On The Time Constant Under General Error Criterion

Ewaldo Santana, Allan Kardec Barros, *Member, IEEE*, and R. C. S. Freire, *Member, IEEE*

***Abstract*—Time constant along with misadjustment offers a manner of analyzing the convergence behavior of adaptive algorithms. In particular, there are some advantages of using nonlinear functions of the error instead of linear ones to have enhanced convergence behavior. However, some equations for the time constant suggested in the literature are noise dependent, yielding an infinite value for the noiseless case, which is obviously wrong. This problem may explain the fact that no works compared the time constants theoretically found to those derived in practice. In this letter, we derive a new time constant which depends on both the inputs and the noise. The results show that the found equation conforms to practical results.**

***Index Terms*—Adaptive systems and adaptive filtering, nonlinear error, time constant.**

I. Introduction

A NUMBER of works, in adaptive filtering, have been developed using the squared error as a cost function. Some well-known algorithms, such as the Kalman Filter, recursive least square (RLS), or least mean square (LMS) are based on that cost function. Maybe due to computational complexity, little attention was given to nonlinear functions of the error in adaptive filtering. However, the exploration of their properties has led to important findings in this area. For example, the LMS algorithm is limited to a hard tradeoff between the final misadjustment error and the convergence time, a fact which has forced some researchers to resort to rather computationally expensive methods [2], [6]. Thus, by simply using nonlinear functions of the error, some works have shown that they yield overall enhanced performance for the algorithm.

In either case of linear or nonlinear functions, the usual analyzed adaptation parameters were time constant and misadjustment. In the latter, the theoretical misadjustment conforms to the empirical one [5]. However, there are no results comparing the theoretical time constant to the practical one. It may be due to the fact that the derived equations yield infinite values from noiseless system [2], [3], [5]. Indeed, in some cases of noiseless systems, the time constant found in those works would be infinite.

In this work, we focus specifically on nonlinear even functions which can be expressed into Taylor series as a combination of the error to even exponents. This is equivalent to adding the even moments of the error, as shown by Barros *et al.* [3]. The advantage of those kinds of surfaces when compared to the squared one is that they naturally yield faster convergence with lower misadjustment [3], [5], [6]. Indeed, it is easily seen that the shape of the performance surface depends on the used criterion. As criteria are functions of the least-mean error model in which the performance surface shapes depend only on the input signal [4], we can guess that the shape for nonlinear functions shall also depend on the input signal. Indeed, the principal axes of the performance surface contours for the squared error corresponds to the eigenvectors of the input correlation matrix. Moreover, the corresponding eigenvalues determine the rate of change of the gradient along the principal axes of the surface contours [4] and, therefore, shall affect the convergence time.

Manuscript received September 12, 2006; revised November 30, 2006. This work was supported by FINEP. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Stefano Galli.

E. Santana and R. C. S. Freire are with the Federal University of Campina Grande, Campina Grande 58109-970, Brazil (e-mail: ewaldo@fama.br; rcsfreire@dee.ufcg.edu.br).

A. K. Barros is with the Federal University of Maranhão, Campus do Bacanga, s/n, São Luís 65080-040, Brazil (e-mail: allan@ufma.br).

Digital Object Identifier 10.1109/LSP.2007.894971

II. Methods

Let us consider that we observe a signal d_k, named measured signal, and a number of others which are included into a vector $\mathbf{X}_k = [x_{k1}, x_{k2}, \ldots, x_{kM}]$, called reference input. We say that the measured signal d_k is composed of the signal we want to extract s_k, added to a noise n_k in the form $d_k = s_k + n_k$. Let us make the following assumptions.

- Each input data vector $\mathbf{X}_k$ is statistically independent of all previous data vectors $\mathbf{X}_j$, $j < k$.
- n_k is white noise statistically independent of $\mathbf{X}_k$.
- All variables have probability distributions which are not necessarily Gaussian.
- The weight vector coefficients are statistically independent of the reference input.

The filtering task is accomplished by changing the weights of the filter which are given by $\mathbf{W}_k = [w_{k1}, w_{k2}, \ldots, w_{kM}]$. The current error is given by $\varepsilon_k = d_k - y_k$, and the output signal by $y_k = \mathbf{W}_k^T \mathbf{X}_k$. We define a cost function $F(\varepsilon_k)$ which is an even continuous function acting upon the error. Moreover, we assume that $F'''(0) = 0$.[1]

In gradient-based adaptive filtering, the algorithms for updating the weights are generally in the form

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu f(\varepsilon_k)\mathbf{X}_k \tag{1}$$

where $f(\varepsilon_k)\mathbf{X}_k$ is the gradient of the cost function $F(\varepsilon)$ with respect to the weights $\mathbf{W}_k$ and μ is a step-size parameter controlling the stability and speed of convergence.

A. Convergence Behavior

Let $\mathbf{V}_k = \mathbf{W}_k - \mathbf{W}^*$ be a weight deviation vector, where $\mathbf{W}^*$ is the optimal weight vector which makes $\varepsilon_k = n_k$. Thus, $\varepsilon_k = n_k - \mathbf{V}_k^T \mathbf{X}_k$. Then, we can rewrite (1) as

$$\mathbf{V}_{k+1} = \mathbf{V}_k + \mu f\left(n_k - \mathbf{V}_k^T \mathbf{X}_k\right)\mathbf{X}_k. \tag{2}$$

[1]There are many functions which satisfy this assumption. Examples are $\ln(\cosh(\varepsilon))$, ε^4, or any linear combination of ε^{2m}, $\forall\, m = 1, 2, 3, \ldots$.

Expressing the nonlinearity $f(\varepsilon_k)$ into a Taylor series expansion about the value $-\mathbf{V}_k^T\mathbf{X}_k$, we obtain

$$f\left(n_k-\mathbf{V}_k^T\mathbf{X}_k\right)=\sum_{i=0}^{\infty}\frac{f^{(i)}\left(-\mathbf{V}_k^T\mathbf{X}_k\right)}{i!}n_k^i \approx f\left(-\mathbf{V}_k^T\mathbf{X}_k\right)+f^{(1)}\left(-\mathbf{V}_k^T\mathbf{X}_k\right)n_k+\frac{1}{2}f^{(2)}\left(-\mathbf{V}_k^T\mathbf{X}_k\right)n_k^2 \quad (3)$$

where $f^{(i)}$ denotes the ith derivative of the function f.

Putting (3) into (2) and applying the expectation on both sides, we obtain

$$E[\mathbf{V}_{k+1}]\approx E[\mathbf{V}_k]+\mu\left\{E\left[f\left(-\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k\right]+\frac{1}{2}E\left[f^{(2)}\left(-\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k\right]\sigma_n^2\right\} \quad (4)$$

in which $\sigma_n^2=E[n_k^2]$ is the noise variance.

Now, let us state the following theorem:

Theorem: Let f be an odd nonlinearity defined, continuous, and whose derivatives $f^{(1)}, f^{(2)},\ldots$ exist in the interval $[-\delta,\delta]$. Moreover, assume that $f^{(2)}(0)=0$. Thus, $E[f(-\mathbf{V}_k^T\mathbf{X}_k)\mathbf{X}_k]\approx -E[f(\mathbf{X}_k\mathbf{X}_k^T)\mathbf{V}_k]$.

Proof: First, note that $f(0)=0$. Let us now express $f(-\mathbf{V}_k^T\mathbf{X}_k)\mathbf{X}_k$ into a Taylor series expansion about the value zero

$$f\left(-\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k\approx f(0)\mathbf{X}_k+f^{(1)}(0)\left(-\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k+\frac{1}{2}f^{(2)}(0)\left(\mathbf{V}_k^T\mathbf{X}_k\right)\left(\mathbf{X}_k^T\mathbf{V}_k\right)\mathbf{X}_k. \quad (5)$$

Taking expectations of both sides and invoking our initial assumptions, we see that

$$E\left[f\left(-\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k\right]\approx -f^{(1)}(0)E\left[\left(\mathbf{V}_k^T\mathbf{X}_k\right)\mathbf{X}_k\right]. \quad (6)$$

In a similar way, we can express $f(\mathbf{X}_k\mathbf{X}_k^T)\mathbf{V}_k$ into a Taylor series about the value zero

$$f\left(\mathbf{X}_k\mathbf{X}_k^T\right)\mathbf{V}_k\approx f(0)\mathbf{V}_k+f^{(1)}(0)\left(\mathbf{X}_k\mathbf{X}_k^T\right)\mathbf{V}_k+\frac{1}{2}f^{(2)}(0)\left(\mathbf{X}_k\mathbf{X}_k^T\right)\left(\mathbf{X}_k\mathbf{X}_k^T\right)\mathbf{V}_k. \quad (7)$$

Again, taking the expectations of both sides and invoking our assumptions, we have

$$E\left[f\left(\mathbf{X}_k\mathbf{X}_k^T\right)\mathbf{V}_k\right]\approx f^{(1)}(0)E\left[\left(\mathbf{X}_k\mathbf{X}_k^T\right)\mathbf{V}_k\right]. \quad (8)$$

Remembering that $(\mathbf{V}_k^T\mathbf{X}_k)\mathbf{X}_k=(\mathbf{X}_k\mathbf{X}_k^T)\mathbf{V}_k$, the theorem is proven. ♢

Then, (4) can be rewritten as

$$E[\mathbf{V}_{k+1}]\approx E[\mathbf{V}_k]-\mu\left(f(\mathbf{R})E[\mathbf{V}_k]+\frac{1}{2}f^{(2)}(\mathbf{R})E[\mathbf{V}_k]\sigma_n^2\right)\approx\left\{\mathbf{I}-\mu\left(f(\mathbf{R})+\frac{1}{2}f^{(2)}(\mathbf{R})\sigma_n^2\right)\right\}E[\mathbf{V}_k] \quad (9)$$

where $\mathbf{R}=E[\mathbf{X}_k\mathbf{X}_k^T]$.

This equation can be used to determine the algorithm convergence condition. Therefore, let us define the eigenvectors and eigenvalues matrices as $\mathbf{Q}$ and $\mathbf{\Lambda}$, respectively. Thus, we have $\mathbf{R}=\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{T}}$.

Now let $\tilde{\mathbf{V}}=\mathbf{Q}^{-1}\mathbf{V}$ represent a rotation on weight vectors $\mathbf{V}$. Using similar reasoning as in the proof of the theorem, we can expand both $\mathbf{Q}f(\mathbf{\Lambda})\mathbf{Q}^{\mathbf{T}}$ and $f(\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{T}})$ in Taylor series. Disregarding the higher order terms, we can see that $f(\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathbf{T}})\approx\mathbf{Q}f(\mathbf{\Lambda})\mathbf{Q}^{\mathbf{T}}$. Then, we can rewrite (9) as follows:

$$E[\tilde{\mathbf{V}}_{k+1}]\approx\left(\mathbf{I}-\mu f(\mathbf{\Lambda})-\frac{\mu}{2}f^{(2)}(\mathbf{\Lambda})\sigma_{\mathrm{n}}^2\right)E[\tilde{\mathbf{V}}_{\mathrm{k}}] \quad (10)$$

which can be easily solved by induction. Starting with the initial guess $\tilde{\mathbf{V}}_0$, we obtain

$$E[\tilde{\mathbf{V}}_k]=\left(\mathbf{I}-\mu f(\mathbf{\Lambda})-\frac{\mu}{2}f^{(2)}(\mathbf{\Lambda})\sigma_{\mathrm{n}}^2\right)^k\tilde{\mathbf{V}}_0. \quad (11)$$

Thus, as k increases, we see that the expected weight vector in (11) reaches the optimum solution (i.e., zero in the $\tilde{\mathbf{V}}$-axis system) only if the right side of the equation converges to zero [1]. This is satisfied by choosing μ so that

$$0<\mu<\frac{2}{f(\lambda_{\max})+\frac{1}{2}f^{(2)}(\lambda_{\max})\sigma_n^2} \quad (12)$$

where $\lambda_{\max}$ is the maximum eigenvalue of $\mathbf{R}$.

Now we can determine the time constant associated with the ith eigenvalue of $\mathbf{R}$.

Since the product of two diagonal matrices is just the matrix of products of the corresponding elements, the term between parentheses in (11) is a diagonal matrix, whose diagonal elements are given by

$$\tilde{v}_i=\left(1-\mu f(\lambda_i)-\frac{\mu}{2}f^{(2)}(\lambda_i)\sigma_n^2\right)^k\tilde{v}_0 \quad (13)$$

for $i=0,\ldots,L$.

Following the steps of Widrow and Stearns [1], we define

$$r_i\triangleq 1-\mu f(\lambda_i)-\frac{\mu}{2}f^{(2)}(\lambda_i)\sigma_n^2 \quad (14)$$

as the rate of the geometric sequence of samples. If one unit of time corresponds to one iteration, we obtain

$$\tau_i\approx\frac{1}{\mu\left(f(\lambda_i)+\frac{1}{2}f^{(2)}(\lambda_i)\sigma_n^2\right)}. \quad (15)$$

When the signal-to-noise ratio (SNR) is sufficiently high to make $f(\lambda_i)\gg(1/2)f^{(2)}(\lambda_i)\sigma_n^2$, we can neglect the second factor in the denominator. Then we have

$$\tau_i\approx\frac{1}{\mu f(\lambda_i)}. \quad (16)$$

Equation (16) leads us a surprising result: the time constant is not influenced by noise at high SNR. It depends only on the performance surface characteristics.

III. Results

In order to determine the accuracy of the derived equations, we carried out different simulations, where we examined the actual time constant, the time constant proposed by us, and those proposed in the literature. To highlight the results, here we show

TABLE I
THEORETICAL AND EMPIRICAL TIME CONSTANTS FOUND FOR DIFFERENT VALUES OF θ. THE LEFT COLUMNS SHOW THE EMPIRICAL VALUES OF ACTUAL τ FOR THREE ALGORITHMS. THE MIDDLE COLUMNS SHOW τ CALCULATED USING (16) FOR EACH ALGORITHM. THE RIGHT COLUMNS SHOW τ CALCULATED BY THE EQUATION PROPOSED AT THE RESPECTIVE WORKS

	τ empirical			τ calculated by our proposal			τ calculated as proposed in [2], [6], [3]		
θ	LMF	LMMN	WEM	LMF	LMMN	WEM	LMF	LMMN	WEM
0.10	860	277	35	705	262	60	518260	761	77
0.09	827	308	32	700	260	64	631950	760	81
0.08	742	276	33	699	265	62	797700	766	78
0.07	761	294	28	702	258	59	1017600	758	76
0.06	748	222	29	713	261	68	1477300	763	84
0.05	744	273	29	709	261	55	1037263	763	73
0.04	711	252	26	704	252	58	3182500	762	74

a simulation where a five tap-delay line adaptive model was used with coefficients [0.0000, 0.2037, 0.5926, 0.2037, 0.0000]. The input signal was a uniform random signal bounded between $[-1, 1]$ and the noise was simulated as a Gaussian signal interference with zero mean and unity variance at different levels. Let us write the measured signal as

$$d_k = s_k + \theta n_k \tag{17}$$

where θ took the following values [0.04, 0.05,0.06, 0.07, 0.08, 0.09, 0.10].

To validate our results, we used three algorithms already proposed in the literature: 1) the least mean fourth (LMF) [2], in which the cost function is $F(\varepsilon) = \varepsilon^4$; 2) the least mean mixed-norm (LMMN) [6], whose cost function is given by $F(\varepsilon) = \vartheta\varepsilon^2 + (1-\vartheta)\varepsilon^4$, where $\vartheta \in [0, 1]$ is the mixing parameter; and 3) the weighted even moment (WEM) [3], which uses $F(\varepsilon) = \sum_{K=1}^{p} K^{p-1}\varepsilon^{2K}$ as the cost function, where K and p are positive integers. For each algorithm and for each noise level, 100 Monte Carlo runs were performed using exactly the same data. We calculated the actual time constant as the time which the error took to reach 36% $(1/e)$ of its initial value. For each algorithm, we found the learning rate which guaranteed convergence. For the LMF, we used $\mu = 0.01$. For the LMMN, we used $\mu = 0.025$. For the WEM, we used $\mu = 0.02$. In Fig. 1, we plotted the ensemble averaged error versus the number of iterations in a trial where the LMF algorithm was used. Moreover, for the sake of clarity, we show the exponential decay $\exp(-t/\tau)$ for two derived equations. A straight line denoting 36% $(1/e)$ of the initial error value is also plotted. We show the results in Table I, where we can see, in the first column, several values for the parameter θ, which is used in (17), denoting the noise contribution. The following nine columns are divided into three groups of three columns. In each group, the first column represents LMF, the second columns represents LMMN, and the third column represents WEM. The leftmost group contains empirical time constants values. The middle group contains the time constants calculated by our proposal. In the right group, we have the time constants calculated by using the equations proposed in the respective works.

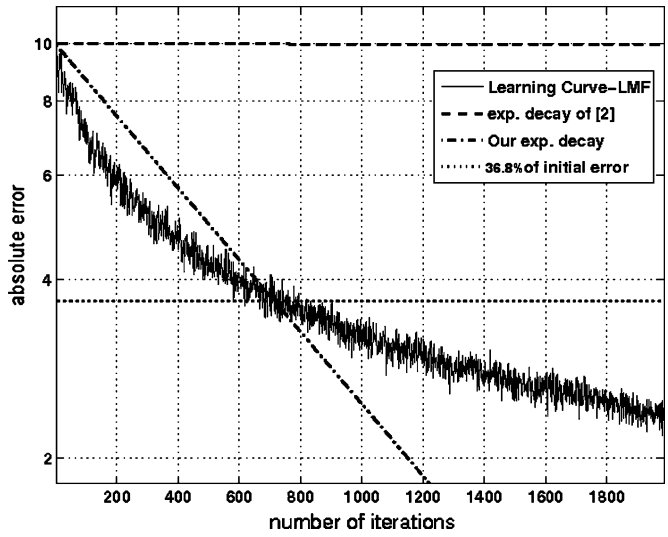

Fig. 1. Typical example of convergence. The straight line denotes the learning curve; the dotted line represents 36.8% of the initial value of the learning curve; the dashed line indicates the exponential decay proposed in [2], and the dash–dotted line represents our proposal to the exponential decay.

IV. DISCUSSIONS AND CONCLUSION

From (15) and (16), we can easily see that the time constant is less influenced by the noise than by characteristics of the performance surface at high SNR. This result was not encountered before in the literature. For example, for LMF, the authors [2] proposed $\tau_i = 1/(12\mu\lambda_i\sigma_n^2)$, which yields high dependence on noise. Moreover, it means that in the noiseless case, this equation yields an infinite value, which is obviously wrong. In Fig. 1, where we use $\theta = 0.06$, we have an inclined (dash-dotted) line representing our proposal for the exponential decay. Notice that this line crosses the learning curve approximately at the same time that the line which represents 36.8% of the initial value, while a horizontal (dashed) line representing the exponential decay proposed by Walach and Widrow clearly shows that it is of very limited accuracy. In Table I, we can verify for the LMF algorithm that for the same θ, the values obtained in the middle column conform with the corresponding measured values in the left column, while we have quite discrepant values in the right column. For the LMMN algorithm, by using (16), we have also an agreement between the values of the left columns and the middle one. Still for the LMMN, despite the values in the right column not being so different when compared with corresponding values at the left column, they are still more than twofold larger. For the WEM algorithm, we did not find significant differences between the values in the middle column and the ones in the right column. Moreover, in the left column, we can notice less influence of the noise for values in each particular algorithm column. This agrees with our conclusion about

the lack of influence of the noise in an adaptive filtering process at high SNR.

REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[2] E. Walach and B. Widrow, "The least mean fourth (LMF) adaptive algorithm and its family," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 2, pp. 603–608, May 1988.

[3] A. K. Barros, J. Principe, Y. Takeuchi, C. H. Sales, and N. Ohnishi, "An algorithm based on the even moments of the error," in *Proc. 8th Workshop on Neural Networks for Signal Processing*, Toulouse, France, 2003, pp. 879–885.

[4] J. C. Principe, N. R. Euliano, and W. C. Lefebvre, *Neural and Adaptive Systems: Fundamentals Through Simulations*. New York: Wiley, 2000.

[5] S. C. Douglas and T. Meng, "Stochastic gradient adaptation under general error criteria," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1335–1351, Jun. 1994.

[6] J. A. Chambers, O. Tanrikulu, and A. G. Constantinides, "Least mean mixed-norm adaptive filtering," *Electron. Lett.*, vol. 30, no. 19, pp. 1574–1575, 1994.