ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 321-326

Separation of Sources in a Class of Post-Nonlinear Mixtures

C.G. Puntonet, M.R. Alvarez, A. Prieto, B. Prieto

Departamento de Arquitectura y Tecnología de Computadores.
Universidad de Granada. 18071 Granada (Spain).
E-mail: cgpuntonet@atc.ugr.es

Abstract. This paper introduces a new approach to recover original signals from their nonlinear mixtures. The method proposed assumes that each observation is a power function of the linear combinations of the sources, and that the input distributions are bounded. We use a neural network combining the geometric approach for the adaptive computation of the coefficients of the unknown mixing matrix with the gradient descent method in order to compute the power function degree. Preliminary results obtained from experiments with synthetic and real signals are included to show the potential and limitations of the procedure.

1 Introduction

The problem of blind separation of sources involves obtaining the signals generated by p sources, s_j, j=1,...,p, from the mixtures detected by p sensors, e_i, i=1,...,p. The mixture of the signals takes place in the medium in which they are propagated, and:

$$e_i(t) = F_i(s_1(t),...,s_j(t),...,s_p(t)) \ , \quad i = 1,...,p \tag{1}$$

where F_i: $\Re^p \rightarrow \Re$ is a function of p variables from the s-space to the e-space. The goal of source separation is to obtain p functions, L_j, such that:

$$s_j(t) = L_j(e_1(t),...,e_i(t),...,e_p(t)) \ , \quad j = 1,...,p \tag{2}$$

where L_j: $\Re^p \rightarrow \Re$ is a function of p variables from the e-space to the s-space. The structure, L, represented by the set of transformations L_j, is the inverse of the set F of transformations F_i. The problem of source separation has traditionally been considered solved when, instead of obtaining the original signals $s_j(t)$, other signals $y_j(t)$ are obtained such that:

$$y_j(t) = G_j(e_1(t),...,e_i(t),...,e_p(t)) \ , \quad j = 1,...,p \tag{3}$$

where G_j: $\Re^p \rightarrow \Re$ is a function of p variables from the e-space to the y-space that generates the sources s_j, multiplied by an undefined scale factor (amplification or attenuation), and in which the indices may be permuted with respect to the original sources. Analytically these indeterminations can be represented as follows:

$$y(t) = \boldsymbol{D} \cdot \boldsymbol{P} \cdot s(t) \ ; \ y(t) = (y_j)^T \ , \ s(t) = (s_j)^T \ , \ y(t) \in \Re^p \ , \ s(t) \in \Re^p \tag{4}$$

where $\mathbf{P} \in \Re^{p\text{x}p}$ is a permutation matrix, and $\mathbf{D} \in \Re^{p\text{x}p}$ is a diagonal matrix. Most of the approaches discussed in the literature deal with linear mixtures, and so transformation

F corresponds to a linear application that can be represented by one matrix **A**=(a_{ij}) while **G** is represented by another, $\mathbf{W}^{-1}$ (which is termed *similar to* A^{-1}) such that, from (4), the following is verified:

$$W^{-1} \cdot A = D \cdot P \tag{5}$$

We have previously proposed various procedures, applicable to linear mixtures, that are based on geometrical properties of source vectors, s(t), and of mixtures, **e**(t), obtained from the hypothesis that the sources are bounded [5,6]. This restriction is plausible since the physical signals (speech, radar, biomedical, etc.) are limited in amplitude. The present paper aims to extend this method to a type of non-linear mixture that approximately models the non linearities introduced in sensors. To date, relatively few studies have been dedicated to the separation of non-linear mixtures and, using only the standard hypothesis of statistical independence of sources it is not possible to recover the original signals [9]. We believe that the model represented by (1) and (2) is too generalized and, in agreement with other authors [1,3,9], that the mixture model should be simplified; concurring with [9], we consider a post-nonlinear model (PNL), according to which signal propagation is performed via a linear transmission channel, after which the sensors introduce the non linearities. Thus, (1) may be expressed as:

$$e_i(t) = F_i(\sum_{j=1}^{p} a_{ij} \cdot s_j(t)) \quad ; \quad i = 1,...,p \tag{6}$$

There exists a great variety of sensors, whose transfer characteristics are modelled by diverse functions [4], e.g. logarithmic or power, and normal practice is to approximate these by a series expansion. Thus, if $x_i(t)$ denominates the signal captured by a sensor, its output, $e_i(t)$, may be represented by a polynomial of the form:

$$e_i(t) = F_i(x_i(t)) = c_0 + c_1 \cdot x(t) + c_2 \cdot x(t)^2 + ... + c_n \cdot x(t)^n \approx c_n \cdot x_i(t)^n \tag{7}$$

where the approximation is valid if $|x_i|>>1$. This is the parametric form considered in the present work, with $x_i=\Sigma a_{ij} \cdot s_j$. Then, the mixing model used is:

$$e_i(t) = (\sum_{j=1}^{p} a_{ij} \cdot s_j)^n \quad , \quad i,j \in \{1,...,p\} \ , \ n \in \mathbb{R}^+ - \{0\} \tag{8}$$

We also assume that the elements of the **A** matrix verify:

$$\begin{aligned} a_{ii} &\neq 0 \ , \quad \forall\, i \in \{1,...,p\} \\ a_{ij} &\geq 0 \ , \quad \forall i,j \in \{1,...,p\} \end{aligned} \tag{9}$$

This hypothesis is equivalent to considering each sensor (i) to be sensitive at least to its associated source (i), and that the influence of the other sources is not inverted.

2. Basis of procedure

In previous papers [7,8] we have shown that, if the sources are bounded, the set of all their possible values, s(t), forms a p-dimensional hyperparallelepiped in the s-space.

Similarly, if the mixture is linear, n=1 in (8), the set of all the possible images, **e**(t), forms a hyperparallelepiped in the e-space. We have also shown that by taking p vectors, $\mathbf{w}_j$, each one located at one of the edges of the hyperaparallelepiped cone that contains the mixing space, as column vectors of a matrix $\mathbf{W}=(w_{ij})$, we obtain a matrix that is similar to **A** and, thus, capable of obtaining the original sources. The problem, therefore, is reduced to that of identifying, among the mixing vectors obtained from the outputs of the sensors, those that are located at the edges of the hyperparallelepiped cone. This can be performed in various ways: for example, where $e_j(t)>0$, selecting mixing vectors for which $e_i(t)/e_j(t)$ is minimum [6]; i.e.

$$w_{ij} = \min\left\{\frac{e_i(t)}{e_j(t)}\right\} \; ; \; e_j(t)>0 \;\Rightarrow\; w_{ij} = \left(\frac{a_{ij}}{a_{jj}}\right)^n \; ; \; i,j\in\{1,..,p\} \tag{10}$$

The value of the exponent, n, is obtained from a parameter k(t), according to the iterative process described in Section 3, adaptively approaching the value of 1/n. Given the values of w_{ij} and k(t), the sources, **y**, may be obtained. Thus, for p=2 we have:

$$Y_i(k,t+1) = c_i \cdot s_i(k,t+1) = e_i(t)^k - (w_{ij}(t)\cdot e_j(t))^k \; ; \; i,j\in\{1,2\} \; , \; i\neq j \tag{11}$$

where c_i = det (**W**).

3. Linearizing the power function

From (11) we can adaptively obtain the centred signals $y_i(t)=Y_i(t)-<Y_i(t)>$, and compute the parameter k(t) using the gradient descent method. This method minimizes the cost or error function, E(t), incrementally updating the value of k(t) according to the error gradient, as follows:

$$k(t+1) = k(t) - \alpha(t) \cdot \frac{\partial E(t)}{\partial k(t)} \tag{12}$$

where α(t) is an adaptation gain which, as in [2], we believe may be obtained by using, again, the gradient method for the same error function:

$$\alpha(t+1) = \alpha(t) - \delta \cdot \frac{\partial E(t)}{\partial \alpha(t)} \tag{13}$$

where δ is the learning rate of α(t). For the error function, we propose using the correlation between signals $y_1(k,t)$ and $y_2(k,t)$, as follows:

$$E(t) = <y_1(k,t)\cdot y_2(k,t)> \;\doteq\; \iint y_1(k,t)\cdot y_2(k,t)\; dt\; dk \tag{14}$$

The objective, then, of the gradient descent procedure, is to obtain the values of k and α that minimize the correlation between $y_1(k,t)$ and $y_2(k,t)$. By deriving E(t) with respect to k, we obtain:

$$\frac{\partial E(t)}{\partial k(t)} = \int y_1(k,t)\cdot y_2(k,t)\, dt \approx \frac{1}{N}\sum_{t=1}^{N} y_1(k,t)\cdot y_2(k,t) \tag{15}$$

where the incremental form of the integral is included in (18) and N gives the number of iterations. As the error function is known, the partial derivative of E(t) with respect to α(t) can be calculated directly, as follows:

$$\frac{\partial E(t)}{\partial \alpha(t)} = \frac{\partial E(t)}{\partial k(t)} \cdot \frac{\partial k(t)}{\partial \alpha(t)} = \frac{\partial E(t)}{\partial k(t)} \cdot \beta(t) \quad ; \quad \beta(t) \doteq \frac{\partial k(t)}{\partial \alpha(t)} \tag{16}$$

Assuming that α(t) does not vary too fast, and deriving (12), we obtain:

$$\frac{\partial k(t+1)}{\partial \alpha(t)} \approx \frac{\partial k(t+1)}{\partial \alpha(t+1)} = \beta(t+1) = \beta(t) - \frac{\partial E(t)}{\partial k(t)} - \alpha(t) \cdot \frac{\partial}{\partial \alpha(t)} \left(\frac{\partial E(t)}{\partial k(t)} \right) \tag{17}$$

From (16), it is easy to see that the last partial derivative of (17) depends on the derivative of the error gradient, which can be calculated as follows:

$$\frac{\partial^2 E(t)}{\partial k^2(t)} = \frac{\partial}{\partial k(t)} \left(\frac{1}{N} \sum_{t=1}^{N} y_i(k,t)\ y_j(k,t) \right) =$$

$$\frac{1}{N} \sum_{t=1}^{N} [\, y_1(k,t) \{ e_2^k \ln e_2 - (w_{21} e_1)^k \ln(w_{21} e_1) \} + y_2(k,t) \{ e_1^k \ln e_1 - (w_{12} e_2)^k \ln(w_{12} e_2) \tag{18}$$

To sum up, k(t) can be adaptively computed by the following expressions:

$$\alpha(t+1) = \alpha(t) - \delta \,.\, \beta(t) \,.\, \frac{\partial E(t)}{\partial k(t)} \tag{19}$$

$$\beta(t+1) = \beta(t) \left(1 - \alpha(t) \,.\, \frac{\partial^2 E(t)}{\partial k^2(t)} \right) - \frac{\partial E(t)}{\partial k(t)} \tag{20}$$

together with (12), (15) and (18).

4. Simulation results

The procedure may be mapped on two simultaneous neural networks, implementing (11), (12), (15), (18), (19) and (20), in order to adaptively obtain the separated signals $y_i(k,t)$, the matrix **W** and k(t), which identifies both the medium and the sensor. Throughout the process, k(t) is adapted according to (12). This section shows the convergence of $w_{ij}(t)$, k(t), α(t) and β(t) during the process. The parameter δ is fixed at a value of 0.5. The initial value for k(t) was k(0)=1. The simulations are as follows:
Simulation 1: *n* = 10. Figure 1 shows the results of the separation of two synthetic signals, corresponding to two uniform noises, distorted with a power function of degree *n*=10. The elements of **A** were $a_{11}=a_{22}=1$ and $a_{12}=a_{21}=0.25$. The obtained weights were $w_{11}=w_{22}=1$ and $w_{12}=w_{21}=9.5.10^{-7}$; here, α was non-adaptive with value α=0.2 and, in the convergence (4000 iterations), $k^c=0.102$ and $\partial E(t)/\partial k(t)=2.5.10^{-3}$. The values of the crosstalk, c(i), for each signal i, were c(1)=-32 dB and c(2)=-28 dB.
Simulation 2: *n* = 0.1. Figure 2 shows the results of the separation of two real signals, the Spanish words "ocho" (eight) and "nueve" (nine), distorted with a power function

of degree n=0.1. The elements of **A** were $a_{11}=a_{22}=1$ and $a_{12}=a_{21}=0.5$. The obtained weights were $w_{11}=w_{22}=1$ and $w_{12}=w_{21}=0.933$; here, α was adaptive with a value in the convergence (4000 iterations) of $\alpha(t)=10$, where $k^c(t)=9.9$, $\beta(t)=0.1$, $\partial E(t)/\partial k(t)=-2.10^{-4}$ and $\partial^2 E(t)/\partial k^2(t)=5.10^{-3}$. The values of the crosstalk, c(i), for the signal i, were c(1)=-39 dB and c(2)=-45 dB.

5 Conclusions

This paper introduces an approach to separate *p* unknown sources, linearly mixed and distorted by a power function of degree *n*. The method, based on geometric considerations to determine the mixing matrix, **W**, adaptively computes the exponent of the function, $n=k^{-1}$, by means of decorrelation of the outputs, $y_i(k,t)$. An artificial neural network is simulated to recursively separate sources and to obtain, in an unsupervised way, their weights; the gradient descent method is used to adapt k(t), and simulations show that no limitation to the value of *n* exists, except $n<0$. The procedure may only be applied to media with coefficients a_{ij} that are positive, and for good separation it is necessary to obtain vectors close to the edges of the parallelepiped containing the mixing space. Future work will concern the study of other types of nonlinear mixtures and the extension of this method to more than two signals.

References

[1] G. Burel, "Blind separation of sources: a nonlinear neural algorithm", Neural networks, Vol.5, pp. 937-947, 1992.

[2] C.Jutten, A.Guerin, N.L. Nguyen Thi, "Adaptive optimization of neural Algorithms", Lecture Notes in Computer Science, Vol. 540, pp.54-61, Springer-Verlag, 1991.

[3] P.Pajunen, "Nonlinear independent component Analysis by self-organizing maps", ICANN'96, Bochum, Germany, July, 1996.

[4] P.A. Paratte; P.Robert, "Systèmes de mesure", Traité d'Electricité, Vol. 17, Presses Polytechniques et Universitaires Romandes, Lausanne, 1996.

[5] C.G.Puntonet, A.Prieto, C.Jutten, M.R.Alvarez, J.Ortega, "Separation of sources: a geometry-based procedure for reconstruction of n-valued signals". Signal Processing, vol. 46, no. 3, pp. 267-284, 1995.

[6] C.G.Puntonet, A.Prieto, "Geometric approach for blind separation of signals", Electronics Letters, Vol. 33, N°. 10, pp. 835-836, 1997.

[7] C.G.Puntonet, A.Prieto, "Neural net approach for blind separation of sources based on geometric properties". Neurocomputing, Vol. 18, 1998.

[8] A.Prieto, C.G.Puntonet,; Prieto,B., "A Neural Learning Algorithm for Blind Separation of Sources Based on Geometric Properties", Signal Processing, Vol. 64, n°. 3, 1997.

[9] A.Taleb, C.Jutten. Nonlinear source separation: The Post-Nonlinear Mixtures. In Proceedings of *ESAN'97,* pp. 279-284. D Facto Brussels, Belgium, 1997.

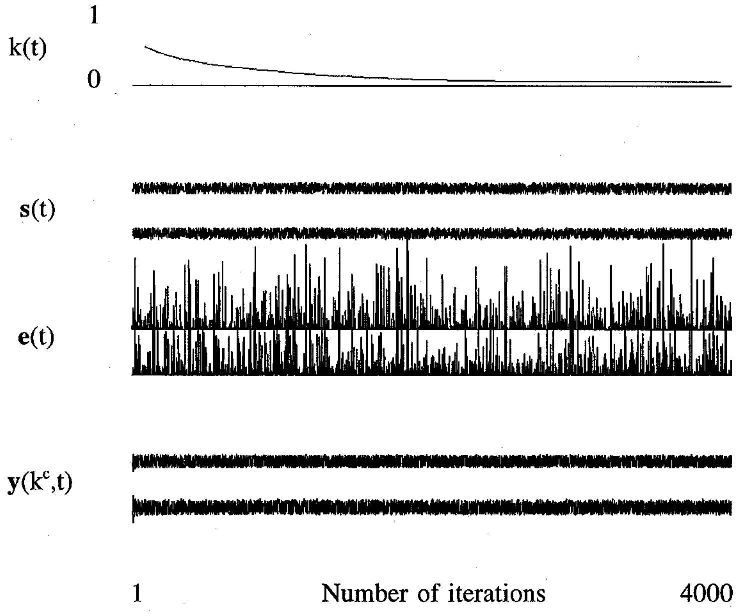

Figure 1. Simulation 1: Two random noises and *n*=10.
(a) k(t) (b) **s**(t), **e**(t) and **y**(t) when k=k^c.

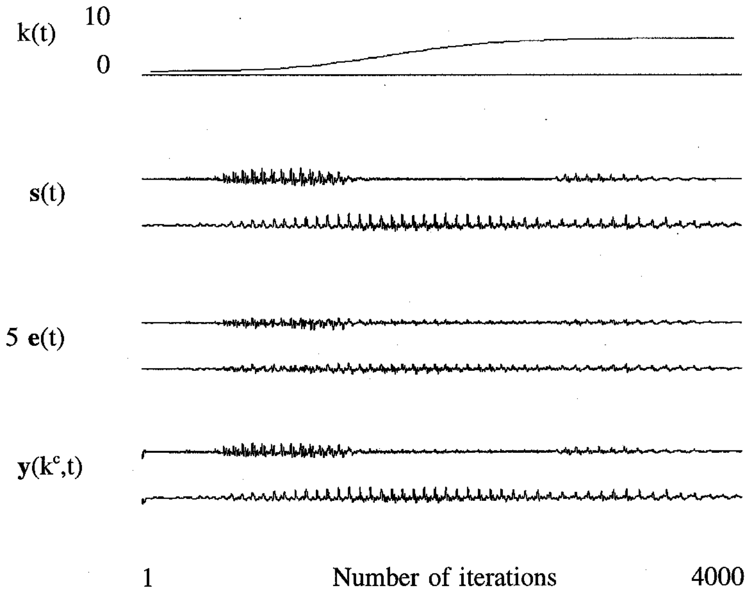

Figure 2. Simulation 2: Two real signals and *n*=0.1.
(a) k(t) (b) **s**(t), **e**(t) and **y**(t) when k=k^c.