MULTISCALE MODEL. SIMUL.
Vol. 7, No. 4, pp. 1795–1810

A SOFT-CONSTRAINED DYNAMIC ITERATIVE METHOD OF BLIND SOURCE SEPARATION*

JIE LIU†, JACK XIN†, AND YINGYONG QI‡

Abstract. The blind source separation problem arises when one attempts to recover source signals from their linear mixtures without detailed knowledge of the mixing process. Solutions are nonunique and have degrees of freedom in scaling and permutation. One may impose equality (hard) constraints to fix these scaling parameters; however, small divisor problems may appear. In this paper, an iterative method is introduced based on information maximization and auxiliary equations for the scaling parameters. The method dynamically selects scaling parameters and avoids divisions, and it is called the soft-constrained method. Global boundedness of the algorithm is proved. The convergence of solutions in the large time and small step size regimes is analyzed. An upscaled, dynamically averaged equation for the separating matrix is derived. The stable and accurate separation performance is illustrated by examples of instantaneous random mixtures of two and eight sound signals.

Key words. blind source separation, soft constraints, division-free, global bounds, convergence

AMS subject classifications. 94A12, 65H10, 65C60

DOI. 10.1137/080736168

1. Introduction.

Blind source separation (BSS) is a major area of research in signal and image processing [5, 6, 8, 9, 13, 18]. It aims at recovering source signals from their mixtures without detailed knowledge of the mixing process. If source signals are time series and modeled statistically as realizations of stochastic processes, as is common for sound signals, one may use the independence property of source signals to find the mixing matrix in the case of linear mixing. The standard linear instantaneous mixing model is

$$X(t) = A_0 S(t), \tag{1.1}$$

where $S(t) \in \mathbb{R}^n$ is the time-dependent source signal vector, n is the number of sources, $A_0 \in \mathbb{R}^{n\times n}$ is the time-independent unknown mixing matrix, and $X(t)$ is the known mixture signal. The goal of BSS is to recover the source signal vector S, without knowing A_0 (therefore blind), provided that all components of $S(t)$ are independent of each other.

To date, there are two types of algorithms in the literature, dynamic (on-line) and batch (off-line). The dynamic method has only limited access to received data, as in real-time processing, while the batch method has access to all available data at the expense of delays, as in processing recordings. The algorithms also differ in objective functions for realizing the independence assumption. For example, the joint-approximate-diagonalization-eigenmatrices (JADE) [7] is a batch method using both

*Received by the editors September 23, 2008; accepted for publication (in revised form) April 22, 2009; published electronically July 10, 2009. This work was partially supported by NSF grant DMS-0712881, NIH grant 2R44DC006734, CORCLR (Academic Senate Council on Research, Computing and Library Resources) faculty research grant MI-2006-07-6, and a Pilot award of the Center for Hearing Research at UC Irvine.

http://www.siam.org/journals/mms/7-4/73616.html

†Department of Mathematics, University of California, Irvine, Irvine, CA 92697 (liuj@math.uci.edu, jxin@math.uci.edu).

‡Department of Mathematics, University of California, Irvine, Irvine, CA 92697, and School of Information Engineering, Shangdong University at Weihai, Weihai, P. R. China (yqi@uci.edu).

second and fourth order statistics of the data streams. It has a well-developed mathematical theory based on linear algebra. The information-maximization (infomax) method [4, 9] is a dynamic method that updates the approximation of the mixing matrix in time with newly received data. Because the received data is limited, the method is stochastic in nature and may only converge to the desired solution in the large time limit up to a scaling and permutation. Though the infomax method is easy to implement, and has a much lower demand for computational resources than a batch method, one must make sure that it is globally bounded (no blow-up) and convergent. As noted recently [10], the boundedness and convergence of the method remain an open problem, especially because there is no theoretical analysis. The authors of [10] proposed a scaled version of the infomax method, the so-called scaled natural gradient (SNG) method, to cure possible divergence. The SNG method [10] attempts to impose a *hard constraint* on the solutions, and *division operation* appears as a result. During a silence period of speech signals, the division operation may lead to a small divisor and instability of solutions, to say the least. The purpose of the current paper is to introduce a *soft-constrained* method to control solutions dynamically so that the method is *division free* and a priori estimates of solutions can be found to guarantee global boundedness and convergence of solutions in a suitable sense. The soft-constraints are realized in terms of two difference equations or discrete ordinary differential equations (ODEs).

The infomax method can be formally extended to the case of convolutive mixtures [3, 10] given by

$$X(t) = [A * S](t), \tag{1.2}$$

where $[A * S](t) = \sum_{i=0}^{q} A_i S(t-i)$ is the linear convolution with $\{A_i\}$, a group of $n \times n$ matrices. The convolutive mixtures appear naturally in room recordings of speech signals. The value of q depends on the mixing environment and may be as large as a thousand or more. The batch algorithm JADE cannot be directly generalized to convolutive mixtures. Though one could use FFT to localize the convolution and apply JADE in each frequency, permutation and scaling ambiguity require much more work, which can be costly and prone to additional errors [16, 17]. The soft-constrained method we introduce readily applies to convolutive mixtures; however, the analysis is more involved and shall be left for another publication. In this paper, we address the issue of uniform boundedness and convergence of the soft-constrained infomax method for the mixing model (1.1).

The infomax method of (1.1) estimates A_0^{-1} (the demixing matrix) by iteration of the form

$$W(k+1) = W(k) - \nu_k G(W(k), X(\cdot)), \tag{1.3}$$

where ν_k is a time step (learning parameter) and the nonlinear function G is the so-called estimating function [2] that depends on W and X. The estimation is up to a nontrivial scaling and permutation because the output is the same to the human ear within this group of actions. The degree of freedom in scaling is what is to be exploited to control the iteration (1.3). If X is a stationary and ergodic random signal, and (1.3) is well-behaved globally in time, the iteration formally converges to W_∞ which is a solution of the deterministic equation $E[G(W,X)] = 0$, where $E[\cdot]$ is the ensemble expectation of G with respect to the distribution of X.

The rest of the paper is organized as follows. In section 2, we give a brief overview and derivation of the estimating function G based on the maximum likelihood principle. In sections 3 and 4, we introduce the soft-constrained infomax iteration and prove its uniform boundedness and convergence. In the small ν limit, we derive an averaged (upscaled) ODE for the evolving demixing matrix. In section 5, we show numerical results on the separation of multiple sound signals in comparison with [10]. Conclusions are presented in section 6.

2. Derivation of infomax estimation function. In this section, we give a derivation of the estimating function G. Suppose the one-point distribution of S at time t is $r(s)$ with $s = (s_1, \dots, s_n)^\top$. Then by change of variable, the probability distribution function (pdf) of X at time t is

$$p(x) = p(x; A_0, r) = |A_0^{-1}| r(A_0^{-1}x) = |A_0^{-1}| \prod_{i=1}^{n} r_i([A_0^{-1}x]_i), \tag{2.1}$$

where $|A_0^{-1}| = \det(A_0^{-1})$, $s = A_0^{-1}x$, and $s_i = [A_0^{-1}x]_i$ is the ith component of $A_0^{-1}x$. We have used the independence assumption $r(s) = \prod_i r_i(s_i)$ in the last equality in (2.1). The time t does not enter explicitly because the problem is instantaneous in time. Following [2] to cast BSS as a semiparametric statistical inference problem, let us determine the matrix A_0 as parameters of the pdf p and estimate A_0 from samples of the random variable X.

Denote the one parameter family of pdf $p(x; A, r)$ as $\{p(x; A, r)\big|A\}$. Given observations of random variable X, A_0 is most likely to maximize the pdf function p or its logarithm on average as the best correlate of the events which occurred. The likelihood function is

$$L_d(A) \equiv E\big[\log p\big(X; A, r\big|A\big)\big], \tag{2.2}$$

where E is the empirical expectation with respect to the observed samples of the random variable X. The maximum likelihood estimation (MLE) of A_0 is to find A_0 to maximize $L_d(A)$, and so

$$E\left(\partial_A \log p(X; A, r)\Big|_{A=A_0}\right) = 0. \tag{2.3}$$

Let us vary A around A_0 as $A = A_0(I - K)$, $K = (K_{ij})$. We claim that

$$\frac{\partial}{\partial K_{ij}} \log p(x; A(K), r)\Big|_{K=0} = \delta_{ij} - f_i(s_i)s_j \tag{2.4}$$

with $f_i = -r_i'/r_i$. The result is known [2, 9]. For completeness of exposition, we give the following proof.

Proof. Because of $A = A_0(I-K)$ and (2.1), we have $\log p(x; A(K), r) = -\log|A_0|$ $-\log|I-K| + \sum_m \log r_m([A^{-1}x]_m)$. Recall that $\partial_B|B| = |B|B^{-\top}$ for any nondegenerate matrix B. Hence,

$$-\partial_{K_{ij}} \log|I-K|\Big|_{K=0} = \delta_{ij}.$$

In order to calculate $\partial_{K_{ij}} \log r_m([A^{-1}x]_m)\big|_{K=0}$, let us write

$$A^{-1}x = (I-K)^{-1}A_0^{-1}x = (I + K + K^2 + \cdots)A_0^{-1}x.$$

So,

$$\begin{aligned}
&\partial_{K_{ij}} \log r_m([A^{-1}x]_m)\Big|_{K=0} \\
&= -f_m([A^{-1}x]_m)\partial_{K_{ij}}([(I+K+K^2+\cdots)A_0^{-1}x]_m)\Big|_{K=0} \\
&= -\delta_{mi} f_i([A^{-1}x]_i)\partial_{K_{ij}}([(I+K+K^2+\cdots)A_0^{-1}x]_i)\Big|_{K=0} \\
&= -\delta_{mi} f_i(s_i)\partial_{K_{ij}}\left(\sum_k K_{ik}[A_0^{-1}x]_k\right) \\
&= -\delta_{mi} f_i(s_i)[A_0^{-1}x]_j \\
&= -\delta_{mi} f_i(s_i) s_j. \qquad \square
\end{aligned}$$

By (2.4) and (2.3), the separation matrix W satisfies

$$E(f(Y)Y^\top) = I \tag{2.5}$$

with $Y = WX = (y_1, \dots, y_n)^\top$ being the recovered source vector and

$$f(Y) = (f_1(y_1), \dots, f_i(y_i), \dots, f_n(y_n))^\top.$$

One may then choose the estimating function $G(W, X)$ in (1.3) to be

$$G(W,X) = E(f(Y)Y^\top - I), \quad \text{regular gradient,} \tag{2.6}$$

$$G(W,X) = E(f(Y)Y^\top - I)W, \quad \text{natural gradient.} \tag{2.7}$$

The natural gradient can be viewed as a regular gradient for the logarithm of W, and so it is naturally scaled relative to the size of W. It has a better mathematical property for numerical algorithms, as we shall see.

The amplitudes of speech signals obey the Laplace distribution [12]. For simplicity, one sets $r_i(s) = r(s) = (2\sigma)^{-1}\exp\{-|s|/\sigma\}$, which upon taking the log-derivative gives $f_i = f = \mathrm{sgn}(y)$, as is the case in our computation later.

We remark that (2.5), or its component form

$$E(f_i(y_i)y_j) = \delta_{ij}, \tag{2.8}$$

is quite natural. This is because if y_i and y_j are accurately estimated source signals, they are mean zero and nearly independent. So ideally we expect $E(f_i(y_i)y_j) = E(f_i(y_i))E(y_j) = 0$ for any function f_i as long as $i \neq j$. As a recovered source signal rescaled by an arbitrary constant is also an acceptable result, there are n scaling constants. We may view $E(f_i(y_i)y_i) = 1$, $i = 1, \dots, n$, as hard constraints (normalization condition) to fix the n scaling constants.

3. Soft-constrained infomax iteration. Equation (2.5) is not suitable for real-time processing because approximating expectation may take large blocks of data streams and cause delays. The idea is to realize (2.5) as a large time limit of a dynamic iteration. Suppose that the received signal $X(t) \in R^n$ comes from a mixture of the source signal vector $S(t) \in R^n$ with independent components. The mixture $X(t)$ is divided into blocks (frames) in time, each of which contains L data points. The infomax algorithm [4, 1] is

$$F(k) = \frac{1}{L}\sum_{i=kL+1}^{(k+1)L} f(Y^k(i))Y^k(i)^\top, \tag{3.1}$$

$$W(k+1) = (1+\mu)W(k) - \mu\, F(k)\, W(k), \tag{3.2}$$

where μ is the step size, f is a vector-valued nonlinear function, and $Y^k(i) = W(k)\,X(i)$. One can view $(I - F)W$ as an approximation in the kth block of the estimating function G of (2.7). For a suitable choice of f, the algorithm (3.1)–(3.2) approximately minimizes the mutual information of the output vector sequence Y over nonsingular matrices W (see [1]) and separates mixtures of sources if the source distributions do not differ much from those used to construct f. However, it is quite challenging to choose $W(0)$ and μ for fast convergence of the iteration without explosive divergence [10]. The standard initial condition is $W(0) = \delta\, I$ for a small δ, and μ is small in the iteration. To control possible divergence, the authors of [10] proposed the following scaled infomax iteration:

$$Y^k(i) = W(k)\,X(i) \qquad \text{with} \quad i = kL+1, \dots, (k+1)L, \tag{3.3}$$

$$W(k+1) = (1+\mu)\,c(k)\,W(k) - \mu\,\frac{c(k)}{d(k)}\,F(k)\,W(k), \tag{3.4}$$

where $c(k)$ is a scaling factor sequence defined as

$$c(k) = \frac{1}{h(d(k))}, \tag{3.5}$$

$$d(k) = \frac{1}{n}\sum_{i,j=1}^{n} |F_{ij}(k)|, \tag{3.6}$$

where h is the inverse function of the magnitude of $Y^\top f(Y)$ or $h(Y^\top f(Y)) = |Y|$ [10, equations (14)–(16)], and F_{ij} are elements of the $n \times n$ matrix F. For f equal to the sign function as in our computation, h equals identity, $c = 1/d$, and $c/d = 1/d^2$. Clearly, if $F \approx 0$ during a silence period in a speech signal, both c and c/d will blow up. This small divisor problem leads to the unstable nature of the computed matrix W and the occurrence of the exceedingly large peaks of its entries in time as we shall see later.

Let us introduce the following soft-constrained infomax iteration:

$$W(k+1) = (1+\nu_k\,\sigma_1(k))\,W(k) - \nu_k\,\sigma_2(k)H(k), \tag{3.7}$$

$$F_1(k) = (1+\nu_k\,\sigma_1(k))\,|W(k+1)| + \lambda(1+\nu_k\,\sigma_1(k)) - a, \tag{3.8}$$

$$F_2(k) = \sigma_2(k)|H(k+1)| + \lambda\sigma_2(k) - a, \tag{3.9}$$

$$1 + \nu_{k+1}\sigma_1(k+1) = (1+\nu_k\sigma_1(k))\,e^{-F_1(k)}, \tag{3.10}$$

$$\sigma_2(k+1) = \sigma_2(k)\,e^{-F_2(k)}, \tag{3.11}$$

where $\nu_k > 0$ can be either a fixed small constant or a sequence going to zero as $k \to \infty$. The λ and a are two positive parameters. The $H(k)$ in (3.7) is defined as

$$H(k) = E_k(f(Y)Y^\top)W(k) \stackrel{\text{def}}{=} \frac{1}{L}\left(\sum_{i=kL+1}^{(k+1)L} f(Y^k(i))Y^k(i)^\top\right) W(k), \tag{3.12}$$

where $Y^k(i) = W(k)X(i)$ is the estimated source vector. The E_k is the sampling average over the kth frame, and each frame contains L samples of $X(i)$. More generally, one can choose different λ and a in F_1 and F_2 and have overlaps between successive frames. Among the five equations (3.7)–(3.11), equation (3.7) is of the form (1.3) with G in (2.7). The other four equations are there to prevent W from blowing up. The idea is that when $|W(k+1)|$ is large, $F_1(k)$, $F_2(k)$ will be positive thanks to (3.8)–(3.9); then $1+\nu_{k+1}\sigma_1(k+1)$ and $\sigma_2(k+1)$ will decrease to prevent $|W(k+2)|$ from growing further. Notice that *there is no division* involved in applying and updating scaling factors σ_1 and σ_2.

Because of the averaging over L sample points in (3.12), the H, W, σ_1, and σ_2 vary on the time scale of frames indexed by k which is slower than the sample scale of input mixture signal $X(i)$ indexed by i. The iterative scheme (3.7)–(3.12) hence contains two time scales.

If $(W(k), c(k), d(k))$ converges in k, then formally

$$E[(d_\infty I - f(W_\infty X)(W_\infty X)^\top)W_\infty] = 0,$$

where the subscript ∞ denotes the limits. If W_∞ is invertible and deterministic, then

$$E[(d_\infty I - f(W_\infty X)(W_\infty X)^\top)] = 0.$$

The estimate of iterations and convergence analysis will be performed in the next section.

4. Uniform bounds and convergence.

4.1. Uniform bounds. Let us first prove the following bounds valid for any positive λ, a, and ν_k.

LEMMA 4.1. *Consider* (3.7)–(3.11) *with* $1+\nu_0\sigma_1(0) > 0$ *and* $\sigma_2(0) > 0$. *For any* $k \geq 1$,

$$|W(k)| \leq e^{a-1} + \nu_k e^{a-1}, \tag{4.1}$$

$$0 < 1 + \nu_k \sigma_1(k) \leq \frac{1}{\lambda} e^{a-1}, \tag{4.2}$$

$$0 < \sigma_2(k) \leq \frac{1}{\lambda} e^{a-1}. \tag{4.3}$$

If $\nu_k \in (0, \nu^*)$ *for some* $\nu^* > 0$, *the iterates are uniformly bounded in* k.

Proof. Inequalities (4.2) and (4.3) follow from (3.10)–(3.11) and the fact that $\max_{x\geq 0} xe^{-x} = 1/e$. Substituting (3.10) and (3.11) in (3.7), applying (3.8)–(3.9) and the fact $\max_{x\geq 0} xe^{-x} = 1/e$ again, we obtain

$$\begin{aligned}|W(k+1)| &\leq |(1+\nu_{k-1}\sigma_1(k-1))W(k)e^{-F_1(k-1)} \\ &\qquad + \nu_k \sigma_2(k-1)H(k)e^{-F_2(k-1)}| \\ &\leq e^{a-1} + \nu_k e^{a-1}.\end{aligned}$$

Here the first terms in F_1 and F_2 give the upper bound $1/e$, the third terms lead to e^a, and the second terms are positive and give upper bound 1. The proof is complete. ∎

REMARK 4.1. *By scaling (normalization), the received signal* $\{X(t)\}$ *is uniformly bounded by one in absolute value. It follows from* (3.12) *that* $|H(k)| \leq C$ *for some* $C > 0$ *uniformly in* k. *If* ν_k *is bounded away from zero (for example, if* ν_k *equals a positive constant), the uniform boundedness of* $\sigma_1(k)$ *follows from* (4.2).

If we further choose constant a small enough, then both $1+\nu_k\sigma_1(k)$ and $\sigma_2(k)$ are bounded away from zero. To see this, let us denote by C the upper bound of $W(k)$ and $H(k)$. The constant C depends only on a, ν_0, and the nonlinearity f.

For simplicity, $C = C(a)$. We show that the iteration will not converge to a trivial solution; in particular, $\sigma_2(k)$ is bounded away from zero uniformly in k.

LEMMA 4.2. *Consider* (3.7)–(3.11) *with* $1+\nu_0\sigma_1(0) > 0$ *and* $\sigma_2(0) > 0$. *For any* $\lambda > 0$ *and* $a > 0$, *there is a small enough value* $\underline{\sigma} > 0$ *depending on* λ *and* a *so that*

$$\sigma_2(k) > \underline{\sigma}, \qquad 1+\nu_k\sigma_1(k) > \underline{\sigma} \tag{4.4}$$

for any $k \geq 0$.

Proof. By the uniform boundedness of $|H(k)| \leq C$, we infer from (3.11) that

$$\sigma_2(k+1) \geq \sigma_2(k)e^{-\sigma_2(k)(C+\lambda)+a}.$$

We argue by induction and assume that $\sigma_2(0) \geq \underline{\sigma}$ and that $\sigma_2(k)$ satisfies $\sigma_2(k) \geq \underline{\sigma}$. Then using (4.3) and denoting $\frac{1}{\lambda}e^{a-1}$ by $\overline{\sigma}$, we obtain

$$\sigma_2(k+1) \geq \min_{\underline{\sigma}\leq\sigma\leq\overline{\sigma}} \sigma e^{a-(C+\lambda)\sigma}.$$

Now for given λ and a, and hence fixed $\overline{\sigma}$ and $C = C(a)$, choose $\underline{\sigma}$ small enough so that

$$\min_{\underline{\sigma}\leq\sigma\leq\overline{\sigma}} \sigma e^{a-(C+\lambda)\sigma} = \underline{\sigma}e^{a-(C+\lambda)\underline{\sigma}} \geq \underline{\sigma}.$$

A similar lower bound holds for $1+\nu_k\sigma_1(k)$. ∎

REMARK 4.2. *Though we do not have an explicit lower bound for* $|W(k)|$, *the numerical results of the next section will show that the coefficient* $(1+\nu_k\,\sigma_1(k))$ *is on average above one in* k, *implying that* $|W(k)|$ *does not converge to zero from a positive value with positive probability. Suppose otherwise that* $|W(k)|$ *is small for* $k \geq k_1$; *then the uniform lower bounds* (4.4) *and* (3.12) *imply that* $(1+\nu_k\,\sigma_1(k))W$ *is dominant over the nonlinear term* $\nu_k\,\sigma_2(k)\,H(k)$ *in* (3.7). *Thanks to the choice of natural gradient in* (3.12), *this property holds even if* $f(\cdot)$ *is a sign function, as chosen for the numerical example in the next section. Then* (3.7) *says that to leading order when* $|W(k)|$ *is sufficiently small,* $W(k+1) \approx (1+\nu_k\,\sigma_1(k))\,W(k)$, *where* $1+\nu_k\,\sigma_1(k)$ *is uniformly positive and above one on average in* k. *Hence* $|W(k)|$ *will grow larger instead of continuing to stay small at most values of* $k \geq k_1$.

4.2. Convergence and source separation. By (3.7), we have

$$W(k_0+N) - W(k_0) = \sum_{i=k_0}^{k_0+N-1} \nu_i(\sigma_1(i)W(i) - \sigma_2(i)H(i))$$

for any positive integers k_0 and N. Suppose that for some k_0

$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=k_0}^{k_0+N-1} \nu_i(\sigma_1(i)W(i) - \sigma_2(i)H(i)) \stackrel{\text{def}}{=} W^* - H^*, \tag{4.5}$$

which is an analogue of the law of large numbers for sequences of random variables, or a form of weak convergence of oscillatory sequences. The superscript star denotes the sequential (empirical) average. The oscillatory nature of the W and H sequences will be demonstrated numerically in the next section. Clearly, (4.5) holds along a subsequence of $N \to \infty$.

Under (4.5), it follows from the uniform upper bound of $|W(k)|$ that

$$H^* - W^* = 0, \tag{4.6}$$

which is a sequentially (temporally) averaged version of the desired source separation (natural gradient) condition

$$G(W, X) = E\left[(f(Y)Y^\top - I)W\right] = 0. \tag{4.7}$$

If the solution sequences are stationary and mixing for large k, the sequential (temporal) and ensemble averages are identical (ergodicity), a property well studied for sequences of random variables [11]. It is an interesting problem to study the mixing property of iterative solutions from the mixing property of source signals. Thanks to the uniform bounds, the soft-constrained algorithm is nontrivial at large k and provides a nontrivial source separating solution. This is how our dynamic on-line method (3.7)–(3.11) realizes the off-line batch condition (2.7) or (4.7). A numerical example will illustrate this point later.

REMARK 4.3. *Unlike the classical Robbins–Monro iteration* [14]*, our method does not require the time step (learning parameter)* $\nu_k \to 0$ *as* $k \to \infty$*. In fact, our numerical computation is found to be more efficient for source separation at a constant time step. The classical Robbins–Monro iteration and related stochastic approximation theory have been well studied; see* [14, 15] *and references therein. Instead, we shall study solutions in the limit of decreasing constant step* ν *in the next section, as in* [15].

4.3. Convergence in the limit $\nu_k = \nu \to 0$. Now we consider $\nu_k = \nu$, a fixed constant. For any given ν, the scheme (3.7)–(3.11) yields a sequence of $W(k)$ depending on ν. We shall use the notation W_k^ν in lieu of $W(k)$ in the following discussion. Similar to the analysis of a numerical scheme of ODEs, where one examines the convergence of the scheme as the time step goes to zero, we study the limit when $\nu \to 0$. Define $Z_k^\nu = \sigma_{1,k}^\nu W_k^\nu - \sigma_{2,k}^\nu H_k^\nu$, which is nonlinear in W_k^ν. Then we write (3.7) as

$$W_{k+1}^\nu = W_k^\nu + \nu Z_k^\nu. \tag{4.8}$$

Let us also define

$$W^\nu(t) = W_k^\nu \quad \text{when } k\nu \le t < (k+1)\nu. \tag{4.9}$$

It follows from Lemma 4.1 that W_k^ν are uniformly bounded in k and ν. At a fixed ν, $\sigma_{1,k}^\nu$ is uniformly bounded in k. Now we assume that (A1) $\sigma_{1,k}^\nu$ is uniformly bounded in k and ν. Under the assumption (A1), $|Z_k^\nu| \le M$ uniformly in k and ν in view of Lemma 4.1. Hence $|W^\nu(t) - W^\nu(s)| \le M|t - s|$. This implies that $W^\nu(t)$ is equicontinuous in the extended sense (see p. 102, section 4.2 of [15]). Hence by [15, Theorem 4.2.2], $\{W^\nu(t)\}$ has a subsequence, still denoted by $\{W^\nu(t)\}$, that converges to a continuous function $W(t)$, uniformly on each bounded interval of t. Further, we assume that (A2) there is a continuous function $\bar{g}$ of W such that

$$\lim_{\nu \to 0} \nu \sum_{k=0}^{t/\nu - 1} Z_k^\nu(W^\nu(s); s \le k\nu) = \bar{g}(W). \tag{4.10}$$

As Z_k^ν is uniformly bounded in (k, ν) and uniformly continuous in W^ν, (4.10) holds up to a subsequence in ν for a bounded input signal $X(i)$. The summand on the left-hand side of (4.10) is a nonlinear nonlocal function of $X(i)$ if σ_1, σ_2, and W are fixed. If in addition $L = 1$, then the sum reduces to the classical form $\nu \sum_{k=0}^{t/\nu-1} h(X(k))$, for an integrable function h, and the convergence (4.10) holds almost surely for a stationary process $X(i)$, known as Birkhoff's ergodic theorem (see Chapter 6 of [11]). It is not hard to show that the convergence still holds for $L \neq 0$. The complication arises when σ_2 depends nonlocally on $X(i)$ via H. Extending Birkhoff's theorem to (4.10) will be left for a future study.

Now let us define

$$Z^\nu(t) = \nu \sum_{k=0}^{t/\nu-1} Z_k^\nu, \tag{4.11}$$

which puts the algorithm in the form

$$W^\nu(t) = W_0^\nu + Z^\nu(t).$$

It follows that

$$W^\nu(t) - W_0^\nu - G^\nu(t) = \nu \sum_{k=0}^{t/\nu-1} \left(Z_k^\nu(W^\nu(s); s \le k\nu) - \bar{g}(W_k^\nu)\right), \tag{4.12}$$

where $G^\nu(t) = \nu \sum_{k=0}^{t/\nu-1} \bar{g}(W_k^\nu)$. Letting $\nu \to 0$ in (4.12), we have $G^\nu(t) \to \int_0^t \bar{g}(W(\tau))\, d\tau$. By assumption (4.10), we find that

$$W(t) - W(0) - \int_0^t \bar{g}(W(\tau))\, d\tau \ = 0, \tag{4.13}$$

the averaged (upscaled) equation on the dynamics of W. We have proved the following.

THEOREM 4.1. *If $\sigma_{1,k}^\nu$ are uniformly bounded in k and ν, then $W^\nu(t)$ has a convergent subsequence that converges to a continuous function $W(t)$. Assume also that there is a continuous $\bar{g}$ such that* (4.10) *is true. Then $W(t)$ satisfies the time averaged ODE $\dot{W} = \bar{g}(W)$.*

5. Numerical examples. We first present numerical results for the 2×2 (2 sources and 2 receivers) instantaneous mixture case and then show the 8×8 case. The mixing matrices are generated randomly by MATLAB functions.

In all the calculations, we take $\nu_k = 0.01$, $\lambda = 0.01$, and $a = 1$. We use the sign function for each component of f with $f = (f_1, f_2)$ or $f = (f_1, \ldots, f_8)$, i.e.,

$$f_i(u) = \frac{u}{|u| + \epsilon} \qquad \text{for } i = 1, 2, \ldots, 8$$

with $\epsilon = 10^{-16}$. The size of each frame is 10 sample points with no overlapping between successive frames.

In the 2×2 case, the inverse of the mixing matrix is

$$\begin{bmatrix} -1.4153 & 0.62268 \\ -1.1648 & 1.1284 \end{bmatrix}. \tag{5.1}$$

The last value of $W(k)$ in the iteration is

$$\begin{bmatrix} 0.45502 & -0.20275 \\ -0.53368 & 0.51716 \end{bmatrix}. \tag{5.2}$$

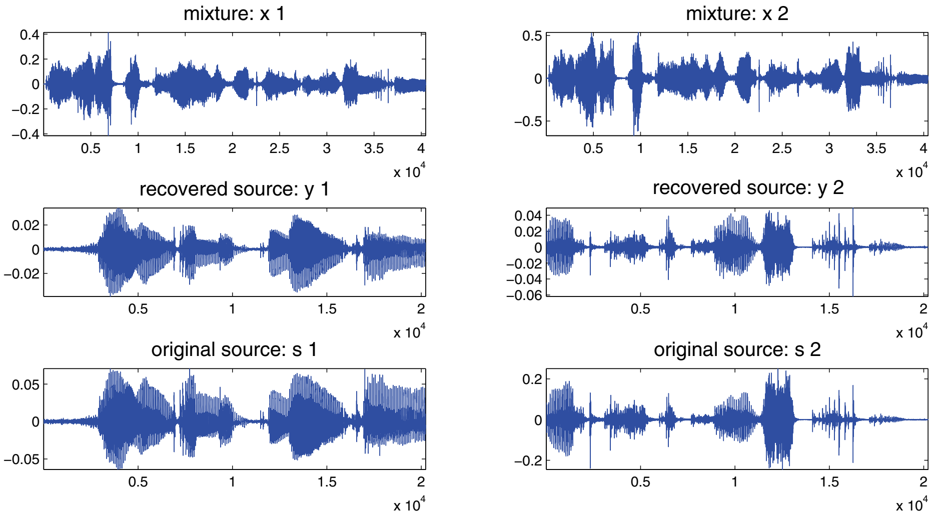

FIG. 1. *The* 2 *sources and* 2 *receivers case. From top to bottom: mixtures, recovered source signals, and original source signals. Only the* 2nd *half in time is shown for each source signal.*

We see that (5.2) and (5.1) are not equal. This is due to the scaling and permutation ambiguities in BSS problems. In fact, if we divide each row by the number in that row having the maximum absolute value, the above two matrices become

$$\begin{bmatrix} 1.0 & -0.440 \\ 1.0 & -0.969 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1.0 & -0.446 \\ 1.0 & -0.969 \end{bmatrix},$$

respectively, and we see that they are really close. In general, there can also be a permutation of rows. The plots of the mixtures, recovered source signals, and original source signals are shown in Figure 1, which indicates a quite good recovery.

For the 8×8 case, the mixing matrix A_0 is a Gaussian random matrix generated with MATLAB command `randn(8,8)`, and its inverse is (after scaling)

(5.3)
$$\begin{bmatrix} 0.362 & 1.0 & -0.704 & -0.528 & 0.197 & 0.0541 & -0.121 & -0.524 \\ 1.0 & 0.957 & -0.247 & 0.170 & -0.177 & 0.905 & 0.610 & -0.111 \\ 0.231 & 0.623 & -0.357 & -0.0990 & -0.0931 & -0.0611 & 1.0 & 0.312 \\ -0.301 & 1.0 & 0.647 & -0.676 & -0.590 & 0.139 & 0.671 & -0.221 \\ -0.110 & 0.446 & 1.0 & 0.681 & 0.283 & -0.0486 & 0.639 & 0.0912 \\ 0.145 & -0.813 & 0.351 & -0.761 & 1.0 & 0.559 & -0.532 & 0.0775 \\ -0.607 & 0.319 & 0.0157 & -0.429 & 1.0 & 0.523 & 0.771 & 0.0778 \\ -0.547 & -0.623 & 0.245 & 0.600 & 0.138 & -0.121 & -0.957 & 1.0 \end{bmatrix}.$$

In order to compare (5.3) with the demixing matrix we will obtain later, we have already divided each row by the largest absolute value in that row.

The mixture is shown in Figure 2, which contains seven speech signals and one music signal. The length of the data is 40459 and the sampling rate is 16000 Hz. The signals last about 2.53 seconds. The total CPU time for processing the whole data is

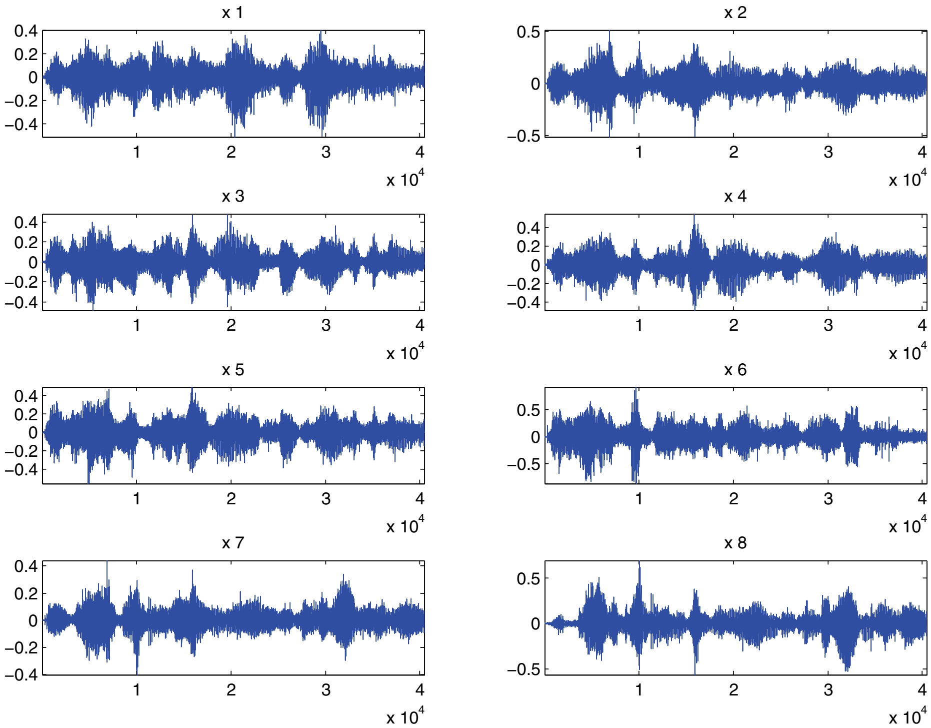

FIG. 2. *Mixture signals.*

2.06 seconds. The computation is done in MATLAB on a Compaq laptop with AMD 1.6GHz CPU.

The second half of the recovered source signals are shown in Figure 3, which may be compared with the source signals shown in Figure 4, subject to a permutation of the ordering of the sources. Because the separation is not achieved immediately, we have only shown the second half of the recovered source signals and the input source signals. Perceptually, a fairly good separation begins to emerge after 1/4 of the entire data has been processed.

The values of F_1, F_2 and $1 + \nu_k\sigma_1(k)$, $\sigma_2(k)$, $\|W(k)\|_1$ are shown in Figure 5. The bounded oscillatory nature of these variables is apparent. The minimum value of $\sigma_2(k)$ is 0.01.

The last value of $W(k)$ in the iteration is as follows. Again, we divide each row by the entry with the largest absolute value in the row:

(5.4)
$$\begin{bmatrix} -0.606 & 0.319 & 0.0172 & -0.429 & 1.0 & 0.523 & 0.772 & 0.0781 \\ 0.343 & 1.0 & -0.699 & -0.553 & 0.204 & 0.0627 & -0.157 & -0.542 \\ -0.106 & 0.478 & 1.0 & 0.694 & 0.263 & -0.0574 & 0.664 & 0.0886 \\ -0.295 & 1.0 & 0.634 & -0.794 & -0.492 & 0.184 & 0.525 & -0.306 \\ 0.142 & -0.785 & 0.353 & -0.762 & 1.0 & 0.562 & -0.530 & 0.0646 \\ 1.0 & 0.967 & -0.253 & 0.164 & -0.166 & 0.917 & 0.628 & -0.109 \\ 0.231 & 0.622 & -0.381 & -0.0776 & -0.113 & -0.0698 & 1.0 & 0.317 \\ 0.539 & 0.657 & -0.233 & -0.576 & -0.129 & 0.125 & 1.0 & -0.946 \end{bmatrix}.$$

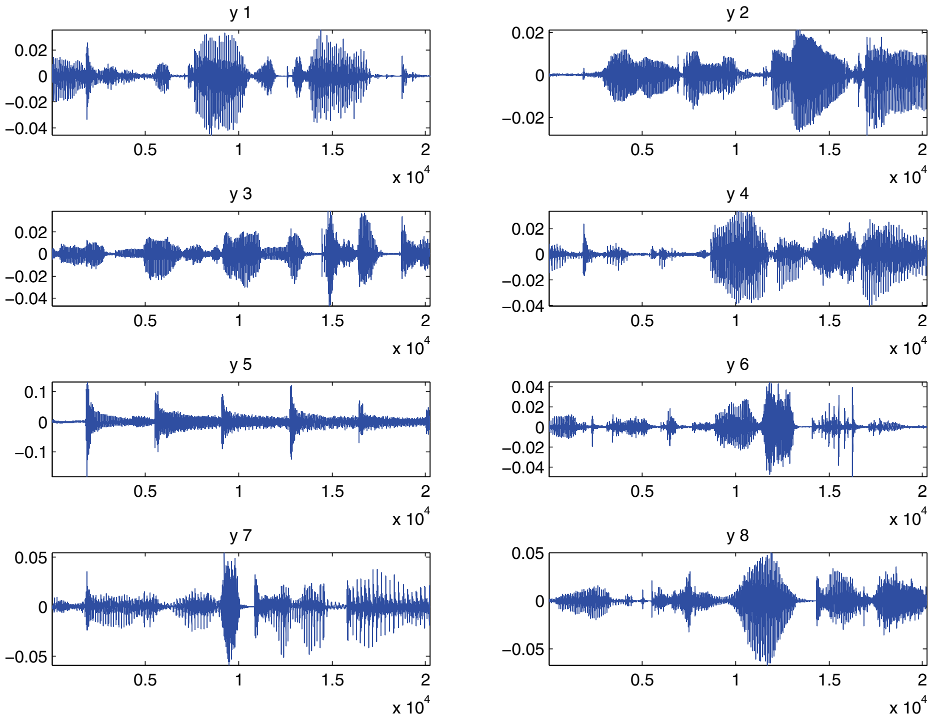

FIG. 3. *Recovered source signals. Only the 2nd half in time is shown.*

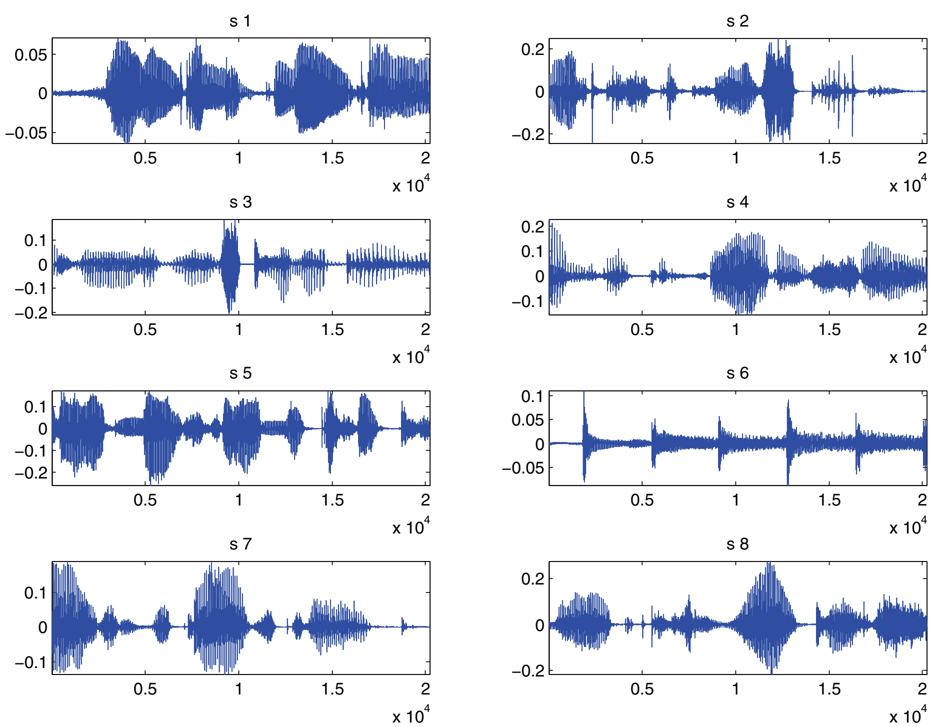

FIG. 4. *The original source signals. Only the 2nd half in time is shown. This should be compared with Figure 3 up to a permutation of the order of the sources.*

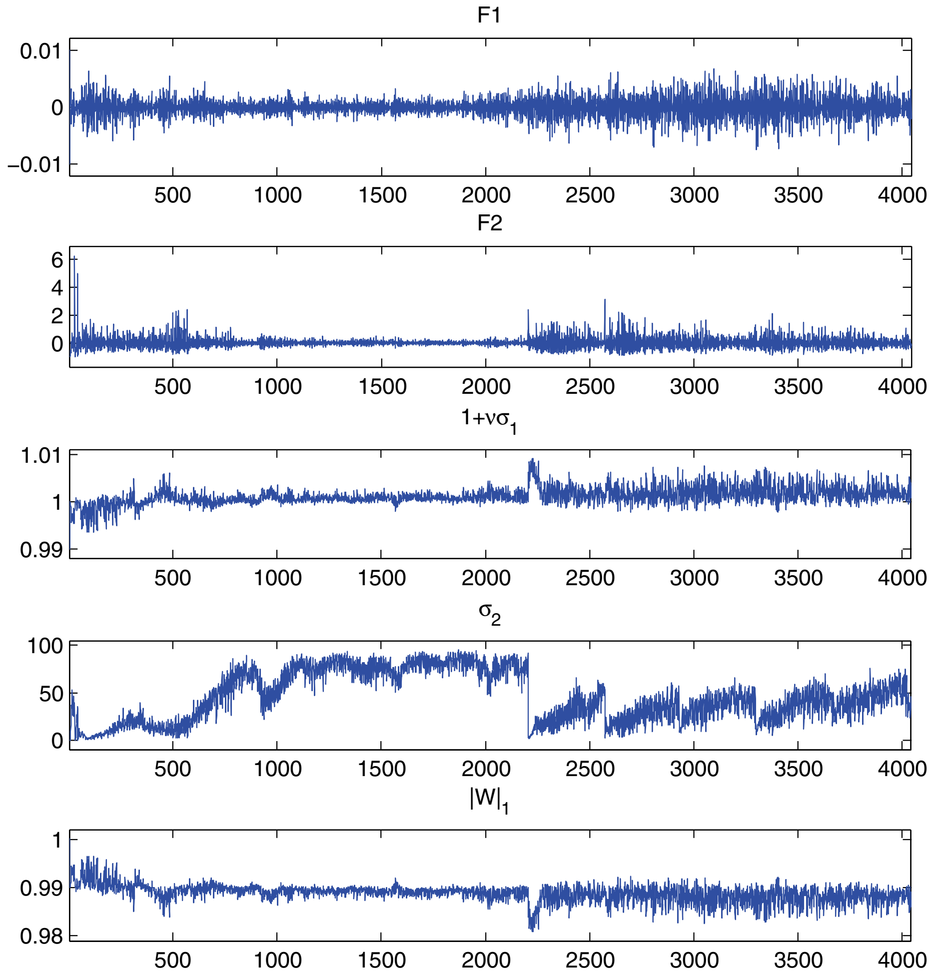

FIG. 5. *Evolution in* k *of* $F_1(k)$, $F_2(k)$, $1+\nu_k\sigma_1(k)$, $\sigma_2(k)$, *and* $\|W(k)\|_1$.

Up to a permutation, (5.4) and (5.3) are roughly the same. To be more precise, if we call the matrix (5.3) by W^* and call the matrix (5.4) by $\hat{W}$, then $W^* \approx \hat{W}([2,6,7,4,3,5,1,8],:)$ after we flip the sign of the last row of $\hat{W}$.

In Figure 5, the mean value of $1+\nu\,\sigma_1(k)$ over $k \in [1,4000]$ is 1.0012. At 81% of the $k \in [1,4000]$, $1+\nu\,\sigma_1(k)$ exceeds one. By the analysis in Remark 4.2, with a chance above 81%, $|W(k)|$ will not go to zero. Indeed, the last frame of Figure 5 shows that $|W(k)|$ is nontrivial.

Finally, for comparison, we implemented the algorithm of [10], or (3.1)–(3.4) to separate the same 8×8 mixture above. The μ is chosen to be 0.1, as suggested in [10], and the initial value of W is $0.1I$. The separated sources are shown in Figure 6. One can see and hear that the quality of the recovered sources is not as good as in ours, even though separation is there. The reason is that their W has a much larger temporal variation than ours, as seen from Figure 7. The last value of $W(k)$ in the iteration

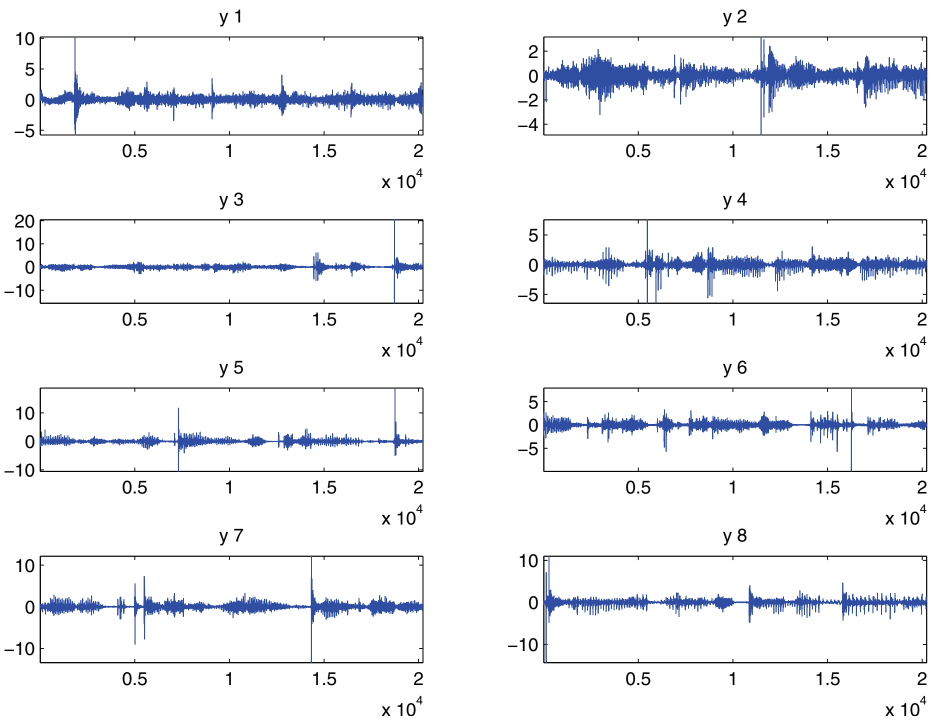

Fig. 6. *Recovered source signals using the algorithm of* [10] ((3.3), (3.4)). *Only the 2nd half in time is shown. Notice the occurrence of large peaks.*

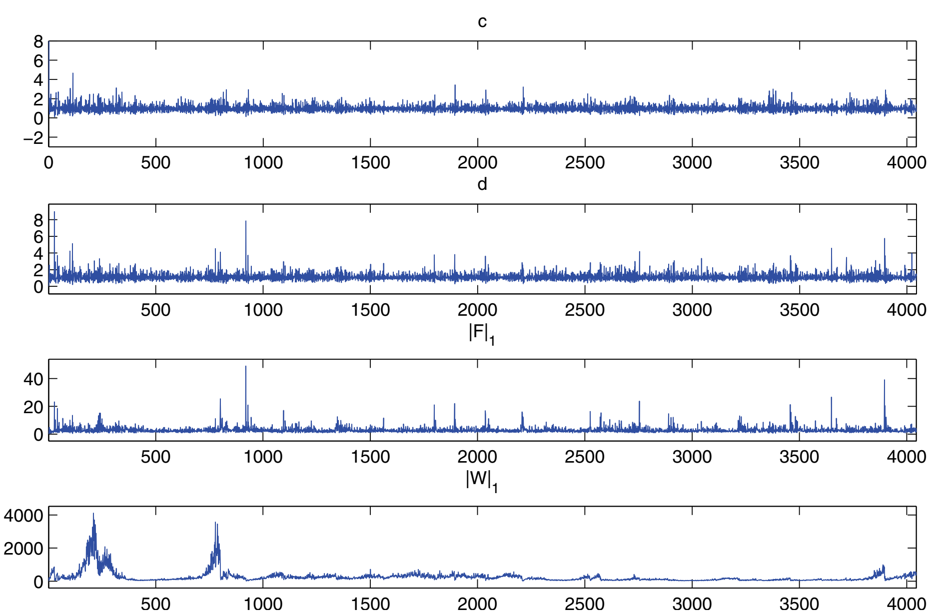

Fig. 7. *Evolution in k of c, d, F, and W of the algorithm of* [10] ((3.3), (3.4)). *Notice the occurrence of large peaks.*

is as follows, where we have divided each row by the entry with the largest absolute value in the row:

(5.5)
$$\begin{bmatrix} -0.667 & 0.979 & -0.491 & 0.665 & -0.692 & -0.561 & 1.0 & -0.00310 \\ -0.716 & -0.127 & 0.365 & -0.0311 & 0.853 & 0.524 & 1.0 & 0.399 \\ 0.0550 & 0.795 & 0.463 & 0.276 & 0.474 & 0.503 & 1.0 & 0.0810 \\ -0.837 & 0.754 & 0.314 & -0.808 & 0.999 & 0.681 & 1.0 & -0.00670 \\ -0.629 & 0.295 & 0.0227 & -0.431 & 1.0 & 0.499 & 0.754 & 0.0790 \\ 1.0 & 0.930 & -0.250 & 0.176 & -0.204 & 0.870 & 0.580 & -0.110 \\ -0.749 & 0.126 & 0.0805 & -0.244 & 1.0 & 0.440 & 0.536 & 0.322 \\ 0.445 & 0.632 & -0.456 & 0.0970 & -0.452 & -0.222 & 1.0 & 0.362 \end{bmatrix}.$$

Even up to permutation, it is impossible to make (5.5) close to (5.3). For example, every entry of the third row of (5.5) is positive, yet such a row is absent in (5.3).

6. Conclusions. We introduced a soft-constrained infomax iteration method, proved uniform estimates of its solutions, and related its large time convergence to the statistical separating criteria. In the small learning step limit, an upscaled ODE is derived to describe the large scale dynamic evolution of a demixing matrix. A numerical example of an instantaneous random mixture of eight sound signals demonstrated excellent separation of signals. The soft-constrained solutions do not involve division and are found to be stable and separating nicely. The hard-constrained solutions may encounter small divisors in their iterations during silent durations of speech and may have large fluctuations in time. Future work will study the soft-constrained method for convolutive mixtures and extensions of Birkhoff's ergodic theorem.

Acknowledgments. The authors would like to thank the referees for their constructive comments.

REFERENCES

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

[2] S. Amari and J.-F. Cardoso, *Blind source separation—semiparametric statistical approach*, IEEE Trans. Signal Process., 45 (1997), pp. 2692–2700.

[3] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang, *Multichannel blind deconvolution and equalization using the natural gradient*, in Proceedings of the First Signal Processing Workshop on Signal Processing Advances in Wireless Communications, Paris, 1997, pp. 101–104.

[4] A. Bell and T. Sejnowski, *An information-maximization approach to blind separation and blind deconvolution*, Neural Comput., 7 (1995), pp. 1129–1159.

[5] A. Bijaoui and D. Nuzillard, *Blind source separation of multispectral astronomical images*, in Mining the Sky: Proceedings of the MPA/ESO/MPE Workshop Held at Garching, Germany, July 31–August 4, 2000, A. J. Banday, S. Zaroubi, and M. Bartelmann, eds., Springer-Verlag, Berlin, 2001, p. 571.

[6] G. Brown and D.-L. Wang, *Separation of speech by computational auditory scene analysis*, in Speech Enhancement, J. Benesty, S. Makino, and J. Chen, eds., Springer, New York, 2005, pp. 371–402.

[7] J.-F. Cardoso and A. Souloumiac, *Blind beamforming for non-Gaussian signals*, IEE Proc. F: Radar Signal Process., 140 (1993), pp. 362–370.

[8] S. Choi, A. Cichocki, H. Park, and S. Lee, *Blind source separation and independent component analysis: A review*, Neural Inform. Process. Lett. Rev., 6 (2005), pp. 1–57.

[9] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*, John Wiley and Sons, New York, 2005.
[10] S. C. Douglas and M. Gupta, *Scaled natural gradient algorithms for instantaneous and convolutive blind source separation*, in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vol. 2, Honolulu, HI, 2007, pp. II-637–II-640.
[11] R. Durrett, *Probability: Theory and Examples*, 2nd ed., Duxbury Press, Belmont, CA, 1996.
[12] S. Gazor and W. Zhang, *Speech probability distributions*, IEEE Signal Process. Lett., 10 (2007), pp. 204–207.
[13] J. Kolba and I. Jouny, *Blind source separation in tumor detection in mammograms*, in Proceedings of the IEEE 32nd Annual Northeast Bioengineering Conference, Easton, PA, 2006, pp. 65–66.
[14] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Springer-Verlag, New York, 1978.
[15] H. J. Kushner and G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed., Appl. Math. 35, Springer-Verlag, New York, 2003.
[16] J. Liu, J. Xin, and Y. Qi, *A dynamic algorithm for blind separation of convolutive sound mixtures*, Neurocomputing, 72 (2008), pp. 521–532.
[17] N. Murata, S. Ikeda, and A. Ziehe, *An approach to blind separation based on temporal structure of speech signals*, Neurocomputing, 41 (2001), pp. 1–24.
[18] M. Naceur, M. Loghmari, and M. Boussema, *The contribution of the sources separation method in the decomposition of mixed pixels*, IEEE Trans. Geosci. Remote Sensing, 42 (2004), pp. 2642–2653.