On the use of the FNTF algorithm in subband acoustic echo cancellation

De l'utilisation du filtre de Newton transversal rapide pour l'annulation des échos acoustiques dans les sous-bandes

François Caron*, Benoît Champagne†, *and* Qing-Guang Liu,‡ *Positron Public Safety Systems, 5101 Buchan Street, Montreal, Quebec H4P 2R9. †Department of Electrical and Computer Engineering, McGill University, Montreal, Quebec H3A 2A7. E-mail: champagne@ece.mcgill.ca. ‡Sony Electric Inc., Consumer AVD Engineering, 3300 Zanker Road, San Jose, California 95134, U.S.A.

The fast Newton transversal filter (FNTF) family of adaptive filtering algorithms bridges the performance gap between the standard normalized least-mean-square (NLMS) and fast recursive least-squares (FRLS) algorithms by allowing the use of linear predictors with variable orders. Recently, it has been shown that the FNTF is an effective scheme for fullband acoustic echo cancellation (AEC) of short echo paths (e.g., mobile context). In this paper, the merits of FNTF for subband AEC of the long echo paths typically associated with the use of hands-free audio terminals in offices (e.g., audioconferencing) are investigated. To this end, a stabilized version of FNTF is incorporated in an oversampled subband structure based on Weaver single-sideband (SSB) modulation. The performance of the resulting subband AEC system is evaluated in terms of convergence speed and computational complexity. The results point to some fundamental limitations of FNTF in this application.

Dans la famille des algorithmes de filtres adaptatifs, le filtre de Newton transversal rapide (FNTR) comble l'écart de performance entre les algorithmes de moindres carrés normalisés et les algorithmes de moindres carrés récursifs en permettant l'utilisation de prédicteurs linéaires d'ordre variable. Il a été récemment démontré que le FNTR est une approche intéressante pour l'annulation des échos acoustiques sur des trajets d'échos courts (comme, par exemple, dans un contexte de communications mobiles). Cet article présente le FNTR pour l'annulation des échos acoustiques dans les sous-bandes dans un contexte des parcours longs typiques de l'utilisation de terminaux audio mains-libres dans les édifices. Dans ce contexte, une version stabilisée du FNTR est ajoutée à une structure suréchantillonnée basée sur une modulation sideband de Weaver. Les performances du système obtenu sont mesurées selon des critères de vitesse de convergence et de complexité algorithmique. Les résultats montrent des limitations intrinsèques du FNTR dans ce type d'application.

I. Introduction

Hands-free audio terminals are being increasingly used as speech interfaces to the telephone network in various situations (e.g., audioconference, mobile-radio, etc.). As a result of the acoustic coupling that exists between the loudspeaker and the microphone, these terminals invariably introduce echoes in the return path. When combined with the long processing/propagation delays of modern digital networks, these acoustic echoes have a devastating impact on the perceived quality of the communications. The control of acoustic echoes generated by hands-free terminals represents a challenging problem for signal-processing engineers. Currently, acoustic echo cancellation (AEC) is seen as the most effective solution for commercial applications [1]–[2]. AEC basically consists of using an adaptive filter to identify the unknown acoustic echo path, and then subtracting the filter output from the microphone signal in an attempt to cancel the echo (see Fig. 1).

AEC is particularly demanding from the viewpoint of adaptive filtering since long, time-varying acoustic impulse responses must be identified; furthermore, the excitation signal (i.e., speech) is coloured and highly nonstationary. Even though recursive least-squares- (RLS) based adaptive filtering algorithms are known to achieve the fastest convergence for speech signal excitations, the normalized least-mean-square (NLMS) algorithm is generally used in practical AEC systems, mostly because of its low computational complexity of $2N$ multiply/adds per iteration (mapi)[1], where N is the number of adaptive filter coefficients. In addition, while the use of a single transversal filter may be adequate in the mobile context where the acoustic impulse responses

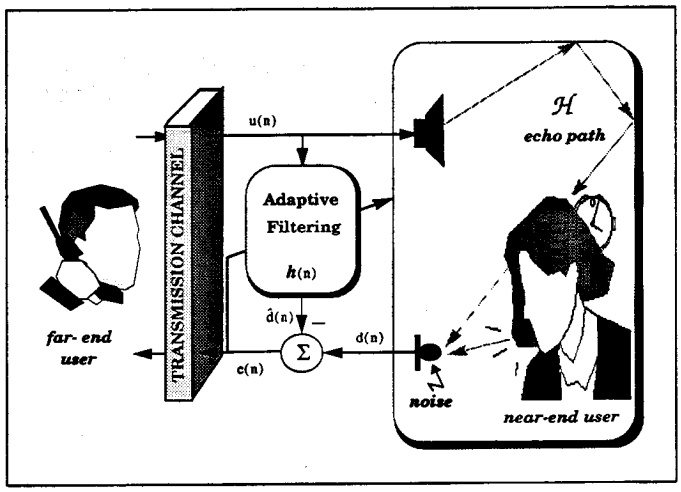

Figure 1: *Adaptive identification applied to AEC.*

are relatively short (e.g., 50 taps at 8-kHz sampling rate), low-cost identification of the long impulse responses typical of audioconference applications (e.g., 2000 taps at 8 kHz) necessitates the use of improved filtering structures with reduced computational complexity. In this respect, one of the most popular schemes is subband adaptive filtering, in which multiple adaptive filters (ADFs) operate in parallel on subband, decimated versions of the loudspeaker and microphone signals [3]–[4].

The fast Newton transversal filter (FNTF) family of adaptive filtering algorithms was proposed in an attempt to bridge the performance

[1] In this paper, unless otherwise indicated, complexity figures will be implicitly expressed in real mapi.

gap between the NLMS and the fast RLS (FRLS) algorithms [5]. Based on the assumption that the filter input signal is an autoregressive process of order M (i.e., AR(M)), where $0 \leq M \leq N$, the FNTF algorithms use extrapolation techniques to efficiently compute the N-th-order linear prediction parameters generally associated with FRLS algorithms. As a result, their computational complexity is only $2N + O(M)$, instead of the minimum of $7N$ for FRLS. Thus, by allowing the AR order M to vary between 0 and N, FNTF offers a trade-off between the low complexity of NLMS and the fast convergence of FRLS.

In a recent study [6], FNTF is shown to be an attractive candidate for AEC applications in the mobile context, where a single fullband transversal filter of short length may be used (i.e., $N = 256$). Indeed, it is observed that FNTF leads to significant improvements in convergence and tracking performance over NLMS even with small values of M, corresponding to a slight increase in computational complexity. However, this conclusion does not hold for longer filters (e.g., $N \geq 1024$), where a loss of performance of FNTF has been reported [7]. We point out that the investigation in [6] and [7] is limited to the use of a single fullband FNTF adaptive filter. In the case of long filters, FNTF might benefit from a subband implementation since the downsampling of the subband signals immediately translates into an equivalent reduction in the subband transversal filter length.

The main objective of this work is thus to investigate the performance of FNTF in a subband adaptive filtering structure for the identification of long impulse responses, such as those typically associated with the use of hands-free audio terminals in offices (e.g., audioconference applications). To conduct this study, a stabilized version of FNTF is incorporated in an oversampled subband identification structure with analysis/synthesis filter banks based on Weaver single-sideband (SSB) modulation. The performance of the resulting subband AEC system is evaluated in terms of convergence speed and computational complexity under varying conditions. The results point to some fundamental limitations of FNTF in this application.

II. On the use of FNTF in AEC

A. System-identification framework for AEC

The application of adaptive, discrete-time system identification to AEC is illustrated in Fig. 1. The unknown system $\mathcal{H}$ is made up of a cascade of the following subsystems: loudspeaker (including D/A converter and analogue amplifier), acoustic echo paths within the room, and microphone (including amplifier and A/D converter). Weak nonlinearities introduced by the electromechanical units are often neglected in AEC applications. However, changes in the echo paths resulting from the motion of persons and/or objects in the room may be quite significant. Accordingly, it is common practice to assume that $\mathcal{H}$ is a linear, time-varying, discrete-time system. The input to the unknown system $\mathcal{H}$ is the signal $u(n)$ from the far-end user, where $n \in \mathbb{Z}$ is the discrete-time index[2], and its output is the microphone signal $d(n)$, which contains additive noise and possibly speech from the near-end user.

For the purpose of system identification, the unknown system $\mathcal{H}$ is modelled as an adaptive finite-impulse-response (FIR) transversal filter operating on the input signal $u(n)$ in parallel to $\mathcal{H}$ and producing an output $\hat{d}(n)$. The time-varying coefficients of the modelling FIR filter are represented by $h_k(n)$ $(k = 0, 1, \ldots, N-1)$, where N is the filter length, and its output at time n is computed as

$$\hat{d}(n) = \sum_{k=0}^{N-1} h_k(n)u(n-k) = \mathbf{h}(n)^T\mathbf{u}(n), \tag{1}$$

[2]In this paper, discrete-time signals are assumed to be uniformly sampled, i.e., $u(n) = u_a(nT_s)$, where $u_a(\cdot)$ is the corresponding analogue signal and T_s is the sampling period.

where the following notation is assumed: $\mathbf{h}(n) = [h_0(n), ..., h_{N-1}(n)]^T$, $\mathbf{u}(n) = [u(n), u(n-1), ..., u(n-N+1)]^T$, and the superscript T denotes transposition. An adaptive weight-control algorithm recursively adjusts the filter weight vector in real time, based on the available information, so as to minimize the power of the residual signal, which is defined as

$$e(n) = d(n) - \hat{d}(n). \tag{2}$$

Different power measures and minimization techniques can be used, leading to different adaptation algorithms. Once the residual signal power has reached a minimum level, the unknown system may be identified from the coefficients $\{h_k(n)\}$. In AEC applications, one is not directly interested in these coefficients; the aim is rather to transmit the residual signal $e(n)$, following echo removal in (2), to the far-end user. When the microphone signal $d(n)$ contains speech from the near-end user, a situation referred to as double-talk, correct operation of the adaptive filter requires that weight adjustment be interrupted; if not, the adaptive filter will try to cancel the near-end signal. To avoid this problem, practical AEC systems use double-talk detectors to control the adaptation process. We shall not be concerned with these devices in our study.

B. The FNTF family of adaptive filters

The FNTF algorithms belong to a modified class of stochastic Newton (SN) adaptive algorithms, whose general form may be expressed as follows:

$$\mathbf{c}_N(n) = -\frac{1}{\lambda}R_N^{-1}(n-1)\mathbf{u}(n), \tag{3}$$

$$\gamma(n) = 1 - \mathbf{c}_N^T(n)\mathbf{u}(n), \tag{4}$$

$$e(n) = d(n) - \mathbf{h}^T(n)\mathbf{u}(n), \tag{5}$$

$$\epsilon(n) = \frac{e(n)}{\gamma(n)}, \tag{6}$$

$$\mathbf{h}(n+1) = \mathbf{h}(n) - \epsilon(n)\mathbf{c}_N(n). \tag{7}$$

In these equations, $R_N(n)$ is an $N \times N$ data covariance matrix estimate required to be symmetric and positive definite; λ is a forgetting factor, such that $0 < \lambda \leq 1$; $\mathbf{c}_N(n)$ is a generalized dual Kalman gain; $e(n)$ (same as (2)) is the *a priori* estimation error; $\epsilon(n)$ is the *a posteriori* estimation error; and $\gamma(n)$ is a conversion factor between these two forms of error.

Equations (5)–(7) are known as the filtering part; the associated complexity of $2N$ defines a lower bound for the SN class. The two major factors that distinguish the various SN algorithms in terms of convergence speed and computational complexity are (a) the specific choice made for $R_N(n)$ in (3), and (b) the algorithmic scheme used to realize (3) and (4) once this choice has been made. For instance, selecting $R_N(n)$ equal to $(\sigma_u^2/\delta)I_N$, where σ_u^2 is an estimate of the input power, $\delta > 0$ is a small constant (step size) and I_N is the identity matrix, results in a normalized form of the LMS algorithm (NLMS). The latter exhibits a low complexity of $2N$, but its convergence is rather slow for speech-like inputs. At the other end of the spectrum, selecting $R_N(n)$ equal to $\sum_{k=0}^{n}\lambda^{n-k}\mathbf{u}(k)\mathbf{u}(k)^T$ leads to the RLS algorithm, which is characterized by rapid initial convergence but high complexity; i.e., $O(N^2)$. In this case, it is actually possible to realize (3) and (4) more efficiently by using forward and backward LS linear predictors of order N, leading to the so-called fast RLS (FRLS) algorithms, including FAEST [8] and FTF [9], both with complexity of $7N$.

One may observe that the choice of $R_N(n)$ leading to the NLMS algorithm is equivalent to the assumption that the input signal $u(n)$ is white noise, i.e., AR(0). Similarly, the choice of $R_N(n)$ leading to the FRLS algorithms may be interpreted as an AR(N) assumption for the input, since these algorithms use linear predictors of order N in their formulation. In the derivation of the FNTF family of algorithms [5], this idea is generalized by assuming that the input signal can be modelled as an AR(M) process, where $0 \leq M \leq N$. This

is justified on the following grounds: (a) The use of AR(0) modelling is inadequate for nonwhite signals, explaining the slow convergence of the NLMS algorithm in this case. (b) For long adaptive filters (i.e., $N \gg 1$), the use of AR(N) predictors is excessive, leading to unnecessary computations in FRLS algorithms. Thus, it is conjectured that by allowing the prediction order M to vary between 0 and N, and by exploiting the AR(M) assumption in (3) and (4), it may be possible to achieve a useful trade-off between computational complexity and convergence speed, effectively bridging the performance gap between NLMS and FRLS.

In [5], the AR(M) assumption is exploited indirectly by regarding the definition of $R_N(n)$ as an extension problem; namely, how to generate the complete $N \times N$ matrix $R_N(n)$, given the entries of the diagonal and all entries of the M off-diagonals. By considering a game theoretic formulation of this problem, an optimal solution is derived in the form of simple time-and-order recursive formulas for the extension of an inverse covariance matrix of order $M+1$, say $R_{M+1}^{-1}(n)$, into the desired $N \times N$ inverse covariance matrix, $R_N^{-1}(n)$. It turns out that the extended matrix $R_N^{-1}(n)$ has all off-diagonals of order greater than M equal to 0, a trait which is characteristic of AR(M) processes. The extension formulas are characterized by the use of quantities pertaining to optimum forward and/or backward linear predictors of order M for the associated matrix $R_{M+1}(n)$. Assuming that the known entries of $R_{M+1}(n)$ are defined as standard exponentially weighted time averages, these predictors are LS optimum and can therefore be computed using FRLS algorithms of order M.

The FNTF algorithms are finally obtained upon substitution of the optimal extension formulas for $R_N^{-1}(n)$ into (3). In effect, this leads to various Levinson's-like recursions with complexity $2M$ or $3M$ for updating the dual-Kalman-gain vector $\mathbf{c}_N(n)$. Thus, using FRLS to realize the required linear predictors of order M, we find that the total computational complexity of the FNTF family is $2N + O(M)$.

C. Using FNTF in AEC: Special considerations

In the context of AEC, one is interested in low-cost, real-time implementation of the adaptive filtering algorithms on a DSP platform, so that the practical issues of memory and computational requirements are of paramount importance. Following [6], we therefore chose FNTF Version 1 in our work because of its minimum requirements in terms of both memory and processing [5]. This version (simply referred to as FNTF in the following) is summarized below:

(a) Using an FRLS forward predictor of order M with input $u(n)$, compute

$$\mathbf{s}_{M+1}(n) = \frac{e_M^f(n)}{\lambda \alpha_M^f(n-1)} \begin{bmatrix} 1 \\ -\mathbf{a}_M(n-1) \end{bmatrix}. \tag{8}$$

(b) Using an FRLS backward predictor of order M with delayed input $u(n_d)$, where $n_d = n - N + M$, compute

$$\mathbf{t}_{M+1}(n_d) = \frac{e_M^b(n_d)}{\lambda \alpha_M^b(n_d-1)} \begin{bmatrix} -\mathbf{b}_M(n_d-1) \\ 1 \end{bmatrix}. \tag{9}$$

(c) Update the dual-Kalman-gain vector $\mathbf{c}_N(n)$ and the conversion factor $\gamma(n)$:

$$\begin{bmatrix} \mathbf{c}_N(n) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{c}_N(n-1) \end{bmatrix} - \begin{bmatrix} \mathbf{s}_{M+1}(n) \\ \mathbf{0}_{N-M} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{N-M} \\ \mathbf{t}_{M+1}(n_d) \end{bmatrix}, \tag{10}$$

$$\gamma(n) = \gamma(n-1) + s_{M+1}^1(n) e_M^f(n) - t_{M+1}^{M+1}(n_d) e_M^b(n_d). \tag{11}$$

(d) Filtering part: Same as (5)–(7) above.

Steps (a) and (b) of the algorithm require the use of FRLS linear prediction error filters of order M [10]. More specifically, in step (a), a forward prediction error filter is applied to the "head" (i.e., top $M+1$ entries) of the data vector $\mathbf{u}(n)$ to generate the quantities needed in (8); namely, the M-dimensional weight vector $\mathbf{a}_M(n-1)$, the residual (*a priori*) error $e_M^f(n)$, and the error power $\alpha_M^f(n-1)$. Similarly, in step (b), a backward prediction error filter is applied to the "tail" (i.e., bottom $M+1$ entries) of the data vector $\mathbf{u}(n)$ to generate the quantities needed in (9); namely, the M-dimensional weight vector $\mathbf{b}_M(n_d-1)$, the residual (*a priori*) error $e_M^b(n_d)$, and the error power $\alpha_M^b(n_d-1)$. The vectors $\mathbf{s}_{M+1}(n)$ and $\mathbf{t}_{M+1}(n_d)$ computed in this way are used in step (c) to update the dual-Kalman-gain vector $\mathbf{c}_N(n)$ and the conversion factor $\gamma(n)$. To initialize the algorithm, [5] recommends use of the soft constraint method as in [9].

FAEST [8] and FTF [9] are particularly attractive for the realization of steps (a) and (b) since they explicitly compute the vector quantities $\mathbf{s}_{M+1}(n)$ (8) and $\mathbf{t}_{M+1}(n)$ (9). However, these algorithms are known to be numerically unstable: the accumulation of round-off errors in their digital implementation leads to divergence of the internal variables. In real-time AEC applications, this behaviour is unacceptable since the adaptive filter must operate correctly over long periods of time. To overcome this problem, [6] recommends the use of a numerical stabilization mechanism proposed in [11]. If one assumes that this mechanism is used in connection with FAEST or FTF in step (a) of the FNTF, the computation of $\mathbf{s}_{M+1}(n)$ requires $6M$ multiplies and three divisions. The same stabilized FAEST or FTF may be used to compute $\mathbf{t}_{M+1}(n)$, which is then delayed by $N-M$ samples as prescribed in step (b). However, to minimize memory requirements in DSP implementations, it is preferable to use a second stabilized FAEST or FTF running in parallel on the tail of the data vector $\mathbf{u}(n)$; this also requires $6M$ multiplies and three divisions. In step (c), computation of the updated gain vector $\mathbf{c}_N(n)$ in (10) requires $2M$ additions, which we count as mapi.[3] Finally, step (d) (see (5)–(7)) requires $2N$ mapi plus one division. Thus, the total complexity of FNTF Version 1 is $2N + 14M + 7k_d$, where k_d is the equivalent complexity of a division expressed in mapi.

As pointed out in [6], even when the stabilization mechanism of [11] is incorporated in FTF, the resulting FNTF algorithm may be unstable unless large values of the forgetting factor are used; i.e., $\lambda > 1 - 1/pN$, where $p \geq 1$. Unfortunately, such large values of λ severely limit the capability of FNTF to track time-varying channels. In AEC applications, where the unknown echo path is changing continually, this is clearly undesirable. In [6], the tracking performance of FNTF is improved via the introduction of an acceleration mechanism in the filtering part, i.e., step (d). Originally proposed in [12], this mechanism amounts to replacing (6) by

$$\epsilon(n) = \frac{e(n)}{\gamma(n) - \rho}, \tag{12}$$

where ρ is a control parameter less than 1. By properly adjusting ρ, the tracking speed of FNTF may be improved without changing the value of λ in the prediction parts (i.e., steps (a) and (b)).

In [6], the use of FNTF (with the above-mentioned stabilization and acceleration mechanisms) is investigated in the context of fullband AEC for mobile-radio applications. The latter are characterized by moderate filter lengths, typically $N = 256$ at 8-kHz sampling rate, corresponding to an acoustic echo path of duration 32 ms. In particular, it is demonstrated experimentally that FNTF yields close-to-FRLS performance with M equal to only 16, corresponding to a complexity about 1.4 times that of NLMS (as compared to 4 times for a stabilized FTF). However, it is noted in [7] that FNTF shows a loss of performance in the audioconference context where large values of N are needed; e.g., $N \geq 1000$ for a fullband system. We provide results supporting this observation in Section IV.

[3] We shall neglect the 2 mapi required for (11) in our count.

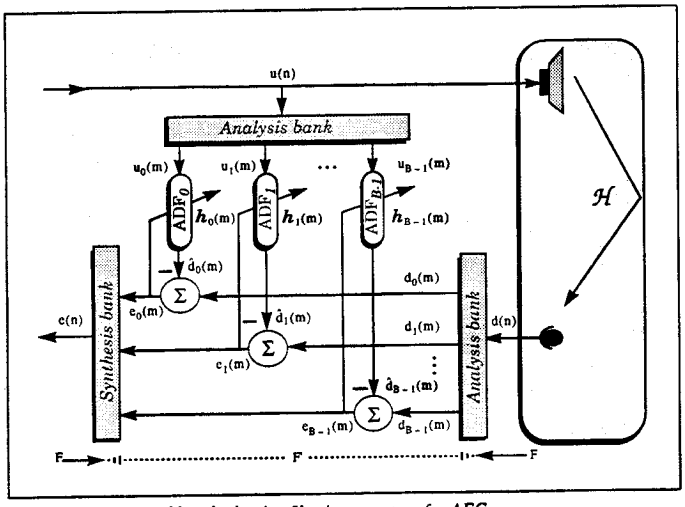

Figure 2: *Generic subband adaptive filtering structure for AEC.*

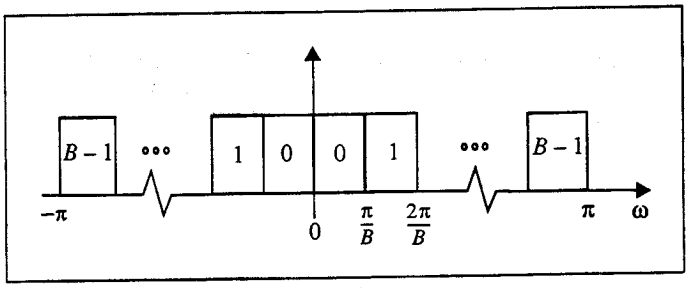

Figure 3: *Subband configuration for Weaver SSB structure.*

III. Subband FNTF structure based on Weaver SSB modulation

In this section, we describe the subband adaptive filtering structure used in our investigation. It consists of multiple FNTFs operating in subbands at a reduced sampling rate, with analysis/synthesis (A/S) filter banks derived from Weaver SSB modulators/demodulators.

A. Subband AEC: Structure and motivation

Subband adaptive filtering [3]–[4] is one of the most popular approaches for the efficient implementation of the long adaptive filters needed for AEC in audioconference applications. It has been studied extensively in recent years (e.g., [1]–[2]) and is now used in commercial products [13].

In subband AEC (see Fig. 2), both the loudspeaker and microphone signals, respectively $u(n)$ and $d(n)$ with sampling rate F_s, are decomposed into B subband signals by so-called analysis filter banks. Each of these banks may be viewed as a set of B bandpass filters followed by decimators: the filters split the spectrum of interest into B adjacent, non-overlapping frequency bands, while the decimators reduce the sampling rate of the filtered signals by an integer factor $K \leq B$. The subband loudspeaker and microphone signals are denoted here by $u_b(m)$ and $d_b(m)$, where $b = 0, \ldots, B-1$, and the integer m is the sampling-time index at the lower rate $F_s' = F_s/K$.

In each subband, an adaptive filter operating at the rate F_s' is used to identify the corresponding subband component of the acoustic echo path $\mathcal{H}$. Specifically, the ADF in the b-th subband uses $u_b(m)$ as input and $d_b(m)$ as reference, and produces an output $\hat{d}(m)$; the latter is subtracted from $d_b(m)$ to produce a subband error $e_b(m) = d_b(m) - \hat{d}_b(m)$. Each ADF operates independently of the others, trying to minimize its own residual $e_b(m)$, so that after convergence $\hat{d}_b(m)$ provides an estimate of the acoustic echo in that subband. Following the ADF, the subband error signals $e_b(m)$ $(b = 0, \ldots, B-1)$ are recombined by a synthesis bank to produce a fullband error signal $e(n)$, at the original rate F_s. In the synthesis bank, the subband signals $e_b(m)$ are first upsampled by a factor K, then passed through narrowband anti-imaging filters, and then summed.

In the context of AEC, subband adaptive filtering offers several advantages over a conventional fullband approach [3]: (a) As a result of signal downsampling in the subbands, a reduction of the total system complexity by a factor of about K^2/B may be achieved. (b) For highly correlated input signals, faster convergence of stochastic gradient algorithms such as NLMS is usually observed due to the decorrelation effect of the downsampling. (c) Processing in individual subbands may be tailored to the specific needs of an application (e.g., longer ADFs in certain subbands), adding considerable flexibility to system design and implementation.

B. Bandpass A/S filtering via Weaver SSB modulation

The design of A/S filter banks for subband AEC applications is a complex problem involving several trade-offs. Specific requirements include the following (see [14]–[15] for additional details):

Near-perfect reconstruction (NPR): This is required to avoid audible distortion of the near-end user's signal, which goes through a cascade of the A/S banks prior to its transmission (see Fig. 2).

Low processing delay: In audioconference applications over digital networks, the extra processing delay incurred by the near-end user's speech through the A/S banks must be kept low.[4]

Flexibility in oversampling: With critical downsampling, i.e., $K = B$, large, audible peaks occur at the subband boundaries in the power spectrum of $e(n)$, as a result of subband aliasing with non-ideal bandpass filters [3]. To avoid this problem, the most efficient approach remains oversampling, i.e., $K < B$. The A/S bank design/structure should allow arbitrary integer choices of K, so that this parameter may be maximized subject to the constraint of no audible aliasing.

Low computational complexity: The extra computational load resulting from A/S processing should be small. For ease of implementation, it may also be desirable that the subband signals be real. These considerations generally lead to the use of oversampled uniform filter banks in subband AEC, with A/S filters derived from modulation of a low-pass prototype (e.g., [16]–[19]). The selection of the filter-bank parameters (number of subbands, downsampling factor and prototype filter) remains a difficult problem and is most often based on analytical approximations and/or empirical guidelines, rather than on optimal design approaches (e.g. [14]–[15], [20]). Various techniques may be used for the realization/implementation of these filter banks, as described in [21] and [22].

In our study of subband FNTF, we have found it convenient to use a Weaver SSB subband structure similar to that in [13]. In addition to satisfying the above requirements for AEC applications, this structure produces purely real subband signals, a result which greatly simplifies the implementation and use of adaptive filtering algorithms in subbands. In the Weaver SSB structure, the subband configuration is derived from cosine modulation of a low-pass prototype. However, to avoid decimation aliasing between negative and positive frequency bands, which is present with cosine-modulated banks even in the oversampling scheme, A/S filtering is achieved via a special form of SSB modulation, known as Weaver modulation. Efficient realizations exist for Weaver SSB subband structures based on polyphase decomposition and weighted overlap-add methods [21], although this is not a critical issue here.

In the Weaver SSB subband structure under consideration, the digital spectrum $-\pi \leq \omega \leq \pi$, where ω denotes the normalized angular frequency, is divided into B real subbands with bandwidth $\omega_\delta = \pi/B$, as shown in Fig. 3. The centre frequency of the b-th subband (positive sideband) is given by $\omega_b = (b + 1/2)\omega_\delta$, with $b = 0, 1, \ldots, B-1$.

[4] In videoconference applications, processing delay is dictated mainly by the image-compression algorithms, so that delay through the A/S banks is not as critical.

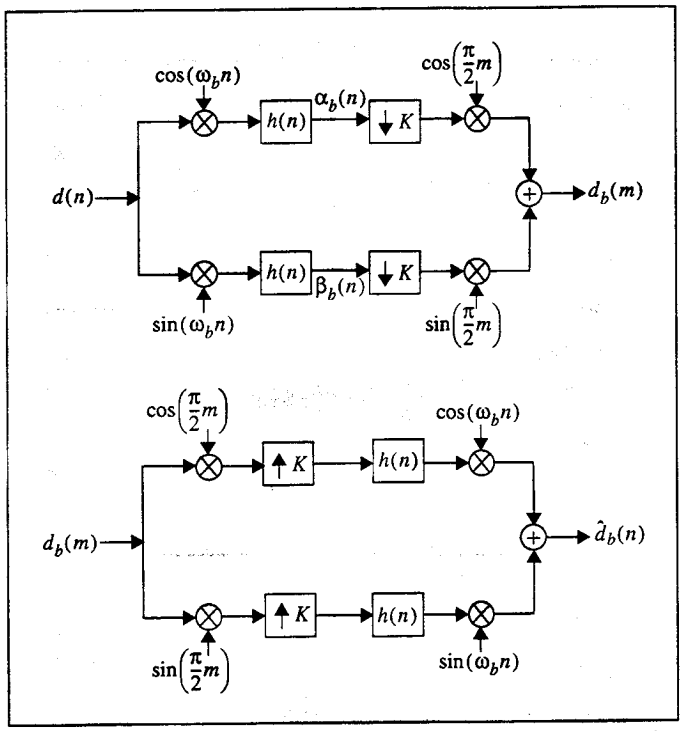

Figure 4: *Schematic representation of Weaver SSB modulator (top) and corresponding demodulator (bottom).*

Each narrowband filter in the analysis bank is in effect a Weaver modulator with centre frequency ω_b; a corresponding Weaver demodulator is used in the synthesis bank. Block diagrams of the Weaver modulator and demodulator subsystems for the b-th subband are shown in Fig. 4. In the modulator, the quadrature components of the input $d(n)$ are first computed via real modulation and low-pass filtering; i.e.,

$$\begin{aligned} \alpha_b(n) &= [d(n)\cos(\omega_b n)] * h(n), \\ \beta_b(n) &= [d(n)\sin(\omega_b n)] * h(n), \end{aligned} \tag{13}$$

where $*$ denotes discrete-time convolution and $h(n)$ is the impulse response of an ideal low-pass filter with cut-off at $\omega_\delta/2$. The quadrature components $\alpha_b(n)$ and $\beta_b(n)$ are then downsampled by a factor $K \leq B$, modulated by $\pi/2$, and summed to produce the subband output $d_b(m)$:

$$d_b(m) = \alpha_b(Km)\cos\left(\frac{\pi m}{2}\right) + \beta_b(Km)\sin\left(\frac{\pi m}{2}\right). \tag{14}$$

The Weaver demodulator simply performs the reverse (i.e., dual) operations, restoring the subband spectrum to its original shape and position via a combination of quadrature modulation and upsampling. In the synthesis bank, a reconstructed version of the input $d(n)$ is finally obtained via summation of all the demodulator outputs, i.e., $\hat{d}_b(n)$ for $b = 0, 1, \ldots, B-1$.

The net effect of the Weaver modulation scheme in the frequency domain is illustrated in Fig. 5, where $D(\omega)$ and $D_b(\omega)$ denote the Fourier transforms[5] of $d(n)$ and $d_b(m)$, respectively. Equations (13)–(14) in effect correspond to an SSB modulation of $d(n)$, shifting the positive sideband from ω_b to $\pi/(2K)$, followed by decimation by K (see [21], Section 2.4, for details). In practice, a non-ideal low-pass filter $h(n)$ must be used, instead of the ideal one. As can be seen from Fig. 5, oversampling (i.e., $K < B$) in Weaver modulation makes it possible to avoid aliasing between positive and negative sidebands by introducing guard bands.

[5]The Fourier transform of a discrete-time signal $d(n)$, $n \in \mathbb{Z}$, is defined as $D(\omega) = \sum_{n=-\infty}^{\infty} d(n)e^{-j\omega n}$.

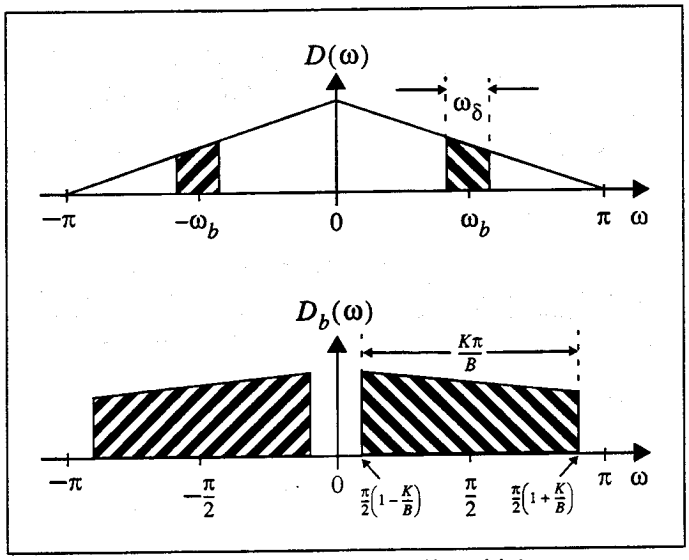

Figure 5: *Frequency-domain interpretation of Weaver SSB modulation.*

In our application, where the sampling rate is $F_s = 8$ kHz, the frequency spectrum from 0 to $F_s/2 = 4$ kHz is divided into $B = 16$ real subbands with bandwidth 250 Hz, or equivalently $\omega_\delta = 0.0625\pi$ in the normalized angular frequency domain. The downsampling factor is set to $K = B/2 = 8$: this practically eliminates subband aliasing and also simplifies the design of the prototype low-pass filter $h(n)$, which is carried out with the window method. Specifically, a Hamming window of length $L = 128$ is applied to an ideal low-pass impulse response with cut-off $\omega_c = 0.0377\pi$. Note that to optimize the resulting A/S bank, i.e., to further reduce amplitude distortion, ω_c is taken to be slightly larger than the required value of $\omega_\delta/2 = \pi/(2B) = 0.03125\pi$. The resulting Weaver SSB subband structure (i.e., cascade of A/S banks) has the following properties: amplitude distortion flat within about ± 0.025 dB; linear phase response (exactly); processing delay of about 16 ms (i.e., $(2L-1)/2$ samples). Informal listening tests of a speech signal going through a cascade of the A/S banks do not reveal any perceptible distortion.

C. Using FNTF in subbands

We finally discuss specific issues related to the use of FNTF in a subband structure based on Weaver SSB modulation.

In our implementation of this scheme, we use B copies of FNTF Version 1, as described in Section II.C, including the acceleration mechanism (12). These FNTFs run independently of each other on their respective subband signals. To simplify the study, the same parameter values (e.g., N and M) are used in all the subbands.[6] In each subband, two independent stabilized FTF algorithms, as described in [6], are used to compute the $\mathbf{s}_{M+1}(n)$ (8) and $\mathbf{t}_{M+1}(n_d)$ (9) values needed in FNTF. The value of the forgetting factor λ is set to $1-1/pN$, where $p \geq 1$ (typically between 2 and 5); such a large value of λ is necessary to extend the stable life of the algorithm [6]. Even in its stabilized form, practical operation of FTF with speech input over long periods of time generally requires either conditional or periodic reinitialization to avoid locking or numerical error buildup. However, with the above choice of λ, the use of reinitialization was not necessary over the time span of our experiments.

For the above settings, the total computational complexity of the B subband FNTF algorithms, expressed in mapi at the original sampling

[6]The use of different parameter values in different subbands may indeed result in improved AEC performance. However, selecting the subband parameters independently (and possibly adaptively) so as to optimize the overall system performance under a constraint of fixed complexity remains a challenging problem.

rate F_s, is given by

$$\frac{B}{K}[2N_s + 14M_s + 7k_d], \tag{15}$$

where N_s denotes the transversal filter length (i.e., number of adaptive coefficients) and M_s denotes the prediction order of the subband FNTFs. Note that the value of N_s required to match a fullband filter of length N is $N_s = N/K = 2N/B$. Formally, a similar relation may be given for the prediction order, i.e., $M_s = M/K = 2N/B$. However, as we will see in Section IV, the effects of varying the prediction order on the convergence behaviour are different for the fullband and subband FNTFs. The value of k_d (i.e., complexity of a division in mapi) in (15) is implementation-dependent, but typical values for commercial DSPs are on the order of 10 or more.

To the above figure, we must add the operation count of the subband processing, i.e., two analysis banks and one synthesis bank. Assuming that the proposed subband structure, which is based on Weaver SSB modulation, is implemented via the WOA approach [21], its computational complexity is about

$$\frac{3}{K}[L + B\log_2 B + B]. \tag{16}$$

This represents a lower bound for the proposed subband FNTF scheme, regardless of the value of the subband filter length N_s. In the following sections, we refer to this scheme as the Weaver SSB subband FNTF algorithm.

IV. Computer experiments and results

The convergence performance of the Weaver SSB subband FNTF algorithm was investigated in the context of audioconference applications (i.e., long echo paths) via computer experiments. Below, we briefly describe the experimental methodology and then proceed with the presentation and discussion of the results.

A. Methodology

As our interest lies in speech communications over the telephone bandwidth, the sampling rate was set to $F_s = 8$ kHz. Different source signals $u(n)$ were tried in the experiments, including noise-like signals, a composite source signal (CSS) and recorded speech. Here, we show results for CSS: a speech-like signal originally proposed for determining the transfer characteristics of hands-free phones [23], but also used as a test signal by ITU Study Group 12 [24] for evaluating the convergence behaviour of AEC systems. The CSS used in this study consists of a succession of identical bursts with reverse polarity, each burst being made up of three sections; namely, voice sound, pseudonoise and pause (see Fig. 6).

To simulate the acoustic echo path $\mathcal{H}$ between the loudspeaker and the microphone in Fig. 1, a fixed impulse response of duration 3200 samples was generated with the image method [25]. The geometric and acoustic parameters of the room were adjusted to match certain specifications given in [26] for audioconference applications: room volume of 90 m^3 and reverberation time of about 400 ms. The impulse response was truncated to a desired length N (same as the fullband adaptive filter length) and convolved with $u(n)$ to produce the echo signal; independent Gaussian white noise was added to the echo to obtain the microphone signal. In our experiments, the noise level was usually set to -30 dB, a level which is representative of audioconference applications.

The signals $u(n)$ and $d(n)$ were used as input and reference signals, respectively, to various FNTF-based AEC algorithms to study their convergence performance. The latter was evaluated in terms of the short-term power of the (fullband) residual echo at the output of the AEC system:

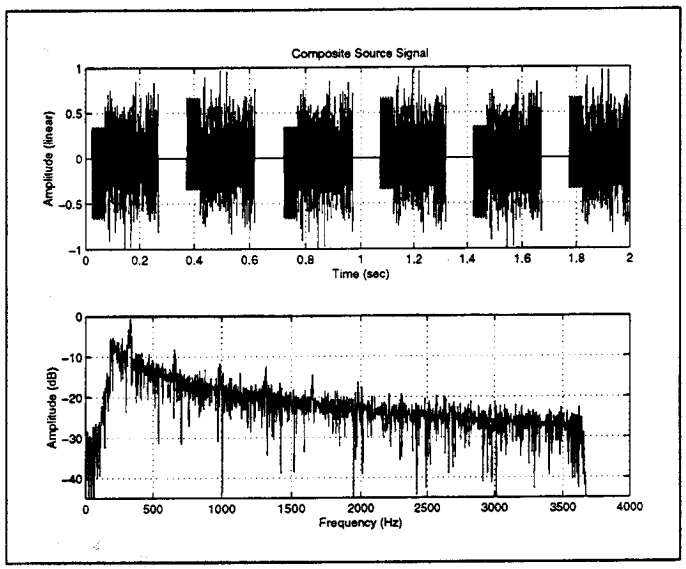

Figure 6: *The CSS signal: Time plot (top) and amplitude spectrum (bottom).*

$$J(n) = \frac{1}{K}\sum_{k=n-K+1}^{n} e(k)^2. \tag{17}$$

Here, $J(n)$ is expressed in decibels, with reference to the echo level at the microphone, and the window length is set to $K = 256$ (32 ms).

Both the fullband FNTF algorithm (as in [6]) and the Weaver SSB subband FNTF algorithm (as described in Section III) were considered. As indicated earlier, the soft constraint approach was used for FNTF initialization. Following a common practice [5]–[6], the filtering part of FNTF (either fullband or subband) was frozen during the first 0.5 s (i.e., from time -0.5 s to 0 s) to eliminate the influence of FTF initialization on the convergence results. During this period, the FIR filter coefficients are null and $J(n)$ yields the power of the microphone signal, i.e., echo plus additive noise. Different values of the prediction order M (or M_s for the subband FNTF) were tried in the experiments. In this respect, recall that $M = N$ corresponds to FTF, or more generally FRLS, while $M = 0$ corresponds to a form of NLMS.

B. Results

In [6], it is demonstrated that in a mobile-radio context with $N = 256$, the use of fullband FNTF with M as small as 16 may yield greatly enhanced (i.e., close to FRLS) performance as compared to NLMS, with very few extra computations (about 40% more). In [7], the same authors note that they have observed a loss of performance of this algorithm in the audioconference context with larger values of N, but no specific results are given. We first present a series of experimental results for the fullband FNTF that corroborate and clarify this statement.

Convergence curves (i.e., $J(n)$ versus time) for the fullband FNTF algorithm are shown in Fig. 7 for a filter length $N = 256$ and various prediction orders $M = 0, 16, 256$. The short time power of the additive noise is also included in the figure (bottom curve at -30 dB) as it represents a lower bound for any adaptive algorithm. It can be seen that for the CSS signal, significant convergence improvements may be obtained with $M = 16$. Fig. 8 shows convergence curves for the fullband FNTF algorithm in the case of a filter length $N = 2048$ and for prediction orders $M = 0, 16, 1024, 2048$. Here, while the choice $M = 16$ does result in faster convergence than $M = 0$ (i.e., NLMS), it is necessary to use much larger prediction orders, say $M = 512$ or more, to get performance comparable to the case $M = 2048$ (i.e., FRLS). For such large values of M, FNTF is no longer a computationally attractive solution.

We next investigate the behaviour of the subband FNTF algorithm for the case $N_s = 256$, corresponding to a fullband filter length of

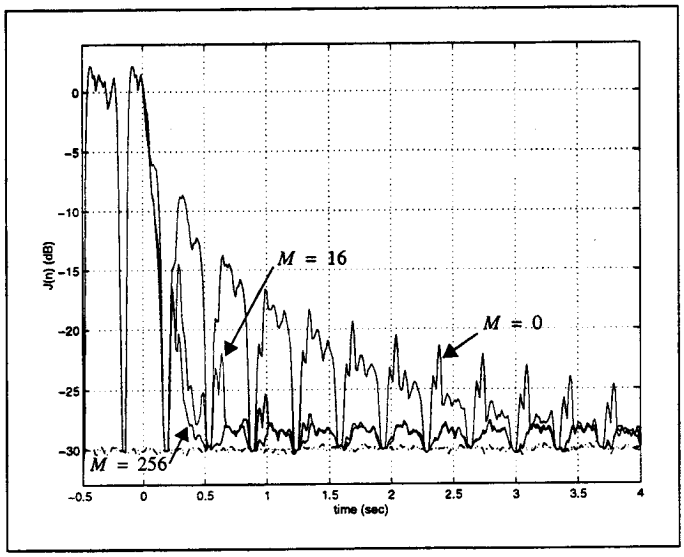

Figure 7: *Convergence curves for fullband FNTF ($N = 256$, $p = 5$ and $\rho = 0.75$). From top to bottom: FNTF with $M = 0, 16, 256$, and noise floor.*

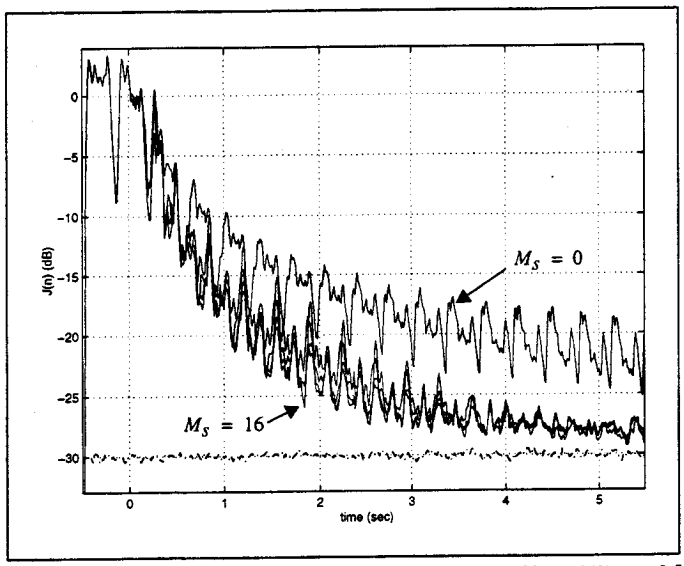

Figure 9: *Convergence curves for subband FNTF ($N_s = 256$ (i.e., $N = 2048$), $p = 2.5$ and $\rho = 0.80$). From top to bottom: subband FNTF with $M_s = 0, 2, 4, 8, 16$, and noise floor.*

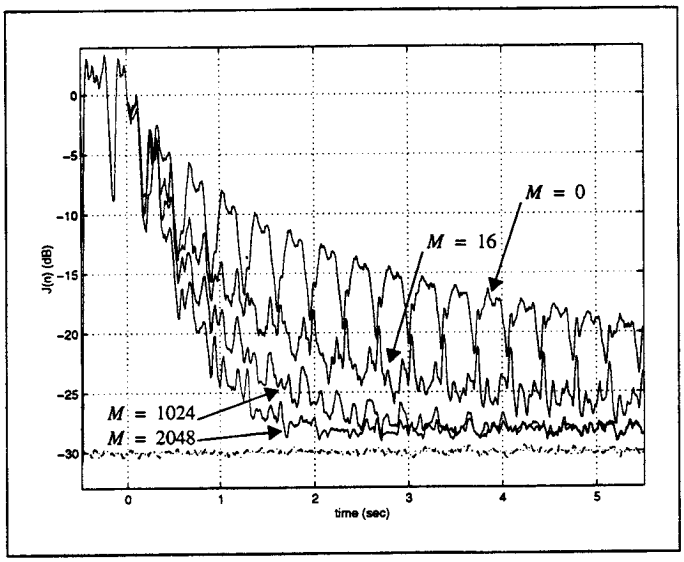

Figure 8: *Convergence curves for fullband FNTF ($N = 2048$, $p = 2.5$ and $\rho = 0.80$). From top to bottom: FNTF with $M = 0, 16, 1024, 2048$, and noise floor.*

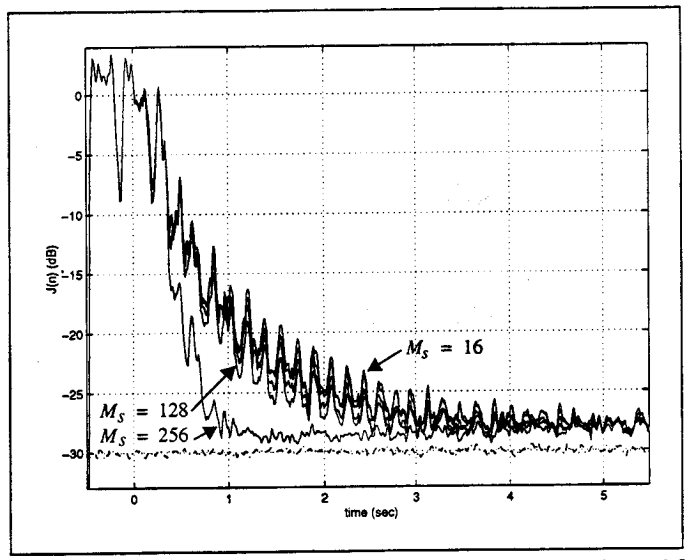

Figure 10: *Convergence curves for subband FNTF ($N_s = 256$ (i.e., $N = 2048$), $p = 2.5$ and $\rho = 0.80$). From top to bottom: subband FNTF with $M_s = 16, 32, 64, 128, 256$, and noise floor.*

$N = 2048$ as in Fig. 8. Convergence curves for the Weaver SSB subband FNTF scheme are shown in Fig. 9 for subband prediction orders $M_s = 0, 2, 4, 8, 16$, and in Fig. 10 for $M_s = 16, 32, 64, 128, 256$. Comparing the curves $M_s = 0$ in Fig. 9 and $M = 0$ in Fig. 8, both of which correspond to NLMS, we first note that the subband algorithm converges faster than its fullband counterpart. A similar observation can be made for the curves $M_s = 256$ in Fig. 10 and $M = 2048$ in Fig. 8, which correspond to the FRLS. These observations are consistent with commonly established knowledge that subband processing increases the convergence speed of NLMS and, to some extent, FRLS algorithms.

Next, and more importantly, we note that the effects of increasing the FNTF's prediction order on the convergence speed are quite different in the subband case than in the fullband case. While a significant gain in convergence speed is achieved by increasing M_s from 0 to 2 in Fig. 9, further increases in M_s in this figure do not yield significant additional improvements. Indeed, not much difference can be seen among the curves for $M_s = 16$ to $M_s = 128$ in Fig. 10. Thus, it appears that the ratio of M_s/N_s required in the subband case to achieve a convergence speed comparable to FRLS is even larger than the corresponding ratio of M/N needed in the fullband case.

To illustrate this last point, we look at the results from a different perspective. Define the convergence time of an AEC algorithm as the time interval it requires to achieve an echo cancellation of 20 dB. Define the normalized convergence time (NCT) of a fullband (subband) algorithm as the ratio of its convergence time to that of the fullband (subband, respectively) stabilized FRLS having the same length. In a similar way, define the normalized computational complexity (NCC) of a fullband (subband) algorithm as the ratio of its complexity in mapi, to that of the fullband (subband) NLMS of the same length. Fig. 11 shows plots of NCT versus NCC for the following cases: (a) fullband FNTF for $N = 256$ (fb256); (b) fullband FNTF for $N = 2048$ (fb2048); (c) subband FNTF for $N_s = 256$ (sb256). Each plot is obtained by varying the prediction order M (M_s for subband algorithms) over the range of permissible values, i.e., from 0 to 256 in cases (a) and (c) and from 0 to 2048 in case (b). In particular, $M = 0$ (or $M_s = 0$ in subband) corresponds to NLMS and thus NCC = 1, while $M = N$ (or $M_s = N_s$ in subband) corresponds to FRLS and thus NCT = 1.

The problems identified previously are clearly visible from Fig. 11. In the fb256 case, an NCC of 2 (obtained with $M = 32$) yields a convergence performance almost identical to that of FRLS, whose NCC is

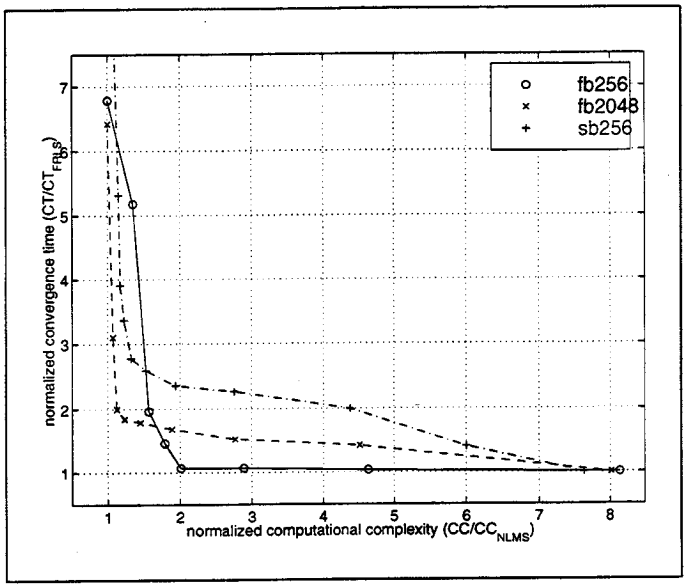

Figure 11: *Normalized convergence time versus normalized computational complexity for fullband FNTF with $N = 256$ and $N = 2048$, and for subband FNTF with $N_s = 256$.*

actually about 4.[7] In the fb2048 case, while an NCC of 2 (obtained with $M \approx 256$) yields a significant improvement in convergence speed, the corresponding NCT of 1.7 leaves significant room for improvement. To significantly reduce this value, we need to further increase the prediction order M, but then FNTF is no longer attractive since its NCC will exceed that of FRLS (i.e., NCC ≥ 4). Finally, in the sb256 case, the situation is even worse since an NCT of 2 or less can only be achieved for NCC larger than 4 (the corresponding value of M_s must exceed 128).

C. Discussion

The results presented above, and those of several other computer experiments that we have run with different types of signals and operating conditions, seem to support the following general conclusions and statements:

1. Contrary to the basic assumptions made in the derivation of FNTF [5], the selection of the parameter M cannot be based strictly on the necessary AR modelling order of the source signal $u(n)$; it is also strongly dependent (increasing function) on the filter length N. A possible explanation for this effect is the inadequacy of the extrapolation procedure in FNTF when $N \gg M$ and the signal is nonstationary, as is the case with CSS or speech. Indeed, referring to (8)–(10), we find that it takes $N-M$ samples for the predictor coefficients $\mathbf{a}_M(n-1)$ to affect all the entries of the gain vector $\mathbf{c}(n)$. This procedure for building $\mathbf{c}(n)$ may be effective only if the signal remains stationary over $N-M$ samples; if not, even the interpretation of $\mathbf{c}(n)$ as a form of dual-Kalman-gain vector is questionable.
2. Although the performance of the subband FNTF is generally better than that of its fullband counterpart, the use of subband processing is not effective in reducing the normalized prediction order, i.e., the ratio M/N necessary for achieving FRLS-like performance with FNTF. In fact, to achieve a similar gain in convergence speed, a ratio of M_s/N_s must be used with subband FNTF that is larger than the value of M/N required for the fullband FNTF. As a possible explanation for this behaviour, we note that the correlation time of the decimated subband components $u_b(m)$ may be longer than that of the original signal $u(n)$, due to the spectral shaping that results from narrowband filtering and oversampling. Further explanations and related comments may be found in Section IV.D below.
3. In the case of long adaptive filters, such as those typically employed in audioconference applications of AEC, the use of FNTF in the subband adaptive filtering scheme appears to be of limited practical value. Indeed, the large values of M needed to achieve significant improvement in convergence speed, i.e., comparable to an FRLS approach, render this algorithm non-attractive from a computational viewpoint. In addition, FNTF suffers from the following practical difficulties: numerical instability of the FRLS predictors (even with the so-called stabilized versions), requiring periodic restart; sensitivity of convergence and tracking performance to the parameters λ and ρ; and difficulty in optimizing the latter.

For small values of M, it appears that the performance improvements resulting from the use of FNTF might be obtained via alternative and simpler approaches, such as the use of an adaptive step-size strategy in NLMS. With respect to the performance improvements observed for larger values of M, we point out that other families of fast algorithms have appeared recently that offer a gradual trade-off between the performance of NLMS and FRLS via the selection of an additional prediction parameter M. Among these, the fast affine projection (FAP) [27] appears to be of particular interest for subband AEC in audioconference applications (i.e., for long impulse responses), for it can achieve FRLS-like performance with relatively small values of M [19], [28].

D. Further notes on the decorrelating effects of the subband decomposition[8]

Some readers might argue that the lack of significant performance improvements in the subband FNTF scheme with increasing prediction order M might have been anticipated on the basis that the subband decomposition provides the bulk of the decorrelation that would normally have been provided by the FNTF predictors in a fullband scheme. However, as explained below, this "subband decorrelation viewpoint" is inaccurate.

While it is true that subband processing provides decorrelation of the input signal in the critical sampling case (i.e., $K = B$) with ideal analysis bandpass filters, the situation is not so straightforward in the practical oversampling scheme (i.e., $K < B$) with non-ideal realizable analysis filters. For simplicity, first consider the case of a white-noise input signal. Assume that a non-ideal analysis filter $h(n)$ is used in the Weaver SSB modulator of Fig. 4, and that $K < B$, corresponding to oversampling. Then, as can be easily inferred from Fig. 5, the resulting signal at the modulator output will have a non-flat power spectral density, with deep notches at $\omega = 0$ and $\pm\pi$, whose specific shapes depend on the characteristics of the analysis filter (transition band and stopband attenuation) as well as the downsampling factor K. Corresponding to these spectral variations, the correlation matrix of the subband output signal will exhibit a large eigenvalue spread (i.e., $\gg 1$). Thus, even with a white-noise input signal, the subband decomposition process (with non-ideal $h(n)$ and $K < B$) will in fact introduce correlation in the subband signals at the output of the analysis bank. As a result, a subband NLMS adaptive filter operating in the oversampling scheme on a white-noise input will converge more slowly than its fullband counterpart, as experimentally evidenced in, for example, [19]. In the case of a more complex signal like CSS or speech, it is very difficult, if not impossible, to predict the combined effects of in-band decorrelation due to subband processing and induced correlation due to non-ideal analysis bandpass filter characteristics in the oversampling scheme. As a result, we believe that the behaviour of the oversampled subband FNTF algorithm could not have been anticipated on the basis of subband decorrelation alone.

To further support this claim, we refer the reader to [28] and [19], where a counter-example of the subband decorrelation viewpoint is provided by the FAP algorithm, which is a computationally fast version of the affine projection algorithm (APA) [27]. The FAP algorithm

[7] For a transversal adaptive filter of length N, the computational complexity of NLMS is about $2N$, while that of a stabilized FRLS is about $8N$, thus yielding an NCC of 4 for FRLS.

[8] The discussion below was motivated by comments from one of the anonymous reviewers.

is characterized by an integer parameter p, a so-called projection order, which plays a role somewhat similar to that of M in the FNTF algorithm. By varying p between 1 and N, FAP offers a trade-off between the low complexity of NLMS and the fast convergence of FRLS. In fact, the affine orthogonal projection mechanism inherent in FAP may be viewed as a kind of decorrelation process, with the projection order p controlling the degree of decorrelation (see also [29]). It turns out that with the oversampled subband FAP scheme, and under operating conditions similar to the ones in this study, a small increase in the projection order p (say from 1 to 8) results in significantly faster convergence of the subband algorithm. This indeed shows that a significant level of decorrelation may be achieved with the internal FAP prediction scheme, or equivalently, that there remains significant correlation after the subband decomposition.

The results of an independent investigation conducted in our laboratory seem to indicate that the FNTF predictors are not as effective as the FAP predictors in removing the remaining subband correlation in the case of CSS and speech signals [30].

V. Conclusions

In this work, we investigated the performance of a so-called Weaver SSB subband FNTF algorithm for AEC of the long echo paths that are typically associated with the use of hands-free audio terminals in offices (e.g., audioconferencing). Based on numerous computer experiments with various types of signals, we have come to the conclusion that the practical merits of this scheme are apparently extremely limited. Indeed, to achieve significant gains in convergence and tracking speed of the subband FNTF algorithm, as compared to a conventional subband NLMS approach, large values of the FNTF prediction order M must be used, thereby making the algorithm uncompetitive from a computational viewpoint.

Of course, we do not claim that our study is exhaustive in the sense that all possibilities for making FNTF work properly in the subband scheme have been explored. Indeed, there may be some clever and ingenious way of using, modifying and/or implementing FNTF so that its apparent limitations are overcome, at least partially. We do not discard such possibilities, nor do we discourage further research in this direction; however, based on our investigation and our working experience with FNTF, we tend to believe that such possibilities are remote. Thus, in the absence of further theoretical knowledge, we conclude that the use of the subband FNTF algorithm is apparently not the most computationally efficient way to improve the convergence performance of subband adaptive filtering in AEC of long echo paths. There seem to be more promising alternatives, such as the subband FAP algorithms described in [19], [28] and [31].

Acknowledgements

Support for this work was provided by a grant from Formation de Chercheurs et Aide à la Recherche (FCAR), Government of Quebec. The authors would like to thank the anonymous reviewers for their comments and suggestions.

References

[1] E. Hansler, "The hands-free telephone problem—An annotated bibliography," *Signal Process.*, vol. 27, pp. 259–271, 1992.

[2] A. Gilloire, "Recent advances in adaptive filtering algorithms for acoustic echo cancellation," in *Proc. Int. Workshop on Acoustic Echo and Noise Control*, Røros, Norway, June 1995, pp. 115–134.

[3] ———, "Experiments with sub-band acoustic echo cancellers for teleconferencing," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 4, Dallas, Tex., 1987, pp. 2141–2144.

[4] W. Kellermann, "Analysis and design of multirate systems for cancellation of acoustical echoes," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 5, New York, N.Y., 1988, pp. 2570–2573.

[5] G.V. Moustakides and S. Theodoridis, "Fast Newton transversal filters—A new class of adaptive estimation algorithms," *IEEE Trans. Signal Process.*, vol. 39, no. 10, pp. 2184–2193, Oct. 1991.

[6] T. Petillon, A. Gilloire, and S. Theodoridis, "The fast Newton transversal filter: An efficient scheme for acoustic echo cancellation in mobile radio," *IEEE Trans. Signal Process.*, vol. 42, no. 3, pp. 509–518, Mar. 1994.

[7] ———, "A comparative study of efficient transversal algorithms for acoustic echo cancellation," in *Proc. EUSIPCO 92*, vol. 1, Brussels, Belgium, 1992, pp. 119–122.

[8] G. Carayannis, D.G. Manolakis, and N. Kalouptsidis, "A fast sequential algorithm for least-squares filtering and prediction," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 31, no. 12, pp. 1394–1402, Dec. 1983.

[9] J. Cioffi and T. Kailath, "Fast, recursive, least squares transversal filters for adaptive processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 32, no. 4, pp. 304–337, Apr. 1984.

[10] S. Haykin, *Adaptive Filter Theory*, 2nd ed., Englewood Cliffs, N.J.: Prentice Hall, 1991.

[11] A. Benallal and A. Gilloire, "A new method to stabilize fast RLS algorithms based on first-order model of the propagation of numerical errors," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 3, New York, N.Y., 1988, pp. 1373–1376.

[12] ———, "Improvement of the convergence speed and tracking capability of the numerically stable FLS algorithms for adaptive filtering," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 2, Glasgow, Scotland, 1989, pp. 1031–1034.

[13] P. Chu, "Weaver SSB subband acoustic echo canceller," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, N.Y., 1993, pp. 8–11.

[14] B. Farhang-Boroujeny and Z. Wang, "Adaptive filtering in subbands: Design issues and experimental results for acoustic echo cancellation," *Signal Process.*, vol. 61, pp. 213–223, 1997.

[15] Q.G. Liu, B. Champagne, and D. Ho, "Simple design of oversampled uniform DFT filter banks with applications to subband acoustic echo cancellation," *Signal Process.*, vol. 80, pp. 831–847, 2000.

[16] H. Perez and A. Amano, "A new subband echo canceller structure," *Trans. IEICE*, vol. E-73, pp. 1625–1631, Oct. 1990.

[17] S. Gay and R.J. Mammone, "Fast converging subband acoustic echo cancellation using RAP on the WE DSP16A," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 2, Albuquerque, N.M., 1990, pp. 1141–1144.

[18] M Tahernezhadi and D. Sosale, "Performance evaluation of a DSP-based teleconferencing system," *IEEE Trans. Consumer Electron.*, vol. 40, pp. 957–962, Nov. 1994.

[19] S. Makino, J. Noebauer, Y. Haneda, and A. Nakagawa, "SSB subband echo canceller using low-order prediction algorithm," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 2, Atlanta, Ga., 1996, pp. 945–948.

[20] Q.G. Liu and B. Champagne, "Simple design of filter banks for subband adaptive filtering," in *Proc. 1997 Int. Workshop on Acoustic Echo and Noise Control*, London, U.K., Sept. 1997, pp. 132–135.

[21] R.E. Crochiere and L.R. Rabiner, *Multirate Digital Signal Processing*, Englewood Cliffs, N.J.: Prentice Hall, 1983.

[22] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Englewood Cliffs, N.J.: Prentice Hall, 1993.

[23] H.W. Gierlich, "A measurement technique to determine the transfer characteristic of hands-free telephone," *Signal Process.*, vol. 27, pp. 281–300, 1992.

[24] ITU-T, COM XII-13-E, "Influence of the exciting signal and the starting point on the convergence behaviour of echo cancellers with or without decorrelation filters," Mar. 1993.

[25] P.M. Peterson, "Simulating the response of multiple microphones to a single acoustic source in a reverberant room," *J. Acoust. Soc. Amer.*, vol. 80, pp. 1527-1529, Nov. 1986.

[26] ITU-T, "Acoustic echo controllers," Recommendation G.167, Mar. 1993.

[27] S.L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 5, Detroit, Mich., 1995, pp. 3023–3026.

[28] Q.G. Liu and B. Champagne, "On the use of a modified fast affine projection algorithm in subbands for acoustic echo cancellation," in *Proc. 1996 IEEE Digital Signal Processing Workshop*, Loen, Norway, Sept. 1996, pp. 354–357.

[29] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Comm. Japan*, vol. 67-A, no. 5, 1984.

[30] Q.G. Liu and B. Champagne, "Adaptive filters for acoustic echo cancellation," INRS-Télécommunications, Montreal, Que., Tech. Rep. 96-15, June 1996.

[31] M. Ghanassi and B. Champagne, "Subband acoustic echo cancellation using the FAP-RLS algorithm: Fixed-point implementation issues," in *Acoustic Signal Processing for Telecommunication*, S.L. Gay and J. Benesty, Eds., Boston: Kluwer Academic, 2000.