Stereophonic Acoustic Echo Cancellation Employing Selective-Tap Adaptive Algorithms

Andy W. H. Khong, *Student Member, IEEE* and Patrick A. Naylor, *Member, IEEE*

Abstract—**Stereophonic acoustic echo cancellation has generated much interest in recent years due to the nonuniqueness and misalignment problems that are caused by the strong interchannel signal coherence. In this paper, we introduce a novel adaptive filtering approach to reduce interchannel coherence which is based on a selective-tap updating procedure. This tap-selection technique is then applied to the normalized least-mean-square, affine projection and recursive least squares algorithms for stereophonic acoustic echo cancellation. Simulation results for the proposed algorithms have shown a significant improvement in convergence rate compared with existing techniques.**

Index Terms—**Adaptive algorithms, partial-update, stereophonic acoustic echo cancellation, tap-selection.**

I. INTRODUCTION

STEREOPHONIC tele- and videoconferencing systems have gained much popularity [1]–[3] in recent years. In applications such as desktop conferencing and hands-free telephony, stereophonic systems provide telepresence to users by enabling listeners to localize speakers in conference meetings where multiple parties might be conversing at the same time. The stereophonic acoustic echo canceller (SAEC) as shown in Fig. 1 suppresses the echo returned to the transmission room so as to enable undisturbed communication between the rooms. This disturbance, caused by echo, increases in severity with the propagation delay of the channel.

A serious problem encountered in SAEC is that the echo canceller coefficients do not in general converge to the true impulse responses of the echo path when the adaptive filters, of length L, are greater than or equal to that of the transmission room's impulse responses, L_T. In such a situation, the solutions for the adaptive filters are nonunique and depend both on the transmission and receiving rooms' impulse responses [4].

In the practical case where $L < L_T$, the problem of nonuniqueness is ameliorated to some degree by the "tail" effect [4]. However, even in such cases, direct application of standard adaptive filtering is not normally successful because the system identification problem is ill-conditioned due to the high interchannel coherence between the two channels' tap-input vectors [2], [4]. This is known as the misalignment problem. Significant undermodeling of the unknown system can further degrade cancellation of echo. To overcome the

misalignment problem in this practical case, several approaches have been employed to decorrelate the two input signals using, for example, nonlinear processing [4]–[6], spectrally shaped random noise [7], [8], comb filtering [9], leaky extended LMS [10], and alternating fixed-point [11] algorithms. The common aim of these algorithms is to achieve decorrelation of input signals $x_1(n)$ and $x_2(n)$ without affecting the quality or stereophonic image of the speech.

In recent years, selective-tap schemes were introduced to reduce computational complexity of, in particular, the normalized least-mean-square (NLMS) algorithm by updating only a subset of taps at each iteration. The techniques of [12], [13], for example, allow implementation of selective-tap algorithms in single-channel echo cancellation with performance close to that of the full update NLMS algorithm. The reduction in computational complexity due to the partial updating is offset to some degree by the computational cost of tap-selection which normally requires a sort operation to be performed. However, efficient approximate schemes have been proposed which address this issue [14]. In this paper, our main motivation is *not* the reduction of complexity of SAEC. Instead, we propose to employ tap-selection as a means to reduce interchannel coherence. Our proposed tap-selection algorithm will be applied to NLMS, affine projection (AP), and the recursive least squares (RLS) algorithms for the SAEC application.

This paper is organized as follows: Section II reviews the SAEC problem. Section III discusses the single channel MMax–NLMS selective-tap algorithm and examines the effect of interchannel decorrelation when the MMax tap-selection criterion is applied in the stereophonic environment. The effect of our proposed exclusive tap-selection on interchannel coherence and the input autocorrelation matrix will also be discussed. An exhaustive tap-selection search technique is initially considered in Section IV to demonstrate the concept of the selective-tap approach in SAEC. In Section V, we proposed an efficient exclusive maximum (XM) tap-selection technique for SAEC involving adaptive filters of realistic order. This methodology is applied in combination with a nonlinear (NL) processor block to form the XMNL-based versions of NLMS, AP, and RLS algorithms. We will also consider the computational complexity of the proposed algorithms in Section V. Section VI presents simulation results while Section VII draws conclusions from our work.

II. STEREOPHONIC ACOUSTIC ECHO CANCELLATION

Stereophonic acoustic echo cancellation, as shown in Fig. 1, can be viewed as a multichannel extension of the single-channel
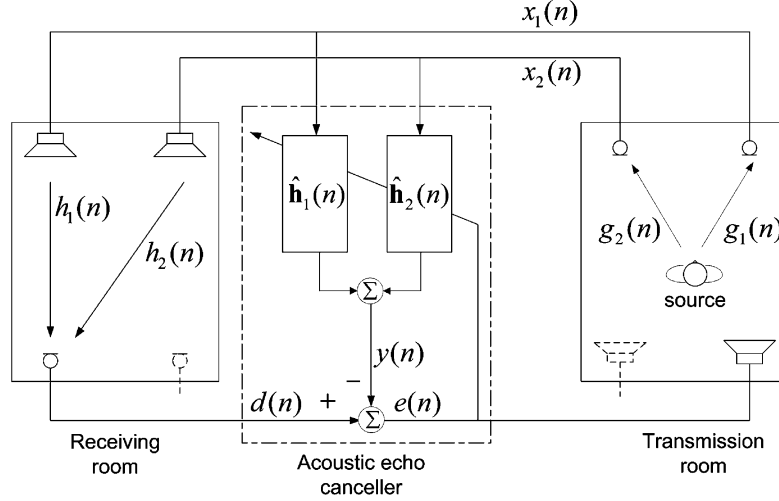
Fig. 1. Schematic diagram of stereophonic acoustic echo cancellation (after [4]). Only one channel of the return path is shown for simplicity.

echo cancellation concept. Two microphones are located in the transmission room depicted on the right. The speech signal is convolved with the transmission room impulse responses $g_1(n)$ and $g_2(n)$ to give $x_1(n)$ and $x_2(n)$, respectively. These stereophonic signals are then transmitted to loudspeakers in the receiving room which in turn are acoustically coupled to the receiving room microphones. We consider only one microphone at the receiving room here for simplicity since similar analysis can be applied to the other channel. We also consider a noiseless case in our discussion without loss of generality. Receiving room impulse responses $h_1(n)$ and $h_2(n)$ for channels $j = 1$, 2, respectively, produce the desired signal $d(n)$ which is given by

$$d(n) = \sum_{j=1}^{2} \mathbf{h}_j^T(n)\mathbf{x}_j(n) \qquad (1)$$

where the superscript ^{T} denotes vector transposition, $\mathbf{h}_j(n) = [h_{j,0}(n) \ h_{j,1}(n) \ \ldots \ h_{j,L-1}(n)]^T$, and $\mathbf{x}_j(n) = [x_j(n) \ x_j(n-1) \ \ldots \ x_j(n-L+1)]^T$. A pair of adaptive filters of length L function as a SAEC by estimating the unknown systems $h_1(n)$ and $h_2(n)$ using filter coefficients $\widehat{h}_1(n)$ and $\widehat{h}_2(n)$. The error signal between the desired signal and its estimate is, thus, given by

$$e(n) = d(n) - \sum_{j=1}^{2} \widehat{\mathbf{h}}_j^T(n)\mathbf{x}_j(n). \qquad (2)$$

If we now consider the use of the method of least squares following the approach of [4], the cost function can be defined as

$$J(n) = \sum_{i=1}^{n} \lambda^{n-i} e^2(i) \qquad (3)$$

where $0 \ll \lambda < 1$ is the forgetting factor. Minimizing the least squares criterion (3) and using (2), we obtain the set of normal equations

$$\widehat{\mathbf{h}}(n) = \Psi^{-1}(n)\Theta(n) \qquad (4)$$

where the weighted time-averaged autocorrelation matrix and cross-correlation vector are defined [4], respectively, as

$$\Psi(n) = \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i)$$
$$= \begin{bmatrix} \Psi_{11}(n) & \Psi_{12}(n) \\ \Psi_{21}(n) & \Psi_{22}(n) \end{bmatrix} \qquad (5)$$

and

$$\Theta(n) = \sum_{i=1}^{n} \lambda^{n-i} d(i)\mathbf{x}(i) \qquad (6)$$

given that $\mathbf{x}(n) = [\mathbf{x}_1^T(n) \ \mathbf{x}_2^T(n)]^T$ and $\widehat{\mathbf{h}}(n) = [\widehat{\mathbf{h}}_1^T(n) \ \widehat{\mathbf{h}}_2^T(n)]^T$ are the concatenated tap-input vectors and filter coefficients, respectively.

It is shown in [4] that when $L \geq L_T$, the solution of (4) is in the form

$$\begin{bmatrix} \widehat{\mathbf{h}}_1(n) \\ \widehat{\mathbf{h}}_2(n) \end{bmatrix} = \begin{bmatrix} \mathbf{h}_1(n) \\ \mathbf{h}_2(n) \end{bmatrix} + \beta(n)\begin{bmatrix} \mathbf{g}_2(n) \\ -\mathbf{g}_1(n) \end{bmatrix} \qquad (7)$$

where $\mathbf{g}_j(n) \in \mathbb{R}^{L_T}$, $j = 1$, 2 is appended with $L - L_T$ zeros and $\beta(n)$ is a scalar quantity. Equation (7) indicates that the solutions for $\widehat{\mathbf{h}}(n)$ are nonunique. In practical cases where $L < L_T$, the time-averaged autocorrelation matrix $\Psi(n)$ is ill-conditioned because the input signals $\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$ are highly correlated [4]. In the case where the filters are shorter than the length of the receiving room impulse response, L_R, a system mismatch error is introduced in the filter coefficients due to undermodeling. Defining $\breve{\mathbf{h}}_j(n)$ as that part of the impulse response of the jth channel in the receiving room which is not modeled by the respective adaptive filter such that $\breve{\mathbf{h}}(n) = [\breve{\mathbf{h}}_1^T(n) \ \breve{\mathbf{h}}_2^T(n)]^T$, and $\mathbf{h}(n) = [\mathbf{h}_1^T(n) \ \mathbf{h}_2^T(n)]^T$ as the concatenated true impulse responses, the system mismatch can be quantified [4] by the normalized misalignment $\eta(n)$

$$\eta(n) = \frac{\|\mathbf{h} - \widehat{\mathbf{h}}(n)\|^2}{\|\mathbf{h}\|^2}$$
$$= \frac{\breve{\mathbf{h}}^T(n)\breve{\Psi}^T(n)\Psi^{-2}(n)\breve{\Psi}(n)\breve{\mathbf{h}}(n)}{\mathbf{h}^T\mathbf{h}} \qquad (8)$$

such that $\| \cdot \|^2$ is defined as the squared l_2-norm operator and

$$
\begin{aligned}
\breve{\Psi}(n) &= \sum_{i=1}^{n} \lambda^{n-i} \begin{bmatrix} \mathbf{x}_1(i) \\ \mathbf{x}_2(i) \end{bmatrix} \begin{bmatrix} \chi_1^T(i-L) & \chi_2^T(i-L) \end{bmatrix} \\
&= \begin{bmatrix} \breve{\Psi}_{11}(n) & \breve{\Psi}_{12}(n) \\ \breve{\Psi}_{21}(n) & \breve{\Psi}_{22}(n) \end{bmatrix}
\end{aligned} \tag{9}
$$

is the time-averaged autocorrelation matrix formed from the "tails" of the input signal such that for the jth channel, $\chi_j(i - L) = [\, x_j(i - L) \quad x_j(i - L - 1) \ldots x_j(i - L_R + 1)\,]^T$.

The fundamental difference between a single channel and stereophonic case can be seen from (8) and (9). In the single-channel case, (9) collapses to only $\breve{\Psi}_{11}(n)$ which consequently reduces $\eta(n)$ in (8). In addition to the error caused by under-modeling of the unknown system, matrices $\breve{\Psi}_{12}(n)$ and $\breve{\Psi}_{21}(n)$ in (9), for the stereo case, are significant because of the high cross-correlation between the two input signals $\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$. As a result, the misalignment for SAEC is significantly higher than for the single channel AEC unless specific processing, such as described here, is employed.

III. MMAX TAP-SELECTION

Selective-tap schemes including sequential and periodic LMS [12] achieve complexity reduction by updating only a subset of taps at each iteration. The MMax–NLMS algorithm proposed in [13] has been shown to suffer only a modest degradation in convergence rate by updating taps corresponding to the M largest magnitude tap-inputs. A theoretical justification for selecting taps corresponding to the largest l_2-norm of tap-inputs is provided in [15] by formulating the problem in terms of the principle of minimum disturbance. In this section, we shall examine briefly the dependence of convergence rate on tap-selection by proposing a new measure $\mathcal{M}$ and investigate the effect of tap-selection on the interchannel coherence in the stereo case. The proposed new measure $\mathcal{M}$ will then be used for the stereo case as an optimization parameter in the subsequent development of our selective tap SAEC algorithms in Section IV.

A. MMax Performance Measure

In the single-channel MMax–NLMS algorithm [13], for an adaptive filter of length L, only those taps corresponding to the M largest magnitude tap-inputs are selected for updating at each iteration. The MMax–NLMS tap-update equation may be expressed as

$$
\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mathbf{Q}(n) \frac{\mu \mathbf{x}(n) e(n)}{\|\mathbf{x}(n)\|^2 + \delta} \tag{10}
$$

where δ and μ are the regularization and adaptive step-size parameters, respectively. The tap-selection matrix

$$
\mathbf{Q}(n) = \mathrm{diag}\{\mathbf{q}(n)\} \tag{11}
$$

contains elements in $\mathbf{q}(n)$ which are represented by

$$
q_i(n) = \begin{cases} 1, & |\mathbf{x}_i(n)| \in \{M \text{ maxima of } |\mathbf{x}(n)|\} \\ 0, & \text{otherwise,} \end{cases} \tag{12}
$$

for tap-indices $i = 1, 2, \ldots, L$.

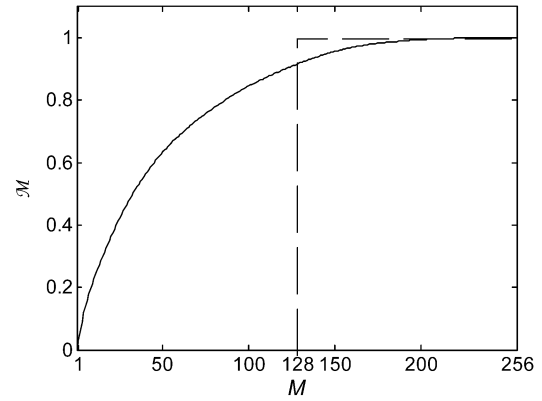The penalty incurred due to tap-selection in the MMax–NLMS algorithm is a decrease in convergence rate for

Fig. 2. Variation of $\mathcal{M}$ with subselection parameter M showing modest reduction of $\mathcal{M}$ within the region $0.5L \leq M < L$ for WGN sequence with $L = 256$.
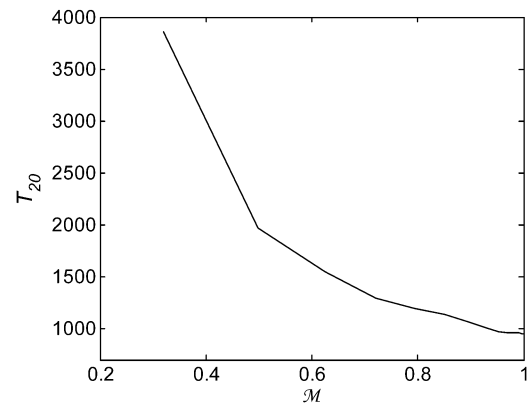

Fig. 3. Number of iterations to converge to -20 dB normalized misalignment as a function of $\mathcal{M}$ for $L = 256$.

a given μ [16]. For an adaptive filter of length L, we examine the dependence of convergence rate on the subselection using the measure $\mathcal{M}(n)$ as the ratio of the energy of the M selected tap-inputs to the energy of the full tap-input vector. Thus, $\mathcal{M}(n)$ may be written as

$$
\mathcal{M}(n) = \frac{\|\mathbf{Q}(n)\mathbf{x}(n)\|^2}{\|\mathbf{x}(n)\|^2}. \tag{13}
$$

Whereas the fundamental concept of MMax tap-selection was presented in [13], our measure $\mathcal{M}(n)$ provides an explicit quantification of the deviation of the selective-tap case from the full update case such that $\mathcal{M}(n) = 1$ corresponds to full update adaptation. Furthermore, $\mathcal{M}(n)$ allows direct extension to the multichannel case as will be shown in Section IV-B.

Fig. 2 shows how $\mathcal{M}$ varies with the size of tap-selection M for zero mean, unit variance white Gaussian noise (WGN) at a particular time iteration n. We note that $\mathcal{M}$ exhibits only a modest reduction for $0.5L \leq M < L$. Fig. 3 shows the number of iterations for MMax–NLMS to achieve -20 dB normalized misalignment for various $\mathcal{M}$ and, hence, verifies our expectation that, over the range $0.5L \leq M < L$, a graceful reduction in convergence rate is obtained as compared to full update adaptation when $M = L$ [17], [18]. Since convergence rate can be seen to increase monotonically with $\mathcal{M}$, we propose that any degradation in convergence performance due to subselection of taps can be minimized by selecting taps so as to maximize $\mathcal{M}$.
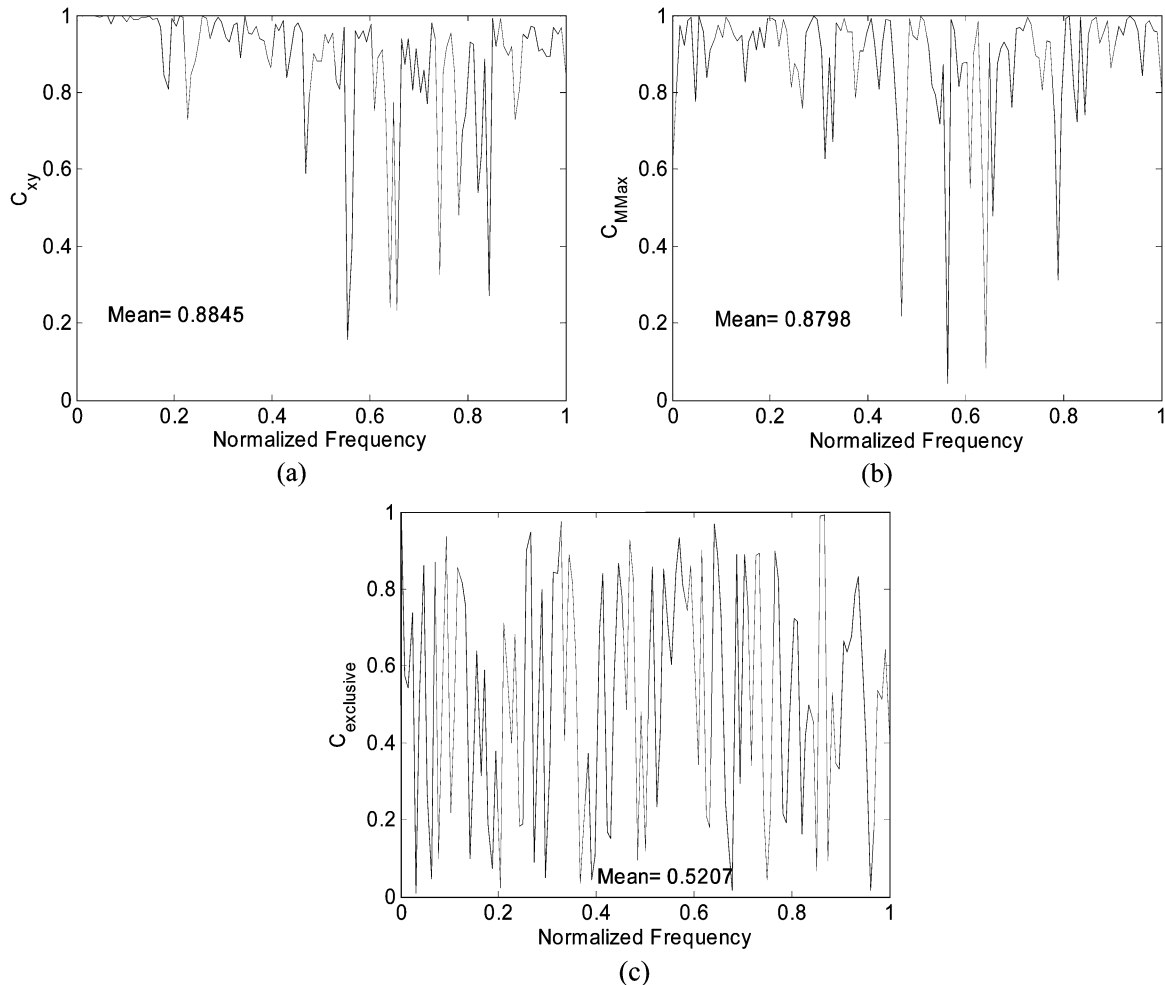
Fig. 4. Squared coherence for (a) $M = L = 512$. (b) $M = 0.5L$ with MMax tap-selection. (c) $M = 0.5L$ with exclusive tap-selection.

B. Interchannel Decorrelation Using Tap-selection

In order to examine the effect of tap-selection on interchannel coherence in SAEC, we first employ the squared coherence function

$$C_{\mathbf{x}_1\mathbf{x}_2}(f) = \frac{|P_{\mathbf{x}_1\mathbf{x}_2}(f)|^2}{P_{\mathbf{x}_1\mathbf{x}_1}(f)P_{\mathbf{x}_2\mathbf{x}_2}(f)} \tag{14}$$

where $P_{\mathbf{x}_1\mathbf{x}_2}(f)$ is the cross power spectrum between the two channels, and f is the normalized frequency.

We consider two tap-input vectors of length $L = 512$ formed from a zero mean, unit variance WGN source convolved with two highly correlated impulse responses $\mathbf{g}_1$ and $\mathbf{g}_2$ each of length 1024, where $\mathbf{g}_1$ is formed using the method of images [19]. In this example, $\mathbf{g}_2$ is then generated using the following relation:

$$\mathbf{g}_2 = \gamma\mathbf{g}_1 + (1 - \gamma)\mathbf{b} \tag{15}$$

where $\mathbf{b}$ is an independent WGN sequence also with zero mean. We have used $\gamma = 0.9$, giving a correlation coefficient of 0.904, to reflect the high interchannel correlation found in practice.

The highly correlated tap-input vectors give rise to a squared coherence close to one across most of the frequency band as shown in Fig. 4(a). In the case shown in Fig. 4(b), taps are selected according to the MMax selection criterion with $M = 0.5L$. It can be seen clearly that MMax tap-selection does not

provide any significant decorrelation. This is because the MMax criterion selects nearly identical tap-indices in both filters for updating, due to the high coherence between the two channel tap-input vectors. This does not achieve our desired effect of decorrelating the signals.

We now, therefore, propose an exclusive tap-selection criterion such that selection of the same tap-index in both channels is not permitted. A simple example of such an exclusive case with $M = 0.5L$ (but not the technique used in our proposed algorithm) is to select the taps corresponding to the M largest magnitude tap-inputs in the first channel and the exclusive set of taps in the second channel. Fig. 4(c) shows the squared coherence plot of such a case. We see that the mean interchannel coherence is significantly reduced from 0.88 to 0.52 and we will use this to illustrate and develop further study of tap-selection in Section IV.

The exclusive tap-selection can be seen as a method for improving the conditioning of the input autocorrelation matrix by considering the case where $\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$ are highly correlated Gaussian inputs. Defining $\mathbf{x}(n) = [\mathbf{x}_1^T(n) \quad \mathbf{x}_2^T(n)]^T$ and $E\{\cdot\}$ as the mathematical expectation operator, the autocorrelation matrix can be expressed as

$$\begin{aligned} \mathbf{R}_{\mathbf{xx}} &= E\{\mathbf{x}(n)\mathbf{x}^T(n)\} \\ &= \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix}. \end{aligned} \tag{16}$$
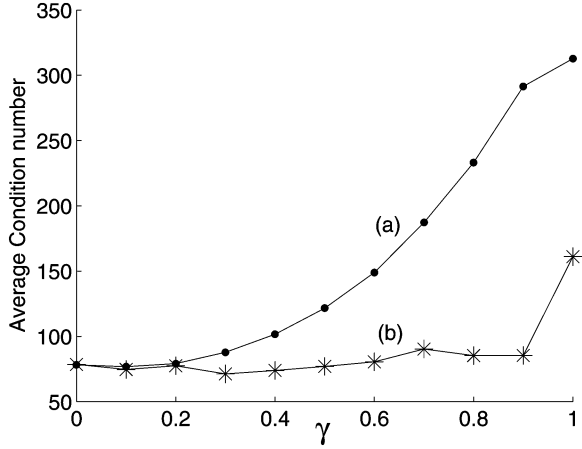
Fig. 5. Effect of exclusive tap-selection on mean condition number for WGN sequence. (a) Without tap-selection. (b) With exclusive tap-selection.

After exclusive tap-selection, the resulting sparse vectors $\widetilde{\mathbf{x}}_1(n) = \mathbf{Q}_1(n)\mathbf{x}_1(n)$ and $\widetilde{\mathbf{x}}_2(n) = \mathbf{Q}_2(n)\mathbf{x}_2(n)$ give rise to $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$. The diagonals and some off-diagonal elements of $\mathbf{R}_{12}$ and $\mathbf{R}_{21}$ are zero. This improves on the conditioning of $\mathbf{R}_{\mathbf{xx}}$ and in the limit where $\widetilde{\mathbf{x}}_1$ and $\widetilde{\mathbf{x}}_2$ are perfectly uncorrelated and white, the autocorrelation matrix is a diagonal matrix $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}} = \mathrm{diag}\{[\,\sigma_1^2 \ldots \sigma_1^2 \quad \sigma_2^2 \ldots \sigma_2^2\,]\}$ with a 2-norm condition number of $\|\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}\|\,\|\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}^{-1}\| = (\max(\sigma_1^2,\sigma_2^2)/\min(\sigma_1^2,\sigma_2^2))$, where σ_j^2 is the jth channel subselected tap-input variance.

Fig. 5 shows the variation of mean condition number of the autocorrelation matrices $\mathbf{R}_{\mathbf{xx}}$ and $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$ as a function of γ. Both the autocorrelation matrices are formed from $\mathbf{x}_1$ and $\mathbf{x}_2$ generated by convolving a WGN sequence with $\mathbf{g}_1$ and $\mathbf{g}_2$ governed by (15) with the additional exclusive tap-selection criterion imposed when generating $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$. For each case of γ, the average 2-norm condition number for 50 trials is computed and plotted as shown in Fig. 5(a) and (b) for $\mathbf{R}_{\mathbf{xx}}$ and $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$, respectively. We see that as γ is reduced, $\mathbf{x}_1$ and $\mathbf{x}_2$ become less correlated and, hence, a reduction of mean condition number for both $\mathbf{R}_{\mathbf{xx}}$ and $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$ is exhibited. In addition, for each case of γ, $\mathbf{R}_{\widetilde{\mathbf{x}}\widetilde{\mathbf{x}}}$ has a lower mean condition number than $\mathbf{R}_{\mathbf{xx}}$ and, hence, exclusive tap-selection gives rise to a better conditioned autocorrelation matrix which in turn allows us to address the misalignment problem discussed in Section II-B.

IV. EXCLUSIVE MAXIMUM TAP-SELECTION

A. Formulation

It has been shown in Section III that exclusive tap-selection can improve the conditioning of the adaptive filtering in SAEC. We wish to develop an adaptive filtering scheme which makes use of this concept without degrading convergence due to partial adaptation. We now, therefore, formulate the joint optimization problem of maximizing the MMax criterion, determined by $\mathcal{M}$, and minimizing the interchannel coherence under the control of tap-selection. This is done using two variables: magnitude weighting, w_m, to describe the "closeness" of the tap-selection to that of the MMax scheme, and coherence weighting,

$w_c = 1 - w_m$, to describe interchannel coherence between the subsampled tap-input vectors. A magnitude weighting of $w_m = 1$ corresponds to selecting coefficients based on the MMax tap-selection criterion only.

We begin by considering $^L C_M$ combinations of selecting $M = 0.5L$ taps from each channel's adaptive filter of length L. Let the combinations be indexed k, $l = 1, 2, \ldots,{}^L C_M$ giving tap-selection sets $\{\beta_k\}$ and $\{\beta_l\}$ for channel 1 and 2 respectively and define $\{\beta_{kl}(n)\}$ as the combined two channel tap-selection set. Let $\widetilde{\mathbf{x}}_k$ be defined as the subselected input vector using tap-selection set $\{\beta_k\}$. For the structure shown in Fig. 1, we define, at each time iteration n, $\mathbf{A}(n)$ and $\mathbf{C}(n)$ as square matrices with elements

$$a_{kl}(n) = \left\|\,|\widetilde{\mathbf{x}}_k(n)| + |\widetilde{\mathbf{x}}_l(n)|\,\right\|_1, \tag{17}$$

$$c_{kl}(n) = \left\langle \frac{|P_{\widetilde{\mathbf{x}}_k \widetilde{\mathbf{x}}_l}(f)|^2}{P_{\widetilde{\mathbf{x}}_k \widetilde{\mathbf{x}}_k}(f) P_{\widetilde{\mathbf{x}}_l \widetilde{\mathbf{x}}_l}(f)} \right\rangle \tag{18}$$

respectively, such that $a_{kl}(n)$ denotes the absolute sum of the selected tap-inputs in a particular tap-selection set $\beta_{kl}(n)$, and $c_{kl}(n)$ is the squared coherence, with $\langle \cdot \rangle$ indicating averaging over frequency, of the two tap-input vectors with $L - M$ unselected inputs in each channel set to zero.

Since the elements in matrix $\mathbf{A}(n)$ contain magnitude sums which are required to be maximized, an integer cost is first associated with each of the elements $a_{kl}(n)$ such that the *least cost* is allocated to the element having the *largest magnitude* in $\mathbf{A}(n)$. We now denote this new magnitude cost matrix as $\underline{\mathbf{A}}(n)$. In a similar manner, each element in $\mathbf{C}(n)$ will be allocated an integer cost such that element corresponding to the *minimum coherence* is allocated the *least cost*. We denote this new coherence cost matrix as $\underline{\mathbf{C}}(n)$. Hence, matrices $\underline{\mathbf{A}}(n)$ and $\underline{\mathbf{C}}(n)$ now contain integer cost values depending on the magnitude sum and interchannel coherence. A total cost matrix $\mathbf{V}(n)$ is then given by

$$\mathbf{V}(n) = w_m \underline{\mathbf{A}}(n) + w_c \underline{\mathbf{C}}(n). \tag{19}$$

We define $\{\beta_{\min}(n)\} = \{\beta_{k_{\min}, l_{\min}}(n)\}$ as the tap-selection set having minimum cost in matrix $\mathbf{V}(n)$ and search for $\{\beta_{\min}(n)\}$ such that

$$k_{\min}, l_{\min} = \arg \min_{k,l}(\mathbf{V}(n)) \quad k, l = 1, 2, \ldots,{}^L C_M. \tag{20}$$

For small L and letting $\widehat{\mathbf{h}}(n) = \begin{bmatrix} \widehat{\mathbf{h}}_1^T(n) & \widehat{\mathbf{h}}_2^T(n) \end{bmatrix}^T$ and $\mathbf{x}(n) = \begin{bmatrix} \mathbf{x}_1^T(n) & \mathbf{x}_2^T(n) \end{bmatrix}^T$, $\mathbf{V}(n)$ can be searched exhaustively such that the tap-selection set $\beta_{\min}(n)$ can then be incorporated into NLMS adaptation as

$$\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mathbf{Q}(n)\frac{\mu \mathbf{x}(n)e(n)}{\|\mathbf{x}(n)\|^2 + \delta} \tag{21}$$

with $\mathbf{Q}(n) = \mathrm{diag}\{[\,\mathbf{q}_1^T(n) \quad \mathbf{q}_2^T(n)\,]\}$ being the two-channel selection matrix such that, at each time iteration n, element u of $\mathbf{q}_1(n)$ and element v of $\mathbf{q}_2(n)$ are defined for $u, v = 1, 2, \ldots, L$ as

$$\{q_{1,u}, q_{2,v}\} = \begin{cases} 1, & \text{if } u, v \in \{\beta_{\min}\} \\ 0, & \text{otherwise} \end{cases}.$$
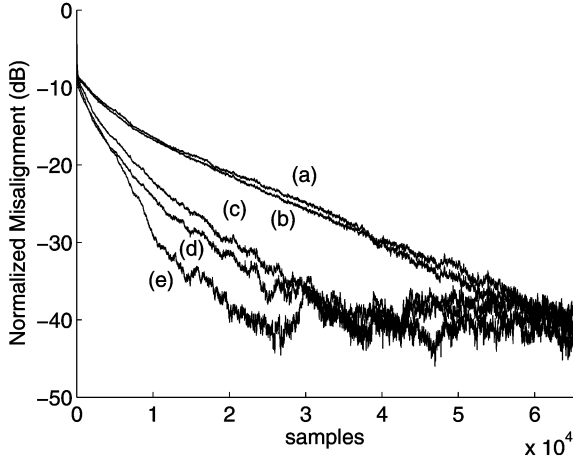
Fig. 6. Normalized misalignment for (a) $w_m = 1$. (b) NLMS. (c) $w_m = 0.9$. (d) 0.7. (e) 0.1. $L = 6$, $M = 3$, $\mu = 0.6$, $\gamma = 0.9$, SNR = 40 dB.

Fig. 6 shows simulation results for the normalized misalignment with different values of magnitude weighting ($w_m = 0.1$, 0.7, 0.9, 1.0). In this example, the input is a zero mean WGN sequence with adaptive filters having six taps per channel and for every iteration, three taps are updated ($L = 6$, $M = 3$). The relationship between impulse responses $\mathbf{g}_1$ and $\mathbf{g}_2$ with lengths $L_T = 12$ is again determined by (15) with $\gamma = 0.9$. The impulse responses $\mathbf{h}_1$ and $\mathbf{h}_2$ are taken from a WGN sequence and are of lengths $L_R = 6$. This choice of L_T and L_R allows us to study the adaptive filters which uniquely determine the unknown system while minimizing the misalignment caused by undermodeling. The normalized misalignment for only one of the two channels is plotted for each case of w_m for reasons of clarity. Uncorrelated measurement noise is added to $d(n)$ such that a signal-to-noise ratio (SNR) of 40 dB is achieved.

The simulation result shows that $w_m = 1$ coincides with MMax–NLMS where performance is close to that of the fully updated NLMS as expected. The highest convergence rate can be seen when $w_m = 0.1$ ($w_c = 0.9$), where there is a high weighting given to the minimization of interchannel coherence. Upon further investigation, it was found that for $w_m = 0.1$, all the tap-selection sets satisfy the exclusive criterion across all time iterations, such that combinations k and l contain no tap-indices in common, i.e,

$$\beta_k(n) \cap \beta_l(n) = \{\phi\}, \quad \forall n \tag{22}$$

where $\{\phi\}$ is a null set. Therefore, we can redefine our optimization problem in the simpler form of a search such that (22) is satisfied while maximizing $\mathcal{M}(n)$ at each iteration.

B. Efficient Realization

The exhaustive search of $\mathbf{V}(n)$ for the optimum exclusive maximum tap-selection is computationally expensive for adaptive filters of higher orders. We now, therefore, propose an efficient alternative to the exhaustive search. In the following, we shall temporarily drop the dependence on n for brevity.

Let us define, at each time iteration n, the interchannel tap-input magnitude difference vector

$$\mathbf{p} = |\mathbf{x}_1| - |\mathbf{x}_2| \tag{23}$$

and

$$\breve{\mathbf{p}} = [\breve{p}_1, \ldots, \breve{p}_L]^T, \quad \breve{p}_1 > \breve{p}_2 > \cdots > \breve{p}_L \tag{24}$$

as $\mathbf{p}$ sorted in descending order. Let $\breve{x}_{1,i}$ and $\breve{x}_{2,i}$ denote the ith tap-input samples of channels 1 and 2, ordered according to the sorting of $\breve{\mathbf{p}}$ such that $\breve{p}_i = |\breve{x}_{1,i}| - |\breve{x}_{2,i}|$, $i = 1, 2, \ldots, L$. In this two-channel case, $\mathcal{M}$ is defined as

$$\mathcal{M} = \frac{\|\mathbf{Q}\mathbf{x}\|^2}{\|\mathbf{x}\|^2} \tag{25}$$

with $\mathbf{Q} = \mathrm{diag}\,\{[\,\mathbf{q}_1^T \quad \mathbf{q}_2^T\,]\}$ and $\mathbf{x} = [\,\mathbf{x}_1^T \quad \mathbf{x}_2^T\,]^T$. Utilizing the robustness of the NLMS algorithm to MMax tap-selection for $0.5L \le M < L$ as discussed in Section III, we consider $M = 0.5L$.

As will be illustrated, the tap-selection set that maximizes $\mathcal{M}$ jointly for both channels contains the M largest elements of $\mathbf{p}$ from channel 1 and the M smallest elements of $\mathbf{p}$ from channel 2, i.e.,

$$\{\breve{x}_{1,1}, \ldots, \breve{x}_{1,M}, \breve{x}_{2,M+1}, \ldots, \breve{x}_{2,L}\}. \tag{26}$$

Hence, at each iteration, element u of $\mathbf{q}_1$ and element v of $\mathbf{q}_2$ are defined for $u, v = 1, 2, \ldots, L$ where

$$q_{1,u} = \begin{cases} 1, & p_u \in \{M\,\text{maxima of }\mathbf{p}\} \\ 0, & \text{otherwise} \end{cases}$$
$$q_{2,v} = \begin{cases} 1, & p_v \in \{M\,\text{minima of }\mathbf{p}\} \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

To verify this, we consider whether the absolute sum given by $\sum_{i=1}^{M} |\breve{x}_{1,i}| + \sum_{i=M+1}^{L} |\breve{x}_{2,i}|$ is greater than ${}^L C_M - 1$ other absolute sums of exclusive tap-inputs combinations. We start by testing whether

$$\sum_{i=1}^{M} |\breve{x}_{1,i}| + \sum_{i=M+1}^{L} |\breve{x}_{2,i}| > \sum_{i=M+1}^{L} |\breve{x}_{1,i}| + \sum_{i=1}^{M} |\breve{x}_{2,i}| \tag{28}$$

holds. Simplifying (28), we obtain

$$\sum_{i=1}^{M} [|\breve{x}_{1,i}| - |\breve{x}_{2,i}|] > \sum_{i=M+1}^{L} [|\breve{x}_{1,i}| - |\breve{x}_{2,i}|]$$
$$\sum_{i=1}^{M} \breve{p}_i > \sum_{i=M+1}^{L} \breve{p}_i, \tag{29}$$

which is valid from the definition of $\breve{\mathbf{p}}$. We next consider the ${}^L C_M - 2$ other possible cases. Suppose, for example, we select tap-indices in the set $\{\breve{x}_{1,2i}, \breve{x}_{2,2i-1}\}$ where $i = 1, 2, \ldots, M$ for which we must now test whether

$$\sum_{i=1}^{M} |\breve{x}_{1,i}| + \sum_{i=M+1}^{L} |\breve{x}_{2,i}| > \sum_{i=1}^{M} |\breve{x}_{1,2i}| + \sum_{i=1}^{M} |\breve{x}_{2,2i-1}| \tag{30}$$

holds. Rewriting (30), we obtain

$$\sum_{i=1}^{M} |\breve{x}_{1,i}| - \sum_{i=1}^{M} |\breve{x}_{2,2i-1}| > \sum_{i=1}^{M} |\breve{x}_{1,2i}| - \sum_{i=M+1}^{L} |\breve{x}_{2,i}|$$

and, hence

$$\sum_{i=1}^{M/2} |\breve{x}_{1,2i}| + \sum_{i=\varphi}^{L} |\breve{x}_{2,2i-1}| + \sum_{i=1}^{M/2} \breve{p}_{2i-1}$$
$$> \sum_{i=1}^{M/2} \breve{p}_{2i+M} + \sum_{i=1}^{M/2} |\breve{x}_{1,2i}| + \sum_{i=\varphi}^{L} |\breve{x}_{2,2i-1}|$$
$$\sum_{i=1}^{M/2} \breve{p}_{2i-1} > \sum_{i=1}^{M/2} \breve{p}_{2i+M} \qquad (31)$$

where $\varphi = 1 + M/2$. Since $M > 0$, (31) is valid from the definition of $\breve{\mathbf{p}}(n)$. Similar analysis can then be used to verify the remaining cases.

As an illustration, consider an SAEC system with channels $j = 1, 2$, adaptive filters each of length $L = 4$ and tap-input vectors $\mathbf{x}_j(n) = [x_{j,1} \quad x_{j,2} \quad x_{j,3} \quad x_{j,4}]^T$. The vector $\mathbf{p}(n)$ may then be expressed as

$$\begin{bmatrix} p_1(n) \\ p_2(n) \\ p_3(n) \\ p_4(n) \end{bmatrix} = \begin{bmatrix} |x_{1,1}(n)| \\ |x_{1,2}(n)| \\ |x_{1,3}(n)| \\ |x_{1,4}(n)| \end{bmatrix} - \begin{bmatrix} |x_{2,1}(n)| \\ |x_{2,2}(n)| \\ |x_{2,3}(n)| \\ |x_{2,4}(n)| \end{bmatrix}. \qquad (32)$$

Consider the example case $p_3 > p_2 > p_1 > p_4$, for a particular time instant. Since $p_3 + p_2 > \cdots > p_1 + p_4$, it can be shown that

$$|x_{1,3}| + |x_{1,2}| + |x_{2,1}| + |x_{2,4}|$$
$$> \cdots > |x_{1,1}| + |x_{1,4}| + |x_{2,2}| + |x_{2,3}| \qquad (33)$$

where $\ldots$ refers to all other pair-wise combinations of p_i, $i = 1$, 2, 3, 4. Thus, the tap-selection corresponding to inputs $x_{1,3}$, $x_{1,2}$, $x_{2,1}$ and $x_{2,4}$ maximizes $\mathcal{M}(n)$ with the minimum coherence constraint satisfied by the exclusivity of the tap-selection.

In this way, the exclusive maximum (XM) tap-selection criterion efficiently selects the best exclusive sets of taps where best here is defined as nearest to MMax jointly for both channels. This is achieved by maximizing the $\mathcal{M}(n)$ measure computed using the taps from both channels. Because of the exclusivity constraint, neither channel in general attains a tap-selection as good as MMax and some degradation in convergence performance is, therefore, to be expected. Nevertheless, our results indicate that such degradation is small compared to the improvement in convergence due to the decorrelating property of XM tap-selection.

We note that the XM tap-selection criterion as described above will result in a selected tap-input vector with lower power than for the MMax criterion for each channel due to the exclusivity constraint. It is to be expected, therefore, that the effect of noise may be relatively more significant in the proposed scheme compared to the MMax scheme. However, simulation results such as shown in Figs. 11 and 12 indicate that any such effects are insignificant compared to the improvements obtained due to the decorrelating properties of the proposed tap-selection.

As a final summary, we note that it is irrelevant to consider other tap-selection sets since they have smaller magnitude sum. This approach has allowed us to eliminate $^{L}C_M \times^{L} C_M - 1$ possible combinations, thus allowing efficient implementation

of the exclusive maximum tap selection which we have denoted XM. We will develop in Section V such efficient practical schemes for use with NLMS, AP, and RLS adaptation.

V. EXCLUSIVE MAXIMUM ADAPTIVE FILTERS

As has been shown in Section III, XM tap-selection can improve the conditioning of $\mathbf{R_{xx}}$ and, hence, improved convergence is expected. The effect of tap-selection for the AP and RLS cases on the autocorrelation matrix will be seen to be similar to that which occurs in the NLMS case shown in Section III-B. The XM approach relies on the existence of a unique solution for the adaptive filter coefficients which is the case for $L < L_T$ as discussed in Section II. As will be shown through simulations in Section VI, XM tap-selection in combination with the nonlinear (NL) preprocessor leads to better conditioning than the use of the NL-preprocessor alone. This combination of XM and NL approaches, which we refer to as XMNL, is highly effective for the cases we have studied and, therefore, we focus on this combined structure for our later experiments. Fig. 7 shows the schematic diagram of the proposed XMNL-based SAEC structure.

A. XM-NLMS Algorithm

The XM tap-selection technique may be incorporated into the NLMS by selecting taps corresponding to the $M = 0.5L$ largest elements of the input magnitude difference vector $\mathbf{p}(n)$ in the first channel and the M smallest elements of $\mathbf{p}(n)$ in the second channel as shown in Table I. Tap-indices are then updated using (10). Simulation results for the XM-NLMS algorithm are shown in Fig. 10(c) and described in Section VI-B.

B. XMNL-NLMS Algorithm

The nonlinear (NL) preprocessor [4] is one of the most effective methods of achieving signal decorrelation without affecting stereo perception by using α as the nonlinearity constant such that

$$\mathbf{x}_1'(n) = \mathbf{x}_1(n) + 0.5\alpha[\mathbf{x}_1(n) + |\mathbf{x}_1(n)|] \qquad (34)$$
$$\mathbf{x}_2'(n) = \mathbf{x}_2(n) + 0.5\alpha[\mathbf{x}_2(n) - |\mathbf{x}_2(n)|]. \qquad (35)$$

We refer to the use of the NL preprocessor with NLMS adaptation as NL–NLMS. Several workers [3], [9], [20] have proposed algorithms in combination with the NL processor so as to achieve low misalignment. In the same manner, we propose a combined algorithm employing the XM tap-selection, to improve the conditioning of the autocorrelation matrix, in addition to the NL preprocessor so as to improve the convergence rate obtained from the use of the NL preprocessor alone. The XMNL–NLMS algorithm is summarized in Table II.

C. XMNL–AP Algorithm

The affine projection algorithm [21] incorporates multiple projections by concatenating past input vectors from time iteration n to time iteration $n - K + 1$, where K is defined as the projection order. We first distinguish the concatenated *subselected* tap-input vector and the concatenated *full* tap-input vector by letting $\widetilde{\mathbf{x}}'(n) = \mathbf{Q}(n)\mathbf{x}'(n)$ be the subselected tap-input vector where $\mathbf{x}'(n) = [\mathbf{x}_1'^T(n) \quad \mathbf{x}_2'^T(n)]^T$ and
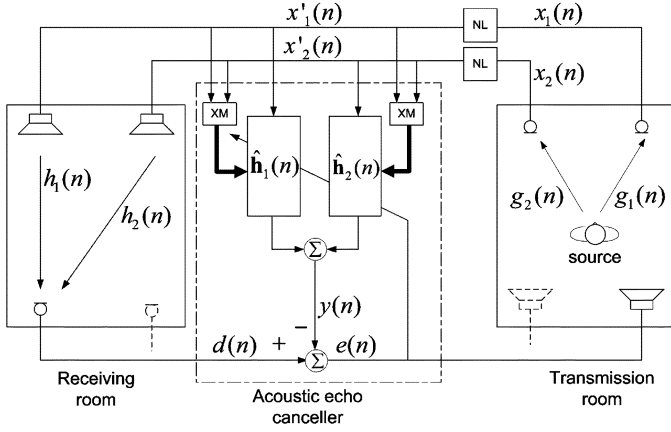
Fig. 7. Schematic diagram of XMNL preprocessor in stereophonic echo canceller. Bold arrows indicate tap-selection control.

TABLE I
XMNL TAP-SELECTION

$$
\begin{aligned}
M &= 0.5L \\
\mathbf{x}'_1(n) &= \mathbf{x}_1(n) + 0.5\alpha\big[\mathbf{x}_1(n) + |\mathbf{x}_1(n)|\big] \\
\mathbf{x}'_2(n) &= \mathbf{x}_2(n) + 0.5\alpha\big[\mathbf{x}_2(n) - |\mathbf{x}_2(n)|\big] \\
\mathbf{x}'(n) &= \big[\mathbf{x}'^T_1(n)\ \mathbf{x}'^T_2(n)\big]^T \\
\widehat{\mathbf{h}}(n) &= \big[\widehat{\mathbf{h}}^T_1(n)\ \widehat{\mathbf{h}}^T_2(n)\big]^T \\
\mathbf{p}(n) &= \big|\mathbf{x}'_1(n)\big| - \big|\mathbf{x}'_2(n)\big| \\
\mathbf{Q}(n) &= \mathrm{diag}\big\{[\mathbf{q}^T_1(n)\ \mathbf{q}^T_2(n)]\big\} \\
q_{1,u} &= \begin{cases} 1 & p_u \in \{M \text{ maxima of } \mathbf{p}\} \\ 0 & \text{otherwise} \end{cases} \\
q_{2,v} &= \begin{cases} 1 & p_v \in \{M \text{ minima of } \mathbf{p}\} \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

$\mathbf{Q}(n) = \mathrm{diag}\big\{[\mathbf{q}^T_1(n)\ \ \mathbf{q}^T_2(n)]\big\}$. The subselected and full tap-input matrix are then denoted, respectively, as

$$
\widetilde{\mathbf{X}}'(n) = \big[\widetilde{\mathbf{x}}'(n)\ \ \widetilde{\mathbf{x}}'(n-1)\ \ \ldots\ \ \widetilde{\mathbf{x}}'(n-K+1)\big]^T \tag{36}
$$

$$
\mathbf{X}'(n) = \big[\mathbf{x}'(n)\ \ \mathbf{x}'(n-1)\ \ \ldots\ \ \mathbf{x}'(n-K+1)\big]^T . \tag{37}
$$

Thus, the tap-update equation for the XMNL–AP algorithm is given as

$$
\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mu\widetilde{\mathbf{X}}'^T(n)[\mathbf{X}'(n)\mathbf{X}'^T(n) + \delta\mathbf{I}]^{-1}\mathbf{e}(n) \tag{38}
$$

where $\mathbf{e}(n) = [e(n)\ e(n-1)\ \ldots\ e(n-K+1)]^T$ and δ is the regularization parameter while $\mathbf{I}$ is a $K \times K$ identity matrix. It can be seen from (38) that for projection order $K = 1$, XMNL–AP is equivalent to XMNL–NLMS.

Note that even though the matrix $\widetilde{\mathbf{X}}'(n)$ is formed by subselecting the tap-input vector, XMNL–AP in general cannot be classified as a partial-update algorithm. This is because the elements in the column vector $\widetilde{\mathbf{X}}'^T(n)[\mathbf{X}'(n)\mathbf{X}'^T(n)]^{-1}\mathbf{e}(n)$ form

TABLE II
XMNL–NLMS

$$
\begin{aligned}
y(n) &= \widehat{\mathbf{h}}^T(n)\mathbf{x}'(n) \\
e(n) &= d(n) - y(n) \\
\widehat{\mathbf{h}}(n+1) &= \widehat{\mathbf{h}}(n) + \mathbf{Q}(n)\frac{\mu\mathbf{x}'(n)e(n)}{||\mathbf{x}'(n)||^2 + \delta}
\end{aligned}
$$

TABLE III
XMNL–AP

$$
\begin{aligned}
\mathbf{X}'(n) &= \big[\mathbf{x}'(n)\ \mathbf{x}'(n-1)\ \ldots\ \mathbf{x}'(n-K+1)\big]^T \\
\widetilde{\mathbf{x}}'(n) &= \mathbf{Q}(n)\mathbf{x}'(n) \\
\widetilde{\mathbf{X}}'(n) &= \big[\widetilde{\mathbf{x}}'(n)\ \widetilde{\mathbf{x}}'(n-1)\ \ldots\ \widetilde{\mathbf{x}}'(n-K+1)\big]^T \\
\mathbf{d}(n) &= \big[d(n)\ d(n-1)\ \ldots\ d(n-K+1)\big]^T \\
\mathbf{y}(n) &= \mathbf{X}'(n)\widehat{\mathbf{h}}(n) \\
\mathbf{e}(n) &= \mathbf{d}(n) - \mathbf{y}(n) \\
\widehat{\mathbf{h}}(n+1) &= \widehat{\mathbf{h}}(n) + \mu\widetilde{\mathbf{X}}'^T(n)\big[\mathbf{X}'(n)\mathbf{X}'^T(n) + \delta\mathbf{I}\big]^{-1}\mathbf{e}(n)
\end{aligned}
$$

a full vector and, therefore, every element of $\widehat{\mathbf{h}}(n)$ will be updated at each iteration. Special cases may occur if there exist any null rows in the matrix $\widetilde{\mathbf{X}}'^T(n)$ resulting in a partial adaptation. Such a situation may arise if there are several consecutive small values of $\mathbf{p}(n)$ such that the "inactive" tap-indices in each channel propagate consistently through $\widetilde{\mathbf{X}}'(n)$ from iteration n to $n - K + 1$. The XMNL–AP algorithm is given in Table III.

D. XMNL–RLS Algorithm

The RLS algorithm is well known to be relatively insensitive to the eigenvalue-spread of the input signal compared to NLMS-based algorithms [21], [22]. Consequently, the rate of convergence of RLS is higher than that of NLMS for speech inputs. However, the algorithm does not address the nonuniqueness problem inherent in SAEC and, hence, a poor rate of convergence is expected when the RLS algorithm is applied directly.

The tap-update equation for RLS is given as

$$
\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \mathbf{k}(n)e(n) \tag{39}
$$

where the Kalman gain and *a priori* error are denoted as $\mathbf{k}(n)$ and $e(n)$, respectively. We note that direct extension of the XM tap-selection approach achieved by sorting the magnitude difference of $\mathbf{k}(n)$ will not achieve the desired convergence. This is because the Kalman gain depends on previous values of the time-averaged inverse correlation matrix Ψ^{-1}, defined in (5), given by

$$
\mathbf{k}(n) = \frac{\lambda^{-1}\Psi^{-1}(n-1)\mathbf{x}(n)}{1 + \lambda^{-1}\mathbf{x}^T(n)\Psi^{-1}(n-1)\mathbf{x}(n)}. \tag{40}
$$

Our approach will be to improve the condition of $\Psi(n)$ by introducing sparseness. This can be achieved by replacing $\mathbf{x}'(n)$

in the Kalman gain calculation and Ricatti update with a sub-sampled input vector $\widetilde{\mathbf{x}}'(n) = \mathbf{Q}(n)\mathbf{x}'(n)$. This ensures that the subsampled input vectors propagate consistently through the memory of the algorithm.

To derive the XMNL–RLS algorithm, we rewrite (5) in terms of the subselected tap-input vector recursively as

$$\widetilde{\Psi}'(n) = \widetilde{\mathbf{X}}'(n)\Lambda(n)\widetilde{\mathbf{X}}'^{T}(n)$$
$$= \lambda\widetilde{\Psi}'(n-1) + \widetilde{\mathbf{x}}'(n)\widetilde{\mathbf{x}}'^{T}(n) \quad (41)$$

where

$$\widetilde{\mathbf{X}}'(n) = [\,\widetilde{\mathbf{x}}'(1) \quad \widetilde{\mathbf{x}}'(2) \quad \ldots \quad \widetilde{\mathbf{x}}'(n)\,]$$

and

$$\Lambda = \mathrm{diag}\{[\,\lambda^{n} \quad \lambda^{n-1} \quad \ldots \quad \lambda\,]\}.$$

Similarly, the cross-correlation vector in (6) may be expressed recursively as

$$\widetilde{\Theta}'(n) = \widetilde{\mathbf{X}}'(n)\Lambda(n)\mathbf{d}(n)$$
$$= \lambda\widetilde{\Theta}'(n-1) + \widetilde{\mathbf{x}}'(n)d(n) \quad (42)$$

where

$$\mathbf{d}(n) = [\,d(1) \quad d(2) \quad \ldots \quad d(n)\,]^{T}.$$

Using the matrix inversion lemma and following the approach of [21] gives

$$\widetilde{\Psi}'^{-1}(n) = \frac{1}{\lambda}\left[\widetilde{\Psi}'^{-1}(n-1) - \mathbf{k}(n)\widetilde{\mathbf{x}}'^{T}(n)\widetilde{\Psi}'^{-1}(n-1)\right] \quad (43)$$

where the modified Kalman gain is given by

$$\widetilde{\mathbf{k}}(n) = \frac{\lambda^{-1}\widetilde{\Psi}'^{-1}(n-1)\widetilde{\mathbf{x}}'(n)}{1 + \lambda^{-1}\widetilde{\mathbf{x}}'^{T}(n)\widetilde{\Psi}'^{-1}(n-1)\widetilde{\mathbf{x}}'(n)} \quad (44)$$

such that $\widetilde{\mathbf{k}}(n) = \left[\widetilde{\mathbf{k}}_1^T(n) \quad \widetilde{\mathbf{k}}_2^T(n)\right]^T$. Defining $e(n)$ as the *a priori* error, the XMNL–RLS tap-update equation is then given by

$$\widehat{\mathbf{h}}(n+1) = \widehat{\mathbf{h}}(n) + \widetilde{\mathbf{k}}(n)e(n). \quad (45)$$

Similar to XMNL–AP, the XMNL–RLS algorithm in general updates all the taps at each iteration since the modified Kalman gain vector $\widetilde{\mathbf{k}}(n)$ is a full column vector except in cases where there exist any null rows in $\widetilde{\Psi}'^{-1}(n)$. The XMNL–RLS algorithm is summarized in Table IV.

E. Computational Complexity

We define the complexity as the total number of multiplications and comparisons per sample period for each channel. Similar to the MMax–NLMS algorithm, the XMNL-based algorithms employ the SORTLINE procedure [23] which requires at most $2 + 2\log_2 L$ comparisons. Thus, the XMNL-NLMS has the same complexity as MMax–NLMS, for $M = 0.5L$, which requires at most $1.5L + 3 + 2\log_2 L$ operations per sample period per channel.

The complexity of AP using the generalized Levinson algorithm is $2LK + 7K^2$ multiplies per sample period [20]. The XMNL–AP algorithm requires an additional $2 + 2\log_2 L$ sorting operations in each channel for $\widetilde{\mathbf{x}}'(n)$. However, due to a reduction in multiplications required when computing

TABLE IV
XMNL–RLS

$\widetilde{\mathbf{k}}(n)$	$= \left[\widetilde{\mathbf{k}}_1^T(n) \ \widetilde{\mathbf{k}}_2^T(n)\right]^T$
$\widetilde{\mathbf{x}}'(n)$	$= \mathbf{Q}(n)\mathbf{x}'(n)$
$\widetilde{\mathbf{k}}(n)$	$= \frac{\widetilde{\Psi}'^{-1}(n-1)\widetilde{\mathbf{x}}'(n)}{\lambda + \widetilde{\mathbf{x}}'^{T}(n)\widetilde{\Psi}'^{-1}(n-1)\widetilde{\mathbf{x}}'(n)}$
$y(n)$	$= \widehat{\mathbf{h}}^T(n)\mathbf{x}'(n)$
$e(n)$	$= d(n) - y(n)$
$\widehat{\mathbf{h}}(n+1)$	$= \widehat{\mathbf{h}}(n) + \widetilde{\mathbf{k}}(n)e(n)$
$\widetilde{\Psi}'^{-1}(n)$	$= \frac{1}{\lambda}\left[\widetilde{\Psi}'^{-1}(n-1) - \widetilde{\mathbf{k}}(n)\widetilde{\mathbf{x}}'^{T}(n)\widetilde{\Psi}'^{-1}(n-1)\right]$

$\widetilde{\mathbf{X}}'^{T}(n)[\mathbf{X}'(n)\mathbf{X}'^{T}(n)]^{-1}$, the complexity for XMNL–AP is $1.5LK + 7K^2 + 2 + 2\log_2 L$ operations per sample period per channel.

The number of multiplications required for the RLS algorithm is $4L^2 + 3L + 2$ per adaptive filter where an additional L multiplications are required for the tap-updates. Due to the sub-selection of input vector $\widetilde{\mathbf{x}}'(n)$, the number of multiplications required for computing $\widetilde{\Psi}'(n)$ for the XMNL–RLS is $1.5L^2 + 1$ while $L(L+0.5)$ multiplications are required for computing the Kalman gain. Hence, the number of operations required for the XMNL–RLS is at most $2.5L(L+1) + 3 + 2\log_2 L$ per sample period per channel.

Figs. 8 and 9 show the variation of complexity with L for XMNL–NLMS, XMNL–AP, and XMNL–RLS algorithm. The projection order for AP-based algorithms is $K = 2$. Although complexity reduction is not the main aim of this work, it can be seen that the XM selective-tap techniques nevertheless bring significant computational savings.

VI. SIMULATION RESULTS

A. Experimental Setup

In all our simulations, impulse responses $\mathbf{g}_1$, $\mathbf{g}_2$, $\mathbf{h}_1$, and $\mathbf{h}_2$ are generated using the method of images [19]. Two microphones are placed 1 m apart in the center of both the transmission and receiving rooms each of dimension $3 \times 4 \times 5$ m. The source is then positioned 1 m away from each microphone in the transmission room. Tap-input vectors $\mathbf{x}'_1(n)$ and $\mathbf{x}'_2(n)$ are obtained by convolving the source with two impulse responses $\mathbf{g}_1$ and $\mathbf{g}_2$ and then applying the nonlinear preprocessor defined in (34) and (35). The desired response $d(n)$ in the receiving room is obtained by summing $\mathbf{h}_1^T\mathbf{x}'_1(n)$ and $\mathbf{h}_2^T\mathbf{x}'_2(n)$. For clarity, the normalized misalignment of only one channel is plotted in each experiment.

B. NLMS Simulations

We examine the performance of XM tap-selection and the NL preprocessor in combination with NLMS adaptation. In
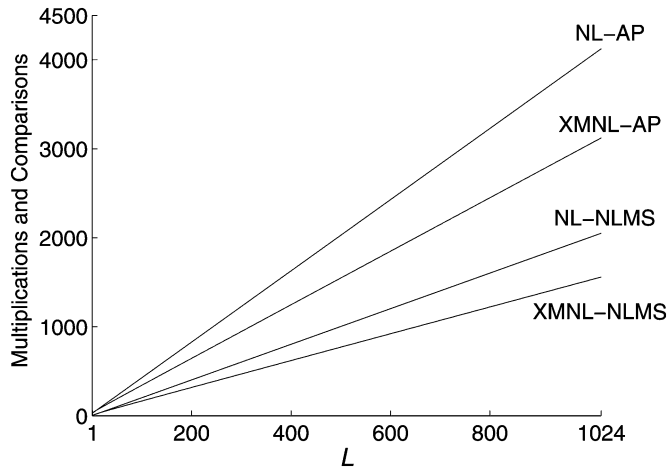
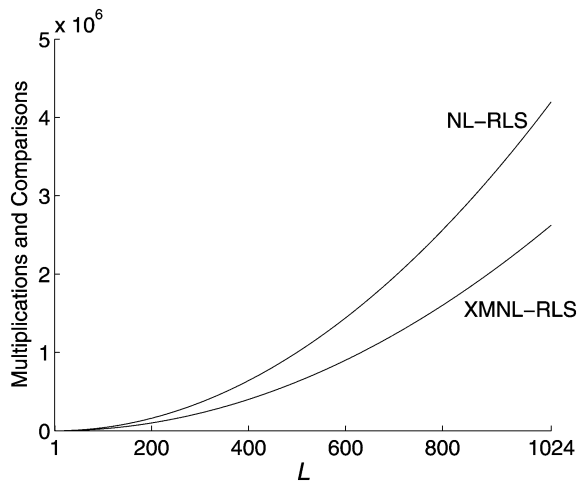Fig. 8.   Computational complexity of NLMS and AP-based algorithms.

Fig. 9.   Computational complexity of RLS-based algorithms.

Fig. 10.   Normalized misalignment for WGN sequence (a) NLMS. (b) NL–NLMS. (c) XM-NLMS. (d) XMNL–NLMS [$L = 256$, $L_T = 1200$, $L_R = 256$, $M = 128$, $f_s = 8$ kHz, $\mu = 0.1$, $\alpha = 0.5$, and SNR = 30 dB].

Fig. 11.   (a) Speech signal and normalized misalignment for (b) NL–AP and (c) XMNL–AP [$L = 512$, $L_T = L_R = 1200$, $M = 256$, $f_s = 8$ kHz, $\mu_{\text{NL-AP}} = 0.5$, $\mu_{\text{XMNL-AP}} = 0.4$, $\alpha = 0.5$, $K = 3$, and SNR = 30 dB].

this experiment, the lengths of the adaptive filters are $L = 256$ while the lengths of the transmission and receiving rooms' impulse responses are $L_T = 1600$ and $L_R = 256$, respectively. Fig. 10 shows the normalized misalignment plot for (a) NLMS, (b) NL–NLMS, (c) XM-NLMS, and (d) XMNL–NLMS. A WGN input signal with a sampling frequency of $f_s = 8$ kHz is used with $M = 128$, and a step-size of $\mu = 0.4$ is chosen for each algorithm. A nonlinear distortion factor of $\alpha = 0.5$ is used [4] and WGN sequence is added to $d(n)$ such that an SNR of 30 dB is achieved. We see that NLMS has the slowest convergence. The convergence rate of XM-NLMS and NL–NLMS increases significantly due to the XM and NL preprocessors, respectively. The XMNL–NLMS algorithm shows even further improvement of approximately 3 to 4 dB compared to NL–NLMS due to the additional improvement in conditioning caused by XM tap-selection. Alternatively, XMNL–NLMS could achieve the same rate of convergence as NL–NLMS but with a lower value of α [18], hence reducing nonlinear distortion.

C. AP Simulations

The performance of the XMNL–AP algorithm is compared with that of the AP algorithm in combination with NL pre-processor (NL–AP) for a recorded speech signal. The impulse
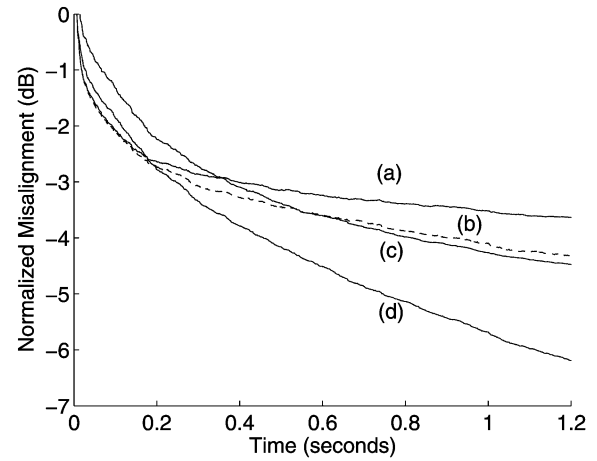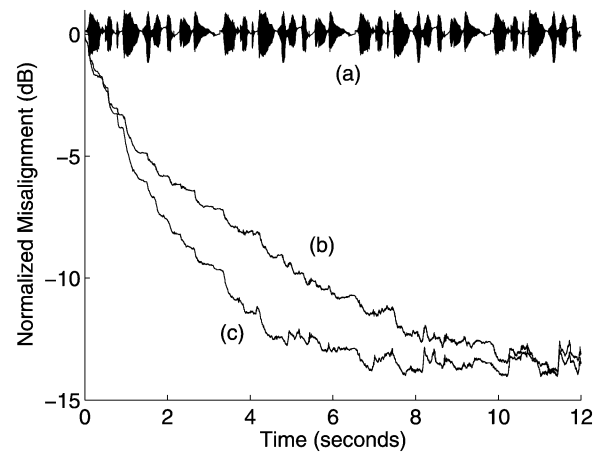
responses are chosen to be of length $L_T = L_R = 1200$, adaptive filters of length $L = 512$ and $M = 256$ are used. We have used a sampling frequency of $f_s = 8$ kHz and an additive WGN is added to the desired signal such that an SNR of 30 dB is achieved. The adaptive step-size for each algorithm is chosen such that they achieve approximately the same final normalized misalignment. A nonlinearity constant of $\alpha = 0.5$ and affine projection order $K = 3$ are used.

We see from Fig. 11 that the rate of convergence of XMNL–AP is significantly higher than that of the NL–AP resulting in approximately 4 dB of improved normalized misalignment. This is again due to the additional improvement in conditioning caused by XM tap-selection. For the arbitrary choice of $\mu_{\text{NL-AP}} = 0.5$, it was found that $\mu_{\text{XMNL-AP}} = 0.4$ gives approximately the same final normalized misalignment.

D. RLS Simulations

In Fig. 12, we compare the performance of XMNL–RLS with that of the RLS incorporating the NL preprocessor (NL–RLS) [4]. We have used $L_T = L_R = 800$, $L = 256$, $M = 128$, and a speech input sequence with sampling frequency of $f_s = $
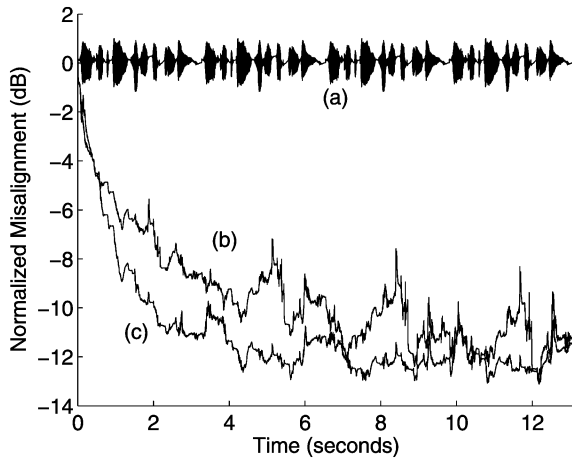
Fig. 12. (a) Speech signal and normalized misalignment for (b) NL–RLS and (c) XMNL–RLS [$L = 256$, $L_T = L_R = 800$, $M = 128$, $f_s = 8$ kHz, $\lambda_{\text{XMNL−RLS}} = 0.99961$, $\lambda_{\text{NL−RLS}} = 0.99957$, $\alpha = 0.5$, and SNR $= 30$ dB].

8 kHz. As before, WGN sequence is added to the desired signal such that an SNR of 30 dB is achieved. A forgetting factor of $\lambda_{\text{XMNL−RLS}} = 1 − [1/(10L)] = 0.99961$ [24] is used for XMNL–RLS while for NL–RLS, $\lambda_{\text{NL−RLS}} = 0.99957$ is used such that both algorithms achieve approximately the same final normalized misalignment.

As shown in Fig. 12, the XMNL–RLS algorithm converged within the range of 6 to 8 s giving a significant improvement in convergence rate corresponding to approximately 3 to 4 dB in normalized misalignment over NL–RLS.

VII. CONCLUSION

In this paper, we have introduced a novel approach to improve convergence in SAEC employing a tap-selection technique. The "closeness" of MMax tap-selection to the full tap-input vector has been quantified by introducing $\mathcal{M}$ which was then used as an optimization parameter in the development of the proposed XM tap-selection technique. It has been shown that the exclusive tap-selection criterion reduces the interchannel coherence of the tap-input vectors and improves the conditioning of the autocorrelation matrix which consequently reduces the misalignment problem. The efficient XM tap-selection technique has been developed as an optimization of the MMax criterion subject to an exclusivity constraint between the tap-selection sets of the two channels. This XM tap-selection has been applied to the NLMS, AP, and RLS algorithms with and without nonlinear preprocessing. Simulation results have shown a significant improvement in the range of 3 to 4 dB during convergence compared with algorithms that use the NL–preprocessor alone. Although complexity reduction is not the main aim of this work, it has been seen that XM selective tap updating nevertheless brings significant computational savings.

REFERENCES

[1] S. Emura, Y. Haneda, and S. Makino, "Enhanced frequency-domain adaptive algorithm for stereo echo cancellation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 2, 2002, pp. 1901–1904.

[2] P. Eneroth, S. L. Gay, T. Gänsler, and J. Benesty, "A real-time implementation of a stereophonic acoustic echo canceller," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 4, pp. 513–523, Jul. 2001.

[3] K. Mayyas, "Stereophonic acoustic echo cancellation using lattice orthogonalization," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 7, pp. 517–525, Oct. 2002.

[4] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 156–165, Mar. 1998.

[5] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. 2001: Springer-Verlag.

[6] S. Shimauchi and S. Makino, "Stereo projection echo canceller with true echo path estimation," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, vol. 5, 1995, pp. 3059–3062.

[7] A. Gilloire and V. Turbin, "Using auditory properties to improve the behavior of stereophonic acoustic echo cancellers," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, vol. 6, 1998, pp. 3681–3684.

[8] T. Tangsangiumvisai, J. A. Chambers, and A. G. Constantinides, "Time-varying allpass filters using spectral-shaped noise for signal decorrelation in stereophonic acoustic echo cancellation," in *Proc. Int. Conf. Digital Signal Processing*, 2002, pp. 1273–1276.

[9] J. Benesty, D. R. Morgan, J. L. Hall, and M. M. Sondhi, "Stereophonic acoustic echo cancellation using nonlinear transformations and comb filtering," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, vol. 6, 1998, pp. 3673–3676.

[10] T. Hoya, Y. Loke, J. Chambers, and P. A. Naylor, "Application of the leaky extended LMS algorithm in stereophonic acoustic echo cancellation," *Signal Process.*, vol. 64, pp. 87–91, 1998.

[11] N. T. Forsyth, J. A. Chambers, and P. A. Naylor, "An alternating fixed-point algorithm for stereophonic acoustic echo cancellation," in *Proc. Inst. Elect. Eng., Vision, Image Signal Process.*, vol. 149, 2002, pp. 1–9.

[12] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits Syst. II*, vol. 44, no. 3, pp. 209–216, Mar. 1997.

[13] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1421–1424, May 1999.

[14] P. A. Naylor and W. Sherliker, "A short-sort M-max NLMS partial update adaptive filter with applications to echo cancellation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, 2003, pp. 373–376.

[15] K. Dogancay and O. Tanrikulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II*, no. 8, pp. 762–769, Aug. 2001.

[16] P. A. Naylor and A. W. H. Khong, "Affine projection and recursive least squares adaptive filters employing partial updates," in *Proc. 38th Asilomar Conf. Signals, Systems, Computers*, vol. 1, Nov. 2004, pp. 950–954.

[17] A. W. H. Khong and P. A. Naylor, "The use of partial update scheme to reduce inter-channel coherence in adaptive stereophonic acoustic echo cancellation," in *Proc. Int. Workshop Acoustic Echo and Noise Control*, 2003, pp. 59–62.

[18] ——, "Reducing inter-channel coherence in stereophonic acoustic echo cancellation using partial update adaptive filters," in *Proc. Eur. Signal Process. Conf.*, 2004, pp. 405–408.

[19] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.

[20] S. L. Gay and J. Benesty, *Acoustic Signal Processing for Telecommunication*. Norwell, MA: Kluwer, 2001.

[21] S. Haykin, *Adaptive Filter Theory*, 4th ed, ser. Information and System Science. Upper Saddle River, NJ: Prentice-Hall, 2002.

[22] M. G. Bellanger, *Adaptive Digital Filters*. New York: Marcel Dekker, 2001.

[23] I. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 795–804, Jun. 1989.

[24] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic echo control: An application of very-high-order adaptive filter," *IEEE Signal Process. Mag.*, vol. 16, no. 4, pp. 42–69, Jul. 1999.
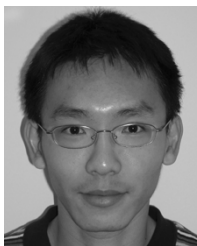
Andy Khong received the B.Eng. degree in electrical and electronics engineering from the Nanyang Technological University, Singapore, in 2002.

Since 2002, he has been with Imperial College London as a Research Postgraduate. His research interests are mainly in the area of adaptive algorithms both in time and frequency domains, as well as their analysis with applications to single- and multichannel acoustic echo cancellation and speech enhancement. He is currently supported by the Engineering and Physical Sciences Research Council (EPSRC), U.K., working on speech enhancement.

Patrick Naylor (M'89) received the B.Eng. degree in electronics and electrical engineering from the University of Sheffield, Sheffield, U.K., in 1986 and the Ph.D. degree from Imperial College, London, U.K., in 1990.

Since 1989, he has been a member of academic staff in the Communications and Signal Processing research group at Imperial College, where he is also Director of Postgraduate Studies. His research interests are in the areas of speech and audio signal processing, and he has worked in particular on adaptive signal processing for acoustic echo control, speaker identification, multichannel speech enhancement, and speech production modeling. In addition to his academic research, he enjoys several fruitful links with industry in the U.K., USA, and in mainland Europe.

Dr. Naylor is an Associate Editor of IEEE SIGNAL PROCESSING LETTERS and a member of the IEEE Signal Processing Society Technical Committee on Audio and Electroacoustics.