

# MEAN-SQUARED ANALYSIS OF THE PARTIAL-UPDATE NLMS ALGORITHM

Stefan Werner, Marcello L. R. de Campos, and Paulo S. R. Diniz

**Abstract** - In this paper, we present mean-squared convergence analysis for the partial-update normalized least-mean square (PU-NLMS) algorithm with closed-form expressions for the case of white input signals. The analysis uses order statistics and the formulas presented here are more accurate than the ones found in the literature for the PU-NLMS algorithm. Simulation results show excellent agreement with the results predicted by the analysis.

**Keywords:** Adaption algorithms, efficient algorithms, partial update, MSE analysis, order statistics, NLMS algorithm.

**Resumo** - Neste artigo apresentamos a análise da convergência na média quadrática do algoritmo least-mean-square normalizado com utilização parcial dos dados (PU-NLMS) para o caso de sinais de entrada brancos. A análise usa conceitos de estatística de variáveis ordenadas e as fórmulas apresentadas mostraram-se mais precisas que outros resultados encontrados na literatura para o algoritmo PU-NLMS. Resultados de simulações mostram ótima concordância com aqueles previstos pela teoria.

**Palavras-chave:** Algoritmos adaptativos, algoritmos eficientes, algoritmos com atualização parcial de dados, estatística de variáveis ordenadas, algoritmo LMS normalizado.

## 1. INTRODUCTION

When implementing an adaptive-filtering algorithm, the affordable number of coefficients that can be used will depend on the application in question, the adaptation algorithm, and the hardware chosen for implementation. With the choice of algorithms ranging from the simple least-mean square (LMS) algorithm to the more complex recursive least squares (RLS) algorithm, tradeoffs between performance criteria such as, e.g., computational complexity and convergence rate, have to be made. In certain applications, the use of the RLS algorithm is prohibitive due to the high computational complexity and in such cases we must resort to simpler algorithms. As an example, consider the acoustic echo cancellation application where the adaptive filter may require thousands of coefficients [1]. This large number of filter coefficients may impair even the implementation of low computational complexity algorithms, such as the normalized least-

mean square (NLMS) algorithm [1].

As an alternative, instead of reducing filter order, one may choose to update only part of the filter coefficient vector at each time instant. Such algorithms, referred to as partial-update (PU) algorithms, can reduce computational complexity while performing close to their full-update counterparts in terms of convergence rate and final mean-squared error (MSE). In the literature one can find several variants of the LMS and the NLMS algorithms with partial updates [2]–[10], as well as more computationally complex variants based on the affine projection algorithm [10, 11].

The objective of this paper is to analyze the partial-update NLMS (PU-NLMS) algorithm introduced in [10, 11]. The results from our analysis, which is based on order statistics, yield more accurate bounds on step size and on the prediction of excess MSE when compared to the results presented in [10, 11]. We also clarify the relationship between the PU-NLMS and M-Max NLMS [5, 6] algorithms, whereby we show that the M-Max NLMS algorithm uses an instantaneous estimate of the step size that achieves the fastest convergence in  $\alpha$ -MSE.

The organization of the paper is as follows. Section 2 reviews and discusses the PU-NLMS algorithm. Section 3 provides an analysis in the mean-squared sense that is novel for this algorithm and allows new insights to its behavior. In Section 4 we validate our analysis of the PU-NLMS algorithm and compare our results with those available in the literature. Conclusions are given in Section 5.

## 2. THE PU-NLMS ALGORITHM

This section reviews and discusses the partial-update NLMS (PU-NLMS) algorithm proposed in [10, 11].

The objective in PU adaptation is to derive an algorithm that only updates a fraction of coefficients of the adaptive filter in each iteration. Let us start by partitioning the input-signal vector  $\mathbf{x}(k) \in \mathbb{R}^N$  and the adaptive filter vector  $\mathbf{w}(k) \in \mathbb{R}^N$  into  $B$  blocks of  $L = N/B$  coefficients each,

$$\begin{aligned} \mathbf{x}(k) &= [x(k) \ x(k-1) \ \cdots \ x(k-N+1)]^T \\ &= [\mathbf{x}_1^T(k) \ \mathbf{x}_2^T(k) \ \cdots \ \mathbf{x}_B^T(k)]^T \end{aligned} \quad (1)$$

$$\begin{aligned} \mathbf{w}(k) &= [w_1(k) \ w_2(k) \ \cdots \ w_N(k)]^T \\ &= [\mathbf{w}_1^T(k) \ \mathbf{w}_2^T(k) \ \cdots \ \mathbf{w}_B^T(k)]^T \end{aligned} \quad (2)$$

Assuming a sequence of desired signals  $\{d(k)\}_{k=1}^{\infty}$ , we can write the sequence of output errors  $\{e(k)\}_{k=1}^{\infty}$  as

$$e(k) = d(k) - \mathbf{w}^T \mathbf{x}(k)$$

Our goal is to find an adaptation algorithm which updates  $N_B$  blocks out of the  $B$  available blocks. Partitioning the

S. Werner is with the Signal Processing Laboratory, Helsinki University of Technology, Espoo, Finland (E-mail: stefan.werner@hut.fi)

M. L. R. de Campos and P. S. R. Diniz are with COPPE/EE-Universidade Federal do Rio de Janeiro, RJ, Brazil, (E-mails: {campos,diniz}@lps.ufrj.br).

filter into blocks of coefficients,  $L > 1$ , rather than single coefficients.  $L = 1$ , may at first sight seem to lack any motivation but it has been shown that choosing  $L > 1$  can reduce the computational load and amount of memory required for the implementation [8]. However, for a given number of coefficients to be updated, choosing  $L = 1$  will result in the fastest convergence rate for white input signals. The reason why a slowdown in convergence speed occurs for  $L > 1$  is explained at the end of this section.

Let the  $N_B$  blocks of coefficients to be updated at time instant  $k$  be specified by an index set  $\mathcal{I}_{N_B}(k) = \{i_1(k), \dots, i_{N_B}(k)\}$  with  $\{i_j(k)\}_{j=1}^{N_B}$  taken from the set  $\{1, \dots, B\}$ . Note that  $\mathcal{I}_{N_B}(k)$  depends on the time instant  $k$ . As a consequence, the  $N_B$  blocks of coefficients to be updated can change between consecutive time instants. A question that naturally arises is ‘‘Which  $N_B$  blocks should be updated?’’ The answer to this question can be related to the optimization criterion chosen for the algorithm derivation.

In the conventional NLMS algorithm, the new coefficient vector can be obtained as the vector  $\mathbf{w}(k+1)$  that minimizes the Euclidean distance  $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2$  subject to the constraint of zero *a posteriori* error. Applying the same idea for the partial update of vector  $\mathbf{w}(k)$ , we take the updated vector  $\mathbf{w}(k+1)$  as the vector minimizing the Euclidean distance  $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2$  subject to the constraint of zero *a posteriori* error with the additional constraint of updating only  $N_B$  blocks of coefficients, where each block contains  $L$  coefficients. For this purpose, we introduce the  $N \times N$  block-selection matrix  $\mathbf{A}_{\mathcal{I}_{N_B}(k)}$  having  $N_B$  identity matrices  $\mathbf{I}_{L \times L}$  on its diagonal and zeroes elsewhere. The matrix multiplication  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{w}(k)$  now removes all the blocks that do not belong to the adaptive filter update. Defining the complementary matrix  $\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} = \mathbf{I} - \mathbf{A}_{\mathcal{I}_{N_B}(k)}$  will give  $\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{w}(k+1) = \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{w}(k)$ , which means that only  $N_B$  blocks are updated.

With this notation the optimization criterion for the partial update can be formulated as

$$\mathbf{w}(k+1) = \min_{\mathbf{w}} \|\mathbf{w} - \mathbf{w}(k)\|^2 \text{ subject to} \quad (3)$$

$$\begin{cases} \mathbf{x}(k)^T \mathbf{w} = d(k) \\ \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} (\mathbf{w} - \mathbf{w}(k)) = \mathbf{0} \end{cases}$$

Applying the method of Lagrange multipliers (see Appendix I) gives

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{e(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2}. \quad (4)$$

We see from (4) that only the blocks of coefficients of  $\mathbf{w}(k)$  indicated by the index set  $\mathcal{I}_{N_B}(k)$  are updated, whereas the remaining blocks are not changed from iteration  $k$  to iteration  $k+1$ .

We now concentrate on the choice of the index set  $\mathcal{I}_{N_B}(k)$ . Substituting the recursions in (4) into (3) we get the Euclidean distance as

$$E(k) = \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 = \frac{e^2(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2}$$

For a given value of  $e^2(k)$ , we can conclude that  $E(k)$  achieves its minimum when  $\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|$  is maximized.

Table 1. The PU-NLMS Algorithm

| PU-NLMS ALGORITHM |  |
|-------------------|--|
| for each $k$      |  |
| {                 |  |
|                   | $e(k) = d(k) - \mathbf{x}^T(k) \mathbf{w}(k)$  |
|                   | $\mathbf{z} = [\ \mathbf{x}_1(k)\ ^2, \dots, \ \mathbf{x}_B(k)\ ^2]$   |
|                   | $[\mathbf{y}, \mathbf{i}] = \text{sort}[\mathbf{z}] \quad \% \mathbf{y}, \mathbf{i}: \text{sorted vector and index vector}$                                |
|                   | $\mathbf{i} = \mathbf{i}(B:-1:B-N_b+1) \% N_b \text{ largest norm blocks}$   |
|                   | $\ \mathbf{A}\mathbf{x}\ ^2 = \sum_{j=1}^{N_b} \mathbf{y}(B-j+1)$  |
|                   | for $i = 1$ to $N_b$   |
|                   | {  |
|                   | $\mathbf{I}_{\text{temp}} = [(\mathbf{i}(i)-1)L+1 : \mathbf{i}(i)L]$   |
|                   | $\mathbf{w}(\mathbf{I}_{\text{temp}}) = \mathbf{w}(\mathbf{I}_{\text{temp}}) + \mu e(k) \mathbf{x}(\mathbf{I}_{\text{temp}}) / \ \mathbf{A}\mathbf{x}\ ^2$ |
|                   | }  |
| }                 |  |

In other words, we should update the  $N_B$  blocks of coefficients of  $\mathbf{w}(k)$  with the largest norm  $\|\mathbf{x}_i(k)\|^2$ ,  $i = 0, 1, \dots, B$ .

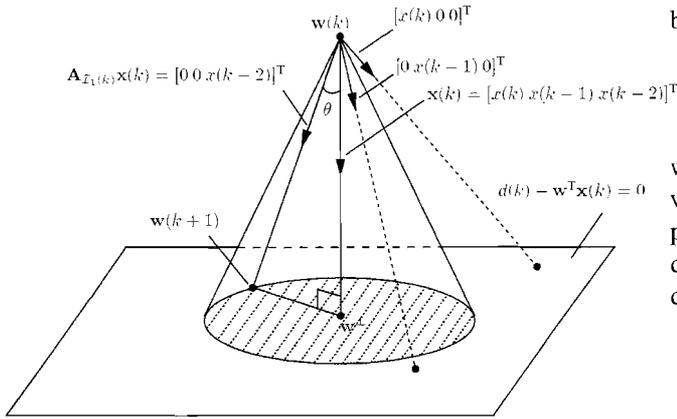
In order to control stability, convergence speed, and error in the mean-squared sense a step size is required, leading to the following final recursion for the PU-NLMS algorithm

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \frac{e(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \quad (5)$$

The pseudo-code for the PU-NLMS algorithm is shown in Table 1. Note that in a practical implementation, the sorting algorithm and calculation of the largest norm blocks need to be implemented with care, see [10].

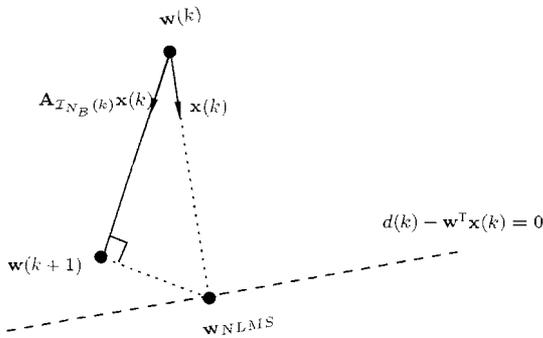
It was mentioned in the beginning of this section that a slowdown in convergence rate will occur for  $L > 1$  in case of white input signals. The reason is that deviation from the optimal input-signal direction  $\mathbf{x}(k)$  is increasing with  $L$ . A geometrical interpretation of the PU-NLMS algorithm update is given in Figure 1 for the case of  $N = 3$  filter coefficients and  $N_B = 1$  block to be updated, where each block contains  $L = 1$  coefficient. In the figure, the component  $x(k-2)$  is the element of largest magnitude in  $\mathbf{x}(k)$ , therefore the matrix  $\mathbf{A}_{\mathcal{I}_1(k)}$ , which specifies the coefficients to update in  $\mathbf{w}(k)$ , is equal to  $\mathbf{A}_{\mathcal{I}_1(k)} = \text{diag}(0 \ 0 \ 1)$ . The solution  $\mathbf{w}^\perp$  in Figure 1 is the solution obtained by the NLMS algorithm abiding the orthogonality principle. The angle  $\theta$  shown in Figure 1 denotes the angle between the direction of update  $\mathbf{A}_{\mathcal{I}_1(k)} \mathbf{x}(k) = [0 \ 0 \ x(k-2)]^T$  and the input vector  $\mathbf{x}(k)$ , and is given from standard vector algebra by the relation  $\cos \theta = \frac{|x(k-2)|}{\sqrt{|x(k)|^2 + |x(k-1)|^2 + |x(k-2)|^2}}$ . In the general case, with  $N_B$  blocks of  $L$  coefficients in the update, the angle  $\theta$  in  $\mathbb{R}^N$  is given by  $\cos \theta = \frac{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|}{\|\mathbf{x}(k)\|}$ .

Finally we note that for a step size  $\mu(k) = \bar{\mu} \|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2 / \|\mathbf{x}(k)\|^2$ , the PU-NLMS in (5) with  $L = 1$  becomes identical to the M-Max NLMS algorithm of [5]. For  $\bar{\mu} = 1$ , the solution is the projection of the solution of the NLMS algorithm with unity step size onto the direction of  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)$ , as illustrated in Figure 2. In next section, where the PU-NLMS algorithm is analyzed, it will be clear that the choice  $\mu = \|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2 / \|\mathbf{x}(k)\|^2$  corresponds



**Figure 1.** Geometric illustration of an update in  $\mathbb{R}^3$  using  $N_B = 1$  block with  $L = 1$  coefficient in the partial update, and with  $|x(k-2)| > |x(k-1)| > |x(k)|$ , the direction of the update is along the vector  $[0 \ 0 \ x(k-2)]^T$  forming an angle  $\theta$  with the input vector  $\mathbf{x}(k)$ .

to the instantaneous estimate of the step size giving the fastest convergence.



**Figure 2.** The solution  $\mathbf{w}(k+1)$  is the PU-NLMS algorithm update obtained with a time-varying step size  $\mu(k) = \|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2 / \|\mathbf{x}(k)\|^2$ , or equivalently, the M-Max NLMS algorithm [5] with unity step size.

### 3. CONVERGENCE ANALYSIS

In this section, the PU-NLMS algorithm [10, 11] is analyzed in the mean-squared sense. New and more accurate bounds on the step size are provided together with closed-form formulas for the prediction of the excess MSE. For the analysis we adopt a simplified model for the signals  $\mathbf{x}(k)$  and  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)$ . The model described in detail in Appendix II uses a simplified distribution for the input-signal vector by employing reduced and countable angular orientations for the excitation which are consistent with the first- and the second-order statistics of the actual input-signal vector. The model was used for analyzing the NLMS algorithm [12], and was shown to yield accurate results. The model was also successfully used to analyze the quasi-Newton (QN) [14] and the binormalized data-reusing LMS (BNDRLMS) [15] algorithms.

It is shown in Appendix II that for the PU-NLMS algorithm to be stable in the mean-squared sense, the step size  $\mu$  should

be bounded as follows:

$$0 < \mu < \frac{2}{\mathbb{E} \left[ \frac{r^2(k)}{\tilde{r}^2(k)} \right]} \approx \frac{2\mathbb{E} [\tilde{r}^2(k)]}{N\sigma_x^2}$$

where  $r^2(k)$  has the same probability distribution as  $\|\mathbf{x}(k)\|^2$ , which in this particular case is a sample of an independent process with chi-square distribution with  $N$  degrees of freedom,  $\mathbb{E} [r^2(k)] = N\sigma_x^2$ , and  $\tilde{r}^2(k)$  has the same probability distribution as  $\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2$ ,

$$\begin{aligned} \tilde{r}^2(k) &= \mathbb{E} \left[ \|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2 \right] \\ &= \mathbb{E} \left[ \sum_{i \in \mathcal{I}_{N_B}(k)} \|\mathbf{x}_i(k)\|^2 \right] \end{aligned} \quad (6)$$

where  $\{\|\mathbf{x}_i(k)\|^2\}_{i=1}^B$  is a sample of a process with a chi-square distribution with  $L$  degrees of freedom. Because only the  $N_B$  blocks with largest norm are considered in the calculation of  $\tilde{r}^2(k)$  we need to evaluate the expression in Equation (6) using *order-statistics*. From Appendix III we get the following formula

$$\begin{aligned} \mathbb{E} [\tilde{r}^2(k)] &= \sum_{j=B+1-N_B}^B \frac{B!}{(j-1)!(B+1-j)!} \times \\ &\int_0^\infty y^2 F_z^{j-1}(y) (1 - F_z(y))^{B-j} f_z(y) dy \end{aligned} \quad (7)$$

where  $F_z(y)$  and  $f_z(y)$  are the cumulative distribution function and the density function, respectively, of a chi-squared variable with  $L$  degrees of freedom. For given  $B$  and  $N_B$ , Equation (7) can be evaluated numerically.

In general the expectation in Equation (6) needs to be evaluated numerically. For the special case of  $L = 2$ , i.e. chi-square distribution is equal to the exponential distribution, and a closed-form solution can be found (see Lemma 1 in Appendix III)

$$\begin{aligned} \mathbb{E} [\tilde{r}_{L=2}^2(k)] &= \\ &\sum_{j=B-N_B+1}^B \sum_{k=0}^{j-1} \frac{2(-1)^k B!}{(B-j)! k! (j-1-k)! (B+1+k-j)^2} \end{aligned} \quad (8)$$

It can also be shown that  $\sigma_x^2 N_B L \leq \mathbb{E} [\tilde{r}^2(k)] \leq N\sigma_x^2$  for i.i.d input signals (see Lemma 2 in Appendix III). A more pessimistic bound on the step size,  $0 \leq \mu \leq 2N_B L / N = 2N_B / B$ , was given in [10] as a consequence of the crude approximation  $\mathbb{E} [\tilde{r}^2(k)] \approx N_B L \sigma_x^2$ . For  $L = 1$  an easily calculated bound that does not require the evaluation of Equation (6) is the one combining the pessimistic bound above with the results for  $L = 2$ . In general  $\mathbb{E} [\tilde{r}^2(k)]$  can also be estimated recursively during the adaptation.

If the step size is chosen within its stability bounds, the final excess MSE after convergence is given by (see Appendix II)

$$\begin{aligned} \Delta \xi_{exc} &\approx N \frac{\mu \sigma_n^2 \sigma_x^2}{2 - \mu \mathbb{E} \left[ \frac{r^2(k)}{\tilde{r}^2(k)} \right]} \mathbb{E} \left[ \frac{1}{\tilde{r}^2(k)} \right] \\ &\approx N \frac{\mu \sigma_n^2 \sigma_x^2}{2\mathbb{E} [\tilde{r}^2(k)] - \mu N \sigma_x^2} \end{aligned} \quad (9)$$

Table 2. Summary PU-NLMS Algorithm Analysis

**Stability range:**

$$0 < \mu < \frac{2E[\tilde{r}^2(k)]}{N\sigma_x^2}, \text{ where } E[\tilde{r}^2(k)] \text{ is given by Eq. (6)}$$

**Excess MSE:**

$$\Delta \zeta_{exc} \approx N \frac{\mu \sigma_n^2 \sigma_x^2}{2E[\tilde{r}^2(k)] - \mu N \sigma_x^2}$$

**Maximum convergence speed:**

$$\mu = \frac{E[\tilde{r}^2(k)]}{N\sigma_x^2}$$

**Recursive estimation of  $E[\tilde{r}^2(k)]$ :**

$$\tilde{r}^2(k) = \alpha \tilde{r}^2(k) + (1 - \alpha) \|\mathbf{A}_{\mathcal{I}_{N_B(k)}} \mathbf{x}(k)\|^2, 0.9 < \alpha < 1$$

**Easily calculated bounds:**

$$L = 1: 0 < \mu < \max[N_b/B, \text{ Eq. (8)}]$$

$$L = 2: 0 < \mu < \text{Eq. (8)}$$

$$L > 3: 0 < \mu < N_b/B$$

When  $N_B L = N$  (full update), Equation (9) is consistent with the results obtained for the conventional NLMS algorithm in [12].

By observing the time evolution of the excess MSE in Equation (16) in Appendix II one can conclude that the maximum convergence speed is obtained for  $\mu = E[\tilde{r}^2(k)/\tilde{r}^2(k)]^{-1} \approx E[\tilde{r}^2(k)/N\sigma_x^2]$ . Use of larger step sizes will neither increase the convergence rate nor decrease the misadjustment. In other words, in practice step sizes above  $\mu_{max}/2$  will not be used. The same can be said about the NLMS algorithm, for which the maximum value for the step size is 2 to guarantee stability but only values smaller than or equal to 1 are used.

Table 2 summarizes the results of the analysis of the PU-NLMS algorithm.

## 4. SIMULATION RESULTS

In this subsection, our analysis of the PU-NLMS algorithm is validated using a system-identification setup. The number of coefficients in the plant chosen was  $N = 64$ , and the input signal was zero-mean Gaussian noise with  $\sigma_x^2 = 1$ . The signal-to-noise ratio (SNR) was set to 30 dB.

Figure 3 shows the learning curves for the case of block size  $L = 1$  using  $N_b = 4, N_b = 8, N_b = 16$ , and  $N_b = 64$  coefficients in the partial update. The curves were obtained through averaging 500 trials. The step size for each value of  $N_b$  was chosen such that convergence to the same level of misadjustment was achieved. The corresponding theoretical learning curves obtained from evaluating Equation (16) in Appendix II were also plotted. As can be seen from the figure, the theoretical curves are very close to the simulations. In Figure 4, the number of coefficients in the partial update is kept fixed,  $N_b L = 8$ , and the number of coefficients in

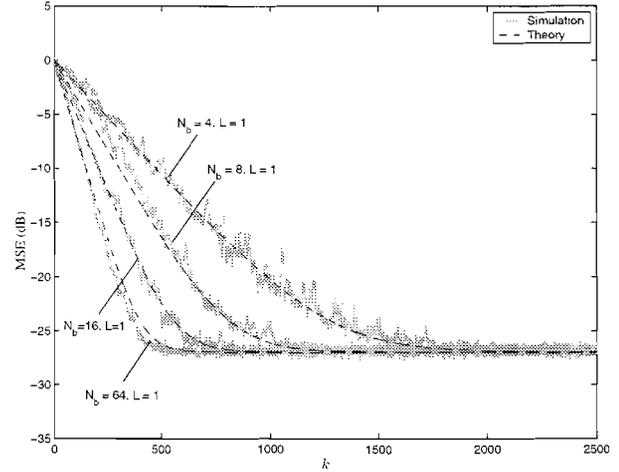
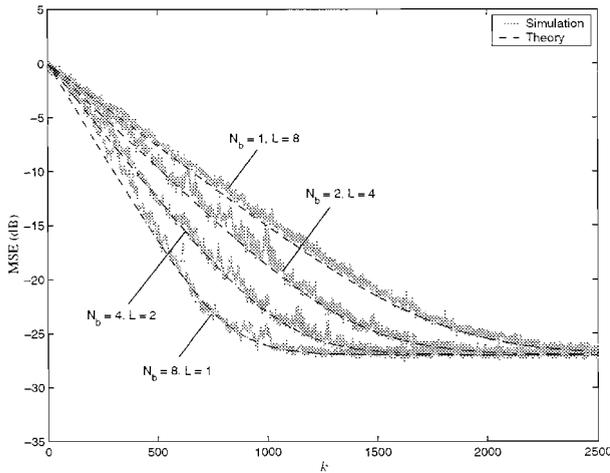


Figure 3. Learning curves for the PU-NLMS algorithm for  $N = 64, L = 1$  coefficient in each block  $N_b = 4, N_b = 8, N_b = 16$  and  $N_b = 64, SNR = 30$  dB.

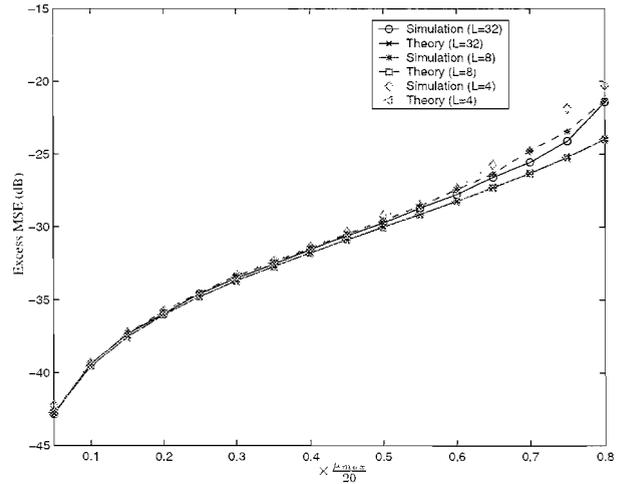
the  $N_b$  blocks are varied. As can be seen from the figure, for a given number  $N_b L$  coefficients in the update, the convergence speed is decreasing with increasing  $L$ . Figure 5 repeats the previous experiment using the recursive estimation of  $E[\tilde{r}^2(k)]$  in Table 2. The resulting curves are very close to the theoretical, validating use of the recursive formula in a practical scenario when a limited knowledge of  $E[\tilde{r}^2(k)]$  can be assumed.

Figure 6 shows the excess MSE as a function of  $\mu$  ranging from  $0.05\mu_{max}$  to  $0.8\mu_{max}$  for different values of  $N_b$ , where  $\mu_{max}$  is given by Equation (18) in Appendix II. Note that the axis is normalized with respect to the maximum step size  $\mu_{max}$ , which is different for each value  $N_b$ . The quantity  $E[\tilde{r}^2(k)]$  needed for the calculation of  $\mu_{max}$  was obtained through numerical integration. For  $N_b = 4, N_b = 8$ , and  $N_b = 16$  the corresponding values were  $E[\tilde{r}^2(k)] = 20.04, E[\tilde{r}^2(k)] = 31.484$ , and  $E[\tilde{r}^2(k)] = 45.794$ , respectively. As can be seen from Figure 6, the theoretical results are very close to the simulations within the range of step sizes considered. Using step sizes larger than  $0.8\mu_{max}$  resulted in poor accuracy or caused divergence. This is expected due to the approximations made in the analysis. However, only step sizes in the range  $\mu \leq 0.5\mu_{max}$  are of practical interest because larger values will neither increase convergence speed nor decrease misadjustment. This fact is illustrated in Figure 7, where the theoretical convergence curves were plotted for different values of  $\mu$  using  $N_b = 8$  and  $N = 64$ . Therefore, we may state that our theoretical analysis is able to predict very accurately the excess MSE for the whole range of practical step sizes.

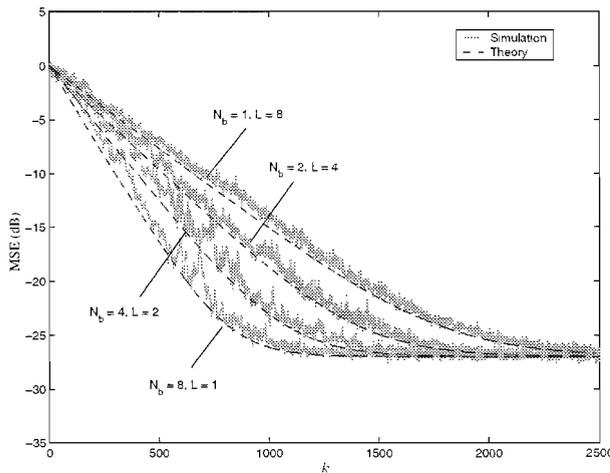
In Figure 8 we compare our results (solid lines) with those provided by [10] (dashed lines). As seen from Figure 8, the results presented in [10] are not accurate even for reasonably high values of  $N_b$ , whereas Figure 6 shows that our analysis is accurate for a large range of  $N_b$ . This comes from the fact that in [10] order statistics was not applied in the analysis, resulting in poor estimates of  $E[\|\mathbf{A}_{\mathcal{I}_{N_B(k)}} \mathbf{x}(k)\|^2]$  for most values of  $N_b < B$ .



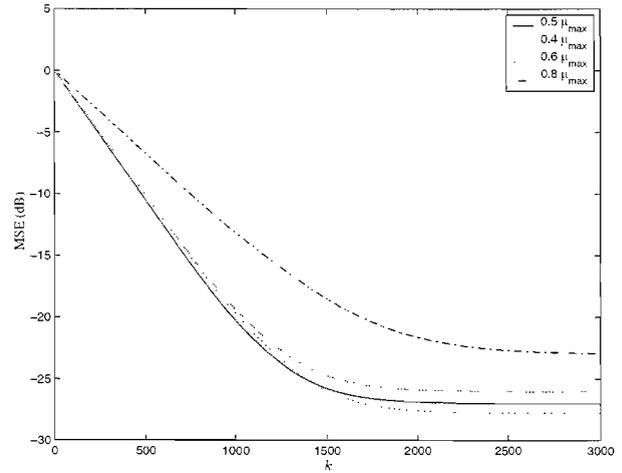
**Figure 4.** Learning curves for the PU-NLMS algorithm for  $N = 64$  and  $N_b L = 8$  in the partial update,  $N_b = 1$ ,  $N_b = 2$ ,  $N_b = 4$ , and  $N_b = 8$ ,  $SNR = 30$  dB.



**Figure 6.** Excess MSE for the PU-NLMS algorithm versus the step size  $\mu$  for  $N = 64$ ,  $L = 1$  coefficient in each block,  $N_b = 4$ ,  $N_b = 8$  and  $N_b = 32$  blocks,  $SNR = 30$  dB.



**Figure 5.** Learning curves for the PU-NLMS algorithm for  $N = 64$  and  $N_b L = 8$  in the partial update using recursive estimation of  $E[\tilde{r}^2(k)]$  (see Table 2) with  $\alpha = 0.95$ ,  $N_b = 1$ ,  $N_b = 2$ ,  $N_b = 4$ , and  $N_b = 8$ ,  $SNR = 30$  dB.



**Figure 7.** Theoretical learning curves for different choice of step size in the PU-NLMS algorithm for  $N = 64$ ,  $L = 1$  and  $N_b = 4$ ,  $SNR = 30$  dB.

## 5. CONCLUSIONS

This paper studied normalized partial-update adaptation algorithms. Convergence analysis for the conventional partial-update NLMS (PU-NLMS) algorithm was presented, which gave further insight to the algorithm in terms of stability, transient and steady-state performances. The analysis was validated through simulations showing excellent agreement. New stability bounds were given for the step size that controls the stability, convergence speed, and final excess MSE of the PU-NLMS algorithm. It was shown that the step size giving the fastest convergence could be related to the time-varying step size of the M-Max NLMS algorithm. These results extend and improve in accuracy previous results reported in the literature. The excellent agreement between the theory and the simulations presented here for the PU-NLMS algorithm has advanced significantly the study of order-statistic-based adaptive filtering algorithms.

## APPENDIX I

The optimization problem in (3) can be solved by the method of Lagrange multipliers having the following objective function

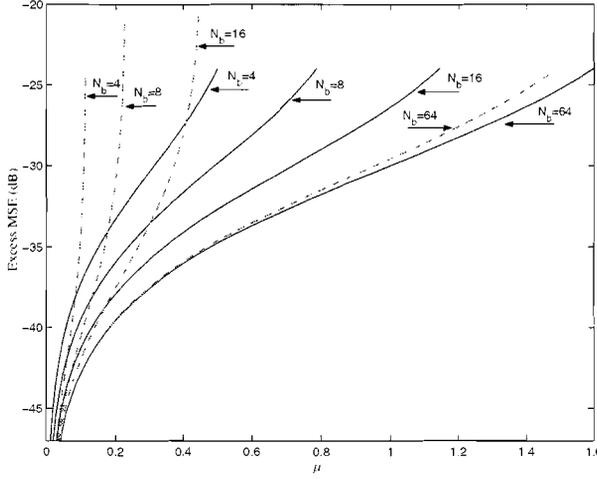
$$f(\mathbf{w}, \lambda_1, \lambda_2) = \|\mathbf{w} - \mathbf{w}(k)\|^2 + \lambda_1(d(k) - \mathbf{x}^T(k)\mathbf{w}) + \lambda_2^T \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} (\mathbf{w} - \mathbf{w}(k)) \quad (10)$$

where  $\lambda_1$  is a scalar and  $\lambda_2$  is an  $N \times 1$  vector. Setting the derivative of (10) with respect to the elements of  $\mathbf{w}$  equal to zero and solving for the new coefficient vector gives us

$$\mathbf{w} = \mathbf{w}(k) + \frac{\lambda_1}{2} \mathbf{x}(k) - \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \frac{\lambda_2}{2} \quad (11)$$

In order to solve for the constraints, multiply Equation (11) by  $\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)}$  and subtract  $\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{w}(k)$  from both sides, i.e.,

$$\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} (\mathbf{w} - \mathbf{w}(k)) = \mathbf{0} = + \frac{\lambda_1}{2} \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) - \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \frac{\lambda_2}{2}$$



**Figure 8.** Comparison of Equation (9) (solid lines) with the excess MSE formula obtained from [10] (dashed lines) for  $L = 1$ ,  $N_b = 4$ ,  $N_b = 8$ ,  $N_b = 16$  and  $N_b = 64$ ,  $SNR = 30$  dB.

where we have used  $\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} = \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)}$ . Therefore,

$$\tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \frac{\lambda_2}{2} = \frac{\lambda_1}{2} \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \quad (12)$$

If we substitute (12) in (11) we get

$$\begin{aligned} \mathbf{w} &= \mathbf{w}(k) + \frac{\lambda_1}{2} \mathbf{x}(k) - \frac{\lambda_1}{2} \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \\ &= \mathbf{w}(k) + \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \frac{\lambda_1}{2} \end{aligned} \quad (13)$$

where we used  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} = (\mathbf{I} - \tilde{\mathbf{A}}_{\mathcal{I}_{N_B}(k)})$ . Finally,  $\lambda_1$  is obtained by pre-multiplying (13) by  $\mathbf{x}^T(k)$ , which gives  $\lambda_1/2 = e(k)/\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2$  for  $\mathbf{x}^T(k) \mathbf{w}(k) = d(k)$ . Our final update is then given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{e(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2}.$$

## APPENDIX II

In this appendix, the PU-NLMS algorithm is analyzed in the mean-squared sense.

### COEFFICIENT ERROR VECTOR

In order to derive expressions for the second-order statistics of the PU-NLMS algorithm we will first derive an expression for the evolution of the coefficient-error vector. Assuming that the desired signal is given by  $d(k) = \mathbf{x}^T(k) \mathbf{w}_{\text{opt}} + n(k)$  and defining the coefficient-error vector as  $\Delta \mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_{\text{opt}}$ , we can express the error as

$$e(k) = n(k) - \mathbf{x}^T(k) \Delta \mathbf{w}(k) \quad (14)$$

Therefore, from Equations (5) and (14) we have

$$\begin{aligned} \Delta \mathbf{w}(k+1) &= \left[ \mathbf{I} - \mu \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right] \Delta \mathbf{w}(k) \\ &\quad + \mu \frac{n(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \end{aligned}$$

### EXCESS MSE FOR WHITE INPUT SIGNALS

For the MSE analysis, we assume that the vectors are excited in a discrete number of directions [12, 14, 15].

The following assumptions are made:

- Independence between  $\mathbf{x}(k)$  and  $\Delta \mathbf{w}(k)$ .
- The vectors  $\mathbf{x}(k)$  and  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)$  are modeled by  $\mathbf{x}(k) = s(k)r(k)\mathbf{v}(k)$  and  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) = \tilde{s}(k)\tilde{r}(k)\tilde{\mathbf{v}}(k)$ , where:

\*  $s(k)$  and  $\tilde{s}(k)$  take on values  $\pm 1$  with probability  $1/2$ .

\*  $r(k)$  and  $\tilde{r}(k)$  are positive real-valued stochastic variables such that  $r^2(k)$  and  $\tilde{r}^2(k)$  have the same probability distribution functions as  $\|\mathbf{x}(k)\|^2$  and  $\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2$ , respectively.

\*  $\mathbf{v}(k)$  is equal to one of the  $N$  orthonormal eigenvectors of  $\mathbf{R} = \mathbf{E}[\mathbf{x}(k)\mathbf{x}^T(k)]$  denoted as  $\mathcal{V}(k)$ , and  $\tilde{\mathbf{v}}(k)$  is equal to one of the  $N$  orthonormal eigenvectors of  $\tilde{\mathbf{R}} = \mathbf{E}[\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T]$  denoted as  $\tilde{\mathcal{V}}(k)$ . For white Gaussian input signals  $\mathbf{v}(k)$  and  $\tilde{\mathbf{v}}(k)$  are uniformly distributed, and  $\mathbf{R}$  and  $\tilde{\mathbf{R}}$  share the same eigenvectors, i.e.,  $\mathcal{V}(k) = \tilde{\mathcal{V}}(k)$ . Therefore,

$$P(\mathbf{v}(k) = \mathcal{V}(k)) = P(\tilde{\mathbf{v}}(k) = \mathcal{V}(k)) = \frac{1}{N} \quad (15)$$

Notice that for any value of  $N_b$ , we have  $s(k) = \tilde{s}(k)$  since the inner product of  $\mathbf{x}(k)$  and  $\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)$  is always positive.

For white input signals, the excess MSE is given by  $\Delta \xi(k+1) = \sigma_x^2 \text{tr}[\text{cov}(\Delta \mathbf{w}(k+1))]$  [1], where

$$\begin{aligned} \text{cov}[\Delta \mathbf{w}(k+1)] &= \mathbf{E}[\Delta \mathbf{w}(k+1) \Delta \mathbf{w}^T(k+1)] \\ &= \mathbf{E} \left[ \left( \mathbf{I} - \mu \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right) \Delta \mathbf{w}(k) \right. \\ &\quad \left. \Delta \mathbf{w}^T(k) \left( \mathbf{I} - \mu \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right)^T \right] \\ &\quad + \mathbf{E} \left[ \mu^2 n^2(k) \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right] \end{aligned}$$

$$\begin{aligned}
 &= E \left[ \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \right] \\
 &- E \left[ \mu \frac{\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right] \\
 &- E \left[ \mu \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right] \\
 &+ E \left[ \mu^2 \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right] \\
 &+ E \left[ \mu^2 n^2(k) \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right]
 \end{aligned}$$

Let us analyze each term separately:

$$\begin{aligned}
 \psi_1 &= \sigma_x^2 \text{tr} \left\{ E \left[ \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \right] \right\} = \Delta \xi(k) \\
 \psi_2 &= \sigma_x^2 \text{tr} \left\{ E \left[ \mu \frac{\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right] \right\} \\
 &= \sigma_x^2 \text{tr} \left\{ E \left[ \mu \frac{\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) s(k) r(k) \bar{s}(k) \bar{r}(k) \mathbf{v}(k) \bar{\mathbf{v}}^T(k)}{\bar{s}^2(k) \bar{r}^2(k)} \right] \right\} \\
 &= \sigma_x^2 E \left[ \mu \Delta \mathbf{w}^T(k) \mathbf{v}(k) \bar{\mathbf{v}}^T(k) \Delta \mathbf{w}(k) \right] E \left[ \frac{s(k) r(k) \bar{s}(k) \bar{r}(k)}{\bar{r}^2(k)} \right] \\
 &= \frac{\mu}{N} E \left[ \frac{r(k)}{\bar{r}(k)} \right] \Delta \xi(k)
 \end{aligned}$$

where we used Equation (15). Since  $\text{tr} \{ \mathbf{A} \mathbf{B} \} = \text{tr} \{ \mathbf{B} \mathbf{A} \}$  we will have  $\psi_2 = \psi_3$ .

$$\begin{aligned}
 \psi_4 &= \sigma_x^2 \mu^2 \times \\
 &\text{tr} \left\{ E \left[ \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right] \right\} \\
 &= \sigma_x^2 \mu^2 \times \\
 &E \left[ \frac{\Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T \mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right] \\
 &= \sigma_x^2 \mu^2 E \left[ \frac{\Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k)}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^2} \right] \\
 &= \sigma_x^2 \mu^2 E \left[ \Delta \mathbf{w}^T(k) \mathbf{v}(k) \mathbf{v}^T(k) \Delta \mathbf{w}(k) \frac{r^2(k)}{\bar{r}^2(k)} \right] \\
 &= \mu^2 \frac{1}{N} E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right] \Delta \xi(k)
 \end{aligned}$$

$$\begin{aligned}
 \psi_5 &= \sigma_x^2 \mu^2 \text{tr} \left\{ E \left[ n^2(k) \frac{\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k) \mathbf{x}^T(k) \mathbf{A}_{\mathcal{I}_{N_B}(k)}^T}{\|\mathbf{A}_{\mathcal{I}_{N_B}(k)} \mathbf{x}(k)\|^4} \right] \right\} \\
 &= \sigma_x^2 \sigma_n^2 \mu^2 E \left[ \frac{1}{\bar{r}^2(k)} \right]
 \end{aligned}$$

Finally we obtain the expression for the excess MSE

$$\begin{aligned}
 \Delta \xi(k+1) &\approx \psi_1 - \psi_2 - \psi_3 + \psi_4 + \psi_5 \\
 &= \left\{ 1 - \frac{\mu}{N} \left( 2E \left[ \frac{r(k)}{\bar{r}(k)} \right] - \mu E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right] \right) \right\} \Delta \xi(k) \\
 &\quad + \mu^2 \sigma_x^2 \sigma_n^2 E \left[ \frac{1}{\bar{r}^2(k)} \right]
 \end{aligned}$$

which can be approximated as

$$\begin{aligned}
 \Delta \xi(k+1) &\approx \left\{ 1 - \frac{\mu}{N} \left( 2 - \mu E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right] \right) \right\} \Delta \xi(k) \\
 &\quad + \mu^2 \sigma_x^2 \sigma_n^2 E \left[ \frac{1}{\bar{r}^2(k)} \right]
 \end{aligned} \tag{16}$$

where the conservative approximation  $E \left[ \frac{r(k)}{\bar{r}(k)} \right] \approx 1$  was used. The stability region in the mean-squared sense for  $\mu$  is

$$0 < \mu < \frac{2}{E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right]} \tag{17}$$

and the step size  $\mu = 1/E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right]$  yields maximum reduction of  $\Delta \xi(k)$  in (16). Further simplifications with  $E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right] \approx \frac{E[r^2(k)]}{E[\bar{r}^2(k)]}$  give us

$$0 < \mu < \frac{2E[\bar{r}^2(k)]}{N\sigma_x^2} \tag{18}$$

where  $E[r^2(k)] = N\sigma_x^2$  and  $E[\bar{r}^2(k)]$  can be calculated using order statistics (see also Appendix III) and knowledge of  $N_b$  and  $N$ . A more pessimistic bound can be obtained by using the relation  $E[\bar{r}^2(k)] \geq N_b L \sigma_x^2$  (see Appendix III) giving

$$0 < \mu < \frac{2N_b L}{N} = \frac{2N_b}{B} \tag{19}$$

which corresponds to the bound given in [10]. We stress that the analysis presented in this appendix shows that step sizes larger than the ones indicated by Equation (19) may be used according to Equation (18).

For  $k \rightarrow \infty$  we have

$$\begin{aligned}
 \Delta \xi_{exc} &\approx N \frac{\mu \sigma_n^2 \sigma_x^2}{2 - \mu E \left[ \frac{r^2(k)}{\bar{r}^2(k)} \right]} E \left[ \frac{1}{\bar{r}^2(k)} \right] \\
 &\approx N \frac{\mu \sigma_n^2 \sigma_x^2}{2E[\bar{r}^2(k)] - \mu E[r^2(k)]} \\
 &= N \frac{\mu \sigma_n^2 \sigma_x^2}{2E[\bar{r}^2(k)] - \mu N \sigma_x^2}
 \end{aligned}$$

### APPENDIX III

In this Appendix it is shown how to obtain numerically  $E[\bar{r}^2(k)]$  used in the step size bound derived in Appendix II. An exact formula is given for the case of  $L = 2$  coefficients in each block. Finally, a lower bound on  $E[\bar{r}^2(k)]$  is provided. This parameter was also required in the analysis of the M-Max NLMS algorithm [6] with  $L = 1$ , which used a similar approach as presented here.

The basic problem here is to calculate  $k$ th moment of ordered statistics. This problem has received much attention in the past, see, e.g., [16]–[18], where recursion formulas and tables were produced for expected values and moments of ordered statistics for various different distributions.

Let  $\mathbf{y} = [y_1 \ y_2 \ \dots \ y_B]^T$  be a vector containing the elements of vector  $\mathbf{z} = [z_1 \ z_2 \ \dots \ z_B]^T$  ordered in value, i.e.,  $y_1 \leq y_2 \leq \dots \leq y_j \leq \dots \leq y_B$ . The probability density function  $f_j(y)$  of the  $j$ th element in  $\mathbf{y}$  is given by [19]

$$f_j(y) = \frac{B! F_z^{j-1}(y) [1 - F_z(y)]^{B-j} f_z(y)}{(j-1)!(B-j)!}$$

where  $f_z(z)$  is the density of the unsorted random variables in vector  $\mathbf{z}$  and  $F_z^{j-1}(z)$  is their cumulative distribution to the

power of  $j - 1$ . The  $k$ th moment of the  $j$ th element is given by

$$\begin{aligned} E[y_j^k] &= \int_{-\infty}^{\infty} y_j^k f_j(y) dy \\ &= \frac{B!}{(j-1)!(B-j)!} \times \\ &\quad \int_{-\infty}^{\infty} y^k F_z^{j-1}(y) [1 - F_z(y)]^{B-j} f_x(y) dy \end{aligned} \quad (20)$$

The PU-NLMS algorithm chooses the  $N_b$  blocks of elements in  $\mathbf{x}(k)$  that have the largest magnitude  $\|\mathbf{x}_i(k)\|^2$ ,  $i = 1, \dots, B$ . For the case of Gaussian input signals,  $z_i = \|\mathbf{x}_i(k)\|^2$  is chi-distributed with  $L$  degrees of freedom. Therefore, if we order the values in  $\{\|\mathbf{x}_i(k)\|^2\}_{i=1}^B$  in magnitude,  $E[\tilde{r}^2(k)]$  can be found by calculating the first moment,  $k = 1$ , in (20) for  $j = B + 1 - N_b, \dots, B$  using the cumulative distribution and density function of a chi-squared variable with  $L$  degrees of freedom, i.e.,

$$\begin{aligned} E[\tilde{r}^2(k)] &= \sum_{j=B+1-N_b}^B \frac{B!}{(j-1)!(B-j)!} \times \\ &\quad \int_0^{\infty} y F_z^{j-1}(y) (1 - F_z(y))^{B-j} f_z(y) dy \end{aligned}$$

which for given  $B$  and  $N_b$  in general needs to be evaluated numerically. An exception is for  $L = 2$  where for Gaussian input-signals, the chi-square distribution with  $L = 2$  degrees of freedom becomes equal to the exponential distribution.

**Lemma 1** *If the input signal  $x(k) \sim \mathcal{N}(0, 1)$ , then if  $L = 2$  we have*

$$\tilde{E}[r^2(k)] = \sum_{j=B-N_B+1}^B \sum_{k=0}^{j-1} \frac{2(-1)^k B!}{(B-j)! k! (j-1-k)! (B+1+k-j)^2}$$

*Proof:*

$$\begin{aligned} E[y_j] &= \frac{B!}{(j-1)!(B-j)!} \times \\ &\quad \int_0^{\infty} y \left(1 - e^{-\frac{y}{2}}\right)^{j-1} e^{-\frac{(B-j)y}{2}} \frac{1}{2} e^{-\frac{y}{2}} dy \\ &= \frac{B!}{(j-1)!(B-j)!} \times \\ &\quad \int_0^{\infty} y \sum_{k=0}^{j-1} \binom{j-1}{k} (-1)^k e^{-\frac{ky}{2}} \frac{1}{2} e^{-\frac{(B+1-j)y}{2}} dy \\ &= \frac{B!}{(B-j)!} \sum_{k=0}^{j-1} \frac{2(-1)^k}{k!(j-1-k)!(B+1+k-j)^2} \end{aligned}$$

where we have used  $\int_0^{\infty} y e^{-ay} dy = \frac{1}{a^2}$ . Evaluation of  $E[r^2(k)] = \sum_{j=B-N_B+1}^B E[y_j]$ , concludes the proof.  $\square$

As was mentioned above,  $L = 2$  is a special case and in general we need to evaluate  $E[r^2(k)]$  numerically. However, with the aid of the previous results we are able to calculate bounds for  $E[\tilde{r}^2(k)]$ , as stated in the following lemma.

**Lemma 2** *If the input signal  $x(k)$  is i.i.d with zero mean and variance  $\sigma_x^2$  then, for  $N_b \leq B$ ,  $E[\tilde{r}^2(k)] = \sum_{j=B+1-N_b}^B E[y_j]$  is bounded as follows:*

$$N_b \sigma_x^2 \leq E[\tilde{r}^2(k)] \leq N \sigma_x^2$$

*Proof:* In the proof we need the following relations

$$E[y_1] \leq E[y_2] \leq \dots \leq E[y(k)] \leq \dots \leq E[y_B] \quad (21)$$

$$\sum_{k=1}^B E[y(k)] = N E[x^2(k)] \quad (22)$$

Relation (21) holds true by definition, and (22) holds true for an arbitrary distribution for which the integral in (20) converges [18], as shown below

$$\begin{aligned} \sum_{j=1}^B E[y_j] &= \sum_{j=1}^B \frac{B!}{(j-1)!(B-j)!} \times \\ &\quad \int_{-\infty}^{\infty} y F_z^{j-1}(y) [1 - F_z(y)]^{B+1-j} f_z(y) dy \\ &= \int_{-\infty}^{\infty} \left\{ \sum_{j=1}^B \frac{B!}{(j-1)!(B-j)!} F_z^{j-1}(y) \right. \\ &\quad \left. [1 - F_z(y)]^{B-j} \right\} y f_z(y) dy \\ &= \int_{-\infty}^{\infty} \left\{ B \sum_{k=0}^{B-1} \frac{(B-1)!}{k!(B-1-k)!} F_z^k(y) \times \right. \\ &\quad \left. [1 - F_z(y)]^{B-1-k} \right\} y f_z(y) dy \\ &= \int_{-\infty}^{\infty} B y f_z(y) dy \\ &= \int_{-\infty}^{\infty} B \sum_{i=1}^{N/B} \sum_{l=1}^L x(k-l+1 - [i-1]L)^2 f_x(x) dy \\ &= N E[x^2(k)] \end{aligned}$$

where we used  $\sum_{k=0}^{B-1} \binom{B-1}{k} p^k q^{B-1-k} = (p+q)^N$ . From relations (21) and (22) it follows that  $N_b \sigma_x^2 \leq E[\tilde{r}^2(k)] \leq N \sigma_x^2$  holds true for  $N_b \leq B$ .  $\square$

The lower bound given by Lemma 2 should be considered loose and is unlikely to be attained, and for Gaussian input signals and using  $L = 1$ , it can easily be shown that the lower bound is never attained unless  $L = N$  [20].

## REFERENCES

- [1] P. S. R. Diniz, *Adaptive Filtering — Algorithms and Practical Implementation*, Boston, MA: Kluwer Academic Press, 2nd Edition, 2002.
- [2] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Transaction Circuits and Systems-II*, vol. 44, pp. 209–216, Mar. 1997.
- [3] S. C. Douglas, "A family of normalized LMS algorithms," *IEEE Signal Processing Letters*, vol. 1, no. 3, pp. 49–51, Mar. 1994.

- [4] S. C. Douglas, "Analysis and implementation of the max-NLMS adaptive filter," *Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, Pacific Grove, CA, pp. 659–663, Oct. 1995.
- [5] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Transaction Signal Processing*, vol. 47, pp. 1421–1427, May 1999.
- [6] M. I. Haddad, K. Mayyas, and M. A. Khasawneh, "Analytical development of the MMAXNLMS algorithm," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'99*, Phoenix, USA, vol. 4, pp. 1853–1856, Mar. 1999.
- [7] M. Godavarti and A. O. Hero III, "Stochastic partial update LMS algorithm for adaptive arrays," *IEEE Sensor Array and Multichannel Signal Processing Workshop*, Boston, USA, pp. 322–326, Mar. 2000.
- [8] T. Schertler, "Selective block update of NLMS type algorithms," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'98*, Seattle, USA, vol. 3, pp. 1717–1720, May 1998.
- [9] S. Attallah and S. W. Liaw, "DCTLMS algorithm employing partial coefficient updates," *IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000*, pp. 218–223, Oct. 2000.
- [10] K. Doğançay and O. Tanrikulu, "Adaptive Filtering algorithms with selective partial update," *IEEE Transaction Circuits and Systems-II*, vol. 48, pp. 762–769, Aug. 2001.
- [11] K. Doğançay and O. Tanrikulu, "Selective-partial-update NLMS and affine projection algorithms for acoustic echo cancellation," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'2000*, Istanbul, Turkey, vol. 1, pp. 448–451, Jun. 2000.
- [12] D. T. M. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms," *IEEE Transactions on Signal Processing*, vol. 41, pp. 2811–2825, Sep. 1993.
- [13] J. Pitas, "Fast algorithms for running ordering and max/min calculation," *IEEE Transaction Circuits and Systems-II*, vol. 36, pp. 795–804, Jun. 1989.
- [14] M. L. R. de Campos and A. Antoniou, "A new quasi-Newton adaptive filtering algorithm," *IEEE Transactions on Circuits and Systems — Part II*, vol. 44, pp. 924–934, Nov. 1997.
- [15] J. A. Apolinário, M. L. R. de Campos and P. S. R. Diniz, "Convergence analysis of the binormalized data-reusing LMS algorithm," *IEEE Transaction on Signal Processing*, vol. 48, pp. 3235–3242, Nov. 2000.
- [16] H. J. Godwin "Some low moments of order statistics," *Annals of Mathematical Statistics*, vol. 20, pp. 279–285, Jun. 1949.
- [17] H. L. Harter "Expected values of normal order-statistics," *Biometrika*, vol. 48, pp. 151–165, Jun. 1961.
- [18] Z. Govindarajulu, "Exact moments of order statistics in samples from the chi-distribution (1 d.f.)," *Annals of Mathematical Statistics*, vol. 33, pp. 1292–1305, Dec. 1963.
- [19] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 1991.
- [20] S. Werner, *Reduced Complexity Adaptive Filtering Algorithms with Applications to Communications Systems*, Ph.D. Thesis, Helsinki University of Technology, Department of Electrical Engineering, Signal Processing Laboratory, 2002.

**Stefan Werner** received the M.Sc. degree in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1998, and the Dr.Tech. degree (with honors) from the Signal Processing Laboratory, Helsinki University of Technology, Helsinki, Finland, in 2002. His research interests are in multiuser commu-

nications and adaptive filtering.

**Marcello L. R. de Campos** was born in Niteroi, Brazil, in 1968. He received the B.Sc. degree (cum laude) from the Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil, in 1990, the M.Sc. degree from COPPE/UFRJ in 1991, and the Ph.D. degree from the University of Victoria, Victoria, BC, Canada, in 1995, all in electrical engineering. In 1996, he was post-doctoral fellow with the Department of Electronics, School of Engineering, UFRJ, and with the Program of Electrical Engineering, COPPE/UFRJ. From January 1997 until May 1998, he was Associate Professor with the Department of Electrical Engineering (DE/3), Military Institute of Engineering (IME), Rio de Janeiro. He is currently Associate Professor with the Program of Electrical Engineering, COPPE/UFRJ. From September to December 1998, he was visiting the Laboratory for Telecommunications Technology, Helsinki University of Technology, Espoo, Finland. Dr. de Campos served as IEEE Communications Society Regional Director for Latin America in 2000 and 2001. In 2001, he received a Nokia Visiting Fellowship to visit the University of Oulu, Oulu, Finland. His research interests include adaptive signal processing, statistical signal processing, and signal processing for communications. He is a Senior Member of IEEE and a Member of the Brazilian Telecommunications Society.

**Paulo S. R. Diniz** was born in Niteroi, Brazil. He received the Electronics Eng. degree (Cum Laude) from the Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil, in 1978, the M.Sc. degree from COPPE/UFRJ in 1981, and the Ph.D. degree from Concordia University, Montreal, PQ, Canada, in 1984, all in electrical engineering. Since 1979, he has been with the Department of Electronic Engineering (undergraduate), UFRJ. He has also been with the Program of Electrical Engineering (graduate), COPPE/UFRJ, since 1984, where he is presently a Professor. He served as Undergraduate Course Coordinator and as Chairman of the Graduate Department. He is one of the three senior researchers and coordinators of the National Excellence Center in Signal Processing. From January 1991 to July 1992, he was a Docent Research Associate with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, B.C., Canada. He also holds the Docenti position at Helsinki University of Technology, Helsinki, Finland. From January 2002 to June 2002, he was a Melchor Chair Professor with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN. His teaching and research interests are in analog and digital signal processing, adaptive signal processing, digital communications, wireless communications, multirate systems, stochastic processes, and electronic circuits. He has published several refereed papers in some of these areas and wrote the books *Adaptive Filtering: Algorithms and Practical Implementation* (Boston, MA: Kluwer, Second Ed., 2002) and *Digital Signal Processing: System Analysis and Design* (Cambridge, U.K.: Cambridge Univ. Press, 2002; with E. A. B. da Silva and S. L. Netto). Dr. Diniz was the Technical Program Chair of the 1995 MWSCAS, Rio de Janeiro, Brazil. He has been on the technical committee of several international conferences including ISCAS, ICECS, EU-SIPCO, and MWSCAS. He has served Vice President for region 9 of the IEEE Circuits and Systems Society and as Chairman of the DSP technical committee of the same Society. He served as associate editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II from 1996 to 1999, the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 1999 to 2002, and of the Circuits, Systems, and Signal Processing Journal from 1998 to 2002. He was a distinguished lecturer of the IEEE Circuits and Systems Society for the year 2000 to 2001.