BLIND SEPARATION OF LINEAR CONVOLUTIVE MIXTURES USING ORTHOGONAL FILTER BANKS

Milutin Stanacevic, Marc Cohen and Gert Cauwenberghs

Department of Electrical and Computer Engineering
and Center for Language and Speech Processing
Johns Hopkins University, Baltimore MD 21218
E-mail: {milutin,marc,gert}@bach.ece.jhu.edu

ABSTRACT

We propose an algorithm and architecture for real-time blind source separation of linear convolutive mixtures using orthogonal filter banks. The adaptive algorithm derives from stochastic gradient descent optimization of a performance metric that quantifies independence not only across the reconstructed sources, but also across time within each source. The special case of a Laguerre section offers a compact representation with a small number of filter taps even under severe reverberant conditions, facilitating real-time implementation in a modular and scalable parallel architecture. Simulations of the proposed architecture and update rule validate the approach.

1. INTRODUCTION

The signal processing problem of separating and deconvolving observed mixtures of unknown independent sources without knowledge of the mixing medium, is known as blind source separation (BSS) or independent component analysis (ICA). The problem is addressed extensively in the literature and different algorithms for a wide range of applications in speech processing, wireless communications and biomedical signal processing exist.

BSS algorithms have been studied in the information-theoretic and statistical signal processing framework. Maximization of entropy of transformed output signals and minimization of mutual information of output signals are main approaches in deriving learning algorithms from information-theoretic perspective [1, 2, 3]. Maximum likelihood estimation (MLE) approach leads to same algorithms as infomax principle. In statistical signal processing, the contrast functions are chosen with respect to statistical measures of independances, i.e. cumulants and nonlinear moments [4, 5].

For linear convolutive mixtures, algorithms have been formulated in time and frequency domain based on the above

This work was supported by ONR YIP (N000149910612) and NSF Career (MIP-9702346).

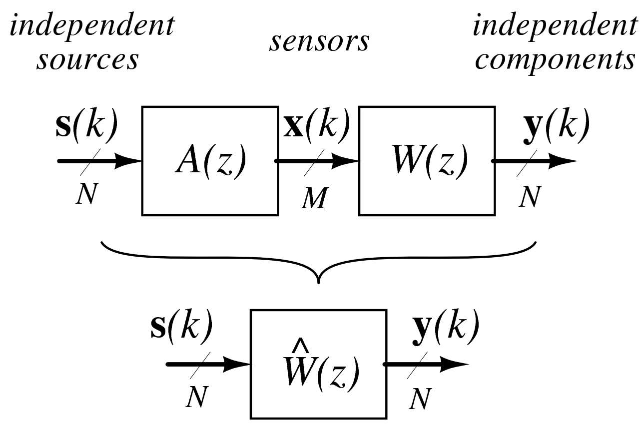

Fig. 1. *Problem Statement*

principles. Amari et. all [6] derived a time-domain algorithm based on a modified maximum entropy formulation. The same algorithm was obtained by Cohen and Cauwenberhgs [7] using nonlinear moments. If it can be assumed that the sources are non-stationary, variety of methods, based on the second-order statistics, can be used for separation [8, 9]. The formulation in the frequency-domain is computationaly more appealing, but the ICA indeterminacy in each frequency been has to be solved [10, 11]. There are also algorithms that combine two domains, with the separation criterion expressed in time-domain, while the rest is done in frequency-domain [12, 13].

Our objective is to reduce the complexity of algorithms by choosing an appropriate representation of the mixing medium. It has been shown that a Laguerre filterbank offers a versatile and compact filter basis for use in adaptive filtering applications [14, 15], and the work presented here extends the use of Laguerre and other orthogonal filter banks to the domain of Independent Component Analysis.

2. PROBLEM STATEMENT

Figure 1 schematizes the problem: unknown independent sources propagate through an unknown medium and are observed by an array of sensors. The task is to recover sources from observed signals using only the assumption that the source signals are mutually independent. The sensors inputs $\mathbf{x}(k)$ are convolutive mixtures of channel impulse response and input signals $\mathbf{s}(k)$

$$\mathbf{x}(k) = \sum_{p=0}^{+\infty} A(p)\mathbf{s}(k-p) \tag{1}$$

where $a_{ij}(p)$ denotes channel impulse response between source i and sensor j at time p. Matrix $A(p)$ is an MxN dimensional matrix, where N is the number of sources and M is number of sensors. The assumption is that $M > N$, since in the case of more sensors than sources prior information about sources is necessary for separation [16].

To recover the sources, the observed signals are processed by a transformation matrix W:

$$\mathbf{y}(k) = \sum_{p=0}^{+\infty} W(p)\mathbf{x}(k-p) = \sum_{q=0}^{+\infty} \hat{W}(q)\mathbf{s}(k-q) \tag{2}$$

where w_{ij} denotes the filter that is the inverted channel impulse response a_{ij} and $\hat{w}_{ij}$ is a total impulse response from source s_i to output signal y_j. We can rewrite the equations (1), (2) in the operator form [1]:

$$\mathbf{x}(k) = A(z)[\mathbf{s}(k)] \tag{3}$$

$$\mathbf{y}(k) = W(z)[\mathbf{x}(k)] = \hat{W}(z)[\mathbf{s}(k)] \tag{4}$$

with

$$W(z,k) = \sum_p W(p)z^{-p}$$
$$A(z) = \sum_p A(p)z^{-p}$$
$$\hat{W}(z,k) = W(z,k)A(z) \tag{5}$$

representing the z-transform of the channel, unmixing transformation and total impulse response, respectively. The following notation $z^{-p}[\mathbf{s}(k)] = \mathbf{s}(k-p)$ is used. We can formulated our aim as optimizing $W(z)$ such that

$$\lim_{k\to\infty} \hat{W}(z,k) = PD(z) \tag{6}$$

where P is NxN permutation matrix and $D(z)$ is a diagonal matrix, where diagonal entries represent delayed delta impulses $\delta(k-p)$.

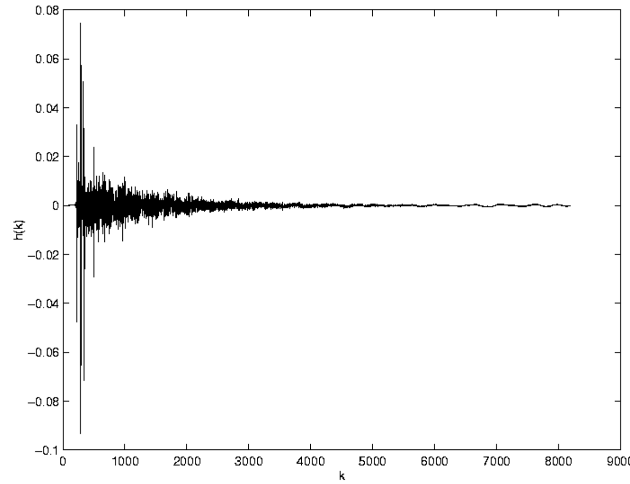

Fig. 2. *Room impulse response*

3. REPRESENTATION

The mixing matrix $A(z)$ represents the physics of propagation between sources and sensors. As an example, consider a sound source recorded in a room using a microphone. The recorded signal will consist of a direct (delayed) copy of the sound source and multi-path copies of signal, modified by the environment. The channel impulse response in this case is the room impulse response, which is dependent on reverberation and absorption characteristics of the room. An FIR filter representing typical room impulse response, as shown in Figure 2, requires a large number of delay elements (8192 in this case) [17]. This damped response can be compactly represented using a Laguerre filter bank, a cascade of a low-pass filter followed by identical all-pass filters. The transfer function of a Laguerre filter is given by

$$L_k(z) = H_0(z)[H_1(z)]^k, \quad k = 0, 1, 2... \tag{7}$$

with

$$H_0(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}} \qquad \text{and}$$
$$H_1(z) = \frac{z^{-1}-a}{1-az^{-1}}. \tag{8}$$

Parameter a represents the pole location, and for $a = 0$ the Laguerre filter is reduced to a simple delay line. Table 1 compares the mean square error between the 8192-tap FIR filter and Laguerre filter responses for different length of Laguerre filters and different pole location a. For $a = 0.5$ and filter length $= 1024$, the approximated response is shown in Figure 3.

To implement the inverting or unmixing matrix of filters, we can employ different structures: using FIR filters to approximate the inverse solution requires a large number of taps, and IIR adaptive filters can result in instability,

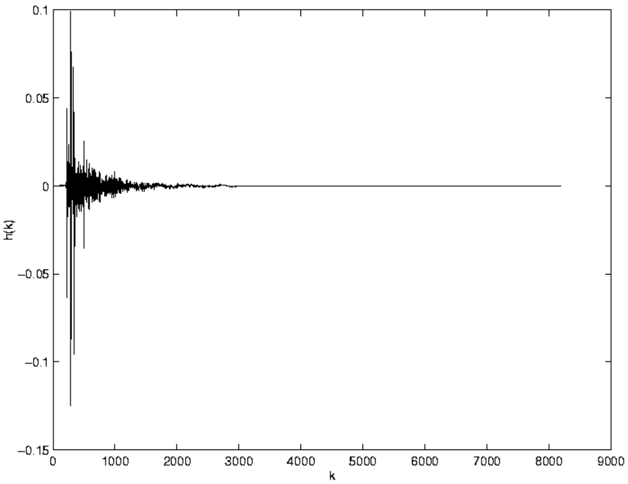

Fig. 3. *Approximation of the room impulse response of Figure 2 using a Laguerre filter bank with a = 0.5 and filter length = 1024.*

filter length	a = 0.2	a = 0.5	a = 0.7
256	8.1	8.2	8.7
512	6.4	7.9	8.6
1024	4.6	6.4	8.5
2048	2.8	4.8	8.1

Table 1. *Mean square error between the room impulse response and its approximation with Laguerre sections of finite length.*

especially if the impulse response has non-minimum phase. Laguerre filters provide local stability due to fixed poles, but still have advantages of IIR filter. As shown, the room impulse response can be represented by using fewer Laguerre filters, and therefore a lower number of filters for inverting the response.

4. ADAPTIVE SOURCE SEPARATION

The mixing coefficients $w_{ij}(n)$ can be expanded through a set of orthogonal functions ϕ_k:

$$w_{ij}(n) = \sum_{k=0}^{+\infty} c_{ij}(k)\phi_k(n) \tag{9}$$

with coefficients $c_{ij}(k)$ equal to

$$c_{ij}(k) = \sum_{n=0}^{+\infty} w_{ij}(n)\phi_k(n). \tag{10}$$

Similarly, the total impulse response is:

$$\hat{w}_{ij}(n) = \sum_{k=0}^{+\infty} \hat{c}_{ij}(k)\phi_k(n) \tag{11}$$

$$\hat{c}_{ij}(k) = \sum_{n=0}^{+\infty} \hat{w}_{ij}(n)\phi_k(n). \tag{12}$$

Substituting (11) in equation (2), we obtain:

$$\mathbf{y}(n) = \sum_{p=0}^{+\infty} \hat{C}(p)\Phi_p(z)[\mathbf{s}(n)]. \tag{13}$$

The cost function used as an optimization criterion is a scalar measure of output signal independence [5]

$$\mathcal{F} = \frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=-\infty}^{\infty}\left(E[y_i(k)y_j(k-l)] - \lambda\delta_{ij}\delta_{l0}\right)^2 \tag{14}$$

where E[ ] is the expectation operator and λ is a normalization constant. This cost function not only attempts to separate, but also to deconvolve (whiten) the outputs. For the simplicity of the derivation, we will also assume that the sources are white to start with, that is:

$$E[s_n(k-r)s_m(k-s)] = \delta_{nm}\delta_{rs} \tag{15}$$

which transforms (14) into

$$\mathcal{F} = \frac{1}{4}\sum_{i,j,l}\left(\sum_{n,p,q}\hat{c}_{in}(p)\hat{c}_{jn}(q)\phi_p(k)\phi_q(k-l) - \lambda\delta_{ij}\delta_{l0}\right)^2 \tag{16}$$

Gradient descent of (16) produces an update rule:

$$\begin{aligned}\Delta\hat{C}(q) &= -\mu\nabla_{\hat{C}}\mathcal{F}\\ &= \mu\Big(\lambda\hat{C}(q) \qquad (17)\\ &\quad -E\big[\sum_p \Phi_p(z)\,[\mathbf{y}(k)]\,\Phi_q(z)\,\big[\mathbf{y}^T(k)\hat{C}(p)\big]\,\big]\Big)\end{aligned}$$

where μ is the learning rate constant. A stochastic on-line weight adaptation rule is obtained by removing the expectation operator in (18) [5]. Independence of output signals beyond second-order statistics (removal of higher order cummulants) is obtained by applying component-wise antisymmetric nonlinear functions $f()$ and $g()$ [4, 5]. The selection of the functions $f()$ and $g()$ depends on the statistics of source signals and have been studied extensively in literature. Finally, substituting $\hat{C}(z) = C(z)A(z)$

$$\begin{aligned}\Delta\,C(q) &= \mu\Big(\lambda C(q) \qquad (18)\\ &- \sum_p \Phi_p(z)\,[f\,((\mathbf{y}(k)))]\,\Phi_q(z)\,\big[g(\mathbf{y}^T(k))C(p)\big]\,\Big)\end{aligned}$$

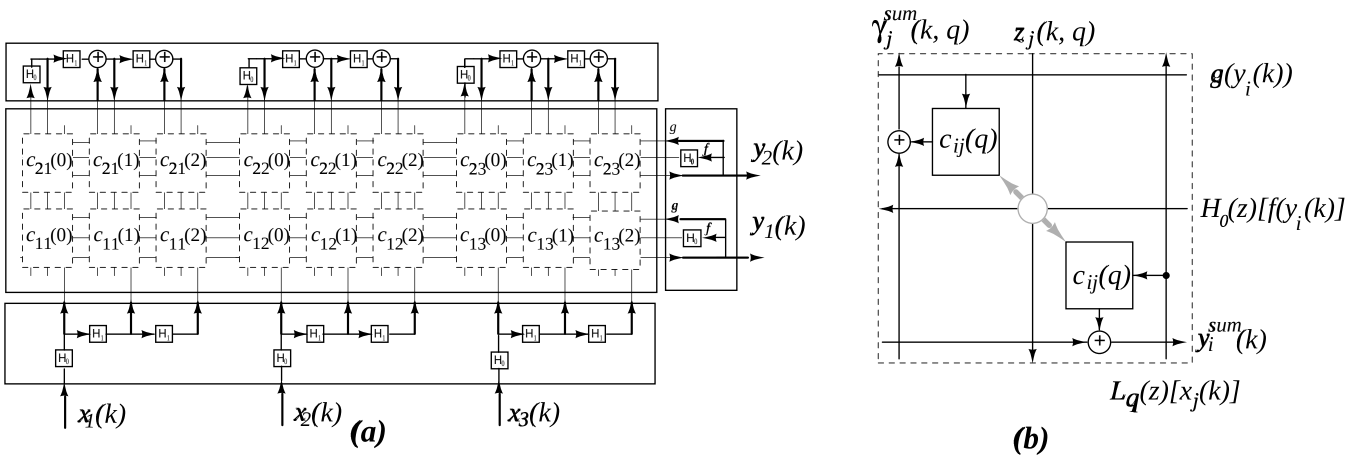

Fig. 4. *Parallel architecture:* **(a)** *example system block diagram for* $N = 2$, $M = 3$, *and* $L = 3$, **(b)** *unit cell diagram.*

When applying delay line for the filters $\Phi_p(z) = z^{-p}$ we retrieve the convolutive ICA algorithm derived in [6, 7].

4.1. Laguerre Filter Bank

Laguerre filter bank is a special case of orthogonal filter bank that offers compact representation. The update rule (18) for Laguerre filter becomes:

$$\begin{aligned} \Delta C(q) &= \mu\Big(\lambda C(q) \\ &- \sum_p L_p(z)\,[f(\mathbf{y}(k))]\,L_q(z)\,\big[g(\mathbf{y}^T(k))C(p)\big]\Big) \end{aligned} \tag{19}$$

which can be rewritten by rearranging contributions to the update over time as:

$$\begin{aligned} \Delta C(q) &= \mu\Big(\lambda C(q) \\ &- H_0(z)[f(\mathbf{y}(k))]\sum_p L_{q-p}(z)[g(\mathbf{y}^T(k))C(p)]\Big), \end{aligned} \tag{20}$$

The problem with this form is that it is non-causal. Whenever $q < p$, the update of weight $C(q)$ depends on the future outputs $\mathbf{y}(\mathbf{k}+\mathbf{p}-\mathbf{q})$. The problem can be solved by delaying the update on the RHS side of (21) [6]. We propose a modification of the rule, by omitting the noncausal terms for which $q < p$ [7]. Therefore (21) simplifies:

$$\begin{aligned} \Delta C(q) &= \mu\Big(\lambda C(q) \\ &- H_0(z)[f(\mathbf{y}(k))]\sum_{p=0}^{q} L_{q-p}(z)[g(\mathbf{y}^T(k))C(p)]\Big) \end{aligned} \tag{21}$$

For this learning rule, we propose the architecture shown in Figure 4a. An enlarged view of the unit cell is shown in Figure 4b. The sensor inputs $\mathbf{x}(k)$ are presented at the bottom of the system and fed to Laguerre filter banks. The signals from the filter banks on bottom are projected across the columns of the array. The outputs $\mathbf{y}(k)$ are obtained by summing across the rows, from left to right. The output signals are passed through nonlinearities $f(\,)$ and $g(\,)$, and they propagate along the rows from the rigth. The inner product of output signals and weights

$$\gamma_j(k,q) = \sum_i C_{ij}(q)\, g(y_i(k)) \tag{22}$$

is accumulated along the columns of array and fed into the filter bank on the top of system. The signals

$$z_j(k,q) = \sum_{p=0}^{q} L_{q-p}\gamma_j(k,p) \tag{23}$$

generated from the filter bank on top are projected along the columns and multiplied with the low-pass version of signal $f(\mathbf{y}(k)$ to generating the weight update.

The advantages of this architecture are local instantaneous computations, reduced complexity and scalability. The architecture lends itself to efficient implementation using either DSPs or custom parallel VLSI.

5. SIMULATIONS

We simulated our proposed architecture and learning rule in a small system: two input source and two outputs ($N = 2$) with Laguerre filter length $L = 6$. Inputs $s_1(k)$ and $s_2(k)$ are uniform white noise signals $\in [-1, 1]$. The weights $\hat{C}(q)$, $q = 0 \ldots L-1$, are initialized with uniform random weights $\in [-1, 1]$. For simple implementation, the functions f and g are the identity map $f(\mathbf{y}(t)) \equiv \mathbf{y}(t)$ and the signum function $g(\mathbf{y}^T(t)) \equiv \mathrm{sgn}(\mathbf{y}^T(t))$.

All simulation results are referenced to the sources, in terms of $\hat{C}$, because of the equivalence of the equations

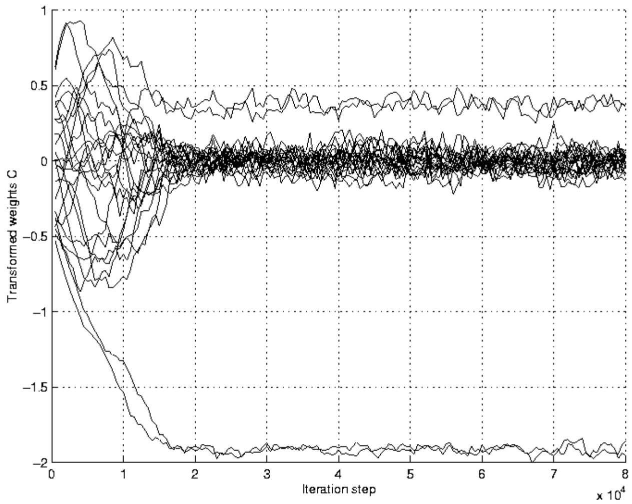

Fig. 5. *Trajectory of the coefficients* $\hat{c}_{ij}(q)$ *over time* k *for the triangularized update rule*

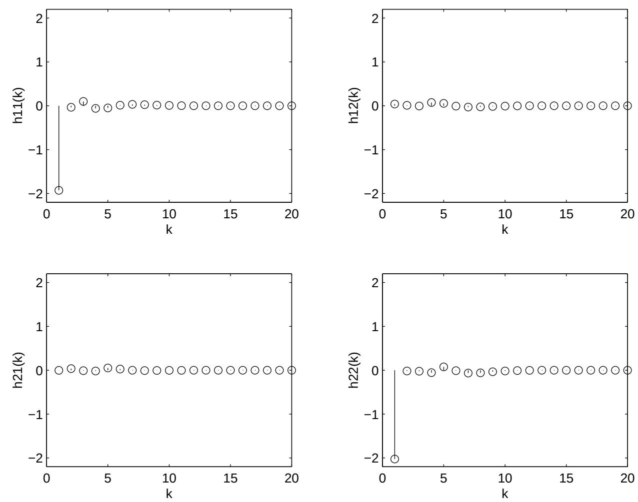

Fig. 6. *Impulse response of each filter after convergence*

of the learning updates under any transformtion $\hat{C}(z) = C(z)A(z)$. Figure 5 show the trajectories of all $2 \times 2 \times 6$ weights in $\hat{C}$ over time. Figure 6 shows the impulse responses of the 2×2 filters. It is clear that $y_1(k)$ corresponds to $-s_1(k)$ and $y_2(k)$ to $-s_2(k)$, which is one of many valid solutions to this unmixing/deconvolution task. The rate of convergence for the proposed architecture using Laguerre filters is approximately ten times faster than the same architecture using simple delay line. One interesting side effect of breaking time symmetry by ommitting the non-causal terms in the update rule is giving rise to a minimum phase response with minumum delay in the reconstruction of the sources.

6. CONCLUSION

In this paper, we have addressed the problem of blind source separation of linear convolutive mixtures using general orthogonal filter banks. The implementation using Laguerre filter banks offers a compact representation with reduced number of taps, and a faster convergence, compared with tapped delay line. Laguerre filters have a free parameter, the pole location, which can be optimized for a particular application. The proposed algorithm can be efficiently implemented in a scalable parallel architecture, with local updates.

7. REFERENCES

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

[2] A. Bell and T. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.

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

[4] J. Herault and J. Jutten, "Space or time adaptive signal processing by neural network models," in *Neural Networks for Computing: AIP conference proceedings 151*, J.S. Denker, Ed., New York, 1986, American Institute of Physics.

[5] A. Cichocki and R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation of sources," *IEEE Trans. Circuits and Systems-I: Fundamental Theory and Applications*, vol. 43, no. 11, pp. 894–906, 1996.

[6] S. Amari, S. Douglas, A. Cichocki, and H.H. Yang, "Novel on-line adaptive learning algorithms for blind deconvolution using the natural gradient algorithm," in *Proc. of 11th IFAC Symposium on System Identification*, Kitakyushu City, Japan, July 1997, pp. 1057–1062.

[7] M. Cohen and G. Cauwenberghs, "Blind separation of linear convolutive mixtures through parallel stochastic optimization," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS'1998)*, 1998, vol. 3, pp. 17–20.

[8] E. Weinstein, M. Feder, and A. Oppenheim, "Multi-channel signal separation by decorrelation," *IEEE*

Trans. on Speech and Audio Processing, vol. 1, no. 4, pp. 405–413, 1993.

[9] L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Transactions Speech and Audio Processing*, pp. 320–327, 2000.

[10] V. Capdevielle, C. Serviere, and J. Lacoume, "Blind separation of wide band sources in the frequency domain," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'95)*, 1995, pp. 2080–2083.

[11] P. Smaragdis, "Blind separation of convolved sound mixtures in the frequency domain," in *Proc. of Int. Workshop on Independence and Artificial Neural Networks*, Tenerife, Spain, 1998.

[12] R. Lambert and A. Bell, "Blind separation of multiple speakers in a multipath environment," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'97)*, Munich, 1997.

[13] T-W. Lee, A. Bell, and R. Orglmeister, "Blind source separation of real world signals," in *Proceedings of International Conference on Neural Networks*, Houston, 1997.

[14] J.G. Harris, J.K. Juan, and H.C. Principe, "Analog hardware implementation of adaptive filter structures," in *Proceedings of International Conference on Neural Networks*, Houston, 1997.

[15] J.C. Principe, N.R. Euliano, and W.C. Lefebvre, *Neural and Adaptive Systems: Fundamentals through Simulations*, John Wiley & Sons, 1999.

[16] A. Gorokhov and P. Loubaton, "Subspace-based techniques for blind separation of convolutive mixtures with temporally correlated sources," *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 9, pp. 812–820, 1997.

[17] A. Westner and V.M. Bove, "Blind separation of real world audio signals using overdetermined mixtures," in *Proceedings of ICA*, Aussois, France, 1999.