A COMPLEMENTARY PAIR LMS ALGORITHM FOR ADAPTIVE FILTERING

Woo-Jin Song[1] *Min-Soo Park*[2]

[1]Department of Electrical Engineering, Pohang University of Science and Technology, Pohang, 790-784, Korea
[2]LG Information and Communications, Anyang, 430-080, Korea
park@aurora.postech.ac.kr

ABSTRACT

This paper presents a new algorithm that can solve the problem of selecting appropriate update step size in the LMS algorithm. The proposed algorithm, called a *Complementary Pair LMS (CP-LMS)* algorithm, consists of two adaptive filters with different update step sizes operating in parallel, one filter re-initializing the other with the better coefficient estimates whenever possible. This new algorithm provides the faster convergence speed and the smaller steady-state error than those of a single filter with a fixed or variable step size.

1. INTRODUCTION

In adaptive filtering, the LMS algorithm is very popular for its simplicity and predictable behavior, but the compromise must be made between the convergence (tracking) speed and the steady-state error. This is because the LMS algorithm updates the adaptive filter coefficients with a term whose magnitude is proportional to the so-called *step size* μ. To obtain the fast convergence speed, μ has to be relatively large but using a large μ produces a large steady-state error. To obtain the small steady-state error, μ has to be relatively small but using a small μ makes the convergence very slow [1].

To solve this problem, many variable step size algorithms [2] [3] [4] [5] [6] that try to achieve both the fast convergence speed and the small steady-state error have been developed. However, the performances of these algorithms are highly dependent on the algorithm parameters which are selected without specific rules.

To provide strictly controlled performance and to eliminate the conflict between the accuracy and the speed aspect, we introduce a new adaptive filtering algorithm called a CP-LMS (Complementary Pair LMS) algorithm. The new algorithm uses two adaptive filters operating in parallel with different step sizes. The estimation errors of two filters are compared, and the coefficients of one filter is used for re-initializing the other filter to speed up the convergence (tracking) speed.

2. ADAPTIVE FIR FILTERING

Let $x(n)$ be the input for a unknown system H, which is modeled by the filter coefficient b_k, $0 \le k \le K$. Then the output $y(n)$ of H is given by

$$y(n) = \sum_{k=0}^{K} b_k x(n-k) = \mathbf{x}^T(n)\mathbf{b} \tag{1}$$

where

$$\begin{aligned} \mathbf{x}^T(n) &= [\, x(n)\ x(n-1)\ \ldots\ x(n-K)\,] \\ \mathbf{b}^T &= [\, b_0\ b_1\ \ldots\ b_K] \,. \end{aligned}$$

Usually $y(n)$ is corrupted by zero-mean additive noise $v(n)$, so the observed output of H is given by

$$d(n) = y(n) + v(n)\ . \tag{2}$$

In the system identification configuration [1], the input to the adaptive FIR filter is $x(n)$ and the filter output $z(n)$ is

$$z(n) = \sum_{k=0}^{K} \hat{b}_k(n) x(n-k) = \mathbf{x}^T(n)\hat{\mathbf{b}}(n) \tag{3}$$

where

$$\begin{aligned} \mathbf{x}^T(n) &= [\, x(n)\ x(n-1)\ \ldots\ x(n-K)\,] \\ \hat{\mathbf{b}}^T(n) &= [\, \hat{b}_0(n)\ \hat{b}_1(n)\ \ldots\ \hat{b}_K(n)\,] \,. \end{aligned}$$

The adaptive filter coefficients $\hat{b}_k(n)$, $0 \le k \le K$ are estimated such that the difference between $d(n)$ and $z(n)$, defined as the estimation error

$$e(n) = d(n) - z(n)\ , \tag{4}$$

approaches zero.

The most popular adaptive FIR filtering algorithm is the LMS (Least Mean Square) algorithm [1], which uses the coefficient update equation :

$$\hat{\mathbf{b}}(n+1) = \hat{\mathbf{b}}(n) + \mu e(n)\mathbf{x}(n)\ . \tag{5}$$

3. COMPLEMENTARY PAIR LMS

A single adaptive filter with a fixed update step size μ poses the following problem. To obtain the fast convergence speed, μ has to be relatively large but using a large μ produces a large steady-state error. To obtain the small steady-state error, μ has to be relatively small but using a small μ makes the convergence very slow. To solve this problem, we employ two adaptive filters with different update step sizes operating in parallel, complementing each other.

Fig. 1 shows the block diagram of the CP-LMS algorithm. One filter which has the larger step size μ_s for the fast convergence speed is called the speed mode filter, whose coefficients are updated by

$$\hat{\mathbf{b}}_s(n+1) = \hat{\mathbf{b}}_s(n) + \mu_s e_s(n)\mathbf{x}(n) \tag{6}$$

where

$$e_s(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{b}}_s(n) \ . \tag{7}$$

The other filter which has the smaller step size μ_a for the small steady-state error is called the accuracy mode filter, whose coefficients are updated by

$$\hat{\mathbf{b}}_a(n+1) = \hat{\mathbf{b}}_a(n) + \mu_a e_a(n)\mathbf{x}(n) \tag{8}$$

where

$$e_a(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{b}}_a(n) \ . \tag{9}$$

The two filters operate in parallel, and $e_s(n)$ and $e_a(n)$ are supplied to the controller that re-initializes the accuracy mode filter.

This re-initialization controller replaces $\hat{\mathbf{b}}_a(n)$ with $\hat{\mathbf{b}}_s(n)$ for every M-th coefficient update, if the local average of $e_s^2(n)$ is less than the local average of $e_a^2(n)$ for J consecutive comparisons with interval length M. This re-initialization of the accuracy mode filter can be mathematically expressed as

$$\hat{\mathbf{b}}_a(n+1) = \begin{cases} \hat{\mathbf{b}}_s(n+1) & \\ \quad \text{if } n(\text{mod } M) = 0 \text{ and} \\ \quad \prod_{j=1}^{J} Q(n-jM) = 1 \\ \hat{\mathbf{b}}_a(n) + \mu_a e_a(n)\mathbf{x}(n) \\ \quad \text{otherwise} \end{cases} \tag{10}$$

where M is the comparison interval, J is the number of comparisons, and $Q(m)$ is a two-valued function defined as

$$Q(m) = \begin{cases} 1 & \text{if } \sum_{i=m}^{m+M} e_s^2(i) < \sum_{i=m}^{m+M} e_a^2(i) \\ 0 & \text{otherwise} \ . \end{cases} \tag{11}$$

Since $Q(m)$ is evaluated only once in every M updates and the past values of $Q(m)$ can be saved, the total computations needed for the CP-LMS is $4K+4$ multiplications and $4K+2$ additions per update.

The size of M should be sufficiently large, $M \gg 1$, so that the statistical average of $e_s^2(n)$ and $e_a^2(n)$ can be obtained. Also, M should be much smaller than N, $M \ll N$, where N is the length of the training input $x(n)$, so that a sufficient number of re-initialization is possible. Therefore, the comparison period M must satisfy the inequality

$$1 \ll M \ll N \ . \tag{12}$$

A simple choice for M is $M = \sqrt{N}$ which satisfies (12) and shows good performance in simulations.

The number J should be chosen according to the training input length and the noise levels. The total comparison length JM should be much smaller than N, $JM \ll N$, so that a prompt re-initialization is ensured. If we have chosen $M = \sqrt{N}$, then this leads to the condition $J \ll M$. For low SNR, we must have $J \gg 1$, so that the mistaken re-initialization, due to the noise signal, is avoided. Therefore, under low SNR, the number of comparison J must satisfy inequality

$$1 \ll J \ll M \ . \tag{13}$$

For high SNR, we can choose J close to 1 for more frequent re-initialization, since the probability of mistaken re-initialization due to the noise signal would be small. This means J only have to satisfy

$$1 < J \ll M \ . \tag{14}$$

The simulations show that, for SNR of 30 dB, the value $J = 3$ is found to be sufficient.

The re-initialization of the accuracy mode filter occurs when $\hat{\mathbf{b}}_s(n)$ approaches faster to the true value $\mathbf{b}$ than $\hat{\mathbf{b}}_a(n)$ due to the larger step size μ_s. Since the accuracy mode filter is re-initialized with better coefficient estimates whenever possible, the convergence can be reached in shorter time with the desired accuracy set by the smaller update step size μ_a.

The step size μ_s could be large as long as stable convergence is maintained. The upper bound of μ_s is given by a well-known inequality [7]

$$\mu_s < \frac{1}{trE\{\mathbf{x}(n)\mathbf{x}^T(n)\}} = \frac{1}{KE\{x^2(n)\}} \tag{15}$$

where K is the vector length of $\mathbf{x}(n)$.

The step size μ_a could be small as long as convergence can be reached within the given training signal length N. Since the total absolute sum of the expected update terms must satisfy

$$\sum_{n=0}^{N-1} |E\{\mu_a e_a(n)x(n-k)\}| \geq |b_k| \ , \tag{16}$$

we can write

$$N\left(\mu_a\sqrt{2E\{d^2(n)\}E\{x^2(n-k)\}}\right) \geq |b_k| \tag{17}$$

where we have used a cross-correlation property [8]

$$|E\{e_a(n)x(n-k)\}| \leq \sqrt{E\{e_a^2(n)\}E\{x^2(n-k)\}} \tag{18}$$

and an assumption $E\left\{e_a^2(n)\right\} \leq 2E\left\{d^2(n)\right\}$. If $|b_k| \leq 1$, then the sufficient condition for μ_a can be derived from (17), which is

$$\mu_a > \frac{1}{N\sqrt{2E\{d^2(n)\}E\{x^2(n)\}}} \ . \tag{19}$$

Combining (15) and (17), we get

$$\frac{1}{N\sqrt{2}\sigma_d\sigma_x} < \mu_a < \mu_s < \frac{1}{K\sigma_x^2} \tag{20}$$

where $\sigma_d^2 = E\{d^2(n)\}$ and $\sigma_x^2 = E\{x^2(n)\}$. However, μ_a must be much smaller than μ_s so that the accuracy mode filter have substantially smaller steady-state error than the speed mode filter, but using step sizes near the lower and the upper bound tend to produce unsatisfactory results. Therefore, more realistic condition for μ_a and μ_s is

$$\frac{1}{N\sqrt{2}\sigma_d\sigma_x} \ll \mu_a \ll \mu_s \ll \frac{1}{K\sigma_x^2} \; . \tag{21}$$

4. ALGORITHM BEHAVIOR

To examine the behavior of the CP-LMS algorithm, we obtain the estimation error vectors

$$\mathbf{p}_s(n) = \hat{\mathbf{b}}_s(n) - \mathbf{b} \tag{22}$$

$$\mathbf{p}_a(n) = \hat{\mathbf{b}}_a(n) - \mathbf{b} \tag{23}$$

from $\hat{\mathbf{b}}_s(n)$ and $\hat{\mathbf{b}}_a(n)$. Also, we assume that the training input signal and the coefficient estimates are uncorrelated.

Combining (1), (2), (6), (7), and (22), we can construct the recursive equation of $\mathbf{p}_s(n)$ such that

$$\mathbf{p}_s(n+1) = (\mathbf{I} - \mu_s\mathbf{x}(n)\mathbf{x}^T(n))\mathbf{p}_s(n) + \mu_s\mathbf{x}(n)v(n) \; . \tag{24}$$

Thus, the mean behavior of $\mathbf{p}_s(n)$ is described by

$$E\{\mathbf{p}_s(n+1)\} = (\mathbf{I} - \mu_s\mathbf{R})E\{\mathbf{p}_s(n)\} \; , \tag{25}$$

where $\mathbf{R} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$.

The mean behavior of $\mathbf{p}_a(n)$ is more complicated due to the re-initializations, and is described by

$$E\{\mathbf{p}_a(n+1)\} = \begin{cases} E\{\mathbf{p}_s(n+1)\} \\ \quad \text{if } n (\text{mod } M) = 0 \text{ and} \\ \quad \prod_{j=1}^{J} \bar{Q}(n - jM) = 1 \\ \\ (\mathbf{I} - \mu_a\mathbf{R})E\{\mathbf{p}_a(n)\} \\ \quad \text{otherwise} \end{cases} \tag{26}$$

where

$$\bar{Q}(m) = \begin{cases} 1 & \text{if } \sum_{i=m}^{m+M} E\{e_s^2(i) - e_a^2(i)\} < 0 \\ \\ 0 & \text{otherwise} \; . \end{cases} \tag{27}$$

To get some meaningful insight of the mean-square behavior, we assume $x(n)$ is a white gaussian signal, so that $\mathbf{R} = \sigma_x^2\mathbf{I}$. Then, using (24), we can calculate $E\{\mathbf{p}_s^T(n+1)\mathbf{p}_s(n+1)\}$, which reduces to

$$E\{\|\mathbf{p}_s(n+1)\|^2\} = \beta_s E\{\|\mathbf{p}_s(n)\|^2\} + K\mu_s^2\sigma_x^2\sigma_v^2 \tag{28}$$

where $\|\mathbf{p}_s(n)\|^2 = \mathbf{p}_s^T(n)\mathbf{p}_s(n), \sigma_v^2 = E\{v^2(n)\}$, and $\beta_s = 1 - 2\mu_s\sigma_x^2 + (K+2)\mu_s^2\sigma_x^4$.

Incorporating the re-initialization operations, the mean-square behavior of $\mathbf{p}_a(n)$ is described by

$$E\{\|\mathbf{p}_a(n+1)\|^2\} = \begin{cases} E\{\|\mathbf{p}_s(n+1)\|^2\} \\ \quad \text{if } n (\text{mod } M) = 0 \text{ and} \\ \quad \prod_{j=1}^{J} \bar{Q}(n - jM) = 1 \\ \\ \beta_a E\{\|\mathbf{p}_a(n)\|^2\} + K\mu_a^2\sigma_x^2\sigma_v^2 \\ \quad \text{otherwise} \end{cases} \tag{29}$$

where $\beta_a = 1 - 2\mu_a\sigma_x^2 + (K+2)\mu_a^2\sigma_x^4$.

From (25), (26), (28), and (29), we can extract the steady state properties of the CP-LMS algorithm, which are

$$E\{\mathbf{p}_s(\infty)\} = \mathbf{0} \tag{30}$$

$$E\{\mathbf{p}_a(\infty)\} = \mathbf{0} \tag{31}$$

$$E\{\|\mathbf{p}_s(\infty)\|^2\} = \frac{K\mu_s\sigma_v^2}{2 - (K+2)\mu_s\sigma_x^2} \tag{32}$$

$$E\{\|\mathbf{p}_a(\infty)\|^2\} = \frac{K\mu_a\sigma_v^2}{2 - (K+2)\mu_a\sigma_x^2} . \tag{33}$$

5. SIMULATIONS

The computer simulations are carried out to investigate the performance gains obtained by using the CP-LMS algorithm. The performance is measured with the *norm squared estimation error*

$$\|\mathbf{p}_a(n)\|^2 = \|\mathbf{b} - \hat{\mathbf{b}}_a(n)\|^2 \; . \tag{34}$$

The compared algorithms are the LMS algorithm with $\mu = 0.001$, the LMS algorithm with $\mu = 0.01$, the CP-LMS algorithm with $\mu_a = 0.001, \mu_s = 0.01, M = 100, J = 3$, and the variable step size (VSS) algorithm [4] which employs a variable step size $\mu(n)$ defined as

$$\mu(n+1) = \begin{cases} \mu_{max} & \text{if } \mu(n) > \mu_{max} \\ \mu_{min} & \text{if } \mu(n) < \mu_{min} \\ \alpha\mu(n) + \gamma e^2(n) & \text{otherwise} \end{cases} \tag{35}$$

where $\mu_{max} = 0.01, \mu_{min} = 0.001, \alpha = 0.97, \gamma = 0.00048$, and $e(n)$ is the estimation error [4].

The training input $x(n)$ of length $N = 10,000$ is a white gaussian pseudo-random sequence with $E\{x(n)\} = 0$ and $E\{x^2(n)\} = 1$. The additive noise $v(n)$ is also a white gaussian pseudo-random sequence with $E\{v(n)\} = 0$ and $E\{v^2(n)\} = 0.001$.

5.1. Stationary case

The estimation target system is given by the all-zero model

$$H_0(z) = 0.674(1 + 0.8z^{-2} + 0.6z^{-4} + 0.4z^{-6} + 0.2z^{-8}) \tag{36}$$

where the scale factor 0.674 in $H_0(z)$ makes the input power and the output power equal. Therefore, the SNR of the training output $d(n)$ is 30 dB.

Fig. 2 shows the simulation results, where "LMS-1" denote the LMS algorithm with $\mu = 0.001$ and "LMS-2" denote the LMS algorithm with $\mu = 0.01$. The averaged data was not used so that the fluctuations of the coefficient estimates can be observed, and the simulation data are reduced by a factor of 100 through subsampling for simpler plots.

As we can see, the CP-LMS algorithm achieves the best performance by converging to the smallest norm squared estimation error in the shortest time. The convergence speed is about 3 times faster than the slowest one, and the estimation accuracy is about 10 dB better than the worst one.

5.2. Non-stationary case

The time-varying estimation target system is given by the all-zero model

$$H_1(z) = 0.674(1+0.8z^{-2}+0.6z^{-4}+0.4z^{-6}+0.2z^{-8}) \quad (37)$$

for the interval $0 \leq n < 5,000$ and

$$H_2(z) = 0.674(1-0.8z^{-2}+0.6z^{-4}-0.4z^{-6}+0.2z^{-8}) \quad (38)$$

for the interval $5,000 \leq n < 10,000$. The scale factor 0.674 in $H_1(z)$ and $H_2(z)$ makes the input power and the output power equal. Therefore, the SNR of the training output $d(n)$ is 30 dB.

Fig. 3 clearly shows that the CP-LMS algorithm has the faster tracking capability and the smaller steady-state error than the other algorithms.

6. CONCLUSION

By employing two adaptive filters with different update step sizes operating in parallel and complementing each other, the CP-LMS algorithm achieves both the fast convergence (tracking) speed and the small steady-state error for adaptive filtering.

Also, compared with the conventional variable step size algorithms for the adaptive FIR filtering, the CP-LMS is more versatile, more robust and simpler to use in practice. Although the new algorithm requires twice as many computations than the single fixed step size algorithm, the performance gains outweigh the extra cost.

REFERENCES

[1] S. S. Haykin, *Adaptive Filter Theory,* Prentice-Hall, Englewood Cliffs, NJ, 1991.

[2] R. W. Harris *et al*, "A variable step (VS) adaptive filter algorithm," *IEEE Trans. on ASSP,* vol. 34, no. 2, pp. 309-316, Feb. 1986.

[3] J. S. Soo and K. K. Pang, "A multistep size (MSS) frequency domain adaptive filter," *IEEE Trans. on SP,* vol. 39, no. 1, pp. 115-121, Jan. 1991.

[4] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Trans. on SP,* vol. 40, no. 7, pp. 1633-1642, Jul. 1992

[5] P. S. R. Diniz and L. W. P. Biscainho, "Optimal variable step size for the LMS/Newton algorithm with application to subband adaptive filtering," *IEEE Trans. on SP,* vol.40, no. 11, pp. 2825-2829, Nov. 1992.

[6] V. J. Mathews and Z. Xie, " A stochastic gradient adaptive filter with gradient adaptive step size, " *IEEE Trans. on SP,* vol. 41, no. 6, pp. 2075-2087, Jun. 1993.

[7] B. Widrow *et al.* , "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE,* vol. 64, no. 8, pp. 1151-1162, Aug. 1976.

[8] Leon W. Couch, *Digital and Analog Communication Systems*, p454, Macmillan Publishing Company, New York, N.Y. , 1990.

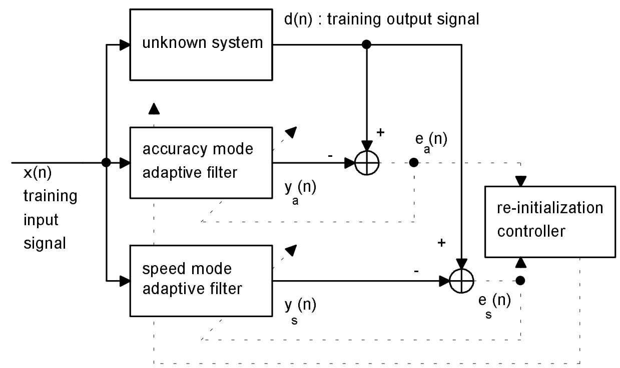

Figure 1. The CP-LMS algorithm block diagram

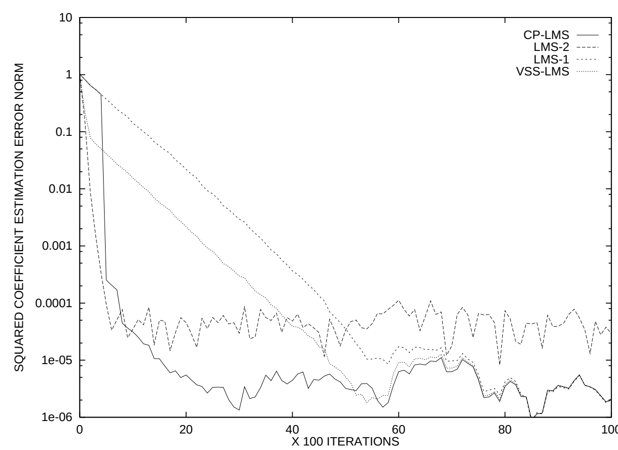

Figure 2. Norm squared estimation error plot

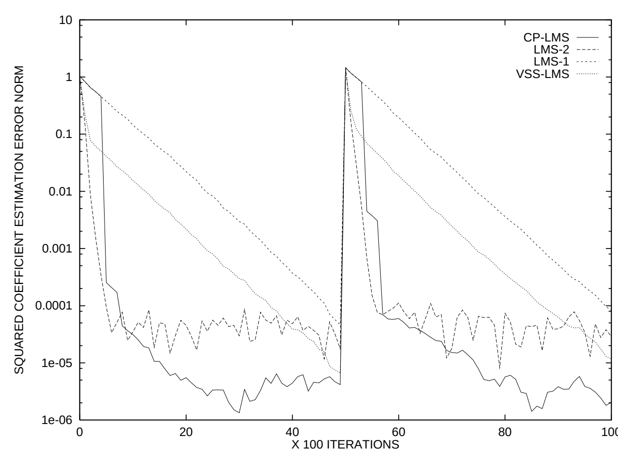

Figure 3. Norm squared estimation error plot