

ADAPTIVE RECURSIVE HIGHER ORDER POLYNOMIAL FILTER

Farhat Fnaiech* ; Mounir Sayadi* *Student Member, IEEE*, and Mohamed Najim **,
Fellow, IEEE

* E.S.S.T.T., Université de Tunis,
5 Av. Taha Hussein, 1008, Tunis, Tunisia, Tel 2161496066, Fax 2161391166

E mail :Farhat.Fnaiech@esstt.rnu.tn

** Equipe Signal & Image, ENSERB,

Av. Albert Schweitzer, BP 99, F33402 Talence cedex, France

Tel +33556846674, Fax +33556848406, E mail : najim@goelette.tsi.u-bordeaux.fr

KEY WORDS: *Recursive nonlinear filter, LMS algorithm, approximate gradient, exact gradient, Volterra (IIR) filter, stability.*

ABSTRACT :

In this paper we present an adaptive recursive nonlinear filter based on the higher order Volterra series infinite impulse response (IIR) structure. The proposed work is an extension of the work of Roy et al [9] in which they limit the Volterra recursive filter to the second order. Adaptive algorithms based on the exact and the approximate gradient will be derived. Stability issues based on the Extended Kalman Filter will be discussed. Simulations results highlight the usefulness of the proposed algorithms. Improvement on the convergence and stability behavior will be evaluated.

1. Introduction

Over the last decade, Volterra filters (FIR) or polynomial filters [6] and nonlinear adaptive infinite impulse-response (IIR) filters have been appealing areas of research and have been considered in many real world applications [12][1]. While Volterra filters have been applied in many applications they still present some limitations because of their computational complexity which increases exponentially with the filter order [9]. As an alternative to using these filters Roy et al in [9] proposed recursive second-order polynomial filters in their study which is limited to the second order. In this paper we shall consider higher order recursive polynomial filters and shall develop the corresponding adaptive gradient based algorithms. In the second part of this paper stability issues of the recursive polynomial filter

using the recursive prediction error algorithm [2] with the Extended Kalman filter will be stated.

2.General adaptive IIR Volterra filters:

The general discrete NARMA [1] [3] [7] model is defined by:

$$\begin{aligned}
 y(n) = & \sum_{n_1=0}^{\infty} a_{n_1} x(n - n_1) + \\
 & \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} a_{n_1 n_2} x(n - n_1) x(n - n_2) + \dots + \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \dots \\
 & \dots \sum_{n_p=0}^{\infty} a_{n_1 n_2 n_3 \dots n_p} x(n - n_1) x(n - n_2) \dots x(n - n_p) + \\
 & \sum_{m_1=1}^{\infty} b_{m_1} y(n - m_1) + \sum_{m_1=1}^{\infty} \sum_{m_2=1}^{\infty} b_{m_1 m_2} y(n - m_1) y(n - m_2) \\
 & + \dots + \sum_{m_1=1}^{\infty} \sum_{m_2=1}^{\infty} \dots \sum_{m_q=1}^{\infty} b_{m_1 m_2 \dots m_q} \\
 & y(n - m_1) y(n - m_2) \dots y(n - m_q) \\
 & + \sum_{n_1=0}^{\infty} \sum_{m_1=1}^{\infty} c_{n_1 m_1} x(n - n_1) y(n - m_1) + \dots \\
 & + \sum_{n_1=0}^{\infty} \dots \sum_{n_p=0}^{\infty} \sum_{m_1=1}^{\infty} \dots \sum_{m_q=1}^{\infty} c_{n_1 n_2 \dots n_p m_1 m_2 \dots m_q} \\
 & x(n - n_1) \dots x(n - n_p) y(n - m_1) \dots y(n - m_q)
 \end{aligned} \tag{1}$$

where $x(n)$ represents the discrete input signal, $y(n)$ is the discrete recursive output signal. Note here that the memory of the recursive and the non recursive parts is infinite. However for computational purposes, in

practice, we may truncate the series to finite order. The adaptive nonlinear IIR filtering problem consists of adapting the filter coefficients:

$$a_0(n), \dots, a_N(n), \dots, a_{NN}(n), \dots, b_1(n), \dots, \dots, \\ b_{11\dots1}(n), \dots, c_{01}(n), \dots, c_{0\dots01\dots1}(n), \dots, c_{N\dots NM\dots M}(n)$$

The objective of the adaptive filter is to exhibit a good estimation $\hat{y}(n)$ of the reference signal $d(n)$. Equation (1) may written in the matrix form as:

$$y(n) = H^T(n)X(n) \quad (2)$$

$$H^T(n) = [a_0(n), \dots, a_N(n), a_{0,0}(n), \dots, a_{NN}(n), \\ a_{000\dots}(n), \dots, a_{010\dots}(n), b_1(n), b_N(n), \dots] \quad (3)$$

$$b_{11}(n), \dots, b_{11\dots1}(n), c_{01}(n), \dots, c_{0\dots01\dots1}(n), \dots] \\ X^T(n) = [x(n), x(n-1), \dots, x^2(n), x(n)x(n-1), \dots] \quad (4)$$

2.1.Gradient method

The estimated gradient is given by:

$$\hat{\nabla}(n) = \frac{\partial e^2(n)}{\partial H(n)} = -2e(n) \frac{\partial y(n)}{\partial H(n)} = -2e(n)H_f(n) \quad (5)$$

where:

$$H_f(n) = \left[\frac{\partial y(n)}{\partial a_0(n)}, \dots, \frac{\partial y(n)}{\partial a_N(n)}, \dots, \frac{\partial y(n)}{\partial a_{0,0}(n)}, \dots \right] \quad (6)$$

$$\left[\frac{\partial y(n)}{\partial a_{N,N}(n)}, \dots, \frac{\partial y(n)}{\partial b_1(n)}, \dots, \frac{\partial y(n)}{\partial c_{0\dots1\dots}(n)} \right]$$

$$\frac{\partial y(n)}{\partial a_i(n)} = x(n-i) + \psi_{a_i}(n)$$

$$\frac{\partial y(n)}{\partial a_{i,j}(n)} = x(n-i)x(n-j) + \psi_{a_{i,j}}(n)$$

$$\frac{\partial y(n)}{\partial c_{m_1\dots m_q m_1\dots m_q}(n)} = x(n-n_1)\dots x(n-n_p)$$

$$y(n-m_1)\dots y(n-m_q) + \psi_{c_{m_1\dots m_q m_1\dots m_q}}(n)$$

where:

$$\psi_v(n) = \sum_{m_1=1}^{\infty} b_{m_1}(n) \frac{\partial [y(n-m_1)]}{\partial v(n)} +$$

$$\sum_{m_1=1}^{\infty} \sum_{m_2=1}^{\infty} b_{m_1 m_2}(n) \frac{\partial [y(n-m_1)y(n-m_2)]}{\partial v(n)} + \dots +$$

$$\sum_{m_1=1}^{\infty} \dots \sum_{m_q=1}^{\infty} b_{m_1\dots m_q}(n) \frac{\partial [y(n-m_1)\dots y(n-m_q)]}{\partial v(n)} + \dots \\ \dots + \sum_{n_1=0}^{\infty} \sum_{m_1=1}^{\infty} c_{n_1 m_1}(n) x(n-n_1) \frac{\partial [y(n-m_1)]}{\partial v(n)} + \dots + \quad (7)$$

2.2.Exact Gradient

By assuming low variations of the adaptive parameter filter it follows: $H(n) \approx H(n-1) \approx \dots \approx H(n-N) \approx \dots$

this implies that in equation (7) we shall replace $v(n)$ by $v(n-1)$. The resulting expression is as follows:

$$\psi_v(n) \approx \sum_{m_1=1}^{\infty} b_{m_1}(n) \frac{\partial [y(n-m_1)]}{\partial v(n-m_1)} + \\ \sum_{m_1=1}^{\infty} \sum_{m_2=1}^{\infty} b_{m_1 m_2}(n) \frac{\partial [y(n-m_1)y(n-m_2)]}{\partial v(n-m_1)} + \dots + \\ \sum_{m_1=1}^{\infty} \dots \sum_{m_q=1}^{\infty} b_{m_1\dots m_q}(n) \frac{\partial [y(n-m_1)\dots y(n-m_q)]}{\partial v(n-m_1)} + \dots \\ + \sum_{n_1=0}^{\infty} \sum_{m_1=1}^{\infty} c_{n_1 m_1}(n) x(n-n_1) \frac{\partial [y(n-m_1)]}{\partial v(n-m_1)} + \dots + \quad (8)$$

$$\sum_{n_1=0}^{\infty} \dots \sum_{m_p=0}^{\infty} \sum_{m_1=1}^{\infty} \dots \sum_{m_q=1}^{\infty} c_{n_1\dots n_p m_2\dots m_q}(n) x(n-n_1)\dots x(n-n_p) \\ \frac{\partial [y(n-m_1)\dots y(n-m_q)]}{\partial v(n-m_1)}$$

2.3. Approximate Gradient

The approximate gradient is given for $\psi_v(n) = 0$. This implies that $H_f(n) = X(n)$.

2.4. Simulation results using the LMS adaptive IIR Volterra filtering based on the exact and approximate gradient

In order to compare the LMS adaptive algorithm based on both approaches we have used the following IIR Volterra filter in a system identification experiment:

$$y(n) = 0.8y(n-1) - 0.5y(n-2) +$$

$$0.7y^2(n) - 0.4y(n)y(n-1) + 0.1y^2(n-1) + x(n)$$

where $x(n)$ is a white gaussian noise of variance 0.01. the learning parameter is

$$\mu = 10^{-4} \cdot ER(dB) = 10 \log \frac{\|H - H_0\|^2}{\|H_0\|^2},$$

where H_0 is the desired vector coefficients.

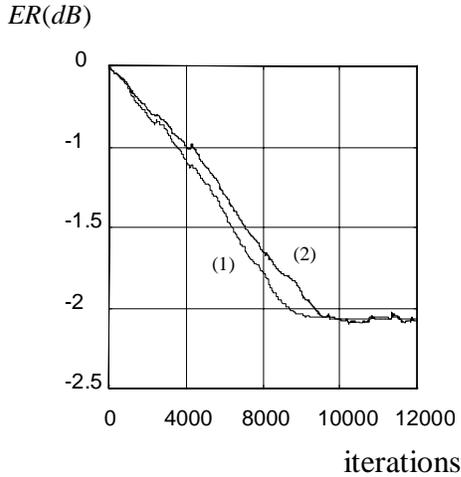


Figure1:(1) exact gradient
(2) approximate gradient

3. Recursive prediction error method with Extended Kalman filter for stabilizing the adaptive IIR filter:

3.1 Problem formulation:

In system identification experiment and in the case of stochastic additive noise, unstable behavior of the recursive adaptive Volterra filter may occur. The recursive prediction error method with the Extended Kalman filter for stabilizing bilinear filter purpose has been used in [2]. In this section similar materials will be developed to the recursive IIR Volterra filter.

3.2 Stability of the predictor:

Let us first choose a simple IIR filter given by:

$$\begin{cases} z(n) = a z(n-1) + b z(n-2) + c x(n-1) + dx(n-2) \\ \quad \quad \quad + g z(n-1)x(n-1) \\ y(n) = z(n) + v(n) \end{cases} \quad (9)$$

In this system the noise enters in additive and multiplicative terms and hence the standard RLS algorithm gives biased estimates. The instrumental variable and the recursive prediction error (RPEM) [4] give best results. The RPEM shows its superiority in simulation but shows some outliers.

	$\hat{a}(100)$	$\hat{b}(100)$	$\hat{c}(100)$
True	1.5	-0.7	1.0
RLS	1.21±0.05	-0.49±0.04	0.89±0.12
RIV	1.47±0.19	0.14±0.13	0.46±0.10
RELS	1.38±0.07	-0.58±0.08	0.80±0.13
RPEM	1.41±0.13	0.64±0.09	0.62±0.90
RPEM*	1.46±0.04	-0.67±0.03	0.92±0.14

	$\hat{d}(100)$	$\hat{g}(100)$
True	0.5	0.12
RLS	0.76±0.11	0.01±0.04
RIV	0.57±0.08	0.03±0.10
RELS	0.93±0.31	0.092±0.05
RPEM	1.11±1.50	0.12±0.03
RPEM*	0.62±0.25	0.12±0.03

Table1: Monte Carlo test (10 realizations of the input signal)for system identification methods to the Volterra filter equation (9). RPEM*(9 realizations) is the RPEM when outlier is discarded. The estimation are given ±the standard deviation, note that the outlier appears in both the parameters $\hat{b}(100) = -0.64 \pm 0.09$ and $\hat{d}(100) = 1.11 \pm 1.50$.

To analyze the stability of the RPEM, it is suitable to consider the state space model of equation (9). Thus this input/ output model may be expressed as:

$$\begin{cases} T(n+1) = AT(n) + DT(n)x(n) + Bx(n) + v(n) \\ y(n) = CT(n) + e(n) \end{cases} \quad (10)$$

where:

$$T(n) = \begin{bmatrix} z(n) \\ z(n-1) \\ x(n-1) \end{bmatrix}, A = \begin{bmatrix} a & b & d \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} c \\ 0 \\ 1 \end{bmatrix},$$

$$D = \begin{bmatrix} g & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, C = [1 \ 0 \ 0], v(n)=0$$

It is clear that the matrices A , B and D depend of the parameter estimation vector H . In the sequel we shall denote these matrices by $A(H)$, $B(H)$ et $D(H)$.

In the case of $e(n)$ white noise and $v(n)=0$ the predictor of system (10) may written by:

$$\begin{cases} \hat{T}(n+1, H) = [A(H) + D(H)x(n)]\hat{T}(n, H) + B(H)x(n) \\ \hat{y}(n/H) = C\hat{T}(n/H) \end{cases} \quad (11)$$

The gradient of this predictor may be determined by:

$$\psi_i(n, H) = \frac{\partial}{\partial H_i} \hat{y}(n, H) = C \hat{T}_i'(n, H)$$

where:

$$\begin{aligned} \hat{T}_i'(n+1, H) = & [A(H) + D(H)x(n)] \hat{T}_i'(n, H) + \\ & \left[\frac{\partial}{\partial H_i} A(H) \right] \hat{T}(n, H) + \left[\frac{\partial}{\partial H_i} D(H) \right] \hat{T}(n, H)x(n) \\ & + \left[\frac{\partial}{\partial H_i} B(H) \right] x(n) \end{aligned} \quad (12)$$

This equation represents the gradient used in the RPEM algorithm. The stability of this gradient depends strongly on the matrix function:

$$F_n = A(\hat{H}(n)) + D(\hat{H}(n))x(n) \quad (13)$$

The outlier appears when the value of F_n yields the poles of equation (11) to be outside the unit circle for a certain input realization.

3.3 Time varying Kalman Filter

In order to stabilize the predictor of the time varying linear system (11) having the system matrix $F(n, H) = A(H) + D(H)x(n)$ we shall use the time varying Kalman Filter [5]:

$$\begin{aligned} \hat{T}(n+1, H) = & F(n, H) \hat{T}(n, H) + B(H)x(n) \\ & + K(n, H)[y(n) - C \hat{T}(n, H)] \end{aligned} \quad (14)$$

$$\hat{y}(n/H) = C \hat{T}(n, H) \quad (15)$$

$$K(n, H) = [F(n, H)P(n, H)C^T]S^{-1}(n, H) \quad (16)$$

$$P(n+1, H) = F(n, H)P(n, H)F^T(n, H) + R_1 \quad (17)$$

$$- K(n, H)S^{-1}(n, H)K^T(n, H) \quad (18)$$

$$S(n, H) = CP(n, H)C^T + R_2$$

with: $R_1 = E(x(n)x^T(n))$, $R_2 = E(e(n)e^T(n))$ and $E(x(n)e^T(n)) = 0$. Note that the term $K(n, H)$ in equation (14) depends on the parameter vector H it should have effect on the gradient $\psi_i(n, H)$. In the extended Kalman filter this effect is neglected see [5].

3.4 Extended Kalman Filter (EKF)

Here the extended Kalman Filter uses both the RPEM for the parameter identification procedure together with the time varying Kalman Filter for state estimation:

$$\hat{H}(n) = \hat{H}(n-1) + R^{-1}(n)\psi(n)(y(n) - \hat{y}(n)) \quad (19)$$

$$R(n) = \lambda(n)R(n-1) + \psi(n)\psi^T(n) \quad (20)$$

$$\hat{T}(n+1) = (F_n - K_n C) \hat{T}(n) + B_n x(n) + K_n y(n) \quad (21)$$

$$\psi_i(n+1) = (F_n - K_n C) \psi_i(n) +$$

$$\frac{\partial}{\partial H_i} [F(n, H)] \Big|_{H=\hat{H}(n)} \hat{T}(n) + \frac{\partial}{\partial H_i} [B(H)] \Big|_{H=\hat{H}(n)} x(n) \quad (22)$$

$$F_n = F(n, \hat{H}(n)) = A(\hat{H}(n)) + D(\hat{H}(n))x(n) \quad (23)$$

$$K_n = [F_n P(n) C^T] S_n^{-1} \quad (24)$$

$$P(n+1) = F_n P(n) F_n^T + R_1 - K_n S_n^{-1} K_n^T \quad (25)$$

$$S_n = CP(n)C^T + R_2 \quad (26)$$

To include the effect of K_n in the EKF we should replace equation (22) by:

$$\begin{aligned} \psi_i(n+1) = & (F_n - K_n C) \psi_i(n) + \frac{\partial}{\partial H_i} [F(n, H)] \Big|_{H=\hat{H}(n)} \hat{T}(n) + \\ & \frac{\partial}{\partial H_i} [B(H)] \Big|_{H=\hat{H}(n)} x(n) + \frac{\partial}{\partial H_i} [K(n, H)] [y(n) - C \hat{T}(n)] \end{aligned} \quad (27)$$

where the expression $\frac{\partial}{\partial H_i} [K(n, H)]$ is

obtained from the derivation of equations (14)-(16). In conclusion the algorithm given by the equations (19)-(21) and (23)-(27) is the stabilized EKF for the IIR Volterra filter given by equation (9).

3.5 Simulation results for the EKF

To highlight the stability efficiency of the EKF we have considered a nonlinear system with a strong nonlinear term which may give an unstable behavior. The system is given by:

$$T(n+1) = aT(n) + bT(n)x(n-1) + cx(n)$$

$$y(n) = T(n) + e(n)$$

$$(28)$$

where $\{e(n)\}$ and $\{x(n)\}$ are respectively the white noise and the input signal which are independent of variances respectively unity and two. The values of the true parameters are $a=1$, $b=-0.7$, $c=0.5$ Figure 2 and Figure 3 show the estimation of the parameter "c" respectively by using the time varying Kalman Filter equations (14)-(18) and the EKF with RPEM equations $\{(19)-(21);(23)-(27)\}$. Note that the RPEM applied alone to the system (28) exhibits unstable behavior.

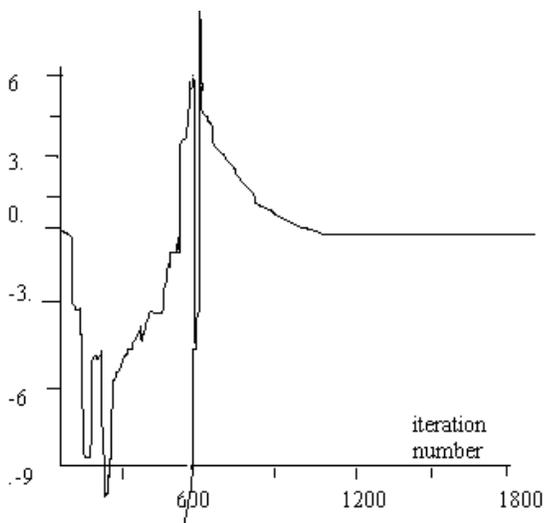


Figure2: Estimated parameter "c" with time varying KF equations (14)-(18).

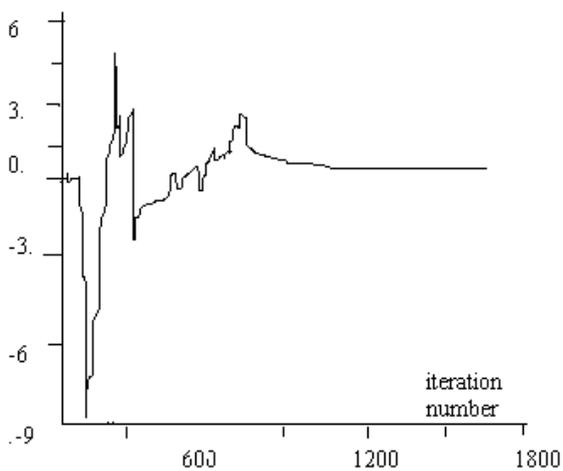


Figure3: Estimated parameter "c" with EKF(RPEM) equations {(19)-(21);(23)-(27)}

4 Conclusion

In this paper we have presented adaptive IIR Volterra filters. The exact and the approximate gradient adaptive algorithms have been developed and applied to IIR Volterra filter. Stability issues of these nonlinear filters have been discussed. In order to study the stability of an input/output IIR Volterra filter, we have switched to the state space representation and found out that the EKF linked with the RPEM may stabilize the adaptive IIR algorithm. Clearly this algorithm is computationally costly since the dimension of the state space depends upon the number of the filter parameters. Further work are led to search for other more efficient stabilizing methods for Volterra filters.

5 References

- [1] S.A. Billings and W.S.F. Voon," Least square parameter estimation algorithms for nonlinear systems," *Int. J. System Sci.* Vol 15. pp.601-615, 1984.
- [2] F. Fnaiech and L. Ljung," Recursive identification of bilinear systems," *Int. J. Control*, vol 45, no 2 , pp.453-470, 1987.
- [3] F. X. Gao and W. Martin Snelgrove," Adaptive nonlinear recursive state space filters," *IEEE Trans. Circuits and Systems* Vol CAS 41. no 11. pp.760-764, November 1994.
- [4] L. Ljung, " System identification theory for the user," *Prentice Hall* 1987.
- [5] L. Ljung and T. Söderström," Theory and practice of recursive identification," *Cambridge MIT press*, 1983.
- [6] V. J. Mathews," Adaptive Polynomial Filters," *IEEE SP Magazine*, pp.10-26, July 1991.
- [7] S.R. Parker and F.A.Perry," A discrete ARMA model for nonlinear system identification," *IEEE Trans. Circuits and Systems*. Vol. CAS 28. no 3. pp.224-233. March 1981.
- [8] M. B. Priestley," Nonlinear and non stationary time series analysis," *Academic Press Limited* 1991.
- [9] E. Roy, R. W. Stewart and T. S. Durrani," Higher order system identification with an adaptive recursive second order polynomial filter," *IEEE SP Letters*, Vol. 3,n°10, pp 276-279,1996.
- [10] M. Sayadi, F. Fnaiech and M. Najim, "Multichannel linear and quadratic adaptive filtering based on the Chandrasékhar fast algorithm," *IEEE Trans. on Signal Processing*. Vol. 47, no 3,1999.
- [11] M. Sayadi, F. Fnaiech_ and M. Najim,"An LMS adaptive second order Volterra filter with a zeroth-order term : steady-state performance analysis in a time-varying environment," *IEEE Trans. on Signal Processing*. Vol 47, no 3,1999.
- [12] T. C. Tsai and D. Anastassiou," Nonlinear recursive filters and applications to image processing," *Proceedings of the IEEE ICASSP'88*, pp. 828-831, NewYork, April, 1988.