Efficient Frequency Domain Implementation of Noncausal Multichannel Blind Deconvolution for Convolutive Mixtures of Speech

Seyedmahdad Mirsamadi, *Student Member, IEEE*, Shabnam Ghaffarzadegan, *Student Member, IEEE*, Hamid Sheikhzadeh, *Senior Member, IEEE*, Seyed Mohammad Ahadi, *Senior Member, IEEE*, and Amir Hossein Rezaie, *Member, IEEE*

***Abstract*—Multichannel blind deconvolution (MCBD) algorithms are known to suffer from an extensive computational complexity problem, which makes them impractical for blind source separation (BSS) of speech and audio signals. This problem is even more serious with noncausal MCBD algorithms that must be used in many frequently occurring BSS setups. In this paper, we propose a novel frequency domain algorithm for the efficient implementation of noncausal multichannel blind deconvolution. A block-wise formulation is first developed for filtering and adaptation of filter coefficients. Based on this formulation, we present a modified overlap-save procedure for noncausal filtering in the frequency domain. We also derive update equations for training both causal and anti-causal filters in the frequency domain. Our evaluations indicate that the proposed frequency domain implementation reduces the computational requirements of the algorithm by a factor of more than 100 for typical filter lengths used in blind speech separation. The algorithm is employed successfully for the separation of speech mixtures in a reverberant room. Simulation results demonstrate the superior performance of the proposed algorithm over causal MCBD algorithms in many potential source and microphone positions. It is shown that in BSS problems, causal MCBD algorithms with center-spike initialization do not always converge to a delayed form of the desired noncausal solution, further revealing the need for an efficient noncausal MCBD algorithm.**

***Index Terms*—Blind speech separation (BSS), filter decomposition, multichannel blind deconvolution (MCBD), noncausal filtering.**

I. Introduction

BLIND separation of signals is a rapidly growing area in the field of digital signal processing which finds various applications in audio and speech processing, telecommunications, and biomedical signal processing. Blind source separation (BSS) is the task of recovering sources from a set of observed mixtures, with virtually no *a priori* information about the sources or the mixing system. The only assumption is that the sources are statistically mutually independent.

Manuscript received August 16, 2011, revised March 04, 2012; accepted May 14, 2012. Date of publication June 05, 2012; date of current version August 13, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Mr. James Johnston.

Authors are with the Department of Electrical Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran 15875-4413, Iran (e-mail: smmscl@aut.ac.ir; ghaffarzadegan@aut.ac.ir; hsheikh@aut.ac.ir: sma@aut.ac.ir; rezaie@aut.ac.ir).

Digital Object Identifier 10.1109/TASL.2012.2202650

Numerous authors have addressed the problem of *instantaneous* BSS [1]–[7], in which the mixtures are assumed to be linear combinations of the sources. In this case, the mixing matrix consists only of real weights. Instantaneous BSS is useful in applications such as electroencephalography (EEG) and image processing. However, in audio applications in real acoustic environments, the assumption of instantaneous mixing does not hold, since each microphone picks up not only direct path propagations, but also several reflections of source signals arriving at different delays. This situation is termed *convolutive* mixing of source signals. Convolutive mixing is characterized by mixing matrices that contain filters representing the acoustic channels from each source to each microphone.

Several researchers have addressed the problem of convolutive BSS in the frequency domain [8]–[10]. Since the convolution operation changes to bin-wise multiplication in the frequency domain, the task of convolutive BSS is decomposed into several instantaneous BSS tasks for different frequency bins. However, such frequency domain approaches suffer from the well known scaling and permutation problems, due to different output ordering and different gains in each frequency bin. Possible solutions to these problems can be found in [11] and [12].

A completely different approach to convolutive BSS, which is followed in this paper, is using multichannel blind deconvolution (MCBD) methods, which are an extension of blind deconvolution techniques to multiple-input multiple-output (MIMO) systems [13], [14]. Since the cost function in these approaches is defined in the time domain, the permutation and scaling problems do not occur. The only problem appears to be a spectral whitening of temporally-correlated speech signals, which happens due to the assumption of independent-and-identically-distributed (i.i.d.) input samples. However, it has been shown that the whitening problem is easily removable through pre-emphasis of the microphone signals and the post-filtering of BSS outputs [14], [15]. One of the earliest works in multichannel blind deconvolution, which has received considerable attention in blind signal processing research, was presented by Amari *et al.* in [16]. Based on the Kullback–Leibler divergence between the joint distribution of output samples and the product of their marginal distributions, they defined a cost function in the time domain. They suggested that minimization of this cost function leads to output samples that are mutually independent, hence achieving blind separation and deconvolution. Since conventional stochastic gradient descent optimization methods are known to suffer from the slow convergence rate problem in blind

deconvolution tasks, two alternative methods, the Natural Gradient by Amari [17] and the Relative Gradient by Cardoso [18] are often employed in blind deconvolution. Using the natural gradient for the minimization of the cost function, the authors in [16] derived an adaptive learning procedure for causal finite impulse response (FIR) demixing filters.

A major problem associated with causal MCBD algorithms is their considerable performance degradation in nonminimum phase mixing environments. In a nonminimum phase MIMO system, the transmission zeros may lie anywhere inside or outside the unit circle. It is known that a stable (FIR) inverse of such systems necessarily includes anti-causal delays [19]. Thus, algorithms that exploit only causal demixing filters fail to effectively invert nonminimum phase mixing systems.

To tackle this problem, some authors have considered the use of center-spike initialization for the diagonal filters of the demixing system [16], [20], [21]. This means that instead of the commonly used unit impulse initialization at the first tap, these filters are initialized with a shifted impulse located at their middle tap. By using center-spike initialization, the demixing filters are expected to be trained to causal FIR filters, which are considered as a delayed form of the original noncausal filters. However, as will be shown in Section VI-C, this solution has a very limited capability in blind speech separation and does not always lead to proper identification of the mixing system.

A fundamental solution for multichannel blind deconvolution in nonminimum phase mixing systems, which will be referred to as noncausal MCBD (NMCBD) algorithm throughout this paper, was proposed by Zhang *et al.* in [22]. They suggested a decomposition of the demixing system into a cascade of anti-causal and causal filters. They proposed a simple cost function for this case, and derived a natural gradient learning procedure for the coefficients of both causal and anti-causal filters.

A problem involved in both causal and noncausal MCBD algorithms is their extensive computational requirements, due to time domain filtering and sample-by-sample adaptation of the filter coefficients. This problem is more severe in the noncausal MCBD algorithm, since two sets of filter coefficients (causal and anti-causal) have to be trained. In the original works [16], [22], the causal and noncausal MCBD algorithms have been employed for randomly generated i.i.d. signals that have been mixed with arbitrarily chosen mixing systems. The zero distribution of these mixing systems have been chosen such that their inverses do not have long impulse responses (6 taps for the causal MCBD and 60 taps for the NMCBD algorithm have been reported as sufficient in [16] and [22], respectively). This is in contrast to audio applications in real acoustic scenarios, where we need long demixing filters to cope with the reverberation in typical rooms. Moreover, most BSS algorithms need many seconds of data (typically 10 to 50 seconds) in order to converge to a stable solution. Thus, when using MCBD algorithms for the task of convolutive BSS in realistic applications, the computation task becomes prohibitively heavy due to the adaptation of many filter coefficients by a large number of input samples.

As an efficient solution for the computational complexity problem of MCBD algorithms, we consider the equivalent frequency domain implementation. It is noteworthy that in general there are two different approaches for implementing adaptive filtering problems in the frequency domain [23]. One is to train different weights in each frequency bin independently, which corresponds to the frequency domain BSS described earlier. This approach suffers from the serious drawbacks of permutation and scaling problems. Additionally, the noncausality problem has not yet been addressed in the frequency domain BSS. These problems motivate us to use another form of frequency domain implementation, which is the equivalent realization of time domain block adaptive filters using fast Fourier transform (FFT). For supervised least mean-square (LMS) adaptive filters, these two approaches have been shown to be equivalent [24]. However, in BSS problems this is not the case due to the ambiguities associated with the independent adaptation in each frequency bin. Apart from the permutation and scaling problems, it is not clear what nonlinearity is suitable for frequency domain adaptation, since the distribution of the discrete Fourier transform (DFT) coefficients generally differs from that of the time domain samples. Fortunately, such problems are not encountered in the FFT-based equivalent realization of block adaptive filters. In the rest of this paper, by frequency domain implementation we mean the second approach (block adaptive filters).

Frequency domain implementation of the causal MCBD algorithm of [16] was proposed by Joho *et al.* in [21] and [25]. In their algorithm, which is named FDMCBD, they used conventional overlap-save procedure for frequency domain causal filtering, and also introduced update equations to carry out the adaptation in the frequency domain. As will be seen in Section VI-C, this algorithm fails in many possible relative positions of the sources and microphones.

In this paper, we propose a novel algorithm for frequency domain implementation of noncausal multichannel blind deconvolution, which will be referred to as the frequency domain NMCBD (FNMCBD) algorithm. The frequency domain implementation of the NMCBD algorithm differs substantially from that of the causal MCBD algorithm due to several issues. First, due to the presence of anti-causal filters in the FNMCBD algorithm, a conventional overlap-save procedure is not suitable. Furthermore, the original time domain update equations of the NMCBD algorithm are more complicated than those of the causal MCBD algorithm, particularly due to the information back-propagation through the causal filters. As a result, deriving a frequency domain formulation for the NMCBD algorithm is more challenging than that of the causal MCBD algorithm.

Our main objective is to derive an efficient implementation that is suitable for speech separation in any arbitrary source and microphone configuration, and to use the algorithm in realistic acoustic scenarios. Causal MCBD algorithms have been previously employed for the separation of convolutive speech mixtures, and successful results have been reported [13], [14], [26]. However, as will be discussed in Section VI-A, in certain source and microphone positions, noncausal demixing filters are crucial for separation. Unfortunately, these situations happen quite frequently in practical applications. Thus, any algorithm with only causal demixing filters would be useless in a general and realistic BSS setup. We will show that the proposed FNMCBD algorithm performs successfully in such BSS problems. We will also show that causal MCBD algorithms with center-spike ini-

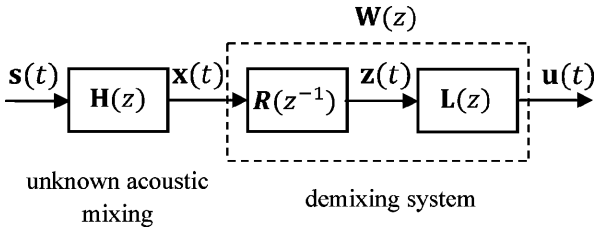


Fig. 1. Convolutive BSS model. The demixing system is a cascade of anti-causal and causal components ($R(z^{-1})$ and $L(z)$) in general.

tialization do not always converge to a delayed version of the original noncausal solution and hence are insufficient.

The rest of this paper is organized as follows. Section II describes the problem formulation and terminology. Section III describes the time domain NMCBD algorithm in both sample-by-sample and block-wise formulations. In Section IV, we present the derivation of the proposed FNMCBD algorithm. In Section V, we analyze the computational complexity of the proposed algorithm. Simulation results are provided in Sections VI, and Section VII concludes the paper.

II. Problem Formulation

A. Convolutive BSS Model

Assume there are N_s sources present in a reverberant room. The source signals are mixed by the unknown acoustic channels of the room and are picked up by N_m sensors. According to the convolutive BSS model of Fig. 1, the observed mixtures are

$$\mathbf{x}(t) = \sum_{p=0}^{M-1} \mathbf{H}_p \mathbf{s}(t-p) \tag{1}$$

where $\mathbf{H}_p$ is a $N_m \times N_s$ coefficient matrix, $\mathbf{H}(z) = \sum_{p=0}^{M-1} \mathbf{H}_p z^{-p}$ is a mixing system of length M, which could be minimum phase or nonminimum phase. $\mathbf{s}(t) = [s_1(t), \ldots, s_{N_s}(t)]^T$ and $\mathbf{x}(t) = [x_1(t), \ldots, x_{N_m}(t)]^T$ are vectors of source and sensor signals at time t, respectively. The element $H_{ij}(z)$ of the mixing system represents the acoustic channel from the jth source to the ith microphone.

In convolutive BSS, the aim is to find a demixing system which leads to an estimate of the source signals

$$\mathbf{u}(t) = \sum_{p=-\infty}^{\infty} \mathbf{W}_p \mathbf{x}(t-p) \tag{2}$$

where $\mathbf{W}_p$ is a $N_s \times N_m$ coefficient matrix, $\mathbf{W}(z) = \sum_{p=-\infty}^{\infty} \mathbf{W}(p) z^{-p}$ is the demixing system, and $\mathbf{u}(t) = [u_1(t), \ldots, u_{N_s}(t)]^T$ is a vector of output signals at time t. $W_{ij}(z)$ is the demixing filter from the mixture x_j to the output u_i.

In the rest of this paper, we focus on the standard square BSS model, i.e., the case of $N_m = N_s$.

The system $\mathbf{W}(z)$ should be the inverse of $\mathbf{H}(z)$ in the sense of

$$\mathbf{G}(z) = \mathbf{W}(z)\mathbf{H}(z) = \mathbf{P}\mathbf{D}(z) \tag{3}$$

where $\mathbf{G}(z)$ is the multichannel global filter from the sources to the BSS outputs, $\mathbf{P}$ is an arbitrary permutation matrix, and $\mathbf{D}(z)$ is a diagonal matrix with arbitrary filters as its diagonal elements. Note that in multichannel blind deconvolution the diagonal elements of $\mathbf{D}(z)$ must be of the pure delay form $d_i(z) = a_i z^{-T_i}$, where a_i is a scaling factor and T_i represents an arbitrary delay in the ith retrieved source signal. In contrast, in convolutive BSS we are not concerned with exact recovery of the source signals. Thus, the diagonal elements $d_i(z)$ are allowed to be arbitrary filters. This is referred to as the *filtering ambiguity* in convolutive BSS.

B. Review of the Filter Decomposition Approach

In the following, an overview of the filter decomposition approach to noncausal multichannel blind deconvolution is provided. In [22], it is proposed to decompose the noncausal demixing filter $\mathbf{W}(z)$ into a cascade structure of the form (Fig. 1)

$$\mathbf{W}(z) = \mathbf{L}(z)\mathbf{R}(z^{-1}) \tag{4}$$

$$\mathbf{L}(z) = \sum_{p=0}^{N} \mathbf{L}_p z^{-p} \tag{5}$$

$$\mathbf{R}(z^{-1}) = \sum_{p=0}^{N} \mathbf{R}_p z^{p} \tag{6}$$

where $\mathbf{L}(z)$ is a causal multichannel FIR filter and $\mathbf{R}(z^{-1})$ is an anti-causal multichannel FIR filter with the constraint $\mathbf{R}_0 = \mathbf{I}$; $\mathbf{I}$ being an identity matrix of size N_s.

The plausibility of such a decomposition becomes obvious by directly analyzing the inverse of the mixing system. Assume $\mathbf{H}(z)$ is a mixing system with no zeros on the unit circle, which might in general be nonminimum phase:

$$\mathbf{H}(z) = \sum_{p=0}^{M-1} \mathbf{H}_p z^{-p}, \qquad \det(\mathbf{H}_0) \neq 0. \tag{7}$$

The determinant of $\mathbf{H}(z)$ is expressed in the general form

$$\det(\mathbf{H}(z)) = c z^{-I_0} \prod_{p=1}^{I_1} (1 - a_p z^{-1}) \prod_{p=1}^{I_2} (1 - b_p z^{-1}) \tag{8}$$

where c is a constant gain factor, I_0, I_1 and I_2 are constant integers, and the parameters a_p and b_p represent minimum phase and maximum phase zeros of $\mathbf{H}(z)$, respectively ($0 < |a_p| < 1$ for $p = 1, \ldots, I_1$, and $|b_p| > 1$ for $p = 1, \ldots, I_2$).

The inverse of $\mathbf{H}(z)$ may then be expressed as

$$\mathbf{H}^{-1}(z) = \det(\mathbf{H}(z))^{-1} \mathbf{H}^{\#}(z) \tag{9}$$

where $\mathbf{H}^{\#}(z)$ is the adjoint matrix of $\mathbf{H}(z)$. Using the relationships

$$(1 - a_p z^{-1})^{-1} = \sum_{q=0}^{\infty} a_p^q z^{-q} \tag{10}$$

$$(1 - b_p z^{-1})^{-1} = -b_p^{-1} z \sum_{q=0}^{\infty} b_p^{-q} z^{q} \tag{11}$$

the filter $\mathbf{H}^{-1}(z)$ can be expressed as

$$\mathbf{H}^{-1}(z) = c^{-1} z^{I_0 + I_2} \prod_{p=1}^{I_2} (-b_p)^{-1} \mathbf{L}(z)\mathbf{R}(z^{-1}) \tag{12}$$

where

$$\mathbf{L}(z)=\sum_{r=0}^{\infty}\mathbf{L}_r z^{-r}=\mathbf{H}^{\#}(z)\prod_{p=1}^{I_1}\sum_{q=0}^{\infty}a_p^q z^{-q} \quad (13)$$

$$\mathbf{R}(z^{-1})=\sum_{r=0}^{\infty}\mathbf{R}_r z^{r}=\prod_{p=1}^{I_2}\sum_{q=0}^{\infty}b_p^{-q} z^{q}\mathbf{I}. \quad (14)$$

It is seen from (13) and (14) that each minimum phase zero of the mixing system produces a causal term in the inverse filter $\mathbf{H}^{-1}(z)$, whereas each maximum phase zero contributes to an anti-causal term. Thus, the filter $\mathbf{H^{-}}^{1}(z)$ inherently bears a cascade structure of causal and anti-causal filters, which justifies the decomposition of (4). $\mathbf{L}(z)$ and $\mathbf{R}(z^{-1})$ are considered to be the inverses of the minimum phase and maximum phase portions of $\mathbf{H}(z)$, respectively. Note that the coefficients $\mathbf{L}_p$ and $\mathbf{R}_p$ of the causal and anti-causal demixing filters have exponentially decaying norms, which enables us to approximate the IIR filters in (13) and (14) with FIR filters of appropriate length.

III. Time Domain Implementation

A. Natural Gradient Learning Algorithm

In this section, the time domain natural gradient NMCBD algorithm is reviewed, mostly based on the notations described in [22]. We will assume real input data and real filter coefficients in the rest of this paper, as it is the case with speech and audio applications. According to the demixing model of Fig. 1, the intermediate signal $\mathbf{z}(t)$ and the output signal $\mathbf{u}(t)$ are expressed as

$$\mathbf{z}(t)=\mathcal{R}\mathcal{X}(t) \quad (15)$$

$$\mathbf{u}(t)=\mathcal{L}\mathcal{Z}(t) \quad (16)$$

where

$$\mathcal{X}(t)=[\mathbf{x}^T(t),\mathbf{x}^T(t+1),\ldots,\mathbf{x}^T(t+N)]^T \quad (17)$$

$$\mathcal{Z}(t)=[\mathbf{z}^T(t),\mathbf{z}^T(t-1),\ldots,\mathbf{z}^T(t-N)]^T \quad (18)$$

$$\mathcal{R}=[\mathbf{R}_0,\ldots,\mathbf{R}_N] \quad (19)$$

$$\mathcal{L}=[\mathbf{L}_0,\ldots,\mathbf{L}_N]. \quad (20)$$

In the above equations, $N+1$ is the length of the demixing filters. For the training of filter coefficients, the following natural gradient updates are used:

$$\Delta\mathbf{R}_p(t+1) = -\eta\sum_{q=1}^{p}\sum_{r=0}^{N}\mathbf{L}_r^T(t)\mathbf{y}(t)\mathbf{z}^T(t-r+q)\mathbf{R}_{p-q}(t),\ p\in[1,N] \quad (21)$$

$$\Delta\mathbf{L}_p(t+1) = \eta\sum_{q=0}^{p}(\delta_{0,q}\mathbf{I}-\mathbf{y}(t)\mathbf{u}^T(t-q))\mathbf{L}_{p-q}(t),\ p\in[0,N] \quad (22)$$

where η is the adaptation step size, $\delta_{i,j}$ is the kronecker delta function, and $\mathbf{y}(t)=g(\mathbf{u}(t))$, in which the nonlinear function $g(\cdot)$ is ideally the score function of the source distributions, defined as

$$g(u)=-\frac{d\log p_u(u)}{du}. \quad (23)$$

However, simulations indicate that the algorithm performance is not very sensitive to the choice of nonlinearity, and thus a precise estimation of the source distributions is not necessary [27]. A frequently used nonlinear function is

$$g(u)=\mathrm{sign}(u)|u|^{\beta-1} \quad (24)$$

which is the score function of a generalized Gaussian distribution. For sub-Gaussian signals, a choice of $\beta>2$ would be suitable, and for super-Gaussian signals such as speech, β must be less than 2.

Note that in the algorithm described above, $\mathbf{R}_0$ is equal to the identity matrix throughout the learning procedure and is never changed.

B. Developing Block-Wise Filtering and Learning

The NMCBD algorithm discussed in Section III-A, follows a sample-by-sample update procedure for the training of demixing filters. Before transforming this algorithm to frequency domain, we need to introduce a block-wise formulation for filtering and adaptation. This means that the filter coefficients are kept constant during a block of input samples, and the whole block leads to a single cumulative update, equal to the average of all updates within that block. Thus, for the kth block, i.e., $t=kL-L+1,\ldots,kL$, we have

$$\mathbf{z}(t)=\mathcal{R}\mathcal{X}(t) \quad (25)$$

$$\mathbf{u}(t)=\mathcal{L}\mathcal{Z}(t) \quad (26)$$

for all t within the block, and

$$\Delta\mathbf{R}_p[k+1] = -\frac{\eta}{L}\sum_{t=kL-L+1}^{kL}\sum_{q=1}^{p}\sum_{r=0}^{N}\mathbf{L}_r^T[k]\mathbf{y}(t)\mathbf{z}^T(t-r+q)\mathbf{R}_{p-q}[k],\ p\in[1,N] \quad (27)$$

$$\Delta\mathbf{L}_p[k+1] = \frac{\eta}{L}\sum_{t=kL-L+1}^{kL}\sum_{q=0}^{p}(\delta_{0,q}\mathbf{I}-\mathbf{y}(t)\mathbf{u}^T(t-q))\mathbf{L}_{p-q}[k],\ p\in[0,N] \quad (28)$$

where L is the block length. In order to express the above equations in a simpler form which is amenable to frequency domain implementation, we define the following parameters:

$$\mathbf{F}_p[k]\triangleq\frac{1}{L}\sum_{t=kL-L+1}^{kL}\mathbf{z}(t-p)\mathbf{y}^T(t),\quad p\in[-N,N] \quad (29)$$

$$\mathbf{V}_p[k] \triangleq -\sum_{q=0}^{N} \mathbf{L}_q^T[k]\mathbf{F}_{q-p}^T[k], \quad p \in [1, N] \tag{30}$$

$$\mathbf{C}_p[k] \triangleq \delta_{0,p}\mathbf{I} - \frac{1}{L}\sum_{t=kL-L+1}^{kL} \mathbf{y}(t)\mathbf{u}^T(t-p), \quad p \in [0, N]. \tag{31}$$

Note that $\mathbf{V}_0[k]$ does not follow (30), and is always equal to a zero matrix. Using the above parameters, the updates (27) and (28) may alternatively be expressed as

$$\Delta\mathbf{R}_p[k+1] = \eta\sum_{q=0}^{p} \mathbf{V}_q[k]\mathbf{R}_{p-q}[k], \qquad p \in [0, N] \tag{32}$$

$$\Delta\mathbf{L}_p[k+1] = \eta\sum_{q=0}^{p} \mathbf{C}_q[k]\mathbf{L}_{p-q}[k], \qquad p \in [0, N]. \tag{33}$$

For the purpose of transforming these equations to the frequency domain, we reformulate them in single-input single-output (SISO) form. That is, we rewrite them in such a way that the updates for all taps of each SISO filter are given separately:

$$f_{mn}(p) = \frac{1}{L}\sum_{t=kL-L+1}^{kL} z_m(t-p)y_n(t), \; p \in [-N, N] \tag{34}$$

$$v_{mn}(p) = -\sum_{\alpha=1}^{N_s}\sum_{q=0}^{N} f_{n\alpha}(q-p)\ell_{\alpha m}(q), \; p \in [1, N] \tag{35}$$

$$c_{mn}(p) = \delta_{0,p}\delta_{m,n} - \frac{1}{L}\sum_{t=kL-L+1}^{kL} y_m(t)u_n(t-p) \tag{36}$$

$$\Delta r_{mn}(p) = \eta\sum_{\alpha=1}^{N_s}\sum_{q=0}^{p} v_{m\alpha}(q)r_{\alpha n}(p-q) \tag{37}$$

$$\Delta\ell_{mn}(p) = \sum_{\alpha=1}^{N_s}\sum_{q=0}^{p} c_{m\alpha}(q)\ell_{\alpha n}(p-q). \tag{38}$$

where in (36)–(38) $p \in [0, N]$. Note that $v_{mn}(0)$ is zero for all $m, n \in [1, N_s]$. In (34)–(38), the block index k has been dropped for simplicity.

IV. Frequency Domain Implementation

The algorithm described in Section III-B, like any other unsupervised adaptive filtering algorithm, consists of two different steps: filtering and adaptation. For an efficient implementation, both steps must be carried out in the frequency domain. In the following, we first present an overlap-save procedure for noncausal filtering. Then we demonstrate how to carry out the coefficient updates for both causal and anti-causal filters in the frequency domain.

A. Overlap-Save for Noncausal Filtering

In frequency domain filtering, proper fragmentation of the input sequence and adequate block overlapping are of great importance. Unlike conventional overlap-save procedure for causal filtering, here we need overlap regions on both sides of the main block, due to the presence of anti-causal and causal filters. Moreover, since frequency domain multiplication corresponds to circular convolution in the time domain, we will have invalid samples on both sides of the resulting vector, which should be discarded.

Now we focus on the BNMCBD algorithm described in Section III-B. As evident from (25)–(28), in order to apply the updates of the kth block, we need a span of $L + 2N$ samples from the input signals (the block itself, plus N samples from each side). Furthermore, due to noncausal filtering, we need additional N samples from each side to form the overlap regions. So we define the following blocks of signals with the total length of $L + 4N$:

$$\mathbf{x}_m[k] \triangleq [x_m(kL-L+1-2N), \ldots, x_m(kL+2N)]^T \tag{39}$$

$$\mathbf{z}_m[k] \triangleq [z_m(kL-L+1-2N), \ldots, z_m(kL+N), \bullet]^T \tag{40}$$

$$\mathbf{u}_m[k] \triangleq [\bullet, u_m(kL-L+1-N), \ldots, u_m(kL+N), \bullet]^T \tag{41}$$

$$\mathbf{y}_m[k] \triangleq [\mathbf{0}_{2N}, y_m(kL-L+1), \ldots, y_m(kL), \mathbf{0}_{2N}]^T \tag{42}$$

where $m \in [1, N_s]$, $\bullet$ represents a vector of length N that contains invalid samples, and $\mathbf{0}_{2N}$ is a vector of zeros with length $2N$. Note that in the FDMCBD algorithm [21], it is assumed that $L = N$, resulting in a substantial simplification of the algorithm. However, we prefer not to follow a similar constraint because it leads to an input vector of length $5N$, which necessarily requires zero-padding for any N in order to obtain a DFT length of power of two.

The filtering equations in the frequency domain are

$$\bar{\mathbf{z}}_m[k] = \sum_{n=1}^{N_s} \bar{\mathbf{r}}_{mn}^*[k] \odot \bar{\mathbf{x}}_n[k] \tag{43}$$

$$\bar{\mathbf{u}}_m[k] = \sum_{n=1}^{N_s} \bar{\boldsymbol{\ell}}_{mn}[k] \odot \bar{\mathbf{z}}_n[k] \tag{44}$$

in which $m \in [1, N_s]$, $\overline{(\cdot)}$ represents a parameter in the frequency domain, $\odot$ denotes element-wise multiplication of vectors, and $*$ denotes complex conjugation. Clearly the filters $\mathbf{r}_{mn}[k]$ and $\boldsymbol{\ell}_{mn}[k]$ have been zero-padded to the DFT length before being transformed to the frequency domain. Note that in (43), $\bar{\mathbf{r}}_{mn}[k]$ is used in conjugate form due to anti-causal filtering.

The nonlinear function $g(\cdot)$ must be applied to the elements of the time domain vector $\mathbf{u}_m[k]$. Thus, we will have

$$\bar{\mathbf{y}}_m[k] = \mathbb{F}(\mathbf{P}_y g(\mathbf{u}_m[k])) \tag{45}$$

$$\mathbf{P}_y \triangleq \mathrm{diag}\{[\mathbf{0}_{2N}, \mathbf{1}_L, \mathbf{0}_{D-L-2N}]\} \tag{46}$$

where $m \in [1, N_s]$, D is the DFT length, $\mathbb{F}$ is a DFT matrix of size D, and $\mathbf{1}_L$ denotes a vector of all ones with length L.

B. Updating Anti-Causal Filters

In this section, we describe the procedure of transforming (34), (35), and (37) to the frequency domain. We start by

rewriting the parameter $f_{mn}(p)$ for different lags in the compact form

$$\begin{bmatrix} f_{mn}(-N) \\ \vdots \\ f_{mn}(N) \end{bmatrix} = \frac{1}{L} \begin{bmatrix} z_m(kL-L+1+N) & \dots & z_m(kL+N) \\ \vdots & \ddots & \vdots \\ z_m(kL-L+1-N) & \dots & z_m(kL-N) \end{bmatrix} \times \begin{bmatrix} y_n(kL-L+1) \\ \vdots \\ y_n(kL) \end{bmatrix}. \quad (47)$$

The particular structure in the above matrix suggests that it can be expanded to a circulant matrix of size D. This is achieved by adding $2N$ columns to each side, and $L+2N-1$ rows to the lower end of the matrix, with appropriate samples of $z_m(t)$ as the new elements. To accommodate the new dimensions, we zero-pad the left-hand side vector of the above equation to the length D, and add $2N$ zeros to both sides of the right-hand side vector to get

$$\mathbf{f}_{mn}[k] = \frac{1}{L}\mathbf{P}_f \hat{\mathbf{Z}}_m^T[k]\mathbf{y}_n[k] \quad (48)$$

where $\hat{\mathbf{Z}}_m^T[k]$ is a $D \times D$ circulant matrix whose first column is the vector $\mathbf{z}_m[k]$ circularly shifted to the left by N samples [see (40)]. The matrix $\mathbf{P}_f$ is defined as

$$\mathbf{P}_f \triangleq \begin{bmatrix} \mathbf{J}_{(2N+1)\times(2N+1)} & \mathbf{0}_{(2N+1)\times(D-2N-1)} \\ \mathbf{0}_{(D-2N-1)\times(2N+1)} & \mathbf{0}_{(D-2N-1)\times(D-2N-1)} \end{bmatrix} \quad (49)$$

in which $\mathbf{J}$ is the exchange matrix. We use this definition because as will be demonstrated later in this section, a time-reversed form of $f_{mn}(p)$ is needed in the rest of the derivation.

Equation (48) now represents a circular convolution, which can be computed in the DFT domain. To achieve this, we premultiply both sides by the DFT matrix to obtain

$$\mathbb{F}\mathbf{f}_{mn}[k] = \frac{1}{L}\mathbb{F}\mathbf{P}_f\mathbb{F}^{-1}\mathbb{F}\hat{\mathbf{Z}}_m^T[k]\mathbb{F}^{-1}\mathbb{F}\mathbf{y}_n[k]. \quad (50)$$

From the circulant matrix theory [28], we know that $\mathbb{F}\hat{\mathbf{Z}}_m^T[k]\mathbb{F}^{-1}$ is a diagonal matrix whose elements are the conjugate DFT coefficients of the first column of $\hat{\mathbf{Z}}_m[k]$

$$\mathbb{F}\hat{\mathbf{Z}}_m^T[k]\mathbb{F}^{-1} = \mathrm{diag}\{(\mathbb{M} \odot \bar{\mathbf{z}}_m[k])^*\} \quad (51)$$

where $\mathbb{M} = \left[e^{j(2\pi N/D)q}\right]_{q=0,\dots D-1}$ is used to account for the circular shift in $\mathbf{z}_m[k]$. Thus, (50) can be rewritten as

$$\bar{\mathbf{f}}_{mn}[k] = \frac{1}{L}\mathbb{F}\mathbf{P}_f\mathbb{F}^{-1}(\mathbb{M}^* \odot \bar{\mathbf{z}}_m^*[k] \odot \bar{\mathbf{y}}_n[k]). \quad (52)$$

Recall that the vector $\mathbf{f}_{mn}[k]$ contains time-lags $-N$ to N of $f_{mn}(-p)$. Thus, by multiplying it by the DFT matrix, we are implicitly introducing a delay of N samples to the original $f_{mn}(-p)$.

Now we wish to compute the parameter $v_{mn}(p)$ in the frequency domain. Clearly, (35) can be interpreted as a linear convolution between the sequences $f_{n\alpha}(-p)$ and $\ell_{\alpha m}(p)$. Since both vectors $\mathbf{f}_{n\alpha}[k]$ and $\boldsymbol{\ell}_{\alpha m}[k]$ are adequately zero-padded, the computations may be equivalently carried out in the frequency domain. So we have

$$\bar{\mathbf{v}}_{mn}[k] = -\mathbb{F}\mathbf{P}_v\mathbb{F}^{-1}\left(\sum_{\alpha=1}^{N_s} \bar{\mathbf{f}}_{n\alpha}[k] \odot \bar{\boldsymbol{\ell}}_{\alpha m}[k]\right) \quad (53)$$

where

$$\mathbf{P}_v \triangleq \begin{bmatrix} \mathbf{0}_{1\times(N+1)} & \mathbf{0}_{1\times N} & \mathbf{0}_{1\times(D-2N-1)} \\ \mathbf{0}_{N\times(N+1)} & \mathbf{I}_{N\times N} & \mathbf{0}_{N\times(D-2N-1)} \\ \mathbf{0}_{(D-N-1)\times(N+1)} & \mathbf{0}_{(D-N-1)\times N} & \mathbf{0}_{(D-N-1)\times(D-2N-1)} \end{bmatrix}. \quad (54)$$

A number of issues must be pointed out to justify the definition of the above matrix. First of all, the convolution of the sequences $f_{n\alpha}(-p)$ and $\ell_{\alpha m}(p)$ will generate $3N+1$ samples, of which only $N+1$ are needed for $v_{mn}(p)$. Hence, the rest of the generated samples must be zeroed out. Second, as mentioned earlier in this section, $\bar{\mathbf{f}}_{mn}[k]$ bears an inherent delay of N samples. This delay will also be reflected in the result of convolution. So the lags $p = 0, \dots, N$ of $v_{mn}(p)$ will appear in the elements $N+1$ to $2N+1$ of the resulting vector. Finally, as we stated in Section III-B, $v_{mn}(0)$ must always remain zero for all m and n. The matrix $\mathbf{P}_v$ in (54) is defined to construct a zero-padded vector from the lags $p = 0, \dots, N$ of $v_{mn}(p)$, considering all of the aforementioned issues.

The last step in updating anti-causal filters is to compute $\Delta r_{mn}(p)$ in the frequency domain. Since (37) also has the form of a linear convolution, we have

$$\Delta\bar{\mathbf{r}}_{mn}[k+1] = \eta \sum_{\alpha=1}^{N_s} \bar{\mathbf{v}}_{m\alpha}[k] \odot \bar{\mathbf{r}}_{\alpha n}[k]. \quad (55)$$

Here again the convolution produces $2N+1$ samples, of which only the first $N+1$ samples are required for the update. Thus, the final update equation for the anti-causal filters would be

$$\bar{\mathbf{r}}_{mn}[k+1] = \bar{\mathbf{r}}_{mn}[k] + \eta\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}\sum_{\alpha=1}^{N_s} \bar{\mathbf{v}}_{m\alpha}[k] \odot \bar{\mathbf{r}}_{\alpha n}[k] \quad (56)$$

in which

$$\mathbf{P}_N \triangleq \mathrm{diag}\{[\mathbf{1}_{N+1}, \mathbf{0}_{D-N-1}]\}. \quad (57)$$

C. Updating Causal Filters

The procedure for updating the causal filters in the frequency domain is mostly similar to that of anti-causal filters. We start by rewriting (36) for different lags in the matrix form

$$\begin{bmatrix} c_{mn}(0) \\ \vdots \\ c_{mn}(N) \end{bmatrix} = \delta_{m,n} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} - \frac{1}{L} \begin{bmatrix} u_n(kL-L+1) & \cdots & u_n(kL) \\ \vdots & \ddots & \vdots \\ u_n(kL-L+1-N) & \cdots & u_n(kL-N) \end{bmatrix} \times \begin{bmatrix} y_m(kL-L+1) \\ \vdots \\ y_m(kL) \end{bmatrix}. \quad (58)$$

Again by expanding the dimensions of the above equation to the DFT length D, we get

$$\mathbf{c}_{mn}[k] = \delta_{m,n}\boldsymbol{\Delta} - \frac{1}{L}\mathbf{P}_N\hat{\mathbf{U}}_n^T[k]\mathbf{y}_m[k] \quad (59)$$

TABLE I
FNMCBD ALGORITHM

Definitions

choose $D \geq L + 4N$

$\mathbb{M} = \left[e^{j\frac{2\pi}{D}Nq}\right]_{q=0,\ldots D-1}$

$\mathbf{P}_N \triangleq \text{diag}\{[\mathbf{1}_{N+1}, \mathbf{0}_{D-N-1}]\}$

$\mathbf{P}_y \triangleq \text{diag}\{[\mathbf{0}_{2N}, \mathbf{1}_L, \mathbf{0}_{D-L-2N}]\}$

$\mathbf{P}_f \triangleq \begin{bmatrix} \mathbf{J}_{(2N+1)\times(2N+1)} & \mathbf{0}_{(2N+1)\times(D-2N-1)} \\ \mathbf{0}_{(D-2N-1)\times(2N+1)} & \mathbf{0}_{(D-2N-1)\times(D-2N-1)} \end{bmatrix}$

$\mathbf{P}_v \triangleq \begin{bmatrix} \mathbf{0}_{1\times(N+1)} & \mathbf{0}_{1\times N} & \mathbf{0}_{1\times(D-2N-1)} \\ \mathbf{0}_{N\times(N+1)} & \mathbf{I}_{N\times N} & \mathbf{0}_{N\times(D-2N-1)} \\ \mathbf{0}_{(D-N-1)\times(N+1)} & \mathbf{0}_{(D-N-1)\times N} & \mathbf{0}_{(D-N-1)\times(D-2N-1)} \end{bmatrix}$

Initialization $(\forall m, n)$

$\bar{\mathbf{r}}_{mn}[0] = \text{FFT}\ \{[\mathbf{r}_{mn}[0], \mathbf{0}_{D-N+1}]\}^T$

$\bar{\boldsymbol{\ell}}_{mn}[0] = \text{FFT}\ \{[\boldsymbol{\ell}_{mn}[0], \mathbf{0}_{D-N+1}]\}^T$

For each block k:

Filtering $(\forall m)$

$\bar{\mathbf{x}}_m[k] = \text{FFT}\ \{[x_m(kL - L + 1 - 2N), \cdots, x_m(kL + 2N)]\}^T$

$\bar{\mathbf{z}}_m[k] = \sum_{n=1}^{N_s} \bar{\mathbf{r}}^*_{mn}[k] \odot \bar{\mathbf{x}}_n[k]$

$\bar{\mathbf{u}}_m[k] = \sum_{n=1}^{N_s} \bar{\boldsymbol{\ell}}_{mn}[k] \odot \bar{\mathbf{z}}_n[k]$

$\mathbf{u}_m[k] = \mathbb{F}^{-1}\bar{\mathbf{u}}_m[k]$

$\bar{\mathbf{y}}_m[k] = \mathbb{F}(\mathbf{P}_y g(\mathbf{u}_m[k]))$

Adaptation $(\forall m, n)$

$\bar{\mathbf{f}}_{mn}[k] = \frac{1}{L}\mathbb{F}\mathbf{P}_f\mathbb{F}^{-1}(\mathbb{M}^* \odot \bar{\mathbf{z}}^*_m[k] \odot \bar{\mathbf{y}}_n[k])$

$\bar{\mathbf{v}}_{mn}[k] = -\mathbb{F}\mathbf{P}_v\mathbb{F}^{-1}(\sum_{\alpha=1}^{N_s} \bar{\mathbf{f}}_{n\alpha}[k] \odot \bar{\boldsymbol{\ell}}_{\alpha m}[k])$

$\bar{\mathbf{c}}_{mn}[k] = \delta_{m,n}\mathbf{1}_D - \frac{1}{L}\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}(\bar{\mathbf{u}}^*_n[k] \odot \bar{\mathbf{y}}_m[k])$

$\bar{\mathbf{r}}_{mn}[k+1] = \bar{\mathbf{r}}_{mn}[k] + \eta\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}\sum_{\alpha=1}^{N_s} \bar{\mathbf{v}}_{m\alpha}[k] \odot \bar{\mathbf{r}}_{\alpha n}[k]$

$\bar{\boldsymbol{\ell}}_{mn}[k+1] = \bar{\boldsymbol{\ell}}_{mn}[k] + \eta\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}\sum_{\alpha=1}^{N_s} \bar{\mathbf{c}}_{m\alpha}[k] \odot \bar{\boldsymbol{\ell}}_{\alpha n}[k]$

where $\Delta = [1, \mathbf{0}_{D-1}]$, and $\hat{\mathbf{U}}_n[k]$ is a circulant matrix whose first column is the vector $\mathbf{u}_n[k]$ defined in (41). By applying the DFT matrix to both sides of the above equation and following the same procedure as we did for $\bar{\mathbf{f}}_{mn}[k]$, we have

$$\bar{\mathbf{c}}_{mn}[k] = \delta_{m,n}\mathbf{1}_D - \frac{1}{L}\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}(\bar{\mathbf{u}}^*_n[k] \odot \bar{\mathbf{y}}_m[k]) \qquad (60)$$

and considering the linear convolution in (38), the final update equation for the causal filters would be

$$\bar{\boldsymbol{\ell}}_{mn}[k+1] = \bar{\boldsymbol{\ell}}_{mn}[k] + \eta\mathbb{F}\mathbf{P}_N\mathbb{F}^{-1}\sum_{\alpha=1}^{N_s} \bar{\mathbf{c}}_{m\alpha}[k] \odot \bar{\boldsymbol{\ell}}_{\alpha n}[k]. \qquad (61)$$

A summary of the proposed frequency domain algorithm is provided in Table I.

V. COMPUTATIONAL COMPLEXITY ANALYSIS

In this section, we examine the computational requirements of the proposed algorithm and compare it to original time domain algorithms. Table II shows the number of real multiplications (N_{RM}) and real additions (N_{RA}) per sample of input, required by each of the three discussed algorithms, namely the original sample-by-sample time domain algorithm (NMCBD) described in Section III-A, the block-wise time domain algorithm (BNMCBD) described in Section III-B, and the proposed frequency domain implementation (FNMCBD) as summarized in Table I. For simplicity, it has been assumed that there are two sources and two sensors in all cases. The computational requirements of one FFT or inverse FFT operation have been counted as $2N\log_2 N$ real multiplications and $3N\log_2 N$ real additions. Moreover, since the computational requirement of the score function $g(\cdot)$ is the same for all of the three algorithms, it has been ignored in evaluating the total amount of computations. It is important to note that multiplications by the matrices $\mathbf{P}_f$, $\mathbf{P}_v$ and $\mathbf{P}_N$ do not count as actual multiplications, since they only correspond to shifting and zeroing of the vector elements in the algorithm.

TABLE II
NUMBER OF REAL MULTIPLICATIONS (N_{RM}) AND REAL ADDITIONS (N_{RA}) PER SAMPLE OF INPUT IN THE TIME DOMAIN AND FREQUENCY DOMAIN ALGORITHMS

Algorithm	N_{RM}	N_{RA}
NMCBD	$20N^2 + 32N$	$16N^2 + 36N$
BNMCBD	$(12 + \frac{8}{L})N^2 + (24 + \frac{8}{L})N$	$(8 + \frac{8}{L})N^2 + (20 + \frac{16}{L})N$
FNMCBD	$\frac{D}{L}(44 + 92\log_2 D)$	$\frac{D}{L}(26 + 138\log_2 D)$

TABLE III
NUMERICAL VALUES OF ($N_{RM} + N_{RA}$) FOR DIFFERENT FILTER LENGTHS N. CRR IS THE COMPLEXITY REDUCTION RATIO OFFERED BY THE FREQUENCY DOMAIN IMPLEMENTATION

N	NMCBD	BNMCBD	FNMCBD	CRR
32	39040	22022	3820	10
64	151808	84998	4280	35
128	598528	333830	4740	126
256	2376704	1323014	5200	457
512	9472000	5267462	5660	1673

To provide better comparison, the values of $N_{RM} + N_{RA}$ in the three algorithms have been evaluated for different filter lengths and are summarized in Table III. We have treated multiplications and additions similarly since many modern CPUs (especially digital signal processors) are capable of performing single-instruction multiplications. A block length of $L = 4N$ is assumed for BNMCBD and FNMCBD algorithms throughout this section, which corresponds to a DFT length of $D = 8N$. Also shown in Table III is the complexity reduction ratio (CRR) which is defined as

$$CRR \triangleq \frac{(N_{RM} + N_{RA})_{\text{in NMCBD algorithm}}}{(N_{RM} + N_{RA})_{\text{in FNMCBD algorithm}}}. \qquad (62)$$

The numerical values in the table clearly demonstrate the considerable computational savings offered by the frequency domain implementation. For most typical filter lengths used in blind speech separation, a reduction ratio greater than 100 is achieved.

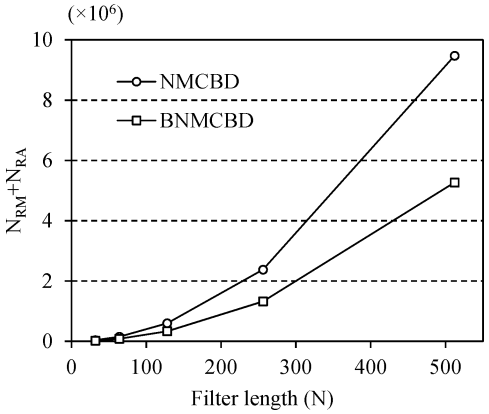


Fig. 2. Overall number of real multiplications and real additions ($N_{RM} + N_{RA}$) per sample of input for different filter lengths (N) in the sample-by-sample and block-wise time domain algorithms, assuming a block length of $L = 4N$.

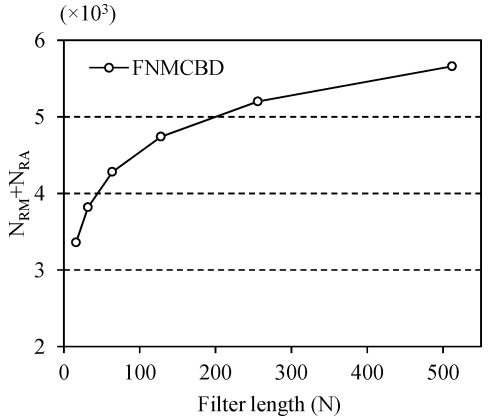


Fig. 3. Overall number of real multiplications and real additions ($N_{RM} + N_{RA}$) per sample of input for different filter lengths (N) in the FNMCBD algorithm, assuming a block length of $L = 4N$.

Fig. 2 compares the overall number of real multiplications and real additions per sample of input in the NMCBD and BNMCBD algorithms for different filter lengths. As observed in the figure, the block-wise formulation only reduces the computational burden by a small constant factor (an approximate factor of 2 for large block lengths). Thus a direct block-wise implementation offers marginal computational advantages over the original sample-by-sample algorithm. This is due to the fact that the block-wise adaptation only saves computations in the filters' updates (32) and (33), because they are carried out only once for every L samples. The computations related to the evaluation of the parameters $\mathbf{F}_p$ and $\mathbf{C}_p$ remain unchanged, since they involve all samples within a block. In contrast, the frequency domain implementation provides savings in all steps of the algorithm, including the computation of all intermediate signals and parameters. This leads to considerably lower computational complexity in the FNMCBD algorithm, which is plotted in Fig. 3 for different filter lengths. Notice the logarithmic trend in the computational requirements of the FNMCBD algorithm for $L = 4N$ (also see Table II), which makes it particularly useful for reverberant rooms where long demixing filters are required. This is in contrast to the time domain algorithms NMCBD and BNMCBD, in both of which the computational complexity increases exponentially with N.

As indicated by Fig. 2, the time domain algorithms NMCBD and BNMCBD require a very large number of computations for each sample of the observed sensor mixtures. Therefore, in speech and audio applications with many thousands of input samples, both of these algorithms are completely impractical. In contrast, the computational requirement of the proposed FNMCBD algorithm is relatively small, making it the only practical choice among the three algorithms for the separation of speech signals using typically available computational resources such as personal computers. As a rough example, based on our experiments for a filter length of 256, a duration of 30 seconds of input data, and a sampling frequency of 8 kHz, the original NMCBD algorithm takes as long as a couple of days on a good PC in order to converge to a satisfactory level of separation. But this time for the FNMCBD algorithm is only a few minutes for most typical block lengths. Such relative consumed times are justified by the entries of Table III.

VI. Simulation Results

In the following, we present the results of the computer simulations that have been performed to evaluate the separation performance of the proposed algorithm. As a performance measure, we use the signal-to-interference ratio (SIR) improvement [29], which is defined for the ith output channel as

$$SIRI_i = SIR_{out,i} - SIR_{\text{in},i}, \qquad i \in [1, N_s] \tag{63}$$

$$SIR_{out,i} = 10\log_{10}\frac{E[u_{i,s_i}^2]}{E\left[\left(\sum_{j\neq i}^{N_s} u_{i,s_j}\right)^2\right]} \tag{64}$$

$$SIR_{\text{in},i} = 10\log_{10}\frac{E[x_{i,s_i}^2]}{E\left[\left(\sum_{j\neq i}^{N_s} x_{i,s_j}\right)^2\right]} \tag{65}$$

where x_{i,s_j} and u_{i,s_j} denote the contribution of the source s_j to the microphone signal x_i and the output signal u_i, respectively.

The tests are all performed for the standard 2×2 convolutive BSS system. The two source signals used in the experiments are two speech signals of a male and a female speaker (from the TIMIT database [30]), each with a duration of 30 seconds and sampled at 8 kHz. As the mixing system, we use impulse responses related to a rectangular room which are obtained by the well known image method [31], [32]. The simulations are carried out for a 5.5 m $\times$ 5 m $\times$ 3 m room with a reverberation time of $T_{60} \simeq 140$ ms. A microphone array with an inter-element spacing of 20 cm has been placed at a fixed position as shown in Fig. 4, while the source positions have been set according to the two different scenarios described in the following sections. In all simulations, the parameter β in the score function (24) is equal to 0.7. Our experiments have shown that this choice of β leads to the best estimation of score function for the input speech signals. The length of causal and anti-causal filters are set to 256 samples ($N + 1 = 256$), which corresponds to a total demixing filter length of 512 samples. Moreover, we choose a block length of 1028 samples, which leads to a DFT length of $L + 4N = 2048$ samples. Note the careful selection of the parameters L and N to obtain a DFT length of power of

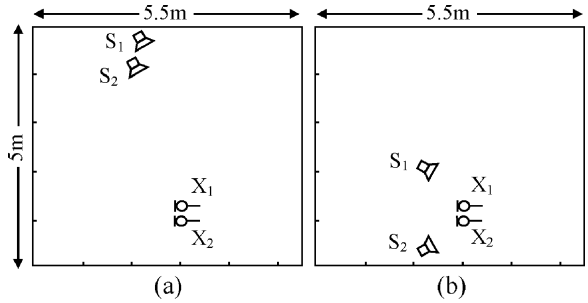


Fig. 4. Positions of the sources (S_1, S_2) and microphones (X_1, X_2). (a) Noncausal BSS problem: both sources are located at the same side of the microphone array. (b) Causal BSS problem: the sources are located at opposite sides of the microphone array. The room height is 3 m in both cases.

2. This is of course not a fundamental limitation on the choices of L and N, since in any case the vector $\mathbf{x}_m[k]$ in (39) may be zero-padded to an appropriate DFT length, allowing L and N to have arbitrary values.

A fixed number of iterations is used in each experiment for the adaptation of filter coefficients. A complete pass through the whole input data is considered one iteration. The number of iterations was determined by practice to be adequate for reaching a maximum final SIR. A constant step size parameter of $\eta = 0.001$ is used for the FNMCBD algorithm in all simulations. This is an optimum step size for the algorithm which was found by experiment. The diagonal filters of both causal and anti-causal demixing systems are initialized with unit impulses in their first taps, whereas the off-diagonal elements are initialized with zero filters.

A. Performance Evaluation in Noncausal BSS Problems

In this example, we evaluate the performance of the proposed algorithm in the mixing scenario of Fig. 4(a). It is known that in such configurations, in which both sources are located at the same side of the microphone array, both anti-causal and causal filters are needed for separation, and the causal filters alone would not achieve any degree of separation [33]. Thus, in a blind scenario with no guarantee on the relative positions of the sources and the microphones, we must inevitably employ a noncausal MCBD algorithm.

The input SIRs of the observed mixtures in the configuration of Fig. 4(a) are 2.6 dB and -2.2 dB. Fig. 5 shows the SIR improvements of the proposed algorithm versus iteration number, averaged over the two channels. It is observed that the algorithm achieves a final SIR improvement of about 16 dB, which shows that the filters $\mathbf{L}(z)$ and $\mathbf{R}(z^{-1})$ have been properly trained to invert the minimum phase and maximum phase portions of the mixing system, respectively. Fig. 6 shows the global filters (the cascade of mixing and demixing filters) after convergence of the algorithm. As observed in the figure, the cross filters $g_{12}(p)$ and $g_{21}(p)$ are zero filters, indicating a successful interference cancellation. In addition to SIRs, the total signal-to-distortion ratios (SDRs), as defined in [34], were also computed for the output signals. The SDR averaged over the two output channels was measured to be 11.5 dB, which indicates a proper recovery of the original speech signals.

For comparison, the original sample-by-sample NMCBD algorithm was employed for the same experimental setup of Fig. 4(a) (although the inefficiency of time domain implementation renders this algorithm impractical for such BSS applications). The resulting average SIR improvement is plotted in Fig. 7 versus iteration number. It is observed from the figure that a sample-by-sample adaptation does not provide a uniform convergence, and also the final achieved SIR is smaller than the FNMCBD algorithm. In contrast, block-wise adaptation often provides a smooth and uniform convergence for most block lengths, possibly because a block update usually achieves better estimates of the (natural) gradient direction for mixing systems that are not time-varying. Thus, the proposed frequency domain algorithm not only provides considerable computational savings, but also offers advantages in terms of convergence properties and separation performance due to block-wise adaptation.

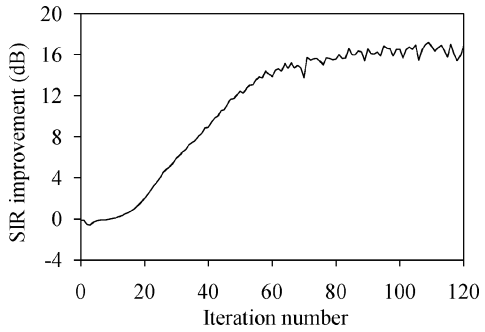


Fig. 5. SIR improvements of the FNMCBD algorithm in the noncausal BSS problem (averaged over the two channels).

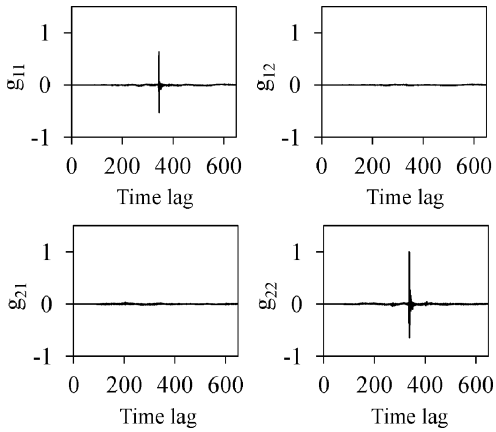


Fig. 6. Global filters of the FNMCBD algorithm after convergence in the noncausal BSS problem.

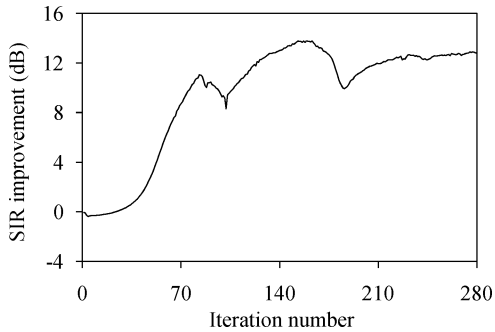


Fig. 7. SIR improvements of the sample-by-sample NMCBD algorithm in the noncausal BSS problem (averaged over the two channels).

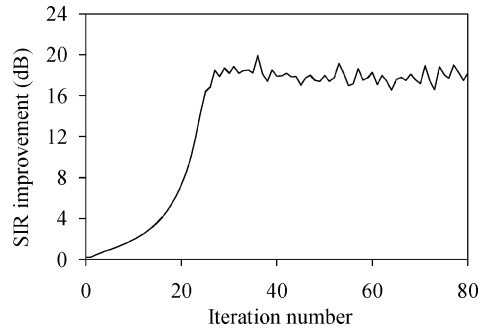


Fig. 8. SIR improvements of the FNMCBD algorithm in the causal BSS problem (averaged over the two channels).

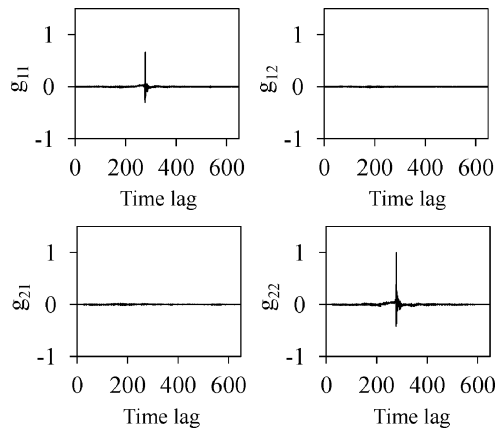


Fig. 9. Global filters of the FNMCBD algorithm after convergence in the causal BSS problem.

B. Performance Evaluation in Causal BSS Problems

This example illustrates the performance of the proposed algorithm in mixing scenarios such as the one shown in Fig. 4(b). In such situations where the sources are located at the opposite sides of the microphone array, only causal filters are sufficient for separation [33]. However, in a completely blind scenario, there is no *a priori* information about the location of the sources. Thus, the noncausal MCBD algorithm summarized in Table I must be directly applicable to these scenarios without removing the anti-causal filter $\mathbf{R}(z^{-1})$. In other words, the algorithm is expected to identify the sufficiency of causal filters and converge to one-sided overall demixing filters.

The configuration of Fig. 4(b) leads to mixture SIRs of 3.9 dB and -1.1 dB. The average SIR improvement of the two channels achieved by the proposed algorithm is depicted in Fig. 8, and the global filters after convergence are shown in Fig. 9. Both figures indicate successful separation. The average SDR value in this case is 13 dB, showing a successful recovery of the source signals.

Fig. 10 shows the anti-causal demixing filters for this example after convergence. It is observed that the algorithm has virtually maintained the initial condition of $\mathbf{R}(z^{-1}) = \mathbf{I}$, effectively removing any contribution from $\mathbf{R}(z^{-1})$ to the overall demixing system. In contrast, the anti-causal filters in the previous example [the situation of Fig. 4(a)] have been trained to appropriate filters which act as essential components of the demixing system (see Fig. 11).

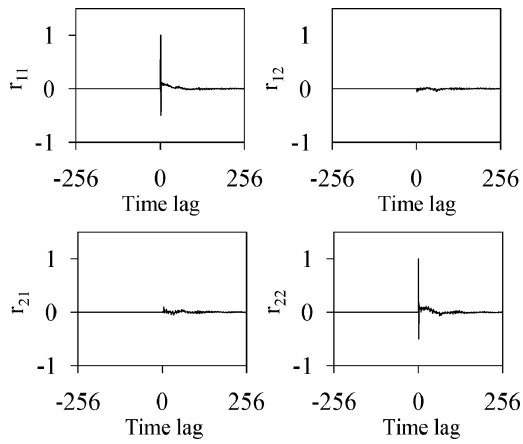


Fig. 10. Anti-causal demixing filters $\mathbf{R}(z^{-1})$ after convergence in the causal BSS problem.

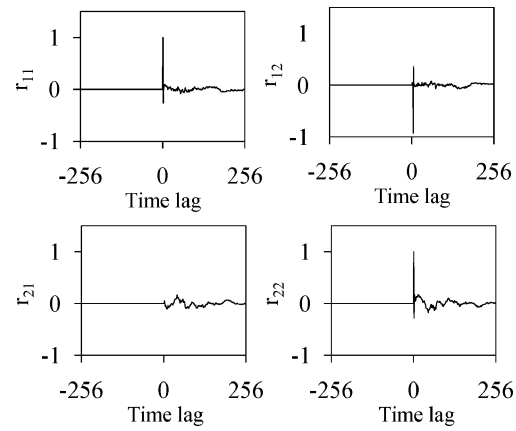


Fig. 11. Anti-causal demixing filters $\mathbf{R}(z^{-1})$ after convergence in the noncausal BSS problem.

C. Comparison Between Causal and Noncausal MCBD Algorithms

A comparison between the performance of the proposed algorithm and the FDMCBD algorithm of [21] is illustrated in Fig. 12 for two different reverberation times. To provide similar conditions, a demixing filter length of 512 taps was used for the FDMCBD algorithm (the equivalent of the choice of $N+1 = 256$ for the filters $\mathbf{L}(z)$ and $\mathbf{R}(z^{-1})$ in the proposed algorithm). As suggested in [21], to deal with nonminimum phase mixing systems, the diagonal filters of the demixing system are initialized with a center spike in the FDMCBD algorithm.

As can be seen in Fig. 12(a) and (b), both algorithms perform almost similarly when the sources are located at the opposite sides of the microphone array. But when both sources are located at the same side, our method clearly exhibits a superior performance as indicated by Fig. 12(c) and (d). In particular, for long reverberation times such as 300 ms, using the same filter length of 512 taps, the FDMCBD algorithm completely fails in achieving any degree of separation in the scenario of Fig. 4(a), whereas our method maintains a reasonable performance. Also shown in Fig. 12 is an explicit performance comparison for a simple FIR nonminimum phase mixing, whose zero distribution is shown in Fig. 13. Since nonminimum phase systems have noncausal (stable) inverses, a noncausal demixing filter is needed in this case. This is verified by the SIRs of Fig. 12(e).

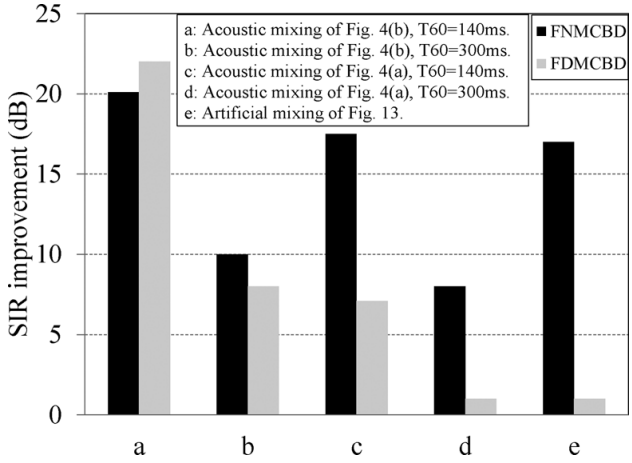


Fig. 12. Comparison of the separation performance (SIR improvement) of FNMCBD and FDMCBD algorithms. (a) causal BSS problem at $T_{60} = 140$ ms, (b) causal BSS problem at $T_{60} = 300$ ms, (c) noncausal BSS problem at $T_{60} = 140$ ms, (d) noncausal BSS problem at $T_{60} = 300$ ms, (e) artificial mixing of Fig. 13.

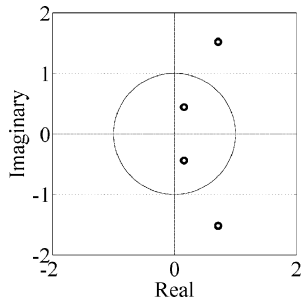


Fig. 13. Zero distribution of the artificial nonminimum phase mixing system.

While the FNMCBD algorithm reaches a satisfactory SIR of nearly 17 dB, the causal filters of FDMCBD algorithm fail to provide any degree of separation.

For further comparison, notice the spectrograms of Fig. 14, which have been plotted for 5 seconds of the original female speech, and the corresponding outputs of FNMCBD and FDMCBD algorithms for the situation of Fig. 4(a). As observed in the figure, the output spectrogram of the FNMCBD algorithm closely resembles that of the original source. In contrast, the output of the FDMCBD algorithm exhibits considerable spectral differences with the original source signal.

It can therefore be concluded that merely by center-spike initialization, causal MCBD algorithms cannot be forced to converge to a delayed form of the noncausal demixing system that is required. As a result, for a BSS solution with global applicability, we have to consider noncausal MCBD algorithms that train causal and anti-causal components of the demxing system separately.

VII. Conclusion

We presented a novel algorithm for efficient and fast implementation of noncausal multichannel blind deconvolution. A noncausal overlap-save procedure was developed for frequency domain filtering, and equivalent frequency domain update equations for both causal and anti-causal filters were derived.

The proposed algorithm offers a considerable saving in the number of operations required for filtering and adaptation.

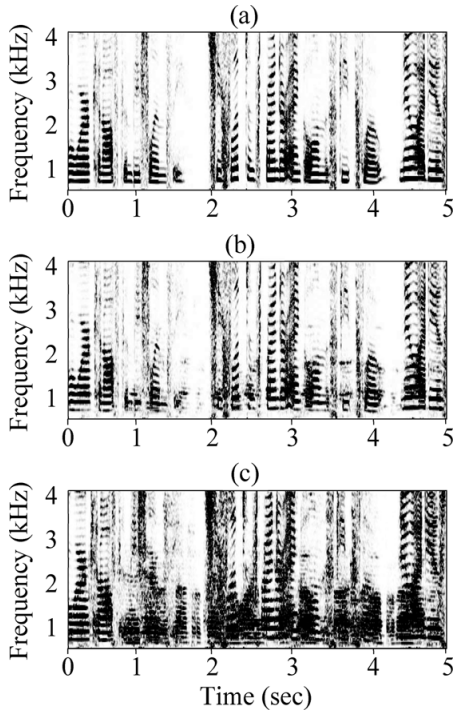


Fig. 14. Spectrogram plots for 5 seconds of original and recovered source signals. (a) original female speech, (b) output of FNMCBD algorithm, and (c) output of FDMCBD algorithm.

Our evaluations indicate that the computational complexity of the proposed algorithm rises slowly as the filter lengths are increased, unlike the exponential growth in the computational burden of the original time domain algorithm. The significance of these computational savings is revealed when the algorithm is used for blind source separation in realistic reverberant rooms, in which long demixing filters must be adapted by a large number of input samples. The extensive computational burden of the original time domain algorithm makes it infeasible for such situations. Furthermore, simulation results indicate the superior performance of the proposed algorithm over causal MCBD algorithms. It is shown through simulations that causal MCBD algorithms with center-spike initialization do not provide promising results for blind source separation in many possible arrangements of the sources and microphones, whereas the noncausal MCBD approach successfully achieves separation in such situations.

References

[1] P. Comon, "Independent component analysis, A new concept?," *Signal Process.*, vol. 36, no. 3, pp. 287–314, Apr. 1994.

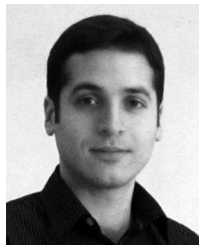
[2] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, no. 6, pp. 1129–1159, Nov. 1995.

[3] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," *Adv. Neural Inf. Process. Syst.*, vol. 8, pp. 757–763, 1995.

[4] D. T. Pham, "Blind separation of instantaneous mixtures of sources via an independent component analysis," *IEEE Trans. Signal Process.*, vol. 44, no. 11, pp. 2768–2779, Nov. 1996.

[5] S. Amari and A. Cichocki, "Adaptive blind signal processing-neural network approaches," *Proc. IEEE*, vol. 86, no. 10, pp. 2026–2048, Oct. 1998.

[6] D. T. Pham, "Blind separation of instantaneous mixtures of sources based on order statistics," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 363–375, Feb. 2000.
[7] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: Wiley , 2001, pp. 147–289.
[8] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," in *Proc. Int. ICSC Workshop Ind. Artif. Neural Netw.*, 1998, pp. 21–34.
[9] S. Ikeda and N. Murata, "A method of ICA in time-frequency domain," in *Proc. Int. Workshop Ind. Compon. Anal. Signal Separ.*, Aussois, France, 1999, pp. 365–370.
[10] H. Sawada, R. Mukai, S. Araki, and S. Makino, "Frequency-domain blind source separation," in *Speech Enhancement*. New York: Springer, 2005, pp. 299–327.
[11] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech Audio Process.*, vol. 12, no. 5, pp. 530–538, Sep. 2004.
[12] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Frequency domain blind source separation of many speech signals using near-field and far-field models," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–13, Jun. 2006.
[13] S. C. Douglas, H. Sawada, and S. Makino, "Natural gradient multichannel blind deconvolution and speech separation using causal FIR filters," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 13, no. 1, pp. 92–104, Jan. 2005.
[14] K. Kokkinakis and A. K. Nandi, "Multichannel blind deconvolution for source separation in convolutive mixtures of speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 200–212, Jan. 2006.
[15] X. Sun and S. C. Douglas, "A natural gradient convolutive blind source separation algorithm for speech mixtures," in *Proc. 3rd. IEEE Int. Workshop Ind. Compon. Anal. Source Separ.*, San Diego, CA, 2001, pp. 59–64.
[16] S. I. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, "Multichannel blind deconvolution and equalization using the natural gradient," in *Proc. IEEE Workshop Signal Process. Adv. Wireless Commun.*, Paris, France, 1997, pp. 101–104.
[17] S. Amari, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, no. 2, pp. 251–276, Feb. 1998.
[18] J.-F. Cardoso and B. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal Process.*, vol. 44, no. 12, pp. 3017–3030, Dec. 1996.
[19] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Process.*. New York: McGraw-Hill, 2005, pp. 54–56.
[20] S. C. Douglas, M. Gupta, H. Sawada, and S. Makino, "Spatio-temporal FastICA algorithm for the blind separation of convolutive mixtures," *IEEE Trans. Speech Audio Process.*, vol. 15, no. 5, pp. 1511–1520, Jul. 2007.
[21] M. Joho and P. Schniter, "Frequency domain realization of a multichannel blind deconvolution algorithm based on the natural gradient," in *Proc. Ind. Compon. Anal. Blind Signal Separ.*, Nara, Japan, 2003, pp. 543–548.
[22] L. Zhang, A. Cichocki, and S. Amari, "Multichannel blind deconvolution of nonminimum-phase systems using filter decomposition," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1430–1442, May 2004.
[23] K. Torkkola, "Blind separation for audio signals-are we there yet?," in *Proc. 1st Workshop Ind. Compon. Anal. Blind Signal Separ.*, Aussois, France, 1999, pp. 239–244.
[24] G. A. Clark, S. R. Parker, and S. K. Mitra, "A Unified approach to time and frequency domain realization of FIR adaptive digital filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 5, pp. 1073–1083, Oct. 1983.
[25] M. Joho and P. Schniter, "On frequency-domain implementations of filtered-gradient blind deconvolution algorithms," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2002, pp. 1653–1658.
[26] T. W. Lee, A. Ziehe, R. Orglmeister, and T. Sejnowski, "Combining time-delayed decorrelation and ICA: Towards solving the cocktail party problem," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Seattle, WA, 1998, pp. 1249–1252.
[27] A. C. Tsoi and L. S. Ma, "Flexible multichannel blind deconvolution, an investigation," in *Proc. IEEE 13th Workshop Neural Netw. Signal Process.*, Toulouse, France, 2003, pp. 349–358.
[28] R. M. Gray, *Toeplitz and Circulant Matrices: A Review*. Boston, MA: Now Publishers, 2005, vol. 2, Foundations and Trends in Communications and Information Theory, pp. 155–329.
[29] R. Aichner, "Acoustic blind source separation in reverberant and noisy environments," Ph.D. dissertation, Dept. Elect. Eng., Erlangen-Nurnberg Univ., Erlangen, Germany, 2007.
[30] W. M. Fisher, V. Zue, J. Bernstein, and D. Pallett, "An acoustic-phonetic database," in *Proc. 113th Meeting Acoust. Soc. Amer.*, May 1987.
[31] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. of Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.
[32] S. G. McGovern, "A model for room acoustics," 2003 [Online]. Available: http://2pi.us/rir.html
[33] R. Aichner, H. Buchner, and W. Kellermann, "On the causality problem in time-domain blind source separation and deconvolution algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Philadelphia, PA, 2005, vol. 5, pp. 181–184.
[34] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Trans. Speech Audio Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.



Seyedmahdad Mirsamadi (S'08) received the B.S. and M.S. degrees in electrical engineering from Amirkabir University of Technology, Tehran, Iran, in 2009 and 2011, respectively. He plans to pursue Ph.D. studies for a continued research in the field of signal processing.

During the M.S. degree, he joined the Multimedia Signal Processing Research Lab at Amirkabir University, where he worked on acoustic signal processing, with a focus on microphone arrays and multichannel signal processing. His main research interests include statistical signal processing, speech recognition and enhancement, adaptive signal processing, microphone array and multichannel signal processing and their applications to acoustic source localization, beamforming, and blind source separation.

Shabnam Ghaffarzadegan (S'09) received the B.S. and M.S. degrees in electrical engineering from Amirkabir University of Technology, Tehran, Iran, in 2009 and 2012, respectively. She is going to pursue Ph.D. studies in the signal processing field.

During the M.S. degree, she worked at the Multimedia Signal Processing Research Lab at Amirkabir University. During this period, she worked on digital signal processing with an emphasis on microphone arrays and multichannel signal processing. Her research interests include signal processing, speech processing, speech recognition, multirate and subband signal processing, adaptive signal processing, microphone array processing, blind source separation, and acoustic source localization.

Hamid Sheikhzadeh (M'03–SM'04) received the B.S. and M.S. degrees in electrical engineering from Amirkabir University of Technology, Tehran, Iran, in 1986 and 1989, respectively, and the Ph.D. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1994.

He was a faculty member in the Electrical Engineering Department, Amirkabir University of Technology, until September 2000. From 2000 to 2008, he was a Principle Researcher with ON semiconductor, Waterloo, ON, Canada. During this period, he developed signal processing algorithms for ultra-low-power and implantable devices leading to many international patents. Currently, he is a faculty member in the Electrical Engineering Department of Amirkabir University of Technology. His research interests include signal processing, biomedical signal processing and speech processing, with particular emphasis on speech recognition, speech enhancement, auditory modeling, adaptive signal processing, subband-based approaches, and algorithms for low-power DSP and implantable devices.

Seyed Mohammad Ahadi (M'87–SM'08) received the B.S. and M.S. degrees in electronics from the Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran, in 1984 and 1987, respectively, and the Ph.D. degree in engineering from the University of Cambridge, Cambridge, U.K., in 1996, in the field of speech processing.

In 1988, he was a Faculty Member with the Department of Electrical Engineering, Amirkabir University of Technology, where he began his teaching profession and became involved in research projects. Since receiving the Ph.D. degree, he has been with the same department, where he is currently teaching several courses and conducting research in electronics and communications. His current research interests include speech processing, acoustic modeling, robust speech recognition, speaker adaptation, speech enhancement, as well as audio and speech watermarking.

Amir Hossein Rezaie (M'09) graduated in electrical engineering from Amirkabir University (Tehran Polytechnic), Tehran, Iran, in 1983 and received the Ph.D. degree in engineering from Bristol University, Bristol, U.K., in 1988.

He is currently an Associate Professor in the Electrical Engineering Department, Amirkabir University. Apart from the teaching duties, he is also the head of the AIMS research lab at Amirkabir University. His main interests are in the field of automation, digital design and array processing. Some of the projects he has supervised have led to industrial applications, e.g., radar-based vehicle speed measurement systems with high-speed photography, synchronous phasor measurement unit (PMU), heat cost allocation systems, building management systems (BMS), and training simulators. He is also the author and coauthor of three technical books in Persian and many technical papers.