Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Signal Processing 86 (2006) 1182–1192

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Adaptive algorithms for sparse echo cancellation

Patrick A. Naylor*, Jingjing Cui, Mike Brookes

Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Road, London SW7 2AZ, UK

Received 10 January 2005; received in revised form 8 April 2005; accepted 8 July 2005
Available online 19 October 2005

Abstract

The cancellation of echoes is a vital component of telephony networks. In some cases the echo response that must be identified by the echo canceller is sparse, as for example when telephony traffic is routed over networks with unknown delay such as packet-switched networks. The sparse nature of such a response causes standard adaptive algorithms including normalized LMS to perform poorly. This paper begins by providing a review of techniques that aim to give improved echo cancellation performance when the echo response is sparse. In addition, adaptive filters can also be designed to exploit sparseness in the input signal by using partial update procedures. This concept is discussed and the MMax procedure is reviewed. We proceed to present a new high performance sparse adaptive algorithm and provide comparative echo cancellation results to show the relative performance of the existing and new algorithms. Finally, an efficient low cost implementation of our new algorithm using partial update adaptation is presented and evaluated. This algorithm exploits both sparseness of the echo response and also sparseness of the input signal in order to achieve high performance without high computational cost.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Echo cancellation; Adaptive filters; Sparse system identification

1. Introduction

Adaptive system identification is a challenging problem especially when the system impulse response is sparse. In this paper, we consider an impulse response or input signal to be 'sparse' if a large fraction of its energy is concentrated in a small fraction of its duration. We refer to the degree of sparseness as a qualitative measure ranging from strongly dispersive to strongly sparse.

One of the important applications of adaptive system identification is the cancellation of echoes in telephony networks as depicted in Fig. 1 in which, for discrete-time index n, $x(n)$ is the input signal, $e(n)$ is the returned echo signal, $\mathbf{h}_{\text{opt}}(n)$ is the echo path impulse response, $\mathbf{h}(n)$ is the adaptive filter's impulse response of length L and $\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-L+1)]^{\text{T}}$ is a vector of input samples. This application has been the subject of much research in recent years. The echo responses that must be identified by such an echo canceller can have sparse characteristics, for example, if echo arises from reflections at the hybrid transformer in telephony networks with unknown delay. The advent of packet-switched telephony has led to a need for the integration of older analog 'plain old telephone systems' (POTS) with modern IP or ATM packet-switch networks. Network gateway products address this need by facilitating the interconnection

*Corresponding author.
E-mail address: p.naylor@imperial.ac.uk (P.A. Naylor).

0165-1684/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2005.09.015

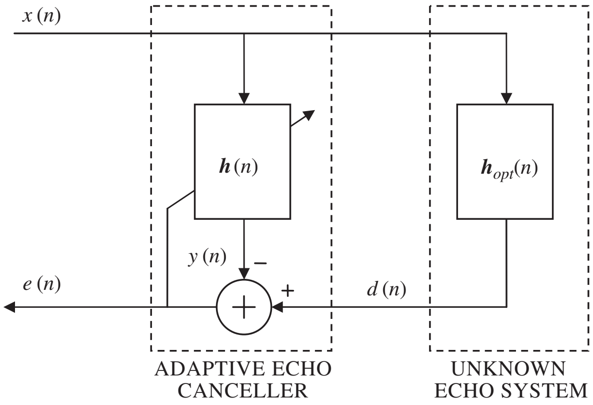

Fig. 1. Adaptive echo cancellation structure.

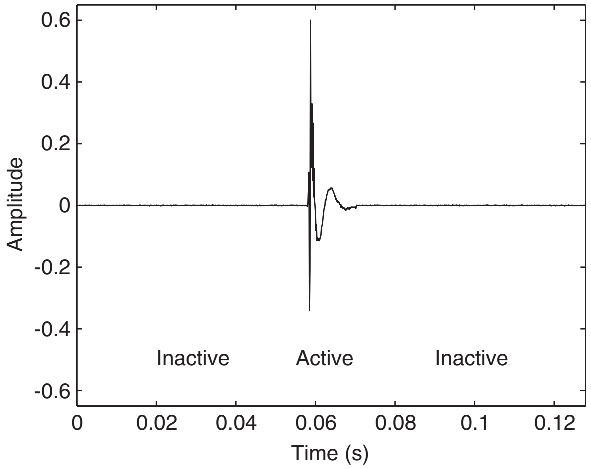

Fig. 2. An example of a sparse impulse response.

of various networks and providing appropriate echo cancellers. In such systems, the hybrid echo response will be subject to an unpredictable bulk delay because propagation delays in the network are unknown and depend on several factors including network loading, quality of service constraints and jitter-buffer configuration. The overall effect is therefore that an ‘active’ region associated with the true hybrid echo response will be located with an unknown delay within an overall response window that has to be sufficiently long to accommodate the worst case bulk delay. Fig. 2 shows an example of a sparse system with an overall response window of 128 ms duration with an active region containing a hybrid response of 12 ms duration. State-of-the-art network gateway echo cancellers are typically designed to cancel up to three independently delayed hybrid echo responses, each of up to 16 ms duration, within an overall window of 128 ms. Such multi-hybrid responses occur in multi-party conference calls. Although our main focus in this paper will be on the single hybrid case, many of the principles apply also to multi-hybrid responses. It has been shown [1] that direct application of normalized LMS (NLMS) [2] echo cancellation in the context of G.168 [3] testing gives unsatisfactory performance when the echo response is sparse. The causes of poor performance include (i) the requirement to adapt a relatively long filter, typically 128 ms corresponding to 1024 coefficients for narrow-band speech with a sampling frequency of 8 kHz, and (ii) the coefficient noise that will unavoidably occur during adaptation for the near-zero-valued coefficients in the inactive regions.

In hands-free telephones and desktop conferencing systems, control of echo due to acoustic coupling from the loudspeaker to the microphone is a key requirement [4]. The echo response due to the loudspeaker-room-microphone system can often be considered sparse because of the bulk delay corresponding to the direct path propagation delay from loudspeaker to microphone. Depending on the application, this direct path bulk delay may be predictable as in the case of a telephone handset, or unpredictable as in the case of a desktop conferencing system with physically separate loudspeakers and microphones. The length of the acoustic echo response in a typical teleconferencing room is in the region of 100 to 400 ms and hence adaptive filters employing 1024 taps or more are typically required in order to achieve adequate levels of echo cancellation [5]. The requirement for long adaptive filters together with the presence of near-zero-valued coefficients during the bulk delay region means that significant performance benefits can often be obtained by applying sparse echo cancellation methods to acoustic echo cancellation applications. Other examples which require the identification of sparse impulse responses include certain types of source localization and the control of feedback in sound reinforcement and hearing aid applications.

In addition to consideration of sparse echo responses, adaptive filters can also be designed to exploit sparseness in the input signal. This second type of sparseness can be usefully exploited when processing speech signals which can be considered to exhibit a degree of sparseness since many of the sample amplitudes are close to zero, for example

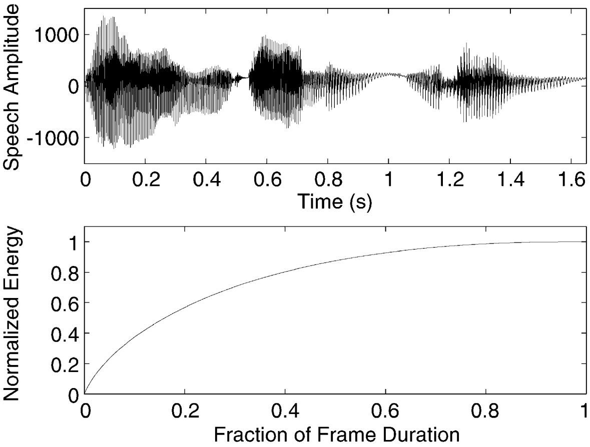

Fig. 3. Illustration of sparseness in a sentence of male speech.

during speech pauses and plosive phonemes. This is illustrated in Fig. 3 which shows, for a typical sentence of male speech analyzed using a frame duration of 128 ms, that 50% of the speech energy is contained within 16% of the frame duration. Partial update adaptive filters can be deployed to exploit this type of sparseness with the aim of reducing computational complexity. Since the magnitudes of tap-updates in LMS-based adaptive filters are proportional to the input sample magnitudes, the tap-updates for near-zero taps will be commensurately small and have little effect on reducing the error. By updating only those taps that will have the greatest effect in reducing the output error, the computations relating to the non-updated taps are saved. Early work on this topic appeared in [6] and subsequently the MMaxNLMS algorithm was reported in [7]. Several other approaches employing partial update adaptive filtering have been proposed including block-based methods [8] and multi-channel techniques [9].

Having now identified two types of sparseness—sparseness in the echo response and sparseness in the input signal—we will first discuss techniques designed to exploit them individually. In Section 2, we will review the technique of proportionate updating in adaptive filters and give examples of several algorithms from the literature; algorithms employing partial updating will be discussed in Section 3. Secondly, we will present a new modified technique for proportionate updating with improved performance in Section 4. Thirdly, we will combine the underlying concepts of proportionate updating and partial updating and, using our modified proportionate scheme, develop the efficient partial update improved proportionate NLMS algorithm (PIPNLMS). Comparative simulation results will be presented in Section 5 whereafter conclusions will be drawn.

2. Proportionate update adaptive filters

In gradient descent and stochastic gradient adaptive algorithms, an objective function is minimized iteratively by adjusting the parameter vector in the direction of the negative error-gradient, or its estimate. The magnitude of the adjustment is regulated by the step-size (or adaptive gain) parameter. The particular feature of proportionate updating is that the effective value of the step-size is determined for each coefficient in a manner dependent on the magnitude of the coefficient. The concept of proportionate updating can be supported intuitively by considering a system identification example in which some tap coefficients are close to zero whilst others are relatively large in magnitude, such as in the example of Fig. 2. By employing a step-size proportional to each coefficient magnitude, those coefficients with the greatest magnitude will be adapted with high adaptation gain causing them to converge quickly. This fast convergence of the large magnitude coefficients gives excellent performance in initial convergence since errors in these coefficients will contribute in greater proportion to the overall error than errors in the small magnitude coefficients.

2.1. Proportionate normalized LMS

The concept of proportionate updating of the NLMS algorithm was originally introduced for echo cancellation applications by Duttweiler in a Bell Laboratories internal technical memo and subsequently in [10]. The underlying principle of PNLMS is to adapt each coefficient with an adaptation gain proportional to its own magnitude as shown in (1)–(6). At initialization of the adaptive filter $\mathbf{h}(n) = \mathbf{0}$, the error contributions due to the coefficients in the active region are most dominant. In subsequent iterations, PNLMS updating applies accordingly high adaptation gain to these coefficients, causing them to converge quickly, thereby obtaining a fast initial reduction in error. A low adaptation gain is applied by PNLMS to the coefficients in the inactive regions, because of their low amplitude, in order that the final misadjustment of the proportionate scheme is no worse than comparable non-proportionate schemes. With reference to Fig. 1, the PNLMS algorithm is given by

$$e(n) = d(n) - \sum_{l=0}^{L-1} h_l(n)x(n-l), \tag{1}$$

$$\gamma_{\min}(n) = \rho \max\{\delta_p, |h_0(n)|, |h_1(n)|, \ldots, |h_{L-1}(n)|\}, \tag{2}$$

$$\gamma_l(n) = \max\{\gamma_{\min}(n), |h_l(n)|\}, \quad 0 \leqslant l < L, \tag{3}$$

$$g_l(n) = \frac{\gamma_l(n)}{(1/L)\sum_{i=0}^{L-1}\gamma_i(n)}, \quad 0 \leqslant l < L, \tag{4}$$

$$\mathbf{G}(n) = \mathrm{diag}\{g_0(n), \ldots, g_{L-1}(n)\}, \tag{5}$$

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu\mathbf{G}(n)\mathbf{x}(n)e(n)}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{x}(n) + \delta_{\mathrm{PNLMS}}}, \tag{6}$$

where δ_{PNLMS} is the regularization constant, L is the length of the adaptive filter and $\mathrm{diag}\{c_0, \ldots, c_{L-1}\}$ is a diagonal matrix with elements $\{c_0, \ldots, c_{L-1}\}$. Typical values for the algorithm constants are given as $\delta_p = 0.01$ and $\rho = 5/L$. An important modification to this algorithm is made in [11] and [12] in which (6) is rewritten as

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu\mathbf{G}(n)\mathbf{x}(n)e(n)}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{G}(n)\mathbf{x}(n) + \delta_{\mathrm{PNLMS}}}. \tag{7}$$

This improves the behavior of the algorithm particularly during initial convergence by scaling the denominator term to balance the proportionate scaling applied in the numerator. The effect of this modification is more significant when the coefficients $|h_l(n)|$, $0 \leqslant l < L$ are far from unity.

2.2. PNLMS++

Although the initial convergence of PNLMS is faster than NLMS when the echo response to be identified is sparse, convergence can be worse than NLMS when the response is more dispersive, in which case the proportionate adaptation gain control of PNLMS is not appropriate [12]. Furthermore, for time-varying systems, it can be seen that PNLMS will perform poorly if the trajectory of a coefficient is required to track through or close to zero since the adaptation gain for that coefficient will become inappropriately small. In [13], it was shown that alternating the coefficient update between PNLMS and NLMS gave similar performance to PNLMS for sparse systems but with better robustness in particular to echo path change. The resulting algorithm, known as PNLMS++, can be used with various switching schemes and good results have been presented when PNLMS and NLMS are used alternately at odd and even sample instants n, respectively.

2.3. IPNLMS

The advantage of proportionate update adaptive filters for sparse echo response identification has been clearly established in the literature [10,13]. The benefits of proportionate updating increase with the sparseness of the system and reduce as the unknown system becomes more diffuse. At some degree of diffuseness, proportionate updating begins to degrade performance compared to non-proportionate adaptation. In the search for an adaptation rule which gives performance always better than NLMS and PNLMS, regardless of whether the unknown systems is sparse or dispersive, Benesty and Gay have proposed improved PNLMS (IPNLMS) [12]. This improved algorithm employs a combination of proportionate and non-proportionate updating and therefore has a similar underlying rationale to PNLMS + +. However, in IPNLMS the proportionate and non-proportionate adaptation steps are merged so that a mixture of both is performed every iteration as shown in (8)–(10). The relative weighting of proportionate and non-proportionate adaptation at each iteration is controlled by a parameter α in the range ±1. Values of -0.5 or 0 are typically used, thereby giving equal or somewhat lower weight to the proportionate component of adaptation compared to the non-proportionate component. The IPNLMS algorithm employs an adaptation scheme given by

$$k_l(n) = \frac{1-\alpha}{2L} + (1+\alpha)\frac{|h_l(n)|}{2\|\mathbf{h}(n)\|_1 + \varepsilon} \quad l = 0, 1, \ldots, L-1, \tag{8}$$

$$\mathbf{K}(n) = \mathrm{diag}\{k_0(n), \ldots, k_{L-1}(n)\}, \tag{9}$$

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu\mathbf{K}(n)\mathbf{x}(n)e(n)}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{K}(n)\mathbf{x}(n) + \delta_{\mathrm{IPNLMS}}}, \tag{10}$$

where δ_{IPNLMS} is the regularization parameter and ε is a small positive constant to avoid division by zero. In the test we have performed in Section 5 and in the results shown in [12], the IPNLMS algorithm consistently performs better than both NLMS and PNLMS using the form of (7). In [14], it has been shown that PNLMS using the form of (6) can outperform IPNLMS during the initial stages of convergence but at the expense of slower convergence thereafter.

2.4. Exponentiated gradient

The concept underpinning exponentiated gradient algorithms was employed in [15] to develop the EG± algorithm. In [16] it was subsequently shown that algorithms of this class can be effectively applied to sparse system identification and are closely related to proportionate update algorithms.

The family of exponentiated gradient algorithms is derived in [16] by considering the minimization of a cost function (after [17])

$$J[\mathbf{h}(n+1)] = D[\mathbf{h}(n+1), \mathbf{h}(n)] + \eta\varepsilon^2(n+1), \qquad (11)$$

where ε is the a posteriori error, the operator $D[\,]$ is some measure of distance and the positive constant η controls the relative weighting of 'correctiveness' compared to 'conservativeness' and is analogous to the step-size in LMS. The exponentiated gradient algorithm is obtained from (11) by choosing $D[\,]$ as the Kullback–Leibler divergence given by

$$D_{\mathrm{KL}}[\mathbf{h}(n+1), \mathbf{h}(n)] = \sum_{l=0}^{L-1} h_l(n+1)\log\left(\frac{h_l(n+1)}{h_l(n)}\right). \qquad (12)$$

In order to operate correctly with both positive and negative coefficients, as shown in [18], it is necessary to write

$$\mathbf{h}(n) = \mathbf{h}^+(n) - \mathbf{h}^-(n), \qquad (13)$$

where $\mathbf{h}^+$ and $\mathbf{h}^-$ contain only positive coefficients.

The concept of exponentiated gradient adaptation can be seen by considering, for example, the exponentiated gradient algorithm with unnormalized weights (EGU±) which is given in [15,19,16] as

$$e(n) = d(n)/\beta - [\mathbf{h}^+(n) - \mathbf{h}^-(n)]^{\mathrm{T}}\mathbf{x}(n), \qquad (14)$$

$$h_l^+(n+1) = h_l^+(n)\mathrm{e}^{\mu' x(n-l)e(n)}, \qquad (15)$$

$$h_l^-(n+1) = h_l^-(n)\mathrm{e}^{-\mu' x(n-l)e(n)}, \qquad (16)$$

where β is a non-zero constant reflecting the unnormalized nature of this algorithm and μ' controls the learning rate. It can be seen that the term corresponding to the gradient estimate in LMS appears here exponentiated and therefore this algorithm is equivalent to using LMS updating on the logarithm of the coefficients.

An interesting relationship has been highlighted in [16] between the exponentiated gradient algorithms and proportionate update algorithms. Consider the EG± algorithm shown in (17)–(21)

$$e(n) = d(n)/\beta - [\mathbf{h}^+(n) - \mathbf{h}^-(n)]^{\mathrm{T}}\mathbf{x}(n), \qquad (17)$$

$$r_l^+(n) = \exp\left[\frac{\mu''}{u}\, x(n-l)e(n)\right], \qquad (18)$$

$$r_l^-(n) = 1/r_l^+(n), \qquad (19)$$

$$h_l^+(n+1) = \frac{u h_l^+(n) r_l^+(n)}{\sum_{j=0}^{L-1}[h_j^+(n)r_j^+(n) + h_j^-(n)r_j^-(n)]}, \qquad (20)$$

$$h_l^-(n+1) = \frac{u h_l^-(n) r_l^-(n)}{\sum_{j=0}^{L-1}[h_j^+(n)r_j^+(n) + h_j^-(n)r_j^-(n)]}, \qquad (21)$$

for $l = 0, 1, \ldots, L-1$ with parameters $u \geqslant \|\mathbf{h}_{\mathrm{opt}}/\beta\|_1$ where $\mathbf{h}_{\mathrm{opt}}$ are the coefficients of the true echo response and μ'' controls the learning rate. For μ'' sufficiently small, the approximation $\exp(\varphi) \approx 1 + \varphi$ can be used to write the approximations [16]

$$r_l^+(n) \approx 1 + \frac{\mu''}{u}\, x(n-l)e(n), \qquad (22)$$

$$r_l^-(n) \approx 1 - \frac{\mu''}{u}\, x(n-l)e(n), \qquad (23)$$

$$\sum_{j=0}^{L-1}[h_j^+(n)r_j^+(n) + h_j^-(n)r_j^-(n)] \approx u + \frac{\mu''}{u}\, y(n)e(n) \approx u. \qquad (24)$$

The update Eqs. (20) and (21) can be combined and, using these approximations, written as

$$\begin{aligned} h_l(n+1) &= h_l^+(n+1) - h_l^-(n+1) \\ &= h_l(n) + \mu'' \frac{h_l^+(n) + h_l^-(n)}{\|\mathbf{h}^+(n)\|_1 + \|\mathbf{h}^-(n)\|_1} \\ &\quad \times x(n-l)e(n). \end{aligned} \qquad (25)$$

In this case, it can be seen that the term $\mu''((h_l^+(n) + h_l^-(n))/(\|\mathbf{h}^+(n)\|_1 + \|\mathbf{h}^-(n)\|_1))$ in EG± has the same effect as the terms $k_l(n)$ in the IPNLMS update (10). Whereas the relative weighting between proportionate and non-proportionate adaptation in IPNLMS is controlled by α, in EG± it is controlled by u. Hence, for small μ'', IPNLMS is a good approximation of the EG± algorithm. When comparing IPNLMS and EG± for practical applications, IPNLMS would normally be preferred since it requires no a priori knowledge of $\mathbf{h}_{\mathrm{opt}}$ and has lower computational complexity.

3. Partial update adaptive filters

It can be advantageous from the point of view of computational complexity to employ adaptive filters that update only a selected subset of size M, out of a total of L, coefficients at each iteration. Various techniques have been proposed in the literature which differ in the criteria used for selecting coefficients for updating. A particularly effective technique is the MMaxNLMS algorithm [7] that updates only M coefficients for which the corresponding elements of the tap-input vector $\mathbf{x}(n)$ are among the M largest in magnitude. This results in a computational saving of L–M updates, although the cost of the convolution required to compute the output error remains unchanged. The computational overhead in identifying the M largest elements of $\mathbf{x}(n)$ can be kept relatively small by employing a fast sort algorithm [20] or an efficient approximate sort technique [21].

The update employed in MMaxNLMS is defined as

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mathbf{Q}(n)\,\frac{\mu \mathbf{x}(n) e(n)}{\|\mathbf{x}(n)\|^2}, \tag{26}$$

where $\mathbf{Q}(n) = \mathrm{diag}\{\mathbf{q}(n)\}$ is a diagonal matrix formed from the elements of $\mathbf{q}(n)$ and the lth element of $\mathbf{q}(n)$ is determined as

$$q_l(n) = \begin{cases} 1, & |x_l(n)| \in M \text{ maxima of } |\mathbf{x}(n)|, \\ 0, & \text{otherwise}, \end{cases} \tag{27}$$

for $l = 0, 1, \ldots, L-1$.

In [22], the sparse partial update NLMS algorithm is developed (SPNLMS) in which the concept embodied in (26) and (27) is modified so that the tap selection criterion considers the product of the tap-input sample and the tap coefficient. In this case the tap-selection criterion is

$$q_l(n) = \begin{cases} 1, & |x_l(n)h_l(n)| \in M \text{ maxima of} \\ & \quad |\mathbf{x}(n) \odot \mathbf{h}(n)|, \\ 0, & \text{otherwise}, \end{cases} \tag{28}$$

for $l = 0, 1, \ldots, L-1$ where $\odot$ represents the element-by-element vector product. To simplify the control of this algorithm and avoid any problems when $\mathbf{h}(n) = \mathbf{0}$, the authors propose switching from (28) to (27) for 1 in T iterations.

Although the SPNLMS algorithm does not employ proportionate updating, it does exploit sparseness in both the echo response and the input signal by making the tap-selection criterion dependent on the product of these two values. Updating of a tap will be avoided if either the tap-input sample or the tap coefficient are sufficiently small. As will be discussed further, joint exploitation of system sparseness and signal sparseness is the concept that underlies the PIPNLMS algorithm developed in Section 4.

4. An improved IPNLMS algorithm with efficient partial update

Section 2 discussed the modification of stochastic gradient adaptive algorithms such that the effective adaptation step-size is proportional to the magnitude of each coefficient. This approach aims to improve the effectiveness of adaptive identification of sparse systems and leads to the PNLMS algorithm [10]. It has also been shown how pure proportionate updating can be improved upon by introducing an amount of non-proportionate updating so that, as in the resulting PNLMS++ and IPNLMS algorithms [12,13], a mixture of proportionate and non-proportionate updating is used. In PNLMS++ the relative weighting of proportionate and non-proportionate updating employed is controlled by the ratio of the number of PNLMS and NLMS iterations. The typical alternating scheme results in equal weighting. In IPNLMS, the relative weighting of the proportionate and non-proportionate terms of the update equation is set by the parameter α in (10). A value of $\alpha = 0$ corresponds to an equal weighting in IPNLMS.

In both PNLMS++ and IPNLMS that the relative weighting between NLMS- and PNLMS-type updating is identical for all taps. We now propose a modified approach for which the relative weighting of proportionate and non-proportionate updating is adjusted individually for each tap. An earlier version of this approach was outlined in [14]. We also present an efficient implementation of the modified approach employing partial updating.

In proportionate updating, taps with large magnitude, such as occur in the active region, are updated using proportionately large values of effective adaptation step-size. This is a great advantage during the early stage of initial convergence since fast adaptation of these large magnitude taps will quickly reduce both output error and normalized misalignment. However, when they have converged near to their optimal values, the large step-size will result in correspondingly large levels of coefficient noise at these taps. In contrast, taps with small magnitude, such as that occur in inactive regions, will be updated with proportionately small values of effective adaptation step-size because of the normalization applied to the gain vector in (4), and will therefore contribute correspondingly small levels of coefficient noise.

The aim of our improved algorithm is to benefit from the advantages of proportionate updating without paying the penalty in terms of coefficient noise for large magnitude taps. This is achieved by using two different values for α. For taps with large magnitude, α is chosen such that non-proportionate updating is weighted more strongly. For taps with small magnitude, α is chosen such that proportionate updating is favored. Implementation of this scheme requires a simple threshold above which taps are considered large. We have used a threshold in the form $\Gamma \times \max(\mathbf{h}(n))$ and determined experimentally that a reasonable choice of Γ is 0.1, though the algorithm is not very sensitive to this choice. The IIPNLMS algorithm is given by

$$e(n) = d(n) - \mathbf{x}(n)^{\mathrm{T}}\mathbf{h}(n), \tag{29}$$

$$g_l(n) = \max(\rho * \max(|\mathbf{h}(n)|), |h_l(n)|), \tag{30}$$

$$\mathbf{g}(n) = [g_0(n), g_1(n), \ldots, g_{L-1}(n)], \tag{31}$$

$$\alpha_l(n) = \begin{cases} \alpha 1, & g_l(n) > \Gamma \times \max(\mathbf{g}(n)), \\ \alpha 2, & g_l(n) \leqslant \Gamma \times \max(\mathbf{g}(n)), \end{cases} \tag{32}$$

$$k'_l(n) = \frac{1-\alpha_l(n)}{2L} + (1+\alpha_l(n))\frac{|h_l(n)|}{2\|\mathbf{h}(n)\|_1 + \varepsilon}, \tag{33}$$

$$\mathbf{K}'(n) = \mathrm{diag}\{k'_0(n), k'_1(n), \ldots, k'_{L-1}(n)\}, \tag{34}$$

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu \mathbf{K}'(n)\mathbf{x}(n)e(n)}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{K}'(n)\mathbf{x}(n) + \delta_{\mathrm{IIPNLMS}}}. \tag{35}$$

It has been seen in Section 3 how a reduction in computational complexity can be achieved through the use of partial updating schemes for adaptive filters. We now wish to integrate such a scheme into the IIPNLMS algorithm. The scheme selected employs an efficient approximation to the MMax tap-selection criterion. The MMaxNLMS partial update scheme [7] has been shown in [9] to introduce only a graceful degradation in convergence performance when $0.5L \leqslant M < L$. However, the reduction in computation made by updating only a subset of tap coefficients is offset to some degree by a computational overhead that arises from the need to sort the tap input vector for the M maximum elements at each iteration. For example, the SORTLINE procedure [20] requires $2\log_2 L + 2$ comparisons. An efficient approximation to the MMax scheme was introduced in [21] known as the short-sort MMax procedure.

The short-sort MMax procedure operates by considering a short segment of the tap input vector $\check{\mathbf{x}}(n) = [x(n), x(n-1), \ldots, x(n-S+1)]$ of length $S \ll L$. Once every S iterations, an efficient insertion sort [23] is performed on $\check{\mathbf{x}}$, as shown in [21], and A coefficients are selected corresponding to the elements of $\check{\mathbf{x}}$ with largest magnitude. This tap-selection is propagated through the filter by incrementing the indices of the selected coefficients by one at each sample period.

Combining the short-sort MMax tap-selection with IIPNLMS leads to the following PIPNLMS algorithm.

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mathbf{Q}(n)\frac{\mu \mathbf{K}'(n)\mathbf{x}(n)e(n)}{\mathbf{x}^{\mathrm{T}}(n)\mathbf{K}'(n)\mathbf{x}(n) + \delta_{\mathrm{IIPNLMS}}}, \tag{36}$$

where $\mathbf{K}'$ is given in (29) to (35),

if $(n \bmod S) = 0$

$$q_l(n) = \begin{cases} 1, & |\check{x}_l(n)| \in A \text{ maxima of } |\check{\mathbf{x}}|, \\ 0 & \text{otherwise}, \end{cases}$$

for $l = 0, 1, \ldots, S-1$,

$$\mathbf{Q}(n) = \mathrm{diag}\{q_0(n), \ldots, q_{S-1}(n), Q_{S-1,S-1}(n-1), \ldots, Q_{L-2,L-2}(n-1)\},$$

else

$$\mathbf{Q}(n) = \mathrm{diag}\{0, Q_{0,0}(n-1), Q_{1,1}(n-1), \ldots, Q_{L-2,L-2}(n-1)\},$$

where $Q_{i,j}(n)$ represents the i,jth element of matrix $\mathbf{Q}$ at sample n.

5. Simulation results

The convergence rate of the above algorithms has been compared in an echo cancellation experiment in which the echo response is sparse as shown in Fig. 2. Convergence rate has been measured in terms of the normalized misalignment $\|\mathbf{h}_{\mathrm{opt}} - \mathbf{h}(n)\|^2/\|\mathbf{h}_{\mathrm{opt}}\|^2$ where $\mathbf{h}_{\mathrm{opt}}$ are the coefficients of the true echo response. The first set of tests employed Gaussian

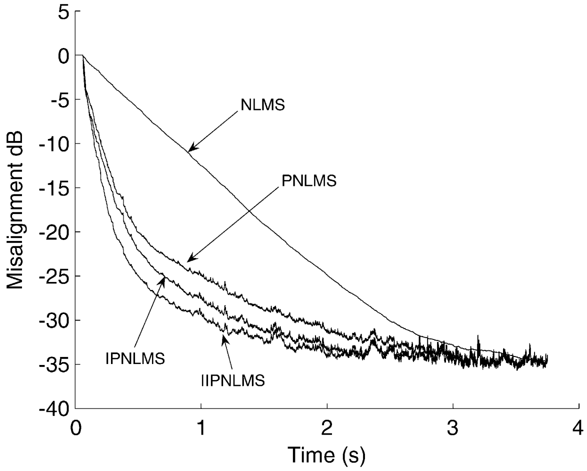

Fig. 4. Convergence for sparse echo response with Gaussian white noise input.

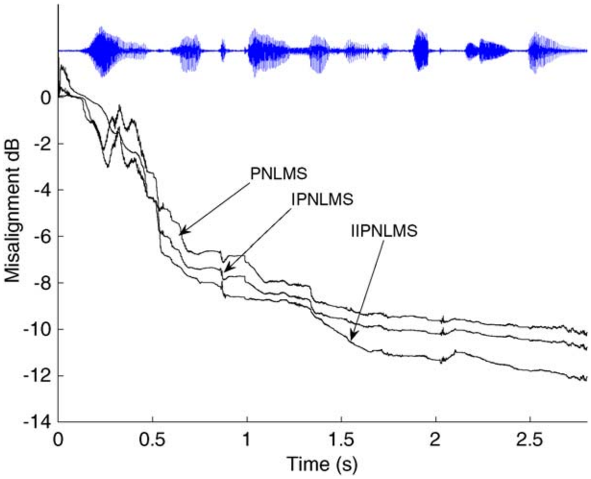

Fig. 5. Convergence for sparse echo response with speech signal input. The speech signal is shown in the upper plot.

distributed white noise input with 8 kHz sampling frequency and measurement noise was injected to give an SNR of 25 dB. The echo response, as shown in Fig. 2, and the adaptive filter were both of length $L = 1024$ and the other parameters were $\mu = 0.2$, $\rho = 0.01$, $\alpha = 0$, $\alpha 1 = -0.5$, $\alpha 2 = 0.5$ and $\Gamma = 0.1$. Fig. 4 shows the convergence curves for NLMS, PNLMS, IPNLMS and IIPNLMS. It can be seen, as expected, that all methods exhibit the same level of final misalignment. The convergence rate of IIPNLMS can be seen to be the fastest of the methods tested with around 2 to 3 dB less misalignment than IPNLMS during the main period of convergence. IPNLMS performs better than PNLMS by a similar margin. Finally, NLMS can be seen to have the slowest rate of convergence taking 1.58 s to converge to -20 dB compared to 0.31, 0.42 and 0.49 s for IIPNLMS, IPNLMS and PNLMS, respectively. Direct comparisons with the exponentiated gradient approach have been omitted here since, as discussed in Section 2.4, IPNLMS would normally be preferred in practice.

The second set of tests employed a speech signal from a male talker. The parameters were unchanged from the white noise tests except that μ was reduced to 0.1 as is typical for speech signals. The convergence results in Fig. 5 show the same performance ranking as in the tests with noise input, though the differences are somewhat decreased. To examine the operation of IIPNLMS, the values of $\alpha_l(n)$ obtained during the speech signal test of Fig. 5 have been plotted at intervals of 12.5 ms in Fig. 6. After initialization, $\alpha_l(n)$ takes the values of either +0.5 or -0.5 indicating an emphasis on PNLMS or NLMS, respectively. It can be seen that $\alpha_l(n)$ varies seemingly randomly for the first 5 analysis intervals, which corresponds to the bulk delay time of the sparse response. After this delay,

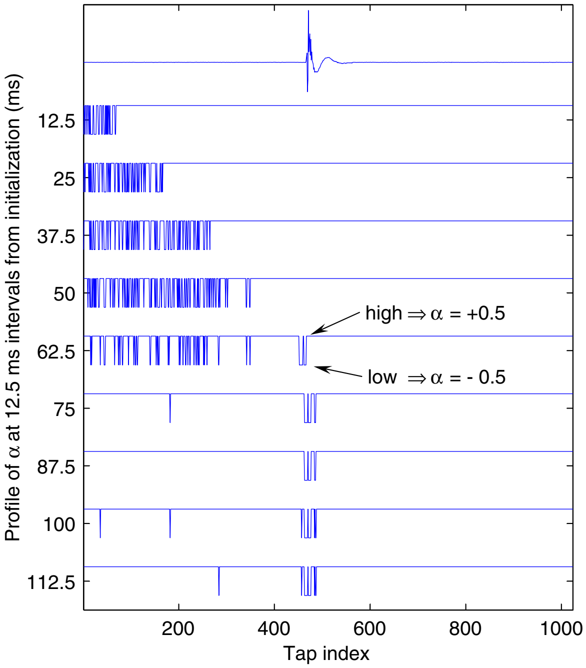

Fig. 6. Evolution of $\alpha_l(n)$ plotted at 12.5 ms intervals after initialization for speech signal input. Top trace shows sparse echo response. Note that the vertical axis labels refer to the elapsed time in ms. Within each horizontal trace, $\alpha_l(n)$ takes the values ±0.5.

at 62.5 ms, $\alpha_l(n)$ begins to detect the active region. After 75 ms of iterations, the active region is located almost exactly. It is clear from Fig. 6 that IIPNLMS successfully identifies active and inactive regions in the sparse echo response and so is able to control $\alpha_l(n)$ appropriately for each coefficient $h_l(n)$, $l = 0, 1, \ldots, L-1$.

A further set of tests has been conducted that illustrate the effectiveness of combining proportionate update and partial update—the former exploiting sparseness in the echo responses and the latter aimed at reducing complexity by exploiting sparseness in the input signal. These results are shown in Fig. 7. This plot compares the performance of fully updated IPNLMS and IIPNLMS to partially updated IIPNLMS algorithms employing MMax tap-selection, IIPNLMS-MMax, and our efficient approximate MMax scheme, PIPNLMS. White Gaussian noise input signals were used and measurement noise was injected at an SNR of 25 dB. The echo response, as shown in Fig. 2 and the adaptive filter were of length $L = 1024$ and $\mu = 0.2$, $\rho = 0.01$, $\alpha = 0$, $\alpha 1 = -0.5$, $\alpha 2 = 0.5$ and $\Gamma = 0.1$. It can be seen that, whereas the performance improvement for our fully updated IIPNLMS algorithm over IPNLMS is around 3 dB, the performance improvement when MMax partial updating is applied drops almost imperceptibly to around 2.5 dB for our IIPNLMS-MMax approach. The performance improvement over IPNLMS achieved by our lower-complexity PIPNLMS algorithm is around 2 dB.

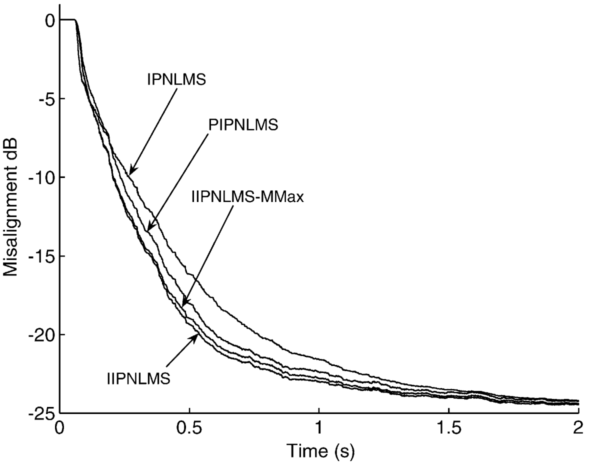

Fig. 7. Comparison of IPNLMS, IIPNLMS with MMax tap-selection and PIPNLMS. $M = 0.5L$, $S = 32$, $A = 16$.

The computational complexity order of the various algorithms is compared in Table 1 in terms of the number of multiply operations per iteration. Efficient recursive implementation of the denominator normalization terms in the coefficient update is assumed in all cases. The short-sort procedure used in PIPNLMS requires only $(A + A(S-A))/S$ comparisons per sample [21] and is assumed negligible compared to L.

6. Discussion and conclusions

The topic of sparse echo cancellation has been discussed, where a sequence is considered ‘sparse’ if a large fraction of its energy is concentrated in a small fraction of its duration. Two types of sparseness have been considered in the context of echo cancellation. Sparseness in the echo response occurs in network echo cancellation, particularly for packet-switched networks, and in acoustic echo cancellation, particularly when the direct acoustic path propagation causes significant bulk delay. Sparseness in the input signal occurs in speech signals to some degree, dependent on the utterance and talker characteristics.

The concept of proportionate updating is effective for identifying sparse echo responses. A number of different algorithms employing this concept have

Table 1
Computational complexity per sample period. The bracketed entries are an example with parameters chosen as in Fig. 7

Algorithm	Multiplies			Comparisons
	Convolution	Tap update	Total	
PNLMS	L	$2L$	$3L$	$2L$
IPNLMS	L	$3L$	$4L$	0
IIPNLMS	L	$3L$	$4L$	$4L$
PIPNLMS	L	$L + 2M$	$2(L + M)$	$4M + (A + A(S-A))/S$
(PIPNLMS)	(L)	$(2L)$	$(3L)$	$(2L + 8.5)$

been reviewed and evaluated. The results confirm that the performance of pure proportionate updating can be improved by incorporating a degree of nonproportionate updating such as in IPNLMS which performs most consistently well.

The trade-off between convergence speed and coefficient noise is an important consideration in this context. For large amplitude coefficients, proportionate updating applies a relatively large effective adaptive step-size. This is beneficial for fast convergence, leading to fast initial reduction in overall error, but introduces relatively high levels of coefficient noise. For small amplitude coefficients, a correspondingly small effective adaptive step-size is applied. This offsets the high coefficient noise on the large amplitude coefficients. The trade-off can be controlled to some extent by modifying the relationship between a coefficient amplitude and the effective adaptive step-size employed for its update. The directly proportional relationship used in PNLMS is modified by the addition of a constant term in IPNLMS and, on average, in PNLMS++. Exponentiation of the gradient estimate is an alternative approach which has been seen to correspond to proportionate updating when a slow learning rate is used. Determining an optimal function for this relationship is an interesting topic of ongoing work. For example, Deng and Doroslovacki [24] have considered the use of the μ-law function.

We have introduced the IIPNLMS algorithm which extends the concept of adjusting the relationship between the effective adaptation step-size and the coefficient amplitude. Whereas existing algorithms use the same relationship for all coefficients, the new algorithm can use different relationships. In IIPNLMS, the algorithm chooses between two different relationships depending on whether the coefficient is considered active or inactive. The motivation for the use of only two relationships comes from the network echo cancellation application in which coefficients can realistically be classified as active or inactive. However, the approach could, in principle, be extended more generally.

The PIPNLMS algorithm that we have presented brings together the two concepts of proportionate updating and partial updating. The proportionate updating is done using our IIPNLMS approach and the partial updating employs the MMax approach either in its true form or else using the efficient short-sort MMax approximation. In this way, PIPNLMS addresses both types of sparseness simultaneously with the aim of achieving improved sparse system identification whilst maintaining low computation complexity. Our results indicate that the performance degradation due to true MMax partial updating with $M = L/2$ is negligible. The complexity reduction associated with the short-sort MMax partial updating is shown in our results to carry a penalty of approximately 1 dB during initial convergence compared to fully updated IIPNLMS but nevertheless retains most of the performance advantage of the IIPNLMS algorithm.

Acknowledgements

The authors wish to thank Jacob Benesty for his helpful discussions and advice during the preparation of this manuscript.

References

[1] O. Tanrikulu, K. Dogancay, Selective-partial-update proportionate normalized least-mean-squares algorithm for network echo cancellation, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, 2002, pp. 1889–1892.

[2] S. Haykin, Adaptive Filter Theory, Prentice-Hall, Englewood Cliffs, NJ, 2002.

[3] ITU-T, Digital Network Echo Cancellers. G.168, International Telecommunication Union, Series G: Transmission Systems and Media, Digital Systems and Networks, 2002.

[4] C. Breining, P. Dreiscitel, E. Hansler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, J. Tilp, Acoustic echo control. An application of very-high-order adaptive filters, IEEE Signal Process. Mag. 16 (July 1999) 42–69.

[5] A. Gilloire, Experiments with sub-band acoustic echo cancellers for teleconferencing, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, 1987, pp. 2141–2144.

[6] S.C. Douglas, Adaptive filters employing partial updates, IEEE Trans. Circuits Systems II 44 (March 1997) 209–216.

[7] T. Aboulnasr, K. Mayyas, Complexity reduction of the NLMS algorithm via selective coefficient update, IEEE Trans. Signal Process. 47 (May 1999) 1421–1424.

[8] K. Dogancay, O. Tanrikulu, Adaptive filtering algorithms with selective partial updates, IEEE Trans. Circuits Systems II 48 (August 2001) 762–769.

[9] A.W.H. Khong, P.A. Naylor, Reducing inter-channel coherence in stereophonic acoustic echo cancellation using partial update adaptive filters, in: Proceedings of European Signal Processing Conference, 2004.

[10] D.L. Duttweiler, Proportionate normalized least-mean-squares adaptation in echo cancelers, IEEE Trans. Speech Audio Process. 8 (September 2000) 508–518.

[11] J. Benesty, T. Gänsler, D.R. Morgan, M.M. Sondhi, S.L. Gay, Advances in Network and Acoustic Echo Cancellation, Springer, Berlin, 2001.

[12] J. Benesty, S.L. Gay, An improved PNLMS algorithm, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, 2002, pp. 1881–1884.

[13] S.L. Gay, An efficient, fast converging adaptive filter for network echo cancellation, in: 32nd Asilomar Conference on Signals Systems and Computers, vol. 1, November 1998, pp. 394–398.

[14] J. Cui, P.A. Naylor, D.T. Brown, An improved IPNLMS algorithm for echo cancellation in packet-switched networks, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, vol. 4, 2004, pp. iv-141–iv-144.

[15] J. Kivinen, M.K. Warmuth, Exponentiated gradient versus gradient descent for linear predictors, Inform. Comput. 132 (January 1997) 1–63.

[16] J. Benesty, Y. Huang, D.R. Morgan, On a class of exponentiated adaptive algorithms for the identification of sparse impulse responses, in: J. Benesty, Y. Huang (Eds.), Adaptive Signal Processing: Applications to Real-World Problems, Springer, Berlin, 2003.

[17] S. Amari, Natural gradient works efficiently in learning, Neural Computation 10 (February 1998) 251–276.

[18] J. Benesty, Y. Huang, The LMS, PNLMS and exponentiated gradient algorithms, in: Proceedings of European Signal Processing Conference, 2004, pp. 721–724.

[19] S.I. Hill, R.C. Williamson, Convergence of exponentiated gradient algorithms, IEEE Trans. Signal Process. 49 (June 2001) 1208–1215.

[20] I. Pitas, Fast algorithms for running ordering and max/min calculation, IEEE Trans. Circuits Systems 36 (June 1989) 795–804.

[21] P.A. Naylor, W. Sherliker, A short-sort M-Max NLMS partial-update adaptive filter with applications to echo cancellation, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, 2003, pp. 373–376.

[22] H. Deng, M. Doroslovacki, New sparse adaptive algorithms using partial update, in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, 2004, pp. 845–848.

[23] D.E. Knuth, The Art of Computer Programming, vol. 3, Sorting and Searching, Addison-Wesley, Reading, MA, 1973.

[24] H. Deng, M. Doroslovacki, Modified PNLMS adaptive algorithm for sparse echo path estimation, in: Proceedings of Information, Sciences, Systems, March 2004, pp. 1072–1077.