S}_e \quad \text{for large } m. \quad (58)$$

3.1. Analysis model

For the following, we assume that the desired response $y(k)$ and the tap-input vector $\mathbf{x}(k)$ are related by the multiple linear regression model [19]

$$y(k) = \mathbf{x}^T(k) \mathbf{h} + n(k), \quad (59)$$

where the $LP \times 1$ vector \mathbf{h} denotes the fixed regression parameter vector of the model and the measurement error $n(k)$ is assumed to be a zero-mean white noise that is independent of $x_p(k) \forall p \in \{1, \dots, P\}$. The equivalent expression in the frequency domain reads

$$\underline{\mathbf{y}}(m) = \mathbf{G}_{L \times 2L}^{01} \mathbf{X}(m) \mathbf{G}_{2LP \times LP}^{10} \mathbf{h} + \underline{\mathbf{n}}(m), \quad (60)$$

where \mathbf{h} and $\mathbf{n}(m)$ are defined in the same way as $\hat{\mathbf{h}}$ in (28) and $\mathbf{y}(m)$ in (22), respectively.

3.2. Convergence in the mean

By noting that

$$(\mathbf{G}_{L \times 2L}^{01})^H \mathbf{G}_{L \times 2L}^{01} = \mathbf{G}_{2L \times 2L}^{01} \quad (61)$$

from (31), the coefficient update (38) can be written in terms of the misalignment vector $\underline{\mathbf{e}}(m)$ as

$$\begin{aligned} \mathbf{h} - \hat{\mathbf{h}}(m) &= \mathbf{h} - \hat{\mathbf{h}}(m-1) \\ &\quad - (1-\lambda)\mathbf{S}^{-1}(m)(\mathbf{G}_{2LP \times LP}^{10})^H \\ &\quad \cdot \mathbf{X}^H(m)\mathbf{G}_{2L \times 2L}^{01}\mathbf{X}(m) \\ &\quad \cdot \mathbf{G}_{2LP \times LP}^{10}[\mathbf{h} - \hat{\mathbf{h}}(m-1)] \\ &\quad - (1-\lambda)\mathbf{S}^{-1}(m)(\mathbf{G}_{2LP \times LP}^{10})^H \\ &\quad \cdot \mathbf{X}^H(m)\mathbf{n}(m). \end{aligned} \quad (62)$$

Taking the mathematical expectation of expression (62), using the independence theory [19], and (55) together with (58), we deduce for large m that $E\{\underline{\mathbf{e}}(m)\} = \lambda E\{\underline{\mathbf{e}}(m-1)\} = \lambda^m E\{\underline{\mathbf{e}}(0)\}$. (63)

Eq. (63) expresses that the convergence rate of the algorithm is governed by λ . Most importantly, the rate of convergence is completely independent of the input statistics (even in the multichannel case). Finally, we have

$$\lim_{m \rightarrow \infty} E\{\underline{\mathbf{e}}(m)\} = \mathbf{0}_{LP \times 1} \Rightarrow \lim_{m \rightarrow \infty} E\{\hat{\mathbf{h}}(m)\} = \mathbf{h}. \quad (64)$$

Now, suppose that λ_t is the forgetting factor of a sample-by-sample adaptive algorithm (operating in the time domain). To have the same effective window length for the sample-by-sample and block-by-block algorithms, we should choose $\lambda = \lambda_t^L$. For example, a typical choice for the RLS algorithm [19] is $\lambda_t = 1 - 1/(3L)$. In this case, a good choice for the frequency-domain algorithm is $\lambda = [1 - 1/(3L)]^L$.

3.3. Convergence of the mean-squared error

The convergence of the algorithm in the mean is not sufficient for convergence to the minimum mean-squared error (MMSE) estimate [19] as it only assures a bias-free estimate $\hat{\mathbf{h}}(m)$. The algo-

rithm converges in the mean square if

$$\lim_{m \rightarrow \infty} J'_f(m) = J'_{f,\min} < \infty, \quad (65)$$

where

$$J'_f(m) = \frac{1}{L} E\{\underline{\mathbf{e}}^H(m)\underline{\mathbf{e}}(m)\}. \quad (66)$$

From (37), the error signal $\underline{\mathbf{e}}(m)$ can be written in terms of $\underline{\mathbf{e}}(m)$ as

$$\underline{\mathbf{e}}(m) = \mathbf{G}_{L \times 2L}^{01}\mathbf{X}(m)\mathbf{G}_{2LP \times LP}^{10}\underline{\mathbf{e}}(m-1) + \mathbf{n}(m). \quad (67)$$

Expression (66) becomes

$$J'_f(m) = \frac{1}{L} J_{\text{ex}}(m) + \sigma_n^2, \quad (68)$$

where the excess mean-squared error is given by

$$\begin{aligned} J_{\text{ex}}(m) &= E\{\underline{\mathbf{e}}^H(m-1)(\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m) \\ &\quad \cdot \mathbf{G}_{2L \times 2L}^{01}\mathbf{X}(m)\mathbf{G}_{2LP \times LP}^{10}\underline{\mathbf{e}}(m-1)\} \end{aligned} \quad (69)$$

and σ_n^2 is the variance of the noise signal $n(k)$. Furthermore, (69) can be written as

$$\begin{aligned} J_{\text{ex}}(m) &= E\{\text{tr}[\underline{\mathbf{e}}^H(m-1)(\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m) \\ &\quad \cdot \mathbf{G}_{2L \times 2L}^{01}\mathbf{X}(m)\mathbf{G}_{2LP \times LP}^{10}\underline{\mathbf{e}}(m-1)]\} \\ &= E\{\text{tr}[(\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m)\mathbf{G}_{2L \times 2L}^{01} \\ &\quad \cdot \mathbf{X}(m)\mathbf{G}_{2LP \times LP}^{10}\underline{\mathbf{e}}(m-1)\underline{\mathbf{e}}^H(m-1)]\} \\ &= \text{tr}\{E\{(\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(m)\mathbf{G}_{2L \times 2L}^{01} \\ &\quad \cdot \mathbf{X}(m)\mathbf{G}_{2LP \times LP}^{10}\underline{\mathbf{e}}(m-1)\underline{\mathbf{e}}^H(m-1)\}\}. \end{aligned}$$

Invoking the independence assumption and using (55), we may reduce this expectation to

$$J_{\text{ex}}(m) = \text{tr}[\mathbf{S}_e \mathbf{M}(m-1)], \quad (70)$$

where

$$\mathbf{M}(m) = E\{\underline{\mathbf{e}}(m)\underline{\mathbf{e}}^H(m)\} \quad (71)$$

is the misalignment correlation matrix.

We derive an expression for the misalignment vector $\underline{\mathbf{e}}(m)$ using the normal equation (34), and (36):

$$\begin{aligned} \underline{\mathbf{e}}(m) &= \mathbf{h} - \hat{\mathbf{h}}(m) \\ &= \mathbf{h} - \mathbf{S}^{-1}(m)\mathbf{s}(m) \\ &= \mathbf{h} - (1-\lambda)\mathbf{S}^{-1}(m) \sum_{i=0}^m \lambda^{m-i} \\ &\quad \cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i)(\mathbf{G}_{L \times 2L}^{01})^H \mathbf{y}(i). \end{aligned} \quad (72)$$

Using $\underline{\mathbf{y}}(m)$ from the model (60), we obtain with (61) and (35):

$$\underline{\boldsymbol{\varepsilon}}(m) = -(1-\lambda)\mathbf{S}^{-1}(m) \sum_{i=0}^m \lambda^{m-i} \cdot (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) (\mathbf{G}_{2L \times 2L}^{01})^H \underline{\mathbf{n}}(i). \quad (73)$$

If we plug this equation into (71), we obtain, after taking the expectations, and noting that for a given input sequence, the only random variable is the white measurement noise $\underline{\mathbf{n}}(m)$:

$$\mathbf{M}(m) = \sigma_n^2 (1-\lambda)^2 \mathbf{S}^{-1}(m) \cdot \left[\sum_{i=0}^m \lambda^{2(m-i)} (\mathbf{G}_{2LP \times LP}^{10})^H \cdot \mathbf{X}^H(i) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(i) \mathbf{G}_{2LP \times LP}^{10} \right] \cdot \mathbf{S}^{-1}(m), \quad (74)$$

where $E\{\underline{\mathbf{n}}(m)\underline{\mathbf{n}}^H(m)\} = \sigma_n^2 \mathbf{I}$ was used. Analogously to (57), we find for the term in brackets in (74):

$$\sum_{i=0}^m \lambda^{2(m-i)} (\mathbf{G}_{2LP \times LP}^{10})^H \mathbf{X}^H(i) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(i) \mathbf{G}_{2LP \times LP}^{10} \approx (1-\lambda^{2(m+1)}) \mathbf{S}_e. \quad (75)$$

Assuming strict equality in (75), using (57), and $1-\lambda^{2(m+1)} = (1-\lambda^{m+1})(1+\lambda^{m+1})$, this leads to

$$\mathbf{M}(m) = \sigma_n^2 (1-\lambda)^2 \frac{1+\lambda^{m+1}}{1-\lambda^{m+1}} \mathbf{S}_e^{-1}. \quad (76)$$

Finally, we obtain for (68) with (70)

$$J'_f(m) = \left[P(1-\lambda)^2 \frac{1+\lambda^m}{1-\lambda^m} + 1 \right] \sigma_n^2. \quad (77)$$

This equation describes the convergence curve of the mean-squared error. One can see that in the steady state, i.e., for large m , the mean-squared error converges to a constant value as desired in (65):

$$J'_f(m \rightarrow \infty) = J'_{f,\min} = [P(1-\lambda)^2 + 1] \sigma_n^2. \quad (78)$$

Moreover, we see from (77) that the convergence behavior of the mean-squared error is independent of the eigenvalues of the ensemble-averaged matrix \mathbf{S}_e . The scalar

$$J_{\text{mis}}(m) = E\{\underline{\boldsymbol{\varepsilon}}^H(m) \underline{\boldsymbol{\varepsilon}}(m)\} \quad (79)$$

describes the convergence of the misalignment, i.e., the coefficient convergence. Using (76), we deduce that

$$\begin{aligned} J_{\text{mis}}(m) &= \text{tr}[\mathbf{M}(m)] \\ &= \sigma_n^2 (1-\lambda)^2 \frac{1+\lambda^{m+1}}{1-\lambda^{m+1}} \text{tr}[\mathbf{S}_e^{-1}] \\ &= \sigma_n^2 (1-\lambda)^2 \frac{1+\lambda^{m+1}}{1-\lambda^{m+1}} \sum_{i=0}^{LP-1} \frac{1}{\lambda_{s,i}}, \end{aligned} \quad (80)$$

where $\lambda_{s,i}$ denotes the eigenvalues of the ensemble-averaged matrix \mathbf{S}_e . It is important to notice the difference between the convergence of the mean-squared error and the misalignment. While the mean-squared error does not depend on the eigenvalues of \mathbf{S}_e (i.e., it is also independent of the channel coherence), the misalignment is magnified by the inverse of the smallest eigenvalue $\lambda_{s,\min}$ of \mathbf{S}_e (and thus of $\mathbf{S}(m)$). The situation is worsened when the variance of the noise σ_n^2 is large. So in practice, at some frequencies, where the signal is poorly excited, we may have a very large misalignment. In order to avoid this problem and to keep the misalignment low, the adaptive algorithm should be regularized by adding small values to the diagonal of $\mathbf{S}(m)$. In Section 6, this important topic is discussed in more detail.

4. Note on generalized frequency-domain adaptive MIMO filtering

In this section, we consider the extension of the algorithm proposed in Section 2 to the general MIMO case, i.e., we have P input signals $x_p(k)$, $p = 1, \dots, P$, and Q desired signals $y_q(k)$, output signals $\hat{y}_q(k)$, and error signals $e_q(k)$, $q = 1, \dots, Q$, respectively (Fig. 3).

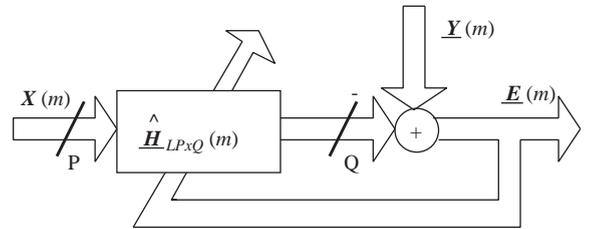


Fig. 3. Adaptive MIMO filtering in the frequency domain.

In the sequel, the following question is discussed: what is the optimum solution based on minimization of error variances?

Let us define signal block vectors $\mathbf{y}_q(m)$, $\mathbf{e}_q(m)$, $\underline{\mathbf{y}}_q(m)$, $\underline{\mathbf{e}}_q(m)$ for each output channel in the same way as shown in (10), (9), (22), and (21), respectively. These quantities can be combined in the $(L \times Q)$ matrices

$$\mathbf{E}(m) = [\mathbf{e}_1(m), \dots, \mathbf{e}_Q(m)],$$

$$\mathbf{Y}(m) = [\mathbf{y}_1(m), \dots, \mathbf{y}_Q(m)],$$

$$\underline{\mathbf{E}}(m) = [\underline{\mathbf{e}}_1(m), \dots, \underline{\mathbf{e}}_Q(m)],$$

$$\underline{\mathbf{Y}}(m) = [\underline{\mathbf{y}}_1(m), \dots, \underline{\mathbf{y}}_Q(m)].$$

We consider two conceivable generalizations of the recursive least-squares error criterion proposed in (29):

Error criterion 1: Separate optimization

The most obvious approach to the problem is to treat each of the Q desired signal channels separately by the algorithm proposed above:

$$J_{f1,q}(m) = (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \mathbf{e}_q^H(i) \mathbf{e}_q(i) \quad (81)$$

for $q = 1, \dots, Q$. This criterion has been traditionally used in all approaches for multichannel echo cancellation, i.e., system identification.

Error criterion 2: Joint optimization

A more general approach foresees to jointly optimize the MIMO filter by the following criterion:

$$\begin{aligned} J_{f2}(m) &= \sum_{q=1}^Q J_{f1,q}(m) \\ &= (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \sum_{q=1}^Q \mathbf{e}_q^H(i) \mathbf{e}_q(i) \\ &= (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \text{tr}[\underline{\mathbf{E}}^H(i) \underline{\mathbf{E}}(i)] \\ &= (1 - \lambda) \sum_{i=0}^m \lambda^{m-i} \|\text{diag}\{\underline{\mathbf{E}}^H(i) \underline{\mathbf{E}}(i)\}\|_1, \end{aligned}$$

where the matrix norm $\|\cdot\|_1$ sums up the absolute values of all matrix elements. Introducing the $(LP \times Q)$ coefficient matrix in the frequency

domain

$$\hat{\mathbf{H}}_{LP \times Q} = \begin{bmatrix} \hat{\mathbf{h}}_{1,1} & \cdots & \hat{\mathbf{h}}_{1,Q} \\ \vdots & \ddots & \vdots \\ \hat{\mathbf{h}}_{P,1} & \cdots & \hat{\mathbf{h}}_{P,Q} \end{bmatrix}, \quad (82)$$

and using the same approach as in Section 2, we obtain the following normal equation:

$$\mathbf{S}(m) \hat{\mathbf{H}}_{LP \times Q} = \mathbf{s}_{LP \times Q}(m). \quad (83)$$

Fortunately, this matrix equation can be easily decomposed into Q equations (34). Therefore, criteria 1 and 2 are strictly equivalent for the behavior of the adaptation. We note, however, that the compact formulation (83) of the normal equation can be used, e.g., to obtain a generalized control of the adaptation for the echo cancellation application [5].

Therefore, for both criteria, the generalized frequency-domain adaptive MIMO filter can be summarized as

$$\mathbf{S}_d(m) = \lambda \mathbf{S}_d(m-1) + (1 - \lambda) \mathbf{X}^H(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m), \quad (84)$$

$$\mathbf{K}(m) = (1 - \lambda) \mathbf{S}_d^{-1}(m) \mathbf{X}^H(m), \quad (85)$$

$$\underline{\mathbf{E}}_{2L \times Q}(m) = \underline{\mathbf{Y}}_{2L \times Q}(m) - \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \hat{\mathbf{H}}_{2LP \times Q}(m-1), \quad (86)$$

$$\begin{aligned} \hat{\mathbf{H}}_{2LP \times Q}(m) &= \hat{\mathbf{H}}_{2LP \times Q}(m-1) \\ &+ \mathbf{G}_{2LP \times 2LP}^{10} \mathbf{K}(m) \underline{\mathbf{E}}_{2L \times Q}(m), \end{aligned} \quad (87)$$

in analogy to Eqs. (50)–(53).

Note that for block size $N < L$, an algorithm is obtained in the same way as mentioned in Section 2.

5. Approximation and special cases

We start this section by giving a very useful approximation of the algorithm proposed in the preceding Section. This approximation can be studied independently from the input signals. It allows us both, to show explicitly the links to the classical single-channel algorithms, and also to

derive new and very efficient multichannel algorithms where the cross-correlations are taken into account. The list of special cases of the framework is not exhaustive and several other algorithms may also be derived.

5.1. Approximation of the frequency-domain Kalman gain

Frequency-domain adaptive filters were first introduced to reduce the arithmetic complexity of the (single-channel) LMS algorithm [13]. Unfortunately, the matrix \mathbf{S}_d is generally not diagonal, so its inversion in (85) has a high complexity and the algorithm may not be very useful in practice. Since \mathbf{S}_d is composed of P^2 sub-matrices

$$\mathbf{S}_{ij} = \lambda \mathbf{S}_{ij}(m-1) + (1-\lambda) \mathbf{X}_i^*(m) \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}_j(m), \tag{88}$$

it is desirable that each of those sub-matrices be a diagonal matrix. In the next paragraph, we will argue that $\mathbf{G}_{2L \times 2L}^{01}$ can be well approximated by the identity matrix with weight 1/2; accordingly, after introducing the positive factor $\mu \leq 2$ in (87), and the matrix $\mathbf{S}'(m)$ approximating $2\mathbf{S}_d(m)$, we then obtain the following approximate algorithm:

$$\mathbf{S}'(m) = \lambda \mathbf{S}'(m-1) + (1-\lambda) \mathbf{X}^H(m) \mathbf{X}(m), \tag{89}$$

$$\mathbf{K}(m) = (1-\lambda) \mathbf{S}'^{-1}(m) \mathbf{X}^H(m), \tag{90}$$

$$\underline{\mathbf{E}}_{2L \times Q}(m) = \underline{\mathbf{Y}}_{2L \times Q}(m) - \mathbf{G}_{2L \times 2L}^{01} \mathbf{X}(m) \hat{\mathbf{H}}_{2LP \times Q}(m-1), \tag{91}$$

$$\hat{\mathbf{H}}_{2LP \times Q}(m) = \hat{\mathbf{H}}_{2LP \times Q}(m-1) + \mu \mathbf{G}_{2LP \times 2LP}^{10} \mathbf{K}(m) \underline{\mathbf{E}}_{2L \times Q}(m), \tag{92}$$

where each sub-matrix of \mathbf{S}' and \mathbf{K} is now a diagonal matrix and $\mu \leq 2$ is a positive number. Note that the imprecision introduced by the approximation in (89) and thus in the Kalman gain (90) will only affect the convergence rate. Obviously, we cannot permit the same kind of approximation in (91), because that would result in approximating a linear convolution by a

circular one, which of course can have a disastrous impact in the adaptive filter behavior.

To justify the above approximation, let us examine the structure of the matrix $\mathbf{G}_{2L \times 2L}^{01}$. We have

$$(\mathbf{G}_{2L \times 2L}^{01})^* = \mathbf{F}_{2L \times 2L}^{-1} \mathbf{W}_{2L \times 2L}^{01} \mathbf{F}_{2L \times 2L}. \tag{93}$$

Since $\mathbf{W}_{2L \times 2L}^{01}$ is a diagonal matrix, $(\mathbf{G}_{2L \times 2L}^{01})^*$ is a circulant matrix. Therefore, inverse transformation of the diagonal of $\mathbf{W}_{2L \times 2L}^{01}$ gives the first column of $(\mathbf{G}_{2L \times 2L}^{01})^*$,

$$\begin{aligned} \mathbf{g}^* &= [g_0^*, g_1^*, \dots, g_{2L-1}^*]^T \\ &= \mathbf{F}_{2L \times 2L}^{-1} [0, \dots, 0, 1, \dots, 1]^T. \end{aligned}$$

The elements of vector \mathbf{g} can be written explicitly as

$$\begin{aligned} g_k &= \frac{1}{2L} \sum_{l=L}^{2L-1} \exp(-j2\pi kl/2L) \\ &= \frac{(-1)^k}{2L} \sum_{l=0}^{L-1} \exp(-j\pi kl/L), \end{aligned} \tag{94}$$

where $j^2 = -1$. Since g_k is the sum of a finite geometric series, we have

$$\begin{aligned} g_k &= \begin{cases} 0.5 & k=0 \\ \frac{(-1)^k}{2L} \frac{1 - \exp(-j\pi k)}{1 - \exp(-j\pi k/L)} & k \neq 0 \end{cases} \\ &= \begin{cases} 0.5 & k=0 \\ 0 & k \text{ even} \\ -\frac{1}{2L} \left[1 - j \cot\left(\frac{\pi k}{2L}\right) \right] & k \text{ odd,} \end{cases} \end{aligned} \tag{95}$$

where $L-1$ elements of vector \mathbf{g} are equal to zero. Moreover, since $(\mathbf{G}_{2L \times 2L}^{01})^H \mathbf{G}_{2L \times 2L}^{01} = \mathbf{G}_{2L \times 2L}^{01}$, then $\mathbf{g}^H \mathbf{g} = g_0 = 0.5$ and we have

$$\mathbf{g}^H \mathbf{g} - g_0^2 = \sum_{l=1}^{2L-1} |g_l|^2 = 2 \sum_{l=1}^{L-1} |g_l|^2 = \frac{1}{4}. \tag{96}$$

We can see from (96) that the first element of vector \mathbf{g} , i.e., g_0 , is dominant in a mean-square sense, and from (95) that the absolute values of the L first elements of \mathbf{g} decrease rapidly to zero as k increases. Because of the conjugate symmetry, i.e., $|g_k| = |g_{2L-k}|$ for $k = 1, \dots, L-1$, the last few elements of \mathbf{g} are not negligible, but this affects

only the first and last columns of $\mathbf{G}_{2L \times 2L}^{01}$ since this matrix is circulant with \mathbf{g} as its first column. All other columns have those non-negligible elements wrapped around in such a way that they are concentrated around the main diagonal. To summarize, we can say that for L large, only the very first (few) off-diagonals of $\mathbf{G}_{2L \times 2L}^{01}$ will be non-negligible while the others can be completely neglected. We also neglect the influence of the two isolated peaks $|g_{2L-1}| = |g_1| < g_0$ on the lower left corner and the upper right corner, respectively. Thus, approximating $\mathbf{G}_{2L \times 2L}^{01}$ by a diagonal matrix, i.e., $\mathbf{G}_{2L \times 2L}^{01} \approx g_0 \mathbf{I} = \mathbf{I}/2$, is reasonable, and in this case we will have $\mu \approx 1/g_0 = 2$ for an optimum convergence rate. For the rest of this paper, we suppose that $0 < \mu \leq 2$.

5.2. Special cases

In the single-channel case $P = Q = 1$, \mathbf{S}' and \mathbf{K} are diagonal matrices and the classical constrained FLMS [13] follows immediately from (89) to (92). This algorithm requires the computation of 5 FFTs of length $2L$ per block. By approximating $\mathbf{G}_{2LP \times 2LP}^{10}$ in (92) to the identity matrix, we obtain the unconstrained FLMS (UFLMS) algorithm [24] which requires only 3 FFTs per block. Many simulations show that the two algorithms have virtually the same performance.

For $N < L$, $\mathbf{S}_d(m)$ in (84) consists of $(K \cdot P)^2$ sub-matrices that can be approximated as shown above. It is interesting that for $N = 1$, the algorithm is strictly equivalent to the RLS algorithm in the time domain. After the approximation, we obtain a new algorithm that we call *extended multidelay filter* (EMDF) for $1 < N < L$ that takes the auto-correlations between the blocks into account [9]. Finally, the classical multidelay filter is obtained by further approximating $\mathbf{S}'(m)$ in (89) by dropping the off-diagonal components in $\mathbf{S}'(m)$:

$$\mathbf{S}''(m) = \text{diag}\{\mathbf{S}_{1,1,0}(m), \dots, \mathbf{S}_{1,1,K-1}(m)\}, \quad (97)$$

where

$$\mathbf{S}_{1,1,k}(m) = \lambda \mathbf{S}_{1,1,k}(m-1) + (1-\lambda) \mathbf{X}_{1,k}^*(m) \mathbf{X}_{1,k}(m)$$

are $(2N \times 2N)$ diagonal matrices.

In the MIMO case, (90) is the solution of a $P \times P$ system of linear equations of block matrices:

$$\mathbf{K}(m) = [\mathbf{K}_1^T(m), \dots, \mathbf{K}_P^T(m)]^T. \quad (98)$$

This allows us to *formally* write the update equation (92) as PQ tightly coupled ‘single-channel’ update equations

$$\hat{\mathbf{h}}_{p,q}(m) = \hat{\mathbf{h}}_{p,q}(m-1) + \mu \mathbf{G}_{2L \times 2L}^{10} \mathbf{K}_p \mathbf{e}_q(m) \quad (99)$$

($p = 1, \dots, P$, $q = 1, \dots, Q$) with the sub-matrices $\mathbf{K}_p(m)$ taking the cross-correlations between the input channels into account. These update equations (99) can then be calculated element-wise and the (cross) power spectra are estimated recursively:

$$\mathbf{S}_{i,j}(m) = \lambda \mathbf{S}_{i,j}(m-1) + (1-\lambda) \mathbf{X}_i^*(m) \mathbf{X}_j(m), \quad (100)$$

where $\mathbf{S}_{j,i}(\cdot) = \mathbf{S}_{i,j}^*(\cdot)$.

It is important to note that the calculation of the Kalman gain (Eqs. (84) and (85)), which is the computationally most demanding part, is completely independent of the number Q of output channels and thus, has to be calculated only once, while the remaining update equations (99) formally correspond to single-channel (U)FLMS algorithms.

In the case of two input channels $P = 2$, the Kalman gain can be written in an explicit form by block-inversion:

$$\mathbf{K}_1 = \mathbf{D}(m) \mathbf{S}_{1,1}^{-1}(m) [\mathbf{X}_1^*(m) - \mathbf{S}_{1,2}(m) \mathbf{S}_{2,2}^{-1}(m) \mathbf{X}_2^*(m)], \quad (101)$$

$$\mathbf{K}_2 = \mathbf{D}(m) \mathbf{S}_{2,2}^{-1}(m) [\mathbf{X}_2^*(m) - \mathbf{S}_{2,1}(m) \mathbf{S}_{1,1}^{-1}(m) \mathbf{X}_1^*(m)], \quad (102)$$

with the abbreviation

$$\mathbf{D}(m) = (1-\lambda) [\mathbf{I}_{2L \times 2L} - \mathbf{S}_{1,2}^*(m) \mathbf{S}_{1,2}(m) \cdot \{\mathbf{S}_{1,1}(m) \mathbf{S}_{2,2}(m)\}^{-1}]^{-1}.$$

The solutions of (90) for more than two input channels may be formulated similarly to the corresponding part of the stereo update Eqs. (101) and (102) (e.g., using Cramer’s rule). These representations allow an intuitive interpretation as a correction of the interchannel-correlations in \mathbf{K}_i between \mathbf{X}_i^* and the other input signals \mathbf{X}_j^* , $j \neq i$.

For three channels, we have (omitting, for simplicity, the block time index m of all matrices)

$$\mathbf{K}_1 = (1 - \lambda)\mathbf{D}^{-1}[\mathbf{X}_1^*(\mathbf{S}_{2,2}\mathbf{S}_{3,3} - \mathbf{S}_{3,2}\mathbf{S}_{2,3}) - \mathbf{X}_2^*(\mathbf{S}_{1,2}\mathbf{S}_{3,3} - \mathbf{S}_{1,3}\mathbf{S}_{3,1}) - \mathbf{X}_3^*(\mathbf{S}_{1,3}\mathbf{S}_{2,2} - \mathbf{S}_{1,2}\mathbf{S}_{2,3})],$$

$$\mathbf{D} := \mathbf{S}_{1,1}(\mathbf{S}_{2,2}\mathbf{S}_{3,3} - \mathbf{S}_{3,2}\mathbf{S}_{2,3}) - \mathbf{S}_{2,1}(\mathbf{S}_{1,2}\mathbf{S}_{3,3} - \mathbf{S}_{1,3}\mathbf{S}_{3,1}) - \mathbf{S}_{3,1}(\mathbf{S}_{1,3}\mathbf{S}_{2,2} - \mathbf{S}_{1,2}\mathbf{S}_{2,3})$$

as the first of the three Kalman gain components with the common factor \mathbf{D} .

Unfortunately, for a higher number of channels, the number of update terms increases rapidly, and the equations become too complicated for practical use. Therefore, a more efficient scheme for these cases will be proposed in Section 7.

6. A dynamical regularization strategy

In most practical scenarios, the desired signal $y(k)$ is disturbed, e.g., by some acoustic background noise. As shown above (c.f. (80)), the parameter estimation (i.e., misalignment) is very sensitive in poorly excited frequency bins. For robust adaptation the power spectral densities $\mathbf{S}_{i,i}$ are replaced by regularized versions according to $\tilde{\mathbf{S}}_{i,i} = \mathbf{S}_{i,i} + \text{diag}\{\delta_i\}$ prior to inversion in (85). The basic feature of the regularization is a compromise between fidelity to data and fidelity to some prior information about the solution [26]. The latter increases the robustness, but leads to biased solutions. Therefore, we propose here a *bin-selective dynamical regularization vector*

$$\delta_i(m) = \delta_{\max} \cdot [e^{-S_{i,i}^{(0)}(m)/S_0}, \dots, e^{-S_{i,i}^{(2L-1)}(m)/S_0}]^T \tag{103}$$

with two scalar parameters δ_{\max} and S_0 . $S_{i,i}^{(v)}$ denotes the v th frequency component ($v = 0, \dots, 2L - 1$) on the main diagonal of $\mathbf{S}_{i,i}$. Note that for efficient implementation, e in (103) may be replaced by a basis 2 and modified S_0 . δ_{\max} should be chosen according to the (estimated) disturbing noise level in the desired signal $y(k)$.

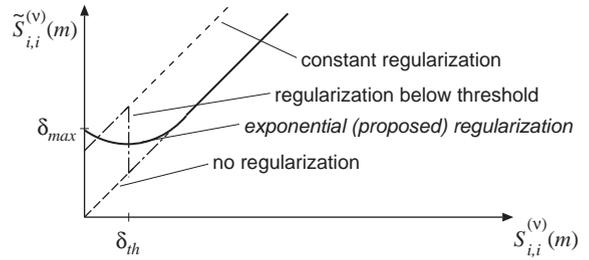


Fig. 4. Different regularization methods (channel i , bin v).

As shown in Fig. 4, this exponential method provides a smooth transition between regularization for low input power and data fidelity whenever the input power is high enough, and yields improved results compared to fixed regularization and to the popular approach of choosing the maximum out of the respective component $\mathbf{S}_{i,i}^{(v)}$ and a fixed threshold δ_{th} . Results of numerical simulations can be found in Section 8. The method also copes well with unbalanced excitation of the input channels, and most importantly, it can be easily extended for the efficient Kalman gain calculation introduced in the next section.

7. Efficient multichannel realization

As will be demonstrated by simulation results and real-world applications in Section 8, the presented algorithm copes well with multichannel input. The cases of a larger number of filter input channels (P larger than 2 or 3) or similarly, a larger number of sub-filters ($N < L$) when using the EMDF algorithm call for further improvement of the computational efficiency. In this section, we propose efficient and stable recursive calculation schemes for the frequency-domain Kalman gain for and the DFTs of the overlapping input data blocks for the case of a large number of filter input channels. Overlapping input data blocks result from an overlap factor $\alpha > 1$, originally proposed in [27]. Incorporating this extension in the proposed algorithm is very simple. Essentially, only the way the input data matrices (17) are

calculated, is modified to

$$\mathbf{X}_p(m) = \text{diag} \left\{ \mathbf{F}_{2L \times 2L} \begin{bmatrix} x_p(m\frac{L}{\alpha} - L) \\ \vdots \\ x_p(m\frac{L}{\alpha} + L - 1) \end{bmatrix} \right\}. \quad (104)$$

Simulations show that increased overlap factors α are particularly useful in the multichannel case.

7.1. Efficient calculation of the frequency-domain Kalman gain

For a *practical implementation* of a system with $P > 2$ channels, we propose computationally more efficient methods to calculate (90) as follows.

Due to the block diagonal structure of (90), it can be simply decomposed w.r.t. the DFT components $v = 0, \dots, 2L - 1$ into $2L$ equations

$$\mathbf{K}^{(v)}(m) = (1 - \lambda)(\mathbf{S}^{(v)}(m))^{-1}(\mathbf{X}^{(v)}(m))^H \quad (105)$$

with (usually small) $P \times P$ unitary and positive definite matrices $\mathbf{S}^{(v)}$ containing the v th components on the block diagonals of $\mathbf{S}'^{-1}(m)$. Both $\mathbf{K}^{(v)}$ and $\mathbf{X}^{(v)}$ are *vectors* of length P . Note that for real input signals x_i we need to solve (105) only for $L + 1$ bins.

A well-known and numerically stable method for this type of problems is the Cholesky decomposition of $\mathbf{S}^{(v)}$ followed by solution via back-substitution, see [17]. The resulting total complexity for one output value is then

$$O(P \cdot \log_2(2L)) + O(P^3), \quad (106)$$

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

For a large number (≥ 3) of input channels (see, e.g., the applications in Section 8) we introduce a recursive solution of (105) that jointly estimates the *inverse* power spectra $(\mathbf{S}^{(v)})^{-1}$ in (89) using the matrix-inversion lemma, e.g., [19]. This lemma relates a matrix

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^H \quad (107)$$

to its inverse according to

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^H\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^H\mathbf{B}, \quad (108)$$

as long as \mathbf{A} and \mathbf{B} are positive definite. Comparing (89) to (107), we immediately obtain from (89) an update equation for the *inverse* matrices

$$(\mathbf{S}^{(v)}(m))^{-1} = \lambda^{-1} \left[(\mathbf{S}^{(v)}(m-1))^{-1} - \frac{(\mathbf{S}^{(v)}(m-1))^{-1}(\mathbf{X}^{(v)}(m))^H \mathbf{X}^{(v)}(m)(\mathbf{S}^{(v)}(m-1))^{-1}}{\lambda(1-\lambda)^{-1} + \mathbf{X}^{(v)}(m)(\mathbf{S}^{(v)}(m-1))^{-1}(\mathbf{X}^{(v)}(m))^H} \right]$$

using the bin-wise quantities introduced in (105) (making the denominator a scalar value). Introduction of the common vector

$$\mathbf{T}_1^{(v)}(m) = (\mathbf{S}^{(v)}(m-1))^{-1}(\mathbf{X}^{(v)}(m))^H \quad (109)$$

in the numerator and the denominator leads to

$$(\mathbf{S}^{(v)}(m))^{-1} = \lambda^{-1}(\mathbf{S}^{(v)}(m-1))^{-1} - \frac{\mathbf{T}_1^{(v)}(m)(\mathbf{T}_1^{(v)}(m))^H}{\lambda^2(1-\lambda)^{-1} + \lambda \mathbf{X}^{(v)}(m)\mathbf{T}_1^{(v)}(m)}. \quad (110)$$

The Kalman gain (105) can then be efficiently calculated (using (110)) by

$$\mathbf{K}^{(v)}(m) = \frac{1-\lambda}{\lambda} \mathbf{T}_1^{(v)}(m) \cdot \left[1 - \frac{(\mathbf{T}_1^{(v)}(m))^H (\mathbf{X}^{(v)}(m))^H}{\lambda(1-\lambda)^{-1} + \mathbf{X}^{(v)}(m)\mathbf{T}_1^{(v)}(m)} \right]. \quad (111)$$

Again, there are common terms

$$\mathbf{T}_2^{(v)}(m) = \mathbf{X}^{(v)}(m)\mathbf{T}_1^{(v)}(m) \quad (112)$$

in (111) and (110).

Note that our approach should not be confused with the classical RLS approach [1] which also makes use of the matrix-inversion lemma. As we apply the lemma independently to small $P \times P$ systems (105), it is numerically much less critical than in the RLS algorithm. Note that for $N = L$, there is no analogon to a more efficient *fast* RLS [11] due to the different matrix structures (vector $\mathbf{X}^{(v)}(m)$ does not reflect a tapped delay line).

The complexity of the different computation methods for the Kalman gains (in MUL/ADDs for one output value $e(k)$) are compared in Fig. 5 for the case $N = L$. Note that our approach is

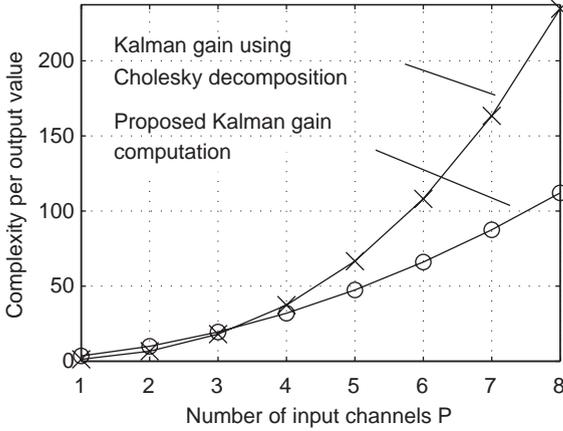


Fig. 5. Complexity (MUL/ADDs) of Kalman gain for $N = L$.

particularly efficient for the extended multidelay filter (block size $N < L$) introduced in Section 5.

7.2. Dynamical regularization for proposed Kalman gain approach

Due to recursion (110) the regularization according to (103) is not immediately applicable. Therefore, an equivalent modification is applied directly to the data matrices $\mathbf{X}^{(v)}(m)$ by addition of mutually uncorrelated white noise sequences to each channel and frequency bin, respectively. Using the modified signal vectors, denoted by

$$\tilde{\mathbf{X}}^{(v)}(m) = \mathbf{X}^{(v)}(m) + \mathbf{N}^{(v)}(m), \quad (113)$$

where $\mathbf{N}^{(v)}(m)$ are the vectors of the white noise signals, we obtain the modified power spectral density matrices (c.f. Eq. (89))

$$\begin{aligned} \tilde{\mathbf{S}}^{(v)}(m) \approx & (1 - \lambda) \sum_{q=0}^m \lambda^{m-q} \mathbf{X}^{(v)H}(q) \mathbf{X}^{(v)}(q) \\ & + (1 - \lambda) \sum_{q=0}^m \lambda^{m-q} \\ & \cdot \text{diag}\{[|N_1^{(v)}(q)|^2, \dots, |N_P^{(v)}(q)|^2]^T\}. \end{aligned} \quad (114)$$

The diagonal elements of the second term can be interpreted as a bin-selective dynamical regularization vector $\delta^{(v)}(m)$ with elements (for channel i

and bin v)

$$\begin{aligned} \delta_i^{(v)}(m) &= (1 - \lambda) \sum_{q=0}^m \lambda^{m-q} |N_i^{(v)}(q)|^2 \\ &= \lambda \delta_i^{(v)}(m-1) + (1 - \lambda) |N_i^{(v)}(m)|^2. \end{aligned} \quad (115)$$

Thus, in order to update the regularization from $\delta_i^{(v)}(m-1)$ to $\delta_i^{(v)}(m)$ with the appropriate speed (determined by λ), we need to add noise with power

$$|N_i^{(v)}(m)|^2 = \frac{\delta_i^{(v)}(m) - \lambda \delta_i^{(v)}(m-1)}{1 - \lambda}. \quad (116)$$

On the other hand, according to (103), the regularization should be chosen according to

$$\begin{aligned} \delta_i^{(v)}(m) &= \delta_{\max} \exp\left(-\frac{S_{i,i}^{(v)}(m)}{S_0}\right) \\ &= \delta_{\max} \\ &\cdot \exp\left(-\frac{\lambda S_{i,i}^{(v)}(m-1) + (1 - \lambda) |X_i^{(v)}(m)|^2}{S_0}\right). \end{aligned} \quad (117)$$

Now, unlike other dynamical regularization methods, the exponential regularization allows simple elimination of the elements $S_{i,i}^{(v)}(m-1)$ of the non-inverted matrix (which need therefore not be computed at all due to the matrix-inversion lemma (108)), since

$$\begin{aligned} \delta_i^{(v)}(m) &= \delta_{\max} \left[\exp\left(-\frac{S_{i,i}^{(v)}(m-1)}{S_0}\right) \right]^\lambda \\ &\cdot \exp\left(-\frac{(1 - \lambda) |X_i^{(v)}(m)|^2}{S_0}\right) \\ &= \delta_{\max}^{1-\lambda} (\delta_i^{(v)}(m-1))^\lambda \\ &\cdot \exp\left(-\frac{(1 - \lambda) |X_i^{(v)}(m)|^2}{S_0}\right). \end{aligned} \quad (118)$$

7.3. Efficient DFT calculation of overlapping data blocks

In this section we address the first term of the computational cost given in (106) which is mainly determined by the DFTs of the frequency-domain adaptive filtering scheme (Fig. 2). The $2L$ -point

DFT calculation in (104) has to be carried out for each of the P loudspeaker signals and is therefore most costly. Moreover, as will be discussed in Section 8, 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 (104) recursively as well.

The derivation

$$x_i^{(k)}(m) = x_i\left(m\frac{L}{\alpha} - L + k\right) \quad (119)$$

denotes the k th component ($k = 0, \dots, 2L - 1$) of the time domain vector (block index m) to be transformed in (104). Let us now consider the v th element on the diagonal of $\mathbf{X}_i(m)$ where $w = e^{-j2\pi/2L}$:

$$X_i^{(v)}(m) = \sum_{k=0}^{2L-1} x_i^{(k)}(m)w^{vk}. \quad (120)$$

Separating the summation into one for previous and one for new input values (Fig. 6), followed by the introduction of the previous vector elements $x_i^{(k)}(m - 1)$ leads to

$$\begin{aligned} X_i^{(v)}(m) &= \sum_{k=0}^{2L-L/\alpha-1} x_i^{(k)}(m)w^{vk} + \sum_{k=2L-L/\alpha}^{2L-1} x_i^{(k)}(m)w^{vk} \\ &= \sum_{k=L/\alpha}^{2L-1} x_i^{(k)}(m-1)w^{v(k-L/\alpha)} \\ &\quad + \Delta X_i^{(v)}(m), \end{aligned} \quad (121)$$

where

$$\Delta X_i^{(v)}(m) = \sum_{k=2L-L/\alpha}^{2L-1} x_i^{(k)}(m)w^{vk} \quad (122)$$

contains the new input values and will be the update term in our recursive scheme. Next, we

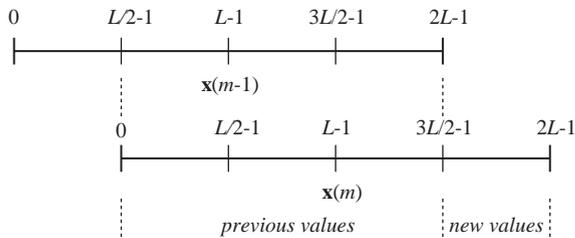


Fig. 6. Example: overlapping data blocks, $\alpha = 2$.

introduce the previous DFT output values $X_i^{(v)}(m - 1)$ by subtracting the vector elements of $\mathbf{x}_i^{(k)}(m - 1)$ of the previous data vector shifted out of the DFT length $2L$:

$$\begin{aligned} X_i^{(v)}(m) &= w^{-vL/\alpha} \left[\sum_{k=0}^{2L-1} x_i^{(k)}(m-1)w^{vk} \right. \\ &\quad \left. - \sum_{k=0}^{L/\alpha-1} x_i^{(k)}(m-1)w^{vk} \right] + \Delta X_i^{(v)}(m) \\ &= w^{-vL/\alpha} X_i^{(v)}(m-1) \\ &\quad - w^{-vL/\alpha} \sum_{k=2L-L/\alpha}^{2L-1} x_i^{(k-2L+L/\alpha)}(m) \\ &\quad \cdot w^{v(k-2L+L/\alpha)} + \Delta X_i^{(v)}(m). \end{aligned} \quad (123)$$

Using (119), we can show that

$$x_i^{(k-2L+L/\alpha)}(m) = x_i^{(k)}(m - 2\alpha + 1). \quad (124)$$

Finally, we obtain

$$\begin{aligned} X_i^{(v)}(m) &= w^{-vL/\alpha} X_i^{(v)}(m-1) \\ &\quad - w^{-v2L} \Delta X_i^{(v)}(m-2\alpha+1) \\ &\quad + \Delta X_i^{(v)}(m). \end{aligned} \quad (125)$$

Again, this recursive update needs to be carried out only for the bins $v = 0, \dots, L$ if $x_i^{(k)}(m)$ is real-valued. Only the update $\Delta X_i^{(v)}(m)$ in this equation has to be calculated explicitly using the L/α new values of the input vector.

With the truncation of the time-domain input vector for calculating $\Delta X_i^{(v)}(m)$ in mind, we consider now the decimation-in-frequency FFT algorithm. Fig. 7 shows a simple example for $2L = 8$ and $\alpha = 2$. $2L - L/\alpha$ 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. Generally, the elimination of operations on zeros in the FFT is referred to as pruning and was first described by Markel [25]. Since then, several pruning algorithms with increased efficiency have been proposed. A summary and further references of different approaches may be found, e.g., in [20].

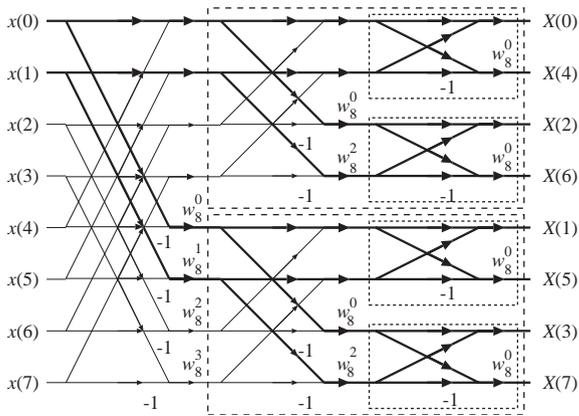


Fig. 7. Illustration of decimation-in-frequency FFT with windowed input.

In summary, using FFT pruning, the recursive DFT approach reduces the first term of the complexity in (106) to $O(P \cdot \log_2(L/\alpha))$ for each output point.

8. Simulations and real-world applications

As mentioned in the introduction, there are many areas of applications for multichannel adaptive filtering. In the following, we demonstrate the performance of our approach in a few examples for hands-free speech communication.

8.1. Multichannel acoustic echo cancellation

For applications such as home entertainment, virtual reality (e.g., games, 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, multichannel acoustic echo cancellation is a key technology whenever hands-free and full-duplex communication is desired (Fig. 8).

The fundamental problem is that the multiple channels may carry linearly related signals which in turn may make the normal equation to be solved by the adaptive algorithm singular. This implies that there is no unique solution to the equation but an infinite number of solutions and it

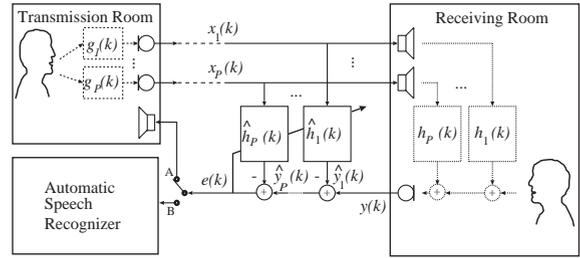


Fig. 8. Multichannel acoustic echo cancellation.

can be shown that all but the true one depend on the impulse responses of the transmission room [29,30]. It is shown in [7] that the only solution to the nonuniqueness problem is to reduce the correlation between the different signals. Three methods of preprocessing can be distinguished: inaudible nonlinear processing, e.g., [7], additive noise (preferably below the masking threshold of human hearing), e.g., [16], and time-varying filtering, e.g., [32]. For the following example, a signal from a common source (in the transmission room) was convolved by P different room impulse responses and nonlinearly, but inaudibly preprocessed according to [7] (P different nonlinearities with factor 0.5). In this subsection we consider only $Q = 1$ microphone in the receiving room. The convergence behavior is shown both in terms of system misalignment (ratio of the squared norms of (62) and the desired response), and in terms of echo return loss enhancement (ERLE) which describes the ratio of the short-term powers of the echo $y(k) - n(k)$ and the residual echo $e(k) - n(k)$. For smoothing the ERLE curves, a moving average filter of length 256 was used.

Fig. 9 illustrates the effect of taking the cross-correlations in (101) and (102) for $P = 2$ into account. As input $x_p(k)$, a common white noise signal was convolved by the room impulse responses in the transmission room. Another white noise signal was added to the echo on the microphone for $SNR = 35$ dB. Here, both the receiving room impulse responses and the modeling filter lengths were chosen to be 1024 (solid lines: proposed, dashed lines: classical UFLMS algorithm).

For simulations with real-world signals, the lengths of the measured receiving room impulse

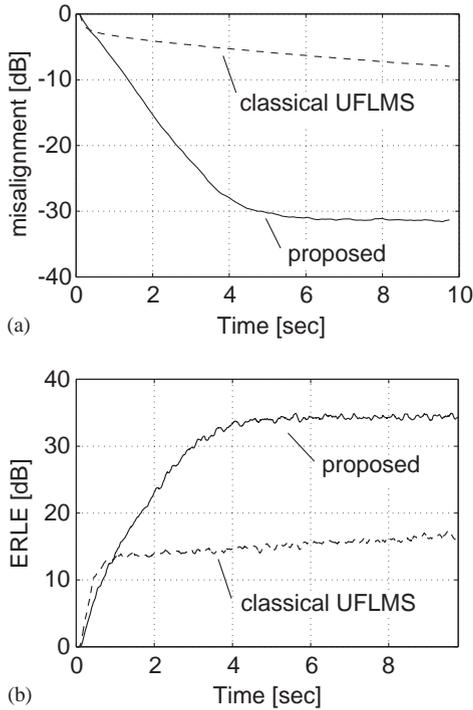


Fig. 9. Effect of taking cross-correlation into account ($P = 2$ channels, $\alpha = 4$). (a) Misalignment, (b) ERLE.

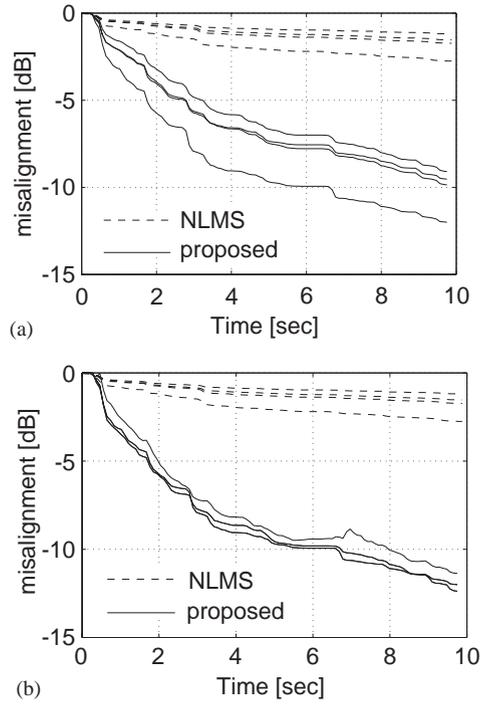


Fig. 10. Misalignment convergence for the multichannel cases $P = 2, 3, 4, 5$. (a) Overlap $\alpha = 4$, (b) overlap α adjusted.

responses were 4096 and the modelling filters were 1024, respectively. One common speech signal from the transmission room serves as input signal. Fig. 10 shows the misalignment convergence of the described algorithm (solid) for the multichannel cases $P = 2$ (lowest curve), 3, 4, 5 (uppermost curve), and the basic NLMS [19] (dashed) for comparison. In (a) the overlap factor α was set to 4 in all cases, while in (b) the overlap factor α was set to 4 for $P = 2$, and adjusted to 8 for $P = 3, 4$, and to 16 for $P = 5$. Using these parameters, the convergence curves for the different numbers of channels are almost indistinguishable. Fig. 11 (a) shows the corresponding ERLE curves.

Fig. 11 (b) compares different regularization methods (white noise distortion as above): no regularization (uppermost curve), constant regularization (dotted), threshold (dashed), exponential with original algorithm (dash-dot), proposed Kalman gain according to Eq. (111) (lower solid line).

We note that for both, stereophonic teleconferencing and hands-free speech recognition applications, real-time systems could be successfully implemented on regular personal computers [10,14].

8.2. Adaptive MIMO filtering for hands-free speech communication

In applications such as hands-free speech recognition, it is very important to reduce interfering noise or competing speech signals, and reverberation of the target speech signal, in addition to the acoustic echo cancellation (Fig. 12).

An efficient approach to address these problems is to replace the single microphone by a microphone array directing a beam of increased sensitivity at the active talker [22]. In any practical system, this scenario presents a MIMO system identification problem for the acoustic echo canceller [10,22]. Fortunately, as noted in Section

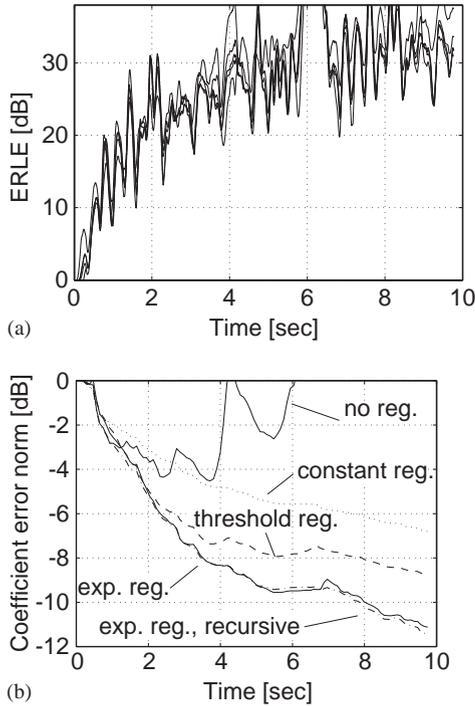


Fig. 11. (a) ERLE convergence for the multichannel cases $P = 2, 3, 4, 5$ and adjusted α and (b) comparison of regularization methods, $P = 5, \alpha = 16$.

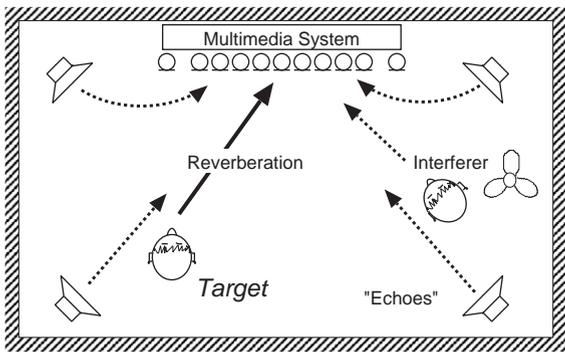


Fig. 12. Hands-free speech recognition in multimedia systems.

4, the costly calculation of the Kalman gain is necessary only once, i.e., it is independent of the number of microphones. Fig. 13 gives an example of a low-complexity structure. Echo cancellation is applied to several beamformer (BF) output signals. The fixed beamformers do not disturb the

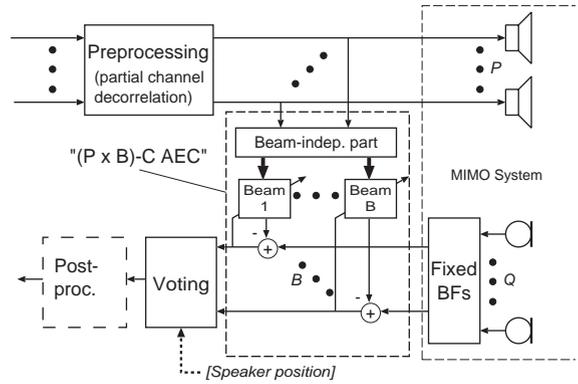


Fig. 13. A human/machine interface for hands-free speech recognition.

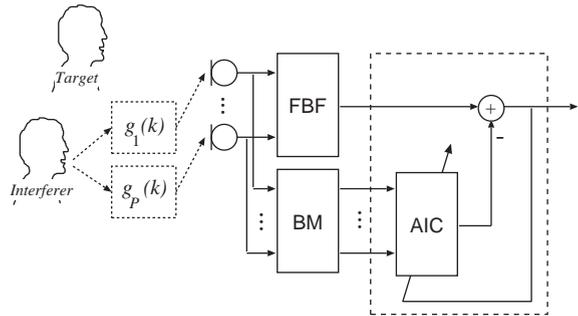


Fig. 14. Generalized sidelobe canceler.

convergence of the echo cancellation and direct beams to all directions of interest [22]. Thanks to the efficient frequency-domain approach, it has become possible to run such a system in real-time on standard PC platforms, e.g., stereo echo cancellation ($L = 4096$) for 5 beams is possible on an Intel-based PC (1 GHz).

8.3. Adaptive beamforming

Next, we show that the proposed generalized multichannel frequency-domain adaptive filter is also an interesting option for adaptive beamforming in hands-free speech applications.

A simple and very effective structure for adaptive beamforming is the generalized sidelobe canceller (GSC) after Griffith and Jim [18] (Fig. 14). The fixed beamformer (FBF) enhances target signal components, and is used as reference

for the adaptation of the adaptive sidelobe cancelling path, which consists of a blocking matrix (BM) and an adaptive interference canceller (AIC). For our considerations, the AIC, a multichannel adaptive filter as shown in Fig. 1, is of particular interest. It is driven by the interferer signals, while the target signal is blocked by the BM.

To ensure robust operation (i.e., to avoid distortion of the target signal), the BM should be adaptive as well [21]. However, for simplicity, we assumed in the simulations this matrix to be fixed, as originally proposed in [18].

Often, if there is a dominant interferer, the underlying normal equation of the AIC is very ill-conditioned as in the case of multichannel acoustic echo cancellation. A low level of background noise usually ensures that there is a unique solution, but the convergence may be slowed down considerably. Fig. 15 shows the Interference Rejection (IR) of a GSC with conventional UFLMS adaptation. For the simulations, $P = 5$ microphone signals were used. The filter lengths were $L = 128$ and the overlap factor was set to $\alpha = 1$. In Fig. 16 (same

parameters), the cross-correlations between the microphone signals were taken into account leading to significant improvement of the interference rejection.

9. Conclusions

In many applications where an adaptive filter is required, frequency-domain algorithms are an attractive alternative to time-domain algorithms, especially for the multichannel case. First, the computational complexity can be low by utilizing the efficiency of the FFT. Second, the convergence is improved if crucial parameters of these algorithms such as the exponential window, regularization, and adaptation step are properly chosen.

In this article a general framework for multichannel frequency-domain adaptive filtering was presented and its efficiency in actual applications was demonstrated. A generic multichannel algorithm with an MMSE convergence that is independent of the input signal statistics can be derived from the normal equation after minimizing a block least-squares criterion in the frequency domain. We analyzed the convergence of this algorithm and discussed some approximations that lead to both, well-known algorithms in the single-channel case, such as the FLMS and UFLMS, and new algorithms such as the EMDF. For the multichannel case the framework is attractive as the cross-correlations between all input signals are efficiently taken into account. We have also presented strategies to improve the computational efficiency further by introducing stable schemes for recursive DFT and Kalman gain computation. Several simulations and real-time implementations illustrate the benefits of the multichannel algorithm.

Acknowledgements

The authors would like to thank W. Herboldt of the telecommunications institute, University of Erlangen-Nuremberg, for fruitful discussions on adaptive beamforming.

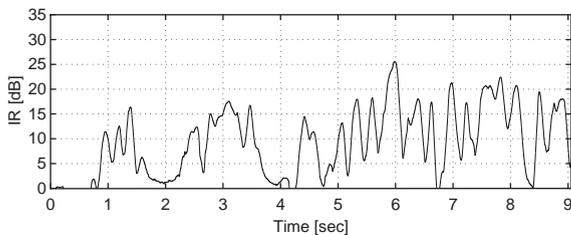


Fig. 15. Interference rejection: generalized sidelobe canceller with classical UFLMS.

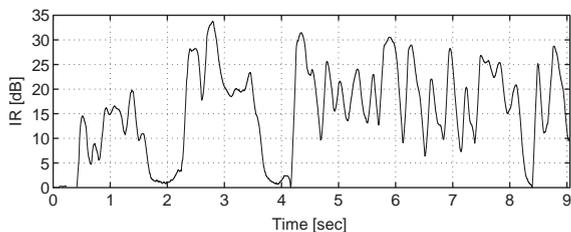


Fig. 16. Interference rejection: generalized sidelobe canceller taking cross-correlations into account.

References

- [1] M.G. Bellanger, Adaptive Digital Filters and Signal Analysis, Marcel Dekker, New York, 1987.
- [2] J. Benesty, F. Amand, A. Gilloire, Y. Grenier, Adaptive filtering algorithms for stereophonic acoustic echo cancellation, Proceedings of International Conference on Acoustics, Speech, and Signal Processing, May 1995, pp. 3099–3102.
- [3] J. Benesty, P. Duhamel, Fast constant modulus adaptive algorithm, IEE Proc. F 138 (August 1991) 379–387.
- [4] J. Benesty, P. Duhamel, A fast exact least mean square adaptive algorithm, IEEE Trans. Signal Process. 40 (12) (December 1992) 2904–2920.
- [5] J. Benesty, T. Gänslér, A multichannel acoustic echo canceller double-talk detector based on a normalized cross-correlation matrix, Proceedings of International Workshop on Acoustic Echo and Noise Control, Darmstadt, Germany, September 2001, pp. 63–66.
- [6] J. Benesty, A. Gilloire, Y. Grenier, A frequency-domain stereophonic acoustic echo canceler exploiting the coherence between the channels, J. Acoust. Soc. Am. 106 (September 1999) L30–L35.
- [7] J. Benesty, D.R. Morgan, M.M. Sondhi, A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation, IEEE Trans. on Speech and Audio Process. 6 (2) (March 1998).
- [8] D.H. Brandwood, A complex gradient operator and its application in adaptive array theory, Proc. IEE, 130, (Pts. F and H) (February 1983) 11–16.
- [9] H. Buchner, J. Benesty, W. Kellermann, An extended multidelay filter: fast low-delay algorithms for very high-order adaptive systems, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong Kong, China, vol. 5 (April 2003) 385–388.
- [10] H. Buchner, W. Herbordt, W. Kellermann, An efficient combination of multichannel acoustic echo cancellation with a beamforming microphone array, in: Proceedings of International Workshop on Hands-Free Speech Communication, Kyoto, Japan, April 2001, pp. 55–58.
- [11] J.M. Cioffi, T. Kailath, Fast recursive-least-squares transversal filters for adaptive filtering, IEEE Trans. Acoust. Speech Signal Process. ASSP-32 (April 1984) 304–337.
- [12] M. Dentino, J. McCool, B. Widrow, Adaptive filtering in the frequency domain, Proc. IEEE 66 (December 1978) 1658–1659.
- [13] E.R. Ferrara Jr., Fast implementation of the LMS adaptive filter, IEEE Trans. Acoust. Speech Signal Process. ASSP-28 (August 1980) 474–475.
- [14] V. Fischer, T. Gänslér, E.J. Diethorn, J. Benesty, A software stereo acoustic echo canceler for microsoft windows, in: Proceedings of International Workshop on Acoustic Echo and Noise Control, Darmstadt, Germany, September 2001, pp. 87–90.
- [15] T. Gänslér, J. Benesty, Stereophonic acoustic echo cancellation and two-channel adaptive filtering: an overview, Int. J. Adaptive Control Signal Process. (February 2000).
- [16] T. Gänslér, P. Eneroth, Influence of audio coding on stereophonic acoustic echo cancellation, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, May 1998, pp. 3649–3652.
- [17] G.H. Golub, C.F. Van Loan, Matrix Computations, second ed., Johns Hopkins, Baltimore, MD, 1989.
- [18] L.J. Griffith, C.W. Jim, An alternative approach to linearly constrained adaptive beamforming, IEEE Trans. Antennas Propagation 30 (1) (January 1982) 27–34.
- [19] S. Haykin, Adaptive Filter Theory, third ed., Prentice Hall Inc., Englewood Cliffs, NJ, 1996.
- [20] S. Holm, FFT pruning applied to time domain interpolation and peak localization, IEEE Trans. Acoust., Speech Signal Process. ASSP-35 (12) (December 1987) 1776–1778.
- [21] O. Hoshuyama, A. Sugiyama, A. Hirano, A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters, IEEE Trans. Signal Process. 47 (10) (October 1999) 2677–2684.
- [22] W. Kellermann, Acoustic echo cancellation for beamforming microphone arrays, in: M.S. Brandstein, D.B. Ward (Eds.), Microphone Arrays, Springer, Berlin, 2001.
- [23] J.C. Lee, C.K. Un, Performance analysis of frequency-domain block LMS adaptive digital filters, IEEE Trans. Circuits Systems CAS-36 (February 1989) 173–189.
- [24] D. Mansour, A.H. Gray, Unconstrained frequency-domain adaptive filter, IEEE Trans. Acoustics, Speech Signal Process. 30 (5) (October 1982).
- [25] J.D. Markel, FFT pruning, IEEE Trans. Audio Electroacoust AU-19 (December 1971) 305–311.
- [26] V.A. Morozov, Regularization Methods for Ill-posed Problems, CRC Press, Boca Raton, FL, 1993.
- [27] E. Moulines, O. Ait Amrane, Y. Grenier, The generalized multidelay adaptive filter: structure and convergence analysis, IEEE Trans. Signal Process. 43 (January 1995) 14–28.
- [28] J. Prado, E. Moulines, Frequency-domain adaptive filtering with applications to acoustic echo cancellation, Ann. Télécommun. 49 (1994) 414–428.
- [29] S. Shimauchi, S. Makino, Stereo projection echo canceller with true echo path estimation, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Process. May 1995, pp. 3059–3062.
- [30] M.M. Sondhi, D.R. Morgan, Stereophonic acoustic echo cancellation—an overview of the fundamental problem, IEEE SP Lett. 2 (8) (August 1995) 148–151.
- [31] J.-S. Soo, K.K. Pang, Multidelay block frequency domain adaptive filter, IEEE Trans. Acoust. Speech Signal Process. ASSP-38 (February 1990) 373–376.
- [32] A. Sugiyama, Y. Joncour, A. Hirano, A stereo echo canceler with correct echo-path identification on an input-sliding technique, IEEE Trans Signal Process. 49 (11) (November 2001) 2577–2587.
- [33] B. Widrow, S.D. Stearns, Adaptive Signal Processing, Prentice-Hall Inc., Englewood Cliffs, NJ, 1985.