ELSEVIER

Neurocomputing 49 (2002) 39–60

NEUROCOMPUTING

www.elsevier.com/locate/neucom

Separation of sources using simulated annealing and competitive learning

C.G. Puntonet[a,*], A. Mansour[b], C. Bauer[c], E. Lang[c]

[a] *Department of Architecture and Computer Technology, University of Granada, Granada, Spain*
[b] *Bio-Mimetic Control Research Center (RIKEN), Nagoya-shi, Japan*
[c] *Department of Biophysics, University of Regensburg, Regensburg, Germany*

Received 10 January 2001; accepted 22 October 2001

Abstract

This paper presents a new adaptive procedure for the linear and non-linear separation of signals with non-uniform, symmetrical probability distributions, based on both simulated annealing and competitive learning methods by means of a neural network, considering the properties of the vectorial spaces of sources and mixtures, and using a multiple linearization in the mixture space. The main characteristics of the method are its simplicity and the rapid convergence experimentally validated by the separation of many kinds of signals, such as speech or biomedical data. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Blind separation; Independent component analysis; Simulated annealing; Competitive learning; Neural networks

1. Introduction

The problem of linear blind separation of sources involves obtaining the signals generated by p sources, vectorially represented by $\boldsymbol{x}(t) = [x_1(t), \ldots, x_p(t)]^{\mathrm{T}}$, from the linear mixture signals, $e(t) = [e_1(t), \ldots, e_p(t)]^{\mathrm{T}}$. The mixture, normally produced in a medium or in the sensors, is characterized by an unknown and non-singular matrix $\boldsymbol{A}(t)$ such that

$$e(t) = \boldsymbol{A}(t)\boldsymbol{x}(t). \tag{1}$$

[*] Corresponding author.
E-mail address: carlos@atc.ugr.es (C.G. Puntonet).

PII: S0925-2312(02)00510-6

If the mixture is stationary, then $\boldsymbol{A}(t)$ is constant, i.e., $\boldsymbol{A}(t)=\boldsymbol{A}$. The goal traditionally sought within the context of separation of sources is to estimate $\boldsymbol{A}(t)$ by means of another matrix $\boldsymbol{W}(t)$ such that the output vector $\boldsymbol{s}(t)$ is obtained as follows:

$$\boldsymbol{s}(t)=\boldsymbol{W}^{-1}(t)e(t). \tag{2}$$

The output $\boldsymbol{s}$ coincides with the original sources, $\boldsymbol{x}$, except for a scale factor and a permutation, i.e.,

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

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

In the framework of independent component analysis, ICA, many kinds of approaches have been presented concerning the blind separation of sources, with applications to real problems in areas such 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, $\boldsymbol{x}(t)$, are mutually independent and have, at most, one Gaussian distribution [3]. The 'Infomax' or independent component analysis algorithm of Bell and Sejnowski [1] 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 [11]. Also using the maximization of the negentropy, Girolami [4] introduces an ICA 'Infomax' algorithm for unsupervised exploratory data analysis and for general linear ICA applied to electroencephalograph (EEG) monitor output. Many solutions for blind separation of sources are based on estimating a separation matrix with algorithms, adaptive or not, that use higher-order statistics, including minimization or cancellation of independent criteria by means of cost functions or a set of equations, in order to find a separation matrix [9,10]. 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 $\boldsymbol{A}$ by determining the slopes of, or any vector on, the edges that are incident on any one of the vertices of the hyperparallelepiped that contains the observation space, i.e., the independent components [13,14]. Another procedure derived in a general context of independent component analysis for separating an instantaneous mixture of sources, based on order statistics has recently been developed by Pham [12], 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.

For non-linear mixtures, a modified self-organizing map algorithm based on competitive learning has been developed by Lin and Cowan [8] to extract the local geometrical structure of distributions obtained from mixtures of statistically independent sources and to perform non-parametric histogram density estimation; this method is appropriate for sharply peaked distributions. For post-non-linear mixtures, a batch procedure based on a maximum likelihood approach has been developed by Taleb and Jutten [20]. In [15] an adaptive procedure is described for the demixing of linear and non-linear mixtures of two signals with probability distributions that are symmetric with respect to their centres, and non-uniform, performing a fixed piecewise linearization in the case of non-linear 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 by Habl et al. [5] to separate EEG signals from tumour patients into spatially independent source signals, the algorithm allowing artefactual signals to be removed from the EEG by isolating brain-related signals into single ICA components. Using an adaptive geometry-dependent ICA algorithm, it has been demonstrated in [16] the possibility of separating biomedical sources, such as EEG signals, after analysing only the observed mixing space, due to the almost symmetric probability distribution of the mixtures.

Recently, some papers are exploring the hybridation of new optimization methods or metaheuristics with classical criteria for blind source separation, by demonstrating the benefits offered in linear and non-linear mixtures with this fusion against local minima, using random elements and not computing first- or second-order derivatives, searching wide solution spaces, finding optimal or near-optimal solutions, avoiding getting trapped in suboptimal solutions, and providing a high degree of flexibility in the energy function [2,17,19]. The approach presented in this paper is intended to be valid for any number of signals and for both linear and non-linear mixtures; it 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 that is valid for all kinds of sources exhibiting a unimodal probability distribution, such as Gaussian, Laplacian, Poisson or Gamma. A new idea introduced in this paper is the hybridation of competitive learning and geometric methods, for quality and better separation, with a simulated annealing technique in order to provide fast initial learning and convergence.

The paper is organized as follows: in the next section the proposed method using competitive learning (Section 2.1), simulated annealing (Section 2.2) and both techniques simultaneously (Section 2.3) is introduced. Section 3 shows some improvements concerning time convergence and accuracy that can be obtained in the network previously presented. In Section 4, the separation matrix obtained from the network and the recursive source separation is shown, and some simulations are presented in Section 5. Finally, the conclusions are summarized in Section 6.

2. Proposed method

We propose an original method for independent component analysis and blind separation of sources that combines adaptive processing with a simulated annealing technique, and which is applied by normalizing the observed space, $e(t)$, in a set of concentric p-spheres in order to adaptively compute the slopes corresponding to the independent axes of the mixture distributions by means of an array of symmetrically distributed

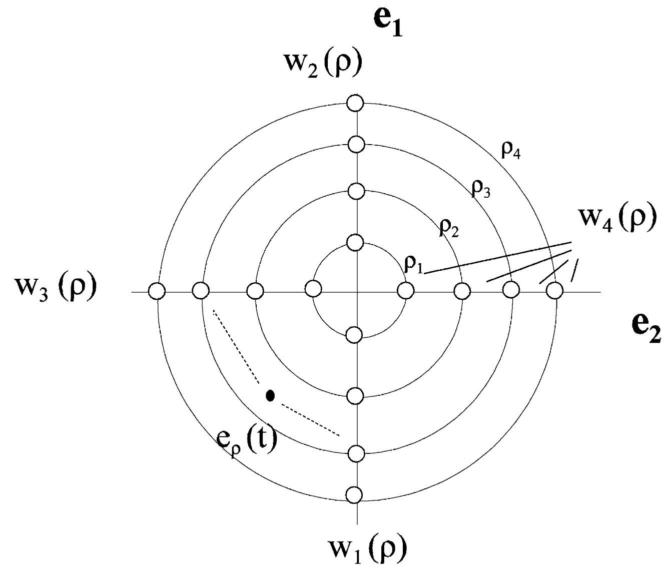

Fig. 1. Array of symmetrically distributed neurons.

neurons in each dimension (Fig. 1). A preprocessing stage to normalize the observed space is followed by the processing or learning of the neurons, which estimate the high density regions in a way similar, but not identical to that of self-organizing maps. A simulated annealing optimization 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, this being a way of improving the performance by speeding up the convergence of the algorithm. In order to work with well-conditioned signals, the observed signals $e(t)$ are preprocessed or adaptively set to zero mean, μ, and unity variance, σ, as follows:

$$e_i(t) = \frac{e_i(t) - \mu_i}{\sigma_i}, \quad i \in \{1,\ldots,p\}. \tag{4}$$

In general, for blind separation and taking into account the possible presence of non-linear mixtures, the observation space $(e_1,\ldots,e_p)$ is subsequently quantized into n spheres of dimension p (p-spheres), circles if $p=2$, each with a radius ρ_k ($k=1,\ldots,n$) covering the points as follows:

$$\rho_{k-1} < \|e(t)\| < \rho_k, \quad \rho_0 = 0 \quad \forall k \in \{1,\ldots,n\}. \tag{5}$$

The integer number of p-spheres, n, ensures accuracy in the estimation of the independent components, and it can be adjusted depending on the extreme values of the mixtures, $e(t)$, in each real problem. Obviously, the value of each radius, ρ_k, depends on the number of p-spheres, n. From now on, we shall use $e(\rho_k,t)$ to denote the vector $e(t)$ that verifies (5). If, in some applications, the mixture process is known to be linear, then the number, n, of p-spheres is set to 1, and a normalization of the space is obtained with $\rho_1 = 1$. Although the quantization given in (5) allows a piecewise linearization of the observed space for the case of non-linear and post-non-linear mixtures, it is also useful with the assumption of linear media since it allows us to detect unexpected non-linearities in some real applications [16].

2.1. Competitive learning

The above-described preprocessing is used to apply a competitive learning technique by means of a neural network whose weights are initially located on the Cartesian edges of the p-dimensional space such that the network has $2p$ neurons, with each neuron $\boldsymbol{w}_i$ being identified with p scalar weights $(w_{i1}, w_{i2}, \ldots, w_{ip})$ per p-sphere. For instance, for mixtures of two sources $(p = 2)$ and $n = 1$ then $e(t) = [e_1(t), e_2(t)]^{\mathrm{T}}$ and the network has four neurons, i.e., the neuron $\boldsymbol{w}_1$ is represented by $[w_{11}(1), w_{12}(1)]$, and the neuron $\boldsymbol{w}_2$ is represented by the weights $[w_{21}(1), w_{22}(1)]$ both neurons initially located on the e_1 edge; the neuron $\boldsymbol{w}_3$ is represented by the weights $[w_{31}(1), w_{32}(1)]$, and the neuron $\boldsymbol{w}_4$ is represented by the weights $[w_{41}(1), w_{42}(1)]$ both neurons initially located on the e_2 edge. The Euclidean distance between a point, $e(\rho_k, t)$, and the $2p$ neurons existing in the p-dimensional space (Fig. 1) is

$$d(i, \rho_k) = \|\boldsymbol{w}_i(\rho_k, t) - e(\rho_k, t)\|, \quad i \in \{1, \ldots, 2p\}, \quad k \in \{1, \ldots, n\}. \tag{6}$$

A neuron, labelled i^*, in a p-sphere ρ_k, is at a minimum distance from the p-dimensional point $e(\rho_k, t)$ and verifies:

$$d(i^*_; \rho_k) = \min\{d(i, \rho_k)\}, \quad i^* \subseteq i \in \{1, \ldots, 2p\}, \quad k \in \{1, \ldots, n\}. \tag{7}$$

The main process for competitive learning when a neuron approaches the density region, in a sphere ρ_k at time t, is given by

$$\boldsymbol{w}_i(\rho_k, t+1) = \boldsymbol{w}_i(\rho_k, t) + \alpha(t) f(e(\rho_k, t), \boldsymbol{w}_i(\rho_k, t)), \quad i \in \{1, \ldots, 2p\} \tag{8}$$

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

$$\begin{aligned} &\boldsymbol{w}_i(\rho_k, t+1) = \boldsymbol{w}_i(\rho_k, t) + \alpha(\rho_k, t)\,\mathrm{sgn}[e(\rho_k, t) - \boldsymbol{w}_i(\rho_k, t)]K_i(t), \\ &K_i(t) = \exp(-\eta^{-1}(t)\|\boldsymbol{w}_i(\rho_k, t) - \boldsymbol{w}_{i^*}(\rho_k, t)\|^2), \quad i^* \subseteq i \in \{1, \ldots, 2p\}. \end{aligned} \tag{9}$$

Here $\eta(t)$ is a neighbourhood decreasing parameter, and $\alpha(t)$ is now geometry-dependent and proportional to $\eta(t)$, as follows:

$$\alpha(\rho_k, t+1) = \eta(t)\rho_k\delta, \quad 0 < \eta(t) < 1, \quad k \in \{1, \ldots, n\}, \tag{10}$$

where δ and ρ_k 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 p-spheres, in order to equalize the angular velocity of the outer and inner weights. Note that the weight

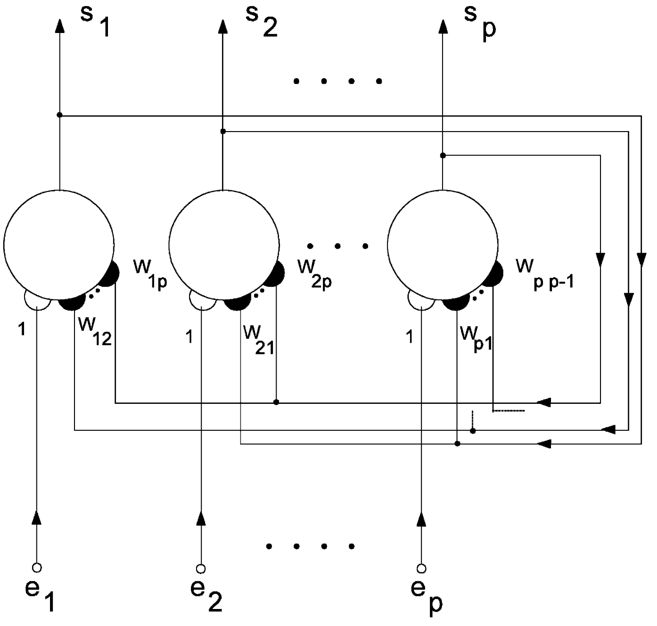

Fig. 2. Recursive neural network.

updating is carried out using the sign function, in contrast to the usual way [6]. As is well known, the term $K(t)$ modulates the learning p-sphere of jurisdiction depending on the value of $\eta(t)$. After the learning process, the weights are maintained in their respective p-spheres, ρ_k, by means of the following normalization:

$$\boldsymbol{w}_i(\rho_k,t)=\frac{\boldsymbol{w}_i(\rho_k,t)\rho_k}{\|\boldsymbol{w}_i(\rho_k,t)\|},\quad i\in\{1,\ldots,2p\},\quad k\in\{1,\ldots,n\}. \tag{11}$$

After converging, at the end of the competitive process, the weights in (11) are located at the centre of the projections of the maximum density points, or independent components, in each p-sphere. It is easy to corroborate that the total number of scalar weights w_{ij} is $2p^2n$. For the purpose of the separation of sources, a matrix, $\boldsymbol{W}$, similar to $\boldsymbol{A}$, and verifying expression (3) is needed, and a recursive neural network similar to the Herault-Jutten [7] network uses, as weights, a continuous function of the $2p^2$ scalar weights per sphere, as shown in Section 2.1, Eq. (14), and in Section 4, Eq. (31), for the general case of p sources (Fig. 2). Once the neural network has estimated the maximum density subspaces by means of an adaptive Eq. (9), and due to the piecewise linearization of the observation space with a number n of p-spheres, a set, $\boldsymbol{\Omega}$, of matrices can be defined as follows:

$$\boldsymbol{\Omega}=\{\boldsymbol{W}_{\rho_1},\ldots,\boldsymbol{W}_{\rho_n}\}, \tag{12}$$

where for p dimensions, the matrices $\boldsymbol{W}\rho_k$ have the following form:

$$\boldsymbol{W}_{\rho_k} = \begin{pmatrix} \boldsymbol{W}_{11\rho_k} & \dots & \boldsymbol{W}_{1p\rho_k} \\ \boldsymbol{W}_{p1\rho_k} & \dots & \boldsymbol{W}_{pp\rho_k} \end{pmatrix}, \quad k \in \{1,\dots,n\}. \tag{13}$$

For linear systems or “symmetric” non-linear mixtures (Simulation 1), the elements of this matrix, $\boldsymbol{W}\rho_k$, obtained using competitive learning are considered to be the symmetric slopes, in the segment of p-sphere radius ρ_k, between two consecutive weights initially located on the same axis, for each dimension j, and finally computed in (9) if the following transformation is carried out under geometric considerations:

$$\boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}(t) = \frac{\boldsymbol{w}_{2ji}(\rho_k,t) - \boldsymbol{w}_{2ji}(\rho_{k-1},t)}{\boldsymbol{w}_{2jj}(\rho_k,t) - \boldsymbol{w}_{2jj}(\rho_{k-1},t)}, \quad i,j \in \{1,\dots,p\}, \ k \in \{1,\dots,n\}. \tag{14}$$

The superscript, c, indicates that the separation matrix has been computed using competitive learning, which will be useful in Section 2.3. Note that Eq. (14) works only with even-labelled weights, $2j$, 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 neurons, $\boldsymbol{w}_2$ and $\boldsymbol{w}_4$, in the circle ρ_1. If $n > 1$, the use of several p-spheres is useful for non-linearity detection, since n different matrices, $\boldsymbol{W}\rho_k$ in (13), are obtained for successive values of ρ_k. The total number of coefficients $\boldsymbol{W}_{ij}\rho_k$ is $p(p-1)n$, since the value of the diagonal elements ($i = j$) in (14) is 1. Nevertheless, Eq. (14) is shown in this form as a particular case of the expression valid for non-linear separation of sources (Section 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 competitive learning method described above, we first propose the hybridation with a 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 is effective if the chosen energy, or cost function, E_{ij}, for the global system is appropriate. The procedure of simulated annealing is well known [18]. Firstly, it is necessary to generate random values of the weights and, secondly, to compute the associated energy of the system. This energy vanishes when the weights achieve a global minimum, the method thus allowing escape from local minima. For the problem of blind separation of sources we define an energy, E, related to the four-order statistics of the original p sources, due to the necessary hypothesis of statistical independence between them, as follows:

$$E = \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} E_{ij}(t) = \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} \langle cum_{22}^2(s_i(t)s_j(t)) \rangle, \quad i,j \in \{1,\dots,p\}, \tag{15}$$

where $cum_{22}(s_i(t), s_j(t))$ is the 2×2 fourth-order cumulant of $s_i(t)$ and $s_j(t)$, i.e.

$$cum_{22}(s_i(t), s_j(t)) = \langle s_i^2(t) s_j^2(t) \rangle - \langle s_i(t)^2 \rangle \langle s_j(t)^2 \rangle - 2\langle s_i(t) s_j(t) \rangle^2 \tag{16}$$

and $\langle x(t) \rangle$ represents the expectation of $x(t)$.

This energy can be estimated using the methods described by Mansour et al. [10]. The change in global energy, ΔE, created by the new state after the generation of random weights, is given by

$$\Delta E = E(t+1) - E(t). \tag{17}$$

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 using the equation

$$P = \mathrm{e}^{-\Delta E/T(t)}, \tag{18}$$

where $T(t)$ is the positive valued temperature at time t that regulates the search granularity for the system's 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):

$$T(t) = \frac{T_0}{1 + \eta(t)}, \tag{19}$$

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 simulated annealing algorithm has been shown above, some modifications to the procedure can be made when this method is applied to the separation of sources. For instance, we propose that the function $\eta(t)$ in (19) should be $\eta(t) = (1+t)^2 - 1$, in order to provide fast convergence. With this process, and using r_{ij} to denote a randomly chosen number in the range $[0, 1]$ for each component (i, j), a separation matrix is easily computed in each p-sphere of radius ρ_k by means of the following rule:

$$\boldsymbol{W}^{\mathrm{s}}_{ij\rho_k}(t) = 2r_{ij} - 1, \quad i, j \in \{1, \ldots, p\}, \quad i \neq j, \quad k \in \{1, \ldots, n\}. \tag{20}$$

The superscript, s, indicates that the separation matrix has been computed using simulated annealing. Note that, as in Eq. (14), the coefficients of the separation matrix in (20) with indexes $i = j$ are set to 1, and thus it is necessary to generate $p(p-1)$ random weights instead of p^2. Furthermore, the simulated annealing process is applied directly to the elements of matrix $\boldsymbol{W}\rho_k$ and does not work with the $\boldsymbol{w}_{ij\rho_k}$ neurons. Once a global minimum is obtained, when the energy in (15) vanishes, the value of the $\boldsymbol{W}\rho_k$ matrix is close to that of the original $\boldsymbol{A}$ matrix, i.e., the $\boldsymbol{W}\rho_k$ 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. Theoretically, the proposed energy function (15) depends on a four-order moment; it has been experimentally corroborated in several simulations as an estimator of statistical independence, with good results being obtained by estimating statistics over $N = 100$ samples or more. Although the use of simulated annealing does not guarantee finding the global optimum with a low number of samples, it provides a good starting point for the competitive learning process.

2.3. Competitive learning with simulated annealing

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

$$\begin{aligned}&\boldsymbol{W}_{ij\rho_k}(t+1) = \boldsymbol{W}^{\mathrm{s}}_{ij\rho_k}(t)\beta(t) + \boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}(t)(1-\beta(t)),\\ &\quad i \neq j \in \{1,\dots,p\}, \quad k \in \{1,\dots,n\},\end{aligned} \tag{21}$$

where $\beta(t)$ is a decreasing function that can be chosen in several ways (Section 3). The main purpose of the proposed equation (21) is to provide a fast initial convergence of the weights by means of simulated annealing during the epoch in which the adaptation of the neural network by competitive learning is still inactive. When the value $\beta(t)$ falls to zero, the contribution of the simulated annealing process vanishes since the random generation of weights ceases, and the more accurate estimation by means of competitive learning begins. The main contribution of simulated annealing here is the fast convergence compared to adaptation rule (9), thus obtaining an acceptable closeness of $\boldsymbol{W}\rho_k$ 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 competitive learning since, with this, the energy in (15) continues to decrease until a global minimum is obtained. The use of different approaches before the competitive learning process in order to estimate the centres of mass, as does standard K-means, is a common practice in expectation maximization fitting of Gaussians, but the complexity of this procedure and the lack of knowledge of the centroids means that simulated annealing is more appropriate, before using competitive learning.

A measure of the convergence in the computation of the independent components with the number of samples or iterations is shown in Figs. 3 and 4, which compare the competitive learning and simulated annealing methods, using the root mean square error, $\varepsilon(t)$, defined as follows:

$$\varepsilon(t) = \frac{1}{p(p-1)}\left(\sum_{i\neq j}(\boldsymbol{W}_{ij}(t) - \boldsymbol{A}_{ij}(t))^2\right)^{1/2}, \quad i,j \in \{1,\dots,p\}. \tag{22}$$

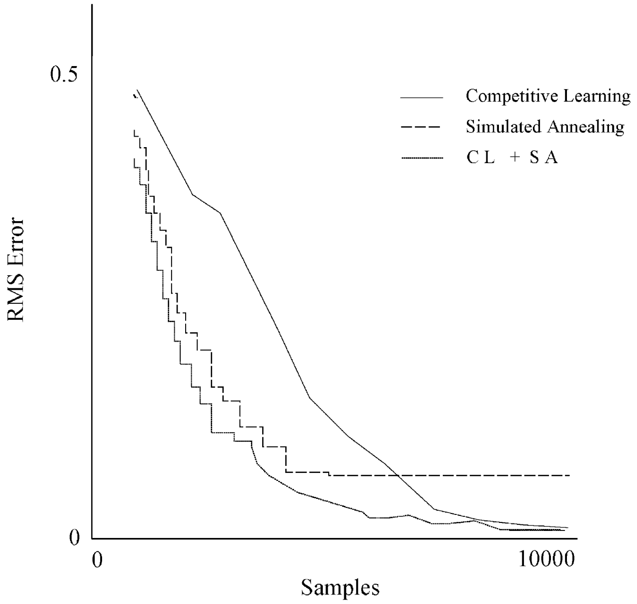

Fig. 3. Comparisons among CL and SA for $p=2$.

Note that, a priori, the unknown matrix $A(t)$ depends on time, although in the simulations it remains constant (Section 5). One parameter that provides information concerning the distribution of a signal, $x(t)$, is the kurtosis, i.e.,

$$k_x = \frac{\langle x(t)^4 \rangle - 3\langle x(t)^2 \rangle^2}{\langle x(t)^2 \rangle^2}, \tag{23}$$

where $\langle x(t) \rangle$ is the expectation of $x(t)$. Fig. 3 shows the root mean square error for linear mixtures of $p=2$ signals and $n=1$, with the two sources having kurtosis values of $\kappa_{s1}=-0.2$ and $\kappa_{s2}=0.2$, respectively, in several experiments. Using simulated annealing and 10,000 samples the error remains at $\varepsilon = 0.05$, whereas using simulated annealing and competitive learning the error becomes $\varepsilon=0.01$ with the same number of iterations. In Fig. 4, the root mean square error in the case of $p=3$ and $n=1$ is shown, the sources having kurtosis values of $\kappa_{s1}=3.1$, $\kappa_{s2}=3.5$ and $\kappa_{s3}=3.2$, respectively. With a larger number of sources to be separated, using simulated annealing and 15,000 samples the error remains at $\varepsilon = 0.06$, whereas using simulated annealing and competitive learning the error becomes $\varepsilon = 0.01$.

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.

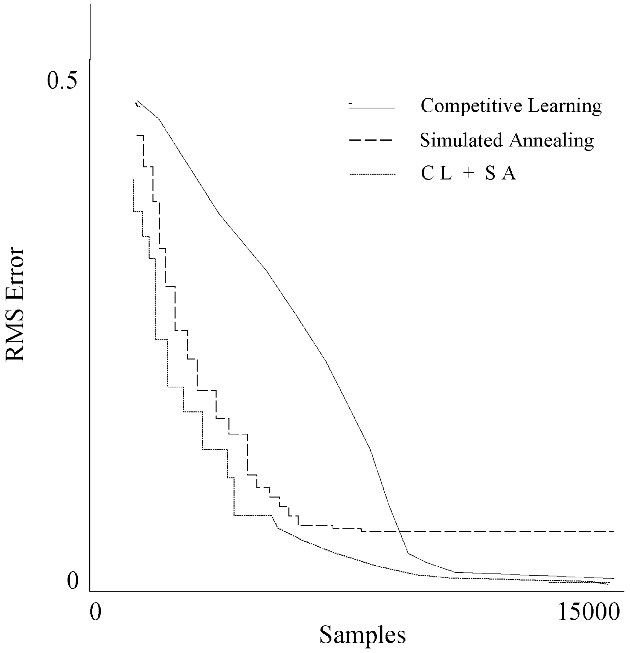

Fig. 4. Comparisons among CL and SA for $p = 3$.

3. Some improvements

The techniques presented in Section 2 can be modified to improve basic performance parameters such as time convergence and accuracy. In this section, we present some experimental improvements that are really used in the simulations, and that do not affect the basic theoretical concepts shown before. For instance, in relation to linear media, we propose eliminating some points that do not provide outstanding information, either by previous preprocessing or adaptively; this is done by means of the average correlation coefficient, computed as follows:

$$\langle c_e \rangle = \frac{1}{p(p-1)} \sum_{i,j} c_{eij}, \quad c_{eij} = \frac{1}{T} \sum_{t=1}^{T} e_i(t) e_j(t), \quad i,j \in \{1,\dots,p\}, \quad i<j \tag{24}$$

and defining a parameter δ:

$$\delta = \exp(-\langle c_e \rangle^2). \tag{25}$$

For linear mixtures, many kinds of sources, such as speech signals, contain unnecessary points near the origin that do not provide useful information when the computation of the distribution axes is being carried out; these can be removed (not processed), with

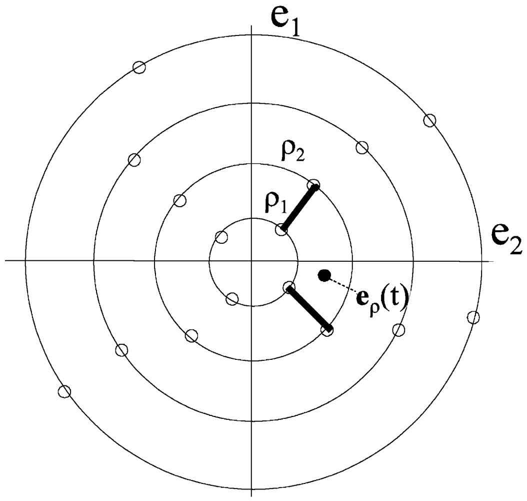

Fig. 5. Subspace associate and linear contour.

$n = 1$ in (5), if the following condition is verified:

$$\|e\| < \sum_i \sigma_i \delta = R, \quad R < \rho_1, \quad i \in \{1, \dots, p\}, \tag{26}$$

where R is the radius of the p-sphere.

Furthermore, and in order to improve time convergence in the competitive learning, Eq. (9) 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, $\boldsymbol{A}$, verifies $A_{ii} > A_{ij}$ $(i, j = 1, \dots, p)$. If this is so, only the representative winner neuron, i_q^*, is active, and it is only necessary to detect the quadrant that $e(\rho_k, t)$ belongs to.

Another fact that speeds up the learning task concerns Eq. (9) for linear or non-linear symmetrical mixtures (Figs. 5 and 6), 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:

$$\begin{aligned} \boldsymbol{w}_i(\rho_k, t+1) &= \boldsymbol{w}_i(\rho_k, t) + (-1)^{win-i} \alpha(t) \operatorname{sgn}(e(\rho_k, t) - \boldsymbol{w}_{win}(\rho_k, t)), \\ \boldsymbol{w}_j(\rho_k, t+1) &= \boldsymbol{w}_j(\rho_k, t) + (-1)^{win-j} \alpha(t) \operatorname{sgn}(e(\rho_k, t) - \boldsymbol{w}_{win}(\rho_k, t)), \\ & win \in \{i, j\}, \quad i \in \{1, 3, \dots, 2p-1\}, \; j \in \{2, 4, \dots, 2p\}. \end{aligned} \tag{27}$$

Some improvements are also feasible in the estimation of the distribution axes in non-linear mixtures, since the spatial neuron order (Fig. 6) in successive p-spheres

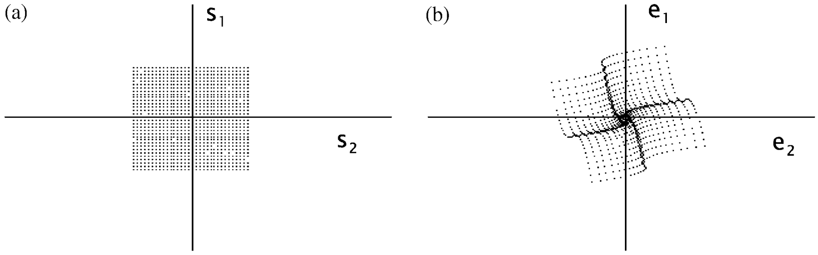

Fig. 6. (a) Space of 32-bit sources. (b) Nonlinear mixture space.

may change due to the form of the density distribution; for correct adaptive separation in Eq. (32) it is necessary to compare, periodically and for each p-sphere, the following two terms:

$$\|\mathbf{w}_i(\rho_k,t) - \mathbf{w}_i(\rho_{k-1},t)\| > \|\mathbf{w}_i(\rho_k,t) - \mathbf{w}_j(\rho_{k-1},t)\|, \quad i \neq j \in \{1,\dots,2p\}. \tag{28}$$

Once this expression is computed, the rearranging is done bottom-up, beginning from the first p-sphere, if (28) is verified. Furthermore, in linear or non-linear mixtures, the real observed signals may exhibit non-uniform density distributions (Fig. 6), and the procedure adaptively generates variable p-spheres 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 p-spheres:

$$\rho_k(\tau+1) = \rho_k(\tau) + \gamma(\lambda_{k-1}(\tau) - \lambda_k(\tau)), \quad k \in \{1,\dots,n\}, \tag{29}$$

where γ is a learning rate.

In relation to simulated annealing, the use of this technique for the blind separation problem, instead of on (18) and (19), is based on the following expressions:

$$P = \mathrm{e}^{-\Delta E/T^2(t)}, \quad T(t) = \frac{T_0}{(1+t)^2}. \tag{30}$$

Eq. (30) allows us to find a global minimum in a fast convergence time using the energy function defined in (15).

Moreover, there are several ways of implementing $\beta(t)$ in (21) in order to switch the two processes, simulated annealing and competitive learning. One of them is to use, simply, a decreasing function $\beta(t)$ similar to that of $T(t)$ in (19) or (30). Another one consists of using the competitive 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 (14) for computation of the weights, although in the general

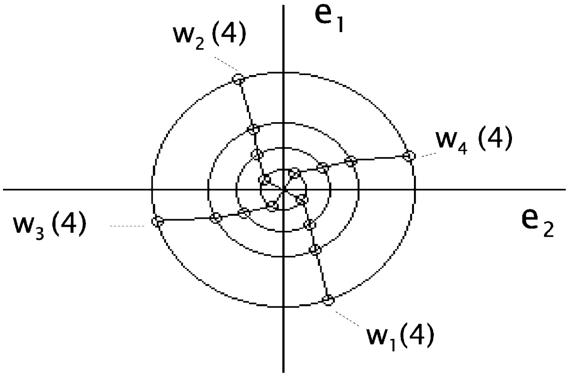

Fig. 7. Neurons configuration.

case and for pure non-linear mixtures (without symmetry at the origin), the above expression must be replaced by a similar one (Fig. 7), as follows:

$$\boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}(t)=\frac{\boldsymbol{w}_{\xi(j)i}(\rho_k,t)-\boldsymbol{w}_{\xi(j)i}(\rho_{k-1},t)}{\boldsymbol{w}_{\xi(j)j}(\rho_k,t)-\boldsymbol{w}_{\xi(j)j}(\rho_{k-1},t)},\quad i,j\in\{1,\dots,p\},\quad k\in\{1,\dots,n\},$$

$$\xi(j)\in\{\xi(1)<\xi(2)<\cdots<\xi(p)\,|\,d(\xi(j),\rho_k)<d(\xi(m),\rho_k)\},$$

$$m\in\{1,\dots,2p\},\ m\neq j. \tag{31}$$

Note that Eq. (14) is a particular case of Eq. (31), with $\xi(j)=2j$, and that the coefficients $\boldsymbol{W}^{\mathrm{c}}_{ii\rho_k}=1$ in both expressions. Eq. (31) means that the p-dimensional subspace associated to the neurons labelled $(\xi(1),\dots,\xi(p))$ around point e_ρ provides the linear contour, between the radius ρ_k and ρ_{k-1}, where the mixture can be considered linear. The number of subspaces generated by the neurons $(\xi(1),\dots,\xi(p))$ in a p-sphere is 2^p, as is the number of matrices $\boldsymbol{W}\rho_k$, and the total number of coefficients $\boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}$ in the p-dimensional space is $2^p p(p-1)n$.

By these means, we recover the sources for non-linear and post-non-linear mixtures, as well as for the linear case. For the purpose of separation, the network uses typical recursive recall, taking into account the p-sphere quantization in the observation space and the matrix computed in (21), i.e.

$$s_i(t+1)=e_i(\rho_k,t)-\sum_{j=1}^{p}\boldsymbol{W}_{ij\rho_k}(t)s_j(t),$$

$$i\in\{1,\dots,p\},\ i\neq j,\ k\in\{1,\dots,n\}. \tag{32}$$

This expression is also used by the simulated annealing process in order to compute the energy function in (15) and (16). Also note that, for non-linear mixtures, expression (32) has to be modified due to the piecewise linear approximation and for geometric reasons, since the slopes or $\boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}$ coefficients belonging from ρ_2 to ρ_n normally does not cross the origin $(e_1, e_2) = (0, 0)$ and Eq. (2) now becomes

$$\boldsymbol{s}(\boldsymbol{t}) = \boldsymbol{W}^{-1}(\rho, t)(e(\rho, t) - \boldsymbol{n}_e(\rho)), \tag{33}$$

where $\boldsymbol{n}_e(\rho)$ are the p coordinates of the slopes $\boldsymbol{W}^{\mathrm{c}}_{ij\rho_k}$ corresponding to the following equation in the plane (i, j), in each one of the 2^p subspaces:

$$e_i(\rho) = \boldsymbol{W}_{ij}(\rho)e_j(\rho, t) + \boldsymbol{n}_{e_i}(\rho), \quad i, j \in \{1, \dots, p\}. \tag{34}$$

Then, Eq. (32) can be rewritten as follows:

$$\begin{gathered} s_i(t+1) = e_i(\rho_k, t) - n_e(\rho_k, t) - \sum_{j=1}^{p} \boldsymbol{W}_{ij\rho_k}(t)s_j(t), \\ i \in \{1, \dots, p\}, \ i \neq j, \ k \in \{1, \dots, n\}. \end{gathered} \tag{35}$$

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 is defined as follows:

$$ct_i = 10 \log \left(\frac{\sum_{t=1}^{N} (s_i(t) - x_i(t))^2}{\sum_{t=1}^{N} (s_i(t))^2} \right), \quad i \in \{1, \dots, p\}. \tag{36}$$

The first simulation, Figs. 5a, b, corresponds to the synthetic non-linear mixture presented by Lin and Cowan [8], for sharply peaked distributions, the original sources being digital 32-bit signals, as follows:

$$\begin{aligned} e_1(t) &= -2\,\mathrm{sgn}[x_1(t)]x_1(t)^2 + 1.1x_1(t) - x_2(t), \\ e_2(t) &= -2\,\mathrm{sgn}[x_2(t)]x_2(t)^2 + 1.1x_2(t) + x_1(t). \end{aligned} \tag{37}$$

As shown in Fig. 6, good estimation of the density distribution is obtained with 20,000 samples, and using $n = 4$ p-spheres ($p = 2$). For the purpose of the separation, the four equation matrices obtained by means of Eq. (12), at the end of the competitive learning process, were:

$$\begin{gathered} \boldsymbol{W}_{\rho_1} = \begin{pmatrix} 1 & 1.7 \\ -1.6 & 1 \end{pmatrix}, \quad \boldsymbol{W}_{\rho_2} = \begin{pmatrix} 1 & 0.25 \\ -0.22 & 1 \end{pmatrix}, \\ \boldsymbol{W}_{\rho_3} = \begin{pmatrix} 1 & 0.2 \\ -0.22 & 1 \end{pmatrix}, \quad \boldsymbol{W}_{\rho_4} = \begin{pmatrix} 1 & 0.1 \\ -0.15 & 1 \end{pmatrix}. \end{gathered} \tag{38}$$

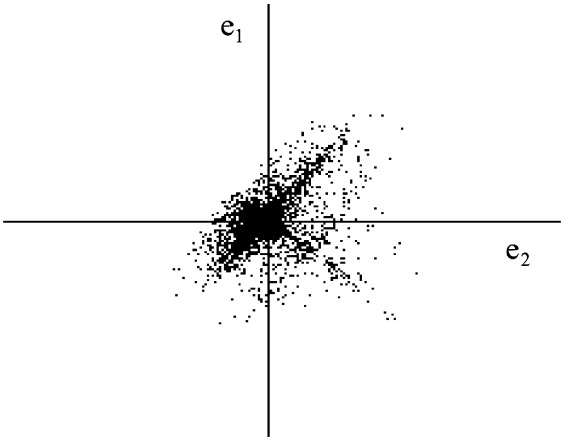

Fig. 8. Mixture space of two real sources.

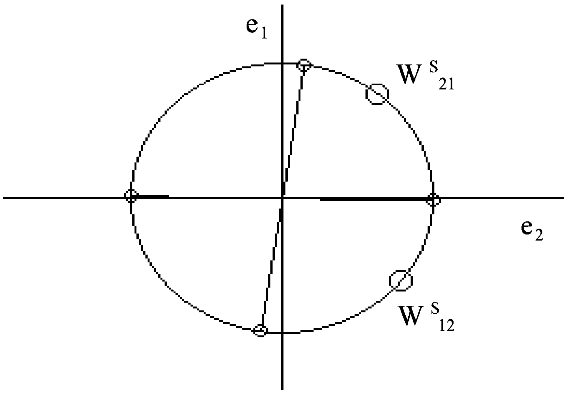

Fig. 9. Initial weights with SA.

The second simulation, shown in Figs. 8–11, concerns the separation of a mixture of two real signals, the Spanish words "dedos (fingers)" and "muñeca (doll)", captured with a 12-bit converter and presenting a signal-to-noise ratio of 24 dB. The correlation coefficient of the original sources was $\langle c_s \rangle = -0.05$, and the kurtosis value was $\kappa_{s1} = 4.7$ and $\kappa_{s2} = 4.2$ for $s_1(t)$ and $s_2(t)$, respectively. The original, $\boldsymbol{A}$, and computed, $\boldsymbol{W}\rho_1$, matrices obtained with 10,000 samples were:

$$\boldsymbol{A} = \begin{pmatrix} 1 & -0.8 \\ 0.8 & 1 \end{pmatrix}, \quad \boldsymbol{W}_{\rho_1} = \begin{pmatrix} 1 & -0.791 \\ 0.788 & 1 \end{pmatrix}. \tag{39}$$

The crosstalk parameter of the separated signals, $s_1(t)$ and $s_2(t)$, was $ct_1(t) = -24$ and $ct_2(t) = -23$ dB, respectively.

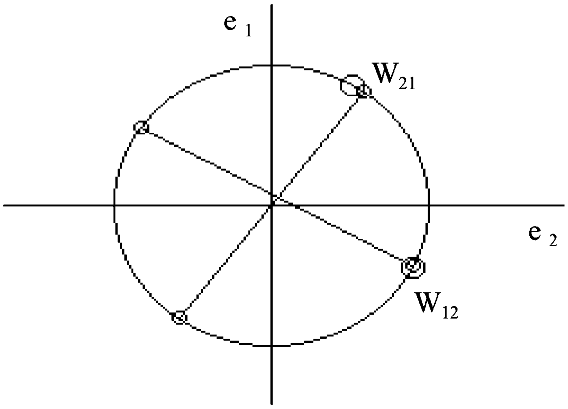

Fig. 10. Final density estimation and weight matrix.

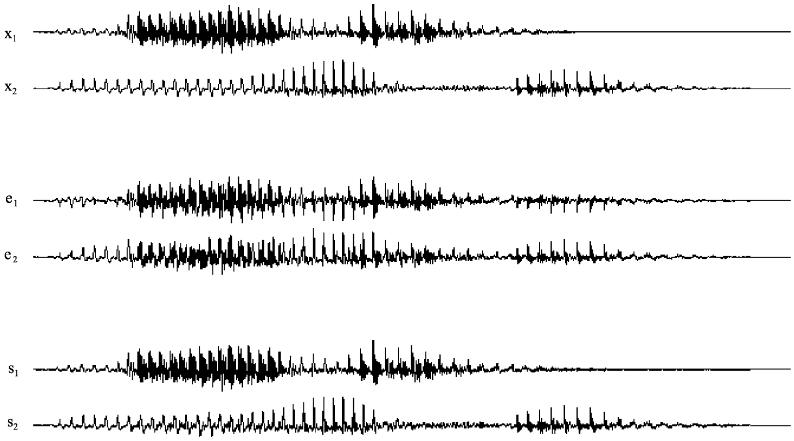

Fig. 11. Sources, observations and separated signals.

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 high density of points on the independent components speeds up convergence when the competitive learning of Eq. (9) 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 Figs. 12–15 with three synthetic supergaussian signals. Note that, although the procedure computes the weights in the three-dimensional space, Figs. 12–14 show the projection of the three-dimensional observation space (e_1, e_2, e_3) onto the (e_1, e_2) plane in order to confirm that the third components of

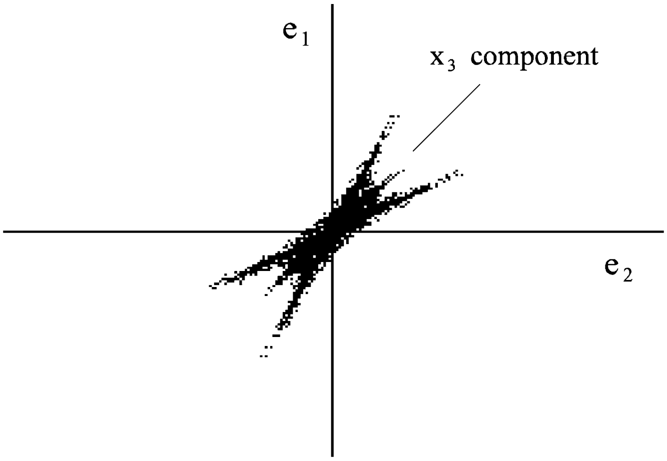

Fig. 12. Mixture of three sources projected on a plane.

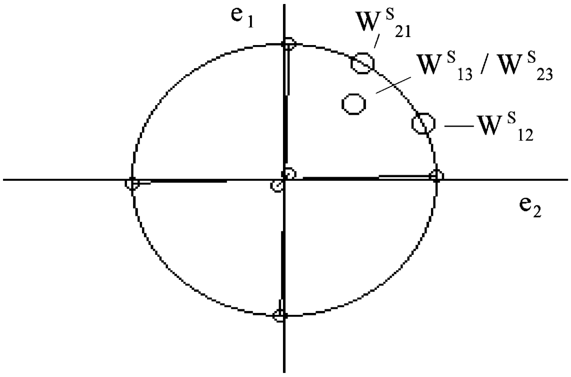

Fig. 13. Initial weights with SA.

matrix $\boldsymbol{W}\rho_k$ are correct. Therefore, the weight $\boldsymbol{w}_6$ provides, in this plane (e_1, e_2), a slope value of $+1$, corresponding to the quotient $(\boldsymbol{W}_{13}/\boldsymbol{W}_{23})$ in (14), with $(i,j)=(1,3)$ and $(2,3)$. The correlation coefficient for the original sources was $\langle c_s \rangle = -0.08$, and the kurtosis, κ_e, of the three observed signals, was $\kappa_{e1} = 3.4$, $\kappa_{e2} = 2.6$ and $\kappa_{e3} = 3.2$. The original, $\boldsymbol{A}$, and weight, $\boldsymbol{W}\rho_1$, matrices obtained with 15,000 iterations were:

$$\boldsymbol{A} = \begin{pmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{pmatrix}, \quad \boldsymbol{W}_{\rho_1} = \begin{pmatrix} 1 & 0.494 & 0.492 \\ 0.505 & 1 & 0.511 \\ 0.519 & 0.502 & 1 \end{pmatrix}. \tag{40}$$

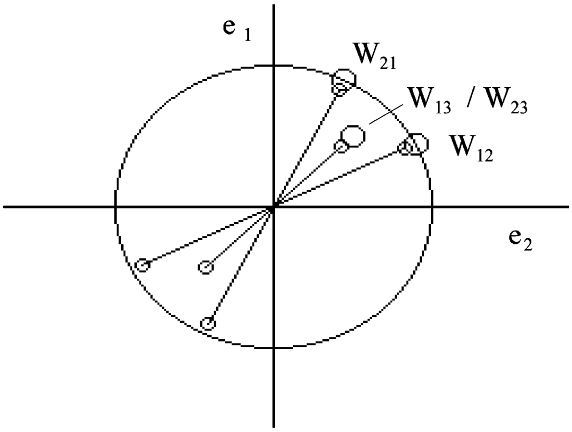

Fig. 14. Final density estimation and weight matrix.

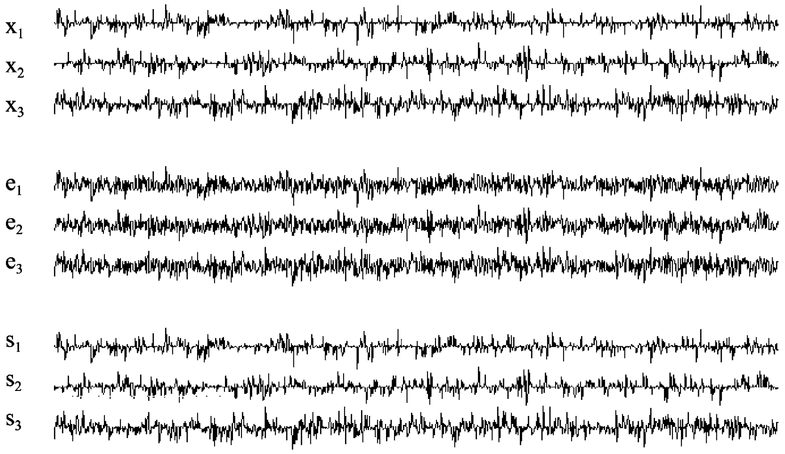

Fig. 15. Sources, observations and separated signals.

The crosstalk parameters of the three signals $s_1(t)$, $s_2(t)$ and $s_3(t)$ were $ct_1(t) = -22$ dB, $ct_2(t) = -32$ dB and $ct_3(t) = -26$ dB, respectively.

6. Conclusions

We have shown a new, powerful adaptive-geometric method based on competitive unsupervised learning and simulated annealing, which finds the distribution axes of the observed signals or independent components by means of a piecewise linearization in the mixture space, the use of one of the methods as simulated annealing in the improvement and optimization of a four-order statistical criterion being an experimental advance.

The main differences compared to papers working in the same line, such as [8], are the use of an optimization method as simulated annealing that speed up convergence, and a network with a concrete number of neurons per dimension also with geometric properties by computing the slopes of each subspace generated. We have shown how to separate the sources in the general non-linear case by means of Eqs. (31)–(35), since the computation of the sources is different when piecewise linearization is used in non-linear mixtures, the network working for linear, post-non-linear and general non-linear mixtures of $p>2$ sources with unimodal distributions.

The algorithm, in its current form, presents some drawbacks concerning the application of simulated annealing to a high number, p, of signals, and the complexity of the procedure $O(2^p p^2 n)$ for the separation of non-linear mixtures, that also depends on the number, n, of p-spheres; the finer the partition the better the separation but the more complex the learning, and perhaps to parallelize the procedure could be a solution in order to use a cluster of computers for applications with a high number of sources or with high number of p-spheres for non-linear applications.

In spite of these questions that remain open, the time convergence of the network is fast, even for more than two subgaussian or supergaussian 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 competitive learning being very precise, as several experiments have corroborated.

Besides the study of noise, future work will concern the application of this method to independent component analysis with linear and non-linear mixtures of biomedical signals, such as in electroencephalograph and functional magnetic resonance imaging, where the number of signals increases sharply, making simulated annealing suitable in a quantized high-dimensional space.

Acknowledgements

This work has been supported in part by the Spanish CICYT project TIC98-0982. The authors would like to thank the reviewers for their useful comments and suggestions. We also thank Fabian Theis for his constructive remarks.

References

[1] A.J. Bell, T.J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159.

[2] R.M. Clemente, C.G. Puntonet, J.I. Acha, A conjugate gradient method and simulated annealing for blind separation of sources, Lecture Notes in Computer Science, Vol. 2085 (Part II: Bio-Inspired Applications of Connectionism), Springer, Berlin, 2001, pp. 810–817.

[3] P. Comon, Independent component analysis, a new concept? Signal Process. 36 (1994) 287–314.

[4] M. Girolami, The latent variable data model for exploratory data analysis and visualization: a generalisation of the nonlinear infomax algorithm, Neural Process. Lett. 8 (1) (1998) 27–39.

[5] M. Habl, C. Bauer, C. Ziegaus, E.W. Lang, F. Schulmeyer, Can ICA help identify brain tumor related EEG signals? Proceedings of ICA'2000, Helsinky, Finland, 2000, pp. 609–614.

[6] S. Haykin, Neural Networks, Prentice-Hall, Englewood Cliffs, NJ, 1999.

[7] C. Jutten, J. Herault, Blind separation of sources, Part I: an adaptive algorithm based on neuromimetic structure, Signal Process. 24 (1) (1991) 1–10.

[8] J.K. Lin, J.D. Cowan, Faithful representation of separable input distributions, Neural Comput. 9 (1997) 1305–1320.

[9] A. Mansour, C. Jutten, A direct solution for blind separation of sources, IEEE Trans. Signal Process. 44 (3) (1996) 746–748.

[10] A. Mansour, A. Kardec Barros, N. Ohnishi, Comparison among three estimators for high order statistics, Fifth International Conference on Neural Information Processing (ICONIP'98), Kitakyushu, Japan, Vol. 21–23, October 1998, pp. 899–902.

[11] M.J. Mckeown, S. Makeig, G.G. Brown, T.-P. Jung, S.S. Kinderm, A.J. Bell, T.J. Sejnowski, Analysis of fMRI data by blind separation into independent spatial components, Human Brain Mapp. 6 (1998) 160–188.

[12] D.T. Pham, Blind separation of instantaneous mixtures of sources based on order statistics, IEEE Trans. Signal Process. 47 (7) (2000) 1712–1725.

[13] A. Prieto, C.G. Puntonet, B. Prieto, A neural learning algorithm for blind separation of sources based on geometric properties, Signal Process. 64 (3) (1998) 315–331.

[14] C.G. Puntonet, A. Prieto, Neural net approach for blind separation of sources based on geometric properties, Neurocomputing 18 (3) (1998) 141–164.

[15] C.G. Puntonet, M.R. Alvarez, A. Prieto, B. Prieto, Separation of Speech Signals for Nonlinear Mixtures, Lecture Notes in Computer Science, Vol. 1607 (II), Springer, Berlin, 1999, pp. 665–673.

[16] C.G. Puntonet, C. Bauer, E.W. Lang, M.R. Alvarez, B. Prieto, Adaptive-geometric methods: application to the separation of EEG signals, Proceedings of ICA'2000, Helsinky, Finland, 19–22 June 2000, pp. 273–278.

[17] I. Rojas, C.G. Puntonet, A. Canas, B. Pino, J. Fernandez, F. Rojas, Genetic algorithms for the blind separation of sources, International Conference on Artificial Intelligence and Applications (AIA'2001), Marbella, Spain, 4–7 September 2001.

[18] P.K. Simpson, Artificial Neural Systems, Pergamon Press, Oxford, 1991.

[19] J. Sole-Casals, C.G. Puntonet, I. Rojas, Simulated annealing, high-order statistics and mutual information for separation of sources, Proceedings of URSI-2001, Madrid, Spain, 19–21 September 2001.

[20] A. Taleb, C. Jutten, Source separation in postnonlinear mixtures, IEEE Trans. Signal Process. 47 (10) (1999) 2807–2820.

Carlos G. Puntonet received his B.Sc. degree in 1982, M.Sc. degree in 1986 and Ph.D. degree in 1994, all from the University of Granada, Spain. These degrees are in electronics physics. Currently, he is an Associate Professor at the "Departamento de Arquitectura y Tecnología de Computadores" at the University of Granada. His research interests lie in the fields of signal processing, linear and non-linear independent component analysis and blind separation of sources, artificial neural networks and optimization methods.

A. Mansour received his Electronic-Electrical Engineering Diploma in 1992 from the Lebanese University (Tripoli, Lebanon), and his M.Sc. and 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'lmages et Reconnaissance de Formes at the INPG. Since August 1997, he has been a Research Scientist at the Bio-Mimetic Control Research Center (BMC) at the Institute 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 and Artificial life & Robotics.

Christoph Bauer was born on the 7th of March 1973 in Regensburg. He studied Physics at the University of Regensburg and at the Trinity College Dublin (Ireland). He finished his Diploma Thesis in May 1998 at the University of Regensburg. Ph.D. studies in Regensburg and at the University of Granada (Spain). Recently finished his thesis on “Independent Component Analysis of Biomedical Signals”. His main research interests include data mining algorithms implemented on artificial neural networks.

Elmar Lang was born on the 19th of July, 1951 in Regensburg. He studied Physics at the University of Regensburg, finished his Diploma Thesis in Physics in 1977, received his Ph.D. in Physics in 1980 and finished his Habilitation in Biophysics in 1988. Currently he is an Associate Professor at the Institute of Biophysics at the University of Regensburg, heading a research group on Neuro- and Bioinformatics. His scientific interests focus on liquid state physics, NMR and Neutron Scattering, Medical Physics, Neuro- and Bioinformatics.