

A Gradient-Based Optimum Block Adaptation ICA Technique for Interference Suppression in Highly Dynamic Communication Channels

Wasfy B. Mikhael¹ and Tianyu Yang²

¹Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816, USA

²Department of Engineering Sciences, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA

Received 21 February 2005; Revised 30 January 2006; Accepted 18 February 2006

The fast fixed-point independent component analysis (ICA) algorithm has been widely used in various applications because of its fast convergence and superior performance. However, in a highly dynamic environment, real-time adaptation is necessary to track the variations of the mixing matrix. In this scenario, the gradient-based online learning algorithm performs better, but its convergence is slow, and depends on a proper choice of convergence factor. This paper develops a gradient-based optimum block adaptive ICA algorithm (OBA/ICA) that combines the advantages of the two algorithms. Simulation results for telecommunication applications indicate that the resulting performance is superior under time-varying conditions, which is particularly useful in mobile communications.

Copyright © 2006 Hindawi Publishing Corporation. All rights reserved.

1. INTRODUCTION

Independent component analysis (ICA) is a powerful statistical technique that has a wide range of applications. It has attracted huge research efforts in areas such as *feature extraction* [1], *telecommunications* [2–4], *financial engineering* [5], *brain imaging* [6], and *text document analysis* [7]. ICA can extract statistically independent components from a set of observations that are linear combinations of these components.

The basic ICA model is $\mathbf{X} = \mathbf{AS}$. Here, \mathbf{X} is the observation matrix, \mathbf{A} is the mixing matrix, and \mathbf{S} is the source signal matrix consisting of independent components. The objective of ICA is to find a separation matrix \mathbf{W} , such that \mathbf{S} can be recovered when the observation matrix \mathbf{X} is multiplied by \mathbf{W} . This is achieved by making each component in \mathbf{WX} as independent as possible. Many principles and corresponding algorithms have been reported to accomplish this task, such as *maximization of nongaussianity* [8, 9], *maximum likelihood estimation* [10, 11], *minimization of mutual information* [12, 13], and *tensorial methods* [14–16].

The Newton-based fixed-point ICA algorithm [8], also known as the fast-ICA, is a highly efficient algorithm. It typically converges within less than ten iterations in a stationary environment. Moreover, in most cases the choice of the learning rate is avoided. However, when the mixing matrix is highly dynamic, fast-ICA cannot successfully track the time

variation. Thus, a gradient-based algorithm is more desirable in this scenario.

The previously reported online gradient-based algorithm [17, page 177] suffers from slow convergence and difficulty in the choice of the learning rate. An improper choice of the learning rate, which is typically determined by trial and error, can result in slow convergence or divergence. In the adaptive learning and neural network area, many research efforts have been devoted to the selection of learning rate in an intelligent way [18–23]. In this paper, we propose a gradient-based block ICA algorithm OBA/ICA, which automatically selects the optimal learning rate.

ICA has been previously proposed to perform blind detection in a multiuser scenario. In [2, 24], Ristaniemi and Joutsensalo proposed to use fast-ICA as a tuning element to improve the performance of the traditional RAKE or MMSE DS-CDMA receivers. Other techniques exploiting antenna diversity have also been presented for interference suppression [25, 26] or multiuser detection [27]. These ICA-based approaches have attractive properties, such as near-far resistance and little requirement on channel parameter estimation. In this contribution, the new OBA/ICA algorithm is applied for baseband interference suppression in diversity BPSK receivers. Simulation results confirm OBA/ICA's effectiveness and advantage over the existing fast-ICA algorithm in highly dynamic channels. Naturally, OBA/ICA is still useful for slowly time-varying or stationary channels.

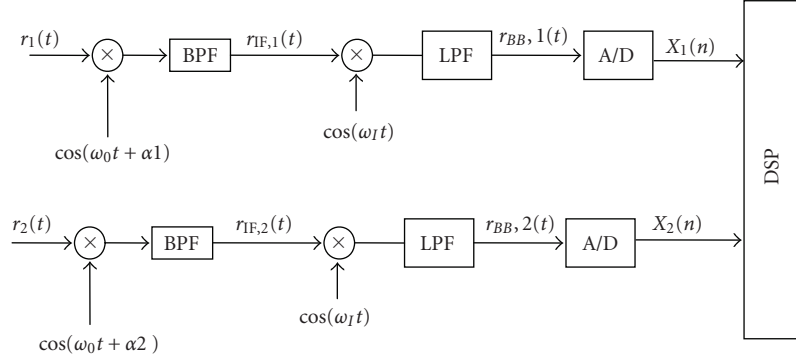


FIGURE 1: Diversity BPSK wireless receiver structure with ICA interference suppression.

The rest of the paper is organized as follows. Section 2 presents the system model for diversity BPSK receiver structure. Section 3 discusses the motivation and basic strategy of OBA/ICA. Section 4 formulates OBA/ICA, and it is also shown that OBA/ICA reduces to online gradient ICA in the simplest case. Section 5 deals with several practical implementation issues regarding OBA/ICA. Section 6 applies OBA/ICA for interference suppression in mobile communications assuming two different types of time-varying channels, and the performance is compared with fast-ICA. Finally, conclusions are given in Section 7.

2. SIGNAL MODEL FOR DIVERSITY BPSK RECEIVERS

Figure 1 shows the simplified structure of a dual-antenna diversity BPSK receiver. We assume the image signal is the primary interferer to be suppressed. The extension to the cases of multiple interferers and/or cochannel interference (CCI) is straightforward, and it is accomplished by the addition of antenna elements. For each receiver processing chain, the received signal is first downconverted from RF to IF, followed by a bandpass filter to perform adjacent channel suppression. Then, the IF signal $r_{IF}(t)$ is downconverted to baseband and lowpass filtered. The baseband signal $r_{BB}(t)$ is digitized to obtain the signal observation $\mathbf{X}(\mathbf{n})$, which is fed into the digital signal processor (DSP) for further processing.

In our signal analysis, frequency-flat fading is assumed. For the k th antenna ($k = 1, 2$), the channel's fading coefficients for the desired signal $s(t)$ and the image signal $i(t)$ are defined as

$$\begin{aligned} f_{sk} &= \alpha_{sk} e^{j\psi_{sk}}, \\ f_{ik} &= \alpha_{ik} e^{j\psi_{ik}}, \end{aligned} \quad (1)$$

where α_{sk} , α_{ik} and ψ_{sk} , ψ_{ik} are the channel's amplitude and phase responses, respectively. The distributions of α_{sk} and α_{ik} are determined by the type of fading channels the signals encounter. Since the signals travel random paths, ψ_{sk} and ψ_{ik} can be modeled as uniformly distributed random phases over the interval $[0, 2\pi)$.

The received signal from the k th antenna, $r_k(t)$, can be expressed as

$$r_k(t) = 2 \operatorname{Re} \{s(t) f_{sk} e^{j(\omega_0 + \omega_I)t} + i(t) f_{ik} e^{j(\omega_0 - \omega_I)t}\}, \quad (2)$$

where $\operatorname{Re}\{\cdot\}$ denotes the real part of a signal, ω_0 and ω_I denote the frequency of the first and the second local oscillators (LO). The multiplication by 2 is introduced for convenience.

After the RF-IF downconversion, the bandpass filtered signal is given by

$$\begin{aligned} r_{IF,k}(t) &= s(t) f_{sk} e^{-j\alpha} e^{j\omega_I t} + s^*(t) f_{sk}^* e^{j\alpha} e^{-j\omega_I t} \\ &+ i(t) f_{ik} e^{-j\omega_I t} e^{-j\alpha} + i^*(t) f_{ik}^* e^{j\omega_I t} e^{j\alpha}, \end{aligned} \quad (3)$$

where the superscript $*$ denotes complex conjugate, and α is the phase difference between the received signal and the first LO signal.

The baseband signal after downconversion to baseband and lowpass filtering is expressed as

$$r_{BB,k}(t) = \operatorname{Re} \{s(t) f_{sk} e^{-j\alpha}\} + \operatorname{Re} \{i(t) f_{ik} e^{-j\alpha}\}. \quad (4)$$

For BPSK signals, $s(t)$ and $i(t)$ are real-valued, so (4) can be written as

$$r_{BB,k}(t) = a_k s(t) + b_k i(t), \quad (5)$$

where the coefficients $a_k = \operatorname{Re}\{f_{sk} e^{-j\alpha}\}$, and $b_k = \operatorname{Re}\{f_{ik} e^{-j\alpha}\}$.

Thus, after A/D converter, the baseband observation is

$$X_k(n) = a_k s(n) + b_k i(n). \quad (6)$$

Each of $s(n)$, $i(n)$, and $X_k(n)$ in (6) represents a one sample signal. Since the signals are processed in frames of length N , s_N , i_N , and $X_{N,k}$ are used to represent frames of N successive samples. Hence,

$$X_{N,k} = a_k s_N + b_k i_N. \quad (7)$$

Therefore, the baseband signal observation matrix is expressed as

$$X = \begin{bmatrix} X_{N,1} \\ X_{N,2} \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} s_N \\ i_N \end{bmatrix} = AS. \quad (8)$$

In system model (8), X is the 2 by N observation matrix, A is the unknown 2 by 2 mixing matrix, and S is the 2 by N source signal matrix, which is to be recovered by ICA algorithm based on the assumption of statistical independence between the desired signal and the interferer. From the above derivation process, it is clear that the mixing matrix is determined by the wireless channel's fading coefficients, which are often time varying. ICA requires that the mixing matrix should be nonsingular, and this is guaranteed due to the randomness of the wireless channel. ICA poses no requirement regarding the relative strength of the source signals, so the operating range for input signal-to-interference ratio (SIR) is quite large. However, in practice, if the interference is too strong, the front-end synchronization becomes problematic. Therefore, there are practical limitations to the application of the proposed technique.

ICA processing has the inherent order ambiguity. Therefore, reference sequences need to be inserted into source signals for the receiver to identify the desired user. Fortunately, in most communication standards, such reference sequences are available.

In this paper, we are primarily concerned about the interference-limited scenario. Therefore, thermal noise is not explicitly included in the signal model. However, ICA algorithm is able to perform successfully in the presence of thermal noise. In Section 6, simulation results will be presented with thermal noise included.

3. BACKGROUND AND MOTIVATIONS

The fast-ICA algorithm is a block algorithm. It uses a block of data to establish statistical properties. Specifically, the "expectation" operator is estimated by the average over L data points, where L is the block size [8]. The performance is better when the estimation is more accurate, that is, L is larger. However, it is very important that the mixing matrix stays approximately constant within one processing block, that is, quasistationary. Thus, the problem with convergence arises when the mixing matrix is rapidly time varying, in which case a large L violates the assumption of quasistationarity.

On the other hand, the online gradient-based algorithm, which updates the separation matrix once for every received symbol, can better track the time variation of the mixing matrix. But it directly drops the "expectation" operator, which results in worse performance than a block algorithm.

Therefore, an algorithm is needed that can better accommodate time variations by processing signals in blocks and automatically selecting the optimal convergence factor. In the following section, such a technique is developed, which is denoted OBA/ICA.

The idea is to tailor the learning rates in a gradient-based block algorithm to each iteration and every coefficient in the

separation matrix, in order to maximize a performance function that corresponds to a measure of independence. In [28], Mikhael and Wu used a similar idea to develop a fast block-LMS adaptive algorithm for FIR filters, which proved to be useful, especially when adapting to time-varying systems.

4. FORMULATION OF OBA/ICA

The algorithm developed here is used for estimating one row, w , of the demixing matrix W . The algorithm is run for all rows. The performance function adopted is the absolute value of kurtosis. Other ICA-related operations, such as mean centering, whitening, and orthogonalization, are identical as fast-ICA. First, the following parameters are defined:

- (i) j : iteration index,
- (ii) M : number of observations,
- (iii) L : length of the processing block,
- (iv) $\underline{w}(j) = [w_1(j), w_2(j), \dots, w_M(j)]^T$: the current row of the separation matrix for the j th iteration. ($i = 1, 2, \dots, M$),
- (v) $x_{l,i}(j)$: the i th signal in the l th observation data vector for the j th iteration. ($l = 1, 2, \dots, L$),
- (vi) $\underline{X}_l(j) = [x_{l,1}(j), x_{l,2}(j), \dots, x_{l,M}(j)]^T$: l th signal observation for the j th iteration,
- (vii) $[G]_j = [X_1(j), X_2(j), \dots, X_L(j)]^T$: observation matrix for the j th iteration.

The l th kurtosis value for the j th iteration is

$$\text{kurt}_l(j) = E\left\{[\underline{w}^T(j)\underline{X}_l(j)]^4\right\} - 3, \quad (9)$$

where it is assumed that the signals and $\underline{w}(j)$ both have been normalized to unit variance.

Then, the kurtosis vector for the j th iteration is

$$\underline{\text{kurt}}(j) = [\text{kurt}_1(j), \text{kurt}_2(j), \dots, \text{kurt}_L(j)]^T. \quad (10)$$

Now the updating formula can be written in a matrix-vector form as

$$\underline{w}(j-1) = \underline{w}(j) - [MU]_j \nabla_B(j), \quad (11)$$

where

$$\begin{aligned} \nabla_B(j) &= \frac{\partial\{\underline{\text{kurt}}^T(j)\underline{\text{kurt}}(j)\}}{\partial\underline{w}(j)} \\ &= \frac{1}{L} \left[\frac{\partial\{\underline{\text{kurt}}^T(j)\underline{\text{kurt}}(j)\}}{\partial w_1(j)} \quad \dots \quad \frac{\partial\{\underline{\text{kurt}}^T(j)\underline{\text{kurt}}(j)\}}{\partial w_M(j)} \right]^T, \end{aligned} \quad (12)$$

$$[MU]_j = \begin{bmatrix} \mu_{B1}(j) & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & \mu_{BM}(j) \end{bmatrix}. \quad (13)$$

Note that in (11), a “+” sign is used instead of “-” as in the steepest descent algorithm. Because our performance function is the absolute value of kurtosis rather than error signal, we wish to maximize the function to achieve maximal non-Gaussianity.

To evaluate (12), we have

$$\begin{aligned} & \frac{\partial \{\text{kurt}^T(j)\text{kurt}(j)\}}{\partial w_i(j)} \\ &= \sum_{l=1}^L \frac{\partial [E\{[w^T(j)x_l(j)]^4\} - 3]^2}{\partial w_i(j)} \\ &= 8 \sum_{l=1}^L [w^T(j)X_l(j)]^3 \text{kurt}_l(j)x_{l,i}(j). \end{aligned} \quad (14)$$

In the derivation of (14), the expectation operator was dropped.

The block gradient vector can be written as

$$\begin{aligned} \nabla_B(j) &= \frac{8}{L} \left[\sum_{l=1}^L [w^T(j)X_l(j)]^3 \text{kurt}_l(j)x_{l,1}(j) \cdots \right. \\ & \quad \left. \sum_{l=1}^L [w^T(j)X_l(j)]^3 \text{kurt}_l(j)x_{l,M}(j) \right]^T \\ &= \frac{8}{L} [G]_j^T [C]_j^3 \text{kurt}(j), \end{aligned} \quad (15)$$

where

$$[C]_j = \begin{bmatrix} \frac{w^T(j)X_1(j)}{L} & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & \frac{w^T(j)X_L(j)}{L} \end{bmatrix} \quad (16)$$

is a diagonal matrix.

From (15), the updating formula (11) becomes

$$\underline{w}(j+1) = \underline{w}(j) + \frac{8}{L} [\text{MU}]_j [G]_j^T [C]_j^3 \text{kurt}(j). \quad (17)$$

Now, the primary task is to identify the matrix $[\text{MU}]_j$ in an optimal sense, so that the total squared kurtosis $\text{kurt}^T(j)\text{kurt}(j)$ is maximized. In order to do that, we express the l th kurtosis value in the $(j+1)$ th iteration by Taylor's series expansion:

$$\begin{aligned} \text{kurt}_l(j+1) &= \text{kurt}_l(j) \\ &+ \sum_{i=1}^M \frac{\partial \text{kurt}_l(j)}{\partial w_i(j)} \Delta w_i(j) \\ &+ \frac{1}{2!} \sum_{m=1}^M \sum_{n=1}^M \frac{\partial^2 \text{kurt}_l(j)}{\partial w_m(j) \partial w_n(j)} \Delta w_m(j) \Delta w_n(j) \\ &+ \cdots, \quad l = 1, 2, \dots, L, \end{aligned} \quad (18)$$

where

$$\Delta w_i(j) = w_i(j+1) - w_i(j), \quad i = 1, 2, \dots, M. \quad (19)$$

In (18), the complexity of the terms increases as the order of the derivative increases. However, if $\Delta w_i(j)$ is small enough, higher-order derivative terms can be omitted. In our experimentation, it is found that this is indeed the case.

The expectation operator in (9) is dropped. Thus,

$$\frac{\partial \text{kurt}_l(j)}{\partial w_i(j)} = 4x_{l,i}(j) [w^T(j)X_l(j)]^3. \quad (20)$$

Then, (18) becomes

$$\begin{aligned} \text{kurt}_l(j+1) &= \text{kurt}_l(j) + 4 [w^T(j)X_l(j)]^3 \sum_{i=1}^M x_{l,i}(j) \Delta w_i(j) \\ &= \text{kurt}_l(j) + 4 [w^T(j)X_l(j)]^3 [X_l^T(j) \Delta w(j)]. \end{aligned} \quad (21)$$

Writing (21) for every l , the matrix-vector form of the Taylor expansion becomes

$$\underline{\text{kurt}}(j+1) = \underline{\text{kurt}}(j) + 4 [C]_j^3 [G]_j \underline{\Delta w}(j). \quad (22)$$

From (17),

$$\underline{\Delta w}(j) = \frac{8}{L} [\text{MU}]_j [G]_j^T [C]_j^3 \text{kurt}(j). \quad (23)$$

Substituting (23) into (22), one obtains

$$\underline{\text{kurt}}(j+1) = \underline{\text{kurt}}(j) + \frac{32}{L} [C]_j^3 [G]_j [\text{MU}]_j [G]_j^T [C]_j^3 \text{kurt}(j). \quad (24)$$

Defining $\underline{q}(j)$ and $[R]_j$ as

$$\underline{q}(j) = [G]_j^T [C]_j^3 \text{kurt}(j) = [q_1(j), \dots, q_M(j)]^T, \quad (25)$$

$$[R]_j = [G]_j^T [C]_j^6 [G]_j = [R_{mn}(j)], \quad 1 \leq m, n \leq M. \quad (26)$$

The total squared kurtosis for the $(j+1)$ th iteration can be written as

$$\underline{\text{kurt}}^T(j+1)\text{kurt}(j+1) = S_1 + S_2 + S_3, \quad (27a)$$

where

$$S_1 = \underline{\text{kurt}}^T(j)\text{kurt}(j), \quad (27b)$$

$$S_2 = \frac{64}{L} \sum_{i=1}^M q_i^2(j) \mu_{Bi}(j), \quad (27c)$$

$$S_3 = \frac{1024}{L^2} \underline{q}^T(j) [\text{MU}]_j [R]_j [\text{MU}]_j \underline{q}(j). \quad (27d)$$

In order to identify $[\text{MU}]_j$ optimally, the following condition

must be met:

$$\frac{\partial\{\text{kurt}^T(j+1)\text{kurt}(j+1)\}}{\partial\mu_{Bi}(j)} = 0, \quad i = 1, 2, \dots, M. \quad (28)$$

Combining (27a) and (28) yields

$$\frac{\partial S_1}{\partial\mu_{Bi}(j)} + \frac{\partial S_2}{\partial\mu_{Bi}(j)} + \frac{\partial S_3}{\partial\mu_{Bi}(j)} = 0. \quad (29)$$

Substituting (27b), (27c), and (27d) into (29), and using the symmetry property of the matrix $[R]_j$ given in (26), the following is obtained:

$$\sum_{k=1}^M [q_k(j)\mu_{Bk}^*(j)r_{ki}(j)] = -\frac{L}{32}q_i(j), \quad (30)$$

where $*$ denotes the optimal value.

Writing (30) for every i , the following matrix-vector equation is obtained:

$$[R]_j[\text{MU}]_j^* \underline{q}(j) = -\frac{L}{32}\underline{q}(j). \quad (31)$$

From (31), we have

$$[\text{MU}]_j^* \underline{q}(j) = -\frac{L}{32}[R]_j^{-1}\underline{q}(j). \quad (32)$$

From (25), (32), and (17), the OBA/ICA algorithm is obtained:

$$\begin{aligned} \underline{w}(j+1) &= \underline{w}(j) + \frac{8}{L}\left(-\frac{L}{32}\right)[R]_j^{-1}\underline{q}(j) \\ &= \underline{w}(j) - 0.25[R]_j^{-1}\underline{q}(j), \end{aligned} \quad (33)$$

where $[R]_j$ and $\underline{q}(j)$ are given by (25) and (26).

Now we show that online gradient-based ICA can be obtained as a special case of the more general OBA/ICA formulation presented above. Let $L = 1$ and let $\mu_{B1}(j) = \mu_{B2}(j) = \dots = \mu_{BM}(j) = \mu_B(j)$, then OBA/ICA simplifies to

$$\underline{w}(j+1) = \underline{w}(j) - 0.25\mu_B^*(j)\underline{X}(j)[\underline{w}^T(j)\underline{X}(j)]^3 \text{kurt}(j), \quad (34)$$

where

$$\mu_B^*(j) = \frac{1}{[\underline{w}^T(j)\underline{X}(j)]^6 [\underline{X}^T(j)\underline{X}(j)]}. \quad (35)$$

If we let $\mu = 0.25\mu_B^*(j)|\text{kurt}(j)|$, the online gradient-based ICA is obtained [17, page 177]:

$$\underline{w}(j+1) = \underline{w}(j) - \mu \left(\text{sign}[\text{kurt}(j)]\underline{X}(j)[\underline{w}^T(j)\underline{X}(j)]^3 \right). \quad (36)$$

5. IMPLEMENTATION ISSUES

5.1. Elimination of the matrix inversion operation

OBA/ICA algorithm, (33), gives the optimal updating formula to extract one row of the separation matrix \mathbf{W} . The

update equation, (33), involves the inversion of the $[R]$ matrix, whose dimensionality is equal to the order of the system M . This operation could be inefficient in the case of a high-order system. This is because the computational complexity of the matrix inversion operation is $O(M^3)$. When M is large, an estimate of $[R]$ can be used. The method proposed here is to use a diagonal matrix $[R]_D$ which contains only the diagonal elements of $[R]$. Thus, the complexity of the inverse operation becomes $O(M)$. From extensive simulations, it is found that the adaptive system repairs itself from this approximation and converges to the right solution in a few additional iterations.

5.2. Computational complexity

Having eliminated the inversion problem, the dominant factor determining the computational complexity is the block size L for most applications of ICA. L is typically larger than the order of the system M . It is easily seen that the number of multiplications and divisions of OBA/ICA is $O(L)$ per iteration, which is equivalent to fast-ICA.

5.3. An optional scaling constant

In practice, a parameter k can be introduced in (33) to further optimize the algorithm performance if a priori information is available regarding the speed of time variation of the channel. Also, since the high-order derivative terms in (18) are dropped in our formulation, an additional adaptation parameter can help to ensure reliable convergence. However, the value of k is not critical, and the algorithm successfully converges over a wide range of k , as is confirmed by our simulations.

Therefore, the optimized updating formula is obtained based on (33) as

$$\underline{w}(j+1) = \underline{w}(j) - 0.25k[R]_j^{-1}\underline{q}(j), \quad (37)$$

where the choice of k is made according to the convergence property and the speed of mixing matrix's time variation.

5.4. Types of time variations

In our simulations two types of time variations are studied, which correspond to two scenarios that can arise in mobile communication applications.

In the first case, the change of the channel is modeled as a continuous linear time variation in the mixing matrix's coefficients. In this case, the ICA algorithm seeks a compromise separation matrix that recovers the source signals with minimum error.

The second type of time variation arises when the user is experiencing handover between two service towers. In this scenario, the mixing matrix's coefficients are modeled by an abrupt change. Note that the ICA processing will only be affected when the abrupt change occurs within one processing block. This is the case studied in our simulation.

When an abrupt change occurs within a processing block, the performance for the block degrades significantly, especially when the block size is large. This is because the converged demixing vector is a compromise between two completely different channel parameters. In order to deal with this situation, we propose to locate the position of the abrupt change within the block. This technique will improve the performance if the performance degradation is due to an abrupt change within the block.

In the search procedure, the demixing matrices obtained through the previous block $\mathbf{W1}$ and the subsequent block $\mathbf{W2}$ are utilized.

First, the block is evenly divided into two subblocks. $\mathbf{W1}$ is used to process the first subblock, while $\mathbf{W2}$ is used to process the second subblock.

If the separation performance for the second subblock is better, it is concluded that the abrupt change occurs within the first subblock. Otherwise, it is concluded that the abrupt change occurs within the second subblock.

Thus, the location of the abrupt change is narrowed down to a subblock. The search process can be continued by dividing that subblock evenly and using $\mathbf{W1}$ and $\mathbf{W2}$ to process the two subblocks, respectively. This procedure can be repeated until the location of the abrupt change is narrowed down to a very small range.

Once the location is identified, the symbols before the abrupt change are processed by $\mathbf{W1}$, and the symbols after the abrupt change are processed by $\mathbf{W2}$.

6. APPLICATION IN MOBILE TELECOMMUNICATIONS

To study the performance of OBA/ICA, computer simulations are performed. The performance measures are the signal-to-interference ratio (SIR) and the number of iterations to convergence N_c . SIR represents the average ratio of the desired signal power to the power of the estimation error, defined as

$$\text{SIR} = 10 \log_{10} \left(\frac{1}{L} \sum_{k=1}^L \frac{s(k)^2}{[s(k) - y(k)]^2} \right), \quad (38)$$

where $s(k)$ is the k th sample of the desired signal, $y(k)$ is the estimate of the $s(k)$ obtained at the output of the ICA processing unit.

For continuous linear time variation, the mixing matrix simulated is chosen as

$$\mathbf{A} = \begin{bmatrix} 1 + l\Delta & 0.5 \\ 0.7 & 2 + l\Delta \end{bmatrix}, \quad (39)$$

where $l = 1, 2, \dots, L$, and Δ is the parameter reflecting the speed of channel variation. Here, it is assumed that the channel's transfer function is frequency-flat over the signal band. Also, the sampling interval of the receiver's A/D converter is negligible compared with $1/\Delta$, which represents the rate of the channel's time variation.

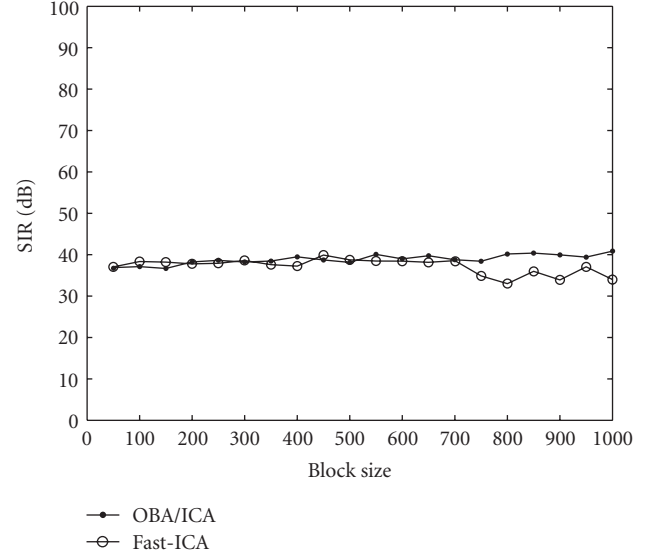


FIGURE 2: Signal-to-interference ratio (SIR) achieved in dB versus the processed block size employing fast-ICA and OBA/ICA ($k=0.5$) when channel conditions vary linearly with time: $\Delta = 0.01$ in (39).

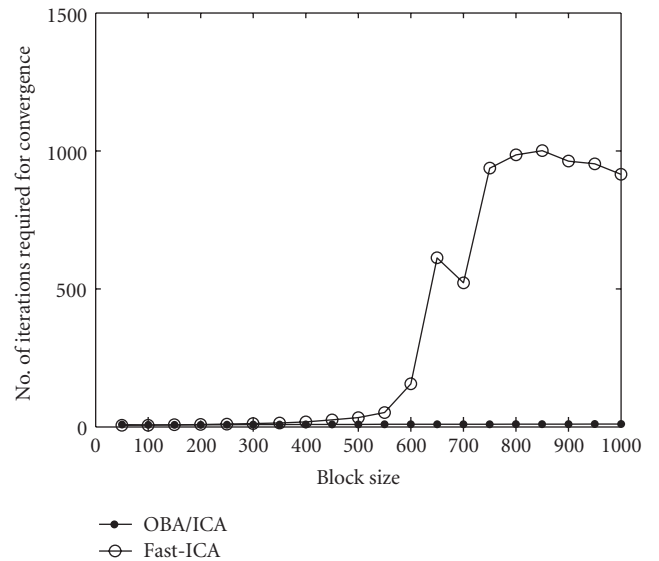


FIGURE 3: Convergence speed of fast-ICA and OBA/ICA ($k=0.5$) versus the processed block size when channel conditions vary linearly with time: $\Delta = 0.01$ in (39).

In our simulations, the block size is varied from 50 symbols to 1000 symbols, with a step size of 50. For each L , SIR and N_c are computed and averaged over 100 simulation runs.

Figures 2 and 3 show the performance and convergence speed of OBA/ICA and fast-ICA for relatively slow time-varying channel condition, that is, $\Delta = 0.01$. The additional scaling factor k in OBA/ICA (37) is 0.5. It is seen that the two algorithms have similar performance except for longer blocks, in which case OBA/ICA has better performance. This indicates OBA/ICA has better capability in dealing with time

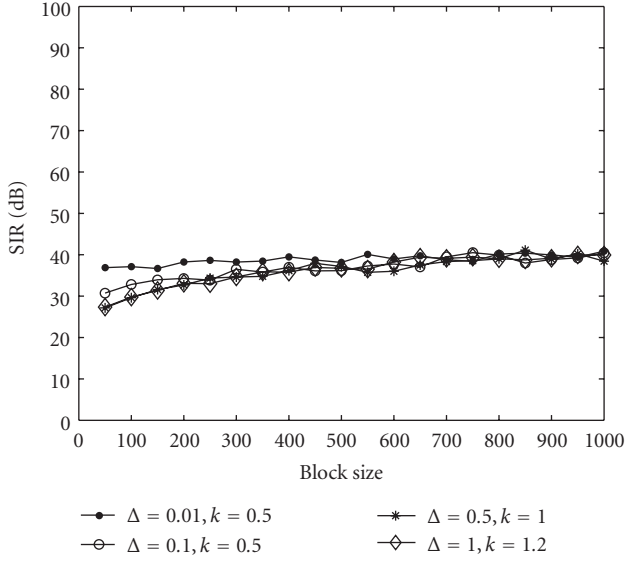


FIGURE 4: SIR achieved in dB versus the processed block size employing OBA/ICA when channel conditions vary linearly with time.

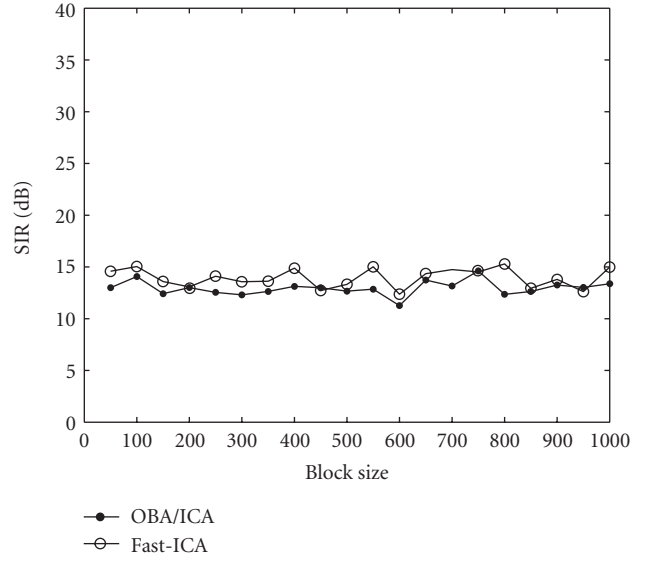


FIGURE 6: SIR achieved by OBA/ICA ($k = 0.5$) and fast-ICA when channel conditions change abruptly.

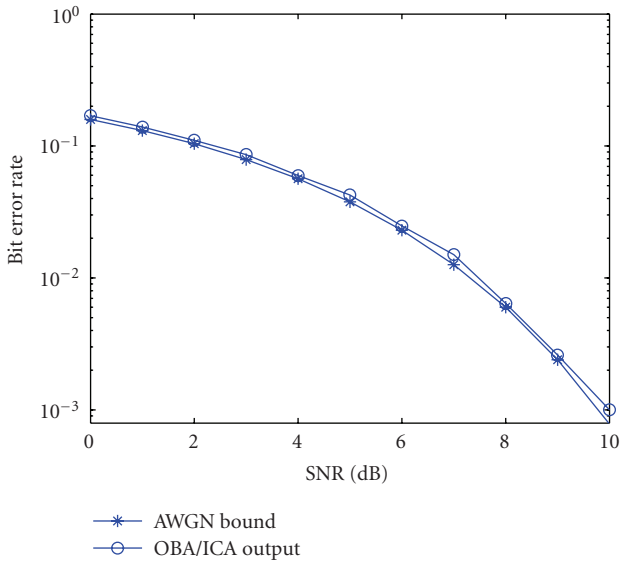


FIGURE 5: Bit error rate (BER) versus SNR employing OBA/ICA.

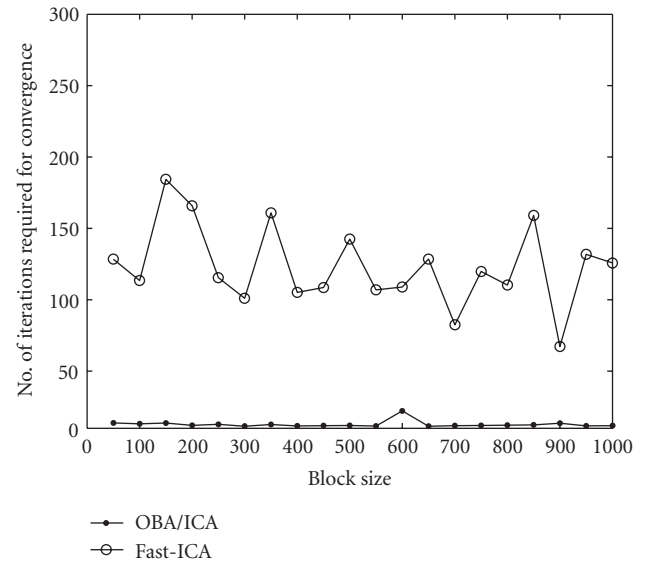


FIGURE 7: Convergence of OBA/ICA ($k = 0.5$) and fast-ICA when channel conditions change abruptly.

variation within one processing block. Also, fast-ICA converges very slowly for long blocks, while OBA/ICA always converges within 20 iterations regardless of the block size.

For faster time variation, that is, $\Delta = 0.1, 0.5, 1$, fast-ICA fails to converge within one thousand iterations, which makes it impractical to use. On the other hand, OBA/ICA always converges within 20 iterations. This is why only the OBA/ICA results are given. The performance for OBA/ICA is given in Figure 4. The optimal k values are given for every Δ . It is observed that a larger k should be used for faster time variation, as expected.

To study the performance of OBA/ICA under noisy conditions, simulations are performed with $\Delta = 0.01$ and thermal noise added. The resulting bit error rate (BER) is plotted versus signal-to-noise ratio (SNR) in Figure 5. As a reference, the BER with additive noise only, known as the AWGN (additive white Gaussian noise) bound, is also shown for comparison. It is clearly seen that OBA/ICA successfully achieves interference suppression in noisy conditions, and the obtained BER is close to the AWGN bound, which corresponds to the interference-free scenario. The convergence of OBA/ICA under noisy conditions requires about 7 to 16

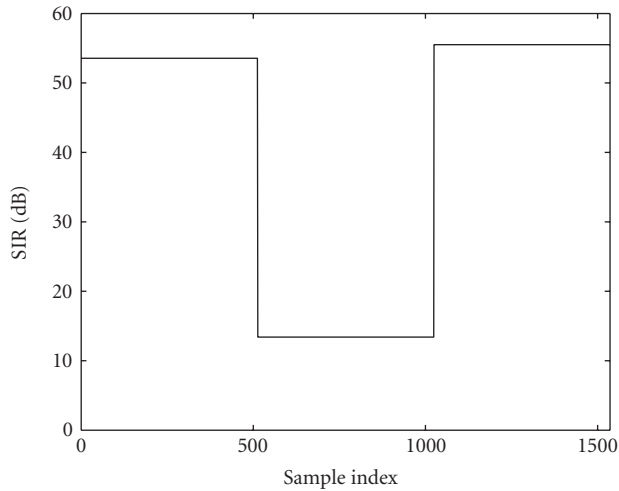


FIGURE 8: SIR achieved by OBA/ICA for three blocks when channel conditions change abruptly in time without finding the location of the sudden change (block size = 512).

iterations, compared to 7 to 10 iterations in the noiseless case. Therefore, a slight increase in the processing time may be required for OBA/ICA in the presence of thermal noise.

Next, fast-ICA and OBA/ICA are compared under abruptly changing channel conditions. To simulate this condition, an abrupt change of the mixing matrix is introduced within the processing block. Figures 6 and 7 compare fast-ICA and OBA/ICA in terms of average SIR and convergence speed without any knowledge about the abrupt change. As expected, the performance of both algorithms degrades when compared to the case of continuous time variation. However, OBA/ICA converges much faster than fast-ICA.

Following the detection of an abrupt change within a certain block, the binary search technique described in Section 5.4 is simulated to detect the location of the abrupt change. As before, one hundred simulation runs are performed and the average performance is given. The block size is chosen to be 512 samples. Figure 8 shows the performance of OBA/ICA for three consecutive blocks when a sudden channel change is simulated at the middle of the second block. Since the adaptive algorithm tries to converge to a compromising demixing matrix for two completely different mixing matrices, the performance for the second block degraded significantly. Figure 9 describes the performance of OBA/ICA after the application of binary search for the second block. As seen, the technique successfully identified the position of the abrupt change denoted by “a,” and the resulting performance for the second block is substantially improved compared to Figure 8.

In addition to these simulation results, in Figures 10 and 11 the residue interference power and the SIR value are shown as a function of the iteration index. Although the whole block is processed with a converged demixing matrix, the two figures illustrate the convergence process of OBA/ICA algorithm.

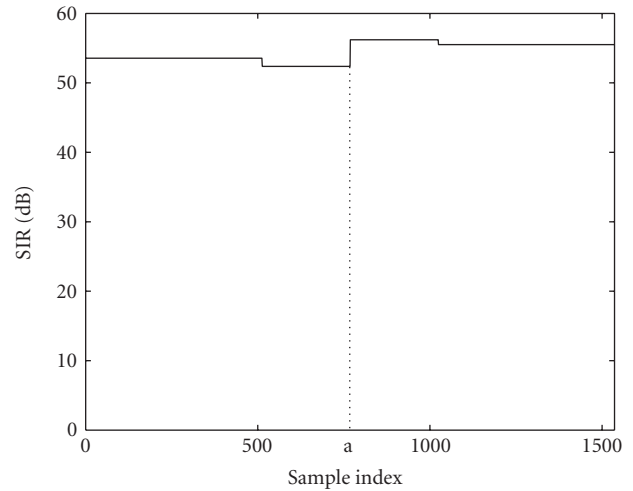


FIGURE 9: SIR achieved by OBA/ICA for three blocks when channel conditions change abruptly in time after finding the location of the sudden change (block size = 512).

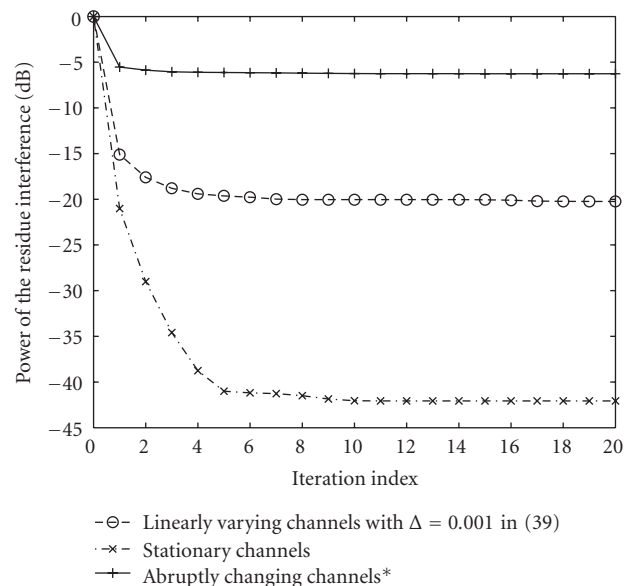


FIGURE 10: Residue interference power averaged over a hundred simulation runs versus iteration number for OBA/ICA assuming block size = 100. *Without finding the location of the abrupt change within the block.

7. CONCLUSIONS

In this paper, a gradient-based ICA algorithm with optimum block adaptation (OBA/ICA) is developed, which tailors the learning rate for each coefficient in the separation matrix and updates those rates at each block iteration. The computational complexity of OBA/ICA for each iteration is equivalent to the fast-ICA. When the channel is time varying, the

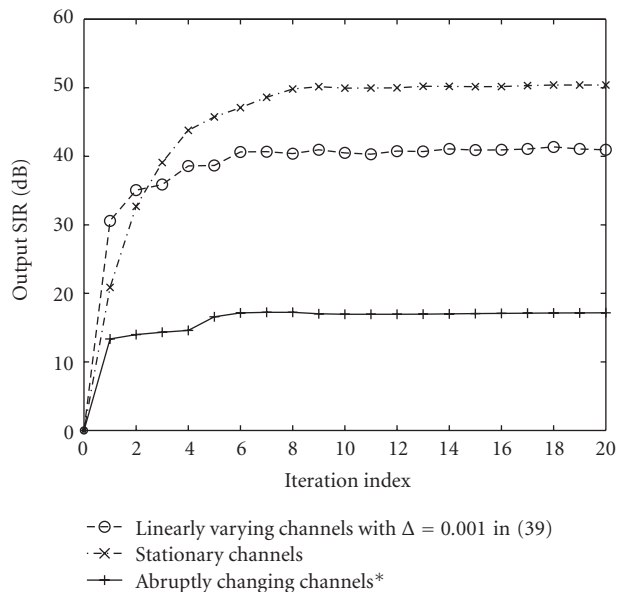


FIGURE 11: Output SIR averaged over a hundred simulation runs versus iteration number for OBA/ICA assuming block size = 100. *Without finding the location of the abrupt change within the block.

proposed technique is superior to the fast-ICA, especially in terms of convergence properties. This is true for changes that are linear or abrupt in nature.

ACKNOWLEDGMENT

The authors are grateful to Dr. Brent Myers, Conexant Systems, Inc., for financial and technical support to the research work reported in this paper.

REFERENCES

- [1] T.-W. Lee, M. S. Lewicki, and T. J. Sejnowski, "ICA mixture models for unsupervised classification of non-Gaussian classes and automatic context switching in blind signal separation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1078–1089, 2000.
- [2] T. Ristaniemi and J. Joutsensalo, "Advanced ICA-based receivers for block fading DS-CDMA channels," *Signal Processing*, vol. 82, no. 3, pp. 417–431, 2002.
- [3] L. Castedo, C. Escudero, and A. Dapena, "A blind signal separation method for multiuser communications," *IEEE Transactions on Signal Processing*, vol. 45, no. 5, pp. 1343–1348, 1997.
- [4] S. Malaroiu, K. Kiviluoto, and E. Oja, "Time series prediction with independent component analysis," in *Proceedings of International Conference on Advanced Investment Technology*, Gold Coast, Australia, January 2000.
- [5] M. McKeown, S. Makeig, S. Brown, et al., "Blind separation of functional magnetic resonance imaging (fMRI) data," *Human Brain Mapping*, vol. 6, no. 5-6, pp. 368–372, 1998.
- [6] C. L. Isbell and P. Viola, "Restructuring sparse high-dimensional data for effective retrieval," in *Advances in Neural Information Processing Systems*, vol. 11, MIT Press, Cambridge, Mass, USA, 1999.
- [7] A. Hyvärinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Computation*, vol. 9, no. 7, pp. 1483–1492, 1997.
- [8] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.
- [9] Z. Malouche and O. Macchi, "Adaptive unsupervised extraction of one component of a linear mixture with a single neuron," *IEEE Transactions on Neural Networks*, vol. 9, no. 1, pp. 123–138, 1998.
- [10] M. Gaeta and J.-L. Lacoume, "Source separation without prior knowledge: the maximum likelihood solution," in *Proceedings of the European Signal Processing Conference (EUSIPCO '90)*, pp. 621–624, Barcelona, Spain, September 1990.
- [11] D.-T. Pham, "Blind separation of instantaneous mixture of sources via an independent component analysis," *IEEE Transactions on Signal Processing*, vol. 44, no. 11, pp. 2768–2779, 1996.
- [12] P. Comon, "Independent component analysis, a new concept," *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.
- [13] J. Principe, D. Xu, and J. W. Fisher III, "Information-theoretic learning," in *Unsupervised Adaptive Filtering*, S. Haykin, Ed., vol. 1, pp. 265–319, John Wiley & Sons, New York, NY, USA, 2000.
- [14] P. Comon and B. Mourrain, "Decomposition of quantics in sums of powers of linear forms," *Signal Processing*, vol. 53, no. 2-3, pp. 93–107, 1996.
- [15] J.-F. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, no. 1, pp. 157–192, 1999.
- [16] A. Yeredor, "Blind source separation via the second characteristic function," *Signal Processing*, vol. 80, no. 5, pp. 897–902, 2000.
- [17] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, New York, NY, USA, 2001.
- [18] R. S. Sutton, "Adapting bias by gradient descent: an incremental version of delta-bar-delta," in *Proceedings of the 10th National Conference on Artificial Intelligence*, pp. 171–176, San Jose, Calif, USA, July 1992.
- [19] N. Murata, K.-R. Müller, A. Ziehe, and S. Amari, "Adaptive on-line learning in changing environments," in *Advances in Neural Information Processing Systems (NIPS '96)*, vol. 9, pp. 599–605, Denver, Colo, USA, December 1996.
- [20] N. Murata, M. Kawanabe, A. Ziehe, K. Müller, and S. Amari, "On-line learning in changing environments with applications in supervised and unsupervised learning," *Neural Networks*, vol. 15, no. 4-6, pp. 743–760, 2002.
- [21] G. B. Orr, *Dynamics and algorithms for stochastic search*, Ph.D. thesis, Department of Computer Science and Engineering, Oregon Graduate Institute, Beaverton, Ore, USA, 1995.
- [22] L. Bottou, "Online algorithms and stochastic approximations," in *Online Learning in Neural Networks*, D. Saad, Ed., pp. 9–42, Cambridge University Press, Cambridge, UK, 1998.
- [23] G. B. Orr and T. K. Leen, "Using curvature information for fast stochastic search," in *Advances in Neural Information Processing Systems*, M. Mozer, M. Jordan, and T. Petsche, Eds., vol. 9, MIT Press, Cambridge, Mass, USA, 1997.
- [24] T. Ristaniemi and J. Joutsensalo, "Advanced ICA-based receivers for DS-CDMA systems," in *Proceedings of IEEE International Conference on Personal, Indoor, and Mobile Radio, Communications*, London, UK, September 2000.

- [25] T. Yang and W. B. Mikhael, "A general approach for image and co-channel interference suppression in diversity wireless receivers employing ICA," *Journal of Circuits, Systems, and Signal Processing*, vol. 23, no. 4, pp. 317–327, 2004.
- [26] I. Kostanic and W. B. Mikhael, "Blind source separation technique for reduction of co-channel interference," *Electronics Letters*, vol. 38, no. 20, pp. 1210–1211, 2002.
- [27] W. Y. Leong and J. Holmer, "Implementing ICA in blind multiuser detection," in *IEEE International Symposium on Communications and Information Technologies*, vol. 2, pp. 947–952, Sapporo, Japan, October 2004.
- [28] W. B. Mikhael and F. Wu, "A fast block FIR adaptive digital filtering algorithm with individual adaptation of parameters," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 1, pp. 1–10, 1989.

Wasfy B. Mikhael received his B.S. degree (honors) in electronics and communications from Assiut University, Egypt, his M.S. in electrical engineering from the University of Calgary, Canada, and D.Eng. degree from Sir George Williams University, Montreal, Canada, in 1965, 1970, and 1973, respectively. He is a Professor in the School of Electrical Engineering and Computer Science, University of Central Florida (UCF),



Orlando. His research and teaching interests are in analog, digital, and adaptive signal processing for one and multidimensional signals and systems, with applications. His present work is in wireless communications, automatic target recognition, image and speech compression, classification and recognition of speakers and facial images. He has more than 250 refereed publications and holds several patents in the field. He has received many research, teaching, and professional service awards from industry and academia. He serves on editorial boards, has chaired several international, IEEE and other, conferences, has served as VP for the IEEE Circuits and Systems Society, and so forth. He has also served on several technical program committees, has organized state-of-the-art technical sessions, and is currently the Chair of the Midwest Symposium on Circuits and Systems steering committee membership.

Tianyu Yang received his B.S. degree in electrical engineering from Zhejiang University, Hangzhou, China, and his Ph.D. degree from the University of Central Florida, Orlando, Florida, USA, in 2001 and 2004, respectively. He is an Assistant Professor in the Department of Electrical and Systems Engineering, Embry-Riddle Aeronautical University, Daytona Beach, Florida. His research interests include adaptive/statistical signal processing, wireless transceiver design, and image/speaker recognition. He has more than 20 publications in refereed journals and conferences, and teaches various courses in electrical engineering and engineering sciences. He is a Member of IEEE, IEE, Eta Kappa Nu, and Phi Kappa Phi.

