A Generalization of Blind Source Separation Algorithms for Convolutive Mixtures Based on Second-Order Statistics

Herbert Buchner, *Member, IEEE*, Robert Aichner, *Student Member, IEEE*, and Walter Kellermann, *Member, IEEE*

***Abstract*—In this paper, we present a general broadband approach to blind source separation (BSS) for convolutive mixtures based on second-order statistics. This avoids several known limitations of the conventional narrowband approximation, such as the internal permutation problem. In contrast to traditional narrowband approaches, the new framework simultaneously exploits the nonwhiteness property and nonstationarity property of the source signals. Using a novel matrix formulation, we rigorously derive the corresponding time-domain and frequency-domain broadband algorithms by generalizing a known cost-function which inherently allows joint optimization for several time-lags of the correlations. Based on the broadband approach time-domain, constraints are obtained which provide a deeper understanding of the internal permutation problem in traditional narrowband frequency-domain BSS. For both the time-domain and the frequency-domain versions, we discuss links to well-known, and also, to novel algorithms that constitute special cases. Moreover, using the so-called generalized coherence, links between the time-domain and the frequency-domain algorithms can be established, showing that our cost function leads to an update equation with an inherent normalization ensuring a robust adaptation behavior. The concept is applicable to offline, online, and block-online algorithms by introducing a general weighting function allowing for tracking of time-varying real acoustic environments.**

***Index Terms*—Adaptive filtering, blind source separation (BSS), convolutive mixtures, multiple-input–multiple-output (MIMO) systems, second-order statistics.**

I. Introduction

THE PROBLEM of separating convolutive mixtures of unknown time series arises in several application domains, a prominent example being the so-called cocktail party problem, where individual speech signals should be extracted from mixtures of multiple speakers in a usually reverberant acoustic environment. Due to the reverberation, the original source signals $s_q(n)$, $q = 1, \ldots, Q$ of our separation problem are filtered by a linear multiple-input–multiple-output (MIMO) system before they are picked up by the sensors. Blind source separation (BSS) is solely based on the fundamental assumption of mutual statistical independence of the different source signals. In the following, we further assume that the number Q of source signals

Manuscript received August 19, 2003; revised August 16, 2004. The guest editors coordinating the review of this paper and approving it for publication were Dr. Mohan Sondhi and Dr. Diemer de Vries.

The authors are with Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, 91058 Erlangen, Germany (e-mail: buchner@LNT.de; aichner@LNT.de; wk@LNT.de).

Digital Object Identifier 10.1109/TSA.2004.838775

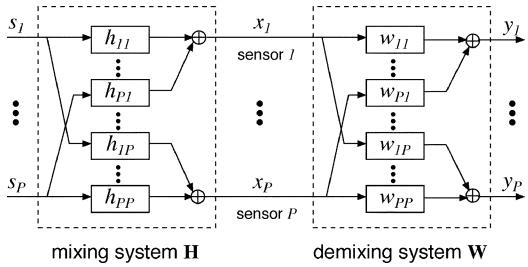


Fig. 1. Linear MIMO model for BSS.

$s_q(n)$ equals the number of sensor signals $x_p(n), p = 1, \ldots, P$ (Fig. 1). An M-tap mixing system is thus described by

$$x_p(n) = \sum_{q=1}^{P} \sum_{\kappa=0}^{M-1} h_{qp}(\kappa) s_q(n-\kappa) \tag{1}$$

where $h_{qp}(\kappa), \kappa = 0, \ldots, M-1$ denote the coefficients of the filter from the qth source to the pth sensor.

In BSS, we are interested in finding a corresponding demixing system according to Fig. 1, where the output signals $y_q(n), q = 1, \ldots, P$ are described by

$$y_q(n) = \sum_{p=1}^{P} \sum_{\kappa=0}^{L-1} w_{pq}(\kappa) x_p(n-\kappa). \tag{2}$$

The separation is achieved by forcing the output signals y_q to be mutually statistically decoupled up to joint moments of a certain order. Note that this approach is not aiming at a deconvolution of the individual signals x_q. For deconvolution it was shown in [1] that for $P > Q$ an exact inversion of the mixing system and thus, a perfect reconstruction of the sources is possible. Moreover, in [2], conditions for blind identification were given. However, in contrast, in BSS, a blind interference cancellation (similar to conventional adaptive beamforming) is performed, so that in fact, the MIMO demixing system coefficients $w_{pq}(\kappa)$ can reconstruct the sources up to an unknown (external) permutation and an unknown filtering of the individual signals (see, e.g., [3]), where, ideally, L should be chosen at least equal to M to allow modeling of all reflections.

In this paper, in order to estimate the P^2L MIMO coefficients $w_{pq}(\kappa)$, we only consider approaches using *second-order statistics*. For convolutive mixtures, frequency-domain BSS is very popular since all techniques originally developed for instantaneous BSS can be applied independently in each frequency bin. In the following, this bin-wise processing, implying a narrowband signal model is denoted as the *narrowband approach*. Such

narrowband approaches for convolutive BSS and methods for instantaneous BSS can be found, e.g., in [3]–[14].

In the context of instantaneous BSS and narrowband approaches for convolutive BSS, it is known that on real-world signals with some time-structure, second-order statistics generates enough constraints to solve the BSS problem in principle, by utilizing one of the following two signal properties [3]:

- Nonwhiteness property by simultaneous diagonalization of output correlation matrices over multiple time-lags, e.g., [4], [5];
- Nonstationarity property by simultaneous diagonalization of short-time output correlation matrices at different time intervals, e.g., [6]–[14].

Unfortunately, this traditional narrowband approach exhibits several limitations as identified in, e.g., [15]–[17]. In particular, the permutation problem, which is inherent in BSS, may then also appear independently in each frequency bin so that extra repair measures have to be taken to address this *internal* permutation. Problems caused by circular convolution effects due to the narrowband approximation are reported in, e.g., [16].

Extending the work in [18], we present, in Section II, a rigorous derivation of a more general class of *broadband* algorithms for convolutive mixtures based on second-order statistics, i.e., the frequency bins are no longer considered to be independent for unrestricted time-domain signals. This general approach inherently avoids the above-mentioned problems, such as the internal permutation ambiguity, discussed in Section III, by introducing a general matrix formulation for convolutive mixtures that includes multiple time-lags. We will demonstrate that the general broadband approach in fact implies the necessity to account for both, the nonwhiteness property and the nonstationarity property simultaneously.

After deriving a generic time-domain algorithm, we also introduce an equivalent broadband formulation in the frequency domain in Section IV by extending the framework of [19] to unsupervised adaptive filtering. Moreover, links between the time-domain and frequency-domain algorithms are established. As discussed in Section III and IV-C, this provides deeper insight into the internal permutation problem of conventional narrowband frequency-domain BSS. For both, the time-domain and the frequency-domain versions, we extensively discuss relations to well-known and state-of-the-art algorithms from the literature in Section II-E, Section IV-C–D, and present several novel algorithms following as special cases of the framework. In this regard, another very useful aspect is that many well-known and powerful techniques from the vast literature on supervised adaptive filtering (e.g., [20]) can now be directly applied to the BSS application.

The framework shown here is applicable for online, block-online, and offline algorithms as it uses a general weighting function allowing for tracking of time-varying environments [21]. The processing delay can be kept low by overlapping/partitioned signal blocks [22].

As illustrated by experimental results, the framework allows an efficient separation of real-world speech signals in reverberant, noiseless environments. Simulation results show that the general broadband approach does not suffer from the above-mentioned specific limitations of narrowband algorithms. The broadband approach has also led to the development of a high-performance real-time BSS system on a regular PC which can cope with reverberant environments (e.g., office rooms) [23]. Moreover, this was also successfully applied to noisy car environments [24].

More recently, it has been shown that the framework presented here can be efficiently extended to higher-order statistics so that in addition to nonwhiteness and nonstationarity, the nongaussianity of the source signals can be exploited [25]. There, it has also been proven that the second-order approach presented here is the optimum second-order BSS approach for convolutive mixtures in the sense of minimum mutual information known from information theory.

II. Generic Block Time-Domain BSS Algorithm

In this section, we first introduce a general matrix formulation allowing a rigorous derivation of time-domain algorithms from a cost function which inherently takes into account the nonwhiteness and nonstationarity properties. We then consider the so-called equivariance property in the convolutive case and the corresponding natural gradient formulation. From this formulation, several well-known and novel algorithms follow as special cases.

A. Matrix Formulation for Convolutive Mixtures With Extension to Several Time-Lags

From Fig. 1, it can be seen that the output signals $y_q(n)$, $(q = 1, \ldots, P)$ of the unmixing system at time n are given by

$$y_q(n) = \sum_{p=1}^{P} \mathbf{x}_p^T(n)\mathbf{w}_{pq} \tag{3}$$

where

$$\mathbf{x}_p(n) = [x_p(n), x_p(n-1), \ldots, x_p(n-L+1)]^T \tag{4}$$

is a vector containing the latest L samples of the sensor signal x_p of the pth channel, and where

$$\mathbf{w}_{pq} = [w_{pq,0}, w_{pq,1}, \ldots, w_{pq,L-1}]^T \tag{5}$$

contains the current weights of the MIMO filter taps from the pth sensor channel to the qth output channel. Superscript ^{T} denotes transposition of a vector or a matrix.

In addition to the filter length L and the number of channels P we need to introduce two more parameters for the following general formulation:

- number of time-lags D taken into account for the correlations in the cost function below $(1 \leq D \leq L)$;
- length N of output signal blocks as basis for the estimates of short-time correlations below.

We define a block output signal vector of length $N \geq PD$ (to obtain a correlation matrix $\mathbf{R}_{\mathbf{yy}}$ of full rank in (17)). From (3) it follows

$$\mathbf{y}_q(m) = \sum_{p=1}^{P} \mathbf{U}_p^T(m)\mathbf{w}_{pq} \tag{6}$$

with block time index m, and

$$\mathbf{y}_q(m) = [y_q(mL), \ldots, y_q(mL+N-1)]^T \quad (7)$$
$$\mathbf{U}_p(m) = [\mathbf{x}_p(mL), \ldots, \mathbf{x}_p(mL+N-1)]. \quad (8)$$

Analogously to supervised block-based adaptive filtering [19], [22], the approach followed here can also be carried out with overlapping/partitioned data blocks to increase the convergence rate and to reduce the signal delay. Overlapping is introduced by simply replacing the time index mL in the equations by $m(L/\alpha)$ with the overlap factor $1 \leq \alpha \leq L$. For clarity, we will omit the overlap factor and will point to it when necessary.

Obviously, $\mathbf{U}_p(m), p = 1, \ldots, P$ in (8) are Toeplitz matrices of size $(L \times N)$

$$\mathbf{U}_p^T(m) = \begin{bmatrix} x_p(mL) & \cdots & x_p(mL-L+1) \\ x_p(mL+1) & \ddots & x_p(mL-L+2) \\ \vdots & \ddots & \vdots \\ x_p(mL+N-1) & \cdots & x_p(mL-L+N) \end{bmatrix}.$$

Next, in order to incorporate D *time-lags* in the cost function as follows, we capture D subsequent output signal vectors (7) in the following $N \times D$ matrix

$$\mathbf{Y}_q(m) = \begin{bmatrix} y_q(mL) & \cdots & y_q(mL-D+1) \\ y_q(mL+1) & \ddots & y_q(mL-D+2) \\ \vdots & \ddots & \vdots \\ y_q(mL+N-1) & \cdots & y_q(mL-D+N) \end{bmatrix}.$$

Using this definition, (6) can be extended to

$$\mathbf{Y}_q(m) = \sum_{p=1}^{P} \mathbf{X}_p(m)\mathbf{W}_{pq} \quad (9)$$

with $\mathbf{X}_p(m)$ and $\mathbf{W}_{pq}$ as determined in the following. We have to ensure linear convolutions for all elements of $\mathbf{Y}_q(m)$ up to the maximum number of time-lags $D = L$ as shown in [18]. Therefore, and also with regard to the frequency-domain realization in Section IV, two blocks of the input signals $\mathbf{U}_p^T$ are required. Thus, the sizes of $\mathbf{X}_p(m)$ and $\mathbf{W}_{pq}$ must be $N \times 2L$ and $2L \times D$, respectively. The matrices $\mathbf{X}_p(m)$ are obtained from the Toeplitz matrices $\mathbf{U}_p$ by doubling their size, i.e.,

$$\mathbf{X}_p(m) = \left[\mathbf{U}_p^T(m), \mathbf{U}_p^T(m-1)\right]. \quad (10)$$

The matrices $\mathbf{U}_p^T(m-1), p = 1, \ldots, P$ are also Toeplitz so that the first row of $\mathbf{X}_p(m)$ contains $2L$ input samples and each subsequent row is shifted to the right by one sample and thus, contains one new input sample. $\mathbf{W}_{pq}$ are $2L \times D$ Sylvester matrices, which are defined as

$$\mathbf{W}_{pq}(m) = \begin{bmatrix} w_{pq,0} & 0 & \cdots & 0 \\ w_{pq,1} & w_{pq,0} & \ddots & \vdots \\ \vdots & w_{pq,1} & \ddots & 0 \\ w_{pq,L-1} & \vdots & \ddots & w_{pq,0} \\ 0 & w_{pq,L-1} & \ddots & w_{pq,1} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & w_{pq,L-1} \\ 0 & \cdots & 0 & 0 \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix}. \quad (11)$$

It can be seen that for the general case $1 \leq D \leq L$ the last $L - D + 1$ rows are padded with zeros to ensure compatibility with $\mathbf{X}_p(m)$. Finally, to allow a convenient notation of the algorithm combining all channels, we write (9) compactly as

$$\mathbf{Y}(m) = \mathbf{X}(m)\mathbf{W} \quad (12)$$

with the matrices

$$\mathbf{Y}(m) = [\mathbf{Y}_1(m), \cdots, \mathbf{Y}_P(m)] \quad (13)$$
$$\mathbf{X}(m) = [\mathbf{X}_1(m), \cdots, \mathbf{X}_P(m)] \quad (14)$$
$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \cdots & \mathbf{W}_{1P} \\ \vdots & \ddots & \vdots \\ \mathbf{W}_{P1} & \cdots & \mathbf{W}_{PP} \end{bmatrix}. \quad (15)$$

B. Cost Function and Algorithm Derivation

The aim of BSS for convolutive mixtures according to Section I, i.e., separation of the output channels, while putting no constraints on the temporal structure of the individual signals, can be expressed in terms of the overall system from the sources to the demixing filter outputs. Analogously to the demixing system $\mathbf{W}$ in (12), this system can be expressed by a blockwise Sylvester matrix $\mathbf{C}$ of suitable dimensions. The mixing process is described analogously to (12) by $\mathbf{X} = \mathbf{SH}$, where $\mathbf{S}$ is the corresponding $N \times P(M+L-1)$ source signal matrix with time shifts, and $\mathbf{H}$ is the $P(M+L-1) \times 2PL$ mixing matrix in Sylvester structure. The dimensions result from the linearity condition of the convolution. Due to the inevitable filtering ambiguity in convolutive BSS (e.g., [3]), the output signals can become mutually independent but there may still exist channel-wise arbitrary filtering. Thus, it is at best possible to obtain an arbitrary *block diagonal* matrix $\mathbf{C} = \mathbf{HW}$, i.e., $\mathbf{C} - \text{bdiag}\,\mathbf{C} = \text{boff}\,\mathbf{C} = \mathbf{0}$ (however, our experience is that with our broadband approach the spectral content of the individual channels is not noticeably affected). The bdiag operation on a partitioned block matrix consisting of several submatrices sets all submatrices on the off-diagonals to zero. Here, the submatrices refer to the different signal channels. Analogously, the boff operation sets all submatrices on the diagonal to zero.

To reach this goal, different approaches exploiting nonwhiteness, nonstationarity, and/or nongaussianity are known. In [25], a unified treatment for convolutive mixtures based on an information-theoretic criterion has been introduced. In the following,

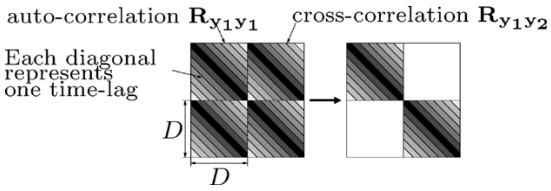


Fig. 2. Illustration of (18) for the 2×2 case.

we focus on wideband solutions based on second-order statistics. Thus, the stochastic signals are described here by a sequence of short-time correlation matrices with lags based on the matrix formulation, introduced above. We define the short-time correlation matrices

$$\mathbf{R}_{\mathbf{xx}}(m) = \mathbf{X}^H(m)\mathbf{X}(m) \tag{16}$$

$$\mathbf{R}_{\mathbf{yy}}(m) = \mathbf{Y}^H(m)\mathbf{Y}(m) \tag{17}$$

of size $2PL \times 2PL$ and $PD \times PD$, respectively. Note that in principle, there are two basic methods to estimate the output correlation matrices for nonstationary output signals: the so-called correlation method, and the covariance method as they are known from linear prediction problems [26]. While the correlation method leads to a slightly lower computational complexity (and to smaller matrices, when implemented in the frequency domain covered in Section IV), we consider the more accurate covariance method in this paper. The correlation method follows as a special case if we assume stationarity within each block [23]. Note also that (17) is full rank since in general we assume $N \geq PD$.

Having defined the compact matrix formulation (12) for the block-MIMO filtering, we now define the following cost function, based on the sequence of lagged correlation matrices:

$$\mathcal{J}(m) = \sum_{i=0}^{\infty} \beta(i,m) \left\{ \log\det \mathrm{bdiag}\mathbf{Y}^{\mathrm{H}}(\mathrm{i})\mathbf{Y}(\mathrm{i}) - \log\det \mathbf{Y}^H(i)\mathbf{Y}(i) \right\}. \tag{18}$$

This cost function, introduced in [18], in generalization of [8] also follows from the general information-theoretic approach in [25]. In Section II-D, it is shown that the equilibrium points of (18) actually correspond exactly to the desired BSS solution boff $\mathbf{C} = \mathbf{0}$.

The quantity β in (18) is a weighting function with finite support that is normalized according to $\sum_{i=0}^{\infty} \beta(i,m) = 1$ which allows offline and online implementations of the algorithms as shown in Section V (e.g., $\beta(i,m) = (1-\lambda)\lambda^{m-i}$ for $0 \leq i \leq m$, and $\beta(i,m) = 0$, elsewhere leads to an efficient online version allowing for tracking in time-varying environments [21]). Since we use the matrix formulation (12) for calculating the short-time correlation matrices $\mathbf{Y}^H(m)\mathbf{Y}(m)$, the cost function inherently includes all D time-lags of all auto-correlations and cross-correlations of the BSS output signals. By Oppenheim's inequality [27] $\sum_q \log\det \mathbf{Y}_q^H\mathbf{Y}_q \geq \log\det \mathbf{Y}^H\mathbf{Y}$, it is ensured that the first term in the braces in (18) is always greater than or equal to the second term, where the equality holds if all block-offdiagonal elements of $\mathbf{Y}^H\mathbf{Y}$, i.e., the *output cross-correlations over all time-lags*, vanish (Fig. 2). In geometrical terms

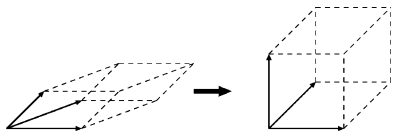


Fig. 3. Parallelepiped.

the mechanism of the optimization criterion (18) can be interpreted using a volume of a parallelepiped spanned by the column vectors of a matrix (Fig. 3) which is described by its determinant. Thus, minimizing (18) corresponds to a simultaneous orthogonalization relative to several subspaces represented by the submatrices $\mathbf{Y}_q^H\mathbf{Y}_q$ on the main diagonal of $\mathbf{Y}^H\mathbf{Y}$.

In this paper, we consider algorithms based on first-order gradients. An extension to higher order gradients would be straightforward but computationally more expensive. In the following equations, we omit the block-time index m for simplicity. In order to express the update equations of the filter coefficients exclusively by Sylvester matrices $\mathbf{W}$, we take the gradient with respect to $\mathbf{W}$ and ensure the Sylvester structure of the result by selecting the nonredundant values using a constraint $(\mathcal{SC})$. For the derivation of the gradient

$$\nabla_{\mathrm{W}}\mathcal{J}(m) = 2\frac{\partial \mathcal{J}(m)}{\partial \mathbf{W}^*} \tag{19}$$

we use the expression (complex version of, e.g., [28] and [29])

$$\frac{\partial}{\partial \mathbf{W}^*} \log\det \mathbf{W}^H\mathbf{R}_{\mathbf{xx}}\mathbf{W} = \mathbf{R}_{\mathbf{xx}}\mathbf{W}(\mathbf{W}^{\mathbf{H}}\mathbf{R}_{\mathbf{xx}}\mathbf{W})^{-1} \tag{20}$$

and (see Appendix)

$$\frac{\partial}{\partial \mathbf{W}^*} \log\det \mathrm{bdiag}\mathbf{W}^H\mathbf{R}_{\mathbf{xx}}\mathbf{W} = \mathbf{R}_{\mathbf{xx}}\mathbf{W}(\mathrm{bdiag}\mathbf{W}^H\mathbf{R}_{\mathbf{xx}}\mathbf{W})^{-1}. \tag{21}$$

Using these relations, it follows from (18)

$$\begin{aligned}\nabla_{\mathbf{W}}\mathcal{J}(m) &= 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{R}_{\mathbf{xx}}\mathbf{W}\{\mathrm{bdiag}^{-1}\mathbf{R}_{\mathrm{yy}} - \mathbf{R}_{\mathrm{yy}}^{-1}\} \\ &= 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{R}_{\mathbf{xy}}\{\mathrm{bdiag}^{-1}\mathbf{R}_{\mathrm{yy}} - \mathbf{R}_{\mathrm{yy}}^{-1}\} \\ &= 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{R}_{\mathbf{xy}}\mathbf{R}_{\mathrm{yy}}^{-1}\{\mathbf{R}_{\mathrm{yy}} - \mathrm{bdiag}\mathbf{R}_{\mathrm{yy}}\} \\ &\quad \cdot \mathrm{bdiag}^{-1}\mathbf{R}_{\mathrm{yy}}\end{aligned} \tag{22}$$

where $\mathrm{bdiag}^{-1}\mathbf{R}_{\mathbf{yy}} = (\mathrm{bdiag}\,\mathbf{R}_{\mathbf{yy}})^{-1}$. With an iterative optimization procedure, the current demixing matrix is obtained by the recursive update equation

$$\mathbf{W}(m) = \mathbf{W}(m-1) - \mu\Delta\mathbf{W}(m) \tag{23}$$

where μ is a stepsize parameter, and $\Delta\mathbf{W}(m)$ is the update which is set equal to $\nabla_{\mathbf{W}}\mathcal{J}(m)$ for gradient descent adaptation.

A simple way to impose the constraint $(\mathcal{SC})$ for this generic update is to select a certain column or the Lth row as a reference (as they contain all filter weights $w_{pq,\kappa}, \kappa = 0, \cdots, L-1$) and generate $\Delta\mathbf{W}(m)$ in the form of (11) from it. As can be shown from the considerations in Section II-D, the steady-state performance does not depend on the choice of the reference

values. For special cases, and frequency-domain versions discussed later, we will give further methods for enforcing this constraint.

C. Equivariance Property and Natural Gradient

It is known that stochastic gradient descent suffers from slow convergence in many practical problems due to dependencies in the data being processed.

In the BSS application, we can show that the separation performance using (23) together with (22) depends on the MIMO mixing system $\mathbf{H}$. For verification, we premultiply (22) by $\mathbf{H}$ which shows that $\mathbf{C}(m) = \mathbf{H}\mathbf{W}(m)$ always depends on $\mathbf{H}$.

Fortunately, a modification of the ordinary gradient has been developed that largely removes all effects of an ill-conditioned mixing matrix $\mathbf{H}$. Termed the *natural gradient* by Amari [30] and the *relative gradient* by Cardoso [31], this modification is usually written in the following way:

$$\nabla_{\mathbf{W}}^{NG}\mathcal{J} = \nabla_{\mathbf{W}}\mathcal{J}\mathbf{W}^H\mathbf{W}. \tag{24}$$

For our approach based on (12), we have to reorder the right side, i.e.,

$$\nabla_{\mathbf{W}}^{NG}\mathcal{J} = \mathbf{W}\mathbf{W}^H\nabla_{\mathbf{W}}\mathcal{J}. \tag{25}$$

This leads to the following expression:

$$\nabla_{\mathbf{W}}^{NG}\mathcal{J}(m) = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{W}\{\mathbf{R}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{R}_{\mathbf{yy}}\} \cdot\mathrm{bdiag}^{-1}\mathbf{R}_{\mathbf{yy}} \tag{26}$$

which is used as update $\Delta\mathbf{W}$ in (23). In the derivation of the natural gradient for instantaneous mixtures in [30], the fact that the demixing matrices form a Lie group has played an important role. However, the blockwise-Sylvester matrices $\mathbf{W}$ after (11), (15) do not form a Lie group. To see that the above formulation of the natural gradient is indeed justified, we again premultiply the update (26), which leads to

$$\Delta\mathbf{C}(m) = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{C}\{\mathbf{R}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{R}_{\mathbf{yy}}\}\mathrm{bdiag}^{-1}\mathbf{R}_{\mathbf{yy}}. \tag{27}$$

The evolutionary behavior of $\mathbf{C} = \mathbf{C}(m)$ depends only on the estimated source signal vector sequence and the stepsize μ, and the mixing matrix $\mathbf{H}$ has been absorbed as an initial condition into $\mathbf{C}(0) = \mathbf{H}\mathbf{W}(0)$ as desired. The uniform performance provided by (26) is due to the so-called equivariance property provided by the relative gradient BSS update [31]. In our case, only the modified relative gradient (25) exhibits this property.

Another well known advantage of using the natural/relative gradient is a reduction of the computational complexity of the update as the inversion of the $PD \times PD$ matrix $\mathbf{R}_{\mathbf{yy}}$ in (22) need not be carried out in (26). Instead, only submatrices of size $D \times D$ have to be inverted. Moreover, this implies that instead of $N \geq PD$ the condition $N \geq D$ is sufficient for the natural gradient update. Therefore, we consider without loss of generality this somewhat relaxed condition in the following.

Moreover, noting that the products of Sylvester matrices $\mathbf{W}_{pq}$ and the remaining matrices in the update (26) can be interpreted as linear convolutions, they can be efficiently implemented by a fast convolution as in [32].

D. On the Convergence to the Desired Solution

In this section, we prove that the equilibrium points of the highly nonlinear multivariate cost function (18) correspond only to *global* optima if parameter D is chosen appropriately, i.e., a convergence to local optima, other than the desired solution (see Section II-B) $\mathrm{boff}\{\mathbf{C}\} = \mathbf{0}$ (or due to the external permutation ambiguity any channel-wise permuted version thereof) can be avoided by the coefficient update (26). However, proper initialization of the demixing filter coefficients (see Section VI) is necessary.

To begin with, we express (27) using $\mathbf{R}_{\mathbf{yy}} = \mathbf{Y}^H\mathbf{Y} = \mathbf{W}^H\mathbf{X}^H\mathbf{X}\mathbf{W} = \mathbf{W}^H\mathbf{H}^H\mathbf{S}^H\mathbf{S}\mathbf{H}\mathbf{W} = \mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}$ exclusively by the input statistics, and the overall system $\mathbf{C}$. Without loss of generality, we consider here the online adaptation ($\beta(i,m) = \delta(i-m)$, see Section V) as variations of $\beta(i,m)$ influence only the *speed* of convergence. By setting $\Delta\mathbf{C} = \mathbf{0}$, we then obtain the condition for equilibrium points from (27) as

$$\begin{aligned}\mathbf{C}\left[\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\mathrm{bdiag}^{-1}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} - \mathbf{I}\right] &= \mathbf{0}\\ \mathbf{C}\left[\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\mathrm{bdiag}^{-1}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\}\right] &= \mathbf{C}\end{aligned}$$

or

$$\mathbf{C}(\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}) = \mathbf{C}\mathrm{bdiag}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} \tag{28}$$

and finally

$$\mathbf{C}\mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} = \mathbf{0}. \tag{29}$$

For the discussion of solutions to this equation, we distinguish the three following cases:

Case 1) $\mathbf{C} = \mathbf{0} \wedge \mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} \neq \mathbf{0}$;
Case 2) $\mathbf{C} \neq \mathbf{0} \wedge \mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} = \mathbf{0}$;
Case 3) $\mathbf{C} \neq \mathbf{0} \wedge \mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} \neq \mathbf{0}$.

Case 1) is impossible as $\mathbf{C} = \mathbf{0}$ also implies $\mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} = \mathbf{0}$. Moreover, the trivial case $\mathbf{C} = \mathbf{H}\mathbf{W} = \mathbf{0}$ is easily avoided by proper initialization of $\mathbf{W}$.

Case 2) gives the desired solutions (and only the desired ones). Since $\mathrm{boff}\{\mathbf{R}_{\mathbf{ss}}\} = \mathbf{0}$ according to the fundamental assumption of independent sources, we obtain from $\mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} = \mathbf{0}$ the desired solutions

$$\mathrm{boff}\{\mathbf{C}\} = 0 \tag{30}$$

and all channel-wise (external) permutations.

Case 3) represents undesired, possibly suboptimum solutions which cannot appear with scalar mixtures and must be precluded by a proper choice of parameter D (controlling the exploitation of nonwhiteness) as will be demonstrated in the sequel.

We discuss Case 3) in two steps. To begin with, we ignore the Sylvester structure of $\mathbf{C}$ and $\mathbf{W}$, imposed by $(\mathcal{SC})$. Then, a necessary and sufficient condition to obtain the desired solution (30) is that the $P(M+L-1) \times DP$-matrix $\mathbf{C}$ is of full rank, making

condition (29) again equivalent to $\mathrm{boff}\{\mathbf{C}^H\mathbf{R}_{\mathbf{ss}}\mathbf{C}\} = \mathbf{0}$, i.e., to the desired case (b). In terms of dimensions, this requires $D = M + L - 1$, and, since, ideally, $L = M$, we have $D = 2L - 1$. Note that in the well-known case of *scalar* mixtures, we have $M = L = 1$, i.e., in principle, global convergence is always guaranteed by taking into account only nonstationarity [6] as D is then chosen to 1. In the convolutive case, we have to increase the number D, i.e., simultaneously exploit the nonwhiteness, to achieve improved performance. This will also be demonstrated in Fig. 8. In addition, for a full rank of matrix $\mathbf{C}$ the different channels of its z-transform must not have any common zeros in the z-plane according to [1]. In practical acoustic scenarios this assumption is usually fulfilled [1].

Now, we take into account the Sylvester structure of $\mathbf{C}$ and $\mathbf{W}$, respectively, imposed by $(\mathcal{SC})$. Generally, the update (26) consists of a matrix multiplication of $\mathbf{W}$ (size $2L \times D$ for each channel) and another matrix (of size $D \times D$ for each channel) based on $\mathbf{R}_{\mathbf{yy}}$. A close examination shows that the second part of (26) constitutes the search space for the coefficient update $\Delta\mathbf{W}(m)$, and with $D = 2L - 1$, all elements of $\mathbf{W}$ can be adjusted independently from each other toward the global optimum as discussed above. However, by examining the channel-wise matrix multiplication of $\mathbf{W}_{pq}$ and the second part of (26), we observe that due to the Sylvester structure, the upper left $L \times L$ submatrix of $\Delta\mathbf{W}_{pq}$ is fully determined by the upper left $L \times L$ submatrices of $\mathbf{W}_{pq}$ and the second part, respectively. Moreover, it can be seen from (11) that when considering only this upper left $L \times L$ submatrix, only the first column or the Lth row contain all L filter weights. Therefore, D is reducible from $2L$ to L if and only if the first column or the Lth row of $\Delta\mathbf{W}$ is chosen as a reference for $(\mathcal{SC})$, i.e., if we pick the first column or Lth row of $\mathbf{W}$ and construct the Sylvester structure from it. Several simulations under various conditions have confirmed this general finding.

E. Special Cases and Links to Known Time-Domain Algorithms

To further illustrate the generalized update (26), and to study links to some known algorithms along with Fig. 4, we consider now the case $P = Q = 2$ for simplicity. In this case, we have

$$\begin{aligned}\nabla_{\mathbf{W}}^{NG}\mathcal{J}(m) &= 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{W} \\ &\quad\cdot\begin{bmatrix}\mathbf{0} & \mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}\mathbf{R}_{\mathbf{y}_2\mathbf{y}_2}^{-1} \\ \mathbf{R}_{\mathbf{y}_2\mathbf{y}_1}\mathbf{R}_{\mathbf{y}_1\mathbf{y}_1}^{-1} & \mathbf{0}\end{bmatrix} \\ &= 2\sum_{i=0}^{\infty}\beta(i,m) \\ &\quad\cdot\begin{bmatrix}\mathbf{W}_{12}\mathbf{R}_{\mathbf{y}_2\mathbf{y}_1}\mathbf{R}_{\mathbf{y}_1\mathbf{y}_1}^{-1} & \mathbf{W}_{11}\mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}\mathbf{R}_{\mathbf{y}_2\mathbf{y}_2}^{-1} \\ \mathbf{W}_{22}\mathbf{R}_{\mathbf{y}_2\mathbf{y}_1}\mathbf{R}_{\mathbf{y}_1\mathbf{y}_1}^{-1} & \mathbf{W}_{21}\mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}\mathbf{R}_{\mathbf{y}_2\mathbf{y}_2}^{-1}\end{bmatrix}\end{aligned} \tag{31}$$

where $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}$, $p, q \in \{1,2\}$ are the corresponding $D \times D$ submatrices of $\mathbf{R}_{\mathbf{yy}}$. It can be seen that (31) exhibits an inherent normalization by the autocorrelation matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}$ as known from the recursive least-squares (RLS) algorithm in supervised adaptive filtering [20]. Note that the matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}$ have to be properly regularized prior to inversion, e.g., by adding a constant to the main diagonal. The update (31) results in an $\mathcal{O}(D^3)$-complexity for a straightforward implementation. However, due to the similarity of the update equation to supervised adaptive filtering, fast calculation schemes and other approximations are well possible [20].

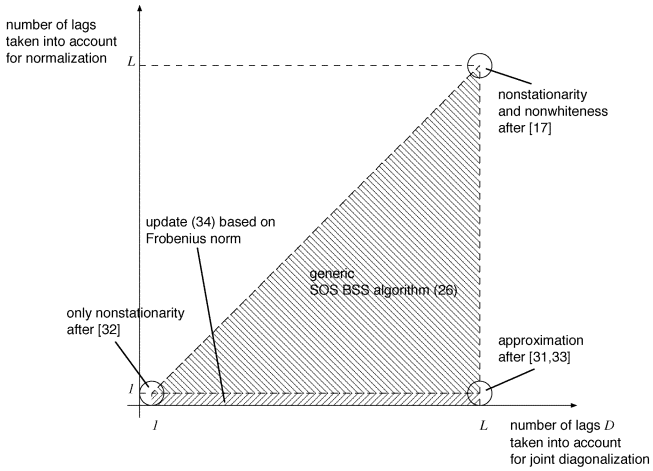


Fig. 4. Overview of time-domain algorithms based on second-order statistics.

For $D = 1$, we obtain the algorithm proposed in [33] which takes only the nonstationarity property into account.

In [32] and [34], a time-domain algorithm was presented that copes very well with reverberant acoustic environments. Although it was originally introduced as a heuristic extension of [33] incorporating several time-lags, this algorithm can be directly obtained from (26) or (31) for $D = L$ by approximating the block diagonals of $\mathbf{R}_{\mathbf{yy}}(m)$ by the output signal powers, i.e.,

$$\tilde{\mathbf{R}}_{\mathbf{y}_q\mathbf{y}_q}(m) = \mathbf{y}_q^H(m)\mathbf{y}_q(m)\mathbf{I} \tag{32}$$

for $q = 1, \ldots, P$. Thus, this approximation is comparable to the well-known normalized least mean squares (NLMS) algorithm in supervised adaptive filtering approximating the RLS algorithm [20]. In addition to the reduced computational complexity, we can ensure the Sylvester structure of the update by using the autocorrelation method as in [32] resulting in Toeplitz matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}(m)$. The straightforward implementation of this algorithm in the time domain leads to a complexity of $\mathcal{O}(L)$. By using fast convolution techniques, the complexity can be reduced to $\mathcal{O}(\log L)$ which has led to an efficient real-time implementation [23].

Another very popular subclass of second-order BSS algorithms, particularly for instantaneous mixtures, is based on a cost function using the Frobenius norm $\|\mathbf{A}\|_{\mathrm{F}}^2 = \sum_{i,j} a_{ij}^2$ of a matrix $\mathbf{A} = (a_{ij})$, e.g., [3], [4], [9]–[12], [35]. Analogously to (18), this approach may be generalized for convolutive mixtures to

$$\mathcal{J}_{\mathrm{F}}(m) = \sum_{i=0}^{\infty}\beta(i,m)\left\|\mathbf{Y}^H(i)\mathbf{Y}(i) - \mathrm{bdiag}\mathbf{Y}^H(i)\mathbf{Y}(i)\right\|_{\mathrm{F}}^2 \tag{33}$$

which leads (after taking the gradient w.r.t. $\mathbf{W}$ in a similar way as shown in the Appendix) to the following update equation:

$$\Delta\mathbf{W}(m) = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{R}_{\mathbf{xx}}\mathbf{W}\{\mathbf{R}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{R}_{\mathbf{yy}}\} = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{R}_{\mathbf{xy}}\{\mathbf{R}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{R}_{\mathbf{yy}}\}. \quad (34)$$

We see that this update equation differs from the more general (22) only by the missing inherent normalization expressed by the inverse matrices $\mathbf{R}_{\mathbf{yy}}^{-1}$ and $\mathrm{bdiag}^{-1}\,\mathbf{R}_{\mathbf{yy}}$. Thus, (34) can be regarded as an analogon to the least mean square (LMS) algorithm [20] in supervised adaptive filtering. However, many simulation results have shown that for large filter lengths L, (34) is prone to instability, while (22) or (26) show a very robust convergence behavior (see Section VI) even for L on the order of hundreds or thousands of filter coefficients in BSS for real acoustic environments.

III. Internal Permutation

Generally, in BSS we distinguish between the noncritical *external* permutation ambiguity among the output channels and the *internal* permutation ambiguity in each frequency bin. While the external permutation ambiguity can be easily solved by incorporating prior geometrical knowledge of the array geometry, i.e., by source *localization*, the internal permutation ambiguity may severely limit the separation performance in frequency-domain algorithms.

A. Broadband Versus Narrowband Cost Function

Although it is widely believed in the literature that the internal permutation may only arise with narrowband frequency-domain criteria, recently some authors have claimed that to a certain degree the same internal permutation in frequency bins may arise in time-domain, i.e., also with broadband criteria [17], [36]. To the authors' knowledge, there are only a few theoretical analyses and little practical evidence for this claim [36]. However, a rigorous proof that this phenomenon does not exist seems difficult to derive.

In [36], a first attempt to analyze the internal permutation problem more closely is documented. Unfortunately, some assumptions were made which lead to discrepancies between the actually observed performance of BSS algorithms in practice and that analysis. In particular, exact ensemble averages or infinitely long block intervals are assumed for the estimation of correlations in the cost function. During these long intervals the signals are assumed to be wide-sense stationary so that adjacent frequency bins are becoming completely independent from each other in that analysis. Unfortunately, these assumptions are well-known to be unsuitable for SOS-based BSS, which requires nonstationarity, e.g., [3]. On the other hand, the assumption of independence between adjacent frequency components is on par with the narrowband approach, and thus is not suitable to explain the behavior of broadband algorithms.

It is known that frequency permutation ambiguities in blind algorithms without prior geometrical information can only be resolved by taking into account different frequency bins simultaneously (e.g., [37]). Based on the interfrequency correlations (see e.g., [38, p. 45] for a convincing illustration), several different approaches have been proposed in the literature. In [38] and [39] narrowband cost functions have been complemented with these correlations to solve the permutation problem. Moreover, these across-frequency correlations are also used in almost all supplementary repair mechanisms for narrowband algorithms, e.g., [10], [40]. A simple smoothness constraint on the filter coefficients has also been proposed [11], [41]. In Section IV below we provide without stationarity assumption (covariance method) a clear relationship between broadband and narrowband algorithms. The resulting generic broadband frequency-domain algorithm inherently includes the inter-frequency correlations and also a generalization of the smoothness constraint and thus, accounts for the internal permutation.

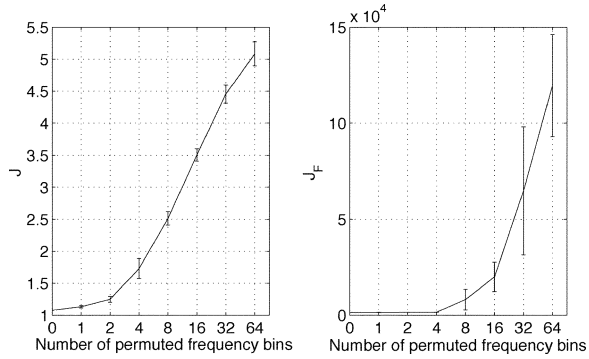


Fig. 5. Effect of permutations on the cost functions.

B. Discussion on Broadband Cost Function

In the following we discuss how well internal permutations can be prevented within the different broadband criteria (18) and (33).

As shown in Section II-E the cost function (18) has the desirable property of an inherent normalization resulting in a fast and robust convergence behavior of the corresponding update. In this regard (18) is clearly superior to the cost function (33) based on the Frobenius norm.

We compare the two broadband criteria $\mathcal{J}$ and $\mathcal{J}_{\mathrm{F}}$ with respect to the internal permutation problem experimentally. For this evaluation we have computed the cost functions for two independent speech signals with a certain percentage of internal permutations. Ideally, $\mathcal{J}$ and $\mathcal{J}_{\mathrm{F}}$ should exhibit a global minimum for 0% permutations and be strictly monotonic decreasing toward small percentages of permutations. For each percentage of permutations we performed 20 Monte Carlo trials in which the permuted frequency bins were randomly selected. The signal length was 10 s at a sampling rate of 16 kHz. The speech signals were segmented into blocks of length $N = 256$ for the estimation of the correlation matrices ($D = 128$ lags). Concerning the weighting function β, we averaged the results obtained from these blocks, which correspond to an offline adaptation described in Section V-A. Fig. 5 underlines the superior behavior of the proposed cost function $\mathcal{J}$. The Frobenius-based criterion $\mathcal{J}_{\mathrm{F}}$ in the right subplot of Fig. 5 shows relatively large confidence intervals and when only a few frequency bins are

permuted the cost function is already close to zero, so that an optimum solution of the internal permutation is not guaranteed with this criterion. In contrast, the proposed criterion $\mathcal{J}$ in the left subplot of Fig. 5 exhibits small confidence intervals and is strictly monotonic decreasing toward smaller percentages of internal permutations. This confirms the high robustness of the generic algorithm against frequency-domain permutations.

IV. Generic Frequency-Domain BSS

Based on the matrix formulation in the time domain (Section II), the following derivation of broadband frequency-domain algorithms shows explicitly the relation between time-domain and many well-known frequency-domain algorithms, as well as some extensions. Moreover, from a link to [14], it becomes clear that the broadband cost function (18) also leads to the very desirable property of an inherent stepsize normalization in the frequency domain. As pointed out in Section II, the conditions for the paramters L, N, and D for the natural gradient adaptation are given by the relations $N \geq D$ and $1 \leq D \leq L$. Therefore, we may assume $N = L$ without loss of generality for the following derivation.

A. General Frequency-Domain Formulation

The matrix formulation introduced for the time-domain in Section II allows a rigorous derivation of the corresponding frequency-domain BSS algorithms. In the frequency domain, the structure of the algorithm depends on the method for estimating the correlation matrices. Here, we consider again the more accurate covariance method [26]. The matrices $\mathbf{X}_p(m)$ and $\mathbf{W}_{pq}$ are now diagonalized in two steps. We first consider the $L \times 2L$ Toeplitz matrices $\mathbf{X}_p(m)$.

Step 1: Transformation of Toeplitz Matrices Into Circulant Matrices

Any Toeplitz matrix $\mathbf{X}_p$ can be transformed, by doubling its size, to a circulant matrix $\mathbf{C}_{X_p}(m)$ [19]. In our case we define the circulant matrix by taking into account (10) by

$$\mathbf{C}_{X_p}(m) = \begin{bmatrix} \mathbf{X}'_p(m-3) & \mathbf{X}_p(m-1) \\ \mathbf{X}_p(m-2) & \mathbf{X}_p(m) \\ \mathbf{X}_p(m-1) & \mathbf{X}'_p(m-3) \\ \mathbf{X}_p(m) & \mathbf{X}_p(m-2) \end{bmatrix} \tag{35}$$

where $\mathbf{X}'_p(m-3)$ is a properly chosen extension ensuring a circular shift of the $4L$ input values in the first column. It follows

$$\mathbf{X}_p(m) = \mathbf{W}^{01_L}_{L\times 4L}\mathbf{C}_{X_p}(m)\mathbf{W}^{1_{2L}0}_{4L\times 2L} \tag{36}$$

where we introduced the windowing matrices

$$\begin{aligned} \mathbf{W}^{01_L}_{L\times 4L} &= [\mathbf{0}_{L\times 3L}, \mathbf{I}_{L\times L}] \\ \mathbf{W}^{1_{2L}0}_{4L\times 2L} &= [\mathbf{I}_{2L\times 2L}, \mathbf{0}_{2L\times 2L}]^T. \end{aligned}$$

Here, we use the following conventions for the windowing matrices in the frequency domain:

- The lower index of a matrix denotes its dimensions.
- The upper index describes the positions of ones and zeros. Unity submatrices are always located at the upper left ("10") or lower right ("01") corners of the respective matrix. The size of these clusters is indicated in subscript (e.g., "01_L").

Step 2: Transformation of the Circulant Matrices Into Diagonal Matrices

Using the $4L \times 4L$ DFT matrix $\mathbf{F}_{4L\times 4L}$, the circulant matrices are diagonalized as follows:

$$\mathbf{C}_{X_p}(m) = \mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{X}}_p(m)\mathbf{F}_{4L\times 4L} \tag{37}$$

where the diagonal matrices $\underline{\mathbf{X}}_p(m)$ representing the frequency-domain versions of $\mathbf{X}_p(m)$, can be expressed by the first columns of $\mathbf{C}_{X_p}(m)$

$$\begin{aligned} \underline{\mathbf{X}}_p(m) = \operatorname{diag}\{\mathbf{F}_{4L\times 4L}\,[&x_p(mL-3L),\ldots,x_p(mL-1) \\ &x_p(mL), x_p(mL+1),\ldots,x_p(mL+L-1)]^T\Big\} \end{aligned} \tag{38}$$

i.e., to obtain $\underline{\mathbf{X}}_p(m)$, we transform the concatenated vectors of the current block and three previous blocks of the input signals $x_p(m)$. Here, $\operatorname{diag}\{\mathbf{a}\}$ denotes a square matrix with the elements of vector $\mathbf{a}$ on its main diagonal. Now, (36) can be rewritten equivalently as

$$\mathbf{X}_p(m) = \mathbf{W}^{01_L}_{L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{X}}_p(m)\mathbf{F}_{4L\times 4L}\mathbf{F}^{1_{2L}0}_{4L\times 2L}. \tag{39}$$

Equations (39) and (38) exhibit a form that is structurally similar to that of the corresponding counterparts of the well-known (supervised) frequency-domain adaptive filters [19]. However, the major difference here is that we need a transformation length of at least $4L$ instead of $2L$. This should come as no surprise, since in BSS using the covariance method, both convolution and correlation is carried out where both operations double the transformation length.

We now transform the matrices $\mathbf{W}_{pq}$ in the same way as shown above for $\mathbf{X}_p$. Thereby, we obtain

$$\mathbf{W}_{pq} = \mathbf{W}^{1_{2L}0}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{W}}_{pq}\mathbf{F}_{4L\times 4L}\mathbf{W}^{1_D0}_{4L\times D} \tag{40}$$

where

$$\begin{aligned} \mathbf{W}^{1_D0}_{4L\times D} &= \left[\mathbf{I}_{D\times D}, \mathbf{0}_{D\times(4L-D)}\right]^T \\ \mathbf{W}^{1_{2L}0}_{2L\times 4L} &= [\mathbf{I}_{2L\times 2L}, \mathbf{0}_{2L\times 2L}] = \left(\mathbf{W}^{1_{2L}0}_{4L\times 2L}\right)^T \end{aligned}$$

and the frequency-domain representation of the demixing matrix

$$\underline{\mathbf{W}}_{pq} = \operatorname{diag}\left\{\mathbf{F}_{4L\times 4L}\,[w_{pq,0},\ldots,w_{pq,L-1},0,\ldots,0]^T\right\}. \tag{41}$$

Equation (40) is illustrated in Fig. 6. Note that the column vector in (41) corresponds to the first column of the $4L \times 4L$ matrix in Fig. 6. Moreover, it can be seen that the premultiplied transformation $\mathbf{W}^{10}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}$ in (40) is related to the demixing

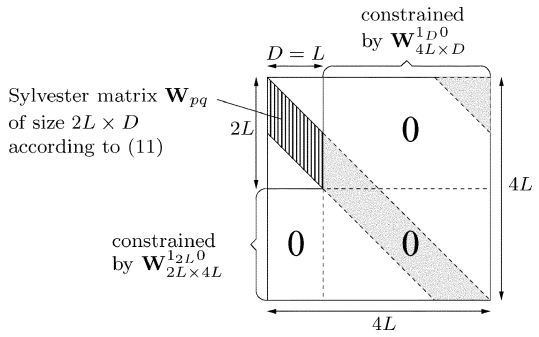


Fig. 6. Illustration of $4L \times 4L$ circulant matrix in (40).

filter taps in the first column of $\mathbf{W}_{pq}$, while the postmultiplied transformation in (40), which we denote by

$$\mathbf{L}_{4L\times D}^{1_D 0} = \mathbf{F}_{4L\times 4L}\mathbf{W}_{4L\times D}^{1_D 0} \tag{42}$$

is related to the introduction of D time-lags (see also Section IV-C). Combining all channels, we obtain

$$\begin{aligned}\mathbf{X}(m) =& \mathbf{W}_{L\times 4L}^{01_L}\mathbf{F}_{4L\times 4L}^{-1}\underline{\mathbf{X}}(m)\\ &\cdot \mathrm{bdiag}\left\{\mathbf{F}_{4L\times 4L}\mathbf{W}_{4L\times 2L}^{1_{2L}0}\right.\\ &\qquad \left.\ldots, \mathbf{F}_{4L\times 4L}\mathbf{W}_{4L\times 2L}^{1_{2L}0}\right\} \qquad (43)\\ \mathbf{W} =& \mathrm{bdiag}\left\{\mathbf{W}_{2L\times 4L}^{1_{2L}0}\mathbf{F}_{4L\times 4L}^{-1}\right.\\ &\qquad \left.\ldots, \mathbf{W}_{2L\times 4L}^{1_{2L}0}\mathbf{F}_{4L\times 4L}^{-1}\right\}\underline{\mathbf{W}}(m)\mathbf{L} \qquad (44)\end{aligned}$$

where $\mathrm{bdiag}\{\mathbf{A}_1, \ldots, \mathbf{A}_P\}$ denotes a block-diagonal matrix with submatrices $\mathbf{A}_1, \ldots, \mathbf{A}_P$ on its diagonal, $\underline{\mathbf{X}}(m)$ and $\underline{\mathbf{W}}(m)$ are defined analogously to (14) and (15), respectively. $\mathbf{L}$ denotes the $4LP \times DP$ matrix

$$\mathbf{L} = \mathrm{bdiag}\left\{\mathbf{L}_{4L\times D}^{1_D 0}, \ldots, \mathbf{L}_{4L\times D}^{1_D 0}\right\}. \tag{45}$$

From (12), (43), and (44) we further obtain

$$\mathbf{Y} = \mathbf{W}_{L\times 4L}^{01_L}\mathbf{F}_{4L\times 4L}^{-1}\underline{\mathbf{X}}(m)\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}(m)\mathbf{L} \tag{46}$$

where

$$\begin{aligned}\mathbf{G}_{4LP\times 4LP}^{1_{2L}0} &= \mathrm{bdiag}\left\{\mathbf{G}_{4L\times 4L}^{1_{2L}0}, \ldots, \mathbf{G}_{4L\times 4L}^{1_{2L}0}\right\}\\ \mathbf{G}_{4L\times 4L}^{1_{2L}0} &= \mathbf{F}_{4L\times 4L}\mathbf{W}_{4L\times 4L}^{1_{2L}0}\mathbf{F}_{4L\times 4L}^{-1}\\ \mathbf{W}_{4L\times 4L}^{1_{2L}0} &= \mathbf{W}_{4L\times 2L}^{1_{2L}0}\mathbf{W}_{2L\times 4L}^{1_{2L}0}\\ &= \begin{bmatrix}\mathbf{I}_{2L\times 2L} & \mathbf{0}_{2L\times 2L}\\ \mathbf{0}_{2L\times 2L} & \mathbf{0}_{2L\times 2L}\end{bmatrix}.\end{aligned}$$

To formulate the cost function (18) equivalently in the frequency domain, we now need to calculate the short-time correlation matrix using (46), i.e.,

$$\begin{aligned}\mathbf{R}_{\mathbf{yy}} &= \mathbf{Y}^H\mathbf{Y}\\ &= \mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L}\end{aligned} \tag{47}$$

where

$$\begin{aligned}\mathbf{S}_{\mathbf{yy}} &= \underline{\mathbf{W}}^H\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\mathbf{S}_{\mathbf{xx}}\cdot\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}} \qquad (48)\\ \mathbf{S}_{\mathbf{xx}} &= \underline{\mathbf{X}}^H\mathbf{G}_{4L\times 4L}^{01_L}\underline{\mathbf{X}}\end{aligned}$$

$$\begin{aligned}\mathbf{G}_{4L\times 4L}^{01_L} &= \mathbf{F}_{4L\times 4L}\mathbf{W}_{4L\times 4L}^{01_L}\mathbf{F}_{4L\times 4L}^{-1}\\ \mathbf{W}_{4L\times 4L}^{01_L} &= \mathbf{W}_{4L\times L}^{01_L}\mathbf{W}_{L\times 4L}^{01_L}\\ &= \begin{bmatrix}\mathbf{0}_{3L\times 3L} & \mathbf{0}_{3L\times L}\\ \mathbf{0}_{L\times 3L} & \mathbf{I}_{L\times L}\end{bmatrix}.\end{aligned} \tag{49}$$

The gradient of the cost function in the frequency domain is now derived in a similar way as in the time domain. Since $\mathbf{Y}^H\mathbf{Y} = \mathbf{L}^H\underline{\mathbf{W}}^H(\mathbf{G}_{4LP\times 4LP}^{1_{2L}0})^H\mathbf{S}_{\mathbf{xx}}\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}\mathbf{L}$, we obtain using the chain rule for matrices $\nabla_{\underline{\mathbf{W}}}\mathcal{J} = \nabla_{(\underline{\mathbf{W}}\mathbf{L})}\mathcal{J}\cdot\mathbf{L}^H$, and thus, with expressions (20) and (21)

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}\mathcal{J}(m) =& 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\mathbf{S}_{\mathbf{xy}}\mathbf{L}\\ &\cdot\left\{\mathrm{bdiag}^{-1}(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L}) - (\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})^{-1}\right\}\mathbf{L}^H\\ =& 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\mathbf{S}_{\mathbf{xy}}\mathbf{L}\\ &\cdot(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})^{-1}\mathbf{L}^H\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{S}_{\mathbf{yy}}\}\mathbf{L}\\ &\cdot\mathrm{bdiag}^{-1}(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})\cdot\mathbf{L}^H\end{aligned} \tag{50}$$

where

$$\mathbf{S}_{\mathbf{xy}} = \mathbf{S}_{\mathbf{xx}}\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}. \tag{51}$$

Equation (50) is the generic frequency-domain analogon to (22) and may be equivalently used for coefficient adaptation.

As in the time domain, where $(\mathcal{SC})$ imposes the Sylvester structure, we need to ensure a diagonal structure of the submatrices $\underline{\mathbf{W}}_{pq}$ in the frequency domain. While the structure of matrix $\underline{\mathbf{W}}$ is independent of D, matrix $\mathbf{L}$ introduces the number of time-lags taken into account by the cost function, as shown by (42) and (46) (see also Fig. 6). To calculate the separated output signals, given a demixing matrix $\underline{\mathbf{W}}$, we need to pick the first column of $\mathbf{Y}$ in (46) (the other columns were introduced in (9) for including multiple time-lags in the cost function). This is done by using $\mathbf{L} = \mathbf{L}_{\mathrm{I}} = \mathrm{bdiag}\{\mathbf{1}_{4L\times 1}, \cdots, \mathbf{1}_{4L\times 1}\}$ in (46). Then, $\underline{\mathbf{W}}\mathbf{L}$ in that equation becomes a $4LP \times P$ matrix $\underline{\mathbf{W}}'$ whose columns correspond to the diagonals of $\underline{\mathbf{W}}$. As a general rule

$$\underline{\mathbf{W}}' = \underline{\mathbf{W}}\mathbf{L}_I \tag{52}$$

and building diagonal submatrices $\underline{\mathbf{W}}_{pq}$ of $\underline{\mathbf{W}}$ using the entries of $\underline{\mathbf{W}}'$, transforms the two equivalent representations into each other. Thus, to formally obtain the update of $\underline{\mathbf{W}}'$ needed for the output signal calculation, we post-multiply (50) by $\mathbf{L}_{\mathrm{I}}$, both simplifying the calculation of (50), and enforcing the diagonal structure of $\underline{\mathbf{W}}_{pq}$ during the adaptation.

In addition to this diagonal structure, we have to ensure the Sylvester structure in the time domain as noted previously. As can be seen in Fig. 6, (41) determines the first column, and thus the whole $4L\times 4L$ Sylvester matrix. In other words, we have to ensure that the time-domain column vector in (41) contains only L filter coefficients and $3L$ zeros. Therefore, we have to tighten the constraint $\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}$ appearing in (50) to $\mathbf{G}_{4LP\times 4LP}^{1_L 0}$. Together with (52) this leads to

$$\Delta\underline{\mathbf{W}}' = \mathbf{G}_{4LP\times 4LP}^{1_L 0}\nabla_{\underline{\mathbf{W}}}\mathcal{J}\mathbf{L}_I. \tag{53}$$

B. Natural Gradient in the Frequency Domain

In Section II-C, it has been shown that the natural gradient for convolutive mixtures introduced there for the time domain yields equivariant adaptation algorithms, i.e., the evolutionary behavior of

$$\mathbf{C}(m) = \mathbf{H}\mathbf{W}(m) \tag{54}$$

and $\Delta\mathbf{C}(m) = \mathbf{H}\Delta\mathbf{W}(m)$ does not explicitly depend on $\mathbf{H}$ in (27).

In this section, we investigate how this formulation of the natural gradient transforms into the frequency domain. To begin with, we start by the following approach containing arbitrary matrices $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$, and $\mathbf{A}_4$:

$$\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J} = \mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_3\underline{\mathbf{W}}^H\mathbf{A}_4\nabla_{\underline{\mathbf{W}}}\mathcal{J}. \tag{55}$$

Now, our task is to determine the four matrices $\mathbf{A}_i$ such that the resulting coefficient update exhibits desired properties.

As a first condition, matrix $\mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_3\underline{\mathbf{W}}^H\mathbf{A}_4$ in (55) must be positive definite, i.e., all its eigenvalues must be positive to ensure convergence. This determines matrices $\mathbf{A}_3$ and $\mathbf{A}_4$ so that we obtain

$$\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J} = \mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_2^H\underline{\mathbf{W}}^H\mathbf{A}_1^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}. \tag{56}$$

As the second, and most important condition, it is required that the equivariance property is fulfilled. Combining (54) with (44), we obtain a relation between $\mathbf{C}$ and the frequency-domain coefficients $\underline{\mathbf{W}}$

$$\mathbf{C} = \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\underline{\mathbf{W}}\mathbf{L} \tag{57}$$

and analogously

$$\begin{aligned}\Delta\mathbf{C} &= \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\Delta\underline{\mathbf{W}}\mathbf{L}\\ &= \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J}\mathbf{L}.\end{aligned} \tag{58}$$

As in the time domain [see (27)], it is required that (58) in combination with the natural gradient (56) can be expressed by $\mathbf{C}$ defined in (57), and therefore, does not explicitly depend on $\mathbf{H}$. This leads to the claim

$$\Delta\mathbf{C} = \underbrace{\mathbf{H}\,\mathrm{bdiag}\{\cdots\}\mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2}_{=\mathbf{C}}\mathbf{A}_2^H\underline{\mathbf{W}}^H\mathbf{A}_1^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}\mathbf{L} \tag{59}$$

and a comparison of (59) with (57) yields the matrices

$$\mathbf{A}_1 = \mathbf{G}_{4LP\times 4LP}^{1_{2L}0},\ \mathbf{A}_2 = \mathbf{L}. \tag{60}$$

Note that $\mathbf{A}_1 = \mathbf{I}$ is not the general solution. This can be verified by inserting (60) in (59), and considering the argument of $\mathrm{bdiag}\{\cdot\}$ according to (44). Finally, we obtain the natural gradient

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J} &= \mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\underline{\mathbf{W}}^H\left(\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\right)^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}\\ &= \mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}\mathbf{G}_{4LP\times 4LP}^{1_D0}\underline{\mathbf{W}}^H\\ &\quad\cdot\left(\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\right)^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}\end{aligned} \tag{61}$$

and together with (48), (50), and (51) it follows the coefficient update

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J} = 2\sum_{i=0}^{\infty}\beta(i,m)&\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\\ &\cdot\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{S}_{\mathbf{yy}}\}\mathbf{L}\cdot\mathrm{bdiag}^{-1}(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})\mathbf{L}^H.\end{aligned} \tag{62}$$

Note that this equation is the broadband frequency-domain analogon to the natural gradient (26) and shows again the convenient property of avoiding one matrix inversion. Formally, as in Section IV-A, (53) can be used to obtain $\Delta\underline{\mathbf{W}}'$.

C. Constraints and the Internal Permutation Problem in Narrowband Frequency-Domain BSS

Two types of constraints appear in the gradient (50) and in the natural gradient update (62):

- The matrices $\mathbf{G}_{\cdots}^{\vdots}$ in (48), (49), (51), and in the update equations are mainly responsible for preventing decoupling of the individual frequency components, and thus avoid the internal permutation among the different frequency bins by introducing interfrequency components (as noted in Section III) into the spectral density matrices.
- Matrix $\mathbf{L}$ has two different functions: on the one hand, it allows joint diagonalization over D time-lags, and on the other hand, it acts as time-domain constraint similar to the matrices $\mathbf{G}_{\cdots}^{\vdots}$ (see Fig. 6).

Concerning matrix $\mathbf{L}$ we can distinguish between four different cases:

1) $D < L$: As in the time domain, this choice allows the exploitation of the nonwhiteness property with up to D time-lags.
2) $D = L$: This is the optimum case as in the time domain.
3) $D > L$: This choice is not meaningful in the time domain. In the frequency domain, however, we can choose D up to the transformation length $4L$ due to the introduced circulant matrix, as shown in Fig. 6. Then, the time-domain constraint is relaxed, which generally leads to a suboptimum solution.
4) $D = 4L$: According to Fig. 6 this corresponds to the traditional narrowband approximation (apart from constraints $\mathbf{G}_{\cdots}^{\vdots}$) so that all matrices $\mathbf{L}$ cancel out in the update equations, which can also be verified using (42).

In Case 4), i.e., by neglecting matrix $\mathbf{L}$ in (50) we obtain a simplified gradient

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}\mathcal{J} = 2\sum_{i=0}^{\infty}\beta(i,m)&\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\mathbf{S}_{\mathbf{xy}}\mathbf{S}_{\mathbf{yy}}^{-1}\\ &\cdot\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{S}_{\mathbf{yy}}\}\mathrm{bdiag}^{-1}\mathbf{S}_{\mathbf{yy}}\end{aligned} \tag{63}$$

and a simplified natural gradient

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J} = 2\sum_{i=0}^{\infty}\beta(i,m)&\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{W}}\\ &\cdot\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\mathbf{S}_{\mathbf{yy}}\}\mathrm{bdiag}^{-1}\mathbf{S}_{\mathbf{yy}}.\end{aligned} \tag{64}$$

Note that these expressions still largely avoid the internal permutation problem of narrowband frequency-domain BSS using

the constraints $\mathbf{G}_{\cdots}^{\vdots}$ in $\mathbf{S}_{\mathbf{xy}}$, $\mathbf{S}_{\mathbf{yy}}$, and in the update equations obtained from inserting (63) or (64) into (53). This is similar to the across-frequency processing in [38].

By additionally approximating $\mathbf{G}_{\cdots}^{\vdots}$ as scaled identity matrices [19] in the gradients, all the submatrices in (63) and (64) become diagonal matrices. This approximation combined with $D = 4L$ [see Case 4)] corresponds to the traditional narrowband approach. Only in this case both equations can be decomposed in its frequency components, i.e., we can equivalently write

$$\nabla_{\underline{\mathbf{W}}}\mathcal{J}^{(\nu)} = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{S}_{\mathbf{xy}}^{(\nu)}\left(\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right)^{-1} \cdot\left\{\mathbf{S}_{\mathbf{yy}}^{(\nu)} - \mathrm{diag}\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right\}\mathrm{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)} \quad (65)$$

and

$$\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J}^{(\nu)} = 2\sum_{i=0}^{\infty}\beta(i,m)\underline{\mathbf{W}}^{(\nu)} \cdot\left\{\mathbf{S}_{\mathbf{yy}}^{(\nu)} - \mathrm{diag}\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right\}\mathrm{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)} \quad (66)$$

respectively, where $\nu = 0,\ldots,4L-1$ denotes the frequency bins. In contrast to $\mathbf{S}_{\mathbf{xy}}$, $\mathbf{S}_{\mathbf{yy}}$, and $\underline{\mathbf{W}}$ in (63), (64) which are $4LP \times 4LP$ matrices each, the corresponding matrices $\mathbf{S}_{\mathbf{xy}}^{(\nu)}$, $\mathbf{S}_{\mathbf{yy}}^{(\nu)}$, and $\underline{\mathbf{W}}^{(\nu)}$ in (65), (66) are only of dimension $P \times P$.

To obtain the updated equations from the approximated gradients, we again apply (53) which contains another constraint $\mathbf{G}_{4LP\times 4LP}^{1_L 0}$ transforming the filter coefficients back in the time domain, zeroing the last $3L$ values, and transforming the result back to the frequency domain. Thus, even if (65) and (66) can be efficiently computed in a bin-selective manner, this constraint prevents a complete decoupling of the frequency-components in the update equations. This procedure appears similarly in the well-known "constrained frequency-domain adaptive filtering" in the supervised case [19], [20]. In BSS, this theoretically founded mechanism largely eliminates the permutation problem in a simple way. It was first heuristically introduced in [41], and also in [11]. A more detailed experimental examination on this constraint was reported in [17] confirming that our theoretically obtained ratio between filter length L and transformation length $4L$ yields optimum separation performance. However, due to the omission of the other constraints in the approximated gradients one will not achieve a perfect solution of the permutation ambiguity as observed experimentally in [17]. Thus, the generic algorithm derivation provides a tradeoff between computational complexity and internal permutation ambiguity, depending on the number of implemented constraints. Traditional narrowband approaches also neglecting the time-domain constraint in (53) require additional measures for solving the permutation problem (e.g., [10], [40]). Note that most of these methods also exploit interfrequency correlations in some way. However, in contrast to the global optimization within the cost function, this subsequent ordering is done using local decisions.

D. Some Links to Known Narrowband Frequency-Domain Algorithms and the Generalized Coherence

The approximated coefficient update (65) is directly related to some well-known narrowband frequency-domain BSS algorithms. In [13], an algorithm that is similar to (65) was derived by directly optimizing a cost function similar to the one in [8] in a bin-wise manner. More recently, it was proposed in [14] to apply the magnitude-squared coherence $|\gamma_{y_p y_q}^{(\nu)}(m)|^2$, $p, q \in \{1,2\}$ as a cost function for narrowband frequency-domain BSS. The coherence has the very desirable property that $0 \leq |\gamma_{y_p y_q}^{(\nu)}(m)|^2 \leq 1$, which directly translates into an inherent stepsize normalization of the corresponding update equation [14]. In particular $|\gamma_{y_1 y_2}^{(\nu)}(m)|^2 = 0$ if $\mathbf{y}_1$ and $\mathbf{y}_2$ are orthogonal, and $|\gamma_{y_1 y_2}^{(\nu)}(m)|^2 = 1$ if $\mathbf{y}_1$ and $\mathbf{y}_2$ are linearly related.

Comparing the update (65) with that derived in [14], we see that an additional approximation of $(\mathbf{S}_{\mathbf{yy}}^{(\nu)})^{-1}$ as a diagonal matrix was used in [14], which results in

$$\nabla_{\underline{\mathbf{W}}}\mathcal{J}^{(\nu)}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\mathbf{S}_{\mathbf{xy}}^{(\nu)}\mathrm{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)} \cdot\left\{\mathbf{S}_{\mathbf{yy}}^{(\nu)} - \mathrm{diag}\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right\}\mathrm{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)}. \quad (67)$$

The coherence function applied in [14] can be extended to the case $P > 2$ by using the so-called generalized coherence [42]. In [18] a link between the cost function (18) and the generalized coherence was established. This relationship allows a geometric interpretation of (18) and shows that this cost function leads to an inherent stepsize normalization for the coefficient updates.

V. Weighting Function

In the generalized cost function (18), we introduced a weighting function $\beta(i,m)$ with the block time indices i, m to allow different realizations of our algorithms. Based on the cost function we previously derived stochastic and natural gradient update equations in the time domain (22), (26) and in the frequency domain (50), (62), respectively. Due to the similar structure of these equations, we will now consider only the time domain for simplicity. Thus we can express (22) and (26) as

$$\Delta\mathbf{W}(m) = \sum_{i=0}^{\infty}\beta(i,m)\mathcal{Q}(i) \quad (68)$$

where $\mathcal{Q}(i)$ denotes the corresponding update term of the ith block. In the following we distinguish three different types of weighting functions $\beta(i,m)$ for offline, online, and block-online realizations [21]. The weighting functions have a finite support, and are normalized such that $\sum_{i=0}^{\infty}\beta(i,m) = 1$.

A. Offline Implementation

When realizing the algorithm as an offline or so-called batch algorithm then $\beta(i,m)$ corresponds to a rectangular window, which is described by $\beta(i,m) = (1/K_{\mathrm{sig}})\epsilon_{0,(K_{\mathrm{sig}}-1)}(i)$, where $\epsilon_{a,b}(i) = 1$ for $a \leq i \leq b$, and $\epsilon_{a,b}(i) = 0$ elsewhere.

The entire signal is segmented into K_{sig} blocks, and then the entire signal is processed to estimate the demixing matrix $\mathbf{W}^j$ where the superscript j denotes the current iteration of the coefficient update

$$\mathbf{W}^j = \mathbf{W}^{j-1} - \frac{\mu}{K_{\mathrm{sig}}}\sum_{i=0}^{K_{\mathrm{sig}}-1}\mathcal{Q}(i). \quad (69)$$

Hence, the algorithm is visiting the signal data repeatedly for each iteration j and therefore, it usually achieves a better performance compared to its online counterpart.

B. Online Implementation

In time-varying environments, an online implementation of (68) is required. An efficient realization can be achieved by using a weighting function with an exponential forgetting factor λ. It is defined by

$$\beta(i,m) = (1-\lambda)\lambda^{m-i}\epsilon_{0,m}(i) \tag{70}$$

where $0 \leq \lambda < 1$. Thus, (68) reads

$$\Delta\mathbf{W}(m) = (1-\lambda)\sum_{i=0}^{m}\lambda^{m-i}\mathcal{Q}(i) \tag{71}$$

where m denotes the current block. Additionally, (71) can be formulated recursively to reduce computational complexity and memory requirements since only the preceding demixing matrix has to be saved for the update [21].

C. Block-online Implementation

Similar to the approach in [43] we can combine the online and offline approach in a so-called block-online method. After obtaining K blocks of length N we process an offline algorithm with $j_{\max}$ iterations. The demixing filter matrix $\mathbf{W}$ of the current block is then used as initial value for the offline algorithm of the next block. This block-online approach allows a tradeoff between computational complexity on the one hand and separation performance and speed of convergence on the other hand by adjusting the maximum number of iterations $j_{\max}$ as we will see in Section VI. See also [23] for more details.

VI. Experiments and Results

A. Experimental Conditions

The experiments have been conducted using speech data convolved with measured impulse responses of a real room ($580\ \mathrm{cm} \times 590\ \mathrm{cm} \times 310\ \mathrm{cm}$), with a reverberation time $T_{60} = 150\ \mathrm{ms}$ and a sampling frequency of $f_s = 16\ \mathrm{kHz}$. A two-element microphone array with an inter-element spacing of 16 cm was used for the recording. The speech signals arrived from two different directions, $-45°$ and $45°$. Sentences spoken by two male speakers (from the TIMIT speech corpus) were selected as source signals. The length of the source signals was 10 s. To evaluate the performance, the signal-to-interference ratio (SIR) was used which is defined as the ratio of the signal power of the target signal to the signal power from the jammer signal. For offline implementations the SIR was calculated over the entire signal length whereas for online implementations it was continuously calculated for each block. In the following the SIR is averaged over both channels.

For choosing the demixing filter length L it can be shown by considering matrix dimensions of $\mathbf{C} = \mathbf{HW}$ according to [1] that for deconvolution, ideally, $L \geq (Q/(P-Q))(M-1)$, where P and Q are the number of sensors and sources, respectively. M denotes the length of the mixing system. Analogous considerations for BSS under the condition $\mathrm{boff}\{\mathbf{C}\} = \mathbf{0}$ due to the filtering ambiguity yield $L \geq ((Q-1)/(P-Q+1))(M-1)$. For the special case $P = Q = 2$ considered here as an example, this gives $L \geq M - 1 \approx M$ as in system identification. There an estimate for the demixing filter length L needed to model the room impulse reponses for a given accuracy of a dB is given by $L = (f_s \cdot T_{60} \cdot a\,\mathrm{dB})/(60\,\mathrm{dB})$. Experimental results confirmed that this can be carried over to the BSS scenario yielding a rough estimate for L. Thus, for a desired maximum SIR of 25 dB we choose $L = 1024$ and the block length of $N = 2048$.

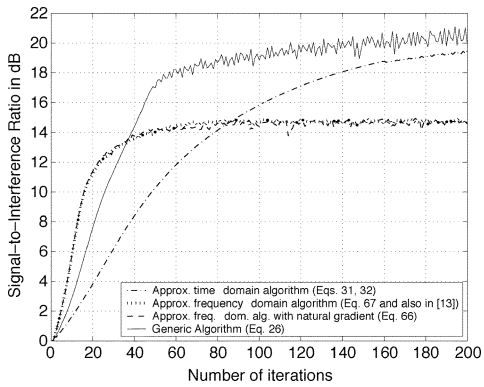


Fig. 7. Comparison of different offline realizations.

The Sylvester constraint ($\mathcal{SC}$) is realized here by picking the first column as discussed in Section II-B. Concerning the initialization of $\mathbf{W}$ it can be shown using (11), (26) that by applying this specific ($\mathcal{SC}$) realization the first coefficients of the filters $\mathbf{W}_{pp}$ must be nonzero. Thus, we use a unit impulse for the first filter tap in each $\mathbf{W}_{pp}$. The filters $\mathbf{W}_{pq}$, $p \neq q$ are set to zero.

B. Experimental Results

In our experiments we compared offline and online realizations and we examined the effect of taking into account different numbers of time-lags D for the computation of the correlation function (17). Comparing different offline algorithms in Fig. 7 shows that the stochastic gradient (67) (dotted) and natural gradient (66) (dashed) narrowband approximations of the generic frequency-domain algorithm exhibit the fastest initial convergence. Note that the dotted curve corresponds to the well-known algorithm after [14] and may be seen as a reference for SOS-based BSS. This is mainly due to the decomposition of the update equation in its frequency components and hence we have an independent update in each frequency bin. The complete decoupling is prevented and therefore also the internal permutation problem is mostly resolved by considering the constraint $\mathbf{G}^{1_L 0}_{4LP\times 4LP}$ (53) similar as in [11], [41]. However, especially for a large filter length L, some frequency bins will still be permuted and thus the narrowband algorithms converge to a lower maximum SIR than their broadband counterparts.

The generic broadband algorithm (26) [or its equivalent frequency-domain formulation (62)] (solid) achieves nearly the same convergence speed as the narrowband frequency-domain algorithms. This confirms that time-domain algorithms can also exhibit a stable and robust convergence behavior for very long unmixing filters. However, in the generic time-domain algorithm this comes with an increased computational cost, as an inversion of a large matrix is required due to the RLS-like normal-

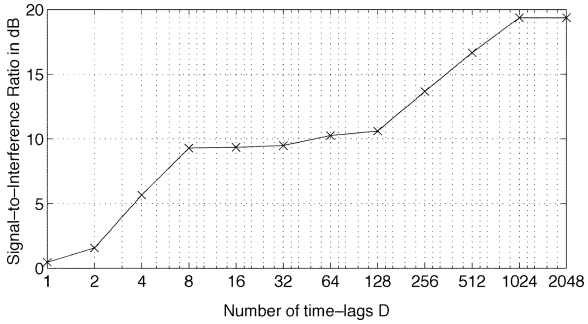


Fig. 8. Effect of taking into account different numbers of lags D.

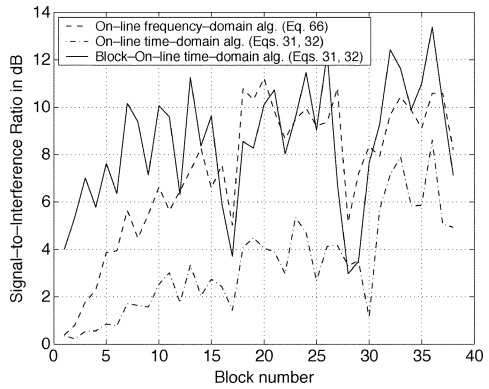


Fig. 9. Comparison of different online realizations.

ization (Section II-E). Moreover, similar to the RLS in supervised adaptive filtering the inverse matrices have to be regularized properly. The approximated version of the generic time-domain algorithm (dash-dotted) shows a slower convergence as the RLS-like normalization is replaced by a diagonal matrix which corresponds to an NLMS-like normalization (32). Note that the broadband algorithms exhibit a higher maximum SIR as they inherently avoid the internal permutation problem.

The relation between the number of lags D used for computing the correlation function $\mathbf{R_{yy}}$, i.e., the exploitation of nonwhiteness, and the SIR is shown in Fig. 8 for an example. As adaptation algorithm an offline version of the approximated time-domain algorithm (32) has been applied and 200 iterations have been computed.

When utilizing only nonstationarity $(D = 1)$, the sources cannot be separated. Including nonwhiteness $(D > 1)$ is especially beneficial for the first lags, which can be explained by the strong correlation of speech signals within the first few lags. By considering these temporal correlations, additional information about the mixtures is taken into account for the simultaneous diagonalization of $\mathbf{R_{yy}}$. A further increase of D up to $D = 1024$ is still improving the maximum SIR as the temporal correlation of the room impulse response is exploited in the adaptation. This confirms the theoretical finding in Section II-D that $D = L$ yields the optimum performance.

Different online realizations are shown in Fig. 9. Again, it can be seen that the approximated frequency-domain algorithm (dashed) exhibits superior initial convergence speed compared to the approximated time-domain algorithm (dash-dotted) due to its NLMS-like normalization. However, this effect can be mitigated by using a block-online adaptation (see Section V-C) of the approximated time-domain algorithm (solid), while in case of the approximated frequency-domain algorithm such improvement from online to block-online adaptation has not been found. For the simulation examples, $K = 8$, $L = 512$, $N = 512$, and $j_{\max} = 10$ iterations were chosen, as these settings also work very well in real-time on current regular PCs [23]. Experiments taking into account also diffuse background noise in a car environment can be found in [24].

VII. Conclusion

A generalization of second-order BSS algorithms for convolutive mixtures was presented. This generalization extends previous work in two directions. First, both, nonstationarity and nonwhiteness of the source signals are explicitly taken into account. Secondly, a general broadband formulation and optimization of a novel cost function was introduced. This approach allows rigorous derivations of both known and novel algorithms in time and frequency domain and provides a deeper understanding and solution of the internal permutation ambiguity appearing in traditional narrowband frequency-domain BSS. Experimental results have shown that this rigorous theoretical approach leads directly to superiour practical BSS algorithms for reverberant acoustic environments in both, time and frequency domains.

Appendix

Here, we prove the derivative

$$\frac{\partial \mathcal{J}_1}{\partial \mathbf{W}^*} = \mathbf{RW}(\mathrm{bdiag}\mathbf{W}^H\mathbf{RW})^{-1} \tag{72}$$

of the first term $\mathcal{J}_1$ in the cost function (18) defined by

$$\mathcal{J}_1 = \log\det \mathrm{bdiag}\mathbf{W}^H\mathbf{RW}. \tag{73}$$

We apply the chain rule for matrices expressed by its elements

$$\frac{\partial \mathcal{J}_1}{\partial (w_{mn}^{MN})^*} = \sum_{k,j,K,J} \frac{\partial \mathcal{J}_1}{\partial a_{kj}^{KJ}} \cdot \frac{\partial a_{kj}^{KJ}}{\partial (w_{mn}^{MN})^*}. \tag{74}$$

In this formulation, the upper indices denote the channel-selective submatrices, and the lower indices denote the elements of the respective submatrix, e.g., w_{mn}^{MN} is the mn-th element of the MN-th submatrix of $\mathbf{W}$. Due to the chain rule, we now first take the derivative of $\mathcal{J}_1$ w.r.t.

$$\mathbf{A} = \mathrm{bdiag}\mathbf{W}^H\mathbf{RW}. \tag{75}$$

For this we can use the known expression

$$\frac{\partial \log\det \mathbf{A}}{\partial \mathbf{A}} = (\mathbf{A}^T)^{-1} \tag{76}$$

for a real matrix $\mathbf{A}$ according to, e.g., [28], [29]. In our case, since we assume $\mathbf{R}^H = \mathbf{R}$, this equation further simplifies, as

$$(\mathbf{A}^T)^{-1} = (\mathbf{A}^H)^{-1} = (\mathrm{bdiag}\mathbf{W}^H\mathbf{RW})^{-1} = \mathbf{A}^{-1}. \tag{77}$$

For the second part in (74) we need to express (75) elementwise

$$a_{kj}^{KJ} = \sum_i \sum_I \left(w_{ki}^{KI}\right)^H b_{ij}^{IJ} \delta_{KJ} = \sum_i \sum_I \left(w_{ik}^{IK}\right)^* b_{ij}^{IJ} \delta_{KJ}. \quad (78)$$

Here, the Kronecker symbol δ_{KJ} corresponds to the $\mathrm{bdiag}\{\cdot\}$ operator zeroing the elements of all off-diagonal submatrices for $K \neq J$, and $b_{ij}^{IJ} = [\mathbf{RW}]_{ij}^{IJ}$. The derivative of this second part reads

$$\frac{\partial a_{kj}^{KJ}}{\partial \left(w_{mn}^{MN}\right)^*} = \frac{\partial}{\partial \left(w_{mn}^{MN}\right)^*} \sum_i \sum_I \left(w_{ik}^{IK}\right)^* b_{ij}^{IJ} \delta_{KJ} = \sum_i \sum_I \delta_{im}\delta_{kn}\delta_{IM}\delta_{KN}\delta_{KJ} b_{ij}^{IJ} = \delta_{kn}\delta_{KN}\delta_{KJ} b_{mj}^{MJ}. \quad (79)$$

Note that according to the Wirtinger calculus [20] matrix $\mathbf{W}$ in $\mathbf{A}$ is treated as a constant in the partial derivative.

Now we plug (75)—(77), and (79) into the chain rule (74) so that we obtain

$$\begin{aligned} \frac{\partial \mathcal{J}_1}{\partial \mathbf{W}^*} &= \sum_{k,j,K,J} [\mathrm{bdiag}^{-1}\mathbf{W}^H\mathbf{RW}]_{kj}^{KJ} \cdot \delta_{kn}\delta_{KN}\delta_{KJ}[\mathbf{RW}]_{mj}^{MJ} \\ &= \sum_j \left[(\mathrm{bdiag}\mathbf{W}^H\mathbf{RW})^{-1}\right]_{nj}^{NN} [\mathbf{RW}]_{mj}^{MN} \\ &= \left[\mathbf{RW}(\mathrm{bdiag}\mathbf{W}^H\mathbf{RW})^{-1}\right]_{mn}^{MN}. \end{aligned} \quad (80)$$

This result corresponds directly to (21).

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their useful comments which stimulated new results on convergence and on the internal permutation problem.

REFERENCES

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

[2] Y. Hua, S. An, and Y. Xiang, "Blind identification of FIR MIMO channels by decorrelating subchannels," *IEEE Trans. Signal Process.*, vol. 51, no. 5, pp. 1143–1155, May 2003.

[3] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: Wiley, 2001.

[4] L. Molgedey and H. G. Schuster, "Separation of a mixture of independent signals using time delayed correlations," *Phys. Rev. Lett.*, vol. 72, pp. 3634–3636, 1994.

[5] L. Tong *et al.*, "Indeterminacy and identifiability of blind identification," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 499–509, May 1991.

[6] E. Weinstein, M. Feder, and A. Oppenheim, "Multi-channel signal separation by decorrelation," *IEEE Trans. Speech Audio Process.*, vol. 1, no. 4, pp. 405–413, Oct. 1993.

[7] S. Van Gerven and D. Van Compernolle, "Signal separation by symmetric adaptive decorrelation: stability, convergence, and uniqueness," *IEEE Trans. Signal Process.*, vol. 43, no. 7, pp. 1602–1612, Jul. 1995.

[8] K. Matsuoka, M. Ohya, and M. Kawamoto, "A neural net for blind separation of nonstationary signals," *Neural Netw.*, vol. 8, no. 3, pp. 411–419, 1995.

[9] J.-F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Mat. Anal. Appl.*, vol. 17, no. 1, pp. 161–164, Jan. 1996.

[10] S. Ikeda and N. Murata, "An approach to blind source separation of speech signals," in *Proc. 8th Int. Conf. Artificial Neural Networks (ICANN)*, Skövde, Sweden, Sep. 1998, pp. 761–766.

[11] L. Parra and C. Spence, "Convolutive blind source separation based on multiple decorrelation," in *Proc. Int. Workshop Neural Networks Signal Processing*, Cambridge, U.K., Sep. 1998, pp. 23–32.

[12] D. W. E. Schobben and P. C. W. Sommen, "A frequency-domain blind signal separation method based on decorrelation," *IEEE Trans. Signal Process.*, vol. 50, no. 8, pp. 1855–1865, Aug. 2002.

[13] H.-C. Wu and J. C. Principe, "Simultaneous diagonalization in the frequency domain (SDIF) for source separation," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, 1999, pp. 245–250.

[14] C. L. Fancourt and L. Parra, "The coherence function in blind source separation of convolutive mixtures of nonstationary signals," in *Proc. Int. Workshop Neural Networks Signal Processing*, 2001, pp. 303–312.

[15] S. Araki *et al.*, "The fundamental limitation of frequency-domain blind source separation for convolutive mixtures of speech," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 109–116, Mar. 2003.

[16] H. Sawada *et al.*, "Spectral smoothing for frequency-domain blind source separation," in *Proc. Int. Workshop Acoustic Echo Noise Control (IWAENC)*, Kyoto, Japan, Sep. 2003, pp. 311–314.

[17] M. Z. Ikram and D. R. Morgan, "Exploring permutation inconsistency in blind separation of speech signals in a reverberant environment," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 2, Istanbul, Turkey, Jun. 2000, pp. 1041–1044.

[18] H. Buchner, R. Aichner, and W. Kellermann, "A generalization of a class of blind source separation algorithms for convolutive mixtures," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, Nara, Japan, Apr. 2003, pp. 945–950.

[19] H. Buchner, J. Benesty, and W. Kellermann, "Multichannel frequency-domain adaptive algorithms with application to acoustic echo cancellation," in *Adaptive Signal Processing: Application to Real-World Problems*, J. Benesty and Y. Huang, Eds. Berlin, Germany: Springer-Verlag, Jan. 2003.

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

[21] R. Aichner, H. Buchner, S. Araki, and S. Makino, "On-line time-domain blind source separation of nonstationary convolved signals," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, Nara, Japan, Apr. 2003, pp. 987–992.

[22] E. Moulines, O. A. Amrane, and Y. Grenier, "The generalized multidelay adaptive filter: structure and convergence analysis," *IEEE Trans. Signal Process.*, vol. 43, pp. 14–28, Jan. 1995.

[23] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, "Real-time convolutive blind source separation based on a broadband approach," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, Granada, Spain, Sep. 2004, pp. 833–840.

[24] R. Aichner, H. Buchner, and W. Kellermann, "Convolutive blind source separation for noisy mixtures," in *Proc. Joint Mtg. German French Acoustical Societies (CFA/DAGA)*, Strasbourg, France, Mar. 2004, pp. 583–584.

[25] H. Buchner, R. Aichner, and W. Kellermann, "Blind source separation for convolutive mixtures exploiting nongaussianity, nonwhiteness, and nonstationarity," in *Proc. Int. Workshop Acoustic Echo Noise Control (IWAENC)*, Kyoto, Japan, Sep. 2003, pp. 275–278.

[26] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. Berlin, Germany: Springer-Verlag, 1976.

[27] A. Oppenheim, "Inequalities connected with definite hermitian forms," *J. London Math. Soc.*, vol. 5, pp. 114–119, 1930.

[28] D. A. Harville, *Matrix Algebra From a Statistician's Perspective*. New York: Springer-Verlag, 1997.

[29] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*. New York: Wiley, 2002.

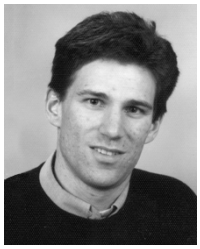
[30] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT, 1996, vol. 8, pp. 757–763.

[31] J.-F. Cardoso, "Blind signal separation: statistical principles," *Proc. IEEE*, vol. 86, pp. 2009–2025, Oct. 1998.

[32] R. Aichner *et al.*, "Time-domain blind source separation of nonstationary convolved signals with utilization of geometric beamforming," in *Proc. Int. Workshop Neural Networks Signal Processing (NNSP)*, Martigny, Switzerland, 2002, pp. 445–454.

[33] M. Kawamoto, K. Matsuoka, and N. Ohnishi, "A method of blind separation for convolved nonstationary signals," *Neurocomput.*, vol. 22, pp. 157–171, 1998.

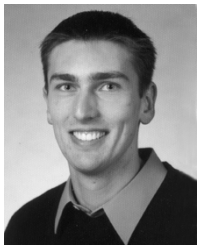
[34] T. Nishikawa, H. Saruwatari, and K. Shikano, "Comparison of time-domain ICA, frequency-domain ICA and multistage ICA for blind source separation," in *Proc. Eur. Signal Processing Conf. (EUSIPCO)*, vol. 2, Sep. 2002, pp. 15–18.
[35] B. S. Krongold and D. L. Jones, "Blind source separation of nonstationary convolutively mixed signals," in *Proc. IEEE Workshop Statistical Signal Array Processing (SSAP)*, Aug. 2000, pp. 53–57.
[36] L. Parra and C. Alvino, "Geometric source separation: merging convolutive source separation with geometric beamforming," *IEEE Trans. Speech Audio Process.*, vol. 10, pp. 352–362, Sep. 2002.
[37] S. Shamsunder and G. Giannakis, "Multichannel blind signal separation and reconstruction," *IEEE Trans. Speech Audio Process.*, vol. 5, no. 6, pp. 515–528, Nov. 1997.
[38] J. Anemüller, "Across-frequency processing in convolutive blind source separation," Ph.D. dissertation, Institute of Physics, Oldenburg University, Oldenburg, Germany, 2001.
[39] K. Rahbar and J. P. Reilly, "Blind source separation of convolved sources by joint approximate diagonalization of cross-spectral density matrices," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, Salt Lake City, UT, May 2001, pp. 2745–2748.
[40] H. Sawada, R. Mukai, S. Araki, and S. Makino, "Robust and precise method for solving the permutation problem of frequency-domain blind source separation," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, Nara, Japan, Apr. 2003, pp. 505–510.
[41] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, pp. 21–34, Jul. 1998.
[42] H. Gish and D. Cochran, "Generalized Coherence," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, New York, 1988, pp. 2745–2748.
[43] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Real-Time blind source separation for moving speakers using blockwise ICA and residual crosstalk subtraction," in *Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA)*, Nara, Japan, Apr. 2003, pp. 975–980.



Herbert Buchner (S'01–M'04) received the Dipl.-Ing (*Full Honors*) and the Dipl.-Ing. degrees in electrical engineering from the University of Applied Sciences, Regensburg, Germany, in 1997 and the University of Erlangen-Nuremberg, Erlangen, Germany, in 2000, respectively.

In 1995, he was a Visiting Researcher at the Colorado Optoelectronic Computing Systems Center (OCS), Boulder/Ft. Collins, where he worked in the field of microwave technology. From 1996 to 1997, he researched at the Cyber Space Laboratories (former Human Interface Laboratories), Research and Development division, Nippon Telegraph and Telephone Corp. (NTT), Tokyo, Japan, working on adaptive filtering for teleconferencing. In 1997 and 1998, he was with the Driver Information Systems Department, Siemens Automotive, Regensburg, Germany. His current areas of interest include efficient multichannel algorithms for adaptive digital filtering, and their applications for acoustic human-machine interfaces, such as multichannel acoustic echo cancellation, beamforming, blind source separation, and dereverberation.

Mr. Buchner received the VDI Award in 1998 for his Dipl.-Ing. thesis from the Verein Deutscher Ingenieure and a Best Student Paper award in 2001. He is a Member of the research staff at the Chair of Multimedia Communications and Signal Processing, University of Erlangen-Nuremberg, Germany.



Robert Aichner (S'03) received the Dipl.-Ing. (*Full Honors*) degree in electrical engineering from the University of Applied Sciences, Regensburg, Germany in 2002.

In 2000, he was an Intern at Siemens Energy and Automation, Atlanta, GA. From 2001 to 2002, he researched at the Speech Open Laboratory, Research and Devlopment division of the Nippon Telegraph and Telephone Corporation (NTT), Kyoto, Japan. There he was working on time-domain blind source separation of audio signals. His current research interests include multichannel adaptive algorithms for hands-free human-machine interfaces and its application to blind source separation, noise reduction, source localization, adaptive beamforming, and acoustic echo cancellation. In 2004, he was a Visiting Researcher at the Sound and Image Processing Laboratory, the Royal Institute of Technology (KTH), Stockholm, Sweden.

Mr. Aichner received the Stanglmeier Award for his intermediate diploma from the University of Applied Sciences, Regensburg in 1999. Since 2002, he has been a Member of the research staff at the Chair of Multimedia Communications and Signal Processing at the University of Erlangen-Nuremberg, Germany.

Walter Kellermann (M'89) received the Dipl.-Ing. degree in electrical engineering from the University of Erlangen, Nuremberg, Germany, in 1983, and the Dr.-Ing. degree from the Technical University, Darmstadt, Germany, in 1988.

He is Professor for communications at the chair of Multimedia Communications and Signal Processing of the University of Erlangen-Nuremberg, Germany. From 1989 to 1990, he was a Postdoctoral Member of the Technical Staff at AT&T Bell Laboratories, Murray Hill, NJ. After some time in industry and at the Fachhochschule Regensburg, Regensburg, Germany, he joined the University of Erlangen-Nuremberg, Germany, in 1999. He authored and coauthored six book chapters and more than 60 papers in journals and conference proceedings. His current research interests include speech signal processing, array signal processing, adaptive filtering, and its applications to acoustic human/machine interfaces. He served as Guest Editor to various journals.

Dr. Kellermann serves as Associate and Guest Editor to IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING.