4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003), April 2003, Nara, Japan

BLIND INVERSION OF WIENER SYSTEM USING A MINIMIZATION-PROJECTION (MP) APPROACH

Massoud BABAIE-ZADEH[1,2], *Jordi SOLÉ-CASALS*[1,3], *Christian JUTTEN*[1]

[1]Institut National Polytechnique de Grenoble (INPG), Laboratoire des images et des signaux (LIS), Grenoble, France

[2]Electrical engineering department, Sharif University of Technology, Tehran, Iran

[3]Signal Processing Group, Universitat de Vic, Sagrada Família 7, 08500, Vic, Spain
mbzadeh@yahoo.com, jordi.sole@uvic.es, Christian.Jutten@inpg.fr

ABSTRACT

In this paper, a new algorithm for blind inversion of Wiener systems is presented. The algorithm is based on minimization of mutual information of the output samples. This minimization is done through a Minimization-Projection (MP) approach, using a nonparametric "gradient" of mutual information.

1. INTRODUCTION

When linear models fail, nonlinear models appear to be powerful tools for modeling practical situations. Many researches have been done in the identification and/or the inversion of nonlinear systems. These assume that both the input and the output of the distortion are available, and are based on higher-order input/output cross-correlation [1] or on the application of the Bussgang and Prices theorems [2, 3] for nonlinear systems with Gaussian inputs. However, in a real world situation, one often does not have access to the distortion input. In this case, the blind identification of the nonlinearity becomes the only way to solve the problem. This paper is concerned by a particular class of nonlinear systems, composed by a linear subsystem followed by a memoryless nonlinear distortion (see Fig. 1 and Fig. 2). This class of nonlinear systems, also known as Wiener systems, is a nice and mathematically attracting model, but also a actual model used in various areas, such as biology [4], industry [5], sociology and psychology (see also [6] and the references therein). Despite its interest, at our knowledge, it only exists one completely blind procedure for inverting such systems [7]. The basic idea of the method is based on source separation techniques. It consists in changing the spatial independence of the outputs - required for inverting nonlinear mixtures - into a time independence of the output - required for inverting the filtered observation, *i.e.*the blind inversion of Wiener system.

This work has been partially funded by the European project Blind Source Separation and applications (BLISS, IST 1999-13077), by the French project Statistiques Avancées et Signal (SASI), by the Direcció General de Recerca de la Generalitat de Catalunya under a grant for Integrated Actions ACI2001, and by the Universitat de Vic under the grant R0912.

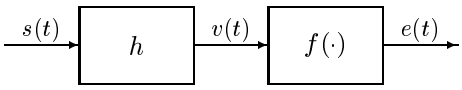

Fig. 1. A Wiener system consists of a filter followed by a distortion.

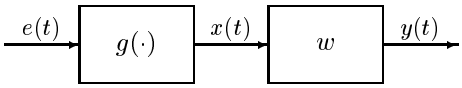

Fig. 2. A Hammerstein system consists of a distortion followed by a filter.

2. PRELIMINARY ISSUES

2.1. Mutual information

For designing a system which generates an output with independent samples, we need a criterion for measuring the independence of different samples. Recall that random variables $y_1, \ldots, y_N$ are independent, if and only if:

$$p_{\mathbf{y}}(\mathbf{y}) = \prod_{i=1}^{N} p_{y_i}(y_i). \qquad (1)$$

A convenient independence measure is mutual information of y_i's, denoted by $I(\mathbf{y})$, which is nothing but the Kullback-

Leibler divergence between $p_{\mathbf{y}}(\mathbf{y})$ and $\prod_{i=1}^{N} p_{y_i}(y_i)$:

$$\begin{aligned} I(\mathbf{y}) &= D(p_{\mathbf{y}}(\mathbf{y}) \parallel \prod_{i=1}^{N} p_{y_i}(y_i)) \\ &= \int_{\mathbf{y}} p_{\mathbf{y}}(\mathbf{y}) \ln \frac{p_{\mathbf{y}}(\mathbf{y})}{\prod_{i=1}^{N} p_{y_i}(y_i)} \mathbf{dy} \end{aligned} \tag{2}$$

It is well-known that this function is always non-negative, and vanishes if and only if the y_i's are independent. Consequently, the parameters of the inverse system (the function g and the coefficients of the inverse filter) can be found based on minimization of the mutual information of the output samples.

To do this minimization, knowing an expression for the "gradient" of the mutual information is helpful. Such an expression, which has been already proposed [8], requires multivariate score functions.

2.2. "Gradient" of mutual information

The variations of mutual information resulted from a small deviation in its argument (the "differential" of mutual information), is given by the following theorem [8]:

Theorem 1 *Let $\mathbf{\Delta}$ be a 'small' random vector, with the same dimension than the random vector $\mathbf{y}$. Then:*

$$I(\mathbf{y}+\mathbf{\Delta}) - I(\mathbf{y}) = E\left\{ \mathbf{\Delta}^T \boldsymbol{\beta}_{\mathbf{y}}(\mathbf{y}) \right\} + o(\mathbf{\Delta}) \tag{3}$$

where $o(\mathbf{\Delta})$ denotes higher order terms in $\mathbf{\Delta}$.

In this Theorem, the function $\boldsymbol{\beta}_{\mathbf{y}}(\mathbf{y})$, called Score Function Difference (SFD) [9], is defined as follows.

Definition 1 (SFD) *The score function difference (SFD) of a random vector $\mathbf{y}$ is the difference between its marginal score function $\boldsymbol{\psi}_{\mathbf{y}}(\mathbf{y})$ (MSF) and joint score function $\boldsymbol{\varphi}_{\mathbf{y}}(\mathbf{y})$ (JSF):*

$$\boldsymbol{\beta}_{\mathbf{y}}(\mathbf{y}) = \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{y}) - \boldsymbol{\varphi}_{\mathbf{y}}(\mathbf{y}) \tag{4}$$

where the marginal score function is defined by

$$\boldsymbol{\psi}_{\mathbf{y}}(\mathbf{y}) = (\psi_1(y_1), \ldots, \psi_N(y_N))^T \tag{5}$$

with

$$\psi_i(y_i) = -\frac{d}{dy_i} \ln p_{y_i}(y_i) = -\frac{p'_{y_i}(y_i)}{p_{y_i}(y_i)}. \tag{6}$$

and the joint score function is defined by

$$\boldsymbol{\varphi}_{\mathbf{y}}(\mathbf{y}) = (\varphi_1(\mathbf{y}), \ldots, \varphi_N(\mathbf{y}))^T \tag{7}$$

with

$$\varphi_i(\mathbf{y}) = -\frac{\partial}{\partial y_i} \ln p_{\mathbf{y}}(\mathbf{y}) = -\frac{\frac{\partial}{\partial y_i} p_{\mathbf{y}}(\mathbf{y})}{p_{\mathbf{y}}(\mathbf{y})} \tag{8}$$

SFD plays an important role for minimizing the mutual information. In fact, for any multivariate differentiable function $f(\mathbf{y})$, we have:

$$f(\mathbf{y}+\mathbf{\Delta}) - f(\mathbf{y}) = \mathbf{\Delta}^T \nabla f(\mathbf{y}) + o(\mathbf{\Delta}) \tag{9}$$

Then, a comparison between (3) and (9) shows that the so-called SFD can be called the *stochastic gradient* of the mutual information.

The following theorem relates the independence of the components of a random vector $\mathbf{y}$ to its SFD [10].

Theorem 2 *The components of the random vector $\mathbf{y}$ are independent, if and only if, its SFD is zero,* i.e.*:*

$$\boldsymbol{\varphi}_{\mathbf{y}}(\mathbf{y}) = \boldsymbol{\psi}_{\mathbf{y}}(\mathbf{y}) \tag{10}$$

3. INVERSION CRITERION

From the previous section, the general idea for determining the inverse system is to take the mutual information of the output samples as the inversion criterion, and then to use a gradient based algorithm for minimizing it. This gradient algorithm is based on the "gradient" of mutual information as proposed by (3).

However, using $I(y(0), y(1), y(2), \ldots)$ is computationally too expensive. This is because of the fact that using SFD as the gradient of mutual information, requires the estimation of multivariate densities, and the computational load of this estimation increases when the number of random variables increases. Consequently, we use the pairwise independence of the output samples:

$$J = \sum_{m=1}^{p} I(y(n), y(n-m)) \tag{11}$$

where p denotes the degree of separating filter. This criterion needs only the estimation of bivariate PDFs.

The criterion (11) is still expensive. For implementing it, we use a stochastic manner, that is, at each iteration we use a different random m between 1 and p. With this trick, in average, we are minimizing the criterion (11) but with less computation (note that the information in different terms of (11) are not totally independent, and hence it can be intuitively seen that this trick does not highly affect the number of required iterations for convergence). This is similar to what is done in [10] for blind separating convolutive mixtures.

4. MINIMIZATION-PROJECTION APPROACH

Recently, a Minimization-Projection (MP) approach for blind source separation has been proposed [11]. The idea of this

approach is using the following algorithm for minimizing $I(\mathbf{y})$:

$$\mathbf{y} \leftarrow \mathbf{y} - \mu\boldsymbol{\beta}_{\mathbf{y}}(\mathbf{y}) \tag{12}$$

This algorithm is similar to the steepest descent gradient algorithm for minimizing a multivariate function. It has been shown [11] that (12) converges to a random vector with independent components and has no local minimum.

However, in (12) the output is directly manipulated and after its convergence, there may be no particular relation between the outputs and the observations. To solve this problem, in each iteration, the mapping $\mathbf{e} \mapsto \mathbf{y}$ is replaced by its projection on the desired family of transformations.

The same idea can be applied in blind inversion of the Wiener system of Fig. 1, with a few modifications. Each iteration of this minimization-projection (MP) approach is composed of the following steps:

- Minimization:
  1. $y(n) \leftarrow y(n) - \mu\dfrac{\partial I(y_1(n), y_2(n-m))}{\partial y(n)}$.
- Projection:
  2. $[g_{\text{opt}}, W_{\text{opt}}(z)] = \underset{g, W(z)}{\operatorname{argmin}} E\{\big(y(n) - [W(z)]g(e(n))\big)^2\}$.
  3. $y(n) = [W_{\text{opt}}(z)]g_{\text{opt}}(e(n))$.

where $\partial I(y_1(n), y_2(n-m))/\partial y(n)$ denotes a non-parametric 'gradient' of $I(y_1(n), y_2(n-m))$ with respect to the output signal. For developing this gradient, we first define the following notation for an arbitrary signal $x(n)$:

$$\mathbf{x}^{(m)}(n) \triangleq \begin{bmatrix} x(n) \\ x(n-m) \end{bmatrix} \tag{13}$$

That is, $\mathbf{x}^{(m)}(n)$ denotes a vector composed of the signal and its delayed version with m units.

Now suppose that there is a small perturbation in the output signal:

$$\hat{y}(n) = y(n) + \epsilon(n) \tag{14}$$

where $\epsilon(n)$ stands for a 'small' signal. Consequently:

$$\hat{\mathbf{y}}^{(m)}(n) = \mathbf{y}^{(m)}(n) + \boldsymbol{\epsilon}^{(m)}(n) \tag{15}$$

Hence from Theorem 1 we have (up to first order terms):

$$\begin{aligned} I(\hat{\mathbf{y}}^{(m)}(n)) - I(\mathbf{y}^{(m)}(n)) &= E\left\{\boldsymbol{\beta}^\star(n)^T\boldsymbol{\epsilon}^{(m)}(n)\right\} \\ &= E\left\{\beta_1^\star(n)\epsilon(n)\right\} + E\left\{\beta_2^\star(n)\epsilon(n-m)\right\} \\ &= E\left\{\beta_1^\star(n)\epsilon(n)\right\} + E\left\{\beta_2^\star(n+m)\epsilon(n)\right\} \\ &= E\left\{\big(\beta_1^\star(n) + \beta_2^\star(n+m)\big)\epsilon(n)\right\} \end{aligned} \tag{16}$$

where $\boldsymbol{\beta}^\star(n) = (\beta_1^\star(n), \beta_2^\star(n))^T \triangleq \boldsymbol{\beta}_{\mathbf{y}^{(m)}}(\mathbf{y}^{(m)}(n))$. Note that in the above simplification we have assumed that the signals are stationary. This equation shows that the desired gradient is:

$$\frac{\partial I(y(n), y(n-m))}{\partial y(n)} = \beta_1^\star(n) + \beta_2^\star(n+m) \tag{17}$$

In other words, moving in the opposite direction of the above gradient (*i.e.* choosing $\epsilon(n) = -\mu\frac{\partial I(y(n), y(n-m))}{\partial y(n)}$) insures a reduction in $I(y(n), y(n-m))$.

5. CALCULATING THE PROJECTED MAPPING

Using the approach of previous section needs, in step 2, the calculation of the function g and of the filter $W(z) = w_0 + w_1 z^{-1} + \cdots + w_p z^{-p}$ which minimize the error:

$$E\{\big(y(n) - [W(z)]g(e(n))\big)^2\} \tag{18}$$

In this section, we propose an iterative algorithm for determining this optimal system, inspired from a similar method used in blind source separation of Post Non-Linear (PNL) mixtures [11].

First, suppose that g and hence $x(n)$ are already known, and we are looking for $W(z)$ which minimizes:

$$E\{\big(y(n) - [W(z)]x(n)\big)^2\} \tag{19}$$

From the orthogonality principle [12], this error is minimized if and only if:

$$E\left\{(y(n) - [W(z)]x(n))x(n-k)\right\} = 0, \quad k = 0, \ldots, p \tag{20}$$

i.e.:

$$\sum_{i=0}^{p} w_i E\left\{x(n-i)x(n-k)\right\} = E\left\{y(n)x(n-k)\right\} \tag{21}$$

for all $0 \le k \le p$. By defining:

$$r_{xx}(i,k) \triangleq E\left\{x(n-i)x(n-k)\right\} \tag{22}$$

$$r_{yx}(k) \triangleq E\left\{y(n)x(n-k)\right\} \tag{23}$$

we have:

$$\sum_{i=0}^{p} r_{xx}(i,k)w_i = r_{yx}(k), \quad k = 0, \ldots, p \tag{24}$$

Then, by defining:

$$\mathbf{R}_{xx} \triangleq \begin{bmatrix} r_{xx}(0,0) & r_{xx}(1,0) & \cdots & r_{xx}(p,0) \\ r_{xx}(0,1) & r_{xx}(1,1) & \cdots & r_{xx}(p,1) \\ \vdots & \vdots & \ddots & \vdots \\ r_{xx}(0,p) & r_{xx}(1,p) & \cdots & r_{xx}(p,p) \end{bmatrix} \tag{25}$$

$$\mathbf{r}_{yx} \triangleq \begin{bmatrix} r_{yx}(0) & r_{yx}(1) & \cdots & r_{yx}(p) \end{bmatrix}^T \tag{26}$$

- Initialization: $x = e$.
- Loop:
  1. Obtain $W(z)$ from (27).
  2. Let $x(n) = [W(z)]^{-1} y(n)$.
  3. Change the order of $x(k)$ such that the function $x = g(e)$ become ascending (see the text).
- Repeat until convergence

Fig. 3. The projection algorithm.

Eq. (24) can be written:

$$\mathbf{R}_{xx}\mathbf{w} = \mathbf{r}_{yx} \tag{27}$$

where $\mathbf{w} = (w_0, w_1, \ldots, w_p)^T$. The above equation determines the optimal filter $W(z)$.

Therefore, by knowing g, we can determine $W(z)$ from (27). But g is not known and must be determined. Note that this function must be invertible and hence monotonous. Here, without loss of generality, we assume that it is ascending. Now, we can propose the following iterative algorithm for determining g:

First assume an initial value for g. This determines $x(n)$ and the optimal $W(z)$. By knowing $W(z)$ and $y(n)$ we can recalculate $x(n)$. The values of $x(n)$ and $e(n)$ automatically define a function g. This function is not necessarily ascending. If it is not ascending, the order of values of $x(n)$ will be changed such that for every $e(n_1) < e(n_2)$ we have $x(n_1) < x(n_2)$. In MATLAB this can be done by the instructions:

```
[temp, index] = sort(e);
x(index) = sort(x);
```

This new x determines a new g and the above procedure is repeated until convergence. Figure 3 shows the final projection algorithm.

It must be noted that we are using a steepest descent iterative algorithm and our initial value for g is ascending. Moreover, at each iteration, only a small modification is done in the values of $x(n)$. Consequently, the above time permutation does not result in a huge modification of the estimated g; it must be seen as a manner for preventing the algorithm to produce a non-ascending g.

Another remark is needed for the step 2 of the algorithm. This step can be done by the recursive equation:

$$x(n) = \frac{1}{w_0}\big(y(n) - w_1 x(n-1) - \cdots w_p x(n-p)\big) \tag{28}$$

However, the filter obtained in (27) may be non-minimum phase, and hence non-invertible. To overcome this difficulty, one may neglect the obtained $W(z)$ in the current iteration, or replace it with the minimum phase version of $W(z)$ (*e.g.* using MATLAB function `polystab`).

Finally, we note that the expected values in the above equations must be estimated. In practice, this is done by empirical averaging.

6. THE ALGORITHM

Having a method for calculating the projected mapping (step 2 of the approach presented in Section 4), the inversion algorithm is evident.

However, some modifications are needed. First, it must be noted that there are mean and scale indeterminacies in both $x(n)$ and $y(n)$ signals. Consequently, for removing their effects, in each iteration the means of $x(n)$ and $y(n)$ are removed and their energies are normalized.

Moreover, two modifications must be done in applying the algorithm of Fig. 3. First, instead of initializing by $x(n) = e(n)$, we can use the value of $x(n)$ obtained in the previous iteration of MP algorithm which is a better initial estimation of $x(n)$. Secondly, we do not wait for the projection algorithm to converge (even, it is possible that it does not converge, when the output cannot expressed as $[W(z)]g(e(n))$). Instead, we simply repeat the loop for some fixed number of iterations, say 5 or 10 (in fact, even 1 iteration is sufficient in many cases, because the whole algorithm is itself iterative, and we use the value of $x(n)$ in the previous iteration for its initial value in this iteration).

Another important point is the initialization of the global MP algorithm, that is, the determination of output for the first iteration. For doing it, we use the approach proposed in [13]. The idea of this approach is as follows:

1. Because of the central limit theorem [12], v tends to have a Gaussian distribution, which is distorted by the nonlinear function f. As an initial estimate for g, we use a nonlinear function which creates a Gaussian x. It can be seen [13] that such a function is given by:

   $$g = \Phi^{-1} \circ F_e \tag{29}$$

   where F_e is the Cumulative Density Function (CDF) of e and Φ^{-1} is the CDF of a (zero mean and unit variance) Gaussian distribution.

2. Since we are looking for an output with independent samples, as an initial estimate for the filter $W(z)$ a filter which creates output with *decorrelated* samples may be used. Such a filter is given by the Linear Predictor Coefficients (LPC) of the x sequence. In MATLAB it can be obtained by the `lpc` function.

The final inversion algorithm is given in Fig. 4. In this figure, p denotes the order of the inverse filter, and K (the number of repetitions of the internal loop) is a small number (1 to 10).

- Initialization: $g = \Phi^{-1} \circ F_e$, $x = g(e)$, $W(z) = lpc(x, p)$, $y(n) = [W(z)]x(n)$.
- Loop:
  1. Choose a random $1 \leq m \leq p$.
  2. Estimate the SFD of $\mathbf{y}^{(m)}$ (as defined in (13)).
  3. Modify the output by (see (17)):
$$y(n) \leftarrow y(n) - \mu \frac{\partial I(y_1(n), y_2(n-m))}{\partial y(n)}$$
  4. For $k = 1, \ldots, K$, do:
     (a) Find $W(z)$ from (27).
     (b) Stabilize $W(z)$.
     (c) Compute $x(n) = [W(z)]^{-1} y(n)$.
     (d) Change the order of $x(k)$ such that the function $x = g(e)$ be ascending.
  5. Remove the mean of x and normalize its energy.
  6. Let $y(n) = [W(z)]\, g\big(e(n)\big)$.
  7. Remove the mean of y and normalize its energy.
- Repeat until convergence

Fig. 4. MP algorithm for blind inversion of Wiener systems.

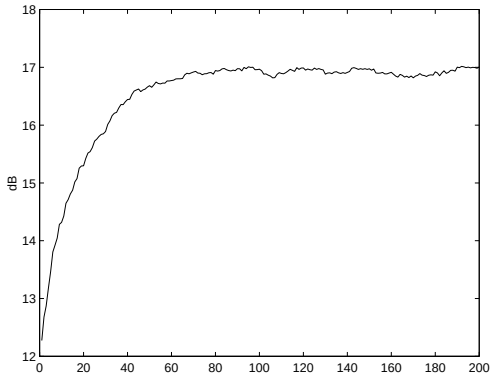

Fig. 5. Averaged output SNRs versus iteration.

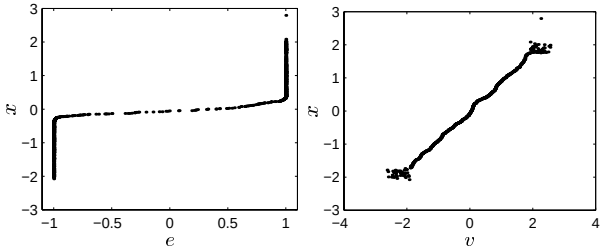

Fig. 6. Left) x versus e, Right) x versus v.

7. EXPERIMENTAL RESULTS

Here, for checking the efficacy of the proposed algorithm, we present an experimental result using uniform random sources and a very hard nonlinear distortion.

In this experiment, the source signal is a uniform random source with zero mean and unit variance. The filter h is the low-pass filter $H(z) = 1 + 0.5z^{-1} - 0.2z^{-2}$, and the nonlinear distortion function is $f(x) = \tanh(10x)$. Then, the algorithm of Fig. 4 is used to obtain the inverse system. The parameters of the algorithm are: $T = 1000$ (number of observed samples), $p = 15$ (order of inverse filter), $\mu = 0.2$ (step size) and $K = 1$ (number of repetition of the internal loop). For estimating the SFD, a method proposed by D.-T. Pham for estimating conditional score functions is used [14].

As performance criterion, we have used the output Signal to Noise Ratio (SNR), defined by:

$$\text{SNR (in dB)} = 10 \log_{10} \frac{E\left\{s^2\right\}}{E\left\{(y-s)^2\right\}} \qquad (30)$$

Figure 5 shows the averaged output SNR's versus iteration, which is taken over 10 runs of the simulation. Also, the distribution of x samples versus e and v samples is shown in Fig. 6. These distributions show the estimated g and the compensated function $g \circ f$. It can be seen that the nonlinearity is not very well compensated at the edges, which is not so important, because there is not a large number of data points in this region (as proved by the value of SNR).

Finally, Figures 7 and 8 show the coefficients of the filters $W(z)$ and $W(z)H(z)$, respectively. Especially, Figure 7 almost reduces to a Dirac: it then shows that the inverse of the linear part is well estimated.

This experiment shows that although the nonlinear distortion is very hard, the algorithm has been capable in estimating the inverse system.

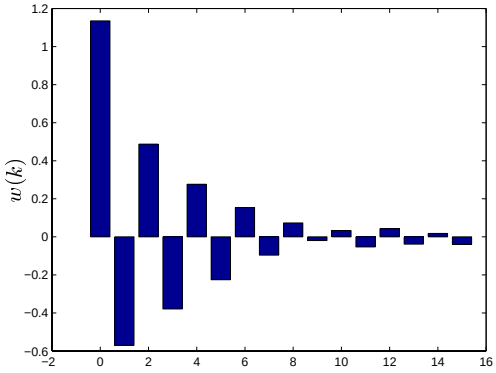

Fig. 7. The coefficients of the estimated inverse filter $w(k)$.

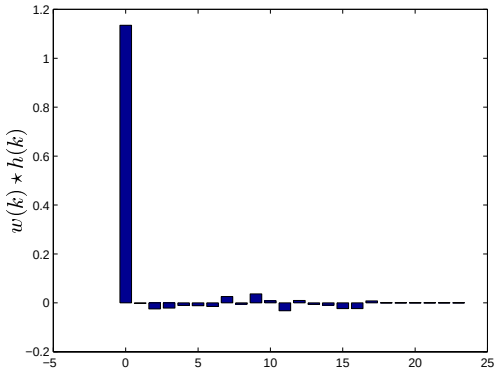

Fig. 8. The coefficients of the global filter $w(k) \star h(k)$.

8. CONCLUSION

In this paper, a new method for blind inversion of wiener systems is proposed. This method is a Minimization-Projection approach for minimizing mutual information of the output samples, which is inspired from a similar approach for blind source separation [11]. The experimental results shows the efficacy of the developed algorithm, even for very hard nonlinear distortions.

9. REFERENCES

[1] S.A. Bellings and S.Y. Fakhouri, "Identification of a class of nonlinear systems using correlation analysis," *Proc. IEEE*, vol. 66, pp. 691–697, 1978.

[2] E.D. Boer, "Cross-correlation function of a bandpass nonlinear network," *Proc. IEEE*, vol. 64, pp. 1443–1444, 1976.

[3] Jacoviti G., A. Neri, and R. Cusani, "Methods for estimating the autocorrelation function of complex stationary process," *IEEE Trans. ASSP*, 1987.

[4] I.W. Hunter, "Frog muscle fibber dynamic stiffness determined using nonlinear system identification techniques," *Biophys. J.*, pp. 49–81, 1985.

[5] R. Bars, I. Bzi, B. Pilipar, and B. Ojhelyi, "Nonlinear and long range control of a distillation pilot plant. in identification and syst. parameter estimation," in *EUSIPCO*, Budapest, 1990, pp. 848–853.

[6] I.W. Hunter and M.J. Korenberg Korenberg, "The identification of nonlinear biological systems: Wiener and hammerstein cascade models," *Biol Cybern.*, pp. 135–144, 1985.

[7] A. Taleb, J. Solé-Casals, and C. Jutten, "Quasi-nonparametric blind inversion of Wiener systems," *IEEE Tr. on Signal Processing*, vol. 49, no. 5, pp. 917–924, 2001.

[8] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "Differential of mutual information function," *IEEE Signal Processing Letters*, 2002, submitted.

[9] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "Blind separating Convolutive Post-Nonlinear mixtures," in *ICA2001*, San Diego, California, December 2001, pp. 138–143.

[10] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "Separating convolutive mixtures by mutual information minimization," in *Proceedings of IWANN'2001*, Granada, Spain, Juin 2001, pp. 834–842.

[11] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "Minimization-projection (MP) approach for blind source separation in different mixing models," in *ICA2003*, April 2003, submitted.

[12] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 1991.

[13] J. Solé-casals, M. Babaie-Zadeh, C. Jutten, and D.-T. Pham, "Improving algorithm speed in post nonlinear mixtures and wiener systems inversion," in *ICA2003*, April 2003, submitted.

[14] D. T. Pham, "Estimation de la fonction score conditionnelle et l'entropie conditionnelle," Tech. Rep., 2002, (in French).