COMPETITIVE CHAOTIC AR(1) MODEL ESTIMATION

David Luengo, Carlos Pantaleon and Ignacio Santamaria
Departamento de Ingenieria de Comunicaciones (DICOM)
Universidad de Cantabria, ETSIIT, Santander, Spain
Phone: +34 942 201392 Ext. 15
Fax: +45 942 201488
E-mail: david,carlos,nacho@gtas.dicom.unican.es
Web: http://gtas.dicom.unican.es

Abstract. Chaotic signals, signals generated by a nonlinear dynamical system in chaotic state, may be useful models for many natural phenomena. In this paper we show a family of first-order difference equations with autocorrelation function identical to first-order autoregressive processes AR(1). We consider the Maximum Likelihood (ML) estimator of the model, and an efficient suboptimal method with reduced computational cost. However, for very large data records or on-line model estimation, even the suboptimal algorithm may have an excessive computational cost. In these cases we propose a low-cost competitive model estimation approach using a LMS-like algorithm for model training and adaption. Computer simulations show the good performance of this model estimation procedure.

INTRODUCTION

Chaos theory has been applied to several areas during the past years. Chaotic signals, signals generated by a nonlinear dynamical system in chaotic state, may be useful in modeling natural phenomena due to their special characteristics. For example, their extreme sensitivity to initial conditions makes signal generation a delicate task, but may be considered an advantage in representing anomalous behaviour of signals over short periods of time [12]. Chaotic models have been proposed for biomedical signals [1], wind velocity fields [2], the sea clutter [4], packet traffic [11], speech waveforms [12], as well as signals arising from many processes in experimental physics.

The application of chaotic modeling is conditioned by the lack of a family of chaotic models that combine a certain generality with easily computable estimation algorithms. Chaotic modeling with the Duffing equation has been considered in [12]. Neural networks as chaotic models have been proposed in [2] and [4]. In these two cases the effect of noise is not considered and, in the case of neural networks, the models are impossible to analyze. One dimensional difference equations (chaotic maps) have also been considered in [11], although for a very specific application and without considering the problem of the model estimation in noise.

Ideally, we would search for the chaotic equivalent of ARMA models. Chaotic signals generated by eventually expanding Piecewise-Linear (PWL) Markov maps could claim the title of chaotic ARMA models, since they have rational spectra [5]. Nevertheless, it is unclear whether it is possible to construct chaotic PWL Markov maps with any desired spectra. Restricting the models to PWL maps allows the analysis of ML signal estimators for a known map. The ML estimator is inconsistent, so the asymptotic distribution for large data records is invalid. However, for a high Signal to Noise Ratio (SNR), the ML estimator is asymptotically unbiased and attains the Cramer-Rao Lower Bound (CRLB) [6]. A closed-form expression for the ML estimator of chaotic signals generated by iterating known PWL maps is derived in [8]. Parameter estimation has received much less attention, relying mostly on linear approaches, although ML estimators have been considered [7]. No closed-form solution is known for the ML estimator of the joint problem (parameter and signal estimator).

In this paper we develop parameter and signal estimators for a class of chaotic difference equations that produce signals with autocorrelation function identical to AR(1) processes. The exact ML model estimator is considered, but its computational cost and inconsistency makes it useless. A suboptimal block estimator that combines an initial parameter estimation stage with a subsequent ML signal estimator using the parameter previously obtained is also considered. However, when the data record is very large or the model must be estimated on-line (in a sample by sample basis) these estimators are not adequate due to their computational cost. In these cases we propose a low-cost competitive parameter estimation scheme using a LMS-like adaptive algorithm for the parameters of the model, which may be complemented in a second stage with the ML signal estimator.

CENTERED SKEW-TENT MAPS

The signals $x[n]$ that we consider in this work are generated according to

$$x[n+1] = F(x[n]) \tag{1}$$

where $F(\cdot)$ is the so called centered skew-tent map

$$F(x) = \begin{cases} \frac{2(1+x)}{1+a} - 1 & x \leq a \\ \frac{2(1-x)}{1-a} - 1 & x > a \end{cases} \tag{2}$$

for some parameter $-1 < a < 1$. This map is a particular case of a PWL map, which may be generally expressed as [8]

$$F(x) = \sum_{i=1}^{M} (a_i + b_i x)\chi_i(x) \tag{3}$$

where M is the number of disjoint convex intervals E_i in which the phase space of x may be divided, and $\chi_i(x)$ is an indicator function that denotes whether x belongs to interval i or not

$$\chi_i(x) = \begin{cases} 1 & x \in E_i \\ 0 & x \notin E_i \end{cases} \tag{4}$$

For the centered skew-tent map we have only two intervals: $E_1 = [-1, a)$ with parameters $a_1 = 2/(1+a)$ and $b_1 = (1-a)/(1+a)$, and $E_2 = [a, 1]$ with parameters $a_2 = -2/(1-a)$ and $b_2 = (1+a)/(1-a)$. This map produces sequences that are chaotic with invariant density $p(x)$ uniform in the range $[-1, 1]$, [13]. If a symbol from a known alphabet is assigned to each of the regions, the dynamics of the map may be characterized by following the different regions that the map visits during its dynamical evolution. This evolution is described by the sign sequence $\mathbf{s} = s[0], ..., s[N-1]$, also called itinerary, where

$$s[n] = i \Leftrightarrow F^{(n)} \in E_i \tag{5}$$

and $F^{(n)}$ is the n-fold functional composition of F. Centered skew-tent maps are onto. Therefore, all the itineraries are possible, and there are 2^N regions in the phase space. Chaotic signals generated by iterating (2) according to (1) have statistical properties that make them the equivalent of AR(1) models, since their autocorrelation function is [9]

$$R_{xx}[m] = r_o a^m \tag{6}$$

with $r_o = 1/3$. Therefore, the parameter a has the same relation with the autocorrelation as in the case of AR(1) processes [13].

BLOCK ESTIMATION OF THE MODEL

Problem statement

The signal model we are considering is

$$y[n] = x[n] + w[n] \qquad n = 0, 1, \ldots, N \tag{7}$$

where $x[n]$ is generated using (2) by iterating some unknown $x[0] \in (-1, 1)$ according to (1) for some parameter $-1 < a < 1$, and $w[n]$ is a stationary, zero-mean, white Gaussian noise with variance σ^2. Model estimation demands obtaining an estimate of the parameter a and of the initial condition $x[0]$ to reproduce $x[n]$. ML model estimation produces the initial condition and the parameter that minimize

$$J(x[0], a) = \sum_{n=0}^{N} \left(y[n] - F^{(n)}(x[0], a) \right)^2 \tag{8}$$

This problem as it is stated has not been solved yet. However, the ML estimator is feasible, although of high computational cost. Minimizing (8) requires the computation of 2^N estimates, one for each possible itinerary, followed by the application of a gradient descent algorithm on a highly complex cost function. Therefore, an alternative algorithm based on the initial estimation of the parameter a using only pairs of samples, followed by a subsequent ML signal estimator is considered [9].

Parameter estimation

To obtain an estimate of the parameter a we exploit the deterministic relation that exists between each sample and the next one. Thus, the resulting forward cost function is

$$J(a) = \sum_{n=1}^{N} (y[n] - F(y[n-1], a))^2 \tag{9}$$

This nonlinear minimization problem may be solved using gradient descent techniques. However, the dependence of the itinerary on the parameter a hinders the performance of the estimator. An alternative backward cost function may be established using the relation that exists between each sample and the previous one

$$J(a) = \sum_{n=1}^{N} (y[n-1] - F^{-1}(y[n], a))^2 \tag{10}$$

Using this cost function the problem may be decomposed into a set of linear ones as a function of the itinerary. This cost function may be expressed as

$$J_{\mathbf{s}}(a) = ||\mathbf{d_s} - \mathbf{z}a||_2^2 \tag{11}$$

where $|| \cdot ||_2^2$ is the squared Euclidean norm, $\mathbf{s}$ is the vector of the sign components of $y[0], ..., y[N-1]$, $\mathbf{z} = [1 + y[1], ..., 1 + y[N]]^T$, and $\mathbf{d_s} = [d[0], ..., d[N-1]]^T$, where

$$d[n] = \begin{cases} 2y[n] - y[n+1] + 1 & x \in E_i \\ 2y[n] + y[n+1] - 1 & x \notin E_i \end{cases} \tag{12}$$

Obtaining the Least Squares (LS) solution requires considering the 2^N possible itineraries and minimizing (11) for each one. Nevertheless, in a moderate/high SNR situation (above 10 dB), it seems reasonable to consider only the $N + 2$ possible itineraries produced by sorting the data samples and dividing them in two continuous sets. Thus, we obtain a Hard-Censoring LS

(HCLS) estimate of the itinerary $\mathbf{s}$ [9]. For a known itinerary, the LS estimate of the parameter a is

$$\hat{a}_{\mathbf{s}} = \left(\mathbf{z}^T\mathbf{z}\right)^{-1}\mathbf{z}^T\mathbf{d}_{\mathbf{s}} = E_{\mathbf{z}}^{-1}\mathbf{z}^T\mathbf{d}_{\mathbf{s}} = \mathbf{z}^{\#}\mathbf{d}_{\mathbf{s}} \tag{13}$$

where $E_{\mathbf{z}} = \mathbf{z}^T\mathbf{z}$ is the squared norm of $\mathbf{z}$, $\mathbf{z}^{\#}$ is the pseudoinverse of $\mathbf{z}$, and the cost function associated to (13) is

$$J_{\mathbf{s}}(\hat{a}_{\mathbf{s}}) = ||(\mathbf{I} - E_{\mathbf{z}}^{-1}\mathbf{z}\mathbf{z}^T)\mathbf{d}_{\mathbf{s}}||_2^2 = ||\mathbf{E}\mathbf{d}_{\mathbf{s}}||_2^2 \tag{14}$$

The HCLS parameter estimate is computed applying (13) using the itinerary that minimizes (14) for the $N+2$ possible itineraries [9]. Note that, since $\mathbf{E}$ does not depend on $\mathbf{s}$, it must be calculated just once, resulting in a moderate computational cost.

Signal estimation

Once the parameter estimate has been obtained, we can apply the ML estimator in [8] to obtain the signal estimate. For a given sign sequence $\mathbf{s}$, and parameter a, the ML estimate of $x[N]$ is the value which minimizes

$$J(x[N]) = \sum_{n=0}^{N-1}\left(y[n] - F_{\mathbf{s}}^{(-(N-n))}(x[N])\right)^2 \tag{15}$$

In our approach the parameter a and the itinerary $\mathbf{s}$ are the ones obtained from the parameter estimation algorithm. To avoid the numerical instability characteristic of the generation of chaotic signals by forward iteration we estimate $x[N]$ instead of $x[0]$. Therefore, we need a closed-form expression for $F^{(-n)}(\cdot)$ similar to the one for $F^{(n)}(\cdot)$ in [8]

$$F_{\mathbf{s}}^{(-n)}(x[N]) = (A_{\mathbf{s}}^{n,N})^{-1}x[N] - \sum_{l=0}^{n-1}(A_{\mathbf{s}}^{l+1,N-n+l+1})^{-1}b_{s[N-n+l]} \tag{16}$$

being

$$A_{\mathbf{s}}^{l,k} = \prod_{n=k-l}^{k-1} a_{s[n]} \tag{17}$$

and $A_{\mathbf{s}}^{0,k} = 1$. Since (16) is linear in $x[N]$, (15) is quadratic, and there is a unique minimum at the ML estimator of $x[N]$, which is given by

$$\hat{x}[N] = \frac{\sum_{n=0}^{N-1}(y[n] + \sum_{l=0}^{N-n-1}(A_{\mathbf{s}}^{l+1,n+l+1})^{-1}b_{s[n+l]})(A_{\mathbf{s}}^{N-n,N})^{-1}}{\sum_{n=0}^{N-1}(A_{\mathbf{s}}^{N-n,N})^{-2}} \tag{18}$$

The rest of the signal $x[0], ..., x[N-1]$ is obtained by iterating backwards from $\hat{x}[N]$ using (16), with the HCLS sign sequence $\mathbf{s}$ obtained from the parameter estimation algorithm.

COMPETITIVE MODEL ESTIMATION

The block model estimation methods presented in the previous section have a high computational cost in the parameter estimation stage, $O(2^N)$ for the LS and $O(N)$ for the HCLS, that may prevent their application for large data records. Besides, for on-line model estimation, the parameter (and thus the model), must be adapted in a sample by sample basis, requiring low-cost, fast estimation procedures. In these cases we propose a competitive estimation of the parameter a of the skew-tent map.

Competitive learning is a well-known neurocomputing paradigm. It can be stated as follows: we have a collection of vector observations $\mathbf{x}(t)$ and a set of reference vectors $\mathbf{m}_i(t)$ initialized randomly. We iteratively choose one of the $\mathbf{x}(t)$ and compare it with all the reference vectors using some metric. The winner of this competition reduces its distance (in the reference metric) to the training vector. When stability is reached, every reference vector represents a group (cluster) of the training data [3].

Competitive learning causes each of the reference vectors to concentrate on a particular group of patterns. This idea has been extended in [10] to models, where several linear models compete for training patterns, concentrating each one in some group of them that share some kind of similarity. In this case we have one model for each interval of the PWL map consisting of the relationship between each sample and the next, being a_i and b_i the parameters to be estimated for each model. Depending on whether we express $y[n]$ as a function of $y[n-1]$, or as a function of $y[n+1]$, we have a forward or backward competitive model estimator.

Forward competitive estimation

For the skew-tent map we have only two models and a single parameter a to estimate. Expressing $y[n]$ as a function of the previous sample $y[n-1]$, the error for each of the two models is

$$e_1[n] = y[n] - \left(\frac{2(1 + y[n-1])}{1 + a[n-1]} - 1 \right) \tag{19}$$

$$e_2[n] = y[n] - \left(\frac{2(1 - y[n-1])}{1 - a[n-1]} - 1 \right) \tag{20}$$

Considering a quadratic cost function, we may develop an LMS-like adaptive algorithm. Thus, the parameter a at the n-th iteration is

$$a[n] = \begin{cases} a[n-1] - \mu e_1[n] \frac{2(1+y[n-1])}{(1+a[n-1])^2} & e_1^2[n] \leq e_2^2[n] \\ a[n-1] + \mu e_2[n] \frac{2(1-y[n-1])}{(1-a[n-1])^2} & e_1^2[n] > e_2^2[n] \end{cases} \tag{21}$$

where $a[0]$ is a randomly chosen value in (-1,1). The fact that the parameter to be estimated is a divisor in (21) causes this algorithm to be very sensitive to noise, limiting its performance.

Backward competitive estimation

A better performance may be obtained formulating the inverse (backward) problem, i.e. expressing $y[n-1]$ as a function of $y[n]$. In this case, the error for each of the two models is

$$e_1[n] = y[n-1] - \left(\frac{(1+a[n-1])(1+y[n])}{2} - 1\right) \tag{22}$$

$$e_2[n] = y[n-1] - \left(1 - \frac{(1-a[n-1])(1+y[n])}{2}\right) \tag{23}$$

and, defining the error after each iteration as

$$e[n] = \begin{cases} e_1[n] & e_1^2[n] \leq e_2^2[n] \\ e_2[n] & e_1^2[n] > e_2^2[n] \end{cases} \tag{24}$$

we may express the parameter a for the n-th iteration as

$$a[n] = a[n-1] + \mu e[n](1+y[n])/2 \tag{25}$$

Therefore we have an LMS algorithm for the estimation of a, where the error is obtained in competition between the models for each of the intervals of the phase space of the map, and the input signal is $(1+y[n])/2$. The competition is pursued until some termination criterion is fulfilled: the mean squared error over the data set falling below a threshold, or maintaining a constant value over several iterations of the algorithm. This method may be used as a stand-alone algorithm for estimating a for on-line model estimation, or as an alternative parameter estimation algorithm for the scheme shown in the previous section. Moreover, the parameter a obtained from the competitive estimator may be used as a segmentation model to obtain the sign sequence $\mathbf{s}$, and consequently the HCLS estimator applying (13). The disadvantage of this approach is its slower convergence rate for the same value of μ than the forward approach.

SIMULATION RESULTS

In this section we analyze the performance of the competitive model estimators and compare them with the block estimators. Concerning parameter estimation we compare the gradient descent approach, the HCLS solution, and the forward and backward competitive approach. In figure 1 we show a typical MSE curve obtained averaging the results of 1000 simulations for each of the different values of SNR, for a skew-tent map with $a = 0.5$, $N = 99$ and a random initial condition. From figure 1 it can be inferred that the backward competitive method improves the performance of the gradient descent algorithm, and is very close to the HCLS estimate. On the other hand, the forward method is close to the gradient descent approach. The difference in performance is due to the division in (21).

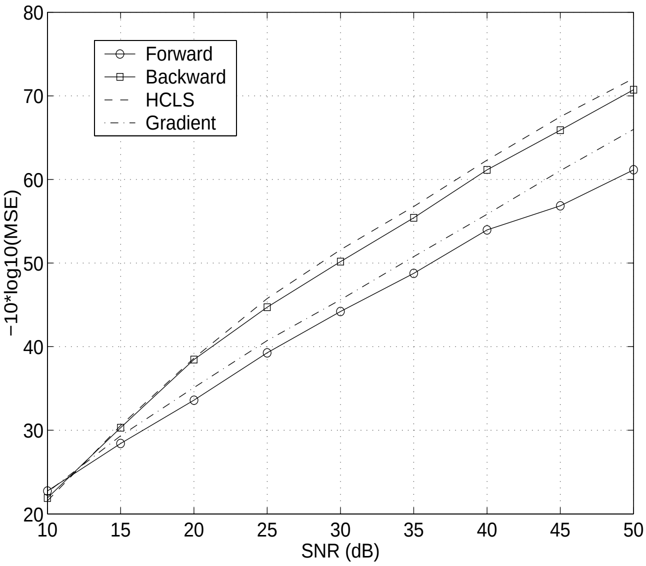

Figure 1: Comparison of alternatives of parameter estimation for $N = 99$, $a = 0.5$.

Both competitive algorithms converge at a relatively fast rate, although the forward method converges faster than the backward method. Figure 2 shows two typical convergence curves for the forward and backward methods. The convergence of the algorithm may be improved increasing μ, thus trading MSE for convergence rate.

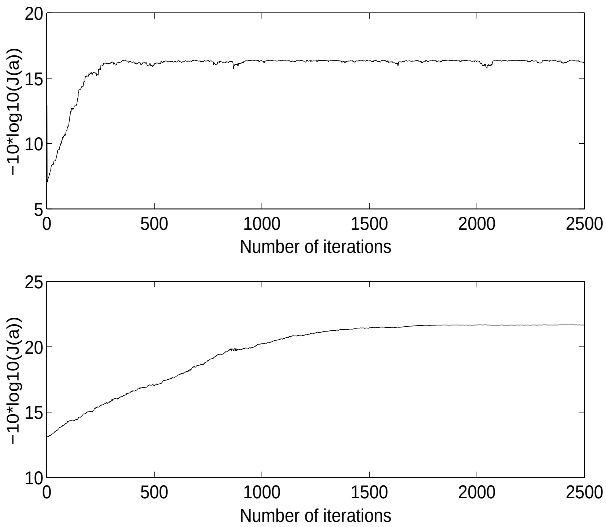

Figure 2: Convergence rate of forward (upper) and backward (lower) competitive estimators for $N = 99$, $a = 0.5$ and $\mu = 0.01$.

Finally, in figure 3 we show the MSE obtained using the backward competitive method for an on-line model estimation. The parameter and itinerary are estimated on-line using a map with 99 iterations and, once convergence has been achieved, the ML estimator is applied to estimate the full signal iterating backwards. This approach improves considerably the SNR of the original signal, and shows a similar performance to the the block estimator [9]. The best peformance is achieved for low absolute values of the parameter a, as in the case of the block estimator.

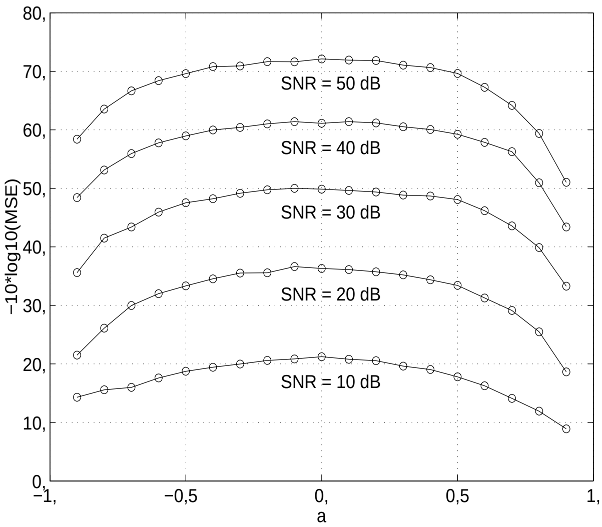

Figure 3: MSE of backward competitive on-line estimator (with ML estimator for signal estimation) for $N = 99$.

CONCLUSIONS

In this paper we have shown a chaotic alternative to AR(1) models, and developed parameter and signal estimators. The ML estimator has been considered, but its huge computational cost and inconsistency make it useless. Therefore, we have considered an alternative algorithm based on the separate estimation of the parameter a and itinerary, followed by an ML signal estimator. However, for large data records, or on-line model estimation, even these block estimators may have an excessive computational cost. In these cases we propose a competitive model estimation approach using an LMS-like algorithm. This approach shows a good performance as a stand-alone model estimation scheme for on-line computation, and almost achieves the performance of the block estimator when used as an alternative parameter estimation method. Further lines of research include developing estimators for other maps, and searching for the chaotic AR(p) and ARMA models.

ACKNOWLEDGEMENTS

This work was partially supported by the European Community and the Spanish MCYT under the projects 1FD97-1066-C02-01 and 1FD97-1863-C02-01.

REFERENCES

[1] M. Akay (ed.), **Nonlinear biomedical signal processing - Vol. II: Dynamic analysis and modeling**, IEEE Press Series on Biomedical Eng., 2001.
[2] F. M. Aparicio-Acosta, “An hybrid noise reduction method for state-space reconstruction,” **Proc. ICASSP**, vol. 3, pp. 121–124, 1993.
[3] S. Haykin, **Neural networks: A comprehensive foundation**, Macmillan Publishing Company, 1994.
[4] S. Haykin and X. B. Li, “Detection of signals in chaos,” **Proceedings of the IEEE**, vol. 83, no. 1, pp. 95–122, 1995.
[5] S. H. Isabelle and G. W. Wornell, “Statistical analysis and spectral estimation for one-dimensional chaotic signals,” **IEEE Trans. on Signal Processing**, vol. 45, no. 6, pp. 1495–1506, 1997.
[6] S. M. Kay, “Asymptotic maximum likelihood estimator performance for chaotic signals in noise,” **IEEE Trans. on Signal Processing**, vol. 43, no. 4, pp. 1009–1012, 1995.
[7] C. Pantaleon, D. Luengo and I. Santamaria, “An efficient method for chaotic signal parameter estimation,” **Proc. EUSIPCO**, vol. 3, pp. 9–12, 2000.
[8] C. Pantaleon, D. Luengo and I. Santamaria, “Optimal estimation of chaotic signals generated by piecewise-linear maps,” **IEEE Signal Processing Letters**, vol. 7, no. 8, pp. 235–237, 2000.
[9] C. Pantaleon, D. Luengo and I. Santamaria, “Chaotic AR(1) model estimation,” **Proc. ICASSP**, vol. VI, 2001.
[10] C. Pantaleon, I. Santamaria and A. R. Figueiras-Vidal, “Competitive local linear modeling,” **Signal Processing**, vol. 49, pp. 73–83, 1996.
[11] P. Pruthi and A. Erramilli, “Heavy tailed ON/OFF source behavior and self-similar traffic,” **Proc. ICC**, vol. 1, pp. 445–450, 1995.
[12] T. F. Quatieri and E. M. Hofstetter, “Short time signal representation by nonlinear difference equations,” **Proc. ICASSP**, pp. 1551–1554, 1990.
[13] H. Sakai and H. Tokumaru, “Autocorrelation of a certain chaos,” **IEEE Trans. on Signal Processing**, vol. 28, no. 5, pp. 588–590, 1980.