# BASIS EXPANSION ADAPTIVE FILTERS FOR TIME-VARYING SYSTEM IDENTIFICATION

*Luca Rugini and Geert Leus*

Delft University of Technology
Faculty of Electrical Engineering, Mathematics, and Computer Science
Mekelweg 4, 2628 CD Delft, The Netherlands
{l.rugini,g.leus}@tudelft.nl

## ABSTRACT

In this paper, we extend the concept of block adaptive filters to what we call *basis expansion adaptive filters*. While in block adaptive filters the system is assumed to be constant within a block, our basis expansion adaptive filters model the time variation of the system within a block by a set of basis functions. This allows us to improve the tracking performance of block adaptive filters considerably. We focus on stochastic gradient type of adaptive filters, although extensions to other types of adaptive filters can be envisioned.

***Index Terms***— Basis expansion model, block adaptive filters, linear time-varying systems

## 1. INTRODUCTION

Adaptive filtering is a widespread and largely investigated option to tackle the problem of system identification [1]. In the last decades, many adaptive algorithms have been proposed ranging from the simple stochastic gradient type of algorithms, lead by the least mean squares (LMS) family, to more elaborated algorithms like those belonging to the recursive least squares (RLS) class [1, 2, 3].

A common feature of most of these algorithms is that they have been initially derived assuming a time-invariant system model. In case of a time-varying model, adaptive algorithms like LMS try to track the time variation of the system impulse response, as analyzed in [4, 5, 6]. However, this tracking is done on a sample-by-sample basis, by using limited information about the past. This past information can be better taken into account by using a block of data rather than one single sample. As a consequence, to track time-varying systems, a block-by-block algorithm could potentially be more beneficial than a sample-by-sample approach, especially if the blocks are overlapping.

Traditionally, block adaptive filtering techniques have been developed to reduce complexity in the time-invariant case, by exploiting fast convolution methods. For example, the block LMS (BLMS) algorithm [3, 7] uses an average gradient over a block of samples to estimate the impulse response. In case of non-overlapping blocks, BLMS has approximately the same steady-state performance of LMS [7, 8], with computational savings due to the shared operations. However, if the blocks are overlapping, as in the sliding-window LMS (SW-LMS), the convergence and the steady-state performance of block adaptive algorithms improve, at the expense of an increased complexity caused by the repeated processing of the same data [3]. The convergence of both BLMS and SW-LMS can further be improved by allowing a time-varying step size, leading to the optimum block adaptive (OBA) and the OBA shifting (OBAS) algorithms [9]. However, the tracking performance of all these block algorithms highly suffers from the averaging operation, since they ignore the time variation within the block and produce a non-negligible delay.

To overcome the tracking problems of conventional block adaptive filters, we try to exploit a suitable model for the time variation within a block. One example is the basis expansion model (BEM), where the time variability is expressed as a linear combination of basis functions, which could be complex exponentials, polynomials, and so on [10, 11, 12, 13]. Specifically, the authors of [12] propose an RLS algorithm that combines recursive block processing with a polynomial model for the time variability.

In this paper, we propose block adaptive stochastic gradient algorithms that incorporate a BEM for the time variation of the system impulse response. The proposed algorithm, called basis expansion LMS (BE-LMS), is a generalization of many already-known adaptive algorithms like normalized LMS (NLMS), OBA, and OBAS [9]. We show by simulations that our proposed BE-LMS is able to track the system time variation better than the NLMS, while maintaining good convergence properties in case of overlapping blocks. To even further improve the tracking capabilities, we also propose another BEM-based adaptive algorithm that has a nice connection with the matrix generalization of the momentum LMS (MLMS) algorithm [14].

## 2. SYSTEM MODEL

The system under consideration is a linear time-varying system with additive noise, whose input-output relation is expressed by

$$y_n = \sum_{l=0}^{L-1} h_{n,l} x_{n-l} + e_n, \qquad (1)$$

where $y_n$ is the output, $h_{n,l}$ is the linear time-varying system impulse response, assumed finite with $L$ taps, $x_n$ is the input, and $e_n$ is the additive noise. Given $x_n$ and $y_n$, our goal is to find a linear time-varying filter $\hat{h}_{n,l}$ such that

$$\hat{y}_n = \sum_{l=0}^{L-1} \hat{h}_{n,l} x_{n-l}$$

is as close as possible to $y_n$. The filter $\hat{h}_{n,l}$ can then be considered as an estimate of the system impulse response $h_{n,l}$.

To describe the proposed adaptive algorithms, we will first reshape the system input-output relation, and then we will make use of the limited time-varying behavior of the system. Let us start by rewriting (1) as $y_n = \mathbf{x}_n^T \mathbf{h}_{n,:} + e_n$, where $\mathbf{x}_n = [x_n, \ldots, x_{n-L+1}]^T$

This research was supported in part by NWO-STW under the VIDI program (DTC.6577) and the VICI program (DTC.5893).

978-1-4244-1714-8/07/$25.00 

and $\mathbf{h}_{n,:} = [h_{n,0}, \ldots, h_{n,L-1}]^T$. Stacking $y_n$ over a block of $N$ samples every $K$ samples, we obtain

$$\mathbf{y}(k) = \mathbf{X}^T(k)\mathbf{h}(k) + \mathbf{e}(k), \quad (2)$$

where $\mathbf{y}(k) = [y_{kK}, \ldots, y_{kK+N-1}]^T$, $\mathbf{X}(k) = \text{diag}(\mathbf{x}_{kK}, \ldots, \mathbf{x}_{kK+N-1})$ is an $LN \times N$ block-diagonal matrix, $\mathbf{h}(k) = [\mathbf{h}_{kK,:}^T, \ldots, \mathbf{h}_{kK+N-1,:}^T]^T$, and $\mathbf{e}(k) = [e_{kK}, \ldots, e_{kK+N-1}]^T$. Hence, we actually formulate a sliding-window data model, where $K$ determines how many samples we shift the $N$-length window in every step.

By exploiting now the limited time-varying behavior of the system, each tap of the system impulse response $h_{n,l}$ can be modeled, over a time range $n \in \{kK, \ldots, kK+N-1\}$, as a superposition of $Q \leq N$ functions $\{b_{n-kK,q}\}_{q=0}^{Q-1}$ weighted by the coefficients $\{c_{q,l}(k)\}_{q=0}^{Q-1}$, as expressed by

$$h_{n,l} = \sum_{q=0}^{Q-1} b_{n-kK,q}c_{q,l}(k).$$

This model is usually referred to as the basis expansion model (BEM) [10, 11]. In matrix-vector notation, the BEM becomes

$$\mathbf{h}_{:,l}(k) = \bar{\mathbf{B}}\mathbf{c}_l(k), \quad (3)$$

where $\mathbf{h}_{:,l}(k) = [h_{kK,l}, \ldots, h_{kK+N-1,l}]^T$, $[\bar{\mathbf{B}}]_{m,q} = b_{m,q}$ where the indices start from zero, and $\mathbf{c}_l(k) = [c_{0,l}(k), \ldots, c_{Q-1,l}]^T$. Note that the basis matrix $\bar{\mathbf{B}}$ is independent of the time shift $kK$. For simplicity reasons, we assume that $\bar{\mathbf{B}}$ is an isometry, i.e., $\bar{\mathbf{B}}^H\bar{\mathbf{B}} = \mathbf{I}_Q$. Good choices for the BEM functions could be polynomials or complex exponentials, leading to a polynomial (POL) BEM [12] or a complex exponential (CE) BEM [10, 11], respectively. In these cases, $\bar{\mathbf{B}}$ is derived as an orthonormalization of a matrix $\tilde{\mathbf{B}}$, where $[\tilde{\mathbf{B}}]_{m,q} = (m - N/2)^q$ for the POL-BEM and $[\tilde{\mathbf{B}}]_{m,q} = e^{j2\pi mq/(\kappa N)}$ for the CE-BEM, with $\kappa$ a positive integer that controls the frequency separation of the basis functions [13].

We can express $\mathbf{h}(k)$ as $\mathbf{h}(k) = [\mathbf{h}_{kK,:}^T, \ldots, \mathbf{h}_{kK+N-1,:}^T]^T = [\mathbf{I}_N \otimes \mathbf{i}_0, \ldots, \mathbf{I}_N \otimes \mathbf{i}_{L-1}][\mathbf{h}_{:,0}^T(k), \ldots, \mathbf{h}_{:,L-1}^T(k)]^T$, where $\mathbf{i}_l$ is the $l$th column of $\mathbf{I}_L$, with indices starting from zero, and $\otimes$ represents the Kronecker product. We now use (3) to obtain $\mathbf{h}(k) = [\mathbf{I}_N \otimes \mathbf{i}_0, \ldots, \mathbf{I}_N \otimes \mathbf{i}_{L-1}](\mathbf{I}_L \otimes \bar{\mathbf{B}})\mathbf{c}(k) = [\bar{\mathbf{B}} \otimes \mathbf{i}_0, \ldots, \bar{\mathbf{B}} \otimes \mathbf{i}_{L-1}]\mathbf{c}(k) = \mathbf{B}\mathbf{c}(k)$, where $\mathbf{c}(k) = [\mathbf{c}_0^T(k), \ldots, \mathbf{c}_{L-1}^T(k)]^T$ and $\mathbf{B} = [\bar{\mathbf{B}} \otimes \mathbf{i}_0, \ldots, \bar{\mathbf{B}} \otimes \mathbf{i}_{L-1}]$. Finally, we can rephrase (2) as

$$\mathbf{y}(k) = \mathbf{X}^T(k)\mathbf{B}\mathbf{c}(k) + \mathbf{e}(k). \quad (4)$$

Since the system impulse response $h_{n,l}$ is assumed to follow a BEM, we put a similar BEM constraint also on the filter $\hat{h}_{n,l}$, thereby reducing the number of unknowns that we want to estimate. Hence, following similar steps as before, we obtain

$$\hat{\mathbf{y}}(k) = \mathbf{X}^T(k)\hat{\mathbf{h}}(k) = \mathbf{X}^T(k)\mathbf{B}\hat{\mathbf{c}}(k), \quad (5)$$

where $\hat{\mathbf{y}}(k)$, $\hat{\mathbf{h}}(k)$, and $\hat{\mathbf{c}}(k)$ represent the estimated versions of $\mathbf{y}(k)$, $\mathbf{h}(k)$, and $\mathbf{c}(k)$, respectively.

## 3. BASIS EXPANSION LEAST MEAN SQUARES

The basic goal of our basis expansion adaptive algorithms is to get $\hat{\mathbf{y}}(k)$ in (5) as close as possible to $\mathbf{y}(k)$ in (4), given $\mathbf{X}(k)$ and $\mathbf{y}(k)$. Basically, we want to minimize $E(\|\epsilon(k)\|^2)$, where $\epsilon(k)$ is the error signal given by:

$$\epsilon(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k) = \mathbf{y}(k) - \mathbf{X}^T(k)\mathbf{B}\hat{\mathbf{c}}(k).$$

By performing this minimization over the BEM coefficients $\hat{\mathbf{c}}(k)$ of the filter $\hat{\mathbf{h}}(k)$, we obtain the Wiener solution

$$\hat{\mathbf{c}}_{opt}(k) = E(\mathbf{B}^H\mathbf{X}^*(k)\mathbf{X}^T(k)\mathbf{B})^{-1}E(\mathbf{B}^H\mathbf{X}^*(k)\mathbf{y}(k)).$$

Clearly, in the absence of a basis expansion modeling error, the Wiener solution $\hat{\mathbf{c}}_{opt}(k)$ is equal to the actual BEM coefficients $\mathbf{c}(k)$ of the system impulse response $\mathbf{h}(k)$. To find this solution adaptively, we follow a standard stochastic gradient descent approach, leading to what we will label the *basis expansion least mean squares (BE-LMS)* algorithm. The BE-LMS algorithm is described by

$$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k), \quad (6)$$

where $\mu(k)$ is a possibly time-varying step size. In our BE-LMS, we implicitly assume that the optimal $\mu(k)$ is used. More specifically, we can derive the step size that maximizes the convergence speed in a similar manner as in [9]. This maximization leads to

$$\mu(k) = \frac{\|\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k)\|^2}{\|\mathbf{X}^T(k)\mathbf{B}\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k)\|^2}.$$

In practice, a small positive constant $\varepsilon$ is added to the denominator of $\mu(k)$ to prevent a possible zero of the denominator [2].

We now show the connections of our BE-LMS algorithm with the vast literature on adaptive algorithms. First of all, it is easy to see that when the block size $N = 1$, the BEM can not really be exploited and only $Q = 1$ is possible. In this case, the BE-LMS boils down to the standard NLMS algorithm, and to the standard LMS algorithm if in addition the step size is kept constant. Another interesting connection occurs when we model the system impulse response as being constant within a window of $N$ samples. In this case, we have a single BEM function, i.e., $Q = 1$, and we model it as a constant function, i.e., $\bar{\mathbf{B}} = [1/\sqrt{N}, \ldots, 1/\sqrt{N}]^T$. It is then easy to show that the BE-LMS coincides with the OBAS algorithm [9], and with the SW-LMS algorithm if in addition the step size is kept constant [3]. In the special case where $K = N$, i.e., we recompute a new set of BEM coefficients every window of $N$ samples, the BE-LMS becomes the OBA algorithm [9], and the BLMS algorithm in case $\mu(k) = \mu$.

The OBAS or SW-LMS have basically been introduced to improve the convergence behavior of NLMS and LMS, but, because of the large window size $N$, they suffer from a worse tracking performance. The BE-LMS for $Q > 1$ solves this problem by modeling the time variation of the system impulse response within the considered window with a BEM. On the other hand, the OBA and BLMS algorithms were mainly introduced to further reduce the complexity of NLMS and LMS, but they naturally suffer in convergence and tracking due to the large shift $K$. The BE-LMS for $Q > 1$ experiences a similar convergence and tracking behavior when the shift parameter $K$ is large. To further improve the performance of BE-LMS, we also propose to include some prior knowledge on the time-varying behavior of $\mathbf{c}(k)$, which actually leads to a BE-LMS algorithm with an acceleration feature that will be called *momentum*.

## 4. BASIS EXPANSION MOMENTUM LMS

Obviously, the time-varying behavior of $\mathbf{c}(k)$ is determined by the time variation of the system. However, part of the transition from $\mathbf{c}(k-1)$ to $\mathbf{c}(k)$ also depends on how the BEM functions of two successive blocks are related. In other words, when the blocks are overlapping, $\mathbf{h}(k)$ shares some elements with $\mathbf{h}(k-1)$, which clearly introduces a relationship between the BEM coefficients $\mathbf{c}(k)$ and

$\mathbf{c}(k-1)$. Therefore, by defining $\mathbf{J}$ as an $N \times N$ shift matrix, with ones along the $K$th upper diagonal and zeros elsewhere, we can model the transition from $\mathbf{h}(k-1)$ to $\mathbf{h}(k)$ as

$$\mathbf{h}(k) = (\mathbf{J} \otimes \mathbf{I}_L)\mathbf{h}(k-1) + \mathbf{g}(k),$$

with $\mathbf{g}(k) = [\mathbf{0}^T_{(N-K)L\times 1}, \mathbf{h}^T_{kK+N,:}, \ldots, \mathbf{h}^T_{(k+1)K+N-1,:}]^T$. Thus, we can write

$$\mathbf{c}(k) = \mathbf{B}^H(\mathbf{J} \otimes \mathbf{I}_L)\mathbf{B}\mathbf{c}(k-1) + \mathbf{B}^H\mathbf{g}(k) = \mathbf{A}\mathbf{c}(k-1) + \mathbf{d}(k),$$

where $\mathbf{A} = \mathbf{B}^H(\mathbf{J} \otimes \mathbf{I}_L)\mathbf{B}$ and $\mathbf{d}(k) = \mathbf{B}^H\mathbf{g}(k)$. Looking at the right-hand-side of the equation above, the first term can be interpreted as a prediction of $\mathbf{c}(k)$, whereas the second term can be viewed as the corresponding prediction error. Hence, by exploiting this relationship in (4), we obtain

$$\mathbf{y}(k) = \mathbf{X}^T(k)\mathbf{B}\mathbf{A}\mathbf{c}(k-1) + \mathbf{X}^T(k)\mathbf{B}\mathbf{d}(k) + \mathbf{e}(k).$$

Now we can consider $\mathbf{X}^T(k)\mathbf{B}\mathbf{A}\mathbf{c}(k-1)$ as a bias on $\mathbf{y}(k)$. By subtracting this bias term, we obtain

$$\mathbf{y}(k) - \mathbf{X}^T(k)\mathbf{B}\mathbf{A}\mathbf{c}(k-1) = \mathbf{X}^T(k)\mathbf{B}\mathbf{d}(k) + \mathbf{e}(k), \quad (7)$$

which is similar to the input-output relation expressed by (4). As a consequence, we can update the prediction error $\mathbf{d}(k)$ in a similar way as before, leading to

$$\hat{\mathbf{d}}(k+1) = \hat{\mathbf{d}}(k) + \mu(k)\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k), \quad (8)$$

where $\epsilon(k)$ is defined as before with

$$\hat{\mathbf{c}}(k) = \mathbf{A}\mathbf{c}(k-1) + \hat{\mathbf{d}}(k). \quad (9)$$

It is worth noting that $\hat{\mathbf{c}}(k)$ in (9) depends on the true model coefficients $\mathbf{c}(k-1)$, and hence is not available. As a consequence, we replace $\mathbf{c}(k-1)$ by its estimated version $\hat{\mathbf{c}}(k-1)$, leading to

$$\hat{\mathbf{c}}(k) = \mathbf{A}\hat{\mathbf{c}}(k-1) + \hat{\mathbf{d}}(k). \quad (10)$$

To get more insight into this updating formula, let us plug (10) into (8) to obtain

$$\hat{\mathbf{c}}(k+1) - \mathbf{A}\hat{\mathbf{c}}(k) = \hat{\mathbf{c}}(k) - \mathbf{A}\hat{\mathbf{c}}(k-1) + \mu(k)\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k),$$

or equivalently

$$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)\mathbf{B}^H\mathbf{X}^*(k)\epsilon(k) + \mathbf{A}(\hat{\mathbf{c}}(k) - \hat{\mathbf{c}}(k-1)).$$

Clearly, this equation is formally similar to (6), but with the additional term $\mathbf{A}(\hat{\mathbf{c}}(k) - \hat{\mathbf{c}}(k-1))$. Such a procedure greatly resembles the momentum LMS (MLMS) algorithm [14], which was introduced to speed up the convergence and the tracking of LMS. This is why we label this algorithm as the basis expansion momentum LMS (BE-MLMS) algorithm. In the simulations section, we will show that in some cases the BE-MLMS algorithm exhibits a better tracking behavior than BE-LMS.

It should be observed that a straightforward application of the momentum approach to the BE-LMS would lead to a different algorithm with respect to our BE-MLMS. Indeed, in the standard momentum approach, the step size is constant and, more importantly, the fixed matrix $\mathbf{A}$ is replaced by a scalar design parameter $\alpha$ [14]. A good value for $\alpha$ would be rather difficult to determine; however, in the proposed set-up, we do not have to make this difficult choice, and an intuitively pleasing value is selected for the matrix $\mathbf{A}$.

## 5. SIMULATION RESULTS

In order to test the proposed block adaptive algorithms, we consider the estimation of a linear time-varying channel driven by a white input sequence of quadrature phase-shift keying (QPSK) symbols, assumed known at the receiver. In a more practical case, only pilot symbols would be known, and the data symbols would be used in a decision-directed way. We assume a multipath FIR channel with $L = 8$ independent taps following a Jakes' Doppler spectrum. The additive noise $e_n$ is assumed white and Gaussian. The signal-to-noise ratio (SNR) is equal to 30 dB. For all the adaptive algorithms, $\varepsilon = 0.0001$ is added to the step-size denominator. To compare the performance of the adaptive algorithms, we look at the mean-squared error (MSE) of the channel estimate. In the MSE computation, we include only the new elements and exclude those elements already computed in the previous overlapping block. As a consequence, we define the MSE as

$$MSE = \frac{1}{KL}E(\|[\hat{\mathbf{h}}(k) - \mathbf{h}(k)]_{L(N-K)+1:LN}\|^2),$$

where $[\mathbf{x}]_{a:b}$ stands for the subvector of $\mathbf{x}$ from the $a$th to the $b$th element.

Figure 1 shows the MSE as a function of the maximum Doppler spread, normalized to the sampling period, when $N = 64$ and $K = 1$. To obtain the simulated MSE, the results are averaged over 10 long channel realizations, after transient effects. Because of the gradient averaging, the OBAS algorithm (or equivalently the BE-LMS algorithm with $Q = 1$) outperforms the NLMS algorithm in the time-invariant case (no Doppler spread). However, the performance of OBAS rapidly degrades for increasing Doppler spread, because more basis functions are necessary to model the time variability of the channel. Similarly, the performance of the NLMS algorithm significantly worsens when the time variation increases, but less than for OBAS. This is due to the smaller lag with respect to OBAS. On the contrary, the BE-LMS algorithm with $Q > 1$ is able to better track the channel variability, and hence its performance improves with respect to the other algorithms. Obviously, for increasing Doppler spread, the best performance is obtained by BE-LMS with an increasing number of basis functions $Q$, which allows to capture the increasing time variability. This explains why $Q = 2$ is optimal only for a relatively small time variability, and worsens for higher Doppler spreads.

Figure 2 illustrates the steady-state performance of the proposed BE-LMS for time-invariant channels as a function of the block size $N$, when $K = 1$. As expected, an increase of the block size produces an improvement for all the block-based algorithms. When the block size is sufficiently large, the BE-LMS algorithms with $Q > 1$ are able to outperform the NLMS algorithm.

In Figure 3, the prediction effect of the BE-MLMS algorithm is displayed using a simulation example, which compares the real part of one tap of the true channel with that of the estimated channel. In this case, $N = 256$, $K = 64$, and the normalized Doppler spread is equal to $0.002$. It is evident that the BE-MLMS could be a useful alternative to the BE-LMS in order to track a time-varying channel when $K > 1$.

## 6. CONCLUSIONS

In this paper, we have developed some block adaptive algorithms that are suitable for the identification of linear time-varying systems. The proposed BE-LMS algorithm, which extends the OBAS approach

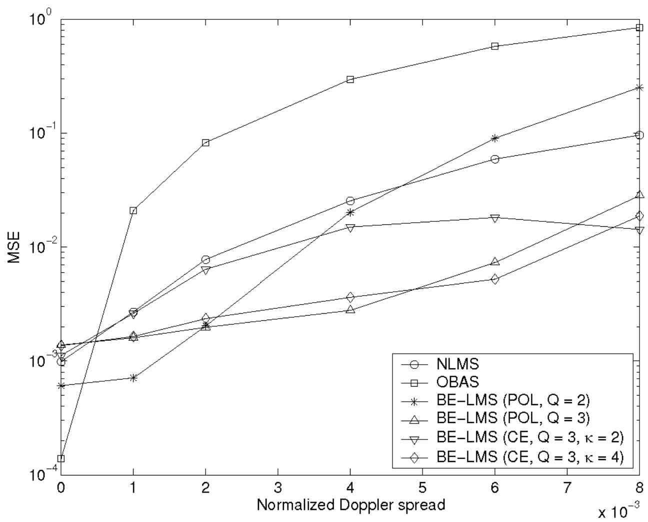


**Fig. 1**. MSE as a function of the normalized Doppler spread.

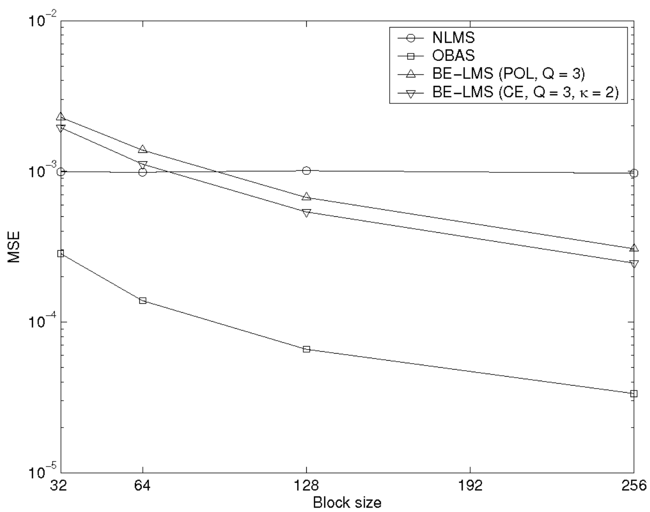


**Fig. 2**. MSE as a function of the block size in the time-invariant case.

to time-varying scenarios, is able to track the system impulse response by conveniently exploiting a basis expansion model of the time variability. As a result, the BE-LMS algorithm outperforms other adaptive algorithms like NLMS and OBAS. In addition, we have pointed out how to exploit the block overlapping in order to obtain a momentum-like block adaptive algorithm called BE-MLMS. Additional features of the proposed algorithms, such as the choice of basis functions and their number, as well as a theoretical analysis of the excess MSE, are currently under investigation.

## 7. REFERENCES

[1] S. Haykin, *Adaptive Filter Theory,* 2nd Ed., Prentice-Hall, 1991.

[2] A. H. Sayed, *Fundamentals of Adaptive Filtering,* Wiley, 2003.

[3] G.-O. Glentis, K. Berberidis, and S. Theodoridis, "Efficient least squares adaptive algorithms for FIR transversal filtering," *IEEE Signal Process. Mag.,* vol. 16, pp. 13-41, July 1999.

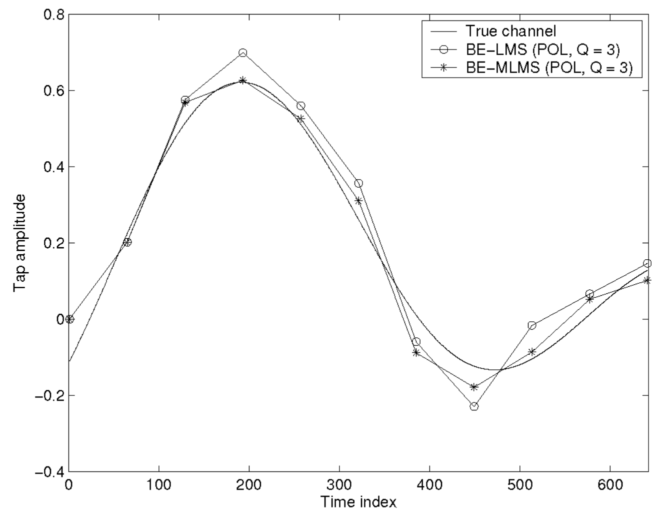


**Fig. 3**. Example of channel estimate by the BE-MLMS algorithm.

[4] E. Eweda, "Comparison of RLS, LMS, and sign algorithm for tracking randomly time-varying channels," *IEEE Trans. Signal Process.,* vol. 42, pp. 2937-2944, Nov. 1994.

[5] B. Farhang-Boroujeny and S. Gazor, "Performance of LMS-based adaptive filters in tracking a time-varying plant," *IEEE Trans. Signal Process.,* vol. 44, pp. 2868-2871, Nov. 1996.

[6] S. Haykin, A. H. Sayed, J. R. Zeidler, P. Yee, and P. C. Wei, "Adaptive tracking of linear time-variant systems by extended RLS algorithms", *IEEE Trans. Signal Process.,* vol. 45, pp. 1118-1128, May 1997.

[7] G. A. Clark, S. K. Mitra, and S. R. Parker, "Block implementation of adaptive digital filters," *IEEE Trans. Acoustics, Speech, Signal Process.,* vol. 29, pp. 744-752, June 1981.

[8] A. Feuer, "Performance analysis of the block least mean square algorithm," *IEEE Trans. Circuits and Systems,* vol. 32, pp. 960-963, Sept. 1985.

[9] W. B. Mikhael and F. H. Wu, "Fast algorithms for block FIR adaptive digital filtering," *IEEE Trans. Circuits and Systems,* vol. 34, pp. 1152-1160, Oct. 1987.

[10] M. K. Tsatsanis and G. B. Giannakis, "Modelling and equalization of rapidly fading channels," *Int. J. Adapt. Control Signal Process.,* vol. 10, pp. 159-176, Mar. 1996.

[11] G. B. Giannakis and C. Tepedelenlioğlu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proc. IEEE,* vol. 86, pp. 1969-1986, Oct. 1998.

[12] D. K. Borah and B. D. Hart, "Frequency-selective fading channel estimation with a polynomial time-varying channel model," *IEEE Trans. Commun.,* vol. 47, pp. 862-873, June 1999.

[13] G. Leus, "On the estimation of rapidly time-varying channels," *Proc. of the European Signal Processing Conference (EUSIPCO 2004),* pp. 2227-2230, Vienna, Austria, Sept. 2004.

[14] S. Roy and J. J. Shynk, "Analysis of the momentum LMS algorithm," *IEEE Trans. Acoustics, Speech, Signal Process.,* vol. 38, pp. 2088-2098, Dec. 1990.