

Variable, Step-Size, Block Normalized, Least Mean, Square Adaptive Filter: A Unified Framework

M. Shams Esfand Abadi¹, S.Z. Moussavi* and A. Mahlooji Far²

Employing a recently introduced framework, within which a large number of classical and modern adaptive filter algorithms can be viewed as special cases, a generic, variable step-size adaptive filter has been presented. Variable Step-Size (VSS) Normalized Least Mean Square (VSSNLMS) and VSS Affine Projection Algorithms (VSSAPA) are particular examples of adaptive algorithms covered by this generic variable step-size adaptive filter. In this paper, the new VSS Block Normalized Least Mean Square (VSSBNLMS) adaptive filter algorithm is introduced, based on the generic VSS adaptive filter. The proposed algorithm shows the higher convergence rate and lower steady-state mean square error compared to the ordinary BNLMS algorithm.

INTRODUCTION

Adaptive filtering has been, and still is, an area of active research that plays an important role in an ever increasing number of applications, such as noise cancellation, channel estimation, channel equalization and acoustic echo cancellation. The least mean square (LMS) and its normalized version (NLMS) are the workhorses of adaptive filtering. In the presence of colored input signals, the LMS and the NLMS algorithms have extremely slow convergence rates. To solve this problem, a number of adaptive filtering structures, based on affine subspace projections [1,2] and multirate techniques, have been proposed in the literature [3–5]. In all these algorithms, the selected fixed step-size can change the convergence rate and the steady-state mean square error. By optimally selecting the step-size, during the adaptation, one can obtain the both fast convergence rate and low steady-state Mean Square Error (MSE). Important examples of the two new Variable Step-Size (VSS) versions of the NLMS and the

Affine Projection (AP) algorithm can be found in [6].

In [7], the generic adaptive filter, based on the weighted, estimated Wiener-Hopf equation, is proposed. The LMS and the NLMS adaptive algorithms, the family of Affine Projection Algorithms (APA), the Transform Domain Adaptive Filters (TDAF) [8] and the Pradhan Reddy Subband Adaptive Filters (PRSAF) [9] are the particular examples that can be covered with this generic adaptive filter.

The objective, in this paper, is firstly to show that the generic adaptive filter proposed in [7] can cover the Block LMS (BLMS) and the Block Normalized LMS (BNLMS) adaptive filter algorithms. Secondly, based on the generic adaptive filter, the generic variable step-size update equation is developed. The VSSNLMS and VSSAPA of [6] can be easily derived from this generic variable step-size adaptive filter. The following proceeds by presenting the VSS version of the BNLMS adaptive filter, named the VSSBNLMS, which is characterized by the fast convergence speed and reduced steady-state MSE, when compared to the ordinary BNLMS adaptive filter algorithm.

The paper is organized as follows: In the following section, the generic variable step-size update equation, forming the basis of the development of the VSS-BNLMS, is introduced. Subsequently, the VSSBNLMS algorithm will be presented. In the next section, the computational complexity of the BNLMS and VSS-

1. Department of Electrical Engineering, Shahid Rajaei Teacher Training University, Tehran, I.R. Iran.

*. Corresponding Author, Department of Electrical Engineering, Shahid Rajaei Teacher Training University, Tehran, I.R. Iran.

2. Department of Electrical Engineering, Tarbiat Modarres University, Tehran, I.R. Iran.

BNLMS is calculated and compared. Finally, before concluding the paper, the advantages of the algorithms are demonstrated by presenting several experimental results.

GENERIC VARIABLE STEP-SIZE UPDATE EQUATION

The generic filter vector update equation at the center of this analysis can be stated as [10-12]:

$$\underline{h}(n+1) = \underline{h}(n) + \mu X(n)W(n)\underline{e}(n). \quad (1)$$

A notation is used, based on the adaptive filtering setup shown in Figure 1 and explained in Table 1.

Note that all vectors are columns, unless explicitly transposed through the superscript, T , notation. For more details, please refer to [10-12].

An important goal for all adaptive filters is the rapid convergence to an accurate solution of the Wiener-Hopf equation in a stationary environment. The Wiener-Hopf equation is:

$$R\underline{h}_t = \underline{r}, \quad (2)$$

where \underline{h}_t is the unknown true filter vector, R is the autocorrelation matrix of the filter input signal, $R = E\{\underline{x}(n)\underline{x}^T(n)\}$, and \underline{r} is the crosscorrelation vector

defined by $\underline{r} = E\{\underline{x}(n)d(n)\}$. $d(n)$ is commonly referred to as the desired signal that arises from the linear model, $d(n) = \underline{x}^T(n)\underline{h}_t + v(n)$, where $v(n)$ is the measurement noise. Since one cannot expect the exact knowledge of R and \underline{r} of Equation 2 and, because it is reasonable to assume those quantities to be time dependent, it makes sense to formulate the adaptive filtering problem as the problem of finding the time dependent solution, $\underline{h}(n)$, to:

$$\hat{R}(n)\underline{h}(n) = \hat{\underline{r}}(n), \quad (3)$$

where $\hat{R}(n)$ and $\hat{\underline{r}}(n)$ denote estimates of the correlation quantities of Equation 2. By defining the $M \times K$ data matrix, as follows:

$$X(n) = [\underline{x}(n), \underline{x}(n-1), \underline{x}(n-2), \dots, \underline{x}(n-K+1)], \quad (4)$$

and, being given some $K \times K$ full rank symmetric weighting matrix $W(n)$, one could reasonably state the estimated Wiener-Hopf equation (Equation 3) as:

$$X(n)W(n)X^T(n)\underline{h}(n) = X(n)W(n)\underline{d}(n), \quad (5)$$

where $\underline{d}(n)$ is a $K \times 1$ vector of desired signal samples, defined as:

$$\underline{d}(n) = [d(n), d(n-1), d(n-2), \dots, d(n-K+1)]^T, \quad (6)$$

that can be obtained from the following equation:

$$\underline{d}(n) = X^T(n)\underline{h}_t + \underline{v}(n), \quad (7)$$

where $\underline{v}(n) = [v(n), v(n-1), \dots, v(n-K+1)]^T$ is the measurement noise vector. It is noticed that, if $W(n) = I$, where I is the identity matrix, the estimates used are standard sample estimates of the correlation quantities involved. The larger the value of K is selected, the better estimates one would expect. Selecting $W(n)$ different from the identity matrix makes it possible to

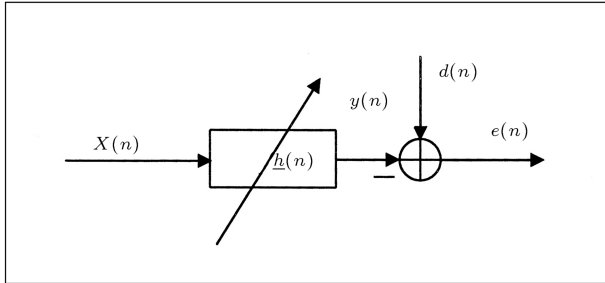


Figure 1. Adaptive filter setup.

Table 1. Explanation of notation.

$\underline{h}(n)$	Length M column vector of filter coefficients to be adjusted at each time instant n
$\underline{x}(n)$	Length M vector of input signal samples to adaptive filter, $\underline{x}(n) = [x(n), x(n-1), \dots, x(n-M+1)]^T$
$\underline{e}(n)$	Length K vector of error samples, $\underline{e}(n) = [e(n), e(n-1), \dots, e(n-K+1)]^T$
$X(n)$	$M \times K$ signal matrix whose columns are given by: $[\underline{x}(n), \underline{x}(n-1), \dots, \underline{x}(n-K+1)]$
$W(n)$	$K \times K$ symmetric weighting matrix
μ	Step-size

use weighted estimates of the correlation quantities. For the case when $W(n) = [X^T(n)X(n)]^{-1}$, or some function of this quantity, it is common to refer to the associated estimates as the data normalized estimates.

Applying a stationary iterative linear equation solver [13] to Equation 5 entails a splitting of the coefficient matrix, $X(n)W(n)X^T(n)$:

$$X(n)W(n)X^T(n) = (\mu.I)^{-1} - [(\mu.I)^{-1} - X(n)W(n)X^T(n)], \quad (8)$$

where μ is step-size and I is the identity matrix, therefore, $\mu.I$ is a $M \times M$ full rank matrix. Furthermore, performing only one iteration, according to the splitting above for each time index, n , the generic update equation, Equation 1, will be obtained, when one makes use of the fact that $\underline{e}(n) = \underline{d}(n) - X^T(n)\underline{h}(n)$. Based on the above, several adaptive filter algorithms, given by specific choices of K and $W(n)$ corresponding to the LMS, the NLMS and the AP algorithms, can be derived [10]. One can also incorporate the BLMS and BNLMS algorithms in this generic update equation. The particular choices and their corresponding algorithms are summarized as the top five lines in Table 2. The last two entries in Table 2 will be explained in the following sections. It is interesting to note that the most common adaptive filtering algorithms can be interpreted as some sort of Richardson iteration [12]; the simplest of all iterative linear equation solvers, applied to a particular estimated Wiener-Hopf equation.

One now proceeds by determining the optimum step-size, $\mu^o(n)$, instead of using μ in the VSS version of Equation 1. The latter equation can be stated in terms of weight error vector, $\underline{\varepsilon}(n) = \underline{h}_t - \underline{h}(n)$, as follows:

$$\underline{\varepsilon}(n+1) = \underline{\varepsilon}(n) - \mu X(n)W(n)\underline{e}(n). \quad (9)$$

Taking the squared norm and expectations from both sides of Equation 7, one obtains:

$$\begin{aligned} E \left\{ \|\underline{\varepsilon}(n+1)\|^2 \right\} &= E \left\{ \|\underline{\varepsilon}(n)\|^2 \right\} \\ &+ \mu^2 E \left\{ \underline{e}^T(n)B^T(n)B(n)\underline{e}(n) \right\} \\ &- 2\mu E \left\{ \underline{e}^T(n)B^T(n)\underline{\varepsilon}(n) \right\}, \end{aligned} \quad (10)$$

where $B(n) = X(n)W(n)$. Equation 10 can be represented in the form of Equation 11:

$$E \left\{ \|\underline{\varepsilon}(n+1)\|^2 \right\} = E \left\{ \|\underline{\varepsilon}(n)\|^2 \right\} - \Delta\mu, \quad (11)$$

where $\Delta\mu$ is given by:

$$\begin{aligned} \Delta\mu &= -\mu^2 E \left\{ \underline{e}^T(n)B^T(n)B(n)\underline{e}(n) \right\} \\ &+ 2\mu E \left\{ \underline{e}^T(n)B^T(n)\underline{\varepsilon}(n) \right\}. \end{aligned} \quad (12)$$

If $\Delta\mu$ is maximized, then, Mean-Square Deviation (MSD) will undergo the largest decrease from iteration n to iteration $n+1$. The optimum step-size will be found with a derivation of $\Delta\mu$, with respect to μ , equal to zero, $\frac{d\Delta\mu}{d\mu} = 0$;

$$\mu^o(n) = \frac{E \left\{ \underline{e}^T(n)B^T(n)\underline{\varepsilon}(n) \right\}}{E \left\{ \underline{e}^T(n)B^T(n)B(n)\underline{e}(n) \right\}}. \quad (13)$$

Introducing the a priori error vectors:

$$\underline{e}_a(n) = X^T(n)\underline{\varepsilon}(n), \quad (14)$$

it is found, from Equation 7, that the error vector is related to an a priori error vector, via Equation 15:

$$\underline{e}(n) = \underline{e}_a(n) + \underline{v}(n). \quad (15)$$

Assuming the noise sequence, $v(n)$, is identically and independently distributed and statistically independent of the regression data, and by neglecting the dependency of $\underline{\varepsilon}(n)$ on the past noises, the following two sub equations are established from the two parts of Equation 13:

Part I:

$$\begin{aligned} E \left\{ \underline{e}^T(n)B^T(n)\underline{\varepsilon}(n) \right\} \\ &= E \left\{ (\underline{\varepsilon}^T(n)X(n) + \underline{v}^T(n)) (B^T(n)\underline{\varepsilon}(n)) \right\} \\ &= E \left\{ \underline{\varepsilon}^T(n)X(n)B^T(n)\underline{\varepsilon}(n) \right\}. \end{aligned} \quad (16)$$

Table 2. Correspondence between special cases of Equation 1 and various adaptive filtering algorithms.

Algorithm	K	$W(n)$
LMS	1	1
NLMS	1	$\ \underline{x}(n)\ ^{-2}$
AP	$1 < K < M$	$[X^T(n)X(n)]^{-1}$
BLMS	$L \leq M, K = L, X(k)$	I
BNLMS	$L \leq M, K = L, X(k)$	$\alpha(k)$

Part II:

$$\begin{aligned}
& E \{ \underline{e}^T(n) B^T(n) B(n) \underline{e}(n) \} \\
&= E \{ \underline{\varepsilon}^T(n) X(n) B^T(n) B(n) X^T(n) \underline{\varepsilon}(n) \} \\
&+ E \{ \underline{v}^T(n) B^T(n) B(n) \underline{v}(n) \} \\
&= E \{ \underline{\varepsilon}^T(n) X(n) B^T(n) B(n) X^T(n) \underline{\varepsilon}(n) \} \\
&+ \sigma_v^2 \text{Tr} (E \{ B^T(n) B(n) \}) . \tag{17}
\end{aligned}$$

Finally, by defining $C(n) = B(n)X^T(n)$, the optimum step-size in Equation 13 becomes:

$$\mu^o(n) = \frac{E \{ \underline{\varepsilon}^T(n) C^T(n) \underline{\varepsilon}(n) \}}{E \{ \underline{\varepsilon}^T(n) C^T(n) C(n) \underline{\varepsilon}(n) \} + C}, \tag{18}$$

where:

$$C = \sigma_v^2 \text{Tr} (E \{ B^T(n) B(n) \}) . \tag{19}$$

Substituting the $\mu^o(n)$ of Equation 18, instead of μ , in Equation 1, the generic variable step-size update equation that covers VSSNLMS and VSSAPA of [6], as special cases, will be obtained. One must now focus on the development of the VSSBNLMS adaptive algorithm.

VARIABLE STEP-SIZE BLOCK NORMALIZED LMS ADAPTIVE FILTER ALGORITHM

The filter coefficients update equation for BNLMS can be stated as:

$$\underline{h}(k+1) = \underline{h}(k) + \mu X(k) \alpha(k) \underline{e}(k), \tag{20}$$

where k is the block index, $\underline{h}(k)$ the length, M column vector of filter coefficients to be adjusted once after the collection of every block of data samples. $X(k)$ is an $M \times K$ input signal matrix, $\underline{d}(k)$ is an $K \times 1$ vector of desired signal samples and $\underline{e}(k)$ is the error signal vector which are defined by:

$$X(k) = [\underline{x}(kL), \underline{x}(kL-1), \underline{x}(kL-2), \dots, \underline{x}(kL-K+1)], \tag{21}$$

$$\underline{d}(k) = [d(kL), d(kL-1), d(kL-2), \dots, d(kL-K+1)], \tag{22}$$

$$\underline{e}(k) = [e(kL), e(kL-1), e(kL-2), \dots, e(kL-K+1)], \tag{23}$$

where L is the block length and the error signal vector is calculated by;

$$\underline{e}(k) = \underline{d}(k) - X^T(k) \underline{h}(k). \tag{24}$$

There are three possible choices for selecting L :

1. $L = M$, which is the optimal choice from the viewpoint of computational complexity;
2. $L < M$, which offers the advantage of reduced processing delay. Moreover, by making the block size smaller than the filter length, one still has an adaptive filtering algorithm computationally more efficient than the conventional LMS algorithm;
3. $L > M$, which gives rise to redundant operations in the adaptive process, the estimation of the gradient vector (computed over L points) now uses more information than the filter itself.

Selecting $L = M$ is more practical in different applications. For the BLMS adaptive filter algorithm $\alpha(k) = I$. In the case of the BNLMS adaptive algorithm, the $K \times K$ matrix, $\alpha(k)$, is a diagonal matrix with the elements, $\alpha(k) = \|X(k)I_i\|^{-2}$, $i = 0, 1, \dots, K-1$, on the diagonal, where I_i is the column number, i , of the $K \times K$ identity matrix, I . Note that terms $\|X(k)I_i\|^2$ are the signal power estimates.

Based on the above and by comparing Equation 20 to Equation 1, which in turn, was identified as an iterative solution strategy for Equation 5, it is immediately observed that the BNLMS update can be interpreted as an iterative solution strategy applied to the weighted Wiener-Hopf-type equation, according to the selecting parameters from Table 2. To get a better performance in a BNLMS adaptive filter, the VSSBNLMS adaptive filter algorithm is presented, based on the generic VSS update equation, which was described in the previous section. It is pointed out that this is a block adaptive algorithm, i.e. one filter vector update is performed each time that L new samples have entered the system. It means that the step-size will be updated for every block.

To simplify the formulation $\alpha_0(k)$ is defined as a diagonal matrix with elements $\alpha_0(k) = \|X(k)I_i\|^{-1}$, $i = 0, 1, \dots, K-1$ on the diagonal. Therefore, $\alpha_0^T(k)$ is also a diagonal matrix with elements $\alpha_0^T(k) = \|X(k)I_i\|^{-T}$, $i = 0, 1, \dots, K-1$ on the diagonal. It is obvious that $\alpha_0(k) = \alpha_0^T(k)$.

Also, by introducing the $\underline{p}(k)$, $\underline{q}(k)$ and, by using the results from Equation 18;

$$\underline{p}(k) = \alpha_0^T(k) X^T(k) \underline{\varepsilon}(k), \tag{25}$$

$$\underline{q}(k) = X(k) \alpha(k) X^T(k) \underline{\varepsilon}(k), \tag{26}$$

the optimum step-size for the BNLMS adaptive filter is given by:

$$\mu^o(k) = \frac{E \{ \|\underline{p}(k)\|^2 \}}{E \{ \|\underline{q}(k)\|^2 \} + C}, \tag{27}$$

where C is a positive constant and can be approximated from Equation 19:

$$C = \sigma_v^2 \text{Tr} \left(E \left\{ \alpha(k) X^T(k) X(k) \alpha(k) \right\} \right). \quad (28)$$

In calculating the optimum step-size from Equation 27, the main problem is that $\underline{p}(k)$ and $\underline{q}(k)$ are not available, since \underline{h}_t is unknown. Therefore, one needs to estimate these quantities.

By taking expectation from both sides of Equations 25 and 26,

$$E \{ \underline{p}(k) \} = E \{ \alpha_0^T(k) X^T(k) \underline{\varepsilon}(k) \}, \quad (29)$$

$$E \{ \underline{q}(k) \} = E \{ X(k) \alpha(k) X^T(k) \underline{\varepsilon}(k) \}, \quad (30)$$

and by substituting $\underline{e}_a(k) = \underline{e}(k) - \underline{v}(k)$ in Equations 29 and 30, one yields:

$$\begin{aligned} E \{ \underline{p}(k) \} &= E \{ \alpha_0^T(k) \underline{e}_a(k) \} \\ &= E \{ \alpha_0^T(k) (\underline{e}(k) - \underline{v}(k)) \} \\ &= E \{ \alpha_0^T(k) \underline{e}(k) \}, \end{aligned} \quad (31)$$

$$\begin{aligned} E \{ \underline{q}(k) \} &= E \{ X(k) \alpha(k) X^T(k) \underline{\varepsilon}(k) \} \\ &= E \{ X(k) \alpha(k) (\underline{e}(k) - \underline{v}(k)) \} \\ &= E \{ X(k) \alpha(k) \underline{e}(k) \}. \end{aligned} \quad (32)$$

These quantities can be estimated with the recursions, presented in the following equations:

$$\hat{\underline{p}}(k) = \beta \hat{\underline{p}}(k-1) + (1 - \beta) (\alpha_0^T(k) \underline{e}(k)), \quad (33)$$

$$\hat{\underline{q}}(k) = \beta' \hat{\underline{q}}(k-1) + (1 - \beta') (X(k) \alpha(k) \underline{e}(k)), \quad (34)$$

where β and β' are smoothing factors, $(0 < \beta, \beta' < 1)$.

Finally, the recursion for the variable step-size BNLMS (VSSBNLMS) adaptive algorithm (VSS-BNLMS) is given by:

$$\underline{h}(k+1) = \underline{h}(k) + \mu(k) X(k) \alpha(k) \underline{e}(k), \quad (35)$$

where:

$$\mu(k) = \mu_{\max} \frac{\|\hat{\underline{p}}(k)\|^2}{\|\hat{\underline{q}}(k)\|^2 + C}. \quad (36)$$

The step-size changes with the $\|\hat{\underline{p}}(k)\|^2$, $\|\hat{\underline{q}}(k)\|^2$ and the constant, C , which can be approximated from Equation 28. It is clear that C is inversely proportional to SNR. To guarantee the update stability, μ_{\max} is selected less than 2.

COMPUTATIONAL COMPLEXITY

In this section, the computational complexity of the BNLMS and the VSSBNLMS is compared. In Table 3, the number of real multiplications and real additions that are required in the evaluation of specific terms for both BNLMS and VSSBNLMS adaptive filter algorithms are shown. The only difference in the computational complexity between BNLMS and VSSBNLMS is in the $\mu(k)$ term. Table 4 shows the number of real multiplications and real additions that are required in the evaluation of this term. The only difference is $4K + 3M + 2$ multiplications and $2K + 2M - 1$ additions per iteration. It is seen that the cost of BNLMS and VSSBNLMS adaptive filter algorithms is $O(MK)$ operations per iteration.

SIMULATION RESULTS

The theoretical results presented in this paper are justified by several computer simulations in a channel estimation setup. The unknown channel has 8 taps

Table 3. Computational cost of BNLMS and VSSBNLMS adaptive filter algorithms per iteration in terms of the number of real multiplications and real additions.

Term	×		+	
	BNLMS	VSSBNLMS	BNLMS	VSSBNLMS
$X^T(k) \underline{h}(k)$	MK	MK	$K(M-1)$	$K(M-1)$
$d(k) - X^T(k) \underline{h}(k)$	-	-	K	K
$\alpha(k)$	$K(M+1)$	$K(M+1)$	$K(M-1)$	$K(M-1)$
$\alpha(k) \underline{e}(k)$	K	K	-	-
$X(k) \alpha(k) \underline{e}(k)$	MK	MK	$M(K-1)$	$M(K-1)$
$\mu(k)$	-	$4K + 3M + 2$	-	$2K + 2M - 1$
$\mu(k) X(k) \alpha(k) \underline{e}(k)$	M	M	-	-
$\underline{h}(k+1)$	-	-	M	M
Total per iteration	$3MK + 2K + M$	$3MK + 6K + 4M + 2$	$3MK - K$	$3MK + K + 2M - 1$

Table 4. Computational cost of the step-size in VSSBNLMS per iteration in terms of the number of real multiplications and real additions.

Term	\times	$+$
$\alpha_0^T(k)\underline{e}(k)$	K	-
$(1 - \beta)\alpha_0^T(k)\underline{e}(k)$	K	-
$(1 - \beta')(X(k)\alpha(k)\underline{e}(k))$	M	-
$\hat{p}(k)$	K	K
$\hat{q}(k)$	M	M
$\ \hat{p}(k)\ ^2$	K	$K - 1$
$\ \hat{q}(k)\ ^2$	M	$M - 1$
$\mu(k)$	2	1
Total per iteration	$4K + 3M + 2$	$2K + 2M - 1$

and selected at random. Two different types of signal, Gaussian and uniformly distributed signals, are used in forming the input signal, $x(n)$:

$$x(n) = \rho x(n-1) + w(n), \quad (37)$$

which is a first order autoregressive (AR) process with a pole at ρ . For the Gaussian case, $w(n)$ is a white, zero-mean, Gaussian random sequence, having unit variance, and ρ is set to 0.9. As a result, a highly colored Gaussian signal is generated. For the uniform case, $w(n)$ is a uniformly distributed random sequence between -1.0 and 1.0 and ρ is again set to 0.9. Measurement noise, $v(n)$, with $\sigma_v^2 = 10^{-3}$, was added to the noise-free desired signal generated through $d(n) = \underline{h}_t^T \underline{x}(n)$. The adaptive filter and the unknown channel are assumed to have the same number of taps. All the simulations are obtained by ensemble averaging over 200 independent trials. Figures 2 to 7 show the learning curves of BNLMs and VSSBNLMS adaptive filter algorithms. Figures 2 to 4 compare the learning curves of BNLMs and VSSBNLMS adaptive filter algorithms with different block length ($L = 4, 8, 16$) and for highly colored Gaussian input. The ensemble averaged learning curves for VSSBNLMS were obtained with $\beta = 0.99$, $\beta' = 0.99$, $C = 0.001$ and $\mu_{\max} = 1$. In the ordinary BNLMs case, the simulation results were obtained for different step-sizes. Figures 5 to 7 show the learning curves for a highly colored uniform input signal. It can clearly be seen that the VSSBNLMS has a fast convergence rate and a low steady-state mean square error, when compared to the ordinary BNLMs algorithm for both highly colored and uniform input signals.

CONCLUSIONS

In this paper, the generic, variable, step-size adaptive filter was presented. This generic VSS adaptive filter can cover VSSNLMS and VSSAPA adaptive filter

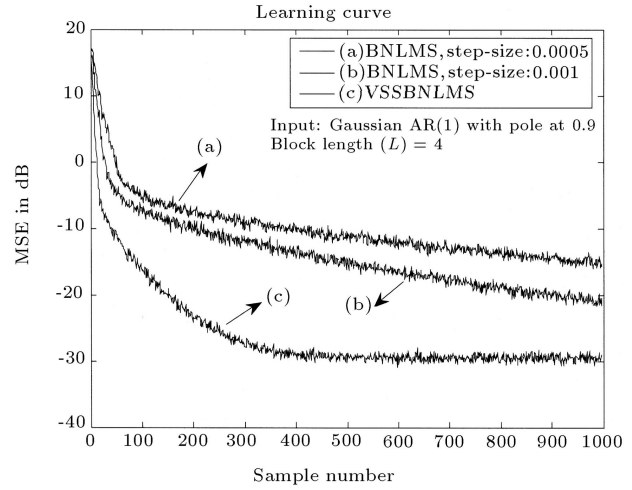


Figure 2. Learning curves of BNLMs with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 4$. Input: Highly colored Gaussian (Gaussian AR(1) with $\rho = 0.9$).

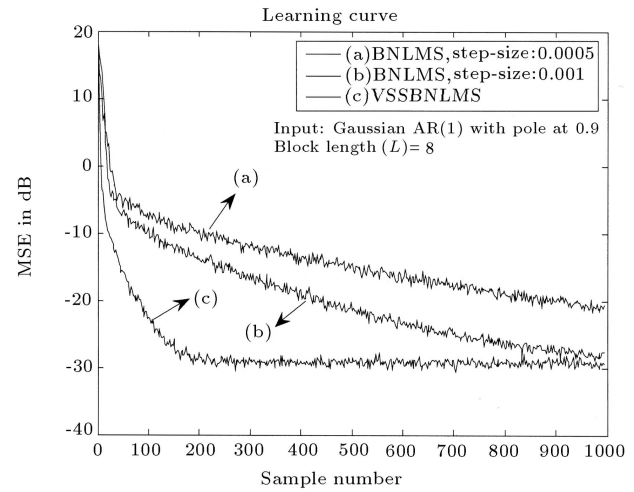


Figure 3. Learning curves of BNLMs with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 8$. Input: Highly colored Gaussian (Gaussian AR(1) with $\rho = 0.9$).

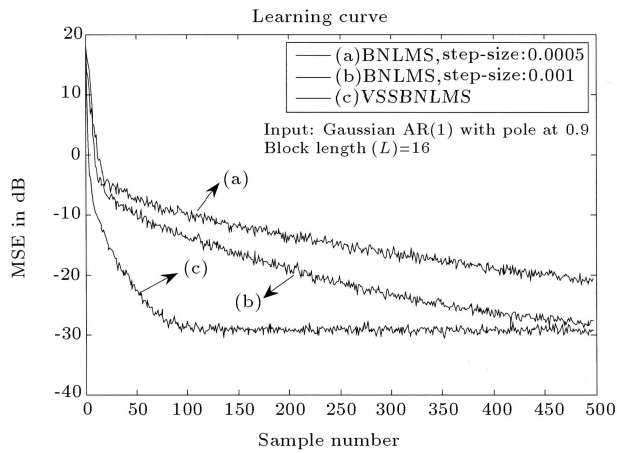


Figure 4. Learning curves of BNLMS with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 16$. Input: Highly colored Gaussian (Gaussian AR(1) with $\rho = 0.9$).

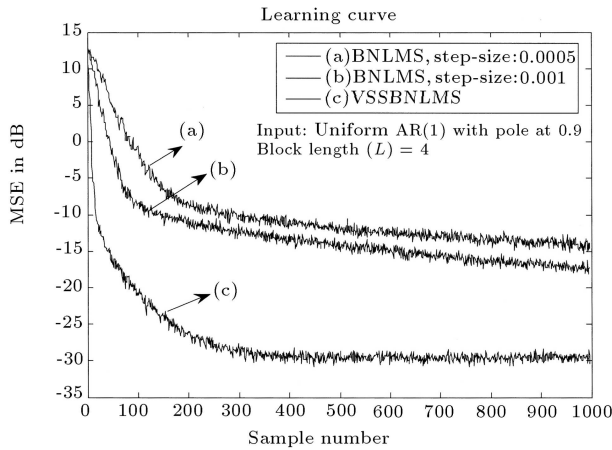


Figure 5. Learning curves of BNLMS with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 4$. Input: Highly colored uniform (uniform AR(1) with $\rho = 0.9$).

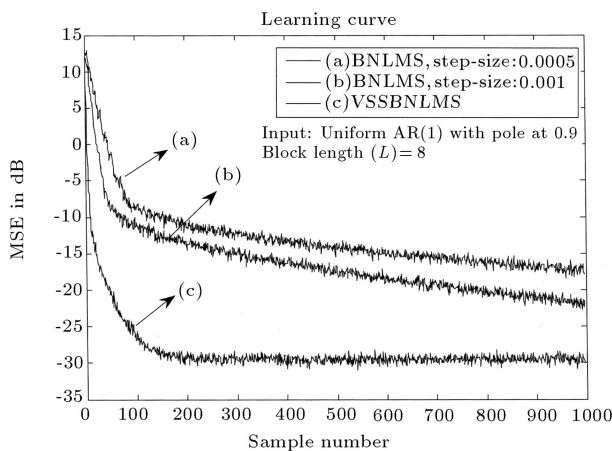


Figure 6. Learning curves of BNLMS with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 8$. Input: Highly colored uniform (uniform AR(1) with $\rho = 0.9$).

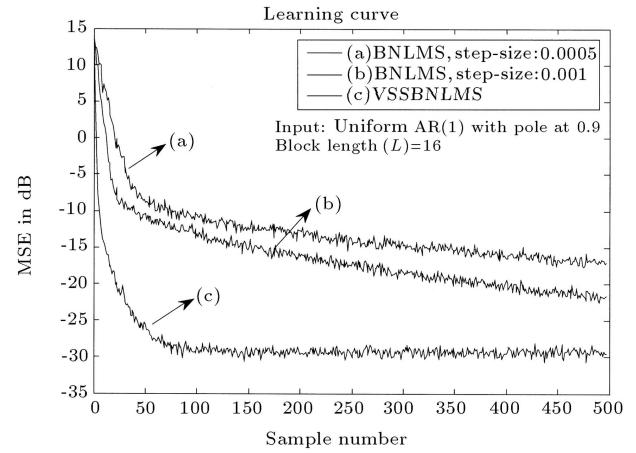


Figure 7. Learning curves of BNLMS with various step-sizes and VSSBNLMS adaptive filter algorithms for $L = 16$. Input: Highly colored uniform (uniform AR(1) with $\rho = 0.9$).

algorithms. Following this, the variable step-size BNLMS, named the VSSBNLMS adaptive filter algorithm, was developed, based on the generic, variable, step-size adaptive filter. The algorithm exhibits fast convergence, while reducing steady-state mean square error, as compared to the ordinary BNLMS adaptive algorithm.

REFERENCES

1. Haykin, S., *Adaptive Filter Theory*, NJ, Prentice-Hall, 4th Edition (2002).
2. Sayed, A.H., *Fundamentals of Adaptive Filtering*, Wiley (2003).
3. Pradhan, S.S. and Reddy, V.E. "A new approach to subband adaptive filtering", *IEEE Trans. Signal Processing*, **47**, pp 655-664 (1999).
4. de Courville, M. and Duhamel, P. "Adaptive filtering in subbands using a weighted criterion", *IEEE Trans. Signal Processing*, **46**, pp 2359-2371 (1998).
5. Lee, K.A. and Gan, W.S. "Improving convergence of the NLMS algorithm using constrained subband updates", *IEEE Signal Processing Letters*, **11**, pp 736-739 (2004).
6. Shin, H.C., Sayed, A.H. and Song, W.J. "Variable step-size NLMS and affine projection algorithms", *IEEE Signal Processing Letters*, **11**, pp 132-135 (Feb. 2004).
7. Husøy, J.H. and Abadi, M.S.E. "A common framework for transient analysis of adaptive filters", in *Proc. Melecon*, Dubrovnik, Croatia, pp 265-268 (May 2004).
8. Husøy, J.H. "On performance analysis of transform domain adaptive filters: A unified perspective", in *15th Intl. Czech-Slovak Sci. Conf. Radioelektronika*, Brno, Czech Republic, pp 96-99 (May 2005).
9. Husøy, J.H. "A novel interpretation and performance analysis of subband adaptive filter algorithms", in

- Proc. of Seventh Intl. Symp. on Signals, Circuits and Systems*, **2**, pp 561-564, Iasi, Romania (2005).
10. Husøy, J.H. "A unified framework for adaptive filtering", in *Proc. NORSIG*, Bergen, Norway, CD-ROM, ISBN 82-993158-5-9 (Oct. 2003)
 11. Husøy, J.H. "Adaptive filters viewed as iterative linear equation solvers", in *Ser. Lecture Notes in Computer Science*, Springer-Verlag, **3401**, pp 320-327 (2005).
 12. Husøy, J.H. and Abadi, M.S.E. "Transient analysis of adaptive filters using a general framework", in *AUTOMATIKA, Journal for Control, Measurement, Electronics, Computing and Communications*, **3-4/45**, pp 121-127 (2004).
 13. Saad, Y., *Iterative Methods for Sparse Linear Systems*, PWS Publishing (1996).