4th WSEAS International Conference on ELECTRONICS, CONTROL and SIGNAL PROCESSING, Miami, Florida, USA, 17-19 November, 2005 (pp.53-57)

A Novel Conjugate Gradient Based Block Adaptive ICA Algorithm for Dynamic Environments

THOMAS T. YANG
Electrical and Systems Engineering Department
Embry-Riddle Aeronautical University
600 S. Clyde Morris Blvd.
Daytona Beach, Florida 32114
U.S.A.

WASFY B. MIKHAEL
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, Florida 32816
U.S.A.

Abstract: This paper presents a novel block adaptive Independent Component Analysis (ICA) algorithm based on the conjugate gradient method. The algorithm is suitable for application in dynamic environments. Simulation results employing the proposed method for interference rejection in wireless receivers are given. It is shown that the new algorithm outperforms the Fast-ICA algorithm under dynamic conditions. Consequently, the proposed algorithm lends itself to practical applications in mobile cellular systems for higher user mobility, handoff, and rapid changing channel conditions.

Key Words: Adaptive DSP, Conjugate Gradient, Dynamic Environments, ICA, BC-ICA

1 Introduction

Independent Component Analysis (ICA) is a statistical technique that extracts statistically independent signals from their linear combinations. Since the technique utilizes only higher-order statistics, it is very attractive in areas where little prior information is available about the mixing process. Under stationary or slow time-varying conditions, the fixed-point Fast-ICA is a highly efficient block algorithm. However, due to its inherent fixed-point assumption, the algorithm lacks the ability to operate in dynamic environments [1].

Previously, conjugate-gradient techniques for adaptive filters have been proposed to use search directions other than the negative gradient direction [2-3]. It has been shown that these techniques possess better convergence properties. Also, the technique is also used for ICA applications [4-5]. In this contribution, a Block Conjugate-gradient ICA (BC-ICA) algorithm is proposed. Computer simulations confirm the effectiveness of the new algorithm in dynamic environments.

2 Formulation

The basic ICA model is given by:

$$\mathbf{X}=\mathbf{AS} \tag{1}$$

Here, $\boldsymbol{X}$ is the observation matrix, $\mathbf{A}$ is the unknown 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 $\mathbf{W}$. Ideally, $\mathbf{W}$ is the inverse of $\mathbf{A}$. This is achieved by making each component in $\mathbf{WX}$ as independent as possible.

The Fast-ICA algorithm is a block algorithm based on the fixed-point assumption. The "expectation" operator in the definition of statistical independence is estimated by the average over L data points, where L is the block size [6]. The performance is better when L is larger. However, it is very important that the mixing matrix stays approximately constant within one processing block, i.e., quasi-stationary. Thus, the problem arises when the mixing matrix is time varying, in which case a large L violates the assumption of quasi-stationarity.

Here we formulate a new adaptive block ICA algorithm based on the conjugate gradient method that outperforms Fast-ICA. Similar to the Fast-ICA, the weight update equation derived here updates each row, one at a time, of the separation matrix. The absolute value of kurtosis is used as the measure of nongaussianity, which is to be maximized. Other ICA related operations such as preprocessing and orthogonalization are identical to Fast-ICA.

To proceed, the following parameters are defined:

j: iteration index.

N: number of observations.

L: length of the processing block.

$\underline{w(j)} = [w_1(j)\ w_2(j)\ \ldots\ldots\ w_N(j)]^T$: the current row of the separation matrix for the jth iteration.

$x_{l,i}(j)$: the ith signal in the lth observation data vector for the jth iteration. ($l = 1, 2, \ldots, L$; $i = 1, 2, \ldots, N$)

$\underline{X_l(j)} = [x_{l,1}(j)\ x_{l,2}(j)\ \ldots\ldots\ x_{l,N}(j)]^T$: lth signal observation for the jth iteration.

$[G]_j = \left[\underline{X_1(j)}\quad \underline{X_2(j)}\quad \ldots\ldots\quad \underline{X_L(j)}\right]^T$: Observation matrix for the jth iteration.

The lth kurtosis value for the jth iteration is

$$kurt_l(j) = E\{[\underline{w^T(j)}\,\underline{X_l(j)}]^4\} - 3 \qquad (2)$$

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

Then, the kurtosis vector for the jth iteration is

$$\underline{kurt(j)} = [kurt_1(j)\ kurt_2(j)\ldots\ldots\ kurt_L(j)]^T \qquad (3)$$

First, the lth kurtosis value in the $(j+1)$th iteration is given by the Taylor series expansion.

$$kurt_l(j+1) = kurt_l(j) + \sum_{i=1}^{N} \frac{\partial kurt_l(j)}{\partial w_i(j)} \Delta w_i(j) + \ldots \quad l = 1, 2, \ldots, L \qquad (4)$$

where

$$\Delta w_i(j) = w_i(j+1) - w_i(j) \quad i = 1, 2, \ldots, N \qquad (5)$$

In (4), if $\Delta w_i(j)$ is constrained to be small enough, higher order derivative terms can be omitted.

We proceed by dropping the expectation operator in (2). Thus,

$$\frac{\partial kurt_l(j)}{\partial w_i(j)} \cong 4x_{l,i}(j)[\underline{w^T(j)}\,\underline{X_l(j)}]^3 \qquad (6)$$

Then, (4) becomes

$$kurt_l(j+1) = kurt_l(j) + 4[\underline{w^T(j)}\,\underline{X_l(j)}]^3[\underline{X_l^T(j)}\,\underline{\Delta w(j)}] \qquad (7)$$

Writing (7) for every l, the Taylor series expansion becomes

$$\underline{kurt(j+1)} = \underline{kurt(j)} + 4[C]_j^3[G]_j\,\underline{\Delta w(j)} \qquad (8)$$

where

4th WSEAS International Conference on ELECTRONICS, CONTROL and SIGNAL PROCESSING, Miami, Florida, USA, 17-19 November, 2005 (pp.53-57)

$$[C]_j = \begin{bmatrix} \underline{w}^T(j)\underline{X}_1(j) & \cdots\cdots & 0 \\ \cdots\cdots & \cdots\cdots & \cdots\cdots \\ 0 & \cdots\cdots & \underline{w}^T(j)\underline{X}_L(j) \end{bmatrix} \quad (9)$$

is a diagonal matrix.

Suppose $\underline{w}(j)$ is updated along direction $\underline{r}(j)$, i.e.,

$$\Delta \underline{w}(j) = \alpha \underline{r}(j) \quad (10)$$

we need to calculate the optimum value for α given $\underline{r}(j)$.

From (10), the Taylor series expansion (8) becomes

$$\underline{kurt}(j+1) = \underline{kurt}(j) + 4\alpha [C]_j^3 [G]_j \underline{r}(j) \quad (11)$$

Our purpose is to maximize the total squared kurtosis in the *(j+1)*th iteration. Define

$$\underline{q}(j) = [G]_j^T [C]_j^3 \underline{kurt}(j) = [q_1(j) \ldots q_N(j)]^T \quad (12)$$

$$[R]_j = [G]_j^T [C]_j^6 [G]_j = [R_{mn}(j)] \qquad 1 \le m, n \le N \quad (13)$$

The total squared kurtosis can be expressed as:

$$\underline{kurt}^T(j+1)\underline{kurt}(j+1) = \underline{kurt}^T(j)\underline{kurt}(j) + 8\alpha \underline{r}^T(j)\underline{q}(j) + 16\alpha^2 \underline{r}^T(j)[R]_j \underline{r}(j) \quad (14)$$

Taking the derivative of the total squared kurtosis (14) with respect to α, and setting the resulting expression to zero, one obtains the optimum value of α given by

$$\alpha = -0.25 \frac{\underline{r}^T(j)\underline{q}(j)}{\underline{r}^T(j)[R]_j \underline{r}(j)} \quad (15)$$

The remaining task is to identify the suitable weight update direction according to the conjugate-gradient method. This technique can be considered as intermediate lying between the steepest descent and Newton's methods, in terms of complexity and convergence properties [2-3].

In the first iteration, the conjugate-gradient method uses the gradient direction given by

$$\nabla_B(j) = \frac{\partial\{\underline{kurt}^T(j)\underline{kurt}(j)\}}{\partial \underline{w}(j)} = \frac{1}{L}\left[\frac{\partial\{\underline{kurt}^T(j)\underline{kurt}(j)\}}{\partial w_1(j)} \cdots \frac{\partial\{\underline{kurt}^T(j)\underline{kurt}(j)\}}{\partial w_M(j)}\right]^T$$

$$= \frac{8}{L}[G]_j^T [C]_j^3 \underline{kurt}(j) \quad (16)$$

In the derivation of (16), the expectation operator is dropped. Now, the BC-ICA algorithm can be described as follows:

(1). Initialize $\underline{w}(0)$, *j*=0.

(2). According to (16), find the direction

$$\underline{g}(j) = \nabla_B(j) = \frac{8}{L}[G]_j^T [C]_j^3 \underline{kurt}(j)$$

a. If $\underline{w}(j)$ has converged, then terminated the algorithm and return $\underline{w}(j)$;

b. If $\underline{w}(j)$ has not converged and *j* = 0, then $\underline{r}(j) = \underline{g}(j)$, and proceed to **step (3)**;

c. If $\underline{w}(j)$ has not converged and *j* > 0, then compute $\underline{r}(j)$ from

$$\underline{r}(j) = \underline{g}(j) + \frac{\underline{g}^T(j)\underline{g}(j)}{\underline{g}^T(j-1)\underline{g}(j-1)} \underline{r}(j-1) \quad (17)$$

(3). Update the weight vector according to $\underline{w}(j+1) = \underline{w}(j) + \alpha \underline{r}(j)$, where α is given by (15).

(4). *j* = *j*+1, go to **step (2).**

3 Simulation Results

Computer simulations are performed for a wireless communication application to evaluate the new technique. Assume a dual-antenna BPSK receiver is receiving two signals simultaneously, namely, the desired signal *s(t)* and an interferer *i(t)* [7]. The signal received by each antenna is a linear combination of *s(t)* and *i(t)*. It is desired to separate the desired signal from the interference. In this scenario, the mixing matrix is determined primarily by the wireless channel's fading coefficients.

The performance measures are the Signal to Interference Ratio (*SIR*) and the number of iterations to convergence. *SIR* represents the average ratio of the desired signal power to the power of the estimation error, defined as:

$$SIR = 10 \log_{10} \left(\frac{1}{L} \sum_{k=1}^{L} \frac{s(k)^2}{[s(k) - y(k)]^2} \right) \quad (18)$$

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

In our simulations two types of time variations are studied. In the first case, the change of the channel is modeled as a continuous linear time variation in the mixing matrix's coefficients, and the ICA algorithm seeks a compromise separation matrix. 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.

For continuous linear time variation, the mixing matrix is modeled as:

$$A = \begin{bmatrix} 1 + l\Delta & 0.5 \\ 0.7 & 2 + l\Delta \end{bmatrix} \quad (19)$$

Where l = 1, 2, … , 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/Δ.

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

First, BC-ICA and Fast-ICA are simulated for linearly time-varying channels, and the parameter Δ is set to 0.01 and 0.03. The achieved SIR and convergence speed in terms of the number of iterations are plotted in **Figs. 1** and **2**. It is found that BC-ICA achieves better SIR than Fast-ICA, and BC-ICA converges much faster.

Next, Fast-ICA and BC-ICA are compared under abruptly changing channel conditions. An abrupt change of the mixing matrix is introduced in the middle of the processing block. As expected, the performance of both algorithms degrades. However, BC-ICA converges faster than the Fast-ICA, as shown in **Fig. 3**.

4 Conclusion

In this paper, a novel block ICA algorithm, BC-ICA, is developed based on the conjugate gradient method. In dynamic environments, it achieves faster convergence and better performance than the Fast-ICA algorithm without increasing the computational complexity. Computer simulations confirm the effectiveness of BC-ICA. Also, it is shown that BC-ICA outperforms the Fast-ICA in both gradual and abrupt time-varying situations.

References:

[1]. A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis", *IEEE Transactions on Neural Networks*, Vol. 10, No. 3, May 1999.

[2]. G. Boray, M. Srinath, "Conjugate gradient techniques for adaptive filters", *IEEE Transactions on Circuits and Systems*, Vol. CAS-39, part I, pp. 1-10, Jan. 1992

[3]. M. Barsoum and W. Mikhael, "A Conjugate-Gradient-Based Block Adaptive Algorithm, its Applications and Performance Evaluation," *The 37th IEEE Midwest Symposium on Circuits and Systems*, Lafayette, LA, 939-942, August 1994.

[4]. A. Cichocki, B. Orster, A. Back, S. Amari, "Online adaptive algorithms in non-stationary environments using a modified conjugate gradient approach", *Proceedings of the 1997 IEEE Workshop on Neural Networks for Signal Processing VII,* pp. 316-325, 1997.

[5]. R. Martin-Clemente, C. Puntonet, J. Acha , "A Conjugate Gradient Method and Simulated Annealing for Blind Separation of Sources", *Proceedings of the 6th International Work-Conference on Artificial and Natural Neural Networks: Bio-inspired Applications of Connectionism-Part II*, pp.810-817, 2001.

[6]. A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis", *IEEE Transactions on Neural Networks*, Vol. 10, No. 3, May 1999.

[7]. T. Yang, W. 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, Sept. 2004.

Fig. 1 Output SIR employing BC-ICA and Fast-ICA for linearly time-varying channels

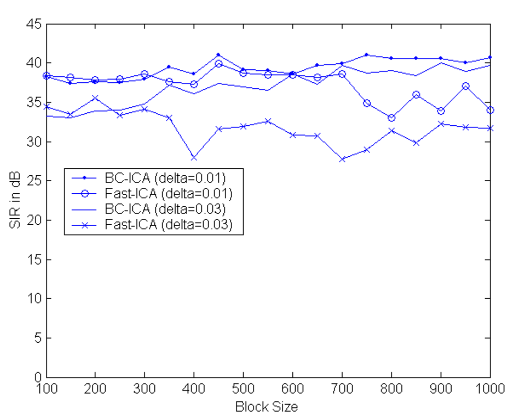

Fig. 2 Convergence of the BC-ICA and Fast-ICA algorithms under linearly time-varying channel conditions

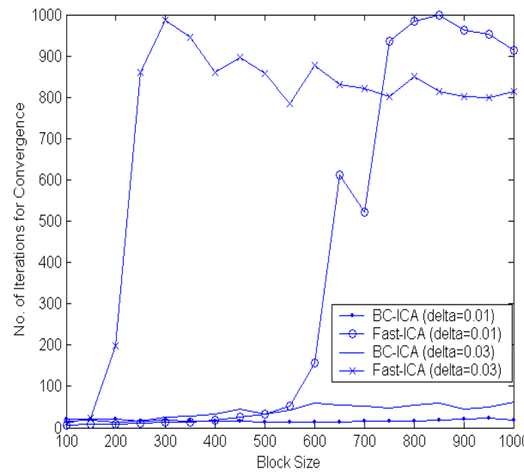

Fig. 3 Convergence of the BC-ICA and Fast-ICA algorithms under abruptly changing channel conditions