IEICE TRANS. FUNDAMENTALS, VOL. E84-A, NO. 10 OCTOBER 2001

PAPER

Blind Separation of Sources using Density Estimation and Simulated Annealing

C. G. Puntonet[†], *and* **A. Mansour** [a) ††], *Regular Member*

SUMMARY This paper presents a new adaptive blind separation of sources (BSS) method for linear and non-linear mixtures. The sources are assumed to be statistically independent with non-uniform and symmetrical PDF. The algorithm is based on both simulated annealing and density estimation methods using a neural network. Considering the properties of the vectorial spaces of sources and mixtures, and using some linearization in the mixture space, the new method is derived. Finally, the main characteristics of the method are simplicity and the fast convergence experimentally validated by the separation of many kinds of signals, such as speech or biomedical data.

key words: Independent Component Analysis (ICA), Decorrelation, High Order Statistics, Density Estimation, Simulated Annealing and Geometrical Approaches.

1. Introduction

The problem of linear blind separation of sources involves obtaining the signals generated by p sources, vectorially represented by $X(t) = (x_1(t), \cdots, x_p(t))^T$, from the linear mixture signals, $E(t) = (e_1(t), \cdots, e_p(t))^T$ (we assume that the number of sources is equal to the number of sensors):

$$E(t) = \mathbf{A}(t)X(t) \tag{1}$$

$\mathbf{A}(t) = (a_{ij}(t))$ stands for the effect of the channel (i.e. the linear mixing matrix in the case of instantaneous mixture). The mixture is considered as stationary, when $\mathbf{A}(t)$ is constant, i.e., $\mathbf{A}(t) = \mathbf{A}$. The separation is considered achieved [1] when one can estimate a matrix $\mathbf{W}(t) = (w_{ij}(t))$ such: The goal traditionally thought within the context of separation of sources is to estimate $\mathbf{A}(t)$ by means of another matrix $\mathbf{W}(t)$ such that the output vector, $S(t)$:

$$S(t) = (s_1(t), \cdots, s_p(t))^T = \mathbf{W}^{-1}(t)E(t), \tag{2}$$

it coincides with the original sources, $X(t)$, except for a scale factor and a permutation, i.e.,

$$\mathbf{W}(t) = \mathbf{A}(t)\mathbf{PD} \tag{3}$$

where $\mathbf{P}$ is a permutation matrix and $\mathbf{D}$ is a full-rank diagonal matrix. Any matrix $\mathbf{W}$ related to $\mathbf{A}$ as in (3) is said to be similar to $\mathbf{A}$.

In the ICA framework, many approaches have been presented, with applications in real world problems [2]: as communications, feature extraction, pattern recognition, data visualization, speech processing and biomedical signal analysis (EEG, MEG, fMRI, etc), considering the hypothesis that the medium where the sources have been mixed is linear, convolutive or non-linear. ICA is a linear transformation that seeks to minimize the mutual information of the transformed data, $E(t)$, the fundamental assumption being that individual components of the source vector, $X(t)$, are mutually independent and have, at most, one Gaussian distribution [3]. The 'Infomax' algorithm [4] is an unsupervised neural network learning algorithm that can perform blind separation of input data into the linear sum of time-varying modulations of maximally independent component maps, providing a powerful method for exploratory analysis of functional magnetic resonance imaging (fMRI) data [5]. Using the maximization of the negentropy, an ICA 'Infomax' algorithm for unsupervised exploratory data analysis applied to electroencephalograph (EEG) monitor output has been introduced [6]. A great number of solutions for BSS are based on the minimization or cancellation of independence criteria (that use higher-order statistics) [7], [8]. From geometric considerations, and for linear mixtures of bounded sources, various algorithms have been presented, all of which find a matrix that is similar to $\mathbf{A}$ by determining the slopes of the edges that are incident on any one of the vertices of the hyperparallelepiped that contains the observation space [9]–[11]. Using a contrast function defined in terms of the Kullback-Leibner divergence or of the mutual information and exploiting the information on the distribution support, another ICA procedure derived for separating an instantaneous mixture of sources, based on order statistics has recently been developed [12].

For non-linear mixtures, a modified self-organizing map algorithm based on density estimation has been developed [13], extracting the local geometrical structure of distributions obtained from mixtures of statistically independent sources and performing non-parametric histogram density estimation; this method is appropriate for sharply peaked distributions. For post-nonlinear mixtures, a batch procedure based on a maximum likelihood approach has been developed [14]. In [15] an

Manuscript received November 15, 2000.
Manuscript revised May 22, 2001.

[†]The author is with the Dept. of Architecture and Computer Technology. University of Granada. Granada, Spain.
[††]Bio-Mimetic Control Research Center (RIKEN), Nagoya, 463-0003 Japan.
a) E-mail: mansour@ieee.org

adaptive procedure is described for the demixing of linear and non-linear mixtures of two signals with probability distribution functions (PDF) that are symmetric with respect to their centres, and non uniform, performing a fixed piecewise linearization in the case of nonlinear mixtures in order to obtain the distribution axes of probability that are parallel to the slopes of the parallelepiped for two sources. ICA is a promising tool for the exploratory analysis of biomedical data. In this context, a generalized algorithm modified by a kernel-based density estimation procedure has been studied in [16] to separate EEG signals from tumour patients into spatially independent signals, the algorithm allowing artifactual signals to be removed from the EEG by isolating brain-related signals into single ICA components. Using an adaptive geometry-dependent ICA algorithm, Puntonet et al. [17] demonstrate the possibility of separating biomedical sources, such as EEG signals, analyzing only the observed mixing space due to the almost symmetric PDF of the mixtures. The approach presented in this paper combines the geometric properties of the distributions, which provide the independent components, with the advantages of competitive neural networks, by means of a dynamic piecewise linearization. Finally, in order to provide fast initial convergence, a simulated annealing technique has been used.

2. Proposed Method

Our method combines adaptive processing with a simulated annealing technique. At first, a preprocessing stage to normalize† the observed space, $E(t)$, in a set of concentric spheres, is needed in order to adaptively compute the slopes corresponding to the independent axes of the mixture distributions by means of an array of symmetrically distributed neurons in each dimension. The normalization stage is followed by the processing or learning of those neurons, which estimate the high density regions in a way similar, but not identical to that of self organizing maps. A simulated annealing method provides a fast initial movement of the weights towards the independent components by generating random values of the weights and minimizing an energy function. In general, for BSS and taking into account the possible presence of non-linear mixtures, the observation space $(e_1, \cdots, e_p)$ is subsequently quantized in n spheres of dimension p, circles if $p = 2$, each with a radius†† $\rho(k)$ $(k = 1 \cdots n)$ covering the points as follows:

$$\rho(k-1) < \|E(t)\| < \rho(k) \tag{4}$$

†In order to work with well conditioned signals, the observed signals $e_i(t)$ are preprocessed or adaptively set to zero mean, μ_i, and unity variance, σ_i, as follows: $e_i(t) = \frac{e_i(t)-\mu_i}{\sigma_i}$, where $i \in \{1, \cdots, p\}$.

††The radius $\rho(k)$ can be determined by equation (21).

$\rho(0) = 0$ and $\forall k \in \{1, \cdots, n\}$. From now on, we use $E(\rho(k), t)$ to denote the vector $E(t)$ that verifies (4). If, in some applications, the mixture process is known to be linear then, the number, n, of layers is set to 1, and a normalization of the space is made with $\rho(1) = 1$. Although the quantization given in (4) allows a piecewise linearization (when n increases) of the observed space for the case of non linear mixtures, it is also useful with the assumption of linear media since it allows us to detect unexpected non linearities [17].

2.1 Density Estimation

The above-described preprocessing is used to apply a density estimation technique by means of a neural network whose weights are initially located on the Cartesian edges of the p-dimensional space, such that there are p neurons with $2p$ weights per layer. The distance between a point, $E(\rho(k), t)$, and the $2p$ weights existing in the p-dimensional space (Figure 3) is:

$$d(i, \rho(k)) = \|\tilde{W}_i(\rho(k), t) - E(\rho(k), t)\| \tag{5}$$

$\tilde{W}_i(\rho(k), t)$ is a p dimensional vector, $i \in \{1, \cdots, 2p\}$, and $k \in \{1, \cdots, n\}$. A winner neuron, labeled i^*, in a layer $\rho(k)$, is at a minimum distance from the point $E(\rho(k), t)$ and verifies:

$$d(i^*, \rho(k)) = \min\{d(i, \rho(k))\} \tag{6}$$

$i \in \{1, \cdots, 2p\}$ and $k \in \{1, \cdots, n\}$. For the sake of simplicity, we use ρ to denote the layer $\rho(k)$ defined in (4). The main learning process for density estimation when a neuron approaches the density region, at time t, is given by:

$$\tilde{W}_i(\rho, t+1) = \tilde{W}_i(\rho, t) + \alpha(t) f(E(\rho, t), \tilde{W}_i(\rho, t))$$

with $\alpha(t)$ being a decreasing learning rate and $i \in \{1, \cdots, 2p\}$. Note that a great variety of suitable functions, $\alpha()$ and $f()$, can be used. In particular, a learning procedure that activates all the neurons at once is adequate by means of a factor, $K(t)$, that modulates competitive learning as in self-organizing systems, i.e.,

$$\begin{aligned} \tilde{W}_i(\rho, t+1) &= \tilde{W}_i(\rho, t) + \\ &\quad \alpha(\rho, t)\mathrm{sgn}[E(\rho, t) - \tilde{W}_i(\rho, t)]K_i(t) \\ K_i(t) &= \exp(-\eta^{-1}(t)\|\tilde{W}_i(\rho, t) - \tilde{W}_{i^*}(\rho, t)\|^2) \end{aligned} \tag{7}$$

Here $\eta(t)$ is a neighborhood decreasing parameter, $i \in \{1, \cdots, 2p\}$, $i^* \in \{1, \cdots, n\}$ and $\alpha(t)$ is now geometry-dependent and proportional to $\eta(t)$, as follows:

$$\alpha(\rho, t+1) = \eta(t)\rho\delta \tag{8}$$

where $0 < \eta(t) < 1, \rho \in \{\rho(1), \cdots, \rho(n)\}$, δ and ρ modify the value of the learning rate, $\alpha(t)$, depending on the correlation of the points in the observation space and on the number of layers in order to equalize the angular

velocity of the outer and inner neurons. Note that the weight update is carried out using the sign function, in contrast to the usual way [18]. As is well known, the term $K(t)$ modulates the learning sphere of jurisdiction depending on the value of $\eta(t)$. After the learning process, the neurons are maintained in their respective layers, ρ, by means of the following normalization:

$$\tilde{W}_i(\rho,t) = \frac{\tilde{W}_i(\rho,t)\rho}{\|\tilde{W}_i(\rho,t)\|} \tag{9}$$

$i \in \{1,\cdots,2p\}$ and $\rho \in \{\rho(1),\cdots,\rho(n)\}$. After converging, at the end of the density estimation process, the weights in (9) will be located at the centre of the projections of the maximum density points, or independent components, in each layer. For the purpose of BSS, a matrix $\mathbf{W}$ similar to $\mathbf{A}$ and verifying expression (3) is needed. Once the neural network has estimated the maximum density subspaces by means of adaptive equation (7), and due to the piecewise linearization of the observation space with n spheres, a set, $\mathbf{\Omega}$, of matrices can be defined as follows:

$$\mathbf{\Omega} = \{\mathbf{W}_{\rho(1)},\cdots,\mathbf{W}_{\rho(n)}\} \tag{10}$$

where, for p dimensions, the matrices $\mathbf{W}_\rho$ ($\rho \in \{\rho(1),\cdots,\rho(n)\}$.) have the following form:

$$\mathbf{W}_\rho = \begin{pmatrix} \mathbf{w}_{11\rho} & \cdots & \mathbf{w}_{1p\rho} \\ \mathbf{w}_{p1\rho} & \cdots & \mathbf{w}_{pp\rho} \end{pmatrix}. \tag{11}$$

For linear systems or "symmetric" non-linear mixtures (as in Figure 2, see section 5 for more details), the elements of this matrix, $\mathbf{W}_\rho$, obtained using density estimation are considered to be the symmetric slopes, in the segment of sphere ρ, between two consecutive neurons initially located on the same axis, for each dimension j, and finally computed in (7) if the following transformation is carried out under geometric considerations:

$$\mathbf{w}^d_{ij\ \rho\{k\}}(t) = \frac{\tilde{w}_{2j\ i}(\rho\{k\},t) - \tilde{w}_{2j\ i}(\rho\{k-1\},t)}{\tilde{w}_{2j\ j}(\rho\{k\},t) - \tilde{w}_{2j\ j}(\rho\{k-1\},t)} \tag{12}$$

where $\tilde{w}_{i\ j}(\rho,t)$ is the jth component of $\tilde{W}_i(\rho,t)$, $i,j \in \{1,\cdots,p\}$ and $\rho \in \{\rho(1),\cdots,\rho(n)\}$. The superscript, d, indicates that the separation matrix has been computed using density estimation, which will be useful in Section 2.3. Note that equation (12) works only with even-labeled neurons, $2j$, and can be simplified for linear media if $n = 1$ and $\rho(0) = 0$; for instance, when $p = 2$ $(j = 1,2)$ it is practical to operate with only two weights, w_2 and w_4, in the circle $\rho(1)$. If $n > 1$, the use of several p-spheres is useful for non-linearity detection, since different matrices, $\mathbf{W}_\rho$ in (11), are obtained for successive values of ρ. Nevertheless, equation (12) is shown in this form as a particular case of the expression valid for non-linear separation of sources (Sect. 4).

2.2 Simulated Annealing

Simulated annealing is a stochastic algorithm that represents a fast solution to some combinatorial optimization problems. As an alternative to the density estimation method described above, we first propose the use of stochastic learning, such as simulated annealing, in order to find a fast convergence of the weights around the maximum density points in the observation space $E(t)$. This technique will be effective if the chosen energy, or cost function, E_{ij}, for the global system is appropriate. The procedure of simulated annealing is well known [19]. It is first necessary to generate random values of the weights and, secondly, to compute the associated energy of the system. This energy vanishes when the weights become a global minimum, the method thus allowing escape from local minima. For BSS problem, we define an energy E similar to the cost function described in [20] and related to the four-order statistics of the original p sources, due to the necessary hypothesis of statistical independence between them, as follows:

$$E = \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} E_{ij}(t) \tag{13}$$

where, $E_{ij}(t) = \mathrm{Cum}^2_{22}(s_i(t), s_j(t))$ and Cum_{22} is the $2x2$ cumulant. the estimation of that energy can be done using the methods described in [21]. The change in global energy, ΔE, created by the new state after the generation of random weights, is given by: $\Delta E = E(t+1) - E(t)$. If $\Delta E < 0$ then the process accepts the change. If $\Delta E > 0$, the system accepts the change providing $P > r$, where r is a number randomly chosen for P, the Boltzmann distribution given ΔE, computed by:

$$P = \exp\left(-\frac{\Delta E}{T(t)}\right) \tag{14}$$

where $T(t)$ is the positive valued temperature at time t that regulates the search granularity for the systems global minimum. If $\Delta E > 0$ and $P < r$, then the network returns all weights to their original state. In each iteration, by incrementing the time t by 1, a new value for the temperature $T(t)$ is calculated, using the following equation (cooling schedule):

$$T(t) = \frac{T_0}{1+\eta(t)} \tag{15}$$

where T_0 is the initial temperature. The parameter $\eta(t)$ is variable, with $\eta(t) = \log(t)$ in the Boltzmann machine but $\eta(t) = t$ in the Cauchy machine. Although the main algorithm of simulated annealing has been shown above, some modifications to the procedure can be made when this method is applied to BSS. For instance, we propose the function $\eta(t)$ in (15) should be

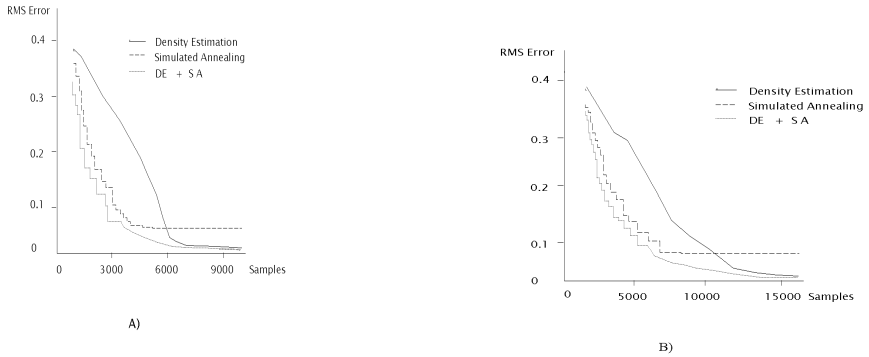


Fig. 1 Comparisons among the convergences of: Density Estimation (DE), Simulated Annealing (SA) and both of them (DE) + (SA). A) Two sources and B) Three sources.

$\eta(t) = (1+t)^2 - 1$, in order to provide fast convergence. With this process, and using $w_{i\ j}(\rho, t)$ to denote the component j of the random weight accepted by the system in a p-sphere of radius ρ, the separation matrix is easily computed by means of the following rule:

$$\mathbf{w}^s_{ij\rho}(t) = w_{ij}(\rho, t) \tag{16}$$

$i \neq j \in \{1, \cdots, p\}$ and $\rho \in \{\rho(1), \cdots, \rho(n)\}$. The superscript, s, indicates that the separation matrix has been computed using simulated annealing. Note that, as in equation (12), the coefficients of the separation matrix in (16) with indexes[†] $i = j$ are set to 1, and thus it is necessary to generate $p(p-1)$ random weights instead of p^2. Once a global minimum is obtained, when the energy in (13) vanishes, the value of the $\mathbf{W}$ matrix is close to that of the original $\mathbf{A}$ matrix, i.e., the W coefficients provide the independent components. This convergence will only be true and possible if a good choice of the energy function, E, has been made [20]. Theoretically, the proposed energy function (13) depends on a four-order cumulant; it has been experimentally corroborated in several simulations as an estimator of statistical independence, obtaining good results by estimating statistics over more than a hundred samples.

2.3 Density Estimation with Simulated Annealing

In spite of the fact that the technique presented in Section 2.2 is fast, the greater accuracy of density estimation by means of the competitive learning shown in Section 2.1 encourages us to consider a new approach. An alternative method for the adaptive computation of the weight matrix $\mathbf{W}$ concerns the simultaneous use of the two methods described in Sections 2.1 and 2.2, i.e., density estimation and simulated annealing. Now, a proposed adaptive rule of the weights is the following:

$$\mathbf{W}_{ij\rho}(t+1) = \mathbf{W}^s_{ij\rho}(t)\beta(t) + \mathbf{W}^d_{ij\rho}(t)(1-\beta(t)) \tag{17}$$

where $i \neq j \in \{1, \cdots, p\}, \rho \in \{\rho(1), \cdots, \rho(n)\}$ and $\beta(t)$ is a decreasing function that can be chosen in several ways (Section 3). The main purpose of equation (17) is to provide a fast initial convergence of the $\mathbf{W}$ coefficients by means of simulated annealing during the epoch in which the adaptation of the neural network by density estimation is still slow. When the value $\beta(t)$ goes to zero, the contribution of the simulated annealing process vanishes since the random generation of weights ceases, and the more accurate density estimation by means of competitive learning begins. The main contribution of simulated annealing here is the fast convergence compared to the adaptation rule (7), thus obtaining an acceptable closeness of $\mathbf{W}$ to the distribution axes (independent components). However, the accuracy of the solution when the temperature, $T(t)$, is low depends mainly on the adaptation rule presented in section 2.1 using density estimation since, with this, the energy in (13) continues to decrease until a global minimum is obtained.

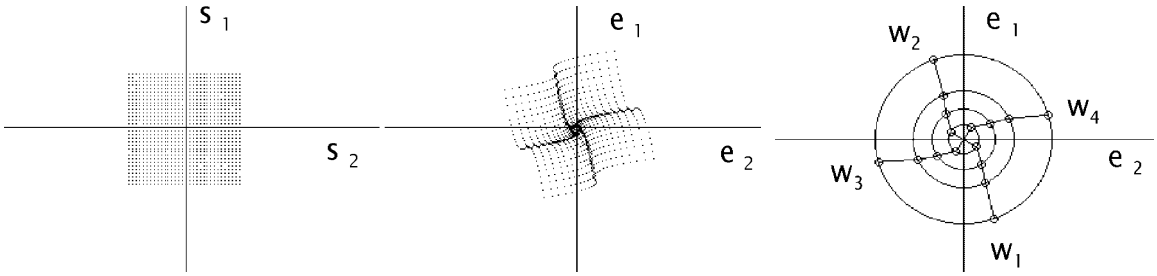
A measure of the convergence in the computation of the independent components with the number of samples or iterations is shown in Figures 1 and 2, which compare the methods, density estimation and simulated annealing, using the root mean square error (RMSE), $\epsilon(t)$, defined as follows:

$$\epsilon(t) = \frac{1}{p(p-1)} \sqrt{\sum_{i \neq j} (w_{ij}(t) - a_{ij}(t))^2} \tag{18}$$

$i, j \in \{1, \cdots, p\}$. Note that, a priori, the unknown matrix $A(t)$ depends on time, although in the simulations it remains constant (Section 5). Figure 1.A shows the RMSE in the case of $p = 2$, with the two sources having kurtosis[††] values of $k_{s1} = -0.02$ and $k_{s2} = 0.02$, respectively. Using simulated annealing and

[†]Using the fact that the separation can be achieved up to a factor, see (3).

[††]The kurtosis can provide some information concerning the distribution of a signal $x(t)$ [22] and it is given by $k_x = \frac{<x(t)^4> - 3<x^2(t)>^2}{<x(t)^2>^2}$, where $< x(t) >$ is the expectation of $x(t)$.



A - Estimated sources. B - Observed signals. C - Evaluation of the weight matrix.

Fig. 2 Separation of non linear mixture.

10000 samples the error remains at $\epsilon = 0.05$, whereas using simulated annealing and density estimation the error becomes $\epsilon = 0.01$ with the same number of iterations. In Figure 1.B the RMSE in the case of $p = 3$ is shown. The three sources have kurtosis values of $k_{s1} = 3.1, k_{s2} = 3.5$ and $k_{s3} = 3.2$, respectively. In this case, with a larger number of sources to be separated, using simulated annealing and 15000 samples the error remains at $\epsilon = 0.06$, whereas using simulated annealing and density estimation the error becomes $\epsilon = 0.01$. Although simulated annealing is a stochastic process, the error values presented here are the result of several simulations and are for guidance only since each experiment presents some randomness and is never the same because of the different mixture matrices and sources.

3. Some Improvements

The techniques presented in Section 2 can be modified to improve basic performance parameters such as time convergence and accuracy. For instance, in relation to density estimation and linear media, we propose to eliminate some points that do not provide outstanding information, either by previous preprocessing or adaptive processing; this is done by means of the average correlation coefficient, computed as follows:

$$< c_e >= \frac{1}{p(p-1)} \sum_{i,j} c_{eij} \text{ and } c_{eij} = \frac{1}{T} \sum_{t=1}^{T} e_i(t) e_j(t)$$

$i, j \in \{1, \cdots, p\}, i < j$ and defining a parameter $\delta = \exp(- < c_e >^2)$. For linear mixtures, many kinds of sources, such as speech signals, contain unnecessary points near the origin that do not provide information when the computation of the distribution axes is being carried out; these can be removed (not processed), with $n = 1$ in (4), if the following condition is verified:

$$\|E\| < \sum_i \sigma_i \delta = R \tag{19}$$

where $R < \rho(1)$ is the radius of the p-sphere and $i \in \{1, \cdots, p\}$. Furthermore, and in order to improve time convergence in the density estimation, equation (7) can be simplified for certain applications in which only a winner neuron, i, approaches the density region in each iteration, thus eliminating the term $K(t)$. A similar type of learning can be used when the learning space of each neuron, i_q, is reduced to its associate quadrant, q_i, the range of q_i being $\pi/2$; this is useful when it is known in certain real applications that the mixing matrix, $\mathbf{A}$, verifies $a_{ii} > a_{ij} \quad (i, j = 1, \cdots, p)$. If this is so, only the representative winner neuron, i_q^*, is active, and it is only necessary to detect the quadrant that $e(\rho, t)$ belongs to. Another fact that speeds up the learning task concerns equation (7) for linear or nonlinear symmetrical mixtures (Simulation 1, Figures 3 and 4), since the symmetry of the distribution of points means that each time a neuron i learns, the other neuron located on the same axis, j, also learns but in the opposite direction and vice versa, as follows:

$$\begin{aligned} \tilde{W}_i(\rho, t+1) &= \tilde{W}_i(\rho, t) \\ &\quad + (-1)^{\bar{\imath}-i} \alpha(t) sgn(E(\rho, t) - \tilde{W}_{\bar{\imath}}(\rho, t)) \\ \tilde{W}_j(\rho, t+1) &= \tilde{W}_j(\rho, t) \\ &\quad + (-1)^{\bar{\imath}-j} \alpha(t) sgn(E(\rho, t) - \tilde{W}_{\bar{\imath}}(\rho, t)) \end{aligned} \tag{20}$$

$\bar{\imath} \in \{i, j\}$, $i \in \{1, 3, \cdots, 2p-1\}$ and $j \in \{2, 4, \cdots, 2p\}$. Some improvements are also feasible in the estimation of the distribution axes in non-linear mixtures, since the spatial neuron order (Figure 5) in successive layers may change due to the form of the density distribution; for correct adaptive separation in equation (23) it is necessary to check, periodically, the following: If $\|w_i(\rho, t) - w_j(\rho - 1, t)\| < \|w_i(\rho, t) - w_i(\rho - 1, t)\|$, then $w_i(\rho - 1, t) = w_j(\rho - 1, t)$, here $i \neq j \in \{1, \cdots, 2p\}$.

Once this expression is computed, the rearranging is done bottom-up, beginning from the first layer. Furthermore, in linear or non-linear mixtures, the real observed signals may exhibit non-uniform density distributions (Figure 4), and the procedure generates adaptively variable layers in accordance with the density of points. Then, the distance between the circles, $\rho(k, \tau)$, in time τ, can be adjusted as a function of the density of points, $\lambda(k, \tau)$, between two successive layers:

$$\rho(k, \tau + 1) = \rho(k, \tau) + \gamma(\lambda(k-1, \tau) - \lambda(k, \tau)) \tag{21}$$

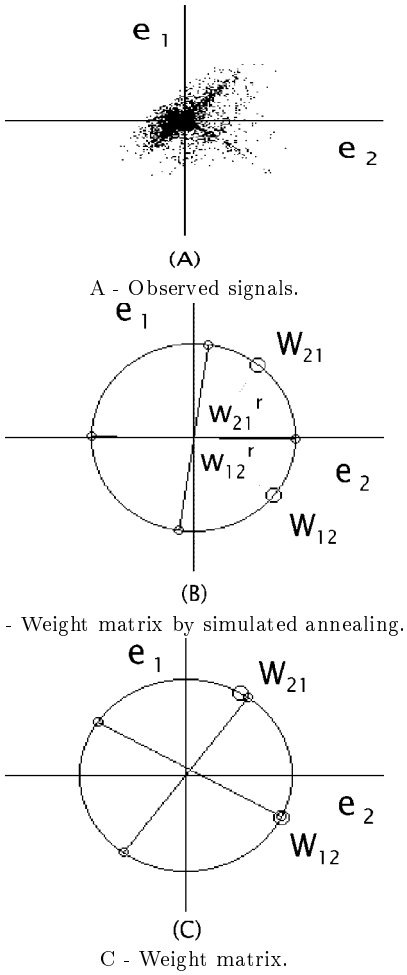


Fig. 3 Simulations in Linear and nonlinear symmetrical mixtures.

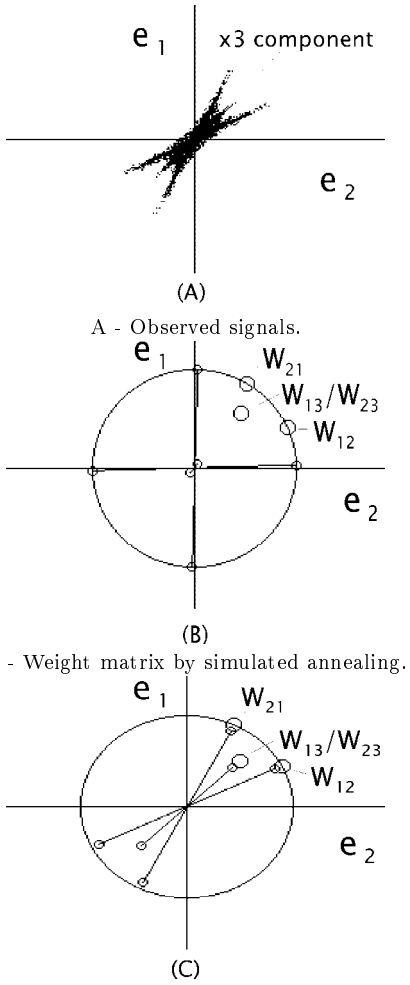


Fig. 5 Simulation in the case of three signals.

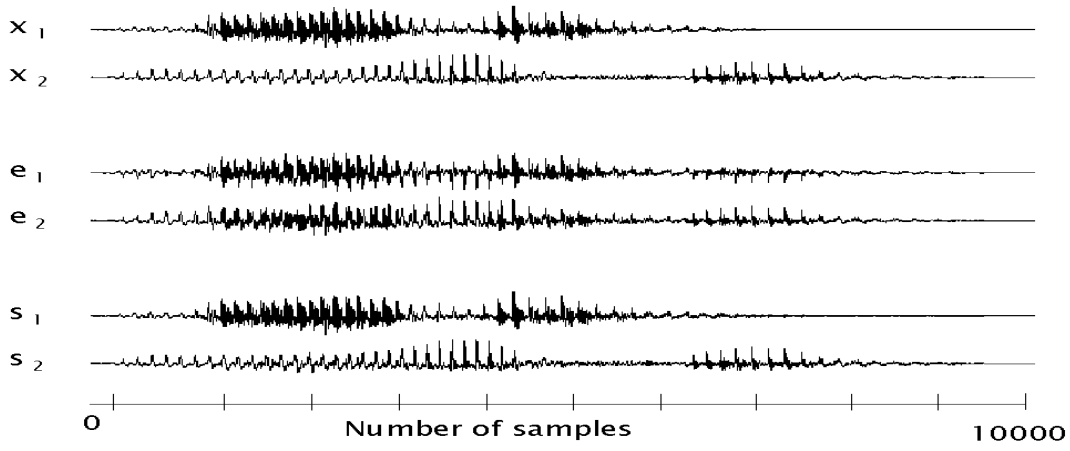


Fig. 4 Experimental results in the case of two sources.

where γ is a learning rate and $k \in \{1, \cdots, n\}$. In relation to simulated annealing, the use of this technique for the BSS, instead of (14) and (15), is based on the following expressions:

$$P = \exp\left(-\frac{\Delta E}{T^2(t)}\right) \text{ and } T(t) = \frac{T_0}{(1+t)^2} \tag{22}$$

Equation (22) allows us to find a global minimum in a fast convergence time using the energy function defined in (13). Moreover, there are several ways of implementing $\beta(t)$ in (17) in order to switch the two processes, simulated annealing and density estimation. One of them is to use, simply, a decreasing function $\beta(t)$ similar to that of $T(t)$ in (15) or (22). Another one consists of using the density estimation process when the energy decreases to a given value. Finally, we propose switching the two processes when no changes in the energy function, $\Delta E = 0$, have occurred in a given time.

4. Separation Matrix

Since the main simulations presented in this paper refer to linear mixtures of signals, we will use expression (12) for computation of the weights, although in the general case and for pure non-linear mixtures (without symmetry at the origin), the above expression must be replaced by a similar one, as follows:

$$\mathbf{w}^d_{ij\rho\{k\}}(t) = \frac{\tilde{w}_{\xi(j)\ i}(\rho\{k\}, t) - \tilde{w}_{\xi(j)\ i}(\rho\{k-1\}, t)}{\tilde{w}_{\xi(j)\ j}(\rho\{k\}, t) - \tilde{w}_{\xi(j)\ j}(\rho\{k-1\}, t)} \tag{23}$$

$i, j \in \{1, \cdots, p\}$, $\rho \in \{\rho(1), \cdots, \rho(n)\}$, $\xi(j) \in \{\xi(1) < \xi(2) < \cdots < \xi(p)$ such $d(\xi(j), \rho) < d(\xi(m), \rho)\}$, $m \in \{1 \cdots 2p\}$ and $m \neq j$. Note that equation (12) is a particular case of equation (23), with $\xi(j) = 2j$, and that the coefficients $W^d_{ii\rho} = 1$ in both expressions. Equation (23) means that the p-dimensional subspace associated to the neurons labeled $(\xi(1), \cdots, \xi(p))$ around point e_p provides the linear contour where the mixture can be considered as linear. For the purpose of separation, the network uses the typical recursive recall, taking into account the layer quantization in the observation space and the matrix computed in (17), i.e.:

$$s_i(t+1) = e_i(\rho, t) - \sum_{j=1}^{p} \mathbf{W}_{ij\rho}(t) s_j(t) \tag{24}$$

$i \in \{1, \cdots, p\}, i \neq j$ and $\rho \in \{\rho(1), \cdots, \rho(n)\}$. This expression is also used by the simulated annealing process in order to compute the energy function in (13).

5. Simulation Results

Three simulations are presented in order to show the efficiency of the proposed algorithms. The crosstalk parameter, ct_i, is used to verify the similarity between the original, x_i, and separated, s_i, signals with N samples, and it is defined as follows:

$$ct_i = 10 \log\left(\frac{\sum_{t=1}^{N}(s_i(t) - x_i(t))^2}{\sum_{t=1}^{N} s_i^2(t)}\right) \tag{25}$$

$i \in \{1, \cdots p\}$. The first one, Figure 2, corresponds to the synthetic non-linear mixture suggested in [13], for sharply peaked distributions, the original sources being digital 32-valued signals with uniform PDF ($x_i(t) \in \{-16, \cdots, -1, 0, 1, \cdots, 15\}$), as follows:

$$\begin{aligned} e_1(t) &= -2\text{sgn}[x_1(t)]x_1(t)^2 + 1.1x_1(t) - x_2(t) \\ e_2(t) &= -2\text{sgn}[x_2(t)]x_2(t)^2 + 1.1x_2(t) + x_1(t) \end{aligned} \tag{26}$$

Using 20000 samples and $n = 4$ layers, good estimation of the density distribution is obtained, Figure 2 C. The four equation matrices obtained (10), using density estimation, were:

$$\mathbf{W}_{\rho(1)} = \begin{pmatrix} 1 & 1.7 \\ -1.6 & 1 \end{pmatrix}; \quad \mathbf{W}_{\rho(2)} = \begin{pmatrix} 1 & .25 \\ -.22 & 1 \end{pmatrix}$$
$$\mathbf{W}_{\rho(3)} = \begin{pmatrix} 1 & .2 \\ -.22 & 1 \end{pmatrix}; \quad \mathbf{W}_{\rho(4)} = \begin{pmatrix} 1 & .1 \\ -.15 & 1 \end{pmatrix}$$

The second simulation, shown in Figures 3 and 4, concerns the separation of a mixture of two real signals, the Spanish words "dedos (fingers)" and "mueca (doll)", captured with a 12 bit-converter, a sampling frequency of 12kHz, and presenting a signal-noise ratio of 24 dB. The correlation coefficient of the original sources was $< cs >= \frac{1}{N}\sum_{t=1}^{N} s_1(t)s_2(t) = -0.05$, and the value of the kurtosis, ks, was $k_{s1} = 4.7$ and $k_{s2} = 4.2$ for $s_1(t)$ and $s_2(t)$, respectively. The original, $\mathbf{A}$, and computed, $\mathbf{W}$, matrices obtained with 10000 samples were:

$$\mathbf{A} = \begin{pmatrix} 1 & -.8 \\ .8 & 1 \end{pmatrix}; \quad \mathbf{W} = \begin{pmatrix} 1 & -.791 \\ .788 & 1 \end{pmatrix}$$

The crosstalk parameter of the separated signals, $s_1(t)$ and $s_2(t)$, was $ct_1(t) = -24$ dB and $ct_2(t) = -23$ dB, respectively. It has been verified that the greater the kurtosis of the signals the more accurate and faster is the estimation, except for the case in which the signals are not well conditioned or are affected by noise, and this is so since a great density of points on the independent components speeds up convergence when competitive learning of equation (7) is used. Moreover, since the distribution estimation is made in the observation space, $E(t)$, and the separation is blind, it is useful to take into account the kurtosis of the observed signals in order to test the time convergence and the precision. A third simulation is presented in Figures 5 and 6 with three synthetic supergaussian signals. Note that Figures 5.A, 5.B and 5.C show the projection of the three-dimensional observation space onto the (e_1, e_2) plane. Therefore, the weight w6 provides, in this plane (e_1, e_2), a slope value of $+1$, corresponding to the quotient (W_{13}/W_{23}) in (12), with $(i, j) = (1, 3)$ and $(i, j) = (2, 3)$. The correlation coefficient for the original sources was $< cs >= -0.08$, and the kurtosis, k_e, of

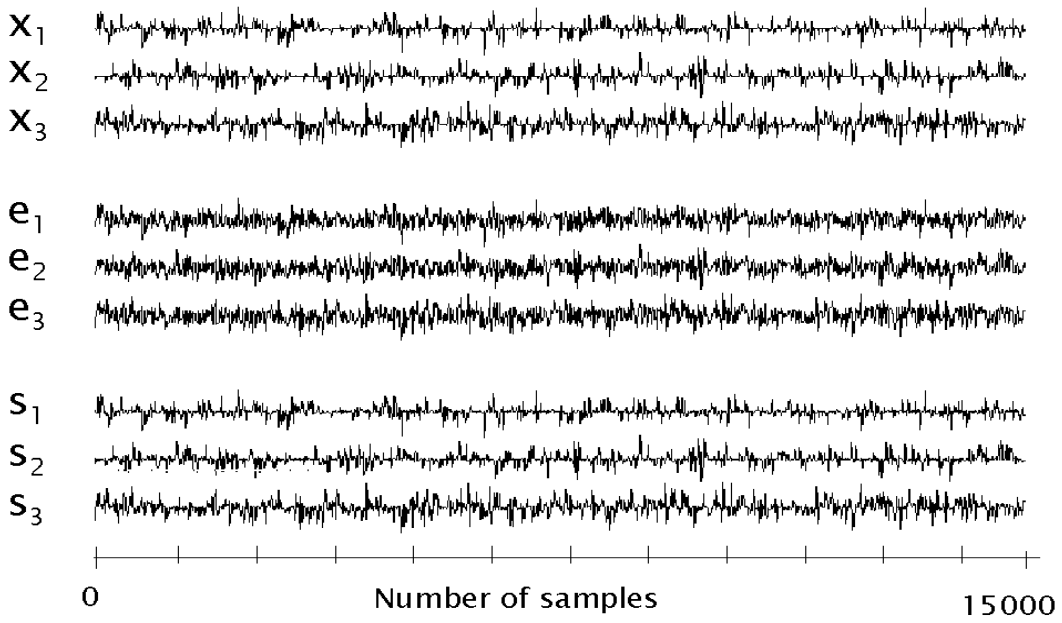


Fig. 6 Simulation results: Three sources.

the three observed signals, was $k_{e1} = 3.4, k_{e2} = 2.6$ and $k_{e3} = 3.2$. The original, $\mathbf{A}$, and weight, $\mathbf{W}$, matrices obtained with 15000 iterations were:

$$\mathbf{A} = \begin{pmatrix} 1 & .5 & .5 \\ .5 & 1 & .5 \\ .5 & .5 & 1 \end{pmatrix}; \ \mathbf{W} = \begin{pmatrix} 1 & .494 & .492 \\ .505 & 1 & .511 \\ .519 & .502 & 1 \end{pmatrix}$$

The crosstalk parameter of the three signals $s_1(t)$, $s_2(t)$ and $s_3(t)$ was $ct_1(t) = -22$ dB, $ct_2(t) = -32$ dB and $ct3_(t) = -26$ dB, respectively.

6. Conclusion

We have shown a new powerful adaptive-geometric method based on competitive unsupervised learning and simulated annealing, in order to find the distribution axes of the observed signals or independent components, by means of a piecewise linearization in the mixture space. The time convergence of the network is fast, even for more than two signals, mainly due to the initial simulated annealing process that provides a good starting point with a low computation cost, and the accuracy of the network is adequate for the separation task, the density estimation being very precise, as several experiments have corroborated. Besides the study of noise, future work will concern the application of this method to ICA with linear or nonlinear mixtures of biomedical signals, such as in EEG and fMRI, where the number of signals increases sharply, making simulated annealing suitable in a quantized high-dimensional space.

Acknowledgments

This work has been supported in part by the Spanish CICYT projects TIC98-0982.

References

[1] P. Comon, C. Jutten, and J. Hérault, "Blind separation of sources, Part II: Statement problem," *Signal Processing*, vol. 24, no. 1, pp. 11–20, November 1991.

[2] A. Mansour, A. Kardec Barros, and N. Ohnishi, "Blind separation of sources: Methods, assumptions and applications.," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E83-A, no. 8, pp. 1498–1512, 2000, Special Section on Digital Signal Processing in IEICE EA.

[3] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol. 36, no. 3, pp. 287–314, April 1994.

[4] Bell A. J. and Sejnowski T. J., "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, November 1995.

[5] M.J. Mckeown, S. Makeig, G.G. Brown, T-P. Jung, S.S. Kinderm, A.J. Bell, and T.J. Sejnowski, "Analysis of fmri data by blind separation into independent spatial components," *Human Brain Mapping*, vol. 6, pp. 160–188, 1998.

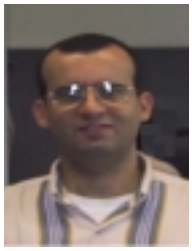
[6] M. Girolami, "The latent variable data model for exploratory data analysis and visualization: A generalisation of the nonlinear infomax algorithm," *Neural Processing Letters*, vol. 8, no. 1, pp. 27–39, 1998.

[7] A. Mansour and C. Jutten, "Fourth order criteria for blind separation of sources," *IEEE Trans. on Signal Processing*, vol. 43, no. 8, pp. 2022–2025, August 1995.

[8] A. Mansour and C. Jutten, "A direct solution for blind separation of sources," *IEEE Trans. on Signal Processing*, vol. 44, no. 3, pp. 746–748, March 1996.

[9] G. Puntonet, C., A. Mansour, and C. Jutten, "Geometrical algorithm for blind separation of sources," in *Actes du XVème colloque GRETSI*, Juan-Les-Pins, France, 18-21 September 1995, pp. 273–276.
[10] C. G. Puntonet and A. Prieto, "Neural net approach for blind separation of sources based on geometric properties," *NeuroComputing*, vol. 18, no. 3, pp. 141–164, 1998.
[11] A. Prieto, C. G. Puntonet, and B. Prieto, "A neural algorithm for blind separation of sources based on geometric prperties.," *Signal Processing*, vol. 64, no. 3, pp. 315–331, 1998.
[12] D. T. Pham, "Blind separation of instantaneous mixtures of sources based on order statistics," *IEEE Trans. on Signal Processing*, vol. 48, no. 2, pp. 1712–1725, Feburary 2000.
[13] J.K. Lin and J.D. Cowan, "Faithful representation of separable input distributions," *Neural Computation*, vol. 9, pp. 1305–1320, 1997.
[14] A. Taleb and C. Jutten, "Source separation in postnonlinear mixtures," *IEEE Trans. on Signal Processing*, vol. 47, no. 10, pp. 2807–2820, October 1999.
[15] C.G. Puntonet, M.R. Alvarez, A. Prieto, and Prieto B, "Separation of speech signals for nonlinear mixtures," 1999.
[16] M. Habl, C. Bauer, C. Ziegaus, E.W. Lang, and F. Schulmeyer, "Analyzing brain tumor related eeg signals with ica algorithms," in *First International Conference on Artificial Neural Networks in Medicine and Biology*, Goteborg, SWEDEN, May 13-16 2000.
[17] C.G. Puntonet, C. Bauer, E. W. Lang, M. R. Alvarez, and B. Prieto, "Adaptive-geometric methods: application to the separation of eeg signals," in *International Workshop on Independent Component Analysis and blind Signal Separation*, Helsinki, Finland, 19-22 June 2000, pp. 273–278.
[18] S. Haykin, *Neural Networks*, Prentice Hall, 1991.
[19] P. K. Simpson, *Artificial Neural Systems*, Pergamon Press, 1991.
[20] A. Mansour and N. Ohnishi, "Multichannel blind separation of sources algorithm based on cross-cumulant and the levenberg-marquardt method.," *IEEE Trans. on Signal Processing*, vol. 47, no. 11, pp. 3172–3175, November 1999.
[21] A. Mansour, A. Kardec Barros, and N. Ohnishi, "Comparison among three estimators for high order statistics.," in *Fifth International Conference on Neural Information Processing (ICONIP'98)*, S. Usui and T. Omori, Eds., Kitakyushu, Japan, 21-23 October 1998, pp. 899–902.
[22] A. Mansour and C. Jutten, "What should we say about the kurtosis?," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 321–322, December 1999.

Carlos G. Puntonet received a B.Sc. degree in 1982, M.Sc.degree in 1986 and his Ph.D. degree in 1994, all from the university of Granada, Spain. These degrees are in electronics physics. Currently, he is an associated Professor at the "Departamento de Arquitectura y Tecnologia de Computadors" at the university of Granada. His research interests lie in the fields of Signal Processing, Independent Component Analysis and Separation of Sources using Artificial Neural Networks.



A. Mansour received his Electronic-Electrical Engineering Diploma in 1992 from the Lebanese University (Tripoli, Lebanon), and his M.Sc. and the Ph.D. degrees in Signal, Image and Speech Processing from the Institut National Polytechnique de Grenoble - INPG (Grenoble, France) in August 1993 and January 1997, respectively. From January 1997 to July 1997, he held a post-doc position at Laboratoire de Traitement d'Images et Reconnaissance de Forme at the INPG, Grenoble, - France.
Since August 1997, he has been a Research Scientist at the Bio-Mimetic Control Research Center (BMC) at the Institut of Physical and Chemical Research (RIKEN), Nagoya, Japan. His research interests are in the areas of blind separation of sources, high-order statistics, signal processing and robotics. He is the first author of many papers published in international journals, such as IEEE Trans on Signal Processing, IEEE Signal Processing Letters, Signal Processing, NeuroComputing, IEICE Trans on Fundamentals of Electronics, Communications and Computer Sciences, Alife & Robotics. He is also the first author of many papers published in the proceedings of various international conferences.