Research Article

Time-Domain Convolutive Blind Source Separation Employing Selective-Tap Adaptive Algorithms

Qiongfeng Pan and Tyseer Aboulnasr

School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada K1N 6N5

We investigate novel algorithms to improve the convergence and reduce the complexity of time-domain convolutive blind source separation (BSS) algorithms. First, we propose MMax partial update time-domain convolutive BSS (MMax BSS) algorithm. We demonstrate that the partial update scheme applied in the MMax LMS algorithm for single channel can be extended to multichannel time-domain convolutive BSS with little deterioration in performance and possible computational complexity saving. Next, we propose an exclusive maximum selective-tap time-domain convolutive BSS algorithm (XM BSS) that reduces the interchannel coherence of the tap-input vectors and improves the conditioning of the autocorrelation matrix resulting in improved convergence rate and reduced misalignment. Moreover, the computational complexity is reduced since only half of the tap inputs are selected for updating. Simulation results have shown a significant improvement in convergence rate compared to existing techniques.

1. INTRODUCTION

Blind source separation (BSS) [1, 2] is an established area of work estimating source signals based on information about observed mixed signals at the sensors, that is, the estimation is performed without exploiting information about either the source signals or the mixing system. Independent component analysis (ICA) [3] is the main statistical tool for dealing with the BSS problem with the assumption that the source signals are mutually independent. In the instantaneous BSS case, signals are mixed instantaneously and ICA algorithms can be directly employed to separate the mixtures. However, in a realistic environment, signals are always mixed in convolutive manner because of propagation delay and reverberation effects. Therefore, much research deals with convolutive blind source separation based on extending instantaneous blind source separation or independent component analysis to convolutive case.

The straightforward choice in time-domain convolutive blind source separation is based on directly extending instantaneous BSS to the convolutive case [4, 5]. This natural approach achieves good separation results once the algorithm converges. However, time-domain convolutive blind source separation suffers from high computational complexity and low convergence rate, especially for systems requiring long FIR filters for the separation.

Frequency domain convolutive BSS [6, 7] was proposed to deal with the expensive computational complexity problem of time-domain BSS. In frequency domain BSS, complex-valued ICA for instantaneous BSS is employed in every frequency bin independently. The advantage of this approach is that any existing complex-valued instantaneous BSS algorithm can be used and the computational complexity is reduced by exploiting the FFT for the computation of convolution which is the basis of popularity of frequency domain approaches. However, the permutation and scaling ambiguity in the ICA algorithm, which is not a problem for instantaneous BSS, becomes a serious problem in frequency domain convolutive BSS. Since frequency domain convolutive BSS is performed by instantaneous BSS at each frequency bin separately, the order and the scale of the unmixed signals are random because of the inherent ambiguity of ICA algorithms. When we transform the separated signals back from frequency domain to time domain, the components at a given frequency bin may not come from the same source signal and may not have a consistent scale factor. Thus, we need to align these components and adjust the scale in each frequency bin so that a separated signal in time domain is obtained from frequency components of the same source signal and with consistent amplitude. This is well known as the

permutation and scaling problem of frequency domain convolutive BSS [8, 9]. These built-in problems in frequency domain approaches make it worthwhile to reconsider ways of reducing the complexity of time-domain approaches and improving their convergence rates.

In recent years, several partial update adaptive algorithms were proposed to model single-channel systems with reduced overall system complexity by updating only a subset of coefficients. Within these partial update algorithms, the MMax NLMS in [10] was reported to have the closest performance to the full update case for any given number of coefficients to be updated. In [11], the MMax selective-tap strategy was extended to the two-channel case to exclusively select coefficients corresponding to the maximum inputs as a means to reduce interchannel coherence in stereophonic acoustic echo cancellation rather than as a way to reduce complexity. Simulation results for this exclusive maximum adaptive algorithm show that it can significantly improve the convergence rate compared with existing stereophonic echo cancellation techniques.

In this paper, we propose using these reduced complexity approaches in time-domain BSS to address complexity and low convergence problems. First, we propose MMax natural gradient-based partial update time-domain convolutive BSS algorithm (MMax BSS). In this algorithm, only a subset of coefficients in the separation system gets updated at every iteration. We demonstrate that the partial update scheme applied in the MMax LMS algorithm for a single channel can be extended to the multichannel time-domain convolutive BSS with little deterioration in performance and possible computational complexity saving. By employing selective-tap strategies used for stereophonic acoustic echo cancellation [11], we propose exclusive maximum selective-tap time-domain convolutive BSS algorithm (XM BSS). The exclusive tap-selection update procedure reduces the interchannel coherence of the tap-input vectors and improves the conditioning of the autocorrelation matrix so as to accelerate convergence rate and reduce the misalignment. The computational complexity is reduced as well since only half of the tap inputs are selected for updating (note that some overhead is needed to select the set to be updated). Simulation results have shown a significant improvement in convergence rate compared with existing techniques. As far as we know, the application of partial update and selective-tap update schemes to time-domain BSS algorithm is in itself novel.

BSS algorithms are generally preceded by a prewhitening stage that aims to reduce the correlation between the different input sources (as opposed to regular whitening where correlation between different samples of the same source is reduced). This decorrelation step leads to a subsequent separation matrix that is orthogonal and less ill-conditioned. The proposed partial update BSS algorithm incorporates this whitening concept into the separation process by adaptively reducing the interchannel coherence of the tap-input vectors.

The rest of this paper is organized as follows. In Section 2, we review blind source separation and its challenges in time domain and frequency domain. In Section 3, we review the single-channel MMax partial update adaptive algorithm for
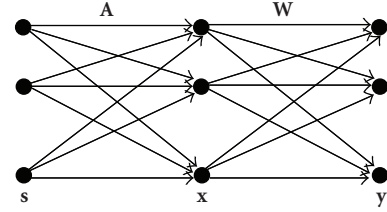

FIGURE 1: Structure of instantaneous blind source separation system.

linear filters. In Section 4, we review exclusive maximum selective-tap adaptive algorithm for stereophonic echo cancellation. We propose the MMax partial update time-domain convolutive BSS algorithm in Section 5 and the exclusive maximum update time-domain convolutive BSS algorithm in Section 6. The tools for assessing the quality of the separation are presented in Section 7 and simulation results for the proposed algorithms for generated gamma signals and speech signals are presented in Section 8. In Section 9, we draw our conclusions from our work.

2. BLIND SOURCE SEPARATION

2.1. Instantaneous time-domain BSS

Blind source separation (BSS) is a very versatile tool for signal separation in a number of applications utilizing observed mixtures and the independence assumption. For instantaneous mixtures, independent component analysis (ICA) can be employed directly to separate the mixed signals.

The ICA-based algorithm for instantaneous blind source separation requires the output signals to be as independent as possible. Different algorithms can be obtained based on how this independence is measured. The instantaneous time-domain BSS structure is shown in Figure 1. In this paper, we use the Kullback-Leibler divergence to measure independence and obtain the BSS algorithm as follows:

$$\mathbf{x} = \mathbf{As},$$
$$\mathbf{y} = \mathbf{Wx}, \tag{1}$$

where $\mathbf{s} = [s_1, \ldots, s_N]^T$ is the vector of source signals, $\mathbf{x} = [x_1, \ldots, x_M]^T$ is the vector of mixture signals, $\mathbf{y} = [y_1, \ldots, y_N]^T$ is the vector of separated signals, $\mathbf{A}$ and $\mathbf{W}$ are instantaneous mixing and unmixing systems and can be described as

$$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \cdot & \cdot & \cdot \\ a_{M1} & \cdots & a_{MN} \end{bmatrix}, \qquad \mathbf{W} = \begin{bmatrix} w_{11} & \cdots & w_{1M} \\ \cdot & \cdot & \cdot \\ w_{N1} & \cdots & w_{NM} \end{bmatrix}. \tag{2}$$

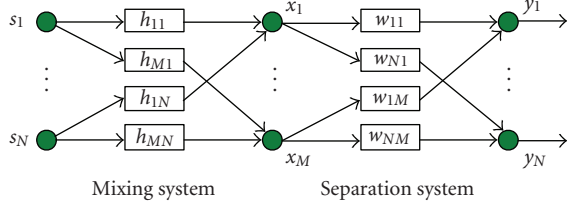The Kullback-Leibler divergence of the output signal vector

FIGURE 2: Structure of convolutive blind source separation system.

is

$$D(p(\mathbf{y}) \parallel q(\mathbf{y})) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{i=1}^{N} p_i(y_i)} d\mathbf{y}, \qquad (3)$$

where $p(\mathbf{y})$ is the probability density of output signals, $p_i(y_i)$ is the probability density of output signal y_i, $q(\mathbf{y})$ is the joint probability density of output signals:

$$
\begin{aligned}
&D(p(\mathbf{y}) \parallel q(\mathbf{y})) \\
&= \int p(\mathbf{y}) \log p(\mathbf{y}) - \sum_{i=1}^{N} \left(\int p(\mathbf{y}) \log p_i(y_i) \right) \\
&= -H(\mathbf{y}) + \sum_{i=1}^{N} H_i(y_i) \\
&= -H(\mathbf{x}) - \log |\det(\mathbf{W})| - \sum_{i=1}^{N} E[\log(p_i(y_i))],
\end{aligned}
$$
$$(4)$$

where $H(\cdot)$ is the entropy operation.

Using standard gradient

$$
\begin{aligned}
\Delta D = \frac{\partial D}{\partial \mathbf{W}} &= -\frac{\partial}{\partial \mathbf{W}} H(\mathbf{x}) - \frac{\partial}{\partial \mathbf{W}} \log(|\det(\mathbf{W})|) \\
&\quad - \frac{\partial}{\partial \mathbf{W}} \sum_{i=1}^{N} E[\log(p_i(y_i))] \\
&= 0 - \mathbf{W}^{-T} + E[\varphi(\mathbf{y})\mathbf{x}^T],
\end{aligned}
$$
$$(5)$$

where $\varphi(y) = [\partial p_1(y_1)/\partial y_1/p_1(y_1), \ldots, \partial p_N(y_N)/\partial y_N/p_N(y_N)]$ is a nonlinear function related to the probability density function of source signals, the coefficients $\mathbf{W}$ in the unmixing system are then updated as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \Delta \mathbf{W},$$
$$\Delta \mathbf{W}_{\text{standard\_grad}} = -\mu \frac{\partial D}{\partial \mathbf{W}} = \mu(\mathbf{W}^{-T} - E[\varphi(\mathbf{y})\mathbf{x}^T]). \qquad (6)$$

However, BSS algorithms have traditionally used the natural gradient [4] which is acknowledged as having better performance. In this case, $\Delta \mathbf{W}$ is given by

$$\Delta \mathbf{W}_{\text{natural\_grad}} = -\mu \frac{\partial D}{\partial \mathbf{W}} \mathbf{W}^T \mathbf{W} = \mu[\mathbf{I} - E(\varphi(\mathbf{y})\mathbf{y}^T)]\mathbf{W}.$$
$$(7)$$

2.2. Convolutive BSS algorithm

The convolutive BSS model is illustrated in Figure 2. N source signals $\{s_i(k)\}$, $1 \leq i \leq N$, pass through an unknown N-input, M-output linear time-invariant mixing system to yield the M mixed signals $\{x_j(k)\}$. All source signals $s_i(k)$ are assumed to be statistically independent.

Defining the vectors $\mathbf{s}(k) = [s_1(k) \cdots s_N(k)]^T$ and $\mathbf{x}(k) = [x_1(k) \cdots x_M(k)]^T$, the mixing system can be represented as

$$
\begin{bmatrix} x_1(k) \\ \cdot \\ x_M(k) \end{bmatrix} = \begin{bmatrix} h_{11}(l) & \cdots & h_{1N}(l) \\ \cdot & \cdot & \cdot \\ h_{M1}(l) & \cdots & h_{MN}(l) \end{bmatrix} * \begin{bmatrix} s_1(k) \\ \cdot \\ s_N(k) \end{bmatrix}, \qquad (8)
$$

where $*$ is convolution operation.

The jth sensor signal can be obtained by

$$x_j(k) = \sum_{i=1}^{N} \sum_{l=0}^{L-1} h_{ji}(l) s_i(k-l), \qquad (9)$$

where $h_{ji}(l)$ is the impulse response from source i to sensor j, L defines the order of the FIR filters used to model this impulse response.

The task of the convolutive BSS algorithm is to obtain an unmixing system such that the outputs of this system $\mathbf{y}(k) = [y_1(k) \cdots y_N(k)]^T$ become mutually independent as the estimates of the N source signals. The separation system typically consists of a set of FIR filters $w_{ij}(k)$ of length Q each. The unmixing system can also be represented as

$$
\begin{bmatrix} y_1(k) \\ \cdot \\ y_N(k) \end{bmatrix} = \begin{bmatrix} w_{11}(l) & \cdots & w_{1M}(l) \\ \cdot & \cdot & \cdot \\ w_{N1}(l) & \cdots & w_{NM}(l) \end{bmatrix} * \begin{bmatrix} x_1(k) \\ \cdot \\ x_M(k) \end{bmatrix}. \qquad (10)
$$

The ith output of the unmixing system is given as

$$y_i(k) = \sum_{j=1}^{M} \sum_{l=0}^{Q-1} w_{ij}(l) x_j(k-l). \qquad (11)$$

By extending the instantaneous BSS algorithm to the convolutive case, we get the time-domain convolutive BSS algorithm as

$$\Delta \underline{\mathbf{W}} = -\mu \frac{\partial D}{\partial \underline{\mathbf{W}}} \underline{\mathbf{W}}^T \underline{\mathbf{W}} = \mu[\mathbf{I} - E(\varphi(\mathbf{y})\mathbf{y}^T)]\underline{\mathbf{W}}, \qquad (12)$$

where $\underline{\mathbf{W}}$ the unmixing matrix with FIR filters as its components.

This approach is the natural extension and achieves good separation results once the algorithm converges. However, time-domain convolutive blind source separation suffers from high computational complexity and low convergence rate, especially for systems with long FIR filters.

Convolutive BSS can also be performed in frequency domain by using short-time Fourier transform. This method is very popular for convolutive mixtures and is based on transforming the convolutive blind source separation problem into instantaneous BSS problem at every frequency bin.
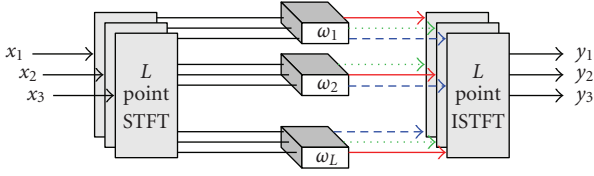
FIGURE 3: Illustration of frequency domain convolutive BSS with frequency permutation.

The advantage of frequency domain convolutive BSS lies in three factors. First the computational complexity is reduced since the convolution operations are transferred into multiplication operations by short-time FFT. Second, the separation process can be performed in parallel at all frequency bins. Finally any complex-valued instantaneous ICA algorithm can be employed to deal with the separation at each frequency bin. However, the permutation and scaling ambiguity in ICA algorithm, which is not a problem for instantaneous BSS, becomes a serious problem in frequency domain convolutive BSS.

This problem can be illustrated by Figure 3. Frequency domain convolutive BSS is performed by instantaneous BSS at each frequency bin separately. As a result, the order and the scale of the unmixed signals are random because of the inherent indeterminacy of ICA algorithms. When we transform the separated signals back from frequency domain to time domain, the components at different frequency bins may not come from the same source signal and may not have consistent scale. Thus, we need to align the permutation and adjust the scale in each frequency bin so that a separated signal in time domain is obtained from frequency components of the same source signal and with consistent amplitude. This is not a simple problem.

3. PARTIAL UPDATE ADAPTIVE ALGORITHM

The basic idea of partial update adaptive filtering is to allow for the use of filters with a number of coefficients L large enough to model the unknown system while reducing the overall complexity by updating only M coefficients at a time. This results in considerable savings for $M \ll L$. Invariably, there are penalties for this partial update, the most obvious of which is reduced convergence rate. The question then becomes which coefficients should we update and how do we minimize the impact of the partial update on the overall filter performance. In this section, we review the MMax partial update adaptive algorithm for linear filters [10] since it forms the basis of our proposed MMax time-domain convolutive BSS algorithm.

Consider a standard adaptive filter set-up where $x(n)$ is the input, $y(n)$ is the output, and $d(n)$ is the desired output, all at instant n. The output error $e(n)$ is given by

$$e(n) = d(n) - y(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n), \quad (13)$$

where $\mathbf{w}(n)$ is the $L \times 1$ column vector of the filter coefficients and $\mathbf{x}(n)$ is the $L \times 1$ column vector $\mathbf{x}(n) =$

$[x(n), \dots, x(n-i), \dots, x(n-L+1)]$ of the current and past inputs to the filter, both at instant n. The ith element of $\mathbf{w}(n)$ is $w_i(n)$ and it multiplies the ith delayed input $\mathbf{x}(n)$, $i = 0, \dots, L-1$.

The basic NLMS algorithm is known for its extreme simplicity provided for coefficient update as given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\frac{\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}, \quad (14)$$

where μ is the step size determining the speed of convergence and the steady state error.

In the single-channel MMax NLMS algorithm [10], for an adaptive filter of length L, the set of M coefficients to be updated is selected as the one that provides the maximum reduction in error. It is shown in [10] that this criterion reduces to the set of coefficients multiplying inputs $x(n-i)$ with the largest magnitude using the standard NLMS update equation. This selective-tap updating can be expressed as

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\mathbf{Q}(n)e(n)\frac{\mathbf{x}(n)}{\|\mathbf{x}(n)\|^2}, \quad (15)$$

where $\mathbf{Q}(n)$ is the tap-selection matrix as

$$\mathbf{Q}(n) = \text{diag}\{\mathbf{q}(n)\},$$

$$q_i(n) = \begin{cases} 1, & |x(n-i-1)| \in \{M \text{ maxima of } |\mathbf{x}(n)|\} \\ 0, & \text{otherwise.} \end{cases}$$

$$(16)$$

An analysis of the mean square error convergence is provided in [10] based on matrix formulation of data-dependent partial updates. Based on the analysis, it was shown that the MMax algorithm provides the closest performance to the full update case for any given number of coefficients to be updated. This was also confirmed in [12].

4. EXCLUSIVE MAXIMUM SELECTIVE-TAP ADAPTIVE ALGORITHM

Recently, an exclusive maximum (XM) partial update algorithm was proposed in [11] to deal with stereophonic echo cancellation. The XM algorithm was motivated by MMax partial update scheme [10] as both select a subset of coefficients for updating in every adaptative iteration. However, in the XM partial update, the goal is not to reduce computational complexity. Rather the exclusive maximum tap-selection strategy was proposed to reduce interchannel coherence in a two-channel stereo system and improve the conditioning of the input vector autocorrelation matrix. We now review the algorithm in [11] here since it forms the basis of our proposed XM time-domain convolutive BSS algorithm.

In stereophonic acoustic environment, the stereophonic signals $\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$ are transmitted to louder speakers in the receiving room and coupled to the microphones in this room by the room impulse responses. In stereophonic acoustic echo cancellation, these coupled acoustic echoes have to be cancelled. Let the receiving room impulse responses for

$\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$ be $\mathbf{h}_1(n)$ and $\mathbf{h}_2(n)$, respectively. Two adaptive filters $\hat{\mathbf{h}}_1(n)$ and $\hat{\mathbf{h}}_2(n)$ of length L in stereophonic acoustic echo canceller are updated to estimate $\mathbf{h}_1(n)$ and $\mathbf{h}_2(n)$. The desired signal for the adaptive filters is

$$d(n) = \sum_{j=1}^{2} \mathbf{h}_j^T(n)\mathbf{x}_j(n), \tag{17}$$

where $\mathbf{h}_j(n) = [h_{j,0}(n), h_{j,1}(n), \ldots, h_{j,L-1}(n)]^T$ and $\mathbf{x}_j(n) = [x_j(n), x_j(n-1), \ldots, x_j(n-L+1)]^T$.

Thus, the error signal is

$$e(n) = d(n) - \sum_{j=1}^{2} \hat{\mathbf{h}}_j^T(n)\mathbf{x}_j(n). \tag{18}$$

Adaptive algorithms such as LMS, NLMS, RLS, and affine projection (AP) can be used to update these two adaptive filters $\hat{\mathbf{h}}_1(n)$ and $\hat{\mathbf{h}}_2(n)$. The exclusive maximum tap-selection scheme is outlined in the following.

(1) At each iteration, calculate the interchannel tap-input magnitude difference vector as $\mathbf{p} = |\mathbf{x}_1| - |\mathbf{x}_2|$.

(2) Sort $\mathbf{p}$ in descending order as $\breve{\mathbf{p}} = [\breve{p}_1, \ldots, \breve{p}_L]^T$, $\breve{p}_1 > \breve{p}_2 > \cdots > \breve{p}_L$.

(3) Order $\mathbf{x}_1$ and $\mathbf{x}_2$ according to the sorting of $\breve{\mathbf{p}}$ as $\breve{\mathbf{x}}_1 = [\breve{x}_1(n), \breve{x}_1(n-1), \ldots, \breve{x}_1(n-L+1)]^T$ and $\breve{\mathbf{x}}_2 = [\breve{x}_2(n), \breve{x}_2(n-1), \ldots, \breve{x}_2(n-L+1)]^T$.

(4) The first channel coefficients corresponding to the M largest elements of $\mathbf{p}$ get updated and the second channel coefficients corresponding to M smallest elements of $\mathbf{p}$ get updated.

It was shown in [11] that this update mechanism applying to LMS, NLMS, RLS, and affine projection (AP) algorithms results in significantly better convergence rate than their existing corresponding algorithms.

5. PROPOSED MMAX PARTIAL UPDATE TIME-DOMAIN CONVOLUTIVE BSS ALGORITHM

From the description of MMax partial update in Section 3, we know that the principle of MMax partial update algorithm for single channel is to update the subset of coefficients which has the most impact on $\Delta\mathbf{w}$. Our proposed MMax partial update convolutive BSS algorithm is based on the same principle.

In the MMax LMS algorithm [10], given $\Delta\mathbf{w}(n) = e(n)\mathbf{x}(n)$, the $e(n)$ is common to all elements of $\Delta\mathbf{w}(n)$, then the larger the $|x(n-i)|$, the larger its impact on error. Thus, in MMax LMS algorithm, the coefficients corresponding to M largest values in $|\mathbf{x}(n)|$ are updated.

However, in time-domain convolutive BSS, $\Delta\underline{\mathbf{W}}$ is as follows:

$$\Delta\underline{\mathbf{W}} = -\mu\frac{\partial D}{\partial\underline{\mathbf{W}}}\underline{\mathbf{W}}^T\underline{\mathbf{W}} = \mu[\mathbf{I} - E(\varphi(\mathbf{y})\mathbf{y}^T)]\underline{\mathbf{W}}. \tag{19}$$

Every element of $\underline{\mathbf{W}}$ is an FIR filter and there is no common value for all elements of $\Delta\mathbf{W}$. Based on MMax partial update

(1) Initialize $\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix}$.

(2) Iteration k

$\mathbf{x}_1 = \{x_1(k), x_1(k-1), \ldots, x_1(k-L+1)\}$;

$\mathbf{x}_2 = \{x_2(k), x_2(k-1), \ldots, x_2(k-L+1)\}$;

$y_1 = \mathbf{w}_{11} \times \mathbf{x}_1^T + \mathbf{w}_{12} \times \mathbf{x}_2^T$;

$y_2 = \mathbf{w}_{21} \times \mathbf{x}_1^T + \mathbf{w}_{22} \times \mathbf{x}_2^T$;

$u_1 = \tanh(y_1)$;

$u_2 = \tanh(y_2)$;

$\Delta\mathbf{W} = \left\{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \times \begin{bmatrix} y_1 & y_2 \end{bmatrix}\right\} \times \mathbf{W}$;

$\Delta\mathbf{W}_{\text{new}} = \begin{bmatrix} \mathbf{Q}_{11} \times \Delta\mathbf{w}_{11} & \mathbf{Q}_{12} \times \Delta\mathbf{w}_{12} \\ \mathbf{Q}_{21} \times \Delta\mathbf{w}_{21} & \mathbf{Q}_{22} \times \Delta\mathbf{w}_{22} \end{bmatrix}$;

$\mathbf{Q}_{ij} = \text{diag}\{\mathbf{q}_{ij}^T\}, \quad i, j = 1, 2$;

$\mathbf{q}_{ij}(m) = \begin{cases} 1 & \Delta\mathbf{W}_{ij}(m) \in \{M \text{ maxima of } \Delta\mathbf{w}_{ij}\} \\ 0 & \text{otherwise}; \end{cases}$

$\mathbf{W} = \mathbf{W} + \mu \times \Delta\mathbf{W}_{\text{new}}$;

$k = k + 1$.

(3) Go to step 2 to start a new iteration.

ALGORITHM 1: MMax partial update convolutive BSS algorithm.

principle, the coefficients with the M largest values of $\Delta\mathbf{W}_{ij}$ are the ones to be updated. We show this algorithm using a 2-by-2 system as an example in Algorithm 1.

From the algorithm description, the challenge compared to the MMax LMS algorithm [10] is that we need to sort the elements in $\Delta\mathbf{W}_{ij}$ *in every iteration*, as opposed to simply identifying the location of one new sample in an already ordered set. However, we only need to update the selected subset of coefficients, which results in some savings.

6. PROPOSED EXCLUSIVE MAXIMUM SELECTIVE-TAP TIME-DOMAIN CONVOLUTIVE BSS ALGORITHM

As we already know from Section 4, exclusive maximum tap selection can reduce interchannel correlation and improve the conditioning of the input autocorrelation matrix. In this section, we examine the effect of tap selection on interchannel coherence reduction and extend this idea to our multichannel blind source separation case.

6.1. Interchannel decorrelation by tap selection

The squared coherence function of $\mathbf{x}_1$, $\mathbf{x}_2$ is defined as

$$C_{\mathbf{x}_1\mathbf{x}_2}(f) = \frac{|P_{\mathbf{x}_1\mathbf{x}_2}(f)|^2}{P_{\mathbf{x}_1\mathbf{x}_1}(f)P_{\mathbf{x}_2\mathbf{x}_2}(f)}, \tag{20}$$

where $P_{\mathbf{x}_1\mathbf{x}_2}(f)$ is the cross-power spectrum between the two mixtures $\mathbf{x}_1$, $\mathbf{x}_2$ and f is the normalized frequency [11].
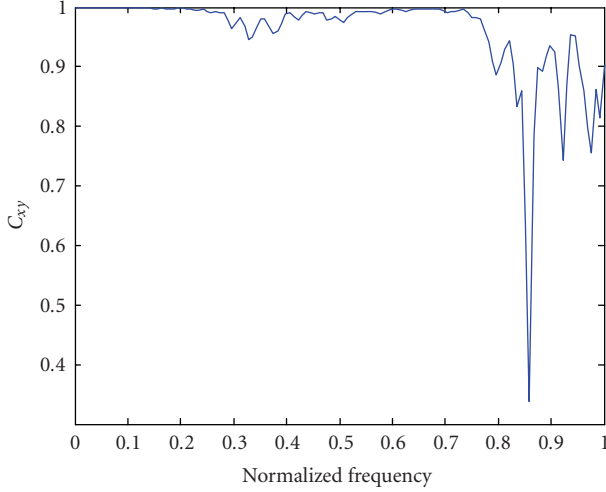
FIGURE 4: Squared coherence for $\mathbf{x}_1$ and $\mathbf{x}_2$ with full tap inputs selected.

FIGURE 5: Squared coherence for $\mathbf{x}_1$ and $\mathbf{x}_2$ with 50% MMax tap inputs selected.

A two-input two-output system is considered in this section. The mixing system used in the simulation is as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_{11} & \mathbf{h}_{12} \\ \mathbf{h}_{21} & \mathbf{h}_{22} \end{bmatrix},$$

$$\mathbf{h}_{11} = \begin{bmatrix} 1 & 0.8 & -0.2 & 0.78 & 0.4 & -0.2 & 0.1 \end{bmatrix},$$

$$\mathbf{h}_{22} = \begin{bmatrix} 0.8 & 0.6 & 0.1 & -0.1 & 0.3 & -0.2 & 0.1 \end{bmatrix}, \tag{21}$$

$$\mathbf{h}_{12} = \gamma \mathbf{h}_{11} + (1 - \gamma)\mathbf{b},$$

$$\mathbf{h}_{21} = \gamma \mathbf{h}_{22} + (1 - \gamma)\mathbf{b},$$

where $\mathbf{b}$ is an independent white Gaussian noise with zero mean.

In the simulation, we set $\gamma = 0.9$ to reflect the high interchannel correlation found in practice between the observed mixtures in a convolutive environment. The two-tap input signals $\mathbf{s}_1$ and $\mathbf{s}_2$ are generated as zero mean, unit variance gamma signals. The mixtures $\mathbf{x}_1$ and $\mathbf{x}_2$ are obtained from the following equations:

$$\mathbf{x}_1 = \mathbf{s}_1 * \mathbf{h}_{11} + \mathbf{s}_2 * \mathbf{h}_{12},$$

$$\mathbf{x}_2 = \mathbf{s}_1 * \mathbf{h}_{21} + \mathbf{s}_2 * \mathbf{h}_{22}, \tag{22}$$

where $*$ is convolution operation.

The squared coherence for the $\mathbf{x}_1$ and $\mathbf{x}_2$ with full taps selected is shown in Figure 4. In Figure 5, the squared coherence for inputs with taps selected according to the MMax selection criterion as described in Section 4 is shown. We can see that the correlation is reduced, but not significantly. Figure 6 shows the squared coherence for signals with exclusive tap selected, that is, the selection of the same tap index in both channels is not permitted. We can see that the correlation is reduced significantly. This confirms that exclusive tap-selection strategy does indeed reduce interchannel coherence and as such improves the conditioning of the input autocorrelation matrix even in the mixing environment of blind source separation case.
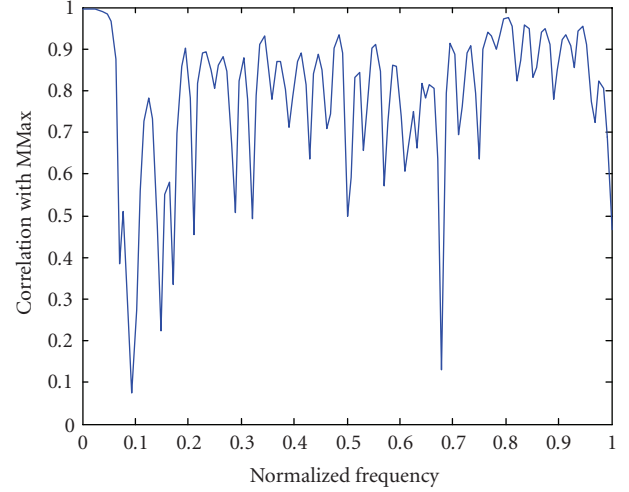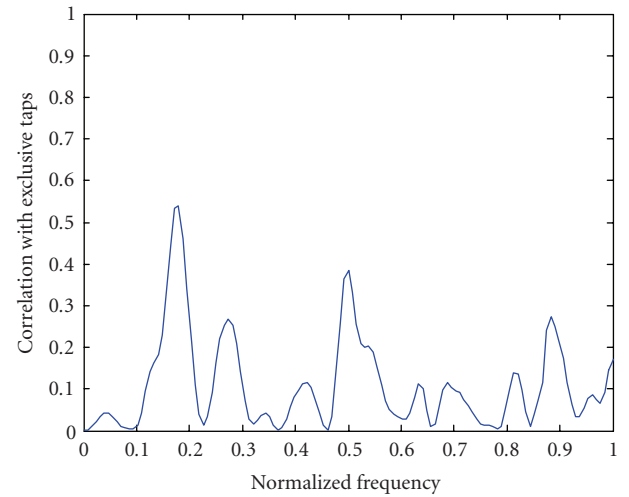

FIGURE 6: Squared coherence for $\mathbf{x}_1$ and $\mathbf{x}_2$ with exclusive maximum tap inputs selected.

6.2. Proposed XM update algorithm for time-domain convolutive BSS

As a result of improved conditioning of input autocorrelation matrix, we expect improved convergence rate in time-domain convolutive BSS when using this update algorithm for a two-by-two blind source separation system.

Based on the exclusive maximum tap-selection scheme proposed in [11], we propose the exclusive maximum time-domain convolutive BSS algorithm (XM BSS) as follows.

Define $\mathbf{p}$ as the interchannel tap input magnitude difference vector at time n as

$$\mathbf{p} = |\mathbf{x}_1| - |\mathbf{x}_2|. \tag{23}$$

Sort $\mathbf{p}$ in descending order as

$$\breve{\mathbf{p}} = [\breve{p}_1, \ldots, \breve{p}_L]^T, \quad \breve{p}_1 > \breve{p}_2 > \cdots > \breve{p}_L. \qquad (24)$$

Order $\mathbf{x}_1$ and $\mathbf{x}_2$ according to the sorting of $\breve{\mathbf{p}}$ such that $\breve{x}_1(n-i)$ and $\breve{x}_2(n-i)$ correspond to $\breve{p}_i = |\breve{x}_1(n-i)| - |\breve{x}_2(n-i)|$.

Taps corresponding to the $M = 0.5L$ largest elements of the input magnitude difference vector $\mathbf{p}$ in the first channel and the M smallest elements of $\mathbf{p}$ in the second channel are selected for the updating of the output signal y_1; Taps corresponding to the $M = 0.5L$ largest elements of the input magnitude difference vector $\mathbf{p}$ in the second channel and the M smallest elements of $\mathbf{p}$ in the first channel are selected for the updating of the output signal y_2. The detailed algorithm is shown in Algorithm 2.

6.3. Computational complexity of the proposed algorithm

The complexity is defined as the total number of multiplications and comparisons per sample period for each channel. In XM convolutive BSS algorithm, we need to sort the interchannel tap input magnitude difference vector. For an unmixing system with filter length L, we require at most $2 + 2\log_2 L$ comparisons per sample period by the SORTLINE procedure [13]. However, the number of multiplications required for computing convolution per sample period is reduced from $4L$ to $2L$ for a two-by-two BSS system. Thus, the overall computational complexity is still reduced provided $L > 2$, which is always satisfied for convolutive BSS case.

7. SEPARATION PERFORMANCE EVALUATION

In this section, we describe separation performance evaluation measurement used in our simulations.

7.1. Performance evaluation by signal-to-interference ratio

The performance of blind source separation systems can be evaluated by the signal-to-interference ratio (SIR) which is defined as the power ratio between the target component and the interference components [14].

In basic instantaneous BSS model, the mixing system is represented with $\mathbf{A}$, the unmixing system is represented with $\mathbf{W}$, the global system can be presented as $\mathbf{P} = \mathbf{A} * \mathbf{H}$. Each element in ith row and jth column of $\mathbf{P}$ is a scalar p_{ij}. The SIR of output i is obtained as

$$\text{SIR}_i = 10\log_{10}\frac{E\{p_{ii}s_i\}}{E\{\sum_{j \neq i} p_{ij}s_j\}}\text{dB} \qquad (25)$$

for instantaneous BSS case.

In the convolutive BSS model, the mixing system is represented with $\mathbf{H}$, the unmixing system with $\mathbf{W}$. We can express the global system as $\mathbf{P} = \mathbf{W} * \mathbf{H}$ and each element in $\mathbf{P}$ is a vector $\mathbf{p}_{ij}$.

---

(1) Initialize $\mathbf{W} = \begin{bmatrix} \mathbf{w}_{11} & \mathbf{w}_{12} \\ \mathbf{w}_{21} & \mathbf{w}_{22} \end{bmatrix}$.

(2) Iteration k

$\quad \mathbf{x}_1 = \{x_1(k), x_1(k-1), \ldots, x_1(k-L+1)\}$;

$\quad \mathbf{x}_2 = \{x_2(k), x_2(k-1), \ldots, x_2(k-L+1)\}$;

$\quad \mathbf{p} = |\mathbf{x}_1| - |\mathbf{x}_2|$;

$\quad \bar{\mathbf{x}}_{11} = \mathbf{Q}_{11} \times \mathbf{x}_1; \qquad \bar{\mathbf{x}}_{21} = \mathbf{Q}_{21} \times \mathbf{x}_1$;

$\quad \bar{\mathbf{x}}_{12} = \mathbf{Q}_{12} \times \mathbf{x}_2; \qquad \bar{\mathbf{x}}_{22} = \mathbf{Q}_{22} \times \mathbf{x}_2$;

$\quad \mathbf{Q}_{11} = \text{diag}\{\mathbf{q}_{11}^T\}$;

$\quad \mathbf{q}_{11}(m) = \begin{cases} 1 & p(m) \in \{M \text{ maxima of } \mathbf{p}\} \\ 0 & \text{otherwise}; \end{cases}$

$\quad \mathbf{Q}_{12} = \text{diag}\{\mathbf{q}_{12}^T\}$;

$\quad \mathbf{q}_{12}(m) = \begin{cases} 1 & p(m) \in \{M \text{ minimum of } \mathbf{p}\} \\ 0 & \text{otherwise}; \end{cases}$

$\quad \mathbf{Q}_{21} = \text{diag}\{\mathbf{q}_{21}^T\}$;

$\quad \mathbf{q}_{21}(m) = \begin{cases} 1 & p(m) \in \{M \text{ minimum of } \mathbf{p}\} \\ 0 & \text{otherwise}; \end{cases}$

$\quad \mathbf{Q}_{22} = \text{diag}\{\mathbf{q}_{22}^T\}$;

$\quad \mathbf{q}_{22}(m) = \begin{cases} 1 & p(m) \in \{M \text{ maxima of } \mathbf{p}\} \\ 0 & \text{otherwise}; \end{cases}$

$\quad \bar{y}_1 = \mathbf{w}_{11} \times \bar{\mathbf{x}}_{11}^T + \mathbf{w}_{12} \times \bar{\mathbf{x}}_{12}^T$;

$\quad \bar{y}_2 = \mathbf{w}_{21} \times \bar{\mathbf{x}}_{21}^T + \mathbf{w}_{22} \times \bar{\mathbf{x}}_{22}^T$;

$\quad u_1 = \tanh(\bar{y}_1)$;

$\quad u_2 = \tanh(\bar{y}_2)$;

$\quad \Delta\mathbf{W} = \left\{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \times \begin{bmatrix} \bar{y}_1 & \bar{y}_2 \end{bmatrix}\right\} \times \mathbf{W}$;

$\quad \mathbf{W} = \mathbf{W} + \mu \times \Delta\mathbf{W}$;

$\quad k = k + 1$.

(3) Go to 2 to start another iteration.

(4) Calculate separated signals as

$\quad y_1 = \mathbf{w}_{11} \times \mathbf{x}_1^T + \mathbf{w}_{12} \times \mathbf{x}_2^T$;

$\quad y_2 = \mathbf{w}_{21} \times \mathbf{x}_1^T + \mathbf{w}_{22} \times \mathbf{x}_2^T$.

ALGORITHM 2: XM convolutive BSS algorithm.

The SIR of output i is obtained as

$$\text{SIR}_i = 10\log_{10}\frac{E\{\mathbf{p}_{ii} * \mathbf{s}_i\}}{E\{\sum_{j \neq i} \mathbf{p}_{ij} * \mathbf{s}_j\}}\text{dB} \qquad (26)$$

for convolutive BSS case, where $*$ is the convolution operation and $E\{\}$ is the expectation operation.

7.2. Performance evaluation by PESQ

When the target signal in our simulations is a speech signal, we will also use PESQ (perceptual evaluation of speech quality) as a measure confirming the quality of the separated signal. The PESQ standard [15] is described in the ITU-T P862 as a perceptual evaluation tool of speech quality. The key feature of the PESQ standard is that it uses a perceptual model analogous to the assessment by the human auditory system. The output of the PESQ is a measure of the subjective assessment quality of the degraded signal and is rated as a value between -0.5 and 4.5 which is known as the mean opinion score (MOS). The larger the score, the better the speech quality.

8. SIMULATIONS

8.1. Experiment setup

In the following simulations, our source signals $\mathbf{s}_1$ and $\mathbf{s}_2$ are generated as gamma signals or speech signals. The gamma signals are generated with zero mean, unit variance. The speech signals used in our simulations include 3 female speeches and 3 male speeches with sample rate 8000 Hz to form 9 combinations. A simple mixing system is used in our simulations to demonstrate and compare separation performance.

The mixing system is given by

$$\mathbf{H} = \begin{bmatrix} 1.0 & 1.0 & -0.75; & -0.2 & 0.4 & 0.7 \\ 0.2 & 1.0 & 0.0; & 0.5 & -0.3 & 0.2 \end{bmatrix}. \quad (27)$$

The mixture signals are obtained by convolving the source signals with the mixing system. The filter length in the separation system is set at 64.

In the following, we will compare the separation performance of the regular convolutive BSS algorithm, MMax partial update BSS algorithm, and XM selective-tap BSS algorithm.

8.2. MMax partial update time-domain BSS algorithm for convolutive mixture

In this simulation, we test the performance of MMax partial update time-domain BSS algorithm for convolutive mixtures. In the following diagram, "reg" means regular time-domain BSS algorithm; "par56" means MMax partial update time domain BSS algorithm with $M = 56$; "par48" means MMax partial update time-domain BSS algorithm with $M = 48$; "par32" means MMax partial update time-domain BSS algorithm with $M = 32$, where M is the number of coefficients updated at each iteration in a given channel.

In the first experiment, we use generated gamma signals as the original signals and use (9) to get the mixture signals. The performance of regular time-domain convolutive BSS algorithm and MMax partial update convolutive BSS algorithm evaluated by the SIR measure defined in (26) is shown in Figures 7 and 8.
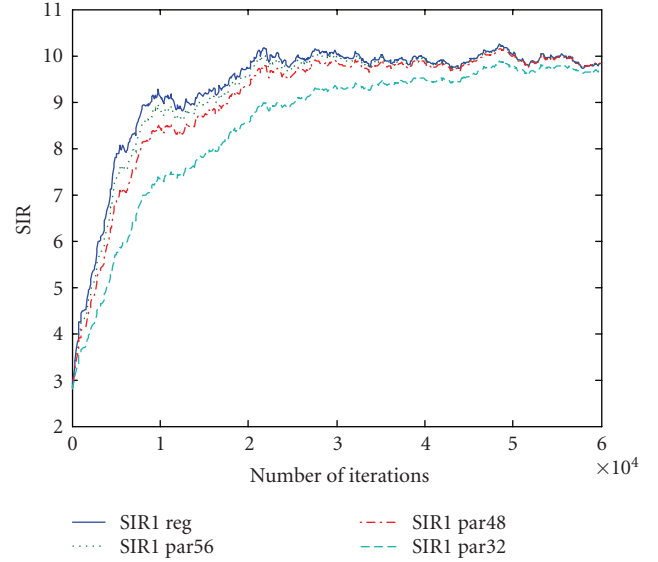

FIGURE 7: Separation performance of time-domain regular convolutive BSS and MMax partial update BSS for gamma signal measured by SIR for the first output.
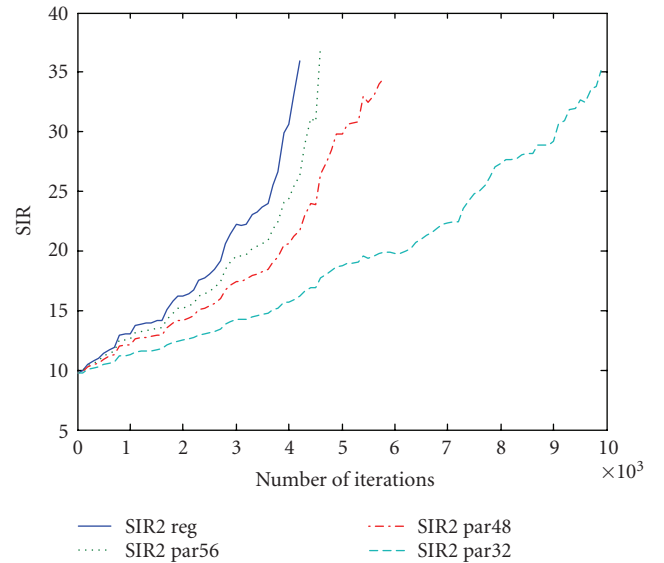

FIGURE 8: Separation performance of time-domain regular convolutive BSS and MMax partial update BSS for gamma signal measured by SIR for the second output.

From these diagrams, we can see that as expected, the MMax partial update convolutive BSS algorithm converges slightly slower than the regular BSS algorithm while only a subset of coefficients gets updated. However, it converges to similar SIR values.

In the second experiment, we use speech signals as the original signals and use the same mixing system to get the mixture signals. In Figures 9 and 10, we show the performance of regular time-domain convolutive BSS algorithm and MMax partial update BSS convolutive algorithm for one
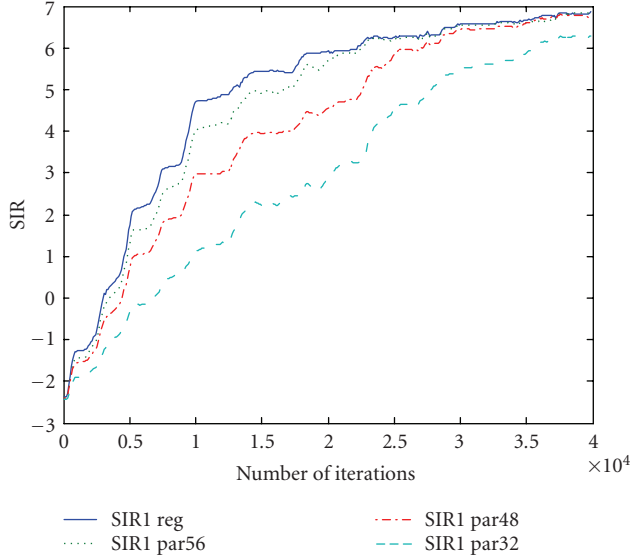
FIGURE 9: Separation performance of time-domain regular convolutive BSS and MMax partial update BSS for speech signal measured by SIR for the first output.
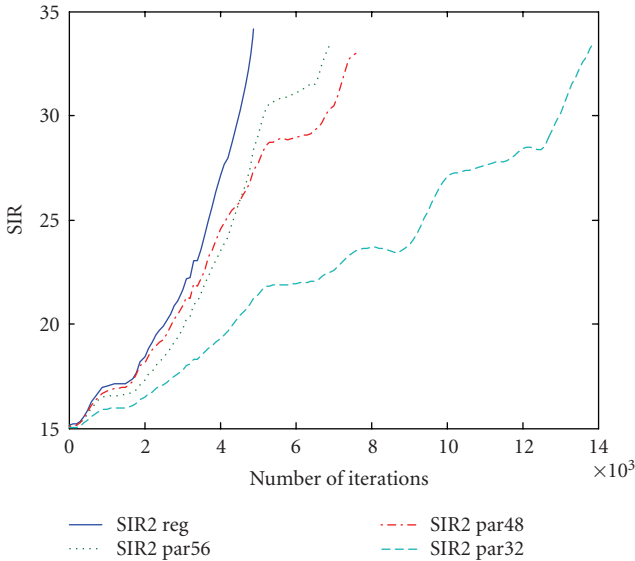

FIGURE 10: Separation performance of time-domain regular convolutive BSS and MMax partial update BSS for speech signal measured by SIR for the second output.

combination of speech signals, the separation performance is evaluated by SIR. The performance for other combinations of speech signals is similar to that shown in Figures 9 and 10.

Since we used speech signals in the second experiment, we also use PESQ to evaluate the separation performance. In the following, we evaluate the similarity between the mixtures, the separated signals from regular and MMax BSS algorithms with the original source signals by PESQ score. Table 1 shows the average PESQ evaluation results for different combinations of female and male speech signals, where (S1,S2)

present the original source signals; (mix1,mix2) present the mixture signals; (regular out1, regular out2) present separated signals from regular BSS algorithm; (partial $M = 56$ out1, partial $M = 56$ out2) present separated signals from MMax BSS algorithm with $M = 56$; (partial $M = 48$ out1, partial $M = 48$ out2) present separated signals from MMax BSS algorithm with $M = 48$; (partial $M = 32$ out1, partial $M = 32$ out2) present separated signals from MMax BSS algorithm with $M = 32$.

From Table 1, we can see that the separation performance evaluated by PESQ is consistent with the SIR results. The separation algorithms make the separated signals more biased to one source signal and away from the other source signal. The separation performance evaluated by PESQ and SIR is also consistent with our informal listening tests.

From the above simulation results, we can see that similar to MMax NLMS algorithm for single-channel linear filters, there is a slight deterioration in performance of the proposed MMax partial update time-domain convolutive BSS algorithm as the number of updated coefficients is reduced. However, the performance at 50% coefficients updated is still quite acceptable.

8.3. Time-domain exclusive maximum selective-tap BSS for convolutive mixture

In this simulation, we test the performance of XM selective tap time-domain BSS algorithm for convolutive mixtures.

In the first experiment, we use generated gamma signals as the original signals and use (9) to get the mixture signals. The performance of regular time-domain convolutive BSS algorithm and XM selective-tap convolutive BSS algorithm evaluated by SIR is shown in Figures 11 and 12.

From Figures 11 and 12, we can see that XM BSS algorithm has much better convergence rate compared with regular BSS algorithm for generated gamma signals.

In the second experiment, we use speech signals as the original signals and use the same mixing system to get the mixture signals. In Figures 13 and 14, we show the performance of regular time-domain convolutive BSS algorithm and XM selective tap BSS convolutive algorithm for one combination of speech signals, the separation performance is evaluated by SIR. The performance for other combinations of speech signals is similar with that shown in Figures 13 and 14.

From the plots, we can see that the XM BSS algorithm has much better convergence rate compared with the regular BSS algorithm for both generated gamma signals and speech signals.

Since we used speech signals in the second experiment, we also use PESQ to evaluate the separation performance. In the following, we evaluate the similarity between the mixtures, the separated signals from regular and XM BSS algorithms with the original source signals by PESQ score. Table 2 shows the average PESQ evaluation results for different combinations of female and male speech signals, where (S1, S2) present the original source signals; (mix1, mix2) present the mixture signals; (regular BSS out1, out2) present separated

TABLE 1: Average PESQ scores for mixtures and separated signals from regular BSS algorithm and MMax BSS algorithm.

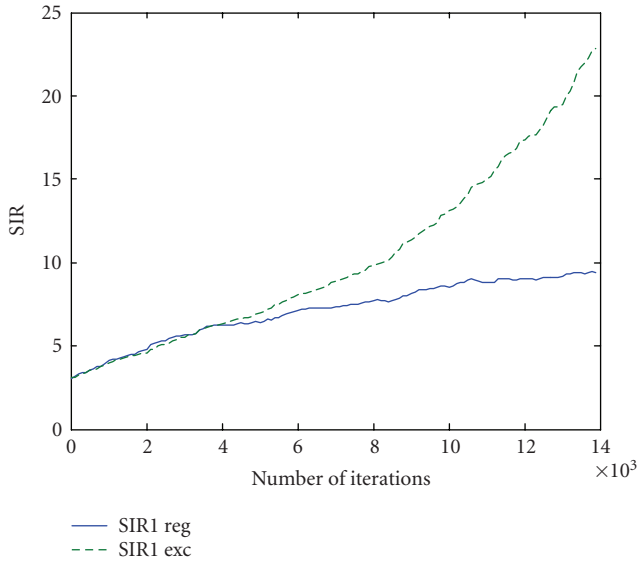| PESQ	Mixture		Regular		Partial $M = 56$		Partial $M = 48$		Partial $M = 32$	
	mix1	mix2	out1	out2	out1	out2	out1	out2	out1	out2
S1	2.119	0.981	2.379	0.612	2.365	0.611	2.352	0.602	2.340	0.599
S2	1.364	2.374	1.076	2.771	1.105	2.702	1.148	2.659	1.029	2.624

FIGURE 11: Separation performance of time-domain regular convolutive BSS and XM selective tap BSS for gamma signal measured by SIR for the first output.
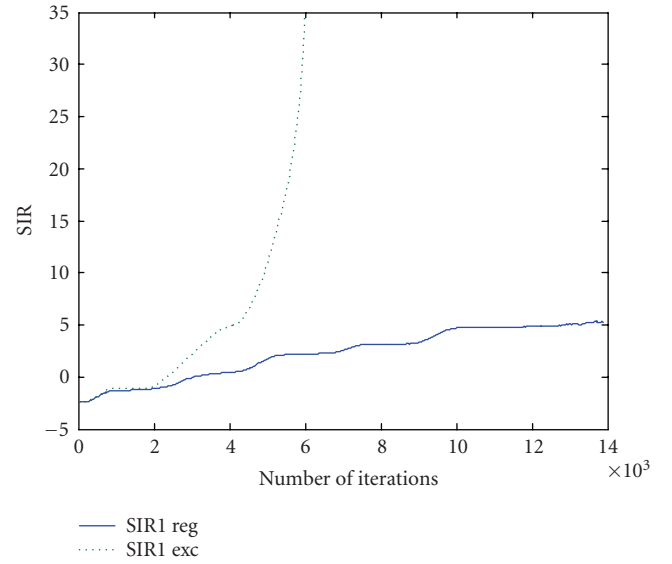

FIGURE 13: Separation performance of time-domain regular convolutive BSS and XM selective tap BSS for speech signal measured by SIR for the first output.
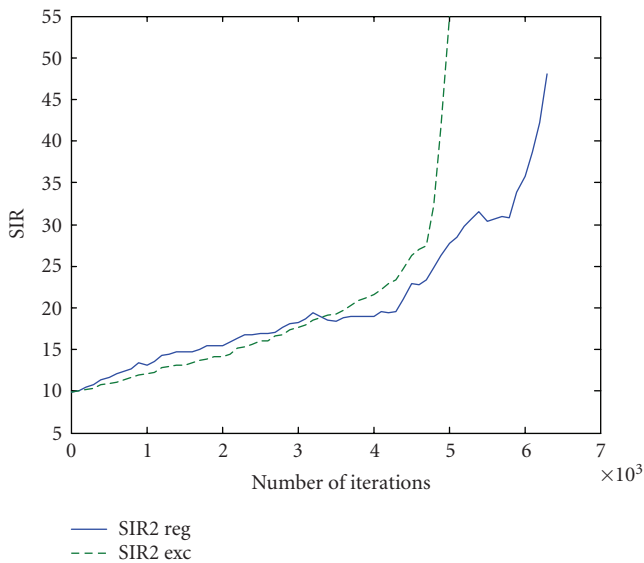

FIGURE 12: Separation performance of time-domain regular convolutive BSS and XM selective tap BSS for gamma signal measured by SIR for the second output.
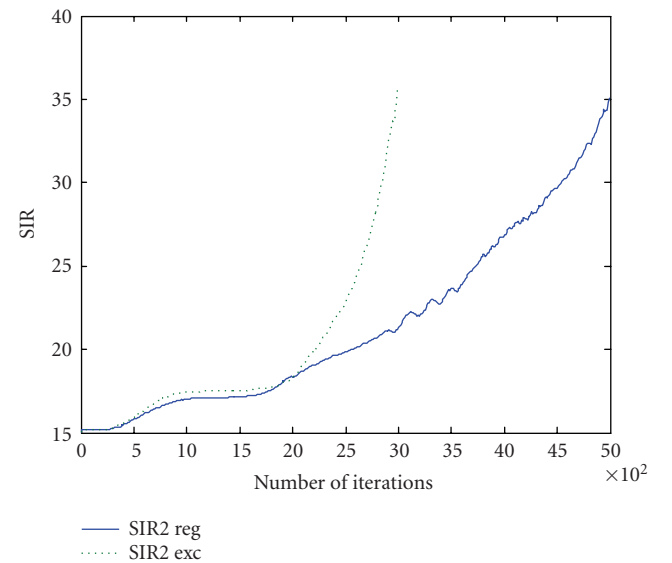

FIGURE 14: Separation performance of time-domain regular convolutive BSS and XM selective tap BSS for speech signal measured by SIR for the second output.

TABLE 2: Average PESQ scores for mixtures and separated signals from regular BSS algorithm and XM BSS algorithm.

PESQ	Mixture		Regular BSS		Xmax BSS	
	mix1	mix2	out1	out2	out1	out2
S1	1.871	0.948	2.037	0.591	2.643	0.463
S2	1.583	2.255	1.215	2.547	1.055	2.560

signals from regular BSS algorithm; (XM BSS out1, out2) present separated signals from XM BSS. The performance evaluation by PESQ is consistent with that measured by SIR. The separation performance evaluated by PESQ and SIR is also consistent with our informal listening tests.

Based on the above simulation, we can see that XM BSS algorithm significantly improves the convergence rate compared with regular time-domain convolutive BSS algorithm.

9. CONCLUSION

In this paper, we investigate time-domain convolutive BSS algorithm and propose two novel algorithms to address the slow convergence rate and high computational complexity problem in time-domain BSS. In the proposed MMax partial update time domain convolutive BSS algorithm (MMax BSS), only a subset of coefficients in the separation system gets updated at every iteration. We show that the partial update scheme applied in the MMax LMS algorithm for single channel can be extended to multichannel natural gradient-based time-domain convolutive BSS with little deterioration in performance and possible computation complexity saving. In the proposed exclusive maximum selective-tap time-domain convolutive BSS algorithm (XM BSS), the exclusive tap-selection update procedure reduces the interchannel coherence of the tap-input vectors and improves the conditioning of the autocorrelation matrix so as to accelerate convergence rate and reduce the misalignment. Moreover, the computational complexity is reduced as well since only half of tap inputs are selected for updating. Simulation results have shown a significant improvement in convergence rate compared with existing techniques. The extension of the proposed XM BSS algorithm to more than two channels is still an open problem.

REFERENCES

[1] S. Haykin, Ed., *Unsupervised Adaptive Filtering, Volume 1: Blind Source Separation*, John Wiley & Sons, New York, NY, USA, 2000.

[2] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing*, John Wiley & Sons, New York, NY, USA, 2000.

[3] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, New York, NY, USA, 2001.

[4] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, "Multichannel blind deconvolution and equalization using the natural gradient," in *Proceedings of the 1st IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications (SPAWC '97)*, pp. 101–104, Paris, France, April 1997.

[5] S. C. Douglas and X. Sun, "Convolutive blind separation of speech mixtures using the natural gradient," *Speech Communication*, vol. 39, no. 1-2, pp. 65–78, 2003.

[6] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, no. 1–3, pp. 21–34, 1998.

[7] L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 3, pp. 320–327, 2000.

[8] 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 Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 530–538, 2004.

[9] M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, vol. 1, pp. 881–884, Orlando, Fla, USA, May 2002.

[10] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Transactions on Signal Processing*, vol. 47, no. 5, pp. 1421–1424, 1999.

[11] A. W. H. Khong and P. A. Naylor, "Stereophonic acoustic echo cancellation employing selective-tap adaptive algorithms," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 3, pp. 785–796, 2006.

[12] S. Werner, M. L. R. de Campos, and P. S. R. Diniz, "Partial-update NLMS algorithms with data-selective updating," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 938–949, 2004.

[13] I. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 6, pp. 795–804, 1989.

[14] S. Makino, H. Sawada, R. Mukai, and S. Araki, "Blind source separation of convolutive mixtures of speech in frequency domain," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E88-A, no. 7, pp. 1640–1654, 2005.

[15] ITU-T Recommend P.862, "Perceptual evaluation of speech quality (PESQ), an objective method for end-to end speech quality assessment of narrowband telephone network and speech codecs," May 2000.