ENHANCING WEAK INPUT MODES FOR IMPROVED NLMS CONVERGENCE

S. Douglas Peters[1] *Benoît Champagne*[2]

[1]Defence Research Establishment Atlantic
9 Grove. St., Box 1012. Dartmouth NS, B2Y 3Z7
(now with Bell Northern Research, Verdun, QC)
[2]INRS-Télécommunications
16 Place du Commerce. Verdun, QC, H3E 1H6

ABSTRACT

In this work, a technique is introduced to whiten the inputs of an adaptive filter in such a way as to improve the convergence of the normalized least mean-squares adaptation algorithm. This approach, based on the orthogonalization of successive input vectors, is shown to provide a better conditioned input while introducing some added misadjustment. It is shown, however, that in some applications the gains achieved are considerably more than the losses incurred.

INTRODUCTION

It has been well documented that the Least-Mean-Squares (LMS) algorithm (as described by Widrow and Hoff in [1]) and its variants converge slowly when the covariance matrix of its input vectors is ill-conditioned (see, for example, [2]). Notwithstanding the work of Slock in [3] which provides an analytic basis that explains the exceptions to this rule, slow convergence remains the primary drawback of LMS-based adaptation procedures. In consequence, a number of methods that speed up LMS convergence have been proposed. In general, the best of these methods involve transforming the input vectors rather than changing the adaptive algorithm *per se*. The attempt is to provide a white input to the adaptive filter, for in this event the convergence of each mode takes place with the same time constant. As a result, the convergence of the Normalized LMS (NLMS) adaptive filter, for example, depends entirely on the number of filter weights and the convergence-controlling parameter, and can be shown to be similar to that of the Recursive Least-Squares (RLS) algorithm, apart from the benefits of RLS initialization to the initial convergence of that method [3,5].

In this paper, the effects of input color on NLMS convergence will be revisited in the context of a simple transformation method. This technique, based on the reduction of dominant input modes, can provide considerable increases in LMS convergence at modest computational cost.

NLMS PRELIMINARIES

The standard NLMS algorithm updates its adjustable filter coefficients, $\hat{\mathbf{w}}$ in accordance with[1]

$$\hat{\mathbf{w}}_{k+1} = \hat{\mathbf{w}}_k + \frac{\bar{\mu} e_k \mathbf{x}_k}{\mathbf{x}_k^T \mathbf{x}_k}.$$

where $\mathbf{x}_k$ is the input vector, the subscript k indicates the current sample number, and $\bar{\mu}$ is the so-called stepsize or convergence-controlling parameter. The error signal, e_k, is given by $e_k = d_k - \hat{\mathbf{w}}_k^T \mathbf{x}_k$, where d_k is the desired response of the adaptive filter.

In the literature, the convergence behaviour of the NLMS algorithm has been linked to the condition[2] of the input covariance matrix, which is usually represented by the ratio of its maximum eigenvalue to its minimum eigenvalue. In particular, Slock has recently provided a very clever decoupled modal analysis to describe the convergence behaviour of this algorithm [3]. Given the modal decomposition of the input covariance matrix

$$\mathbf{R} \triangleq E\left[\mathbf{x}\mathbf{x}^T\right] = \sum_{j=1}^{N} \lambda_j \mathbf{v}_j \mathbf{v}_j^T,$$

where E is the expectation operator, λ_j and $\mathbf{v}_j$ are the eigenvalues and orthonormalized eigenvectors of that matrix, respectively, Slock shows that convergence in the direction of a given input eigenvector takes place with time constant given by

$$\tau_j = \frac{\mathrm{tr}\,\mathbf{R}}{\bar{\mu}(2-\bar{\mu})\lambda_j}. \tag{1}$$

where tr $(\cdot)$ denotes trace. This has a number of interesting implications. For example, the convergence of the NLMS algorithm in two situations with the same input covariance condition may vary widely. With N adjustable filter weights, $\bar{\mu} = 1$, and a condition of $\chi(\mathbf{R})$, this variability may be as much as

$$\chi(\mathbf{R}) + N - 1 < \tau_{\max} < (N-1)\chi(\mathbf{R}) + 1$$

where $\tau_{\max}$ is the time constant corresponding to the minimum eigenvalue, and this range is determined by the two cases of all other eigenvalues equal to the minimum eigenvalue and all other eigenvalues equal to the maximum eigenvalue, respectively.

INPUT ORTHOGONALIZATION

Suppose that the dominant components of the input space could be isolated and removed. Applying such a process to each input vector, related auxiliary input vectors, $\mathbf{x}_k^*$, could be found that consist of those components that are orthogonal to the dominant eigenvectors of $\mathbf{R}$. Now, given that meaningful values of desired signal, d_k^*, could also be found that correspond to these auxiliary inputs, the auxiliary vectors could be applied to the adaptive filter as if they were the original input vectors. The eigenvalues corresponding to the modes which were dominant in the input covariance

[1] We assume real data throughout. The formulation with complex data is, of course, straightforward

[2] or the condition number

0-7803-2766-7-9/95/$4.00 © 1995 IEEE

are now likely to be greatly reduced in the auxiliary covariance. Depending on the extent of this reduction, one of two situations may result. In the first case, the auxiliary input condition may be much better than the condition of the actual input covariance, having removed a large part of the dominant components. In this event, we may dispense with the actual inputs, applying just the auxiliary inputs to the adaptive filter. On the other hand, in the unlikely case that the dominant components are removed altogether then the condition of the auxiliary covariance is extremely poor. In this case, both the actual and auxiliary inputs should be applied to the adaptive filter. In either case, better adaptive convergence will result.

Intuitively, the modes excited by such auxiliary inputs will be those that are present but underexcited in the actual input vectors. If some mode is absent in the input vectors, however, the auxiliary inputs will never recover it. Of course, this represents the case of impersistently exciting inputs, which will result in non-convergence for LMS-based adaptive algorithms and instability for RLS-based algorithms [2].

In general, we have no *a priori* knowledge of the structure of the input covariance matrix. Let us consider, however, the orthogonalization of a number of successive input vectors. In general, this may be accomplished via an orthogonal projection matrix as given by

$$\mathbf{P}_{i,k} = \mathbf{I}_N - \mathbf{X}_{i,k}\left(\mathbf{X}_{i,k}^T\mathbf{X}_{i,k}\right)^{-1}\mathbf{X}_{i,k}^T \tag{2}$$

where $\mathbf{I}_N$ is the $N \times N$ identity matrix and $\mathbf{X}_{i,k}$ is the (assumed full-rank) matrix made up of the $i < N$ previous input vectors:

$$\mathbf{X}_{i,k} = [\mathbf{x}_{k-1}\ \mathbf{x}_{k-2}\cdots\mathbf{x}_{k-i}].$$

The vector

$$\mathbf{x}_{i,k}^* \triangleq \mathbf{P}_{i,k}\mathbf{x}_k \tag{3}$$

is now orthogonal to the i latest input vectors. That is,

$$\mathbf{x}_{k-l}^T\mathbf{x}_{i,k}^* = 0, \quad l = 1, 2, \cdots, i.$$

In order to meaningfully apply the auxiliary inputs $\mathbf{x}_{i,k}^*$ to an adaptive filter, we will manufacture a corresponding "desired response", $d_{i,k}^*$ that approximately embodies the correlation that exists between the actual input and actual desired signal, d_k. This relationship is most commonly written as

$$d_k = \mathbf{w}_0^T\mathbf{x}_k + \epsilon_k, \tag{4}$$

where $\mathbf{w}_0$ is the adaptive target (the Wiener filter), and ϵ is called the additive (or residual) noise. By rewriting (3) as $\mathbf{x}_{i,k}^* = \mathbf{x}_k - \mathbf{X}_{i,k}\mathbf{f}_{i,k}$ where

$$\mathbf{f}_{i,k} \triangleq \left(\mathbf{X}_{i,k}^T\mathbf{X}_{i,k}\right)^{-1}\mathbf{X}_{i,k}^T\mathbf{x}_k,$$

we have,

$$d_{i,k}^* \triangleq \mathbf{w}_0^T\mathbf{x}_{i,k}^* + \epsilon_{i,k}^* = d_k - [d_{k-1}\ d_{k-2}\cdots d_{k-i}]\mathbf{f}_{i,k};$$
$$\epsilon_{i,k}^* = \epsilon_k - [\epsilon_{k-1}\ \epsilon_{k-2}\cdots\epsilon_{k-i}]\mathbf{f}_{i,k}. \tag{5}$$

In general, $\mathbf{x}_{i,k}^*$ and $\epsilon_{i,k}^*$ will be dependent. As N becomes large, however, this dependence will diminish since in this limit the quantities $r_m \triangleq \mathbf{x}_k^T\mathbf{x}_{k-m}$ become independent of the input $\mathbf{x}_k$.

Let us now reflect on the covariance of the auxiliary input vector, $\mathbf{x}_{i,k}^*$, defined above. In effect, we would like to determine the structure of the matrices

$$\mathbf{R}_i^* \triangleq E\left[\mathbf{x}_i^*\mathbf{x}_i^{*T}\right] = E\left[\mathbf{P}_i\mathbf{x}\mathbf{x}^T\mathbf{P}_i\right] =$$
$$\mathbf{R} + E\left[\mathbf{X}_i\mathbf{f}_i\mathbf{f}_i^T\mathbf{X}_i^T\right] - E\left[\mathbf{X}_i\mathbf{f}_i\mathbf{x}^T\right] - E\left[\mathbf{x}\mathbf{f}_i^T\mathbf{X}_i^T\right] \tag{6}$$

where the sample subscript, k, has been removed for clarity under the assumption of input stationarity.

INTERESTING CASES

In general, (6) is a rather difficult expression to simplify. In this section, two limiting cases will be considered. In the first instance, the inputs will be taken to be i.i.d., and in the second, the inputs will be taken to be correlated in the manner of a transversal filter.

Let us consider the case of $\mathbf{R}_1^* = E\left[\mathbf{P}_1\mathbf{x}\mathbf{x}^T\mathbf{P}_1\right]$ under i.i.d. input vectors. Let us also take N to be sufficiently large that the following expressions can be shown to hold

$$E\left[(\mathbf{I}_N - \mathbf{P}_1)\mathbf{x}\mathbf{x}^T(\mathbf{I}_N - \mathbf{P}_1)\right] \approx \frac{\operatorname{tr}\mathbf{R}^2}{\operatorname{tr}^2\mathbf{R}}\mathbf{R};$$
$$E[\mathbf{P}_1] \approx \mathbf{I}_N - \frac{\mathbf{R}}{\operatorname{tr}\mathbf{R}}.$$

Under these approximations, we have the eigenvalues of $\mathbf{R}_1^*$ given by

$$\lambda_{1,j}^* = \lambda_j\left(1 - 2\frac{\lambda_j}{\operatorname{tr}\mathbf{R}} + \frac{\operatorname{tr}\mathbf{R}^2}{\operatorname{tr}^2\mathbf{R}}\right).$$

In particular, the eigenvalue of $\mathbf{R}_1^*$ corresponding to the dominant direction of $\mathbf{R}$ will be considerably less than the corresponding eigenvalue of $\mathbf{R}$. In general, this suggests that $\chi(\mathbf{R}_1^*) < \chi(\mathbf{R})$. That is, the covariance of the first auxiliary input is better conditioned than that of the actual inputs. In the pathological case when $\lambda_{\max} \approx \operatorname{tr}\mathbf{R}$ would we find that $\chi(\mathbf{R}_1^*) > \chi(\mathbf{R})$. In this event, the application of both the actual inputs and the auxiliary inputs to the adaptive filter (i.e., the use of both as "driving inputs" at each sample) will result in all modes being represented better than they were with the actual inputs alone.

Second, let us consider a case in which the input vectors are not i.i.d., as is the case for an adaptive transversal filter whose inputs $x(n)$ are highly correlated. for example. Let the autocorrelation of such an input signal be given by

$$R_x(m) \triangleq E[x(n)x(n-m)] = \alpha^{|m|}, \quad |\alpha| < 1.$$

Now let us investigate the structure of $\mathbf{R}_i^*$ in the limit that N gets large. In this circumstance, we can take the quantities $r_m = \mathbf{x}_k^T\mathbf{x}_{k-m}$ to have a Gaussian distribution and to be independent of $\mathbf{x}_k$. The expected values of these quantities are simply $E[r_m] = N\alpha^{|m|}$. Moreover, we will make use of the approximations (valid for $N \gg 1 - \alpha$) that

$$E\left[\mathbf{X}_i\mathbf{f}_i\mathbf{f}_i^T\mathbf{X}_i^T\right] \approx E\left[\mathbf{X}_i\mathbf{g}_i\mathbf{g}_i^T\mathbf{X}_i^T\right]$$
$$E\left[\mathbf{X}_i\mathbf{f}_i\mathbf{x}^T\right] \approx E\left[\mathbf{X}_i\mathbf{g}_i\mathbf{x}^T\right],$$

where

$$\mathbf{g}_i \triangleq E\left[\mathbf{X}_i^T\mathbf{X}_i\right]^{-1}E\left[\mathbf{X}_i^T\mathbf{x}\right] = [0\ 0\ 0\cdots 0\ \alpha]^T.$$

Under these assumptions, (6) becomes

$$\mathbf{R}_i^* \approx (1+\alpha^2)\mathbf{R} - \alpha E\left[\mathbf{x}_k\mathbf{x}_{k-1}^T + \mathbf{x}_{k-1}\mathbf{x}_k^T\right]$$
$$= (1-\alpha^2)\mathbf{I}_N, \quad i = 1, 2, \cdots, N-1.$$

This is quite a remarkable result. Take, for example, the case in which $\alpha = 0.9$ and $N = 50$. Since the ratio of the smallest eigenvalue of the input covariance matrix to the trace of that matrix is approximately 1:1000, we would expect that the weight vector component in the direction of the corresponding eigenvector would converge about one thousand times as slow as the weight vector component in the dominant input direction. Applying just the first auxiliary input to our adaptive filter, however, we would expect optimal NLMS convergence under the same circumstances.

PRACTICAL CONSIDERATIONS

In this section, algorithmic details and computational complexity will be addressed. In order to refer to the numerous possible algorithms made available through the use of input orthogonalization, a subscript notation is adopted. In general, the NLMS$_{\mathcal{S}}$ algorithm, where $\mathcal{S}$ is some nonempty set of integers such that $\mathcal{S} \subset \{0, 1, \cdots, N-1\}$, results. The integer elements of $\mathcal{S}$ correpond to the inputs used to drive the adaptation process. For example, NLMS$_{\{0\}}$ is simply the standard NLMS algorithm while NLMS$_{\{2\}}$ refers to the algorithm in which only the second auxiliary input is used to drive the NLMS process. Due to the complexity of the general NLMS$_{\mathcal{S}}$ algorithm, we adopt an approximation which does not exactly perform an orthogonalization after the first order projection. For low projection orders, however, losses in performance are very small compared to the gains in efficiency. A specification for the NLMS$_{\{i\}}$ process using the approximate orthogonalization is supplied below with the superscript asterisks omitted for clarity. At right are given the approximate number of multiplications and divisions required for each step in the case of a transversal adaptive filter.

		$\times$	$\div$
1	$\mathbf{x}_{0,k} = \mathbf{x}_k;\ \ d_{0,k} = d_k$		
2	$y_k = \hat{\mathbf{w}}_k^T \mathbf{x}_{0,k}$	N	
3	for $j = 1$ to i		
4	$\quad f_{j,k} = \dfrac{\mathbf{x}_{k-j}^T \mathbf{x}_{j-1,k}}{\mathbf{x}_{k-j}^T \mathbf{x}_{k-j}}$	1	1
5	$\quad \mathbf{x}_{j,k} = \mathbf{x}_{j-1,k} - \mathbf{x}_{k-j} f_{j,k}$	N	
6	$\quad d_{j,k} = d_{j-1,k} - d_{k-j} f_{j,k}$	1	
7	endfor		
8	$e_{i,k} = d_{i,k} - \hat{\mathbf{w}}_k^T \mathbf{x}_{i,k}$	N	
9	$\hat{\mathbf{w}}_{k+1} = \hat{\mathbf{w}}_k + \dfrac{\bar{\mu} e_{i,k} \mathbf{x}_{i,k}}{\mathbf{x}_{i,k}^T \mathbf{x}_{i,k}}$	N	1

The output of the adaptive filter is denoted by y_k. Note that it is not necessary to explicitly construct the orthogonal projection matrices to obtain any $\mathbf{x}_i^*$. A Schmidt orthogonalization, which is approximated in lines 4 and 5 above, is likely to be a more practical alternative. In the case of a transversal filter, the relationship between $f_{i,k}$ and $f_{i,k-1}$ provides for the minimal arithmetic requirements reported for line 4.

Using the informal complexity notation common in the literature in which the NLMS complexity is on the order of $2N$ (i.e., $\tilde{O}[2N]$), the complexity of NLMS$_{\{i\}}$ is $\tilde{O}[(i+3)N]$ for $i > 0$ and a transversal filter. The current recommendation is to limit the application of auxiliary inputs to the first or perhaps second orthogonalization, as there is usually no advantage of extending the concept further. For comparison, "fast" RLS methods have been reported with $\tilde{O}[8N]$, also for a transversal filter. For N-input adaptive filters, of course, the complexities will be significantly larger.

MISADJUSTMENT

The steady-state misadjustment behaviour of the NLMS algorithm has been investigated thoroughly (see, for example, [3] or [4]). The performance of this algorithm using auxiliary inputs, however, is not a simple matter to quantify. The difficulty arises due to the fact that the available misadjustment formulas depend on the driving input and the corresponding minimum mean squared error (MMSE). The misadjustment of interest, however, is a function of the actual inputs, $\mathbf{x}$, and the corresponding MMSE. Under the usual simplifying assumption that the additive noise is independent of $\mathbf{x}$ and white, the misadjustment at convergence can be written as

$$\mathcal{M} \triangleq \frac{E\left[\left(\mathbf{v}^T \mathbf{x}\right)^2\right]}{E[\epsilon^2]} \tag{7}$$

where the weight error vector is defined as $\mathbf{v} \triangleq \mathbf{w}_0 - \hat{\mathbf{w}}$, and ϵ is the additive noise in the assumed model (4). Unfortunately, the standard NLMS misadjustment formula, namely, $\mathcal{M} = \frac{\bar{\mu}}{2-\bar{\mu}}$ applies only in the case in which the actual inputs drive the NLMS process. In the case of auxiliary inputs, we have

$$\frac{E\left[\left(\mathbf{v}^T \mathbf{x}_i^*\right)^2\right]}{E[\epsilon_i^{*2}]} = \frac{\bar{\mu}}{2-\bar{\mu}},$$

but the connection between this expression and (7) is, in general, difficult. As an example, however, let us consider the case of NLMS$_{\{1\}}$ applied to a large N (in absolute terms and with respect to the input autocorrelation) transversal filter. In this event, we may take $\mathbf{f}_1$ to be independent of $\mathbf{x}$, and $E[\mathbf{f}_1] \approx R_x(1)/R_x(0)$, which gives us

$$\mathcal{M}_1 = \frac{\bar{\mu}}{2-\bar{\mu}} + \frac{2R_x(0)R_x(1)E\left[\mathbf{v}_k^T \mathbf{x}_k \mathbf{v}_k^T \mathbf{x}_{k-1}\right]}{\left[R_x^2(0) + R_x^2(1)\right] E[\epsilon^2]}.$$

Unfortunately, the second term on the right hand side of this expression is problematic. It is clear, however, that the misadjustment will be greater for NLMS$_{\{1\}}$ than for the standard NLMS process. For i.i.d. input vectors, however, it is easily shown that these two algorithms provide the same misadjustment performance.

A SIMULATED EXAMPLE

In this section, the frequently utilized example of an adaptive equalizer as found in §9.13 of [2] is examined. Here, the channel is modelled as a finite-duration impulse response filter whose impulse response is given by

$$h_n^{ch} = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\frac{2\pi}{W}(n-2)\right)\right], & n = 1, 2, 3 \\ 0, & \text{otherwise} \end{cases}$$

We use $W = 3.5$, representing the most challenging case considered in [2]. This value of W results in an input covariance condition of 46.8 for $N = 11$. The simulation here copies that in [2] in all respects.

In Figure 1, the eigenvalues of the covariance matrices of the vectors $\mathbf{x}_i^*$ for $i = 0, 1, 2$ are displayed. We observe that the estimated condition of $\mathbf{R}_2^*$ is more than ten times smaller than that of the true covariance matrix. The eigenvalues here were estimated based on 10^4 input samples.

Figure 2 shows the simulated learning curves of the NLMS$_{\{i\}}$ algorithms for $i = 0, 1, 2$. These curves (and

those in Figure 3) represent an ensemble average over 200 independent trials. For clarity, the upper and lower curves of Figures 2 and 3 are offset by +5 and −5 dB, respectively. The convergence-controlling parameter, $\bar{\mu}$, assumed the value of 0.5 for those cases illustrated in Figure 2. We note that the initial slope of the learning curve (a nominal measure of convergence) for NLMS$_{\{2\}}$ compares favourably to that which would have been obtained had the inputs been white. This slope may be calculated from (1) in the case when all of the modal time constants are equal. For example, for $\bar{\mu} = 0.5$, a slope of $-10 \log e/\tau \approx -0.3$dB/sample would be optimal. This slope is also in evidence in the lowest line of Figure 3. This learning curve represents the performance of the RLS algorithm in similar circumstances. In this instance, the RLS "forgetting factor", λ, was chosen in order to match the tracking performance of the NLMS filter with $\bar{\mu} = 0.5$ in accordance with [4]

$$\bar{\mu} \approx \frac{2(1-\lambda)N}{2+(1-\lambda)N}. \tag{8}$$

The remaining curves of Figure 3 represent a comparison of the robustness of the NLMS$_{\{2\}}$ and RLS algorithms. Shown are the learning curves of NLMS$_{\{2\}}$ with $\bar{\mu} = 0.6$ and RLS with similar tracking behaviour according to (8) (i.e.,$\lambda = 0.922$). Note that while these curves exhibit a similar initial slope, the small adaptive time constant resulting from the current values of λ and $\bar{\mu}$ has deleterious effects for both algorithms. Essentially, the excitation is such that the algorithms cannot acquire useful information on the least excited modes using such a small time constant. A longer time constant (smaller $\bar{\mu}$ or larger λ) is required. For the NLMS$_{\{2\}}$ algorithm, this difficulty results in a poorer steady-state misadjustment than would otherwise be expected. For the RLS algorithm, on the other hand, instability results. This suggests that the generalized NLMS$_S$ algorithm can provide tracking performance comparable to the similarly-tuned RLS method with better efficiency and robustness.

Note that the initial inverse covariance estimates for the RLS algorithm were taken to be $(1-\lambda)\mathbf{I}_N$. This initialization, which may be considered poor, was used for purposes of comparison. Proper initialization of the RLS algorithm would result in better initial convergence, but is immaterial when considering the tracking performance.

CONCLUSIONS

A method has been proposed to whiten the input vectors for application to adaptive filters. It has been shown that this approach is suitable for improving the convergence behaviour of NLMS adaptive filters. Based on the orthogonalization of successive input vectors, the proposed technique can reliably and efficiently enhance the performance of this practical adaptive algorithm. Analysis has been provided to demonstrate the basis for this method in two limiting cases. The resulting complexities and misadjustment have been discussed, and the algorithm has been shown to converge in a manner comparable to that of RLS adaptation.

REFERENCES

[1] B. Widrow and M. E. Hoff, Jr., "Adaptive switching circuits," *IRE WESCON Conv. Rec.*, pt. 4, pp. 96–104., 1960.

[2] S. Haykin, *Adaptive Filter Theory*, 2nd Ed., Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

[3] D. T. M. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms," *IEEE Trans. Signal Processing*, vol. 41, pp. 2811–2825, Sept. 1993.

[4] S. D. Peters, *Doubly Adaptive Filters for Nonstationary Applications*, Ph.D. dissertation, University of Victoria, Victoria, B.C., 1993.

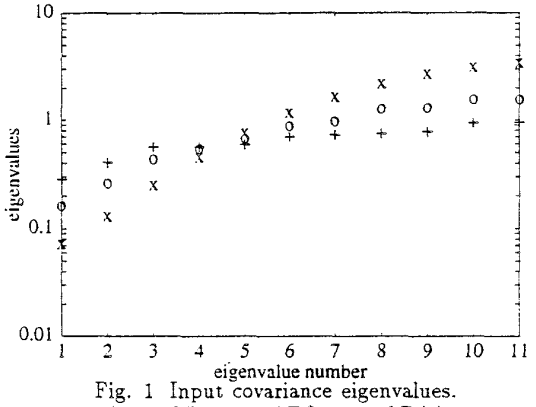

Fig. 1 Input covariance eigenvalues.
(x of R: o of $\mathbf{R}_1^*$; + of $\mathbf{R}_2^*$.)

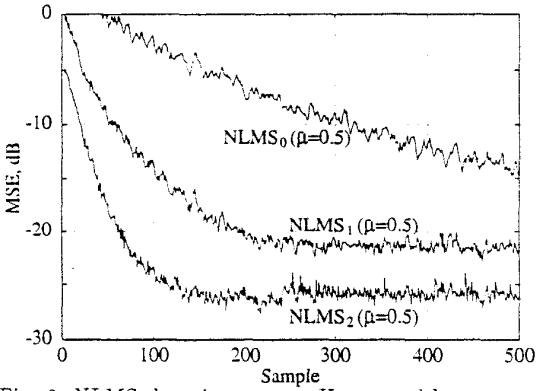

Fig. 2 NLMS$_i$ learning curves. Upper and lower curves are offset by +5 and −5 dB, respectively.

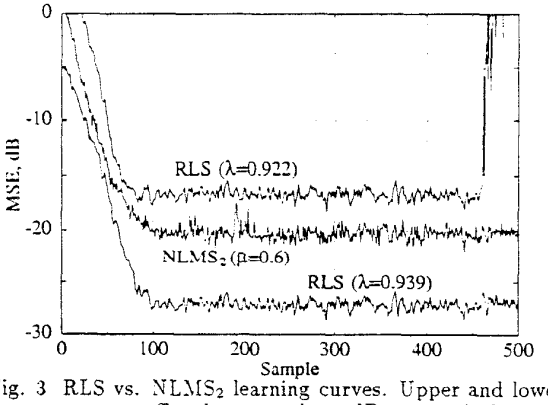

Fig. 3 RLS vs. NLMS$_2$ learning curves. Upper and lower curves are offset by +5 and −5 dB, respectively.