

Enhanced-Convergence Normalized LMS Algorithm

"DSP Tips and Tricks" introduces practical design and implementation signal processing algorithms that you may wish to incorporate into your designs. We welcome readers to submit their contributions to the Associate Editors, Rick Lyons (r.lyons@ieee.org) or Britt Rorabaugh (dspboss@aol.com).

Least mean square (LMS) algorithms have found great utility in many adaptive filtering applications. This article shows how the traditional constraints placed on the update gain of normalized LMS algorithms are overly restrictive. We present relaxed update gain constraints that significantly improve normalized LMS algorithm convergence speed.

BACKGROUND

The scalar form of the LMS algorithm is well known as

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mu e(k)\mathbf{x}(k) \quad (1)$$

with μ = scalar update gain, $e(k)$ the error at time k , and $\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-N+1)]^T$ an N -dimensional column vector at time k . The scalar value for μ results in a constant, uniform update gain. The

update gain, μ , can be replaced with the matrix

$$\mathbf{M} = \begin{bmatrix} \mu & 0 \\ 0 & \mu \end{bmatrix}, \quad (2)$$

resulting in an LMS algorithm defined as

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \mathbf{M}e(k)\mathbf{x}(k). \quad (3)$$

The update gain is constant and uniform across all coefficients of the vector $\mathbf{h}(k)$.

An example of a variable step size with a uniform update gain is the normalized LMS algorithm (NLMS), defined as

$$\mathbf{h}(k+1) = \mathbf{h}(k) + \frac{\mu e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k)}, \quad (4)$$

where the effective update gain is $\mu/\mathbf{x}^T(k)\mathbf{x}(k)$ and it varies with σ_x^2 , the variance of the input. Although the update gain is variable, it is still uniform across all coefficients of \mathbf{h} .

In some research [1]–[3], the update gain matrix \mathbf{M} has been replaced by

$$\mathbf{M} = \begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_N \end{bmatrix}, \quad (5)$$

with each μ_m ($1 \leq m \leq N$) not necessarily equal, to form a nonuniform gain matrix. Some performance characteristics for this form have been published [1]. The upper bound for μ_m of \mathbf{M} has been stated, generally, as $\mu_m \leq 2/\lambda_{\max}[4]$ or $\mu_m \leq 2/3 \cdot \text{tr}[\mathbf{R}]$ [5], [6]. While these are upper bounds, they are not, in general, the largest upper

bounds that maintain stability of the algorithm.

Numerous researchers have sought improvement in convergence speed for the NLMS algorithm. Each has been limited, however, to the criterion that the maximum limit for the update gain is either $\mu_m \leq 2/\lambda_{\max}$ or $\mu_m \leq 2/3 \cdot \text{tr}[\mathbf{R}]$. This article uses an update gain greater than the accepted maximum by limiting the sum of μ_m of matrix \mathbf{M} to the number of coefficients in the adaptive filter. We express our limitation as

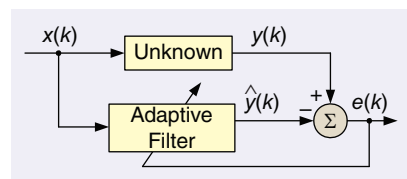
$$\sum_{m=1}^N \mu_m \leq N. \quad (6)$$

That is, if the adaptive filter has $N = 256$ coefficients, then the sum of μ_m is restricted to being no greater than 256. Our restriction in (6) does two things:

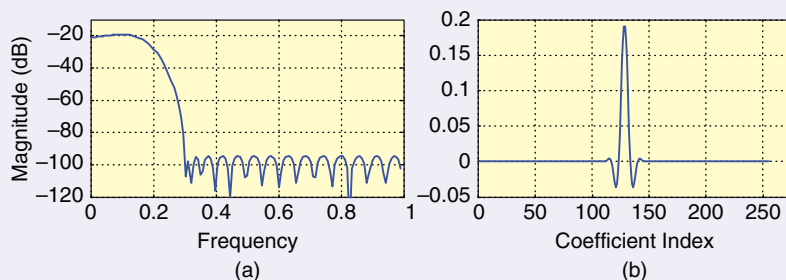
- 1) It increases the update gain, subject to this limitation, during convergence for those coefficients that are coincident with the nonzero (or relatively large) portion of the amplitude of the unknown impulse response, which we call active coefficients.
- 2) It makes those coefficients of the adaptive filter that are inactive small (or zero).

EXAMPLES

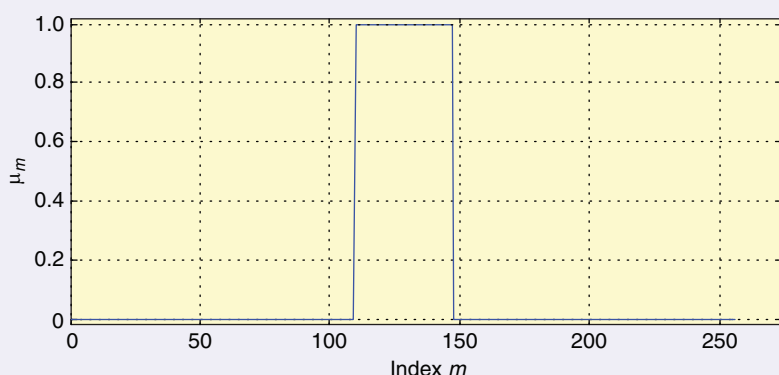
Let's use an example to illustrate our improved update gain constraint. Using the classic system identification configuration of Figure 1, the unknown system is modeled as a low-pass function whose power spectral density and impulse response are shown in Figure 2. The purpose of the system is to adjust the adaptive filter's coefficients to reduce the $e(k)$ error signal to zero, in which case the



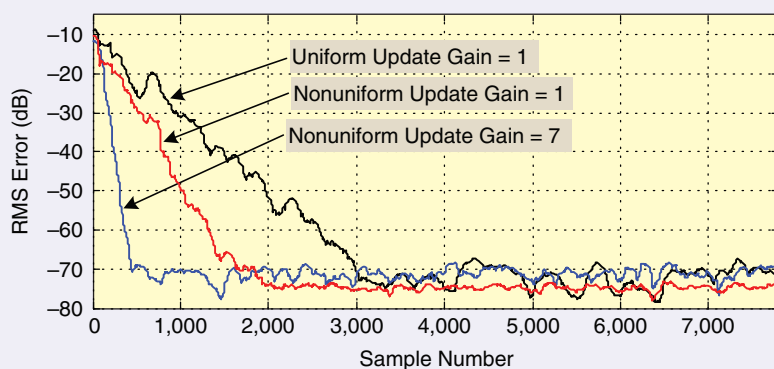
[FIG1] Classic system identification adaptive filter configuration.



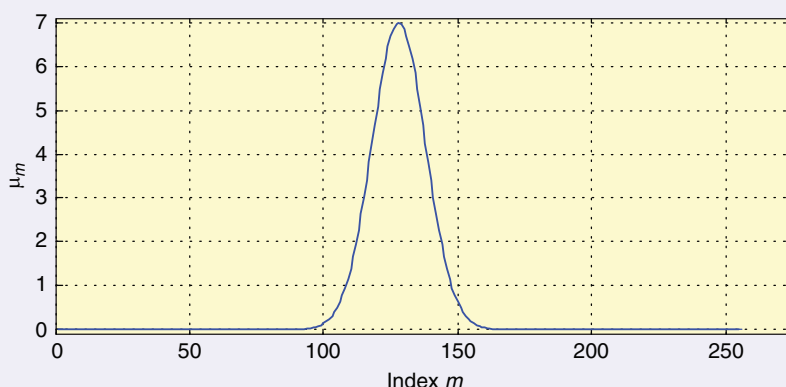
[FIG2] Unknown system: (a) power spectral density; (b) impulse response.



[FIG3] Nonuniform update gain = 1 matrix elements.



[FIG4] Performance comparison of filter update gain configurations.



[FIG5] Gaussian nonuniform update gain matrix elements.

adaptive filter's \mathbf{h} coefficients identify (match) the impulse response of the unknown system.

Three versions of Figure 1's adaptive filter, having $N = 256$ coefficients, were implemented. One adaptive filter is allowed to converge using a constant, uniform update gain of one, and the other two converged using non-uniform update gains. The nonuniform update gains were a constant value of seven for the active coefficients and zero for the inactive coefficients for the second adaptive filter. The values used for the third adaptive filter were one for the active coefficients and zero for the inactive coefficients, as shown in Figure 3. The comparative convergence of the three adaptive filters is shown in Figure 4.

For the second filter's nonuniform update gain of seven—very much higher than the traditionally accepted update gain upper bound—the algorithm remained stable, and the convergence speed is very much improved. Improved convergence performance can be realized if the update gain of the adaptive filter is increased in accordance with our specified limit in (6), instead of the traditional limits.

An additional example of nonuniform update gain matrix elements, μ_1 to μ_N , is shown in Figure 5. Those update gain elements are the result of the Gaussian equation

$$\mu_m = \frac{1}{\sigma\sqrt{2\pi}} \varepsilon^{-\frac{(m-\bar{m})^2}{2\sigma^2}}, \quad (7)$$

with \bar{m} set such that it is coincident with the middle of the nonzero (active) coefficients of the unknown impulse response in Figure 2(b). In (7), $\sigma = 10$ and, again, index m ranges from 1 to N , the number of coefficients in the adaptive filter. The long-term performance of the Gaussian update gain configuration indicates that it should be used for initial convergence, with reversion to a more traditional gain configuration after initial convergence.

(continued on page 95)

not only the largest one [7], [4]. This is naturally accomplished by many of the methods we discussed, by simply including the terms corresponding to all leaf nodes in the decision tree that the algorithm has visited. If the symbol vectors have different a priori probabilities, then under certain circumstances $P(s)$ can be incorporated by appropriately modifying y and H [6], [4].

CONCLUSIONS

The goal of this lecture has been to provide an overview of approaches to (2), in the communications receiver context. Which method is the best in practice? This depends much on the purpose of solving (2): what error rate can be tolerated, what is the ultimate measure of performance (e.g., frame-error-rate, worst-case complexity, or average complexity), and what computational platform is used. Additionally, the bits in s may be part of a larger code word and

different s vectors in that code word may either see the same H (slow fading) or many different realizations of H (fast fading). This complicates the picture, because notions that are important in slow fading (such as spatial diversity) are less important in fast fading, where diversity is provided anyway by time variations. Detection for MIMO has been an active field for more than ten years, and this research will probably continue for some time.

AUTHOR

Erik G. Larsson (erik.larsson@isy.liu.se) is a professor and head of division for communication systems in the EE department (ISY) of Linköping University, Sweden. For more information, visit www.commsys.isy.liu.se.

REFERENCES

- [1] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time

communications architecture," *IEE Electron. Lett.*, vol. 35, pp. 14–16, Jan. 1999.

- [2] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1639–1642, July 1999.

- [3] L. G. Barbero and J. S. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 2131–2142, June 2008.

- [4] E. G. Larsson and J. Jaldén, "Soft MIMO detection at fixed complexity via partial marginalization," *IEEE Trans. Signal Processing*, vol. 56, pp. 3397–3407, Aug. 2008.

- [5] P. Tan and L. Rasmussen, "The application of semidefinite programming for detection in CDMA," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 1442–1449, Aug. 2001.

- [6] B. Steingrimsson, Z.-Q. Luo, and K. Wong, "Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels," *IEEE Trans. Signal Processing*, vol. 51, no. 11, pp. 2710–2719, Nov. 2003.

- [7] B. M. Hochwald and S. Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

- [8] C. Windpassinger and R. Fischer, "Low-complexity near-maximum-likelihood detection and precoding for MIMO systems using lattice reduction," in *Proc. IEEE Information Theory Workshop*, 2003, pp. 345–348.

- [9] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2201–2214, Aug. 2002.

SP

[dsp **TIPS&TRICKS**] continued from page 82

AUTHOR

Maurice Givens (Maurice.Givens@gastech-nology.org) is a specialist in the research and design of digital signal processing at Gas Technology Institute and an adjunct lecturer at Wright College, Chicago. He is a registered professional engineer, a Senior Member of the IEEE, and a senior member of NARTE.

REFERENCES

- [1] R. Harris, D. Chabries, and P. Bishop, "A variable step (VS) adaptive filter algorithm," *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-34, pp. 309–316, Apr. 1986.
- [2] J. Evans, P. Xue, and B. Liu, "Analysis and implementation of variable step size adaptive algorithms," *IEEE Trans. Signal Processing*, vol. 41, pp. 2517–2535, Aug. 1993.
- [3] T. Haweel and P. Clarkson, "A class of order statistic LMS algorithm," *IEEE Trans. Signal Processing*, vol. 40, pp. 44–53, Jan. 1992.
- [4] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: Analysis and simulations," *IEEE Trans. Signal Processing*, vol. 45, pp. 631–639, Mar. 1997.
- [5] D. Pazaitis and A. Constantinides, "A novel kurtosis driven variable step-size adaptive algorithm," *IEEE Trans. Signal Processing*, vol. 47, pp. 864–872, Mar. 1999.
- [6] R. Kwong and E. Johnston, "A variable step size LMS algorithm," *IEEE Trans. Signal Processing*, vol. 40, pp. 1633–1642, July 1992.

SP

We want to hear from you!

Do you like what you're reading?
Let us know—send an e-mail to the Editor-in-Chief.
Letters may be published in future issues and edited for style.