International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 5, No.5 (Sep. 2016)
http://dx.doi.org/10.12785/ijcds/050506

A New Variable Length LMS Algorithm for Partial Update Adaptive Filtering Driven by Cyclostaionary Signal

Thamer M. Jamel[1] and Faez Fawzi Hammood[2]

[1 2] Department of the Communications, Engineering, University of Technology, Baghdad, Iraq

Received 7 Jan. 2016, Revised 5 Mar. 2016, Accepted 5 Apr. 2016, Published 1 Sep. 2016

Abstract: The problem with some methods of Partial Update (PU) is that when using a special kind of input (Cyclostationary Signal), the system fails to converge or becomes unstable. These problems have been solved in this paper by using a newly proposed algorithm with two concepts of dynamic length Least Mean Square (LMS). The first is the length variation of the total filter coefficients N , while the second variation applies to the number of coefficients to be updated at each iteration M. This algorithm is called New Variable Length LMS algorithm NVLLMS. In the NVLLMS algorithm, the value of the Mean Error Square (MSE) is used to control N and M. Moreover, the step size is time varying. The proposed algorithm shows better performance compared with PU LMS algorithms through simulation results of system identification .

Keywords: Partial Update (PU), Least Mean Square (LMS) , New Variable Length LMS (NVLLMS), Cyclostationary Signal (CS).

1. INTRODUCTION

Cyclostationary Signals (CS), are random process signals with statistical parameters, that fluctuate periodically with time [1]. Some examples of these signals are produced by samplers, scanners, multiplexors, and modulators [1].

If an adaptive filter which uses a Least Mean Square (LMS) algorithm and is driven by correlated input signals (for example cyclostationary signal), then this algorithm suffers from some drawbacks like slow convergence or instability. One method presented overcomes these drawbacks is the selection of the adaptive filter length [2], or one can choose to update only part of the filter coefficient vector (Partial Update) at each time instant. Such algorithms can reduce computational complexity and on the other hand perform close to the full-update methods in term of convergence rate and Mean Error Square (MSE). There are several variants of the LMS algorithms with partial update methods [3-14]. In addition, partial-update adaptive filters may suffer from stability or convergence problems when the input signal is cyclostationary or periodic [15].

In this paper, a new algorithm is proposed which is called, New Variable Length LMS (NVLLMS) algorithm. Several variable lengths LMS is reported [16-22]. The NVLLMS algorithm represents a mix of PU LMS and variable length filter (VL) to solve the stability and convergence problems, and to reduce the number of total coefficients required. This algorithm is applied to system identification for different cases, and compared with other PU LMS algorithms. The proposed algorithm shows better performance through the simulation and solves the problems of the PU LMS algorithms.

2. PARTIAL UPDATE ALGORITHMS

The Partial-Update (PU) adaptive filtering method is a straightforward approach to control the computational complexity because it only updates part of the coefficient vector instead of updating the full coefficient filter vector. This method plays an important role to improve the cost, performance, portability and size, since the computational complexity of an adaptive filter are an important consideration in many practical applications, especially the applications that require the adaptive filter to have a very large number of coefficients like the acoustic echo cancellation and channel equalization [15].

To explain the use of partial updates, the least-mean square (LMS) algorithm is presented [15]:

E-mail address: drthamerjamel@gmail.com, faezfawzi@yahoo.com

$$\begin{vmatrix} w_1(k+1) \\ . \\ . \\ . \\ w_N(k+1) \end{vmatrix} = \begin{vmatrix} w_1(k) \\ . \\ . \\ . \\ w_N(k) \end{vmatrix} + \mu\ e(k) \begin{vmatrix} x(k) \\ . \\ . \\ . \\ x(k-N+1) \end{vmatrix} ,$$

$$k = 0,1,2,3,\ldots \qquad (1)$$

or

$$\mathbf{W}(k+1) = \mathbf{w}(k) + \mu\, e(k)\boldsymbol{X}(k)$$

Where **w**(k) is tapping coefficient vector , μ is the step size parameter, e(k) is the error signal , and ***X*(**k) is an input signal vector.

Fig.1 shows the concept of the PU method, which indicates that only M out of N coefficients are updated at iteration k.

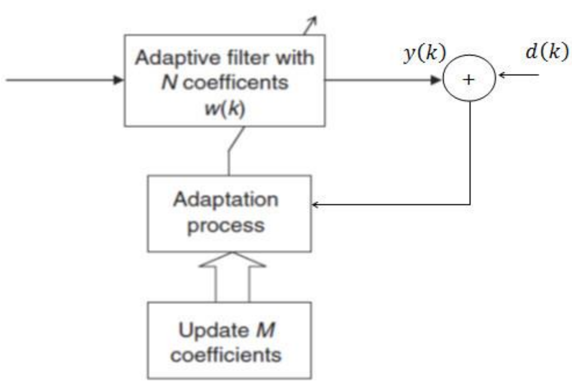

Figure1. The concept of PU method

This can be accommodated by modifying the adaptation algorithm in the equation (1) to

$$w(k+1) = w(k) + \mu e(k) I_M(k) x(k),$$
$$k = 0,1,2,\ldots \qquad (2)$$

Where $I_M(k)$ is a diagonal matrix with M ones and $N-M$ zeros on its diagonal indicating which M coefficients are to be updated at iteration M [15] :

$$I_M = \begin{bmatrix} i_1(k) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & i_N(k) \end{bmatrix} , \quad \sum_{j=1}^{N} i_j(k) = M \quad ,$$
$$k = 0,1,2,3,\ldots \qquad (3)$$

If $i_j(k) = 1,\ j = 1,\ldots,N,$ then the coefficient $w_j(k)$ gets an update at iteration k, otherwise it remains unchanged.In this paper, we consider some of the basic partial update methods as follows:

A. *Periodic Partial Update*

In this method the update complexity is distributed over a number of iterations, in order to reduce the computational complexity [15].

$$w(k+1) = w(k) + f(k),\ \ k = 0,1,2,\ldots$$
$$y(k) = h(w(k), x(k)) \qquad (4)$$

Where y(k) is the output of adaptive filter , and

$$f(k) = [f_1(k), f_2(k), \ldots, f_N(k)]^T \qquad (5)$$

is the $N \times 1$ update vector,

Using periodic partial updates for the adaptive filter coefficients in equation (4) results in [15]:

$$w\big((k+1)S\big) = w(kS) + f(kS),$$
$$k = 0,1,2,\ldots. \qquad (6)$$
$$w\big((kS+i)\big) = w(kS),$$
$$i = 0,1,2,\ldots,S-1$$
$$y(k) = h(w(\lfloor k/S \rfloor S), x(k))$$

Where S is the period of coefficient updates

The adaptive filter updates its coefficients every S^{th} iteration at $k = 0, S, 2S, 3S, \ldots,$

$$w\big((k+1)S\big) = w(kS) + 2\mu e(kS) x(kS),$$
$$k = 0,1,2,3,\ldots.$$
$$w\big((kS+i)\big) = w(kS),\ \ i = 0,1,2,\ldots,S-1 \qquad (7)$$
$$S = \left\lceil \frac{N}{M} \right\rceil \qquad (8)$$

B. *Sequential Partial Update*

In order to reduce the computational complexity of updating a subset of the adaptive filter coefficients at each iteration , the sequential partial update method is used [3]. The sequential partial update method adapts a partition of the coefficient vector within the complexity constraints at each iteration [15]. The sequential PU method may choose the first element at iteration k, choose the second element at iteration $k+1$, and the third element at iteration $k+2$. Updating a subset of the weights can reduce computational complexity. The sequential-partial-update LMS algorithm is given by:

$$w(k+1) = w(kS) + \mu e I_M(k) x(k), \qquad k = 0,1,2,\ldots. \qquad (9)$$

$$I_M(k) = \begin{vmatrix} i_1(k) & 0 & \cdots & 0 \\ 0 & i_2(k) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \ddots & i_N(k) \end{vmatrix},$$

$$i_j(k) = \begin{cases} 1 if\ j \in \zeta_{(k\ mod\ B)+1} \\ 0 \quad otherwise \end{cases} \quad (10)$$

Where B = N/M is an integer equal to the number of subsets (each B includes M coefficient). The adaptation process is roughly proportional to B so the sequential partial update method designs the subset as [15]:

$$\zeta_1 = \{1,2,3, \ldots, M\},$$
$$\zeta_2 = \{M+1, M+2, \ldots, 2M\},$$
$$\vdots$$
$$\zeta_B = \{(B-1)M+1, (B-1)M+2, \ldots, N\} \quad (11)$$

C. Stochastic Partial Updates

The stochastic method can be implemented as a randomized version of the sequential method. The stochastic partial update has the same update equation as the sequential partial update. The method uses the following coefficient selection matrix [15]:

$$I_M(k) = \begin{vmatrix} i_1(k) & 0 & \cdots & 0 \\ 0 & i_2(k) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \ddots & i_N(k) \end{vmatrix},$$

$$i_j(k) = \begin{cases} 1 & if\ j \in \zeta_{m(k)} \\ 0 & otherwise \end{cases} \quad (12)$$

where $m(k)$ is an independent random process .

D. Updates M-Max

In the M-Max Update the weight vector is updated according to the position of the first M-max elements of the input x. This method aims to find the subset of the update vector that can make the biggest contribution to the convergence [15].The coefficient selection matrix $I_M(k)$ is defined by:

$$I_M(k) = \begin{vmatrix} i_1(k) & 0 & \cdots & 0 \\ 0 & i_2(k) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \ddots & i_N(k) \end{vmatrix},$$

$$i_j(k) = \begin{cases} 1 \ \ if|x(k-j+1)| \in max_{1\le l\le N}(|x(k-l+1)|, M) \\ 0 \quad\quad otherwise \end{cases} \quad (13)$$

3. VARIABLE LENGTH (VL) ADAPTIVE FILTER

The VLLMS algorithm is dynamically updated the tap-length of the LMS adaptive filter to optimize performance and to reduce the power in the adaptive filter core. This algorithm is also used as a PU when the number of tapes needed is greater than the existing. A variable length LMS adaptive filter will be introduced with the configuration shown in Fig. 2 [24].

Fig.2 shows three parallel filters. The coefficient vector of filter l is $h_l(k)$ with length $N_l(l = 1,2,3)$. The three estimation errors $e_l(k)$ are input into a length control unit which computes estimates of the MSE for each filter, and based on these estimates the updated filter lengths and step size are generated. The optimal filter length produced by filter number three.

The algorithm of operation can be summarized as follows [24]:

1) Initialization:

$$N_1(0) = N_0, N_2(0) = N_0 + 1, N_3(0) = N_0 + 2 \quad (14)$$

$$h_1(0) = 0_{N1}, h_2(0) = 0_{N2},\ h_3(0) = 0_{N3} \quad (15)$$

$$\mu_1(0) = \frac{\mu N_0}{N_1(0)},\ \mu_2(0) = \frac{\mu N_0}{N_2(0)},\ \mu_3(0) = \frac{\mu N_0}{N_3(0)}, \quad (16)$$

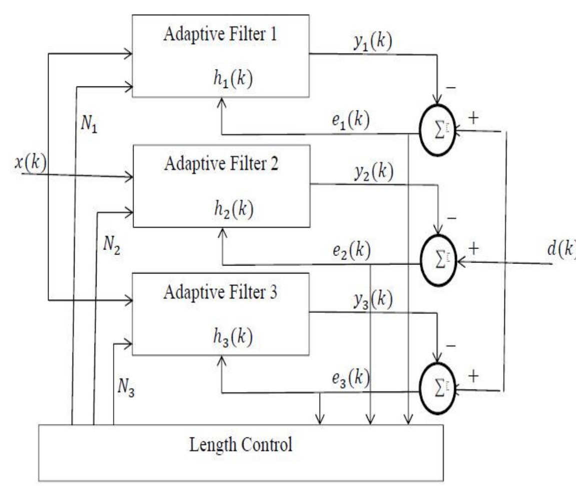

Figure 2. VLLMS filter configuration

Where N_0 is the initial filter length, N_1, N_2 *and* N_3 are the initial filter length of filter 1,2 and 3 respectively. μ is the step size parameter of filter1

$h_1(0)$, $h_2(0)$ *and* $h_3(0)$ are the initial filter coefficient of filter 1, 2 and 3 respectively.

$\mu_1(0), \mu_2(0)$ *and* $\mu_3(0)$ are the initial step size of filter 1, 2 and 3 respectively

2) For each $n = 1,2,\ldots$
Compute the errors:

$$e_l(n) = d(n) - \sum_{i=0}^{M_l-1} h_{li}(n)x(n-i-l),$$
$$l = 1,2,3 \quad (17)$$

Where $h_{li}(k)$ is the i^{th} coefficient of the adaptive filter l.

Update the coefficients:

$$h_l(n+1) = h_l(n) - \mu_{li}(n)x(n)e_l(n),$$
$$l = 1,2,3 \quad (18)$$

At each L^{th} iteration:

Compute the following average m_l

$$m_l = \frac{1}{L}\sum_{j=n+L+1}^{n} e_l^{\,2}(j)\,, l = 1,2,3 \quad (19)$$

Update the lengths:

$$N_1(n+1) = \begin{cases} N_1(n)+1, & if \quad m_1 > m_2 > m_3 \\ N_1(n), & if \quad m_1 > m_2 or\ m_{22} \le m_{33} \\ N_1(n)-1, & otherwise \end{cases} \quad (20)$$

$$N_2(n+1) = N_1(n+1)+1,\ N_3(n+1) = N_1(n+1)+2 \quad (21)$$

Update step size

$$\mu_l(n+1) = \frac{\mu N_o}{N_l(n+1)}, l = 1,2, \quad (22)$$

Where m_i $(i = 1,2,3)$ acts as the statistic square error in this algorithm. It is shown that for an uncorrelated input signal $x(n)$ which satisfies:

$E[x(k)x(k-n)] = 0\ for\ n \ne 0$, the following is valid:

$$\frac{J_\infty^{(l)}}{J_\infty^{(m)}} = \frac{J_{min}^{(l)}}{J_{min}^{(m)}} = \begin{cases} \ge 1, & if \quad N_1 \le N_2 < N_{opt} \\ < 1, & if \quad N_2 < N_1 < N_{opt} \\ 1, & if \quad N_1 > N_{opt}\ or\ N_2 > N_{opt} \end{cases}$$

$$l = 1,2,3 \qquad m = 1,2,3 \qquad l \ne m \quad (23)$$

Where $J_\infty^{(l)} = E\left[e^2(\infty)^{(l)}\right], l = 1,2,3$ is the steady state MSE, $J_{min}^{(l)}, l = 1,2,3$ is the minimum steady state MSE and N_{opt} is the optimal filter length [24].

4. Cyclostationary Process and PU Problems

Cyclostationary processes are random data whose statistical characteristics vary periodically with time [25]. The cyclostationary input signal is defined as a signal of order n(in the wide sense) if and only if it is possible to find some n^{th} -order nonlinear transformation of the signal that will generate finite-strength additive sine-wave components, which result in spectral lines.

$$x(k) = a_i u(k),$$

$$i = \begin{cases} 1\ if\ k\ mod\ 8 = 1 \\ 2\ if\ k\ mod\ 8 = 2 \\ 3\ if\ k\ mod\ 8 = 3 \\ 4\ if\ k\ mod\ 8 = 4 \\ 5\ if\ k\ mod\ 8 = 5 \\ 6\ if\ k\ mod\ 8 = 6 \\ 7\ if\ k\ mod\ 8 = 7 \\ 8\ if\ k\ mod\ 8 = 0 \end{cases} \quad (24)$$

Where $u(k)$ is an uncorrelated Binary-Phase-Shift-Keying (BPSK) signal taking on values ±1 with equal probability, and the a_i are constant weights that multiply $u(k)$ periodically, creating a cyclostationary signal with cyclic period 8.

In PU, there are some problems with the adaptive filter convergence and stability when cyclostationary inputs are applied as follows:

1) Periodic PU method: it has the problem of convergence difficulties.
2) Sequential PU method: it has the problem of convergence difficulties.
3) M-Max PU method: it has the problem of instability.

5. A New Proposed Algorithm (New Variable Length LMS (NVLLMS))

The new proposed algorithm combines the benefits of the VLLMS and the PU-LMS algorithms in order to solve the convergence and stability problems of periodic, stochastic and M-Max PU. This algorithm is called NVLLMS. The concept of changing the number of coefficients to be updated at each iteration will be used.

1) Design Aspects

The following design aspects are considered as follows:

1) Variable length of partial update coefficients (variable M).
2) Variable adaptive filter length (variable N).
3) The step size will be variable.
4) Modifying the length is depending on the value

of MSE at each iteration and the value of MSE for a group of iteration. The group of iteration can be defined by blocks , such that each block has the same size. The MSE is calculated for each block by the equation (Sum(e)/block size).

2) Proposed Algorithm

The proposed new algorithm introduces an adaptive filter with variable length N and also has the ability to vary the number of coefficients to be updated at each iteration M when using one of the PU methods. This can be introduced by the following configuration as shown in Fig. 3.

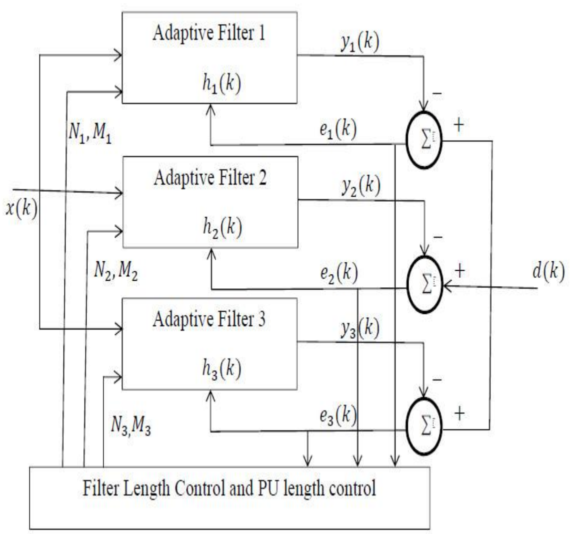

Figure 3. The NVLLMS filter configuration with variable N and M.

The algorithm of operation sequences can be summarized as follows:

1) Initialization:

$$N_1(0) = N_0, N_2(0) = N_0 + 1, N_3(0) = N_0 + 2 \tag{25}$$

$$h_1(0) = 0_{N1}, h_2(0) = 0_{N2},\ h_3(0) = 0_{N3} \tag{26}$$

$$\mu_1(0) = \frac{\mu N_1(0)}{N_0},\ \mu_2(0) = \frac{\mu N_2(0)}{N_0},\ \mu_3(0) = \frac{\mu N_3(0)}{N_0}, \tag{27}$$

The step size is increased instead of decreasing, in order to increase the speed of convergence [17]. Keeping that at all iterations the value of step size must be in the range of stability condition.

2) For each $k = 1, 2, \ldots$

Compute the errors: $e_l(k)$

$$e_l(k) = d(k) - \sum_{i=0}^{N_l-1} h_{li}(k)x(k-i-l), l = 1,2,3 \tag{28}$$

Update the coefficients:

$$h_l(k+1) = h_l(k) - \mu_{li}(k)x(k)e_l(k), l = 1,2,3 \tag{29}$$

At each L^{th} iteration:

Compute the average of $e_l(k)$ which denoted by m_l , where m_l is average error.

$$m_l = \frac{1}{L}\sum_{j=k+L+1}^{n} {e_l}^2(j)\, , l = 1,2, \tag{30}$$

At each block of iteration with size D:

Compute the average of m_l for each block which denoted by m_b , where m_b is MSE .

$$m_b = \frac{1}{D}\sum_{j=k+L+1}^{n} {e_l}^2(j)\, , l = 1,2,3 \tag{31}$$

Where D is the size's number of iterations that the block contained.

The number sample at each block can be determined by using the mod operation. If the size of the block is D then $(k \bmod D)$ must be zero to compute the average of each block. The value of M is updated when the system fails to converge or becomes unstable, and that happens when the value of average error for a block is greater than the previous block. The blocks are used to recognize the system output if it is converging and stable or not, and that is done by comparing the first item in the block with the last item from the same block, and also comparing both items with a reference value (this value is set according to the estimated output).

The system fails to converge

$$if\ \ m_b\,(i) \approx\ m_b\,(i+bs)\ \ and\ \ m_b\,(i)\ \neq \text{output} \tag{32}$$

The system is unstable

$$if\ \ m_b\,(i) <\ m_b\,(i+bs) \tag{33}$$

Where the bs is the number of iterations per each block.

$$M(k+1) = \begin{cases} M(k) + \text{update size}, & if\ \ \text{sustem fail to converge} \\ & \text{or unstable} \\ M_0\ \ if\ \ M(k) = N - \text{update size} & \\ M(k), & otherwise \end{cases} \tag{34}$$

The updated size differs according to the application and it may be positive or negative, the case of negative update is taking place when the value of M initially starts with high value so it has to be decreased to cover all values of M. The maximum value can be reached by M must be less than N to keep the PU concept. If the variation of M has no effect on the output, then the system will reset the value of M to the initial value M_0 .

Filter length variation is done according to the following procedure:

$$N_2(k+1) = N_1(k+1) + 1,\ N_3(k+1) = N_1(k+1) + 2 \tag{36}$$

Update step size

$$\mu_l(k+1) = \frac{N_l(k+1)\mu_0}{N_0}, l = 1,2, \tag{37}$$

The equation of the step size update is changed to increase the step size in order to increase the convergence speed, but the boundary of stability is the same for both cases. The block size is another factor that can be set to increase the accuracy of the result Decreasing the block size will increase the accuracy and vice versa.

6. SIMULATION RESULTS

The basic block diagram of system identification application is shown in Fig.4.

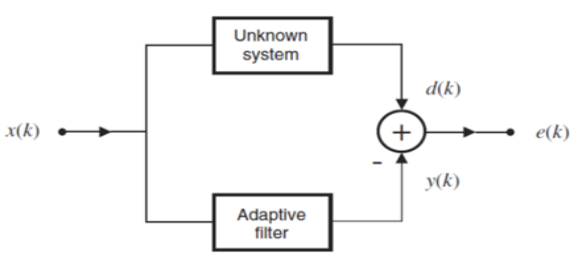

Figure 4. System Identification set up

The input signal $x(k)$ is applied to the unknown system and to the adaptive filter [2, 17].While the desired signal $d(k)$ is the output of the unknown system.

Assume the unknown system has an impulse response given by [2]: $h = [0.1\ 0.3\ 0.0 - 0.2 - 0.4 - 0.7 - 0.4 - 0.2]^T$. The input signal is white Gaussian with variance σ_n^2 equals 10^{-4}. The adaptive filter has 8 coefficients. The number of iterations k equals 10000 and μ equals 5×10^{-4}. The error signal is then given by equation (38)

$$e(k) = d(k) - y(k) \tag{38}$$

$$y(k)=w^T(k)x(k) \tag{39}$$

At each iteration , the adaptive filter updates its coefficients in order to minimize an appropriate norm of the error signal $e(k)$. When the error norm is minimized in a statistical sense, the corresponding $w(k)$ gives an estimate of the unknown system parameters.The cyclostationary input given by equation (24) is used.

A. Case(1):Traditional Full Update LMS and PU LMS with cyclostationary input :-

Fig. 5 shows the time-averaged learning curves for full update LMS and periodic, sequential, stochastic and M-Max PU LMS algorithms when cyclostationary input is applied

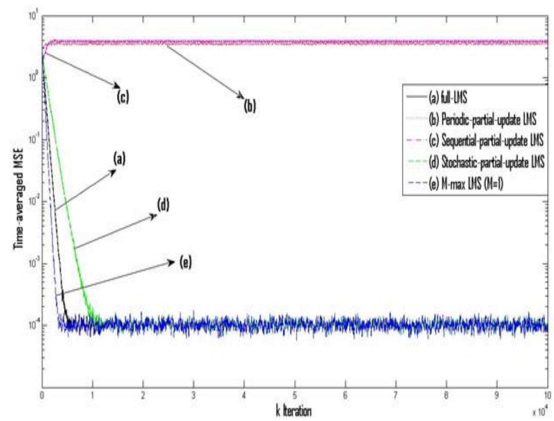

Figure 5. Time-averaged learning curves (MSE) for a full update , periodic, sequential, stochastic and M-Max PU LMS algorithms for a cyclostationary input.

It can be observed that the periodic and the sequential for S=2 PU LMS fails to converge altogether, whereas the full-update and other PU LMS algorithms appear not to have been affected by the cyclostationary input signal. Moreover, also it can be seen that for M=1 , the M-Max LMS is faster convergence than the full-update.

B. Case(2): Traditional Full Update LMS and PU LMS with shift cyclostationary input :-

The shifted version of the cyclostationary input signal is considered and is given by:

$$x(k) = a_i u(k), \quad i = (k \bmod 8) + 1 \tag{40}$$

The new signal has the same cyclic period as the previous one and it is obtained by advancing the signal shown in equation (24) by one sample [15]. Fig.6 shows the time-averaged learning curves for full update LMS and periodic S=2, sequential M=7, stochastic M=7 and M-Max (M=1) PU LMS algorithms.

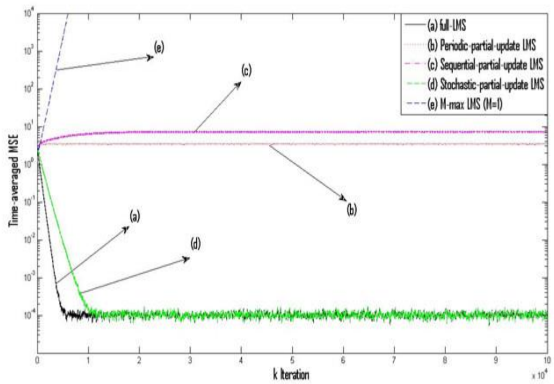

Figure 6. Time-averaged learning curves (MSE) for the full update, periodic, sequential, stochastic and M-Max M=1 PU LMS algorithms for a shifted cyclostationary input

Fig.6 shows that periodic and the sequential PU LMS fails to converge and the M-Max PU becomes unstable while the full-update and stochastic PU LMS algorithms appear not to have been affected by the new cyclostationary input signal and the stochastic is more stable compared to other methods of LMS PU.

C. Case(3): Applying a new proposed (NVLLMS) Algorithm

The NVLLMS algorithm will be used with the cyclostationary input, which is the same input that used before with periodic, sequential and M-max PU. NVLLMS will try to solve the problems of convergence and stability of PU algorithms.

1) The periodic PU.

Fig.7 shows the two learning curves of periodic PU, curve (a) when using periodic PU LMS and curve (b) when using the periodic PU NVLLMS algorithm. This figure shows that, when the NVLLMS is applied, the system successfully converged, but with a short delay , (about 500 iterations) This delay depends on the number of iterations at each block.

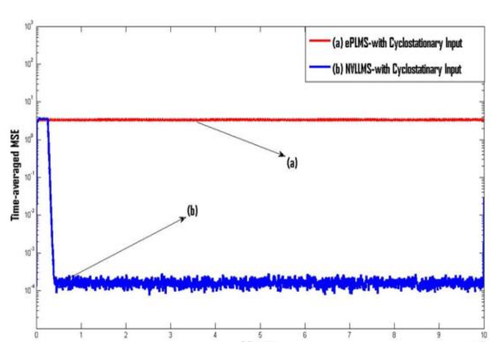

Figure 7. Time-averaged learning curves (MSE) for (a) a periodic PU LMS, (b) for a periodic PU NVLLMS algorithm for a cyclostationary input.

2) The sequential PU.

Fig. 8 shows the two learning curves of sequential PU, curve (a) when using the sequential PU LMS, and curve (b) for the sequential PU NVLLMS algorithm. It can be observed that when the NVLLMS is applied, the system also succeeded to converge with a small delay (about 1500 iterations) similar to the previous case.

3) The M-max PU.

Fig. 9 shows the learning curve of M-max PU, curve (a) when using M-max PU LMS, and curve (b) for the M-max PU NVLLMS algorithm. As shown , the proposed lgorithm produces a stable system (after 2000 iterations) instead of the old situation which showing system instability.

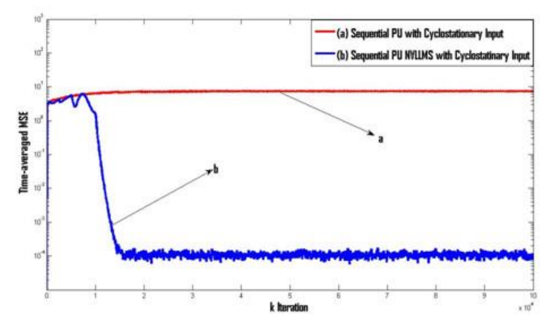

Figure 8 . Time-averaged learning curves (MSE) for (a) a sequential PU LMS, (b) a sequential PU NVLLMS algorithm for a cyclostationary input.

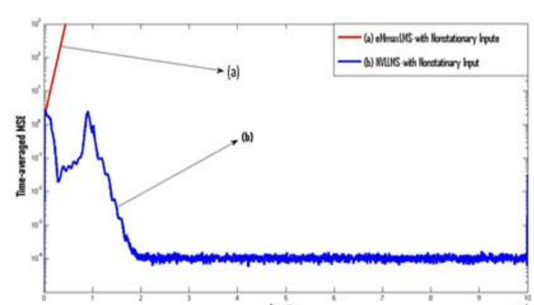

Figure 9. Time-averaged learning curves (MSE) for (a) an M-Max PU LMS, (b) an M-Max PU NVLLMS algorithm for a cyclostationary input

Using NVLLMS with periodic and sequential PU makes the system succeeded to converge, and in the case of M-Max it produces a stable system .

4) Case(4):System Identification with Nonstationary Impulse Response

The cyclostationary signal is a special class of nonstationary signals which are random in nature, but exhibit periodicity in their statistics [26]. The nonstationary impulse response is the case when the impulse response of the system is not constant, but it is at some point will be changed, causing the system to start learning again to follow the change. In this section, the impulse response will be changed by multiplying the original one by -2 at iteration 3000. This can be used as a performance test for the NVLLMS algorithm. Fig.10 shows the learning curve for both full update LMS and NVLLMS algorithms . It is evident that the NVLLMS has faster convergence rate and better tracking capability compared with a full update LMS algorithm.

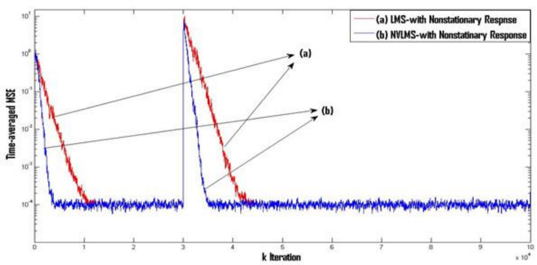

Figure 10. Time-averaged learning curves (MSE) with a nonstationary impulse response for full update (a) LMS, (b) NVLLMS algorithm

7. CONCLUSIONS

In this paper a new proposed algorithm called NVLLMS is proposed in order to overcome some of problems of periodic, stochastic PU (fail to converge) and M-Max PU (becomes unstable) algorithms that are driven by cyclostationary signal. The proposed algorithm used three design aspects which are variable length of partial update coefficients (variable M) , variable adaptive filter length (variable N), and variable step size. Both N and M parameters are controlled by MSE. Simulation results using system identification set up shows that , using NVLLMS with periodic and sequential PU makes the system succeeded to converge after 500 and 1500 iterations respectively , and in the case of M-Max it produces a stable system after 2000 iterations. This delay (500-2000 iterations) depends on the number of iterations at each block (D).

Moreover, the proposed algorithm shows the fast convergence rate, and better tracking compared to a full band LMS algorithm for nonstationary environments.

REFERENCES

[1] Gardner, W., and L. Franks. "Characterization of cyclostationary random signal processes." *IEEE Transactions on information theory* 21.1 (1975): 4-14.

[2] Paulo, SR Diniz. "Adaptive filtering: algorithms and practical implementation." *The international series in Engineering and Computer Scienc* (2008).

[3] Douglas, Scott C. "Adaptive filters employing partial updates." *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 44.3 (1997): 209-216.

[4] Riera-Palou, F., James M. Noras, and D. G. M. Cruickshank. "Linear equalisers with dynamic and automatic length selection." *Electronics Letters*37.25 (2001): 1.

[5] Bilcu, Radu Czprian, P. Kuosmanen, and K. Egiazarian. "A new variable length LMS algorithm: theoretical analysis and implementations."*Electronics, Circuits and Systems, 2002. 9th International Conference on*. Vol. 3. IEEE, 2002.

[6] Tanrıkulu, Oğuz, and Kutluyıl Doğançay. "Selective-partial-update proportionate normalized least-mean-squares algorithm for network echo cancellation." *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*. Vol. 2. IEEE, 2002.

[7] Arezki, M., et al. "A new algorithm with low complexity for adaptive filtering."*Engineering Letters* 18.3 (2010): 205.

[8] Godavarti, Mahesh, and Alfred O. Hero. "Partial update LMS algorithms."*IEEE Transactions on Signal Processing* 53.7 (2005): 2382-2399.

[9] Khong, Andy WH, et al. "A low complexity fast converging partial update adaptive algorithm employing variable step-size for acoustic echo cancellation." *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2008.

[10] Schuldt, Christian, Fredric Lindstrom, and Ingvar Claesson. "Low-Complexity Adaptive Filtering Implementation for Acoustic Echo Cancellation." *TENCON 2006-2006 IEEE Region 10 Conference*. IEEE, 2006.

[11] Dariusz Bismor," Variable Step Size Partial Update LMS Algorithm for Fast Convergence", The 22nd International Congress of Sound and Vibration ICSV22, Florence, Italy, 12-16 July 2015.

[12] Ahmed, Zouhour Ben, and Nabil Derbel. "Identification of Volterra-PARAFAC models using partial update LMS algorithms." *2015 7th International Conference on Modelling, Identification and Control (ICMIC)*. IEEE, 2015.

[13] Shah, Syed Muslim. "Riemann–Liouville operator-based fractional normalised least mean square algorithm with application to decision feedback equalisation of multipath channels." *IET Signal Processing* (2016).

[14] Kurczyk, Sebastian, and Marek Pawelczyk. "Nonlinear Active Noise Control using partial-update Filtered-s LMS algorithm." *Methods and Models in Automation and Robotics (MMAR), 2015 20th International Conference on*. IEEE, 2015.

[15] Dogancay, Kutluyil. *Partial-update adaptive signal processing: Design Analysis and Implementation*. Academic Press, 2008.

[16] Rusu, Corneliu, and Colin FN Cowan. "Novel stochastic gradient adaptive algorithm with variable length." *European Conference on Circuit Theory and Design (ECCTD'01)*. 2001.

[17] Zhang, Yonggang, et al. "Steady-state performance analysis of a variable tap-length LMS algorithm." *IEEE Transactions on Signal Processing* 56.2 (2008): 839-845

[18] Bilcua Radu Ciprian, Pauli Kuosmanenb and Karen Egiazarian ," On length adaptation for the least mean square adaptive filters", Signal Processing 86.10 (2006) 3089–3094.

[19] Gu, Yuantao, Kun Tang, and Huijuan Cui. "LMS algorithm with gradient descent filter length." *IEEE Signal Processing Letters* 11.3 (2004): 305-307.

[20] Gong, Yu, and Colin FN Cowan. "An LMS style variable tap-length algorithm for structure adaptation." *IEEE Transactions on Signal Processing* 53.7 (2005): 2400-2407.

[21]Wei, Ying, and Zhibin Yan. "Variable tap-length LMS algorithm with variable tap-length adaptation step size." *2016 Chinese Control and Decision Conference (CCDC)*. IEEE, 2016.

[22] Mohanty, Basant Kumar, and Sujit Kumar Patel. "Efficient very large-scale integration architecture for variable length block least mean square adaptive filter." *IET Signal Processing* 9.8 (2015): 605-610

[23] BeiXie, Partial Update Adaptive Filtering, Ph.D. Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, December, 2011.

[24] Liu, Huicui, et al. "Variable length LMS adaptive filter for carrier phase multipath mitigation." *GPS solutions* 15.1 (2011): 29-38.

[25] Napolitano, Antonio. "Cyclostationary Signal Processing and its Generalizations." (2014).

[26] Gardner, William A. *Cyclostationarity in communications and signal processing*. STATISTICAL SIGNAL PROCESSING INC YOUNTVILLE CA, 1994.

Thamer M. Jamel was born in Baghdad , Iraq . He graduated from the University of Technology with a Bachelor's degree in electronics engineering in 1983. He received a Master's degree in digital communications engineering from the University of Technology in 1990 and a Doctoral degree in communication engineering from the University of Technology in 1997. He is associate professor in the communication engineering department at University of Technology, Baghdad, Iraq. His research interests include adaptive signal processing for digital communications systems.

Faez Fawzi was born in Baghdad , Iraq . He graduated from the University of Technology with a Bachelor's degree in Communications engineering in 2006. He received a Master's degree in digital communications engineering from the University of Technology in 2014 His research interests include adaptive signal processing.