Blind Separation of a Mixture of Uniformly Distributed Source Signals; A Novel Approach

Jayanta Basak*
Shun-ichi Amari
Laboratory for Information Synthesis, RIKEN Brain Science Institute, Institute of Physical and Chemical Research (RIKEN), Wako-shi, Saitama 351-01, Japan

A new, efficient algorithm for blind separation of uniformly distributed sources is proposed. The mixing matrix is assumed to be orthogonal by prewhitening the observed signals. The learning rule adaptively estimates the mixing matrix by conceptually rotating a unit hypercube so that all output signal components are contained within or on the hypercube. Under some ideal constraints, it has been theoretically shown that the algorithm is very similar to an ideal $O(1/T^2)$ convergent algorithm, which is much faster than the existing $O(1/T)$ convergent algorithms. The algorithm has been generalized to take care of the noisy signals by adaptively dilating the hypercube in conjunction with its rotation.

1 Introduction

Blind separation (Amari & Cardoso, 1996; Amari, Cichocki, & Yang, 1995, 1996; Yang & Amari, 1997; Amari, Chen, & Cichocki, 1997; Bell & Sejnowski, 1995; Cardoso & Laheld, 1996; Comon, 1994; Jutten & Hérault, 1991; Karhunen & Joutsensalo, 1994; Oja & Karhunen, 1995; Oja, 1995) refers to the task of separating independent signal sources from the sensor outputs in which the signals are mixed in an unknown channel—a multiple-input, multiple-output linear system. This problem arises in many areas, such as speech recognition, data communication, signal processing, and medical science.

There are various approaches for dealing with the task of blind separation. In the independent component analysis (ICA) approach, the signals are transformed in such a way that the dependency between individual signal components is minimized. ICA was proposed by Comon (1994) for this purpose (see also Amari et al., 1995; Yang & Amari, 1997). In the entropy maximization approach (Bell & Sejnowski, 1995), the output components are transformed by a nonlinear transfer function, so that the output distribution is contained within a finite hypercube. The information content of

* The author is on lien from Machine Intelligence Unit, Indian Statistical Institute, Calcutta, India.
the output, as measured by entropy, is maximized, which forces the output components to be as uniformly spread over the hypercube as possible. Karhunen and Joutsensalo (1994), Oja and Karhunen (1995), and Oja (1995) also developed a new technique for blind separation based on nonlinear principal component analysis, which is an extension of the linear principal component analysis (PCA) algorithm (Oja, 1982; see also Amari, 1977). In a completely different approach (Prieto, Puntonet, Prieto, & Rodriguez-Alvarez, 1997), assuming bounded input distributions, source signals were separated based on some geometric properties. However, no theoretical justification as to convergence of the algorithm was provided.

These approaches can be unified from the viewpoint of information geometry of the Kullback-Leibler divergence measure (Amari et al., 1995, 1997; Yang & Amari 1997; Amari, 1998), and statistical efficiency and dynamical stability of algorithms are discussed under the assumption that the probability density functions of the source signals have differentiable form. However, the source probability distributions are sometimes not differentiable, and the Fisher information diverges. In such a case, we can have much more efficient algorithms than those the Cramer-Rao theory (Rao, 1973) gives.

As a typical example, we focus on adaptive separation of uniformly distributed source signals. The uniform distribution is not differentiable at the extrema, and as a result, Fisher information does not exist. Therefore, the problem is nonregular from the statistical viewpoint. It is assumed here that the mixing matrix of the source signals is orthogonal, and the task is to find a suitable orthogonal linear transformation adaptively within a connectionist framework for recovering the random source signals. However, this restriction will be relaxed.

Theoretically we show that there exists a statistical estimator by which the original signals are recovered within squared error of $O(\frac{1}{T^2})$ when $T$ examples are shown. This $O(\frac{1}{T^2})$ convergent estimator is much better than any unbiased estimator having an optimal $O(\frac{1}{T})$ convergence rate (Rao, 1973) where Fisher information exists. We then propose a practical online algorithm that is better than conventional $O(\frac{1}{T})$ convergent algorithms.

The orthogonal uniformly distributed signals are always contained within a hypercube under the noiseless condition. In the proposed algorithm, a learning rule is designed within the connectionist framework (exploiting only local properties) such that a unit hypercube is suitably rotated based on the observed samples in order to contain totally the source signal components. The learning rule is similar to that proposed in the EASI algorithm (Cardoso & Laheld, 1996). However, in the proposed method, a special nonlinear function is designed to take care of the uniform distribution, which results in much faster convergence of the separation algorithm. A similar function may be approached by the nonlinear PCA analysis. The proposed algorithm can adaptively adjust the learning rate and therefore is able to perform blind separation even in the presence of a changing mixing matrix.
2 Separation Under Noiseless Condition

Let there be \( n \) independent signal sources \( s_i(t); \ i = 1, 2, \ldots, n \) which are mixed by an unknown orthogonal mixing matrix \( A \) to give rise another \( n \) signal components \( x_i(t); \ i = 1, 2, \ldots, n \), that is,

\[
x(t) = As(t)
\]

where \( x(t) = [x_1(t), x_2(t), \ldots, x_n(t)]' \) and \( s(t) = [s_1(t), s_2(t), \ldots, s_n(t)]' \), and \( A' A = I \). \( A' \) is the transpose of \( A \). This article treats the case where the probability distribution of \( s(t) \) is independently and identically distributed (i.i.d.) subject to the uniform distribution,

\[
p(s) = \begin{cases} \frac{1}{2^n} & |s_i| \leq 1, \text{ for all } i \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}
\]

The task is to estimate \( A \) only from the given signals \( x(t) \). In other words, an orthogonal linear transformation \( W \) is to be estimated in such a way that

\[
y(t) = Wx(t)
\]

becomes a permutation of \( s(t) \), that is, \( WA = C \), \( C \) being an arbitrary permutation matrix.

Under the assumption of orthogonal transformation \( x = As \) of the uniformly distributed original source signals \( s \), the input signals \( x \) can be thought of as lying within a hypercube; the orientation of the hypercube in the \( N \)-dimensional space is defined by the transformation matrix \( A \). Therefore, the problem of blind source separation in this case boils down to the problem of finding a suitable orientation of the hypercube such that the input signals are totally contained within the hypercube,

\[
|y_i(t)| \leq 1, \quad \text{for all } i \text{ and } t,
\]

where \( t \) is an occurrence of the \( i \)th output signal and \( y = Wx \). In other words, the weight matrix is to be updated in such a way that it causes a rotation of the hypercube in \( N \)-dimensional space, which is empirically given as follows.

At any instant, let \( W \in SO(n) \), where \( SO(n) \) is the set of all \( n \times n \) special orthogonal matrices, such that for any \( B \in SO(n) \), \( B B = B B = I \). The mixing matrix \( A \) is an instance of \( SO(n) \). Then \( y = WAs = Cs \) where \( C \) is an instance of \( SO(n) \). Any \( C \) can be represented as

\[
C = \exp(\eta Z),
\]

where \( \eta \) is a constant and \( Z \) is an antisymmetric matrix with \( \|Z\|_F = 1 \). \( Z \) together with \( \eta \) has \( n(n-1)/2 \) free variables, which are the local coordinates.
of $C$. Similarly, $W + dW$ can be represented as $W + dW = \exp(\eta Z)W$ since the product of two orthogonal matrices is always orthogonal. In other words,

$$dW = (\exp(\eta Z) - I)W.$$  \hfill (2.5)

This can be equivalently written as,

$$dW = \left(\eta Z + \frac{\eta^2}{2}Z^2 + O(\eta^3)\right)W.$$ \hfill (2.6)

A first-order empirical learning rule is

$$dW = \eta ZW,$$ \hfill (2.7)

which is also explicitly used in the EASI algorithm (Cardoso & Laheld, 1996).

### 2.1 Error Measure.

For a given input $x(t)$ at any instant $t$, let the output $y(t)$ lie outside the hypercube. Let there be $p$ components of $y$ for which

$$|y_{ip}| > 1, \quad \forall p, i_p \in [1, n].$$ \hfill (2.8)

Ideally, in the noiseless condition, the hypercube is to be rotated in such a way that the outlier falls just on the closest bounding hyperplane of the unit hypercube. The total error due to the presence of the outlier (see Figure 1) can be expressed as

$$e = \sum_{i; |y_i| > 1} (|y_i| - 1).$$ \hfill (2.9)

In other words, the error is equal to the minimum distance of the outlier from the hypercube. The distance is measured as the distance between the outlier and the projection of the outlier onto the unit hypercube.

The average error over all instances of the output is given as

$$\langle e \rangle = \frac{1}{T} \sum_t \sum_{i; |y_i(t)| > 1} (|y_i(t)| - 1).$$ \hfill (2.10)

The average error $\langle e \rangle$ (see equation 2.10), is the training error that is to be minimized based on the observed samples.

The Kullback-Leibler divergence measure does not exist in the case of uniform distribution. Ideally, the error measure in the case of uniform distribution should decrease the variational distance between the observed probability distribution $p(y; W)$ and the probability distribution of the source signals $p(y; A^{-1})$. The error in terms of variational distance is given as

$$E = D[p(y; W) : p(y; A^{-1})]$$

$$= \int |p(y; W) - p(y; A^{-1})|dy.$$ \hfill (2.11)
Figure 1: Two-dimensional view of the hypercube \((i-j)\) plane in the noiseless condition. The outlier \(y\) has two components outside the hypercube.

Equivalently, the error can also be expressed in terms of the Hellinger distance as

\[
E = \int \left( \sqrt{p(y; W)} - \sqrt{p(y; A^{-1})} \right)^2 dy. \tag{2.12}
\]

In section 3, we show that under certain ideal constraints, the algorithm minimizes the variational distance.

2.2 Formulation of the Learning Rule. The hypercube is to be rotated in such a way that the error \(\langle e \rangle\) is minimized. According to the natural gradient descent algorithm (Amari, 1998), we can write the updating rule of \(W\) in terms of the instantaneous variables as

\[
dW \propto - \frac{\partial e}{\partial W} W. \tag{2.13}
\]

When \(W\) belongs to \(SO(n)\), the natural gradient can be written as

\[
dW \propto - \left[ \frac{\partial e}{\partial W} W \right]_A, \tag{2.14}
\]
where $[X]_A$ denotes the antisymmetric part of matrix $X$ (see Cardoso & Laheld, 1996).

By definition, $\frac{\partial e}{\partial W}$ is the rate of change of $e$ with respect to $W$, that is,

$$\Delta e = e(W + \text{d}W) - e(W) = \frac{\partial e}{\partial W}\text{d}W. \quad (2.15)$$

Evaluating the partial derivative of $e$ (considering $\text{d}W$ has $n^2$ free parameters), we get

$$\frac{\partial e}{\partial W_{ij}} = x_isgn(y_i), \quad \text{for } |y_i| > 1$$
$$= 0 \quad \text{otherwise.} \quad (2.16)$$

Note that the derivative is taken only for the variables for which $e > 0$. The derivative does not exist when $e = 0$; that is, $e$ has only directional derivatives. From equation 2.16, we have

$$\left(\frac{\partial e}{\partial W}\right)_{ij} = y_isgn(y_i) \quad \text{for } |y_i| > 1$$
$$= 0 \quad \text{otherwise.} \quad (2.17)$$

Therefore, the online learning rule at any instant $t$ (i.e., for the $t$th sample) is given as

$$W(t + 1) = W(t) + \eta(t)(\text{y}(t)\text{g}(\text{y}(t)) - \text{g}(\text{y}(t))\text{y}(t))W(t), \quad (2.18)$$

where $\eta(t)$ is the learning rate at the $t$th instant to be derived in the following, and

$$g(y_i) = \begin{cases} sgn(y_i) & \text{if } |y_i| > 1 \\ 0 & \text{otherwise.} \end{cases} \quad (2.19)$$

Note that $z_{ij} = 0$ when $|y_i|, |y_j| > 1$ and $|y_i| = |y_j|$. This indicates the fact that the hypercube, in such a condition, is to be rotated in such a way that the outlier falls on the hyperedge of intersection of the closest ($i$th and $j$th) bounding hyperplanes of the unit hypercube.

### 2.3 Learning Rate

Under noiseless condition, the hypercube is to be rotated in such a way that the outlier falls just on the closest bounding hyperplane of the unit hypercube. The change in the instantaneous value of the output vector $\text{y}$ due to the change in the weight matrix can be expressed as $\text{dy} = \text{dWx}$. From equations 2.7 and 2.3, $\text{dy}$ can be written as

$$\text{dy} = \eta\text{c}, \quad (2.20)$$

where $\text{c}$ is the correction vector and is given as $\text{c} = (\text{yg(y)} - \text{g(y)y})\text{y}$. 

The change in the output vector $y$ due to the change in the weight matrix should be such that the components of $y$ that are greater than unity become just equal to unity. Therefore, the learning rate $\eta$ is to be chosen in such a way that

$$|y_i + \eta c_i| \leq 1$$

for all $i$. In other words, $\eta$ is bounded by

$$\frac{|y_i| - 1}{|c_i|} \leq \eta \leq \frac{|y_i| + 1}{|c_i|}$$

for each $i$.

Under the noiseless condition, since the signal vectors are contained within a unit hypercube, there must be a certain amount of rotation for each outlier observation such that the outlier just touches the closest bounding hyperplane. In other words, there always exists a range for $\eta$ that satisfies the set of inequalities in equation 2.22 for all values of $i$. The minimum value of $\eta$ in this range is given as

$$\eta = \max_{i,|c_i|>0} \left\{ \frac{|y_i| - 1}{|c_i|}, 0 \right\}.$$  \hfill (2.23)

If $y$ is perfectly contained within the hypercube, then $|c_i| = 0$ for all $i$.

Note that the selection of $\eta$ in equation 2.23 is consistent with the set of inequalities only under the noiseless condition. In the noisy condition, there may not exist any valid bound for $\eta$ that satisfies the set of inequalities in equation 2.22 for all values of $i$. This is due to the fact that in the noisy condition, the points may not be totally contained within the hypercube. As a result, an outlier may never touch any bounding hyperplane even after rotation. In such cases, an optimal amount of rotation is to be performed such that the distance of the point from the hypercube is minimized. This is discussed in section 4.

3 Theoretical Analysis of the Rate of Convergence

Let $W_0$ be the desired weight matrix such that $W_0 = A^{-1}$. At any instant let $t$, $W(t)$ be the estimated weight matrix. Let the hypercube spanned by the weight matrix $W$ in the $N$-dimensional output space be denoted by $H(W)$ (see Figure 2). $H(W)$ is defined by the set

$$H(W) = \left\{ y(W) \mid ||(W_0 W^{-1} y(W))_i|| \leq 1, \forall i \in [1, n] \right\},$$  \hfill (3.1)

where $y(W) = Wx = WW_0^{-1}s$. In other words, $H(W)$ is the hypercube containing all possible output vectors in the output space generated by the
weight matrix $W$. Similarly, $H(W_0)$ is the hypercube containing all possible output vectors generated by $W_0$ and is given by

$$H(W_0) = \{ y(W_0) | |(y(W_0))_i| \leq 1, \forall i \in [1, n] \}. \quad (3.2)$$

For a more meaningful representation, let us denote $H(W)$ by $H(C)$ by substituting $C = WW_0^{-1}$ and and $H(W_0)$ by $H(I)$.

Let us write $W$ as

$$W = W_0 + \delta W. \quad (3.3)$$

A deviation of $W$ from the desired value of $W_0$ indicates a rotation of the corresponding hypercube in the $N$-dimensional output space. Since $\delta W$ can be expressed in terms of an $N$-dimensional antisymmetric matrix, $\delta W$ can be expressed as $\frac{n(n-1)}{2}$-dimensional vector. In the vicinity of the true solution, $\delta W$ can be written as

$$\delta W = V\delta W \quad (3.4)$$

where $V$ is an antisymmetric matrix.

**Claim.** The rate of convergence of an online learning algorithm (an adaptive estimator) can be obtained from $\langle (\delta W)^2 \rangle$. If $\langle (\delta W)^2 \rangle \propto \frac{1}{T}$, the algorithm is said to be $O(\frac{1}{T})$ convergent. Here $T$ denotes the number of examples for which the weight matrix $W$ is updated. According to the Cramer-Rao
bound (Rao, 1973), the maximum achievable rate of convergence for any unbiased estimator \((\delta \mathbf{W} = 0)\) is \(O(\frac{1}{T})\) where Fisher information exists. For the uniform distribution, Fisher information diverges, and therefore, the Cramer-Rao bound is not applicable. Here we show that for the uniform distribution, there exists an ideal estimator that is \(O(\frac{1}{T})\) convergent. The proposed algorithm is expected to become analogous to this ideal estimator under certain ideal constraints.

The rate of convergence of the learning algorithm can be obtained from the second-order moment of \(\delta \mathbf{W}\). Equivalently, it can be obtained from \(\sum_{i<j} V_{ij}^2\). The average training error can be computed as (see appendix A)

\[
\langle e \rangle = \frac{5}{12} \| \mathbf{V} \|_F^2 - \frac{1}{4} \sum_{j,k} (V^2)_{jk}.
\]

(3.5)

The average training error has a second-order relationship with \(\delta \mathbf{W}\). The variational distance, as described in section 2.2, is given as (see appendix B)

\[
D = \frac{1}{2} |H(C) - H(I)| = \frac{1}{2^{n-1}} \sum_{i<j} |V_{ij}| + \text{higher-order terms of } V_{ij} \text{s.}
\]

(3.6)

In the vicinity of the true solution, considering \(V_{ij} \text{s} \) to be sufficiently small, we can approximate

\[
D = 2^{n-1} \sum_{i<j} |V_{ij}|.
\]

(3.7)

**Assumption.** Ideally, the hypercube at any instant \(t\) is to be rotated in such a way that it aligns with the true one as close as possible. Since the true hypercube contains all the output vectors, the generated hypercube should be rotated in such a way that it contains all previous instances of outputs:

\[
|y_i(\tau)| \leq 1 \quad \forall i \text{ and } \forall \tau \in [0, t].
\]

(3.8)

Under such an ideal restriction we can have the convergence of \(\sum_{i<j} |V_{ij}|\) to zero.

The learning algorithm minimizes the training error by rotating the hypercube in such a way that an outlier always touches the closest bounding hyperplane. However, it does not guarantee that all previous instances of the signal components are contained within the hypercube, even after its rotation. This strong constraint requires sufficient memory to store all the previous points or at least the points near all the boundary hyperplanes.
that are viable to shoot out after rotating the hypercube. The online algorithm considers only \( \tau = t \), and does not consider the previous instances, although it is expected that the hypercube will be rotated in a way that most of the points will be within it. Here we consider a hypothetical estimator where the constraint in equation 3.8 is satisfied, and analyze its rate of convergence. The learning algorithm as described in section 2 is identical to the hypothetical algorithm for \( n = 2 \). This estimator might not be realized by an online learning algorithm.

**Theorem 1.** The convergence rate of the hypothetical estimator is of \( O(\frac{1}{T^2}) \).

**Proof.** Let \( h(W) \) denote the volume of intersection of the hypercubes spanned by present \( W \) and the desired \( W_0 = A^{-1} \). Then

\[
h(W) = \left\{ y \mid |y_i| \leq 1 \land |(W_0W^{-1}y)_i| \leq 1, \forall i \in [1, n] \right\}
\]

\[
= \left\{ y \mid |y_i| \leq 1 \land |(C^{-1}y)_i| \leq 1, \forall i \in [1, n] \right\}.
\]

Since for each \( i, s_i \in [-1, 1] \) and \( W, A \in SO(n) \) (section 2.1),

\[
h(W) = 2^n \quad \text{when } W = W_0. \quad (3.10)
\]

When \( W \) is not equal to \( \delta W \), we have

\[
\delta W = W_0 - W = VW.
\]

We can write

\[
h(W) = 2^n - \frac{1}{2} |H(W) - H(W_0)|
\]

\[
= 2^n - \frac{1}{2} |H(C) - H(I)|. \quad (3.12)
\]

From equation 3.6, we can write

\[
h(W) = 2^n \left( 1 - \frac{1}{2} \sum_{i < j} |V_{ij} + \text{higher-order terms of } V_{ij}s \right). \quad (3.13)
\]

Considering \( W \) to be in the vicinity of the true solution, we can ignore the higher-order terms and therefore

\[
h(W) = 2^n \left( 1 - \frac{1}{2} \sum_{i < j} |V_{ij}| \right). \quad (3.14)
\]
Note that \( h(W) \) is not differentiable at \( W = W_0 \), but the directional derivative of \( h(W) \) exists. Therefore we can always express \( h(W) \) as in equation 3.14 near \( W_0 \).

\[
p(C^{-1}y) = \begin{cases} \frac{1}{2^n} & \text{for } |(C^{-1}y)_i| \leq 1, \forall i \\ 0 & \text{otherwise.} \end{cases} \tag{3.15}
\]

Therefore we can write,

\[
\text{Prob}[|y_i| \leq 1 \forall i \in [1, n]] = \frac{h(W)}{2^n}. \tag{3.16}
\]

After presentation of \( T \) samples, let the weight matrix be \( W \). Then it can be argued that no instances of signal vectors of these \( T \) samples fall outside \( h(W) \). This is due to the fact that we considered the ideal condition where the hypercube is rotated in such a way that the outlier touches the hypercube and all the previous instances are contained within or on the hypercube. Therefore, at least one instance of signal vector will be contained on the boundary of the hypercube—in the strip spanned between \( h(W) \) and \( h(W + dW) \). Therefore,

\[
\text{Prob}\{W(t) \in W \sim W + dW| t = T\} = q(W)\text{Prob}[|y_i| \leq 1, i \in [1, n];
\]

\[t = 0, 1, \ldots, T\}dW \tag{3.17}
\]

where \( q(W) \) is any admissible estimate for \( W(t) \). Let \( p_w(W) \) be the probability distribution of \( W \) obtained by the learning algorithm. Then from equations 3.16 and 3.17, we can write

\[
p_w(W) = q(W)\left(\frac{h(W)}{2^n}\right)^T. \tag{3.18}
\]

From equation 3.18, we can write

\[
p_w(W) = K_m(1 - \frac{1}{2} \sum_{i<j} |V_{ij}|)^T, \tag{3.19}
\]

where \( K_m \) is the normalizing constant for \( M \)-dimensional vector \( \delta W \). Note that \( \delta W \) having \( m = n(n-1)/2 \) free parameters, can be represented by an \( M \)-dimensional vector. Here we consider the deviation of \( W \) from \( W_0 \) to be small and therefore restore only the constant part in the expansion of \( q(W) \). For a sufficiently large value of \( T \), equation 3.19 can be written as

\[
p_w(W) = K_m \exp\left(-\frac{T}{2} \sum_{i<j} |V_{ij}|\right). \tag{3.20}
\]
In order to evaluate $K_m$, we consider

$$\int p_W(W)dW = 1. \quad (3.21)$$

Let us represent $[V_{ij}]$ by an $M$-dimensional vector $[v_1, v_2, \ldots, v_m]$ for the sake of simplicity in representation. From equation 3.20,

$$1 = K_m \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \exp\left(-\frac{T}{2}(|v_1| + |v_2| + \cdots + |v_m|)\right)dv_1dv_2\cdots dv_m$$

$$= K_m \left(\frac{T}{4}\right)^m. \quad (3.22)$$

Therefore,

$$K_m = (T/4)^m. \quad (3.23)$$

The rate of convergence of the idealized learning algorithm can be obtained from the second-order moment of $\delta W$. Equivalently, the order of convergence can be derived from $E(\sum_{i<j}|V_{ij}|^2)$. In simplified notation,

$$\left\langle \left(\sum_{i<j} |V_{ij}| \right)^2 \right\rangle = K_m \int_{-\infty}^{\infty} \cdots , \ldots,$n

$$\int_{-\infty}^{\infty} \left(\sum_{i} |v_i| \right)^2 \exp\left(-\frac{T}{2} \sum_{i} |v_i| \right) dv_1dv_2, \ldots, dv_m. \quad (3.24)$$

In order to evaluate the exact rate of convergence, let us denote the left-hand side of equation 3.24 by $I_m$. Then

$$I_m = K_m \int_{-\infty}^{\infty} \cdots \left(\int_{-\infty}^{\infty} \left[|v_m|^2 + 2|v_m| \left(\sum_{1}^{m-1} |v_{m-1}| \right) + \left(\sum_{1}^{m-1} |v_{m-1}| \right)^2 \right] \right)$$

$$\exp\left(-\frac{T}{2} |v_m| \right) dv_m$$

$$\exp\left(-\frac{T}{2} \sum_{0}^{m-1} |v_i| \right) dv_1 \cdots dv_{m-1}$$

$$= \frac{K_m}{K_{m-1}} \left[\frac{4I_{m-1}}{T} + \frac{32}{T^3} \right]$$

$$= I_{m-1} + \frac{8}{T^2}, \quad (3.25)$$

where $I_{m-1}$ is the integral for $m-1$ dimension. Evaluating, $I_m$ we get

$$I_m = \frac{8m}{T^2}. \quad (3.26)$$
Therefore,
\[
\left( \sum_{i<j} |V_{ij}| \right)^2 = \frac{8m}{T^2}.
\] (3.27)

For any fixed \( n \) we have a fixed value of \( m \), and therefore the algorithm has a convergence rate of \( O\left(\frac{1}{T} \right) \).

Note that the rate of convergence increases linearly with the dimension. The Kullback-Leibler divergence-based method as given by the following equation,
\[
\Delta W = \eta \left( I - \phi(y) y^T \right) W,
\] (3.28)
has a convergence rate of \( O\left(\frac{1}{T} \right) \).

4 Separation in the Noisy Environment

In the presence of noise, a mixed signal vector can be represented as
\[
x = As + n,
\] (4.1)
where \( n \) is the noise vector. Let us assume \( n \) is generated from i.i.d. gaussian distribution. Since \( A \in SO(n) \), each individual signal component can be written as
\[
s_i(t) = \tilde{s}_i(t) + \lambda N(t),
\] (4.2)
where \( \tilde{s}_i \) is the \( i \)th component of the true signal, \( N(t) \) is \( \mathcal{N}(0, 1) \) distributed gaussian noise, and \( \lambda \) is the noise amplitude. Therefore,
\[
p_i(s_i) = \frac{1}{4} \left[ \text{erf} \left( \frac{s_i + 1}{\sqrt{2\lambda}} \right) + \text{erf} \left( \frac{1 - s_i}{\sqrt{2\lambda}} \right) \right].
\] (4.3)

Differentiating equation 4.3, we get
\[
\hat{p}_i(s_i) = \frac{1}{2\sqrt{\pi}} \left[ \exp \left( -\frac{(s_i + 1)^2}{2\lambda^2} \right) - \exp \left( -\frac{(1 - s_i)^2}{2\lambda^2} \right) \right],
\] (4.4)
that is, \( \hat{p}_i(s_i) = 0 \), for \( s_i = 0 \). For small value of \( \lambda \), \( |\hat{p}_i(s_i)| \) takes a large value near the extrema of the uniform distribution, and it is very small at other values of \( s_i \). In other words, Fisher information is mostly concentrated around the extrema of the uniform distribution. Therefore, the performance of a
separation algorithm, in the presence of small amount of noise, is mostly
dependent on the distribution of the signal components around extrema,
that is, the boundary of the ideal hypercube.

In order to formulate an adaptive separation algorithm for the noisy
condition, we consider only the points near the boundary of the hypercube,
and the probability distribution is approximated as

\[
p_i(s_i) = \begin{cases} 
  K & \text{for } |s_i| \leq 1 \\
  K \exp \left( -\frac{(|s_i| - 1)^2}{2\epsilon^2} \right) & \text{for } |s_i| > 1 
\end{cases},
\]  

(4.5)

where \( \epsilon \) is a parameter dependent on the noise amplitude and \( K \) is the
normalizing constant. Therefore, the joint density function of the source
signals can be written as

\[
p(s) = K^n \exp \left( -\frac{D(s)}{2\epsilon^2} \right),
\]  

(4.6)

where

\[
D(s) = \sum_{i=1}^{n} (\sigma(s_i))^2
\]  

(4.7)

\( \sigma(s_i) \) is given as

\[
\sigma(s_i) = \begin{cases} 
  |s_i| - 1 & \text{if } |s_i| > 1 \\
  0 & \text{otherwise.}
\end{cases}
\]  

(4.8)

\( D(s) \) represents the amount of deviation of the signals from their true dis-
tribution due to the presence of noise. The expected amount of deviation
from the true density function can be computed as

\[
\langle D(s) \rangle = \epsilon^2.
\]  

(4.9)

The hypercube in the noiseless condition is thus transformed to a dilated
hypercube with a deviation depending on the noise amplitude. The dilated
hypercube is to be suitably rotated to accommodate the signal vectors.

At any instance, let \( y \) be an outlier and \( \tilde{y} \) be the projection of the outlier
onto the dilated hypercube (see Figure 3). Then it can be shown that for any
\( i \), if \( |y_i| > 1 \), then

\[
|y_i| - |\tilde{y}_i| = (|y_i| - 1) \left( 1 - \frac{\epsilon}{\sqrt{D(y)}} \right),
\]  

(4.10)
where $\epsilon$ is a parameter related to $\epsilon$, which can be user specified or can be determined adaptively. Note that the point $y$ is an outlier only when $\sqrt{D(y)} > \epsilon$. Therefore, the error to be minimized in the noisy condition is

$$
\langle \epsilon \rangle = \frac{1}{T} \sum_i \sum_{|y_i| > 1} \left( |y_i| - 1 \right) \left( 1 - \frac{\epsilon}{\sqrt{D(y)}} \right).
$$

(4.11)

Applying the natural gradient descent learning rule to the instantaneous values of the variables (see section 2), and restoring only the antisymmetric part of the matrix $-\frac{\partial}{\partial W} W^\prime$, we get the learning rule

$$
W(t+1) = W(t) + \eta(t) \left( y(t) g(y(t))' - g(y(t)) y(t)' \right) W(t),
$$

(4.12)

where

$$
g(y_i) = \begin{cases} 
\left( 1 - \frac{\epsilon}{\sqrt{D(y)}} \right) \text{sgn}(y_i) & \text{if } |y_i| > 1 \land \sqrt{D(y)} > \epsilon \\
\frac{\epsilon (\delta - \text{sgn}(y_i))}{\delta} \sum_k |y_k| > 1 \left( |y_k| - 1 \right) & \text{otherwise},
\end{cases}
$$

(4.13)

Note that the nonlinear function $g(y)$ in the noisy environment (see equation 4.13) reduces to the same rule in equation 2.19, under the noiseless condition if we choose $\epsilon$ to be 0.

In the noisy case, an outlier may never touch the dilated hypercube for a given $\epsilon$. Therefore, $\eta$ at each instant is selected in such a way that the difference between the instantaneous change and the desired change in $y$ is minimized. The desired change $\Delta y$ is defined as

$$
\Delta y_i = \begin{cases} 
- \left( 1 - \frac{\epsilon}{\sqrt{D(y)}} \right) \left( |y_i| - 1 \right) \text{sgn}(y_i) & \text{if } |y_i| > 1 \land \sqrt{D(y)} > \epsilon \\
0 & \text{otherwise},
\end{cases}
$$

(4.14)

$\Delta y$ provides the componentwise error that occurred due to the presence of the outlier. $\eta$ is chosen to minimize $\|\Delta y - dy\|^2$ where $dy = \eta c$ is the instantaneous change in output. $c = (yg(y)' - g(y)y')y$ is the correction vector. Solving for the minima we get,

$$
\eta = \frac{\Delta y \cdot c}{\|c\|^2}.
$$

(4.15)

In other words, $\eta$ is the normalized dot product (i.e., the angle) between the correction vector and the vector representing the desired change in the output.
The parameter $\varepsilon$ can be determined by minimizing the average error $\langle e \rangle$ with respect to $\varepsilon$. In other words,

$$\Delta \varepsilon \propto -\frac{\partial \langle e \rangle}{\partial \varepsilon}$$  \hspace{1cm} (4.16)

that is,

$$\Delta \varepsilon \propto \frac{1}{T} \sum_{i} \sum_{i, |y_i| > 1} \sum_{D(y) > \varepsilon} \frac{(|y_i| - 1)}{\sqrt{D(y)}}.$$  \hspace{1cm} (4.17)

Since the learning rule of weight matrix $W$ is dependent on the parameter $\varepsilon$, $\varepsilon$ cannot be simultaneously determined with $W$. In order to perform this task, $\varepsilon$ is updated in such a way that the hypercube is always dilated to a minimum extent. In order to ensure the minimum extent of dilation, the hypercube is first rotated in order to minimize the error. The dilation of
the hypercube is then performed based on the residual error. From equation 4.17, the parameter \( \varepsilon \) is changed in the online mode according to the following rule:

\[
\Delta \varepsilon(t) = \begin{cases} 
\gamma(t) \sum_{i: |y_i| > 1} \sqrt{D(y)} \frac{(|y_i| - 1)}{D(y)} & \text{for } \sqrt{D(y)} > \varepsilon(t) \\
0 & \text{otherwise.}
\end{cases}
\] (4.18)

\( \gamma(t) \) is constant decreasing with time such that \( \lim_{t \to \infty} \gamma(t) = 0 \), \( \lim_{t \to \infty} \sum_i \gamma(t) \to \infty \), and \( \sum_i \gamma^2(t) < \infty \).

5 Experimental Results

The effectiveness of the proposed method is demonstrated on three randomly generated source signals,

\[ s(t) = [N_1(t), N_2(t), N_3(t)] \]

where each \( N_i(t) \) is uniformly distributed in \([-1, 1]\). The mixing matrix \( A \) is an arbitrarily chosen orthogonal matrix. The proposed method is also compared with the Kullback-Leibler divergence-based algorithm. The performance of these algorithms is compared considering the same mixing matrix and the same initializations of \( W \).

The performance index is measured by

\[
\text{index} = \sum_{i=1}^{n} \left( \sum_{j=1}^{n} \frac{|C_{ij}|}{\max_k |C_{ik}|} - 1 \right) + \sum_{j=1}^{n} \left( \sum_{i=1}^{n} \frac{|C_{ij}|}{\max_k |C_{kj}|} - 1 \right),
\] (5.1)

where \( C = [C_{ij}] = WA \). When \( C \) is close to the identity matrix, this is essentially the same as the measure

\[
\sum_{i,j} |V_{ij}| = 1 - h(W)
\] (5.2)

by the variational distance.

\( W \) is updated such that it always remain orthogonal matrix. As described in section 2.1, the updating rule for \( W \) is

\[
\Delta W = (\exp(\eta Z) - I) W.
\] (5.3)

Due to the complexity in the direct computation of \( \exp(\eta Z) \), it can be evaluated from the second-order expansion given as

\[
\Delta W = \eta Z W + \frac{1}{2} \eta^2 Z^2 W.
\] (5.4)
\[ \Delta W \text{ can also be derived from the differential equation,} \]

\[ \frac{dW(t)}{dt} = Zw(t), \quad \text{for } t \in [0, \eta]. \tag{5.5} \]

For a small value of \( \eta \), we can approximate equations 5.4 and 5.5 by

\[ \Delta W = \eta Zw. \tag{5.6} \]

In this simulation, we set some previously defined small constant \( \kappa \), such that if \( \eta < \kappa \) then we compute using equation 5.6; otherwise compute by either equation 5.4 or 5.5. We define

\[ k = \left[ \frac{\eta}{\kappa} \right] \quad \text{and} \quad \kappa_0 = \eta - k\kappa. \tag{5.7} \]

\( W \) is updated as

\[
\begin{align*}
W &= W + \kappa (y_g(y') - g(y)y')W, \quad \text{for } k \text{ steps} \\
W &= W + \kappa_0 (y_g(y') - g(y)y')W, \quad \text{in the } k + 1 \text{ step.} \tag{5.8}
\end{align*}
\]

This kind of updating becomes equivalent to equation 5.5 for infinitesimally small \( \kappa \). In the simulation we consider \( \kappa = 0.5 \). After each iteration \( W \) is normalized by

\[ w_{ij} = \frac{w_{ij}}{\sqrt{\sum_{k=1}^{n} w_{kj}^2}}. \]

The results are compared with the Kullback-Leibler (KL) divergence measure–based method. In the KL divergence measure–based method, the nonlinear function does not exist for uniform distribution. However, one typical choice of the nonlinear function for subgaussian distribution (negative kurtosis) is \( \phi_i(y_i) = y_i^2 \), where \( a \) is a positive constant. The rate of convergence of the KL divergence–based method can be increased by increasing the learning-rate constant at the cost of stability. In order to have a good trade-off between convergence and stability, a decaying learning-rate constant is used. The learning rate is experimentally chosen, and it has been found that the KL divergence-based method performs optimally for

\[ \eta = \frac{0.05}{1 + 0.005t} \tag{5.9} \]

for the given uniform distribution and chosen \( \phi \).

Figure 4 demonstrates the effectiveness of the proposed and the KL divergence measure–based methods. The proposed algorithm exhibits better
Blind Separation of a Mixture of Uniformly Distributed Source Signals

Figure 4: (Top) Performance of the proposed algorithm in terms of index under noiseless condition. $\eta$ is chosen from equation 2.23. (Middle) Performance of the KL divergence measure–based algorithm under noiseless condition with the nonlinearity $\phi(y) = y^a$, $a = 3$. (Bottom) Performance of the KL divergence measure–based algorithm under the same condition with $a = 11$.

performance in terms of the speed of convergence and stability. In both cases, the same sequence of input, the same initialization, and the same mixing matrix are used.

The proposed algorithm incorporates the adaptive learning rate (see equations 2.23 and 4.15). In other words, the algorithm is also able to perform blind separation in the presence of a changing mixing matrix. Note that the adaptive learning rate for blind separation has also been studied in Murata, Muller, Ziehe, and Amari (1996) and Cichocki, Amari, Adachi, and Kasprazak (1996). In the proposed technique the adaptation of learning rate for uniformly distributed signals is performed in a completely different but effective way (see equation 2.23), which is illustrated in Figure 5.

6 Conclusions

Within a restricted domain of uniformly distributed signals, a new algorithm for blind separation is proposed. The mixing matrix is assumed to be
orthogonal. The proposed method is conceptually different from the methods based on the maximization of entropy (Bell & Sejnowski, 1995), which is based on the minimization of mutual information (Yang & Amari, 1997) and independent component analysis (Comon, 1994). The learning rule is similar to the EASI algorithm (Cardoso & Laheld, 1996), although a different nonlinear function is used in the proposed technique. A similar kind of nonlinearity may also be derived from the nonlinear principal component analysis (Karhunen & Joutsensalo, 1994, 1995; Oja, 1995; Oja & Karhunen, 1995; Oja, Karhunen, Wang, & Vigario, 1995). Theoretically, it has been shown that the proposed algorithm is very similar to an ideal $O(1/T)$ convergent algorithm, whereas the existing algorithms are only Fisher efficient, that is, $O(1)$ convergent. The algorithm may also be extended conceptually for any nonorthogonal mixing matrix in future.

**Appendix A: Training Error**

The average training error (see equation 2.10) is given as

$$
\langle e \rangle = \frac{1}{T} \sum_{i:|y_i|>1} (|y_i| - 1)
$$

$$
= \sum_{i} \mathcal{E} (\sigma(y_i)),
$$

(A.1)

where $\mathcal{E}$ stands for expectation and $\sigma(y_i)$ is given as

$$
\sigma(y_i) = \begin{cases} 
|y_i| - 1 & \text{if } |y_i| > 1 \\
0 & \text{otherwise}
\end{cases}
$$

(A.2)

From equation 3.4, $y$ can be written as

$$
\tilde{y} = y - Vy,
$$

(A.3)
where
\[ \tilde{y} = W_0 W^{-1} y = C^{-1} y. \]

Therefore,
\[ y_i = \tilde{y}_i + \sum_{i,j} V_{ij} y_j. \quad (A.4) \]

In order to compute \( \mathcal{E}(\sigma(y_i)) \), we can consider only the positive values of \( y \)'s without the loss of generality. Therefore,
\[
\mathcal{E}(\sigma(y_i)) = \int_0^1 dy_1 \cdots \int_0^1 dy_{i-1} \int_0^1 dy_{i+1} \cdots \int_0^1 \sum_{i,j} V_{ij} y_i dy_i \\
= \frac{1}{6} \sum_j V_{ij}^2 + \frac{1}{4} \sum_{j \neq k} V_{ij} V_{ik}. \quad (A.5)
\]

Considering \( V_{ij} = -V_{ji} \), the average error can be expressed as
\[
\langle \varepsilon \rangle = \frac{5}{12} \| V \|_F^2 - \frac{1}{4} \sum_{j,k} (V^2)_{jk}. \quad (A.6)
\]

Therefore the average training error has a second-order relationship with \( \delta W \).

**Appendix B: Variational Distance**

In the ideal condition, the output distribution generated by the demixing matrix should be equal to the input distribution
\[
p(y) = p(y, C^{-1}) \quad (B.1)
\]

As described in section 2.2, the difference between the input and output distributions can be measured by the variational distance, given as,
\[
D[p(y; C), p(y)] = \int |p(y; C) - p(y)| dy \\
= \frac{1}{2} |H(C) - H(I)|, \quad (B.2)
\]

where \( H(C) \) and \( H(I) \) has the same definition as in equation (3.1). Therefore,
\[
D = \left\{ y ||(C^{-1} y)_i| \leq 1 \land |y_i| > 1 \text{ for some } i \right\}. \quad (B.3)
\]
In the vicinity of the true solution, $C$ can be expanded as (from equations 3.3 and 3.4),

$$C = I + V,$$

where $V$ is an antisymmetric matrix. Similarly, we can consider

$$C^{-1} = I - V.$$

Therefore

$$y_i = (C^{-1}y)_i + \sum_{j \neq i} V_{ij}y_j.$$  \hfill (B.4)

The hyperboundary corresponding to $(C^{-1}y)_i = 1$ becomes

$$y_i = 1 + \sum_{j \neq i} V_{ij}y_j.$$  \hfill (B.5)

Similar expression can be obtained for $(C^{-1}y)_i = -1$. Without loss of generality we can consider all $V_{ij}$s to be positive for $i < j$. Therefore, $D$ can be evaluated by considering the volume enclosed within the bounding hyperplanes corresponding to $y_i = 1$ and that given by equation B.5. In the vicinity of the true solution, considering $V_{ij}$s to be sufficiently small, we get

$$|H(C) - H(I)| = 2^n \sum_i \int \cdot \cdot \cdot \int \sum_{j \neq i} |V_{ij}|y_jdy_1dy_2 \cdot \cdot \cdot dy_{i-1}dy_{i+1} \cdot \cdot \cdot dy_n. \hfill (B.6)$$

The multiplying factor of two is due to the two opposite bounding hyperplanes corresponding to 1 and $-1$. The limits of the integrals are

$$y_j \in [0, 1 + \sum_{k \neq j} V_{jk}y_k].$$

Evaluating the integral we get

$$D = \frac{1}{2}|H(C) - H(I)|$$

$$= 2^{n-1} \sum_{i < j} |V_{ij}| + \text{higher-order terms of } V_{ij}$$.  \hfill (B.7)

In the vicinity of the true solution, considering $V_{ij}$s to be sufficiently small, we can approximate

$$D = 2^{n-1} \sum_{i < j} |V_{ij}|. \hfill (B.8)$$


Received August 21, 1997; accepted August 12, 1998.