ANALYSIS OF GRADIENT-BASED ADAPTATION ALGORITHMS FOR LINEAR AND NONLINEAR RECURSIVE FILTERS

Christophe Vignat, Christine Uhl and Sylvie Marcos*

Laboratoire des Signaux et Systèmes
ESE-Plateau de Moulon, 91192 Gif sur Yvette, Cedex, France
* Thomson CSF/RGS, 66 rue du Fossé blanc, 92231 Gennevilliers, France

ABSTRACT

In this paper, the problem of adapting linear and nonlinear recursive filters through a gradient-based optimization procedure is considered. The rigourous application of this technique implies a time-growing computation load. Recently, a method for estimating the weight updates was introduced, leading to a new class of algorithms. Here, the convergence properties of these algorithms, when applied to a linear, then nonlinear recursive filter, are exhibited, through a dynamical analysis of the adaptation process. Since the general analysis is very difficult, the case of a first order filter with a constant input is considered. Significant results are obtained in this particular application.

INTRODUCTION

In many fields of signal processing, such as time series prediction and modelling or system identification, the use of recursive linear or nonlinear filters is desirable. The adaptation ability of these filters is essential when the nonstationnary environment implies that the system acquires and continuously tracks an optimal state. Then, adaptive algorithms aim at updating the system parameters.

The LMS algorithm is of common use in linear filtering because of its simplicity. It consists in a stochastic approximation of the gradient algorithm minimizing the mean square error at the system ouput. However, in addition to the specific problems of recursive filters (i.e. non-unicity of the optimal filter, possible unstability), appears the difficulty of evaluating the recursive quantities (output error and its partial derivatives) involved in the parameter updating [1]. These quantities should theoretically be computed, at time n, as the output of a recursive filter whose parameters are fixed at their latest value, taking into account all the past of the input signal. This results in a memory length and a computation load which grow with n. This is not realistic for an on-line application.

A class of algorithms has been recently proposed [2], consisting in truncating the memory of the system to fixed values, i.e. L for the system output and L' for the derivatives. These finite memory recursive LMS algorithms (FMRLMS(L,L')) are briefly described in the first section.

In section 2, the behavior of these algorithms (stability, speed and accuracy of convergence) is analysed as a function of L and L'. The global system, consisting in the recursive filter plus the adaptation process, is a discrete time dynamical system whose behavior is described by a nonlinear recurrence:

$$X(n+1) = F(X(n)). \tag{0.1}$$

X(n) is the state vector at time n, and its dimension is the order of the system. The analysis of this system is performed through the search of fixed points and second order cycles as well as their associated multipliers. The simple case of a first order linear, and then nonlinear, filter with a constant input is considered. Results are given with respect to the values of L and L' and compared to simulation results.

1. GRADIENT-BASED ADAPTATION OF RECURSIVE FILTERS

1.1. The gradient algorithm

Let e(n) be the output of a recursive nonlinear filter with input s(n), transverse parameters a_j, $0 \leq j \leq T$ and recursive parameters b_i, $1 \leq i \leq R$

$$e(n) = \sum_{i=1}^{R} b_i f(e(n-i)) + \sum_{j=0}^{T} a_j s(n-j). \tag{1.1}$$

Note that such a nonlinear recursive filter can be a part of a DPCM speech encoder or of a decision feedback loop equalizer. It can also be viewed as a formal neuron with a feedback and with a time signal as input [2]. Some examples of function f will be given below.

In an adaptive context, a_j and b_i parameters are updated in order to minimize the mean square error between e(n) and a target response d(n). In the following, we will set d(n)=0. A

0-7803-0532-9/92 $3.00 © 1992 IEEE

classical minimization procedure consists in using a gradient-based algorithm. The straightforward application of the LMS algorithm to the case of recursive filters leads to the following parameters update formula:

$$\begin{cases} a_j(n+1)=a_j(n)+\mu e(n)\alpha_j(n) & j=0, \dots, T \\ b_i(n+1)=b_i(n)+\mu e(n)\beta_i(n) & i=1, \dots, R \end{cases} \quad (1.2)$$

where $\alpha_j(n)$ and $\beta_i(n)$ are computed recursively as follows:

$$\begin{cases} \alpha_j(n) \triangleq -\dfrac{\partial e(n)}{\partial a_j} = s(n-j)+\sum_{k=1}^{R} b_k f'(e(n-k))\alpha_j(n-k) \quad (1.3) \\ \beta_i(n) \triangleq -\dfrac{\partial e(n)}{\partial b_i} = e(n-i)+\sum_{k=1}^{R} b_k f'(e(n-k))\beta_i(n-k) \end{cases}$$

At time n, all the derivatives $\alpha_j(n-k)$, $\beta_i(n-k)$ and the outputs $e(n-k)$ used in (1.1) and (1.3) should be evaluated with the coefficients $a_j=a_j(n)$ and $b_k=b_k(n)$ being fixed. Accordingly, at time n, $\alpha_j(n)$, $\beta_i(n)$ and $e(n)$ should be computed with an unbounded growing memory and fixed parameters. This cannot be fulfilled in practice.

1.2. Finite memory recursive algorithms

Recently, a general class of algorithms was introduced, which consists in truncating, for the calculus of $e(n)$ (resp. $\alpha_j(n)$ and $\beta_i(n)$), the memory to a fixed length L (resp. L'). This class of FMRLMS(L,L') algorithms was shown to include most of the gradient-based algorithms existing in the litterature for adaptive recursive linear filtering [2]. These FMRLMS(L,L') algorithms consist in replacing the recurrences (1.1) and (1.3) by

$$\begin{cases} e^{(n)}(n) = \sum_{i=1}^{R} b_i(n) f(e^{(n)}(n-i)) + \sum_{j=0}^{T} a_j(n)s(n-j) \\ e^{(n)}(n-1) = \sum_{i=1}^{R} b_i(n) f(e^{(n)}(n-i-1)) + \sum_{j=0}^{T} a_j(n)s(n-j-1) \\ --- \qquad\qquad (1.4) \\ e^{(n)}(n-L+1) = \sum_{i=1}^{R} b_i(n) f(\tilde{e}^{(n)}(n-i-L+1)) + \sum_{j=0}^{T} a_j(n)s(n-j-L+1) \end{cases}$$

$$\begin{cases} \beta_i^{(n)}(n)=-f(e^{(n)}(n-i))+\sum_{k=1}^{R} b_k(n) f'(e(n-k))\beta_i^{(n)}(n-k) \\ \beta_i^{(n)}(n-1)=-f(e^{(n)}(n-i-1))+\sum_{k=1}^{R} b_k(n) f'(e(n-k-1))\beta_i^{(n)}(n-k-1) \\ --- \qquad\qquad (1.5) \\ \beta_i^{(n)}(n-L'+1)=-f(e^{(n)}(n-i-L'+1)) \\ \qquad +\sum_{k=1}^{R} b_k(n) f'(e(n-k-L'+1))\tilde{\beta}_i^{(n)}(n-k-L'+1) \end{cases}$$

$\alpha_j(n)$ is determined in the same way as $\beta_i(n)$ with s instead of f(e). The superscript (n) means that the values are computed with $a_j(n)$ and $b_i(n)$ being fixed. Several choices to initialize these recurrences are possible [3]. In this paper, we choose $\tilde{e}^{(n)}(n-i-L+1)=e^{(n-1)}(n-i-L+1)$, $\tilde{\beta}_i^{(n)}(n-k-L'+1) = \tilde{\beta}_i^{(n-1)}(n-k-L'+1)$. In this case, the recursive nature of the system is maintained.

Note that the extended LMS algorithm [1] corresponds to L=1 and $\beta_i^{(n)}(n)=-f(e^{(n)}(n-i))$ which cannot be derived from (1.5). In the following, a generalization of the extended LMS, $L\geq1$ and $\beta_i^{(n)}(n)=-f(e^{(n)}(n-i))$, will be referred to by artificially noting L'=0.

2. BEHAVIOR OF THE FMRLMS ALGORITHMS

2.1. Definitions

X is a fixed point of the non linear recurrence (0.1) if:

$$F(X)=X \quad (2.1)$$

A second order cycle of the non linear recurrence (0.1) is a set of 2 points C=[X, Y] satisfying:

$$F(X)=Y,\ F(Y)=X,\ F(X)\neq X,\ F(Y)\neq Y. \quad (2.2)$$

Their attractive or repulsive nature is determined by their multipliers which are the eigenvalues of J[G(X)], J being the jacobian, and G(X)=F(X) if X is a fixed point, G(X)=F(F(X)) if X is a point of a second order cycle. Note that the multipliers are independent of the point X of the cycle. If all the multipliers have their moduli smaller than 1, the cycle is attractive. If one of the multipliers has a modulus larger than 1, the cycle is repulsive.

2.2. The investigated system

Since the study in the general case is very difficult, we deal with the case of a first order recursive nonlinear system, with a constant input s, adapted by the FMRLMS algorithm. It is completely determined by a recurrence of type (0.1) which is of order 2 ($X(n)=(e^{(n)}(n),b(n))$) if L'=0, or 3 ($X(n)=(e^{(n)}(n), b(n), \beta^{(n)}(n))$) if $L'\neq0$:

$$e^{(n)}(n) = s - b(n)f(e^{(n)}(n-1)),\quad L\geq1, \quad (2.3)$$
$$b(n+1)=b(n)+\mu e^{(n)}(n)\beta^{(n)}(n)$$
$$\beta^{(n)}(n) = f(e^{(n)}(n-1))-b(n)f'(e^{(n)}(n-1))\beta^{(n)}(n-1)) \text{ if } \mathbf{L'\neq0}$$
$$\beta^{(n)}(n) = f(e^{(n)}(n-1)) \quad \text{if } \mathbf{L'=0}.$$

Accordingly, if L'=0 the system is of order 2 and if L'≠0, it is of order 3. This case is the basis to a generalization for more complex systems. The superscript (n) will be omitted in the following.

2.3. The linear system: f(x)=x

This case was investigated in [4]. There is no fixed point for $s\neq0$. If $s\neq0$, two different cycles, C_1 and C_2, are found according as $L'=0$ or $L'\neq0$ and $L>1$.

When **L'=0**, $C_1=[(s, 1), (0, 1)]$. Its multipliers are complex conjugate. Note that in C_1, b=1 is the stability limit for a first order recursive linear filter. The case L=1 (extended LMS) is critical since the modulus of the multipliers is equal to 1. There is no general theory determining the nature of the cycle C_1 in this case. The simulations show that b(n) moves around 1 at a fixed distance without leading to instability: the cycle is neither attractive nor repulsive. The latter has been referred to as the selfstabilization phenomenon of the LSM algorithm [5]. When $L>1$, C_1 is a focus with multipliers modulus equal to $|1-\mu s^2(L-1)/2|$. If $\mu s^2(L-1)<4$, C_1 is attractive and the selfstabilization is reinforced. This is illustrated in figure 1.

When **L'≠0**, $C_2=[(e_1,b_1,\beta_1), (e_2,b_2,\beta_2)]$ with e_1 and e_2 not equal to 0, and b_1 and b_2 not equal to 1. The points of the cycle and the multipliers can be computed numerically; the system is unstable (repulsive cycle) if L=1 [5] and stable (attractive cycle) if $L>1$. In addition, the larger L', the faster the convergence to C_2. This is illustrated in figure 2 and 3.

2.4. The nonlinear system: f(x)=A sign(x)

Since $f'(x)=0$ except for x=0, necessarily **L'=0.** In this case, it is easy to show that there exists a fixed point (0, s/A). Its multipliers $(0,1-\mu A^2)$ are independent of s and L, and the fixed point is attractive. Futhermore, no second order cycle such as $b_1=b_2$ exists.

2.4. The nonlinear system: f(x)=A tanh(px), p<∞

This nonlinear function, characterized by its slope p at the origin, is common in neural networks and can be viewed as a derivable approximation function of the sign function used in quantification or decision systems.

There are no fixed points for $s\neq0$. Two different cycles, C_3 and C_4, are found according as $L'=0$ or $L'\neq0$.

When **L'=0**, $C_3 = [(0, s/f(s)), (s, s/f(s))]$. In the case of $p>>A/s$, f(s)=A and f'(s)=0 and the multipliers $(0, 1-\mu psA)$ of C_3 are found independent of L. Cycle C_3 is illustrated in figure 4.

In the case of $p<<A/s$ such as $f(s)\approx ps$ and $f'(s)\approx p$, the multipliers of C_3 are found equal to $(1, 1-\mu p^2s^2(L-1)/2)$. This is illustrated in figure 5; note that in this case $s/f(s)>s/A$.

When **L'≠0**, the determination of second order cycles is untractable. For L=L'=1, approximated results can be found and their generalization to larger values of (L,L') can be observed on simulations.

In the case of $p<<A/s$ $(f(s)\approx ps)$, $C_4=[(e_1,b_1,\beta_1),(e_2,b_2,\beta_2)]$. The two points of the cycle are different but close. In first approximation, we find (see also in figure 6): e_1 # $e_2 = s/2$, b_1 # $b_2 = 1/p$, β_1 # $\beta_2 = s/(1+p^2)$.

In the case of $p>>A/s$ $(f(s)=A)$ such as $f(e)\approx A$, the results are shown to be the same as in the case L'=0 with a large slope.

In the case of $p\approx A/s$, assuming that $0<|e|<s$, e is between the linear part and the threshold part of f(x). Thus, in first approximation e_1 # $e_2 = s/(1+p/f'(e))$, b_1 # $b_2 = 1/f'(e)$.

The influence of p on the cycle points are shown in figure 7.

3. CONCLUSION

The behavior of adaptive recursive linear and nonlinear filters when adapted with the FMRLMS(L,L') algorithms was investigated in a simple case. In the linear case or nonlinear case with a small slope at the origine, the larger L and L', the faster and more accurate the convergence. This is not the case in the nonlinear case. Note, however, that increasing L and L' can be interesting in nonstationary environments or when the system is close to its stability limit; it is to say, each time that b(n) may be very different from b(n+1). At last, note that neither in the linear case, nor in the nonlinear case the FMRLMS algorithms give the same results when L'=0 and L'≠0: we found that the minimum of the mean square output is reached more accurately for L'≠0.

REFERENCES

[1] J.J.Shynk, "Adaptive IIR filtering", IEEE ASSP Magazine, pp.4-21, 1989.
[2] S.Marcos et al., "A unified framework for gradient algorithms used for filter adaptation and neural network training", Int. Journal of Circuit Theory and Appli., to appear
[3] O.Nerrand et al., "Feedback neural networks for nonlinear adaptive filtering", Proceedings of NEURO-NIMES 91, pp.753-756, 1991.
[4] C.Vignat, S.Marcos, O.Macchi, "Une classe d'algorithmes d'adaptation de filtres récursifs présentant des propriétés stabilisantes", Proc. GRETSI, p.497-500, 1991.
[5] M.Jaidane, O.Macchi, "Stability of adaptive recursive filters", Proc. ICASSP 88, pp.1503-1505, 1988.

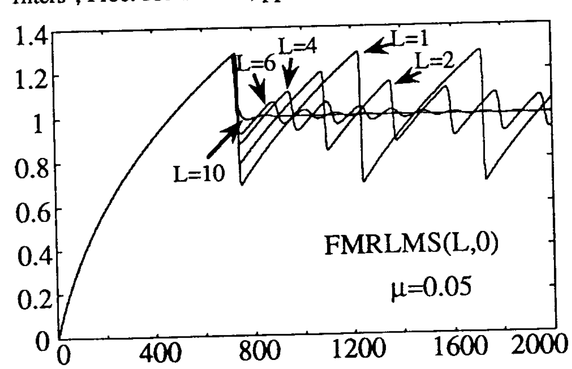

Figure 1 : FMRLMS(L,0): Linear case.

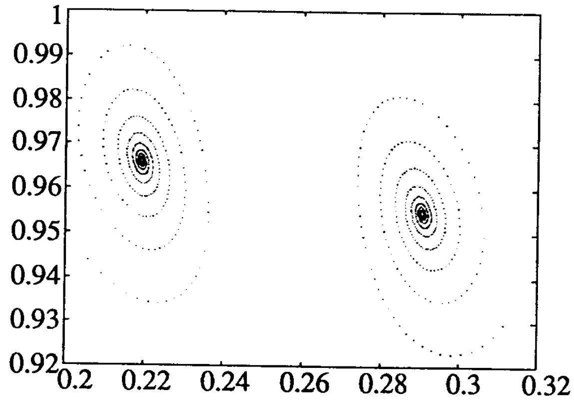

Figure 2 : FMRLMS(2, 2): linear case.
s=0.5, μ=0.05

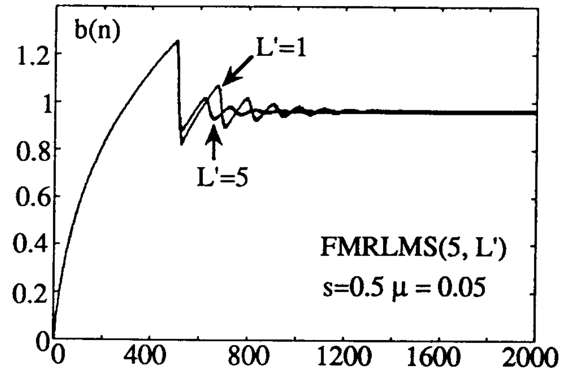

Figure 3 : FMRLMS(5, L'): linear case.

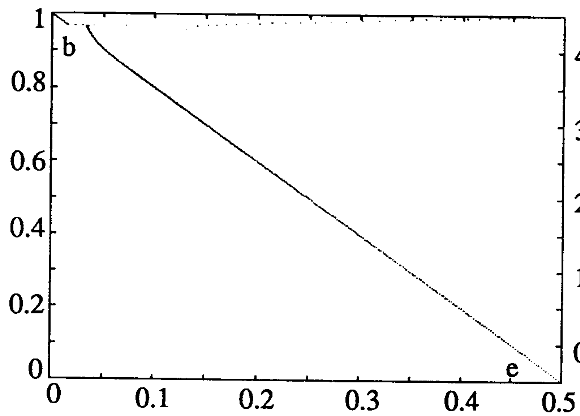

Figure 4 : FMRLMS(2, 0): nonlinear case.
A=0.5, p=30, s=0.5, μ=0.05

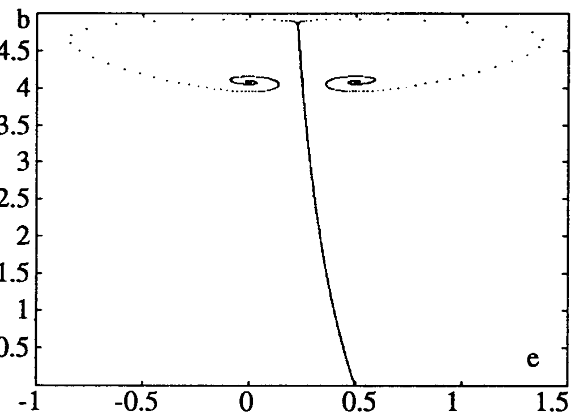

Figure 5 : FMRLMS(2, 0): nonlinear case.
A=0.5, p=0.25, s=0.5, μ=0.05

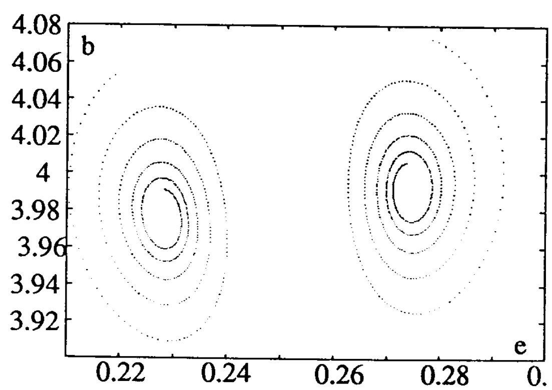

Figure 6 : FMRLMS(3, 3): nonlinear case.
A=0.5, p=0.25, s=0.5, μ=0.05

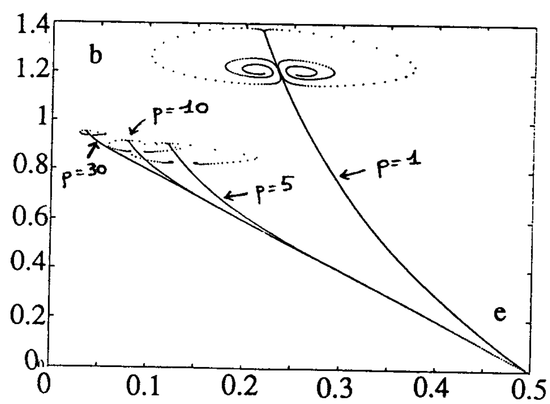

Figure 7 : FMRLMS(4, 4): nonlinear case.
A=0.5, μ=0.05