

A Novel Nonlinearity Measure

M. A. Sayid and P. A. Naylor

Imperial College, Exhibition Road, UK

Recent publications in adaptive filtering have shown great interest in the use of nonlinear structures. In particular the quadratic Volterra adaptive filter has proved very popular. In this paper we start by demonstrating the need for nonlinear adaptive filters. Subsequently we present a method for the estimation of the level of nonlinearity of a system so as to be able to select the appropriate adaptation algorithm. Lastly, we apply the nonlinear measure to 2 examples from the open literature.

1. INTRODUCTION

The ideas of linearity permeate throughout adaptive filtering, and signal processing in general, despite the well known fact that real systems and signals often contain significant amounts of nonlinearity. Nevertheless this linear assumption has proved quite successful over the last four decades. However, as the need for greater understanding and accuracy increases so nonlinear techniques are being seriously studied as a practical alternative to more established (linear) methods.

The major drawback to more widespread use of nonlinear techniques has been their mathematical and computational tractability. Much of the theory is defined in terms of integral operators which can only be solved using iterative approximations. Most of the present interest is directed towards Volterra type structures. The reasons for this are that not only can it represent a large class of systems but, more importantly, its output is still linear with respect to its system kernels. This enables many of the ideas of linear filtering to be retained as the only change in the structure of the filter is that a linear filter is replaced by a non-linear one. Indeed the linear filter can be shown to be the special case of the non-linear Volterra filter of order one.

Since the structure of the Volterra adaptive filter is a direct translation of the linear adaptive filter to the non-linear domain it is reasonable to expect them to have similar adaptation schemes. Consequently, most adaptation algorithms presently in use employ the LMS concept.

In section 2 some basic theory behind Volterra Filters is outlined, including the commonest adaptation algorithms. The main purpose of this section is to assess the viability of non-linear adaptive filtering. In particular a system identification scheme is used to demonstrate our ideas. As

a means of comparison the linear LMS and NLMS algorithms are also used. Both the noise free and additive noise cases are dealt with.

In section 3 we discuss the problem of finding a technique to measure the level of nonlinearity present in a system. We outline a new measure which is applicable to not only Volterra structures but to any separable nonlinear structure. Conclusions are drawn from this study in section 4, followed by some relevant references.

2. THE NEED FOR NONLINEAR ADAPTIVE FILTERING

The issue of demonstrating the need for nonlinear structures is approached using the system identification set-up where the system to be identified is the sum of an 8th order linear MA section and a single quadratic nonlinearity. That is, the desired signal input to the adaptive filter can be described as

$$\begin{aligned} d(k) &= \text{Linear Part} + \text{Quadratic Part} \\ &= H(z)x(k) + \alpha x^2(k) \end{aligned} \quad (2.1)$$

where $H(z)$ is the z-domain transfer function of an MA(8) system [4], $x(k)$ is the adaptive filter input, and α is a controllable constant which is used to alter the nonlinear contribution to the desired signal $d(k)$.

2.1 Adaptation Rules

The performance of the adaptive filter is evaluated using the linear and nonlinear adaptation rules described in Tables 2.1-2.2. Further details of these algorithms can be found in [2], [3], [5], [6]. Notice that only the weight update formulas, for algorithms other than the linear and nonlinear LMS, are outlined, since all other equations remain unchanged.

The first adaptation scheme outlined in Tables 2.1-2.2 is the well-known LMS routine. Although it is easy to implement and is computationally attractive some form of normalisation is usual when working with real signals. The most basic form of normalisation possible is to normalise with respect to the input signal. The resulting algorithm will be referred to as the Standard NLMS algorithm. This is one of the commonest

LMS Algorithm
Initialization : $w(0) = 0$
Adaptive Filter : $y(k) = w^T(k)x(k)$
Error Equation : $e(k) = d(k) - x(k)$
Weight Update : $w(k+1) = w(k) + \mu x(k)e^*(k)$
Standard NLMS Algorithm
Weight Update : $w(k+1) = w(k) + \frac{\tilde{\mu} x(k)e^*(k)}{\gamma + x^T(k)x(k)}$
$0 < \tilde{\mu} < 2, \quad \gamma < 1$
Power NLMS Algorithm
Initialization : $p(k) = 1$
Power Update : $p(k+1) = \lambda p(k) + (1-\lambda)x^2(k)$
Weight Update : $w(k+1) = w(k) + \frac{\tilde{\mu} x(k)e^*(k)}{p(k)}$
$0 < \tilde{\mu} < 2, \quad 0 < \lambda < 1$

Table 2.1 Linear Adaptation Schemes

LMS Algorithm
Initialization : $w_1(0) = 0, w_2(0,0) = 0$
Adaptive Filter : $y(k) = w_1^T(k)x(k) + w_2^T(k,k)x^T(k)x(k)$
Error Equation : $e(k) = d(k) - x(k)$
Weight Update : $w_1(k+1) = w_1(k) + \mu_1 x(k)e^*(k)$ $w_2(k+1,k+1) = w_2(k,k) + \mu_2 x^T(k)x(k)e^*(k)$
Standard NLMS Algorithm
Weight Update : $w_1(k+1) = w_1(k) + \frac{\tilde{\mu}_1 x(k)e^*(k)}{\gamma_1 + x^T(k)x(k)}$ $w_2(k+1,k+1) = w_2(k,k) + \frac{\tilde{\mu}_2 x(k)e^*(k)}{\gamma_2 + x^T(k)x(k) ^2}$
$0 < \tilde{\mu}_1, \tilde{\mu}_2 < 2, \quad \gamma_1 , \gamma_2 < 1$
Power NLMS Algorithm
Initialization : $p_1(0) = 1, p_2(0,0) = 1$
Power Update : $p_1(k+1) = \lambda p_1(k) + (1-\lambda)x^2(k)$ $p_2(k+1,k+1) = \lambda p_2(k) + (1-\lambda) x^T(k)x(k) ^2$
Weight Update : $w_1(k+1) = w_1(k) + \frac{\tilde{\mu}_1 x(k)e^*(k)}{p_1(k+1)}$ $w_2(k+1,k+1) = w_2(k,k) + \frac{\tilde{\mu}_2 x(k)e^*(k)}{p_2(k+1,k+1)}$
$0 < \tilde{\mu}_1, \tilde{\mu}_2 < 2, \quad 0 < \lambda_1, \lambda_2 < 1$

Table 2.2 Nonlinear Adaptation Schemes

stochastic gradient techniques used in practice, however, closer examination will show a number of problems. Firstly, there exists no universal rule for the correct selection of the denominator constant γ in the weight update equation(s). It should "obviously" be small with respect to the level of input power, however, what is the definition of "small". In practice the choice tends to be empirical or based on a priori knowledge of the input signal. Our experience has been that even small variations in the choice of γ can produce dramatically differing results.

Another problem with the Standard NLMS is that the normalisation is done with respect to the overall signal power, even though the power flow in each filter tap is unequal. In the Power NLMS this is overcome by normalising each filter tap by the power flowing through it. This is achieved by recursively estimating the normalisation factor. The lowpass filtering operation which is inherent to this estimation acts also to introduce a smoothing effect.

The exponential forgetting factor λ used in the Power NLMS performs a similar role to the forgetting factor λ in the RLS algorithm. Indeed it can be shown [5] that the Power NLMS and the RLS are related.

2.2 System Identification

For the simulations presented the input signal in each case was a white Gaussian (0,1) signal.

The linear and nonlinear results are outlined in Figure 2.1-2.4. Consider the linear LMS, noise free case indicated in Figure 2.1(a). At $\alpha=0.0$ the learning curve converges to a steady state value of -135dB in 225 iterations. [Since there is no additive noise present the expected steady state value is - ∞ dB, the -135dB level is a consequence of the finite precision used in our simulations]. For an α of 0.01 the steady state value is seen to be -34dB. As the value of α continues to increase by an order of magnitude the steady state value worsens by 20dB each time. In fact the $\alpha=10.0$ learning curve is only marginally stable as it initially rises before settling into a narrow band.

The linear Standard NLMS performance shown in Figure 2.1(b) exhibits a similar performance to the LMS ones, except for three noticeable differences. Firstly, steady state is reached more quickly than the corresponding LMS learning curve.

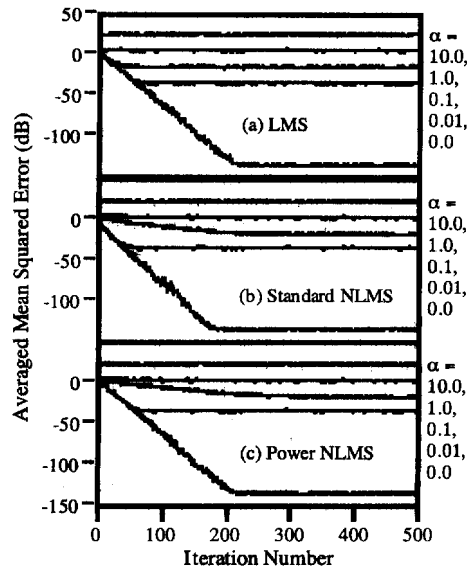


Figure 2.1 Linear learning curves with no additive noise

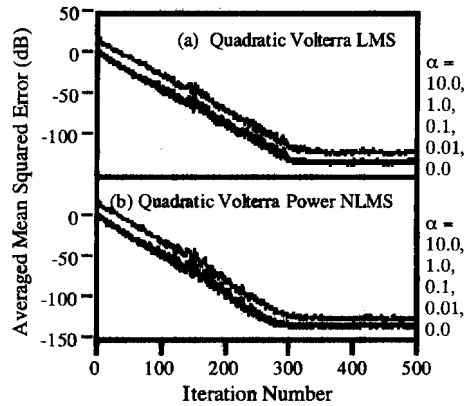


Figure 2.2 Nonlinear learning curves with no additive noise

Secondly, at higher values of α (≥ 1) the steady state value attained is marginally improved. Lastly, even at $\alpha=10.0$ there is not a hint of approaching instability in the learning curve. The performance of the Power NLMS, Figure 2.1(c), is similarly improved over the LMS. However, the Power NLMS curves are seen to be smoother than the corresponding Standard NLMS ones.

The Quadratic Volterra adaptive filter employed here has the following filter equation.

$$y(k) = \sum_{m_1=0}^{N-1} w_1(m_1)x(k-m_1) + \sum_{m_1=0}^{N-1} \sum_{m_2=0}^{N-1} w_2(m_1, m_2)x(k-m_1)x(k-m_2) \quad (2.2)$$

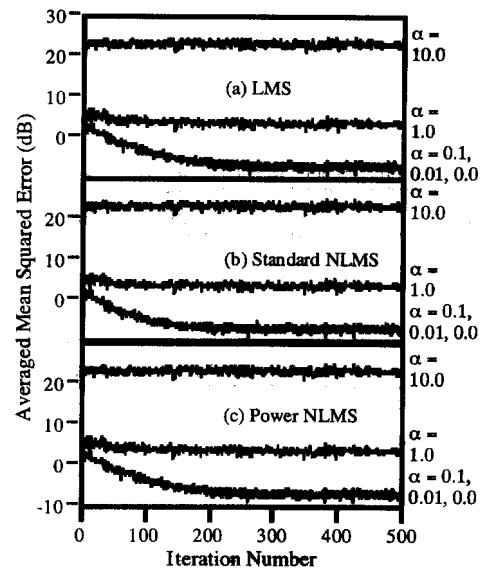


Figure 2.3 Linear learning curves with 10 dB additive noise

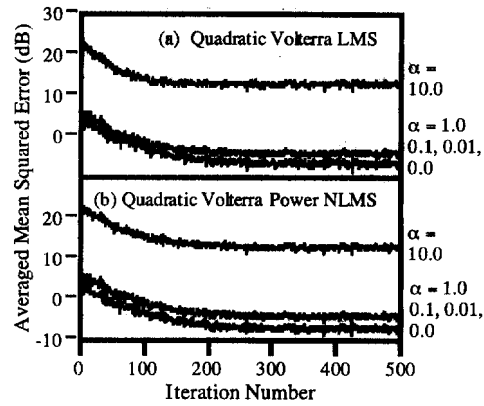


Figure 2.4 Nonlinear learning curves with 10 dB additive noise

Since the Volterra filter is quadratic there are two stepsizes to account for, as indicated in Table 2.2. For our simulations we have found that using the same μ for the linear and nonlinear sections produces the best results. The noise free case simulations are outlined in Figure 2.2. For $\alpha \leq 1$ the learning curves are closely bunched at the -135dB steady state value. Convergence occurring in 340 iterations for all cases. At $\alpha=10.0$ the steady state value is worsened by 15dB to -120dB, reached after 360 iterations. The differences between the LMS and the Standard and Power NLMS were as in the linear case. Therefore we have elected not to show the Volterra Standard NLMS learning curves.

An important characteristic of Volterra Filters can be seen in Figure 2.1 and 2.2. The time constants of the

Volterra Filter increases with the addition of each new term. This is particularly visible in the $\alpha=0.0$ learning curve. Whereas the linear filters achieve steady state in little over 200 iterations the Volterra Filter does not achieve steady state until 340 iterations. This observation is true in general of nonlinear systems. In the case of Volterra Filters it is expected the addition of cubic, quartic, etc., sections will further increase convergence times.

For 10dB Gaussian additive noise the linear LMS results are shown in Figure 2.3(a). [Notice that the learning curves in Figure 2.3 and 2.4 start at a level slightly above 0dB, and not 0dB. This value is equal to $10 \log_{10}(d^2)$]. For $\alpha \leq 0.1$ the learning curves are practically superimposed. Convergence occurs in 300 iterations, with a final mse which is 10dB below its starting level. At $\alpha=1.0$ the learning curve converges very slightly, falling from 5dB to 4dB in 600 iterations. By the time $\alpha=10.0$ the learning curve is showing small signs of divergence.

The linear NLMS learning curves are indicated in Figure 2.3(b,c). Again performance is comparable to those obtained for the LMS algorithm, bar the observations made for the noise free case. For $\alpha \leq 0.1$ the learning curves are much more closely bunched than previously.

The Volterra filter results are shown in Figure 2.4. When $\alpha \leq 0.1$ the learning curves are identical. In each case steady state is reached after 300 iterations, and the final mse is 10dB below its starting value. The $\alpha=1.0$ curve is similar but with a worsened steady state value that is 8dB below its starting level. That is, a worsening in performance of 2dB. Some convergence is visible even when $\alpha=10.0$.

3. LEVEL OF NONLINEARITY

3.1 Definition of SIM

The simulations presented in Section 2 have shown how the presence of a single nonlinear term can adversely affect the performance of an adaptation algorithm, depending on the magnitude of the nonlinear contribution. Presently, we are not aware of any techniques which, when given an arbitrary signal can estimate the level of nonlinearity present in the signal.

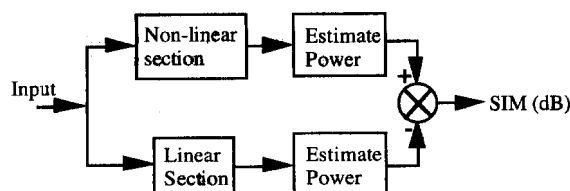


Figure 3.1 Set-up to measure the level of nonlinearity (SIM) present in a system, in dBs

Traditionally, systems have been described as being "weakly" or "strongly" nonlinear. However, what exactly do we mean by a weakly or strongly nonlinear system? The descriptions are clearly open to conjecture. What is required is a measure of some form which can assign an unambiguous figure to the level of nonlinearity. In real applications such questions are difficult to answer. However, it is still of considerable importance to have some measure to account for the presence of the nonlinearity, at the very least to be able to determine the usefulness, or otherwise, of new and existing algorithms.

Here, we propose an a posteriori measure of the level of nonlinearity present in a given system, using the ideas of energy flow through the system. By "given" we mean that a parametric description of the system is available. The system is also assumed to be separable. That is, the system is assumed to have its linear and nonlinear sections connected in parallel. This rules out nonlinear systems of the Wiener and Hammerstein types, however, the parametric Volterra structure is general enough to include most nonlinear systems. The continuous time Volterra Series is an infinite series which can be made to approximate almost any system, no matter how nonlinear it is. Since the Quadratic Volterra Adaptive Filter, (2.2), is a discrete representation of the continuous time Volterra Series, of order 2, it is quite possible to model Wiener and Hammerstein systems.

The parametric description need not be exact and any preliminary parametric model, including state space models, can be used to ascertain a sufficient measure of the level of nonlinearity present. For the present we have elected to structure the quadratic Volterra model using a Finite Impulse Response or transversal filter purely because of its popularity in signal processing. Even if the actual system were not of this type our test will still give an accurate enough result to allow the user to decide better whether to continue modelling with a nonlinear adaptive filter or switch to a linear adaptive filter.

The technique is to estimate and compare the powers of the signals emerging from the linear and nonlinear sections of the system. This estimate of the level of nonlinearity present in the system is named the Simple nonlinearity Measure and defined as

$$\begin{aligned} \text{SIM (dB)} &= [\text{NonLinear signal power}] \\ &\quad - [\text{Linear signal power}] \\ &= 10 \log_{10} [NLP^2] - 10 \log_{10} [LP^2] \end{aligned} \quad (3.1)$$

The experimental set-up to calculate SIM is indicated in Figure 3.1. A consequence of the definition in (3.1) is that the overall system must be separable.

From the definition given in (3.1) it is seen that if SIM takes on a large negative value then the system is described as being weakly nonlinear. That is, the power of the nonlinear section is small in

comparison to that of the linear section. Conversely, if SIM is large and positive then the system may be described as being strongly nonlinear.

For the system (2.1) the value of SIM as the magnitude of the nonlinear coefficient varies is illustrated graphically in Figure 3.2. Notice that the input signal is white Gaussian (0,1) in nature, however, other input signals can also be used.

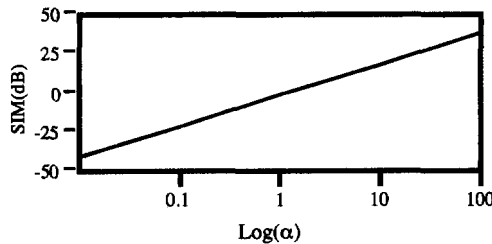


Figure 3.2 Plot of Levels of Nonlinearity for the system defined in (2.1)

At values of α less than unity SIM is seen to be large and negative, approaching 0dB as α passes 1. Hence the system may be described as being weakly nonlinear. Indeed the results given in Figure 2.1-2.4, show that at these values of SIM the linear LMS and NLMS algorithms can still achieve convergence. However, as α increases SIM increases rapidly and from the results given in Section 2 it is seen that the linear adaptation algorithms are no longer effective. The plot in Figure 3.2 is seen to be linear, this is a consequence of the quadratic nonlinearity, (2.1), that is used to perform the simulations. The squaring operation in (2.1) effectively becomes linear when the logarithm operator is applied, and the linear nature of the plot is further emphasised by plotting the logarithm of α on the horizontal axis.

Our results indicate that a SIM value of 0dB would be a reasonable centre point. That is, as SIM becomes progressively positive the system may be described as becoming more strongly nonlinear. Similarly as SIM decreases from the 0dB point the system may be described as becoming more weakly nonlinear. The idea is depicted in Figure 3.3.

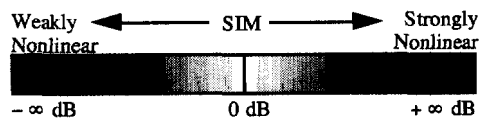


Figure 3.3 System Nonlinearity according to SIM value

As a bonus we have found additional uses for our measure which will also be of some practical use. A glance through the nonlinear literature will show a variety of algorithms in use. Some are seen to have rapid convergence and approach steady state in a matter of a few hundred iterations, while others are comparatively slow and do not reach steady state until after thousands of

iterations. Using our measure it can be shown conclusively that the reason for this discrepancy lies at least in part in the systems being identified. Those examples which are seen to convergence quickly were found to have low levels of nonlinearity while the slow responsive algorithms were found to be identifying systems with high levels of nonlinearity. The implications of this result are far reaching because it means that it would be possible to perform better comparisons of algorithms performances. In other words, once a level of nonlinearity is decided upon it can be used as a benchmark figure to compare other existing and new algorithm performances.

3.2 Applications of SIM

To demonstrate the use of the nonlinearity measure we have selected two (quadratic) nonlinear systems from the open literature with which to test SIM. The systems are taken from references [1] and [3], and have the following SIM values.

$$\text{System [1] : SIM} = -4.94\text{dB} \quad (3.2)$$

$$\text{System [3] : SIM} = 10.37\text{dB} \quad (3.3)$$

Comparing these values with Figure 3.3 and our earlier discussion we have the following. The power of the linear section in System [1] is of greater magnitude than that of its nonlinear section. In System [3] the opposite is the case, now the nonlinear section power is greater than the linear section power. On the SIM scale, Figure 3.3, these values indicate that System [1] may be amenable to linear approximation, but System [3] is unlikely to respond to linear adaptation alone.

To test these assertions we have obtained the learning curves for System [1] and [3], indicated in Figure 3.4 and 3.5. As previously the input signal was white Gaussian (0,1); the optimum step-sizes were used in all cases. For lack of space only the Power NLMS performance is shown.

From Figure 3.4(a) it can be seen that the nonlinear adaptive filter attains steady state in around 350 iterations. The linear adaptive filter performance shown in Figure 3.4(b) indicates an initial decrease in mse during the first 1000 iterations, after which it remains at a mse level of 2dB. In Figure 3.5(a) the nonlinear adaptive filter learning curve for System [3] is shown. As in the previous case there is a sharp decrease in the mse before steady state is reached in under 3000 iterations. The linear adaptive filter learning curve, Figure 3.5(b), shows no visible signs of convergence, the mse is constant at 19dB throughout. Altering the step-size and filter length of the linear adaptive filter did not result in any noteworthy improvements in performance of either System [1] or [3].

These results would appear to backup our earlier assertions. That is,

(a) $\text{SIM}(\text{System [3]}) > \text{SIM}(\text{System [1]})$: From Figure 3.4(a) and 3.5(a) it is seen that the learning curve for System [3] takes almost 9 times longer to reach steady

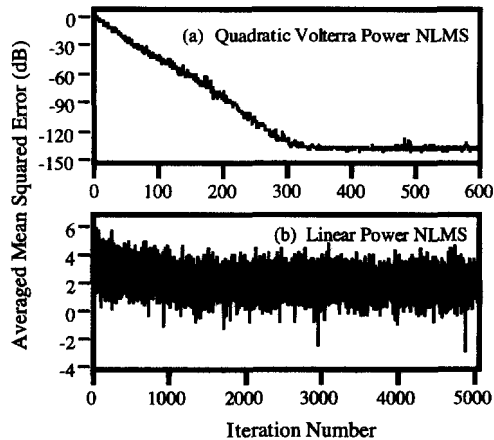


Figure 3.4 Nonlinear and linear learning curves for System[1]

state. Since the level of nonlinearity is greater in System [3] it would reasonably be expected to have longer convergence times.

(b) $\text{SIM}(\text{System [3]}) > 0\text{dB}$. $\text{SIM}(\text{System [1]}) < 0\text{dB}$: The linear learning curves in Figure 3.4(b) and 3.5(b) show that while the linear adaptive filter in System [1] is to a small extent still able to influence the adaptation, the linear adaptive filter in System [3] has no affect. Moreover the final mse in Figure 3.4(b) is 17dB better than that in Figure 3.5(b).

These results would also seem to indicate that our choice of 0dB as the centre point of the SIM scale is appropriate. As has been shown, once a system is found to have a SIM value in excess of 0dB little should be expected in the way of performance from a linear adaptive filter alone. Conversely, if a system is found to have a SIM value below the centre point then a linear adaptive filter may be worth investigating. In the case of System [1] employed here, which had a SIM value of only 5dB below the centre point, convergence was small but clearly visible with the linear adaptive filter. If the SIM value were to decrease further then the influence of the linear routine would increase accordingly, as was found to be the case in section 2.2.

4. CONCLUSIONS

In this paper we have presented a new tool for the study of nonlinear systems. We began by recalling some of the commonest stochastic gradient adaptation algorithms and made some comments on their suitability. Then the effects of a squaring nonlinearity on the performance of the linear and nonlinear adaptive filter were shown. It was shown how such an apparently simple nonlinear function could have devastating effects on the performance of a linear adaptive filter. Next we addressed the important issue of measuring the amount of nonlinearity in a system. A new estimator, the Simple nonlinearity Measure, was then outlined. Two examples from the

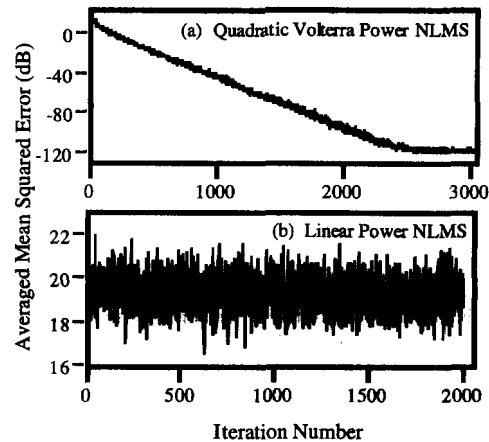


Figure 3.5 Nonlinear and linear learning curves for System[3]

open literature were used to demonstrate the SIM. These results indicate that our assumptions in its derivation are valid and the measure does indeed predict well the performance of linear and nonlinear adaptive filters in the presence of nonlinear signals. In particular using this measure it may be possible to do away with the need for testing a system with both a linear and nonlinear adaptation routine just to see which performs better.

The testing of different adaptation algorithms is also likely to be much improved and made more reliable using the SIM. Any system can be employed in the testing so long as the SIM value is constant. This will ensure a sense of consistency in the tests.

5. REFERENCES

1. Davila, C.E., A.J. Welch and H.G. Rylander, "A Second-Order Adaptive Volterra Filter with Rapid Convergence", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol.ASSP-35, No.4, PP.1259-1263, September 1987.
2. Haykin, S., *Adaptive Filter Theory*, 2nd Ed., Prentice-Hall, 1991.
3. John Mathews, V., "Adaptive Polynomial Filters", IEEE Signal Processing Magazine, PP.10-26, July 1991.
4. Raghuveer, M. R. and C. L. Nikias, "Bispectrum Estimation: A Parametric Approach", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol.ASSP-33, No.4, PP.1213-1230, October 1985.
5. Treichler, J.R., C.R. Johnson and M.G. Larimore, *Theory and Design of Adaptive Filters*, John Wiley, 1987.
6. Widrow, B. and S.D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, 1985.