SOURCE SEPARATION TECHNIQUES APPLIED TO LINEAR PREDICTION

Jordi Solé i Casals (1), Christian Jutten(2), Anisse Taleb (2)

(1)Department of Signal Theory and Communications
University of Vic, Sagrada Família 7, 08500, Vic (Catalunya, Spain)
(2)INPG-LIS, 46 Av. Félix Viallet, 38031, Grenoble Cedex, France

ABSTRACT

The prediction filters are well known models for signal estimation, in communications, control and many others areas. The classical method for deriving linear prediction coding (LPC) filters is often based on the minimization of a mean square error (MSE). Consequently, second order statistics are only required, but the estimation is only optimal if the residue is independent and identically distributed (iid) Gaussian. In this paper, we derive the ML estimate of the prediction filter. Relationships with robust estimation of auto-regressive (AR) processes, with blind deconvolution and with source separation based on mutual information minimization are then detailed. The algorithm, based on the minimization of a high-order statistics criterion, uses on-line estimation of the residue statistics. Experimental results emphasize on the interest of this approach.

1. INTRODUCTION

The prediction filters are well known models for signal estimation or modelisation, in communications, speech processing, control and many others areas. The classical method for estimating linear prediction coding (LPC) [1] filters consists in minimizing of a mean square error (MSE). As a consequence, the method is very simple because second order statistics are only required, but the estimation is only optimal if the residue is independent and identically distributed (iid) Gaussian.

However, if the residue is not Gaussian, the estimation is no longer optimal. If one knows the theoretical statistics, it is possible to improve the estimation by using optimal (higher order) statistics. Otherwise, *i.e.* if the statistics is not known, one can wonder how to implementing a quasi-optimal estimation.

This paper is organized as follows. In Section 2, we derive the maximum likelihood (ML) estimate of LPC and show that it only coincides to the classical method in the Gaussian case. In Section 3, we compute the ML estimate in the general case, which clearly involves the score functions. In section 4, we show the relationships with blind deconvolution and recent advances in source separation, which inspire a new, quasi-optimal LPC algorithm. Section 5 is devoted to experiments and comparisons between the new and classical LPC algorithms. Finally, the major results and outline of future works are summarized in the conclusion.

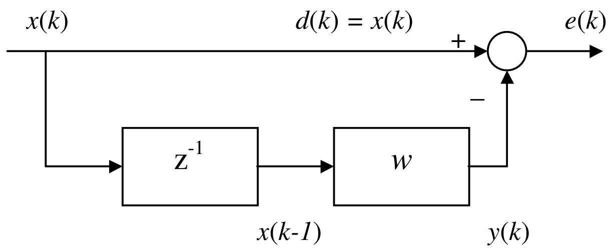

Figure 1. Block diagram linear prediction coding system.

2. CLASSICAL LPC

The classical LPC methods are based on the minimization of a mean square error, defined as the difference between the input signal $x(k)$ and the predicted signal $y(k)=[w(z)]x(k-1)$, where $w(z)$ is a L-th order causal finite impulse response filter, *i.e.* a filter whose entries $w_i = 0$ for $i \notin \langle 0,\dots,L-1\rangle$. The block diagram of a linear predictor is shown in Fig. 1.

From Fig. 1, it is easy to derive the cost function to be minimized:

$$J = E[e^2(k)] = E[(x(k) - y(k))^2]$$
$$= E[x^2(k)] - 2\sum_{n=0}^{L-1} w_n E[x(k)x(k-n-1] \quad (1)$$
$$+ \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} w_m w_n E[x(k-m-1)x(k-n-1]$$

Denoting $E[x(k)x(k-l)] = R_{xx}(l)$, the cost function reduces to :

$$J = R_{xx}(0) - 2\sum_{n=0}^{L-1} w_n R_{xx}(n+1) + \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} w_m w_n R_{xx}(m-n) \quad (2)$$

which is the classical expression of the LPC criterion. Minimizing this function with respect to the filter entries provides the classical LS linear predictor.

This estimation can be viewed as a maximum likelihood (ML) estimate in the special case of independent and identically distributed (i.i.d.) Gaussian error. In fact, first consider only the prediction at time k. Taking into account the relation $y(k) = x(k) - e(k)$, and denoting $p_E(.)$ the probability density function (pdf) of the residue $e(k)$, the likelihood of the estimation is :

$$p(y(k)/x(k), \mathbf{w}) = p_E(x(k) - y(k)). \quad (3)$$

where **w** denotes the parameter vector.
Now consider the prediction done using N successive samples. Assuming that the errors $e(k)$ are iid, and using the Bayes theorem, the likelihood of the N samples is:

$$\begin{aligned} & p(y/x(k), x(k-1), \ldots, x(k-N+1), \mathbf{w}) \\ & = \prod_{i=0}^{N-1} p(y(k+i)/x(k-i), \mathbf{w}) \\ & = \prod_{i=0}^{N-1} p_E(x(k+i) - y(k+i)) \\ & = \prod p_E(e(k+i)) \end{aligned} \quad (4)$$

Taking the natural logarithm, the ML estimate is then:

$$ArgMax_{\mathbf{w}}\left[\sum_{i=à}^{N-1} \ln(p_E(e(k+i))\right]. \quad (5)$$

Assuming that the error $e(k)$ is a Gaussian zero mean random variable:

$$p_E(e(k+i)) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{e^2(k+i)}{2\sigma^2}\right), \quad (6)$$

the Maximum Likelihood estimation is:

$$ArgMin_{\mathbf{w}}\left[\sum_{i=0}^{N-1}(e(k+i))^2\right]. \quad (7)$$

As it is well known, in the Gaussian case, asymptotically, the ML is nothing but the minimum mean square error (MMSE) estimate.

3. HIGHER ORDER METHOD

Unfortunately, if the error is not Gaussian, the MMSE estimate is no longer equal to the ML estimate. In fact, from (5), one can compute the ML equation by deriving the equation with respect to the entries w_j :

$$\begin{aligned} \sum_{i=0}^{N-1} \frac{\partial}{dw_j} \ln(p_E(e(k+i))) &= \sum_{i=0}^{N-1} \frac{p'_E}{p_E}(e(k+i))\frac{\partial e(k+i)}{\partial w_j} \\ &= -\sum_{i=0}^{N-1} \psi_E(e(k+i))x(k+i-j-1) \end{aligned}$$

where $\psi_E(.)$ denotes the derivative of $\ln p_E(.)$, the so-called score function. Consequently, asymptotically, for any error distribution, the ML estimate of w_j, $j = 0, \ldots, L-1$, is equivalent to the equation set:

$$E[\psi_E(e(k))x(k-j-1)] = 0, \quad j = 0, \ldots, L-1. \quad (8)$$

Basically, the score function is a nonlinear function, except in the Gaussian case. Then, equation (8) prove that the optimal ML estimate involves higher (than 2) order statistics, except in the Gaussian case. Implementation of equations (8) suggests two questions :

- How is it possible to estimate the actual statistics of the residue, which is generally unknown ?
- What performance gain can be obtained with the optimal criterion ?

4. LPC, DECONVOLUTION AND SOURCE SEPARATION

Before to address these questions, we emphasize in this section on the relationships between LPC, blind deconvolution and source separation.

LPC is based on the assumption that the signal $x(k)$ is linear, *i.e.* the linear auto-regressive (AR) filtering of an iid sequence $n(k)$:

$$H(z)x(k) = n(k) \quad (9)$$

where $H(z) = 1 + h_1 z^{-1} + \cdots + h_K z^{-K}$.

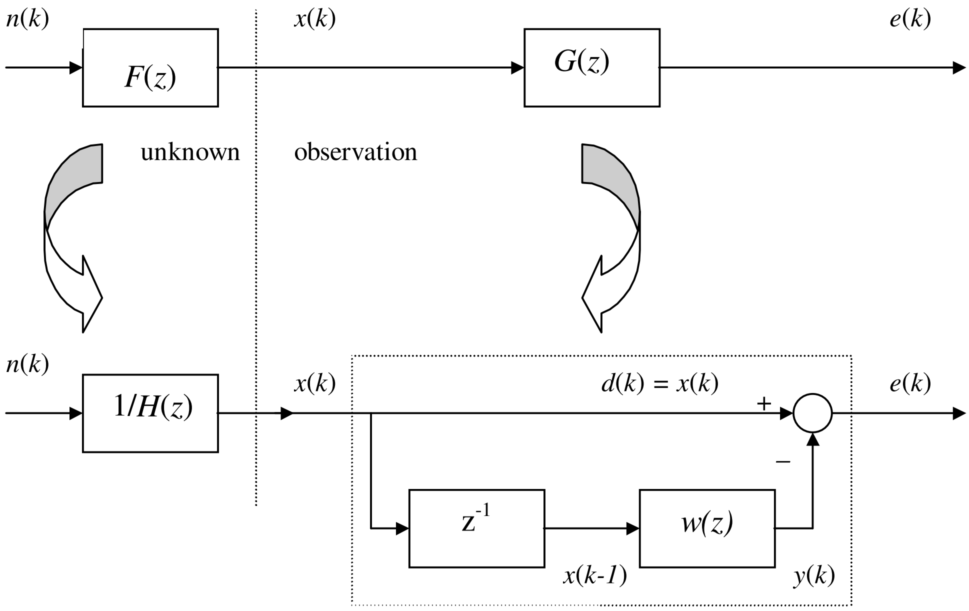

Figure 2. On the top, the convolution system $F(z)$ and the deconvolution system $G(z)$. The recovered signal $y(k)$ is $s(k)$ iff $G(z)F(z)$=1. On the down, the classical LPC viewed as a deconvolution problem: $1/H(z)$ is the unknown filter and the dashed bloc its inverse, to be estimated, in order to recover $e(k)$=$n(k)$.

It is clear that the linear prediction of $x(k)$ is nothing but the deconvolution of $x(k)$, with a filter with the constrained structure of Fig. 1, and that the optimal solution should verify:

$$1 - z^{-1}w(z) = H(z) \tag{10}$$

as shown in Fig. 2.If $n(k)$ is Gaussian, classical LPC and second order deconvolution are equivalent, because the optimal filter must provide Gaussian residue $e(k)$. On the contrary, if $n(k)$ is not Gaussian, the error residue must not be Gaussian. Equations (8) show that the optimal deconvolution as well as the optimal LPC must provide iid residue $e(k)$ with the same pdf than $n(k)$ .Both problems involve higher order statistics, and the knowledge of the pdf or of the score function is required for choosing the optimal (high order) statistics at the ML sense.

Recently, Taleb *et al.* [2, 3] addressed the problem of Wiener system blind inversion using source separation methods. Of course, this approach can also be used for blind linear deconvolution. Assuming the observed signal $x(k)$ satisfies

$$x(k) = [F(z)]n(k) \tag{11}$$

where $F(z)$ is an invertible filter and $n(k)$ an iid.

The deconvolution problem consists in estimating a filter $G(z)$ such that $G(z)F(z) = 1$. The output $e(k)$ of $G(z)$ must then be the iid sequence $n(k)$:

$$e(k) = [G(z)]x(k) \tag{12}$$

The key idea is based on the following parameterization. Denote $\mathbf{x}(k) = (\cdots, x(k), x(k+1), \cdots, x(k+K), \cdots)$, and $\mathbf{e}(k)$ and $\mathbf{n}(k)$ similarly, then equation (11) writes :

$$\mathbf{x}(k) = \mathbf{F}\mathbf{n}(k) \tag{13}$$

where $\mathbf{F}$ is the infinite matrix :

$$\mathbf{F} = \begin{pmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & f(p+1) & f(p) & f(p-1) & \cdots \\ \cdots & f(p+2) & f(p+1) & f(p) & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \tag{14}$$

Similarly, equation (12) writes

$$\mathbf{y}(k) = \mathbf{G}\mathbf{x}(k)\,. \tag{15}$$

Since successive samples of $n(k)$ are iid, the components of the vector $\mathbf{n}(k)$ are spatially statistically independent. Then, equation (13) and (15) are nothing but a mixing model and its separation matrix. Consequently, the blind deconvolution problem is equivalent to an infinite-size blind source separation problem.

Constraining the size of the matrix $\mathbf{G}$, one can deduce a truncated estimation of the filter $G(z)$ by using source separation methods. Anyway, it is now well known that optimal blind source separation requires the knowledge of the source ($s(k)$) pdf or of its score function.

Many methods have been proposed for estimating the score function. Pham *et al.* [4] proposed a simple estimation of score using the projection on a basis of nonlinear functions. Another approach consists in estimating directly the score function with a universal nonlinear model like a multilayer perceptron, according to a LS method. Details on this approach can be found in [5]. A nonparametric approach can also been used, based on the pdf estimation, and followed by a derivation, which give the estimation:

$$\hat{\psi}_E(e) = \frac{\hat{p}'_E}{\hat{p}_E}(e) \tag{16}$$

The pdf estimation can be done with kernel estimators [6], whose a key parameter is the kernel width. This parameter can be chosen easily, but with a good efficiency, with the "rule of thumb" proposed in [6]. This method has been used in the experiments of the following section.

The first question of Section 3 is solved. The quasi-optimal higher order predictor is then showed in Fig. 3. It is a cascade of a classical LPC filter followed by a score function estimator block, which allows to adjust the filter $w(z)$ by satisfying :

$$E\left[\psi_E(e(k))\,x(k-j-1)\right] = 0, \quad j = 0,\cdots,K$$

Algorithm. Denoting $X = \{x(1), x(2), \ldots, x(N)\}$ the observation sequence, the quasi-optimal LPC algorithm writes (for more details, see [2, 3]):

Require: X

% filter initialized to the Dirac function
w = 1
for $k = 1$ to N **do**
  % w filter output initialisation
  $y(k) = [w(z)]x(k-1)$
end for
% error (output) estimation
E=X-Y
repeat
  % score function estimation
  $\psi_E(e)$ estimation
  % estimation of the croscorrelation between
  % observation and score function (Eq. (8))
  $\gamma_{x,\psi_E(e)}$ estimation
  % equivariant adaptation of filter coefficients
  $w \leftarrow w + \mu\left\{\gamma_{x,\psi_E(e)} + \delta\right\} * w$
  for $k = 1$ to N **do**
    % output estimation
    $y(k) = [w(z)]x(k-1)$
  end for
to convergence
% final prediction error, output of the LPC system
E=X-Y

5. EXPERIMENTS

In this section, we compare LPC filters obtained with (i) second-order statistics (the classical method, optimal for Gaussian error pdf) using the Matlab LPC function, and (ii) the quasi-optimal (batch) algorithm, where the score function is estimated using (16) and a kernel estimation of the pdf (our algorithm).

The signal $x(k)$, used as input signal of the LPC system, is generated by the linear filtering of a random noise (Gaussian or non Gaussian) with an all-pole filter $1/H(z)$. Thus, the optimal filter $w(z)$ of the LPC system should satisfy (10).

Performance is evaluated in terms of the parametric square error, averaged over 50 iterations, and has been computed for signal lengths from 100 to 1000 samples.

In Fig. 4, we can see the parametric square errors obtained with the theoretical filter $H(z) =$ **[1,0.5,-**

`0.2]`, for non Gaussian noise. As intended, the quasi-optimal algorithm (diamonds) provides a better performance than the Matlab LPC (small squares), whatever the number of samples of the signal. To emphasize on the interest of the score function estimation, we show experimentally (stars) that an arbitrary choice of the score function, i.e. of the higher order statistics, leads to intermediate performance.

In Fig. 5, we can see the results obtained with the same filter $H(z) =$ `[1,0.5,-0.2]`, but with Gaussian noise. As intended, classical LPC (small squares) and the quasi-optimal algorithm (diamonds) provide very close performance. The classical LPC is better for small samples, probably because of the score function estimation error. In that case, the arbitrary choice of the score function $\psi(e) = atn(e)$ leads to similar results (stars) (see Fig. 6).

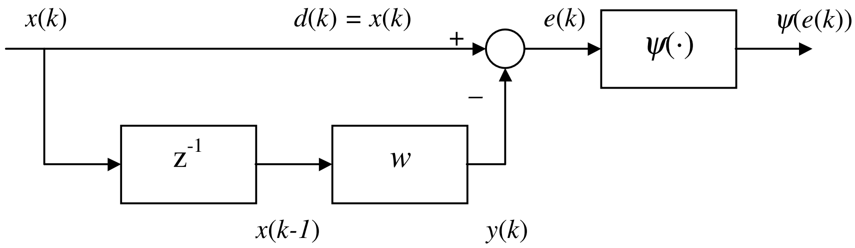

Figure 3. Block diagram of the quasi-optimal higher order predictor.

Theoretical performance can be deduced from Beran results [7] concerning autoregressive processes, since LPC and RA modeling are dual problems (see Section 4).Denoting $\mathbf{w}^*$ the exact solution, and $\hat{\mathbf{w}}_{MMSE}$ and $\hat{\mathbf{w}}_{\psi}$ the estimates based respectively on second order statistics (Gaussian residue) and on higher order statistics (score function of residue is $\psi_E(e)$), Beran shows that the asymptotic distribution of $N^{1/2}(\hat{\mathbf{w}}_{\psi} - \hat{\mathbf{w}})$ as $N \to \infty$ is Gaussian $(0, \|\psi\|^2 \Gamma^{-1})$, with $\|\psi\|^2 = \int \psi^2(u) p(u) du$ and $\psi(u) = p'(u) / p(u)$, while the asymptotic distribution of $N^{1/2}(\hat{\mathbf{w}}_{MMSE} - \hat{\mathbf{w}})$ as $N \to \infty$ is Gaussian $(0, \sigma^2 \Gamma^{-1})$. Then, the asymptotic efficiency of $\hat{\mathbf{w}}_{MMSE}$ is always less than or equal to the asymptotic efficiency of $\hat{\mathbf{w}}_{\psi}$, with equality if and only if the residue pdf is Gaussian. Experiments confirm these results, and emphasize on the performance gain of quasi-optimal LPC, especially for small samples.

6. SUMMARY

Inspired by source separation techniques, we have presented a new algorithm for performing linear prediction, which gives better results than the classical LPC methods, especially for small samples.

The method is based on a criterion which requires the knowledge of error pdf, or more precisely of the score functions. Implicitly, this criterion involves higher order statistics, which can be chosen optimally with a good estimation of the score function, e.g. computed from kernel estimators of the error pdf.

For non Gaussian noise (RND[3]), experiments show that this method is always better than Matlab LPC function, simply based second order statistics. For Gaussian noise, performance obtained with the two methods are equal.

Currently, practical and theoretical issues are both addressed: (i) how improve the estimation of the *score* function for enhancing the results and obtaining faster and better algorithms, (ii) computing the performance of the method according to the estimation error on the score function.

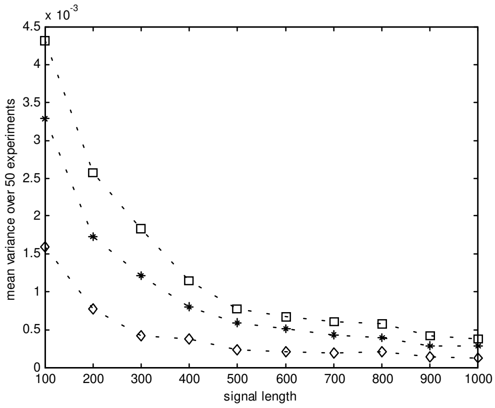

Figure 4. Error variance averaged over 50 iterations, for signal lengths from 100 to 1000 samples, with non Gaussian (RND^3) noise. Small squares are obtained by the Matlab LPC function ; Diamond are the results of the quasi-optimal algorithm ; Stars corresponds to arbitrary higher order statistics due to the approximation $\psi(e) = atn(e)$.

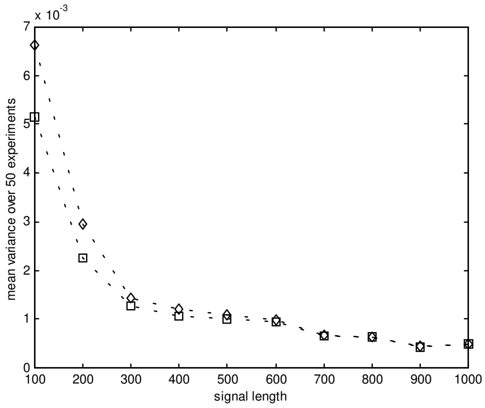

Figure 5. Error variance averaged over 50 iterations and for signal lengths from 100 to 1000 samples, with Gaussian noise. The results obtained by classical LPC (small squares) and quasi-optimal (diamonds) algorithms are very close.

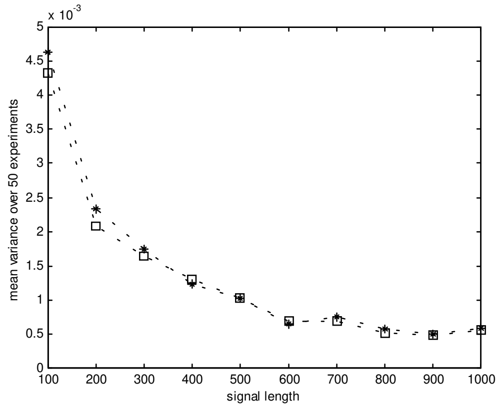

Figure 6. Error variance averaged over 50 iterations and for signal lengths from 100 to 1000 samples, with Gaussian noise. The results obtained by classical LPC (small squares) and heuristic higher order (stars) algorithms are very close.

7. REFERENCES

[1] Proakis J.G., D.G.Manolakis, *Digital Signal Processing. Principles, Algorithms and Applications*, Prentice Hall, (1996)

[2] Taleb A. Solé J., Jutten C., Blind Inversion of Wiener Systems. *IWANN 99*, Alicante (Spain), pp. 655-664 *(*1999)

[3] Taleb A. Solé J., Jutten C., Quasi-Nonparametric Blind Inversion of Wiener Systems. Submitted to *IEEE Transactions on Signal Processing*. April 1999

[4] Pham D.-T, Garat Ph., Jutten C., Separation of mixtures of independent sources through a maximum likelihood approach. *EUSIPCO 92*, Brussels (Belgium), Vol. 2, pp. 771-774 (1992)

[5] Taleb A., Jutten C., Entropy optimization. Application to blind source separation. *ICANN 97*, Lausanne (Switzerland), pp. 529-534 (1997)

[6] Härdle W., *Smoothing Techniques with implementation in S*, Springer-Verlag, (1991)

[7] Beran R., Adaptive estimates for autoregressive processes. Annals of the Institute of Statistical Mathematics, Vol 28, pp. 77-90 (1976)