A LOW COMPLEXITY FAST CONVERGING PARTIAL UPDATE ADAPTIVE ALGORITHM EMPLOYING VARIABLE STEP-SIZE FOR ACOUSTIC ECHO CANCELLATION

Andy W. H. Khong [1], *Woon-Seng Gan* [2], *Patrick A. Naylor* [1], *Mike Brookes* [1]

[1]Department of Electrical and Electronic Engineering, Imperial College London, UK
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

ABSTRACT

Partial update adaptive algorithms have been proposed as a means of reducing complexity for adaptive filtering. The MMax tap-selection is one of the most popular tap-selection algorithms. It is well known that the performance of such partial update algorithm reduces with reducing number of filter coefficients selected for adaptation. We propose a low complexity and fast converging adaptive algorithm that exploits the MMax tap-selection. We achieve fast convergence with low complexity by deriving a variable step-size for the MMax normalized least-mean-square (MMax-NLMS) algorithm using its mean square deviation. Simulation results verify that the proposed algorithm achieves higher rate of convergence with lower computational complexity compared to the NLMS algorithm.

Index Terms— acoustic echo cancellation, partial update adaptive filtering, variable step-size, adaptive algorithms

1. INTRODUCTION

The profound interest in adaptive filtering with finite impulse response (FIR) arises due to its extensive application in signal processing. One of the most popular adaptive algorithms is the normalized least-mean-square (NLMS) algorithm [1][2] which has been applied to many applications including acoustic echo cancellation (AEC). To achieve effective echo cancellation, a replica of the echo is generated by means of modelling the Loudspeaker-Room-Microphone (LRM) system using an adaptive filter as shown in Fig. 1. Implementation of an acoustic echo canceller poses great challenges due to (i) the highly time-varying nature of the impulse response [3] and (ii) the long duration of the LRM system, which can require several thousands of filter coefficients for accurate modelling. Much recent research has aimed to develop fast converging algorithms that are necessary to track time variations in the LRM system. In addition, a typical room impulse response in the region of 50 to 300 ms requires an FIR adaptive filter with 400 to 2400 taps at 8 kHz sampling frequency. Since the NLMS algorithm requires $\mathcal{O}(2L)$ multiply-accumulate (MAC) operations per sampling period, it is very desirable to reduce the computational workload of the processor, especially for the real-time implementation of AEC algorithms in portable devices where power budget is a constraint. As a result, a class of partial update adaptive filtering algorithms has been proposed that share the characteristic of executing tap update operations on only a subset of the filter coefficients at each iteration.

Partial update adaptive algorithms differ in the criteria used for selecting filter coefficients to update at each iteration. The Periodic-LMS and Sequential-LMS algorithms [4] employ tap-selection schemes that are independent of the input data. In the Periodic-LMS algorithm, reduction in computation is achieved at each time iteration by updating a subset of filter coefficients periodically whereas the Sequential-LMS algorithm employs the instantaneous gradient estimate at each time iteration by decimation in the tap space. In contrast, data dependent tap-selection criteria are employed in later algorithms including Max-LMS [5] and MMax-NLMS [6][7]. Block-based and transform domain algorithms which generalized MMax-NLMS [8] have also been proposed. More recently, the MMax tap-selection criterion has been extended to a class of selective-tap algorithms including the MMax affine projection (MMax-AP) and MMax recursive least squares (MMax-RLS) algorithms [9]. The performance of these MMax-based adaptive algorithms for time-varying LRM systems has also been analyzed [9] and extended for the multichannel case [10]. It has been shown that the performance of MMax tap-selection is better than Periodic- and Sequential-LMS algorithms [11].

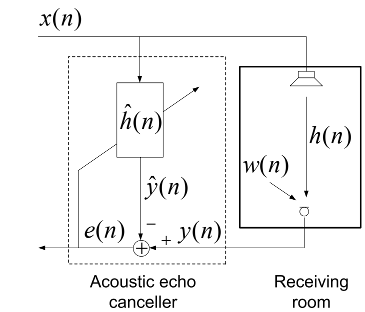

Fig. 1. Acoustic echo cancellation.

It is found that as the number of filter coefficients updated per iteration in a partial update adaptive filter is reduced, the computational complexity is also reduced but at the expense of some loss in performance. Hence the goal of the designers of such algorithms is to find ways to reduce the number of coefficients updated per iteration in a manner which degrades algorithm performance as little as possible. The aim of this paper is to propose a low complexity, fast converging adaptive algorithm for AEC. It has been shown in [9] that the convergence performance of MMax-NLMS is dependent on the step-size when identifying a LRM system. This motivates us to jointly utilize the low complexity of MMax tap-selection and the improvement in convergence performance brought about by a variable step-size. We begin by first analyzing the mean-square deviation of MMax-NLMS and deriving a variable step-size in order to increase its rate of convergence. We show through simulation examples that the proposed variable step-size MMax-NLMS (MMax-NLMS$_{\rm vss}$) algorithm achieves *higher rate of convergence* with *lower computational complexity* compared to NLMS for both white Gaussian noise (WGN) and speech inputs.

2. THE MMAX-NLMS ALGORITHM

Figure 1 shows an echo canceller in which, at the n^{th} iteration, $y(n) = \mathbf{x}^T(n)\mathbf{h}(n)$ where $\mathbf{x}(n) = [x(n), \ldots, x(n-L+1)]^T$ is the tap-input vector while the unknown LRM system $\mathbf{h}(n) = [h_0(n), \ldots, h_{L-1}(n)]^T$ is of length L. An adaptive filter $\widehat{\mathbf{h}}(n) = [\widehat{h}_0(n), \ldots, \widehat{h}_{L-1}(n)]^T$, which we assume [3] to be of equal length to the unknown system $\mathbf{h}(n)$, is used to estimate $\mathbf{h}(n)$ by adaptively minimizing the *a priori* error signal $e(n)$ using $\widehat{y}(n)$ defined by

$$e(n) = \mathbf{x}^T(n)\mathbf{h}(n) - \widehat{y}(n) + w(n), \quad (1)$$

$$\widehat{y}(n) = \mathbf{x}^T(n)\widehat{\mathbf{h}}(n-1) \quad (2)$$

with $w(n)$ being the measurement noise.

In the MMax-NLMS algorithm [6], only those taps corresponding to the M largest magnitude tap-inputs are selected for updating at each iteration with $1 \leq M \leq L$. Defining the subselected tap-input vector

$$\widetilde{\mathbf{x}}(n) = \mathbf{Q}(n)\mathbf{x}(n), \quad (3)$$

where $\mathbf{Q}(n) = \text{diag}\{\mathbf{q}(n)\}$ is a $L \times L$ tap selection matrix and $\mathbf{q}(n) = [q_0(n), \ldots, q_{L-1}(n)]^T$, element $q_i(n)$ for $i = 0, 1, \ldots, L-1$ is given by,

$$q_i(n) = \begin{cases} 1 & |x(n-i)| \in \{M \text{ maxima of } |\mathbf{x}(n)|\} \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

where $|\mathbf{x}(n)| = \big[|x(n)|, \ldots, |x(n-L+1)|\big]^T$. Defining $\|.\|^2$ as the squared l_2-norm, the MMax-NLMS tap-update equation is then

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\mu\mathbf{Q}(n)\mathbf{x}(n)e(n)}{\|\mathbf{x}(n)\|^2 + \delta}, \quad (5)$$

where δ is the regularization parameter. Defining $\mathbf{I}_{L\times L}$ as the $L \times L$ identity matrix, we note that if $\mathbf{Q}(n) = \mathbf{I}_{L\times L}$, i.e., with $M = L$, the update equation in (5) is equivalent to the NLMS algorithm. Similar to the NLMS algorithm, the step-size μ in (5) controls the ability of MMax-NLMS to track the unknown system which is reflected by its rate of convergence. To select the M maxima of $|\mathbf{x}(n)|$ in (4), MMax-NLMS employs the SORTLINE algorithm [12] which requires $2\log_2 L$ sorting operations per iteration. The computational complexity in terms of multiplications for MMax-NLMS is $\mathcal{O}(L+M)$ compared to $\mathcal{O}(2L)$ for NLMS.

As explained in Section 1, the performance of MMax-NLMS normally reduces with the number of filter coefficients updated per iteration. This tradeoff between complexity and convergence can be illustrated by first defining

$$\eta(n) = \big\|\mathbf{h}(n) - \widehat{\mathbf{h}}(n)\big\|^2 / \big\|\mathbf{h}(n)\big\|^2 \quad (6)$$

as the normalized misalignment. Figure 2 shows the variation in convergence performance of MMax-NLMS with M for the case of $L = 2048$ and $\mu = 0.3$ using a white Gaussian noise (WGN) input. For this illustrative example, WGN $w(n)$ is added to achieve a signal-to-noise ratio (SNR) of 20 dB. It can be seen that the rate of convergence reduces with reducing M as expected. The dependency of the asymptotic performance and rate of convergence on M for MMax-NLMS has been analyzed in [9].

3. MEAN SQUARE DEVIATION OF MMAX-NLMS

It has been shown in [9] that the convergence performance of MMax-NLMS is dependent on the step-size μ when identifying a LRM system. Since our aim is to reduce the degradation of convergence performance due to partial updating of the filter coefficients, we propose

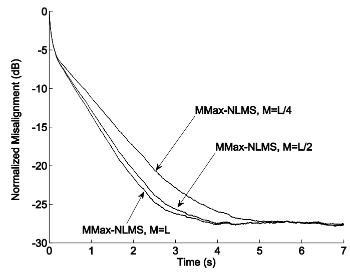

Fig. 2. MMax-NLMS: Variation of convergence rate with number of filter coefficients selected for adaptation M for $L = 2048$, $\mu = 0.3$, SNR=20 dB.

to derive an adaptive step-size for MMax-NLMS. A similar approach was adopted in [13] for NLMS by analyzing the mean square deviation (MSD) of NLMS. Similar to the analysis of MMax-NLMS under time-varying unknown system conditions as shown in [9], we assume that the MMax-NLMS algorithm is able to track the unknown system. The MSD of MMax-NLMS can be obtained by first defining the system deviation as

$$\boldsymbol{\epsilon}(n) = \mathbf{h}(n) - \widehat{\mathbf{h}}(n), \quad (7)$$

$$\boldsymbol{\epsilon}(n-1) = \mathbf{h}(n) - \widehat{\mathbf{h}}(n-1). \quad (8)$$

Subtracting (8) from (7) and using (5), we obtain

$$\boldsymbol{\epsilon}(n) = \boldsymbol{\epsilon}(n-1) - \frac{\mu\mathbf{Q}(n)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{x}(n) + \delta}. \quad (9)$$

Defining $\mathcal{E}\{\cdot\}$ as the expectation operator and taking the mean square of (9), the MSD of MMax-NLMS can be expressed iteratively as

$$\begin{aligned}\mathcal{E}\big\{\|\boldsymbol{\epsilon}(n)\|^2\big\} &= \mathcal{E}\big\{\boldsymbol{\epsilon}^T(n)\boldsymbol{\epsilon}(n)\big\} \\ &= \mathcal{E}\big\{\|\boldsymbol{\epsilon}(n-1)\|^2\big\} - \mathcal{E}\big\{\psi(\mu)\big\}, \quad (10)\end{aligned}$$

where

$$\mathcal{E}\{\psi(\mu)\} = \mathcal{E}\left\{\frac{2\mu\widetilde{\mathbf{x}}^T(n)\boldsymbol{\epsilon}(n-1)e(n)}{\|\mathbf{x}(n)\|^2} - \frac{\mu^2\|\widetilde{\mathbf{x}}(n)\|^2 e^2(n)}{\Big[\|\mathbf{x}(n)\|^2\Big]^2}\right\} \quad (11)$$

and similar to [13], we assume that the effect of the regularization term δ on the MSD is small. The subselected tap-input vector $\widetilde{\mathbf{x}}(n)$ is defined by (3). As can be seen from (10), in order to increase the rate of convergence for the MMax-NLMS algorithm, we choose step-size μ such that $\mathcal{E}\{\psi(\mu)\}$ is maximized.

4. THE PROPOSED MMAX-NLMS$_{\text{VSS}}$ ALGORITHM

Following the approach of [13], we differentiate (11) with respect to μ. Setting the result to zero, we obtain,

$$\mathcal{E}\left\{\frac{\mu(n)e(n)\|\widetilde{\mathbf{x}}(n)\|^2 e(n)}{\big[\|\mathbf{x}(n)\|^2\big]^2}\right\} = \mathcal{E}\left\{\boldsymbol{\epsilon}^T(n-1)\widetilde{\mathbf{x}}(n)\Big[\|\mathbf{x}(n)\|^2\Big]^{-1} e(n)\right\}$$

giving the variable step-size

$$\mu(n) = \mu_{\max} \times \frac{\boldsymbol{\epsilon}^T(n-1)\widetilde{\mathbf{x}}(n)\Big[\|\mathbf{x}(n)\|^2\Big]^{-1}\mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1)\|\mathbf{x}(n)\|^2}{\|\widetilde{\mathbf{x}}(n)\|^2\boldsymbol{\epsilon}^T(n-1)\mathbf{x}(n)\Big[\|\mathbf{x}(n)\|^2\Big]^{-1}\mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1) + \sigma_w^2\mathcal{M}(n)},$$

where $0 < \mu_{\max} \leq 1$ limits the maximum of $\mu(n)$ and we have defined [9]

$$\mathcal{M}(n) = \frac{\|\widetilde{\mathbf{x}}(n)\|^2}{\|\mathbf{x}(n)\|^2} \quad (12)$$

as the ratio between energies of the subselected tap-input vector $\widetilde{\mathbf{x}}(n)$ and the complete tap-input vector $\mathbf{x}(n)$, while $\sigma_w^2 = \mathcal{E}\{w^2(n)\}$. To simplify the numerator of $\mu(n)$ further, we utilize the relationship $\widetilde{\mathbf{x}}(n)\mathbf{x}^T(n) = \widetilde{\mathbf{x}}(n)\widetilde{\mathbf{x}}^T(n)$ giving

$$\mu(n) = \mu_{\max} \times \frac{\boldsymbol{\epsilon}^T(n-1)\widetilde{\mathbf{x}}(n)\left[\|\mathbf{x}(n)\|^2\right]^{-1}\widetilde{\mathbf{x}}^T(n)\boldsymbol{\epsilon}(n-1)\|\mathbf{x}(n)\|^2}{\|\widetilde{\mathbf{x}}(n)\|^2\boldsymbol{\epsilon}^T(n-1)\mathbf{x}(n)\left[\|\mathbf{x}(n)\|^2\right]^{-1}\mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1) + \sigma_w^2\mathcal{M}(n)}.$$

We can now simplify $\mu(n)$ further by letting

$$\widetilde{\mathbf{p}}(n) = \widetilde{\mathbf{x}}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}\widetilde{\mathbf{x}}^T(n)\boldsymbol{\epsilon}(n-1), \quad (13)$$

$$\mathbf{p}(n) = \mathbf{x}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}\mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1), \quad (14)$$

from which we can then show that

$$\|\widetilde{\mathbf{p}}(n)\|^2 = \mathcal{M}(n)\boldsymbol{\epsilon}^T(n-1)\widetilde{\mathbf{x}}(n)\left[\|\mathbf{x}(n)\|^2\right]^{-1}\widetilde{\mathbf{x}}^T(n)\boldsymbol{\epsilon}(n-1),$$

$$\|\mathbf{p}(n)\|^2 = \boldsymbol{\epsilon}^T(n-1)\mathbf{x}(n)\left[\|\mathbf{x}(n)\|^2\right]^{-1}\mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1).$$

Following the approach in [13], and defining $0 << \alpha < 1$ as the smoothing parameter, we can estimate $\widetilde{\mathbf{p}}(n)$ and $\mathbf{p}(n)$ iteratively by

$$\widetilde{\mathbf{p}}(n) = \alpha\widetilde{\mathbf{p}}(n-1) + (1-\alpha)\widetilde{\mathbf{x}}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}e_a(n), \quad (15)$$

$$\mathbf{p}(n) = \alpha\mathbf{p}(n-1) + (1-\alpha)\mathbf{x}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}e(n), \quad (16)$$

where we have used $e(n) = \mathbf{x}^T(n)\boldsymbol{\epsilon}(n-1)$ in (16) while the error $e_a(n)$ due to active filter coefficients $\widetilde{\mathbf{x}}(n)$ in (15) is given as

$$e_a(n) = \widetilde{\mathbf{x}}^T(n)\boldsymbol{\epsilon}(n-1) = \widetilde{\mathbf{x}}^T(n)[\mathbf{h}(n) - \widehat{\mathbf{h}}(n-1)]. \quad (17)$$

It is important to note that since $\widetilde{\mathbf{x}}^T(n)\mathbf{h}(n)$ is unknown, we need to approximate $e_a(n)$. Defining $\overline{\mathbf{Q}}(n) = \mathbf{I}_{L\times L} - \mathbf{Q}(n)$ as the tap-selection matrix which selects the inactive taps, we can express

$$e_i(n) = \left[\overline{\mathbf{Q}}(n)\mathbf{x}(n)\right]^T\boldsymbol{\epsilon}(n-1)$$

as the error contribution due to the inactive filter coefficients such that the total error $e(n) = e_a(n) + e_i(n)$. As explained in [9], for $0.5L \leq M < L$, the degradation in $\mathcal{M}(n)$ due to tap-selection is negligible. This is because, for M large enough, elements in $\overline{\mathbf{Q}}(n)\mathbf{x}(n)$ are small and hence the errors $e_i(n)$ are small, as is the general motivation for MMax tap-selection [7]. We can then approximate $e_a(n) \approx e(n)$ in (15) giving

$$\widetilde{\mathbf{p}}(n) \approx \alpha\widetilde{\mathbf{p}}(n-1) + (1-\alpha)\widetilde{\mathbf{x}}(n)\left[\mathbf{x}^T(n)\mathbf{x}(n)\right]^{-1}e(n). \quad (18)$$

Using (16) and (18), the variable step-size is then given as

$$\mu(n) = \mu_{\max}\frac{\|\widetilde{\mathbf{p}}(n)\|^2}{\mathcal{M}^2(n)\|\mathbf{p}(n)\|^2 + C} \quad (19)$$

where $C = \mathcal{M}^2(n)\sigma_w^2$. Since σ_w^2 is unknown, it is shown that we can approximate C by a small constant, typically 0.01 [13].

We note that the computation of (16) and (18) each requires M additions. In order to reduce computation even further, and since for M large enough the elements in $\overline{\mathbf{Q}}(n)\mathbf{x}(n)$ are small, we can approximate $\|\mathbf{p}(n)\|^2 \approx \|\widetilde{\mathbf{p}}(n)\|^2$ giving

$$\mu(n) \approx \mu_{\max}\frac{\|\widetilde{\mathbf{p}}(n)\|^2}{\mathcal{M}^2(n)\|\widetilde{\mathbf{p}}(n)\|^2 + C}. \quad (20)$$

Table 1. The MMax-NLMS$_{\rm vss}$ algorithm

$0 << \alpha < 1$, $\quad 0 < \mu_{\max} \leq 1$, $\quad C = 0.01$,	
$e(n)$	$= y(n) - \mathbf{x}^T(n)\widehat{\mathbf{h}}(n-1) + w(n)$
$\mathbf{Q}(n)$	$= \mathrm{diag}\{\mathbf{q}(n)\}$
$q_i(n)$	$= \begin{cases} 1 & \lvert x(n-i)\rvert \in \{M \text{ maxima of } \lvert\mathbf{x}(n)\rvert\} \\ 0 & \text{otherwise} \end{cases}$
$\widetilde{\mathbf{x}}(n)$	$= \mathbf{Q}(n)\mathbf{x}(n)$
$\widetilde{\mathbf{p}}(n)$	$= \alpha\widetilde{\mathbf{p}}(n-1) + (1-\alpha)\widetilde{\mathbf{x}}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}e(n)$
$\mathcal{M}(n)$	$= \|\widetilde{\mathbf{x}}(n)\|^2/\|\mathbf{x}(n)\|^2$
$\mu(n)$	$= \mu_{\max}\|\widetilde{\mathbf{p}}(n)\|^2/[\mathcal{M}^2(n)\|\widetilde{\mathbf{p}}(n)\|^2 + C]$
$\widehat{\mathbf{h}}(n)$	$= \widehat{\mathbf{h}}(n-1) + \mu(n)\dfrac{\mathbf{Q}(n)\mathbf{x}(n)e(n)}{\|\mathbf{x}(n)\|^2 + \delta}$

When $\mathbf{Q}(n) = \mathbf{I}_{L\times L}$, i.e., $M = L$, MMax-NLMS is equivalent to the NLMS algorithm and from (12), $\mathcal{M}(n) = 1$ and $\|\widetilde{\mathbf{p}}(n)\|^2 = \|\mathbf{p}(n)\|^2$. As a consequence, the variable step-size $\mu(n)$ in (20) is consistent with that presented in [13] for $M = L$. The proposed MMax-NLMS$_{\rm vss}$ is summarized in Table 1.

5. COMPUTATIONAL COMPLEXITY

We now discuss the computational complexity in terms of the number of multiplications required for the proposed MMax-NLMS$_{\rm vss}$ algorithm at each sample iteration. Computation of (18) and $\|\widetilde{\mathbf{p}}(n)\|^2$ for (20) require M multiplications each. The computation of $\|\mathbf{x}(n)\|^2$ and $\|\widetilde{\mathbf{x}}(n)\|^2$ for $\mathcal{M}(n)$ in (12) requires 2 multiplications and a division using recursive means [11]. More importantly, since the term $\widetilde{\mathbf{x}}(n)[\mathbf{x}^T(n)\mathbf{x}(n)]^{-1}e(n)$ is already computed in (18), no multiplications are now required for the update equation in (5). Hence, including the computation of $\mathbf{x}^T(n)\widehat{\mathbf{h}}(n-1)$ for $e(n)$, MMax-NLMS$_{\rm vss}$ requires $\mathcal{O}(L+2M)$ multiplications per sample period compared to $\mathcal{O}(2L)$ for NLMS. The number of multiplications required for MMax-NLMS$_{\rm vss}$ is thus less than NLMS when $M < L/2$. We note that although MMax-NLMS$_{\rm vss}$ requires an additional $2\log_2 L$ sorting operations per iteration using the SORTLINE algorithm [12], its complexity is still lower than NLMS. As with MMax-NLMS, we would expect the convergence performance for MMax-NLMS$_{\rm vss}$ to degrade with reducing M. However, we shall show through simulation results that any such degradation is offset by the improvement in convergence rate due to $\mu(n)$.

6. SIMULATION RESULTS

We demonstrate the performance of MMax-NLMS$_{\rm vss}$ in terms of the normalized misalignment $\eta(n)$ defined in (6) using both WGN and speech inputs. Impulse response $\mathbf{h}(n)$ is generated using the method of images [14] in a room of dimension $4\times 5\times 3$ m. The microphone and source positions are placed at coordinates $\{2.01, 2.5, 1.6\}$ and $\{2.1, 1.5, 1.6\}$ respectively. With a sampling rate of 8 kHz and a reverberation time of 256 ms, the length of the impulse response is $L = 2048$. Similar to [13], we have used $C = 0.01$, $\alpha = 0.95$ and we added WGN $w(n)$ to $y(n)$ in order to achieve an SNR of 20 dB. We have used $\mu_{\max} = 1$ for MMax-NLMS$_{\rm vss}$ while step-size μ for the NLMS algorithm is adjusted so as to achieve the same steady-state performance for all simulations.

We illustrate first the improvement in convergence rate due to the variable step-size $\mu(n)$ for the MMax-NLMS$_{\rm vss}$ algorithm. Figure 3 shows the improvement in convergence performance of MMax-NLMS$_{\rm vss}$ over MMax-NLMS for the cases of $M = 1024$ and 512. For each case, the proposed MMax-NLMS$_{\rm vss}$ algorithm achieves an

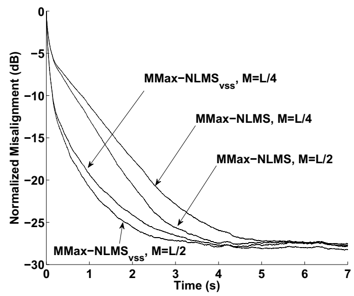

Fig. 3. Comparison between MMax-NLMS$_{\rm vss}$ and MMax-NLMS for $L = 2048$ and SNR$= 20$ dB using WGN input.

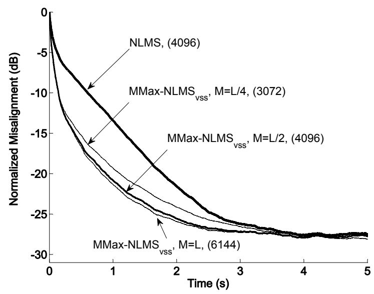

Fig. 4. WGN input: Comparison between convergence performance of MMax-NLMS$_{\rm vss}$ with NLMS for $L = 2048$ and SNR$= 20$ dB.

improvement of approximately 7 and 5.5 dB over MMax-NLMS in terms of normalized misalignment during initial convergence.

We next illustrate the tradeoff between computational complexity and convergence performance for the MMax-NLMS$_{\rm vss}$ algorithm. We also compare the performance of MMax-NLMS$_{\rm vss}$ and NLMS for a WGN input as shown in Fig. 4. The step-size of NLMS has been adjusted in order to achieve the same steady-state normalized misalignment. This corresponds to $\mu = 0.3$. The number of multiplications per sample iteration required for each case is depicted between braces in the figure. As can be seen from Fig. 4, for the same number of multiplications of 4096, the improvement of MMax-NLMS$_{\rm vss}$ in terms of normalized misalignment compared with NLMS is approximately 8 dB during initial convergence. More importantly, the proposed MMax-NLMS$_{\rm vss}$ algorithm outperforms NLMS even with lower complexity when $M = 512$. This improvement in normalized misalignment of 7 dB (together with a reduction of 25% in terms of multiplications) over NLMS is due to variable step-size for MMax-NLMS$_{\rm vss}$. The MMax-NLMS$_{\rm vss}$ achieves the same convergence performance as the NLMS$_{\rm vss}$ [13] when $M = L$.

The performance of MMax-NLMS$_{\rm vss}$ for a male speech input is depicted in Fig. 5. For this simulation, we have used $L = 2048$ and an SNR=20 dB as before. In order to illustrate the benefits of our proposed algorithm, we used $M = 512$ for both MMax-NLMS and MMax-NLMS$_{\rm vss}$. This gives a 25% savings in multiplications per iteration for MMax-NLMS$_{\rm vss}$ over NLMS. As can be seen, even with this computational savings, the proposed MMax-NLMS$_{\rm vss}$ algorithm achieves an improvement of 3.5 dB in terms of normalized misalignment over NLMS.

7. CONCLUSIONS

We have proposed a low complexity fast converging partial update MMax-NLMS algorithm by introducing a variable step-size during

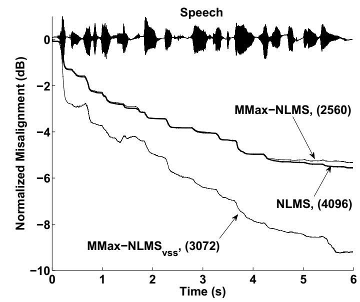

Fig. 5. Speech input: Comparison between convergence performance of MMax-NLMS$_{\rm vss}$ with NLMS for $L = 2048$, $M = 512$ and SNR$= 20$ dB.

adaptation. This is derived by analyzing the mean-square deviation of MMax-NLMS. In terms of convergence performance, the proposed MMax-NLMS$_{\rm vss}$ algorithm achieves approximately 7 and 3.5 dB improvement in normalized misalignment over NLMS for WGN and speech input respectively. More importantly, the proposed algorithm can achieve higher rate of convergence with lower computational complexity compared to NLMS.

8. REFERENCES

[1] B. Widrow, "Thinking about thinking: the discovery of the LMS algorithm," *IEEE Signal Processing Mag.*, vol. 22, no. 1, pp. 100–106, Jan. 2005.

[2] S. Haykin, *Adaptive Filter Theory*, 4th ed., ser. Information and System Science. Prentice Hall, 2002.

[3] E. Hänsler, "Hands-free telephones- joint control of echo cancellation and postfiltering," *Signal Processing*, vol. 80, no. 11, pp. 2295–2305, Nov. 2000.

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

[5] ——, "Analysis and implementation of the MAX-NLMS adaptive filter," in *Proc. Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, 1995, pp. 659–663.

[6] T. Aboulnasr and K. Mayyas, "Selective coefficient update of gradient-based adaptive algorithms," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, vol. 3, 1997, pp. 1929–1932.

[7] ——, "MSE analysis of the M-Max NLMS adaptive algorithm," in *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, vol. 3, 1998, pp. 1669–1672.

[8] K. Dogancay and O. Tanrikulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 762–769, Aug. 2001.

[9] A. W. H. Khong and P. A. Naylor, "Selective-tap adaptive filtering with performance analysis for identification of time-varying systems," *IEEE Trans. Audio Speech Language Processing*, vol. 15, no. 5, pp. 1681 – 1695, Jul. 2007.

[10] ——, "Stereophonic acoustic echo cancellation employing selective-tap adaptive algorithms," *IEEE Trans. Speech Audio Processing*, vol. 14, no. 3, pp. 785–796, Jul. 2006.

[11] A. W. H. Khong, "Adaptive algorithms employing tap selection for single channel and stereophonic acoustic echo cancellation," Ph.D. dissertation, Imperial College London, Feb. 2006.

[12] I. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Trans. Circuits Syst.*, vol. 36, no. 6, pp. 795–804, Jun. 1989.

[13] H.-C. Shin, A. Sayed, and W.-J. Song, "Variable step-size NLMS and affine projection algorithms," *IEEE Signal Processing Lett.*, vol. 11, no. 2, pp. 132–135, Feb. 2004.

[14] 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.