COMPARATIVE STUDY OF NEW VERSIONS OF THE NEWTON TYPE ADAPTIVE FILTERING ALGORITHM

Mohamed DJENDI[1], *Mohamed RAHIM*[1], *Abderrezak GUESSOUM*[1], *Martin BOUCHARD*[2], *Daoued BERKANI*[3]

[1] : University Saad Dahleb of BLIDA, Signal Processing and Imaging Laboratory (LATSI), Route de Soumaa B.P.270 BLIDA 09000, ALGERIA, e-mails: *{m_djendi, m_rahim, guessouma}@hotmail.com*

[2]: School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa (Ontario), KIN 6N5, e-mail: *bouchard@site.uottawa.ca*

[3]: Electrical Engineering Department, Ecole Nationale Polytechnique, 10 Avenue Hassan Badi, B.P.182 El-Harrach, AGEIRS 16000, ALGERIA, e-mail: *dberkani@hotmail.com*

ABSTRACT

In this work, we present five new versions of the Fast Newton Transversal Filter algorithm (FNTF). The first algorithm is based on a simple modification of the filtering part, by introducing a scalar accelerator parameter. The second algorithm is based on the use of a block filtering technique to actualize the local filter coefficients. The third algorithm is a modification of the second algorithm by the use of the final filtering errors to actualize the filters coefficients. The fourth and the fifth algorithms are based respectively on the combination of features from the first algorithm with features of the second and third algorithms. These five new algorithms were proposed to improve the convergence speed of the original version of the FNTF algorithm for the identification of acoustic impulse responses, and to also improve their tracking ability when these systems vary in time. The different algorithms and their comparative results are presented.

1. INTRODUCTION

The increasing power of Digital Signal Processors (DSPs) and VLSI technology allows the use of very long adaptive filters in applications such as automatic control, system identification, channel equalization, interference rejection, echo cancellation, etc. The task is to estimate the filter response in such a way that for a given input signal, its output tracks a desired response signal in an optimal way. Several types of adaptive filtering algorithms have been proposed in order to get a much faster convergence than the Normalized Least-Mean-Squares (NLMS) adaptive filter, when used for system identification with correlated inputs like speech [1]. One of these algorithms is the FNTF algorithm, which produces a performance similar to Fast Recursive Least-Squares (FRLS) transversal adaptive filters. Indeed, the convergence rate of the NLMS depends on the input signal statistics, and this signal is implicitly modeled as a white noise sequence. In standard FRLS algorithms, an autoregressive modeling of the input signal is performed in the prediction part, and consequently, the input statistics no longer affect the convergence. This AR modeling of order L (L being the order of the filter to be estimated) amounts to 6L multiplies per sample. Recalling that in some applications, like acoustic echo cancellation, real time identification of long impulse responses is required, the corresponding FRLS computational load is prohibitive [2]. Assuming an autoregressive input of order N, with N much smaller than L (which is generally true for speech), a large complexity reduction can be achieved by using the FNTF algorithm [2]. In this paper, we propose five new versions of the FNTF algorithm. A comparative study is performed for the convergence and tracking performance of the five proposed versions.

2. DERIVATION OF THE ORIGINAL VERSION OF THE FNTF ADAPTIVE ALGORITHM

Let us first recall the filtering part equation of the FNTF

$$\hat{y}_t = H_{L,t-1}^T X_{L,t}, \text{ echo estimation} \tag{1}$$

$$\bar{\varepsilon}_{L,t} = y_t - \hat{y}_t, \text{ error estimation} \tag{2}$$

$$H_{L,t} = H_{L,t-1} - \tilde{C}_{L,t}\gamma_{L,t}\bar{\varepsilon}_{L,t}, \text{ filter update} \tag{3}$$

where ^{T} denotes matrix transpose. The impulse response is modeled by the vector $H_{L,t}$ of size L, $\hat{y}_t$ is the estimated output signal, and $\bar{\varepsilon}_{L,t}$ is the a priori error computed before adaptation. The vector $X_{L,t}$ contains the last L input samples, $X_{L,t} = [x_t, x_{t-1}, \ldots, x_{t-L+1}]$. The dual Kalman gain $\tilde{C}_{L,t}$ and the likelihood variable $\gamma_{L,t}$ are defined as:

$$\tilde{C}_{L,t} = -\lambda^{-1} R_{L,t-1}^{-1} X_{L,t} \quad (4), \qquad \gamma_{L,t} = \frac{1}{1 - \tilde{C}_{L,t}^T X_{L,t}} \quad (5)$$

where $R_{L,t}$ is the LxL short-term covariance matrix of the input signal. In the exponentially weighted least-squares (LS) case, the covariance matrix is updated by the following recursive equation:

$$R_{L,t} = \lambda R_{L,t-1} + X_{L,t} X_{L,t}^T, \quad 0 < \lambda < 1 \tag{6}$$

The FNTF algorithm [1] is built around an efficient recursive estimation of $\tilde{C}_{L,t}$ and $\gamma_{L,t}$, which are derived from the two following equivalent forms of $R_{L+1,t}^{-1}$:

$$R_{L+1,t}^{-1} = \begin{bmatrix} R_{L,t}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -b_{L,t} \\ 1 \end{bmatrix} \begin{bmatrix} -b_{L,t} & 1 \end{bmatrix} \beta_{L,t}^{-1} \tag{7}$$

$$R_{L+1,t}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & R_{L,t-1}^{-1} \end{bmatrix} + \begin{bmatrix} 1 \\ -a_{L,t} \end{bmatrix} \begin{bmatrix} 1 & -a_{L,t} \end{bmatrix} \alpha_{L,t}^{-1} \quad (8)$$

where $a_{L,t}$ and $b_{L,t}$ are forward and backward predictors of order L, and $\alpha_{L,t}$ and $\beta_{L,t}$ are the power of the forward and backward predictor errors. To illustrate the FNTF derivation, let us take a simple example, that is N=L-1. Assuming that the input signal is autoregressive of order L-1, we want to find a positive definite matrix $R_{L+1,t}$ in two partitioned forms to exhibit the elements to be extrapolated. We get [2]:

$$R_{L+1,t}^{a} = \begin{bmatrix} \Gamma_{0,t}^{a} & \Gamma_{L,t-1}^{aT} \\ \Gamma_{L,t}^{a} & \Gamma_{L,t-1} \end{bmatrix}, \quad R_{L+1,t}^{b} = \begin{bmatrix} R_{L,t} & \Gamma_{L,t-1}^{b} \\ \Gamma_{L,t}^{bT} & \Gamma_{0,t}^{b} \end{bmatrix} \quad (9)$$

with $\Gamma_{0,t}^{a} = r_{0,t}$, $\Gamma_{0,t}^{b} = r_{0,t-L}$, $\Gamma_{L,t}^{aT} = \left[r_{1,t}, r_{2,t}, ..., \hat{r}_{L,t}^{a} \right]$, and $\Gamma_{L,t}^{bT} = \left[\hat{r}_{1,t}^{b}, r_{L-1,t-1}, ..., r_{1,t-L+1} \right]$ which are estimated according to $r_{i,t} = \lambda r_{i,t} + x_t x_{t-1}$ for $0 \le i \le L-1$. Note that $\hat{r}_{L,t}^{a}$ and $\hat{r}_{L,t}^{b}$ are the only unknown elements. Using the previous notation, the LS optimum forward predictor and the error variance of order L are given respectively by:

$$a_{L,t} = R_{L,t-1}^{-1} \Gamma_{L,t}^{a} \quad (10)$$

$$\alpha_{L,t} = \Gamma_{0,t}^{a} - \Gamma_{L,t}^{0T} a_{L,t} \quad (11)$$

In (11), the minimum forward power depends on the unknown element $\hat{r}_{L,t}$. The basic idea underlying the FNTF algorithm is to compute this unknown element by maximizing $\alpha_{L,t}$. That is, we choose $\hat{r}_{L,t}^{a}$ to be the worst possible choice with respect to the forward predictor estimate. This criterion is similar to the Maximum entropy principle for extrapolating the autocorrelation sequence of a stationary process [3]. To achieve this maximization, the trick is to take into account the growing order formulation of (10),(11), that we can get from the RLS lattice filter theory:

$$a_{L,t} = \begin{bmatrix} a_{L-1,t} \\ 0 \end{bmatrix} + \begin{bmatrix} b_{L-1,t-1} \\ -1 \end{bmatrix} K_{L,t}^{a} \quad (12)$$

$$\alpha_{L,t} = \alpha_{L-1,t} - \beta_{L-1,t-1} (K_{L,t}^{a})^{2} \quad (13)$$

where the reflection coefficient $K_{L,t}^{a}$ is defined by:

$$K_{L,t}^{a} = -\frac{1}{\beta_{L-1,t-1}} (\hat{r}_{L,t}^{a} - \Gamma_{L-1,t-1}^{bT} a_{L-1,t}) \quad (14)$$

Indeed, since the variance of the backward error $\beta_{L-1,t-1}$ is positive, $\alpha_{L,t}$ takes its maximum value when $K_{L,t}^{a}$ is set to zero, i.e.: $\hat{r}_{L,t}^{a} = \Gamma_{L-1,t-1}^{b,t} a_{L-1,t}$ (15)

Then, choosing this optimum value, the forward predictor and the minimum error power of the extrapolated process are:

$$a_{L,t} = \begin{bmatrix} a_{L-1,t} \\ 0 \end{bmatrix} \quad (16), \qquad \alpha_{L,t} = \alpha_{L-1,t} \quad (17)$$

Applying the same approach to backward parameters, i.e. maximizing $\beta_{L,t}$ with respect to $\hat{r}_{L,t}^{b}$, we find that $\hat{r}_{L,t}^{b} = \hat{r}_{L,t}^{a}$ and:

$$b_{L,t} = \begin{bmatrix} 0 \\ b_{L-1,t-1} \end{bmatrix} \quad (18), \qquad \beta_{L,t} = \beta_{L-1,t-1} \quad (19)$$

The results show that $R_{L+1,t}^{a} = R_{L+1,t}^{b}$ can be extrapolated from $R_{L,t}$, and moreover the corresponding predictors are given by (16),(17) or (18),(19). These predictors of order L are equal to the optimal predictor of order L-1 extended with a zero. Let us now come back to the general case, that is extrapolation from N to L. Following a similar reasoning, it is possible to extrapolate $R_{N+1,t}$ up to $R_{L+1,t}$ in a recursive manner. Applying (16),(17) and (18),(19) recursively from L to N, we obtain:

$$a_{L,t} = \begin{bmatrix} a_{N,t} \\ 0_{L-N} \end{bmatrix} \quad (20), \qquad \alpha_{L,t} = \alpha_{N,t} \quad (21)$$

$$b_{L,t} = \begin{bmatrix} 0_{L-N} \\ b_{N,t_D} \end{bmatrix} \quad (22), \qquad \beta_{L,t} = \beta_{N,t_D} \quad (23)$$

Using these truncated predictors instead of full order predictors in (7),(8) and following the classical derivation of the FRLS, we can find update equations for the extrapolated dual Kalman gain and the likelihood variable. All quantities of order N are computed by solving the prediction problem of order N [2],[3].

3. THE NEW VERSIONS OF THE NEWTON ALGORITHM

3.1. Algorithm 1

In order to improve the performance of the FNTF algorithm of Section 2 in the case of non-stationary channels and inputs, we introduce simple modifications which don't impair the nice numerical properties of the FNTF. The tracking behaviour is improved by controlling the adaptation gain in the filtering part [4]. The equation of the transversal filter coefficients update (3) becomes as follows :

$$H_{L,t} = H_{L,t} - \frac{1}{1-\mu\gamma_{L,t}} C_{L,t} \bar{\varepsilon}_{L,t} \quad (24)$$

where the term of $C_{L,t} = \gamma_{L,t} \tilde{C}_{L,t}$, and $\gamma_{L,t}$ is provided by the prediction part of the FNTF algorithm. The new introduced parameter μ is a control parameter which must be chosen close to 1 in order to ensure the convergence of the modified algorithm [4]. When μ is chosen close to 1, the adaptation gain is increased, therefore, the tracking and convergence speeds are improved (this is paid in return by a greater sensitivity to the output noise). Thus, the choice of the forgetting factor λ, which is related to the numerical stabilization method [1],[6] (e.g. λ>1-1/(2L+3.5)), is more dependent on tracking considerations. Locking of the FNTF algorithm can be observed with non-stationary input signals like speech. This locking comes from the temporarily poorly exciting characteristics of the input signal. It can be prevented by reinitialising properly the algorithm when a tendency to locking is detected; the detection can be done easily from inspection of the likelihood variable (5) which then goes to zero very quickly [5].

3.2. Algorithm 2

This algorithm is based on a modification which allows to obtain a prediction structure working with a forgetting factor λ lower than that of the basic initial structure. A new structure was applied which makes it possible to substitute for the global algorithm P algorithms of equal or unequal lengths working each one on a reduced number of parameters [6]. This technique yields to carry out P FNTF algorithms working as in the structure represented in Fig. 1, for the case with μ=0. The P adaptive algorithms for the filtering of y_t are adapted by local filtering errors. The equations update of the P transversal filters are:

$$H_{L_i,t} = H_{L_i,t-1} - \bar{\varepsilon}_{i,t} C_{L_i,t} \quad , \ i = 1,2,\ldots,P \tag{25}$$

where $\sum_{i=1}^{P} L_i = L$

We note that the Kalman gain of section i, with dimension L_i, is calculated by a FNTF with an exponential forgetting factor based on L_i [1], and from the portion of signal taken at the exit of section (i-1) and the entry of the section (i+1), which corresponds to the modeling of the input signal x_t delayed by $L_1 + L_2 + \ldots L_p$ samples. We have applied this new structure to the original FNTF version. The filtering error of section i is calculated as follow.

$$\bar{\varepsilon}_{i,t} = \bar{\varepsilon}_{i-1,t} - H_{L_i,t-1}{}^T X_{L_i,t} \qquad (\varepsilon_{0,t} = y_t) \tag{26}$$

$$X_{L_i,t}^T = (x_{t-L_i-1} \ldots\ldots x_{t-L_{i-1}-L_i+1}) \qquad (L_0 = 0) \tag{27}$$

3.3. Algorithm 3

This algorithm is a modification of the previous modified FNTF algorithm (algorithm 2). Algorithm 3 is based on the use of the total (or final) filtering error for the update of the P adaptive filters. This allows to obtain a new version of the FNTF algorithm with a better performance. The basic equations of this algorithm are obtained from algorithm 2 and by substituting the local errors in (25) by the final filtering error at the output of the last cell:

$$\bar{\varepsilon}_{L,t} = y_t - \sum_{i=1}^{P} H_{i,t-1}^T X_{L_i,t} \tag{28}$$

as shown in Fig.2 for the case with μ=0.

3.4. Algorithm 4

The equations of algorithm 4 are based on the update equation of algorithm 1 and the split structure of algorithm 2 (the same structure as Fig.1). This mixing improves the performance of the resulting algorithm when it's used with time varying systems, but it doesn't improve the convergence speed for stationary cases.

3.5. Algorithm 5

This algorithm is based on an update equation as in algorithm 1 and on the structure of algorithm 3 (as in Fig. 2). Thus a scalar variable μ is introduced into the filtering parts. This version gathered the advantages of algorithms 1 and 3, which is why it has proved to be the best compared to the other versions. This version gave good results compared to the other algorithms for the tests of convergence speed with stationary acoustic channels, and the best tracking capability when tests with non-stationary acoustic channels were performed.

$$\bar{\varepsilon}_{L,t} = \bar{\varepsilon}_{L,,t-1} - H_{L_i,t-1}{}^T X_{L_i,t} \tag{29}$$

$$H_{L,t} = H_{L,t-1} - \frac{1}{1-\mu\gamma_{L_i,t}} C_{L_i,t} \bar{\varepsilon}_{L,t} \tag{30}$$

$$H_{L_i,t} = H_{L_i,t-1} - \bar{\varepsilon}_{i,t} C_{L_i,t} \qquad i = 1,2,\ldots,P \tag{31}$$

4. NUMERICAL STABILIZATION OF THE NEW ALGORITHMS

We have used a basic numerical stabilization method [1] to stabilize the five proposed algorithms. The adaptation of this method to the proposed techniques allows for each subdivision of the P adaptive filters that are generated by the new FNTF algorithms to propagate divergence indicator variables which are calculated as follow:

$$\begin{aligned} \xi_{N,t} &= \bar{r}_{N,t} + \lambda\beta_{N,t-1}\tilde{C}_{N+1,t}^{N+1} - \bar{r}_{N,t}^{\gamma} \\ &= \bar{r}_{N,t} + \mu^{\gamma}\xi_{N,t} - \bar{r}_{N,t}^{\beta} \\ &= \bar{r}_{N,t} + \mu^{\beta}\xi_{N,t} - \bar{r}_{N,t}^{b} \\ &= \bar{r}_{N,t} + \mu^{b}\xi_{N,t} \end{aligned} \tag{32}$$

where the parameters μ^{γ}, μ^{β} and μ^{b} allow the modification and the control of numerical error propagation in the algorithms. To ensure the numerical stabilization of each of the P subdivisions of the proposed new FNTF versions, the condition on the forgetting factor value for each subdivision has to be satisfied:

$$1 \succ \lambda \succ 1 - \frac{1}{2N+3.5} \quad [1],[6] \tag{33}$$

5. EXPERIMENTAL RESULTS

To test the convergence and tracking capabilities of the new FNTF algorithms, an acoustic channel variable in time was used. This 1700 samples channel represents a room with a person slowly moving in the room. The channel is measured under real conditions [6]. To evaluate the performance of each FNTF version, the following criteria was used:

$$J_t = 10 Log_{10}\left(\left\langle \bar{\varepsilon}_{L,t}^2 \right\rangle / \left\langle y_{L,t}{}^2 \right\rangle\right) \tag{34}$$

where $\left\langle \bar{\varepsilon}_{L,t}^{2} \right\rangle$ and $\left\langle y_{L,t}{}^{2} \right\rangle$ represent the means of L values of the filtering error and the echo signal. Convergence curves with a real non-stationary system and a slow movement of the person in the room [6] are displayed in Figs. 3 and 4 for each new version of the FNTF algorithm. Fig. 3 shows the better behavior of algorithm 3 over algorithm 1 and algorithm 2. It can also be seen that the filtering part modification (algorithm 1) is more effective than the temporal subdivision of the prediction part (algorithm 2). Fig. 4 compares the best algorithm of Fig.3 (algorithm 3) with algorithms 4 and 5. Even though each algorithm improves the tracking capability of the original FNTF version, it can be seen that the improvement of algorithm 5 is better than the others. Algorithm 5 uses both a split prediction and the final filtering error.

6. SUMMARY

In this work, 5 new versions of the Fast Newton Transversal Filter algorithm were presented. All the proposed algorithms improve the convergence speed for stationary cases, except the second algorithm. The simulation results have shown that all the proposed algorithms improve the tracking ability performance. The comparative study has also shown that the fifth algorithm is more effective than the others, because it gathered the advantages of two techniques, i.e. the filtering part modification and its temporal subdivision. It has also been observed that the performance properties of the new algorithms depend directly on the accelerated parameter value μ and on the subdivision number P. It should be noted that no numerical divergence problems were experienced in the simulations.

7. REFERENCES

[1] A. Benallal, « Etude des algorithmes MCR et application à l'identification de réponses impulsionnelles acoustiques », thèse de Doctorat, université de Rennes I, France, Jan 1989.

[2] G. Moustakides, S. Theodoridis, « Fast Newton Transversal Filters -A New class of adaptive estimation algorithms », IEEE transaction on signal processing, vol.ASSP-39, N°10, PP.2184-2193, October 1991.

[3] P. Petillon, A. Gilloire, S. Theodoridis , «The Fast Newton Transversal Filters:An Efficient scheme for acoustic Echo Conciliation in Mobile Radio », IEEE Transaction on signal processing, vol.42, N°3, March 1994.

[4] A. Benallal, A. Gilloire, « Improvement of the convergence speed and of the tracking capability of the numerical stable FLS algorithms for adaptive filtering » Proc. ICASSP 89, paper D5.3

[5] A. Benallal, A. Gilloire, « Instabilité et stabilité numérique des algorithmes de moindre carrés rapides excités par la parole» XII ième Colloque GRETSI, Juan Les Pin, pp. 509-512, juin 1989.

[6] Djendi M., Benallal A., Guessoum A., and Berkani D., " Three new versions for the Newton type adaptive filtering algorithm", Proc. IEEE ISSPA2003, vol.2, PP.559-562, France 2003.

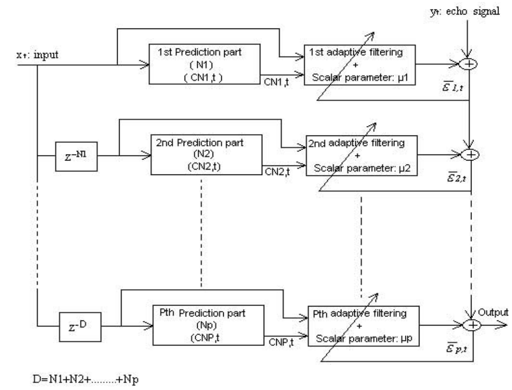

Figure 1. Temporal subdivision of the FNTF algorithm, first approach. For μ=0: algorithm 2, for μ=μ_i: algorithm 4.

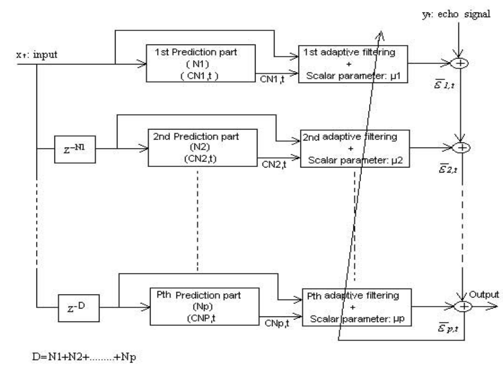

Figure 2. Temporal subdivision of the FNTF algorithm, second approach. For μ=0: algorithm 3 , for μ=μ_i: algorithm 5.

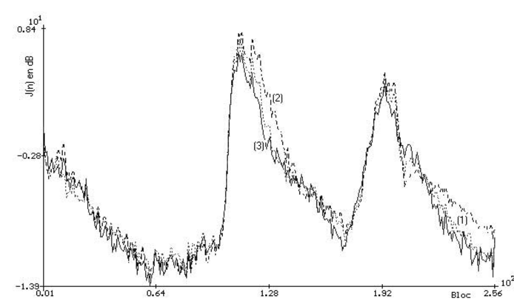

Figure 3. Convergence and tracking performance of algorithm 1, algorithm 2, and algorithm 3.

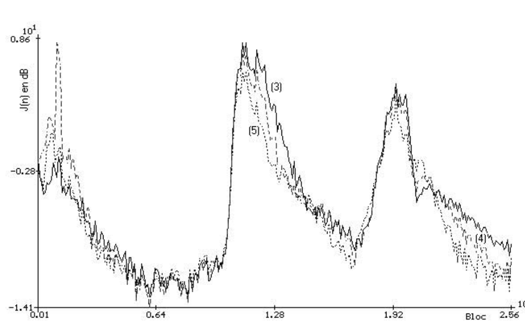

Figure 4. Convergence and tracking performance of algorithm 3, algorithm 4, and algorithm 5.