FACTA UNIVERSITATIS (NIŠ)
SER.: ELEC. ENERG. vol. 17, April 2004, 21-32

Variable Step-Size LMS Adaptive Filters for CDMA Multiuser Detection

Karen Egiazarian, Pauli Kuosmanen, and Radu Ciprian Bilcu

Abstract: Due to its simplicity the adaptive Least Mean Square (LMS) algorithm is widely used in Code-Division Multiple access (CDMA) detectors. However its convergence speed is highly dependent on the eigenvalue spread of the input covariance matrix. For highly correlated inputs the LMS algorithm has a slow convergence which require long training sequences and therefore low transmission speeds. Another drawback of the LMS is the trade-off between convergence speed and steady-state error since both are controlled by the same parameter, the step-size. In order to eliminate these drawbacks, the class of Variable Step-Size LMS (VSSLMS) algorithms was introduced. In this paper, we study the behavior of some algorithms belonging to the class of VSSLMS for training based multiuser detection in a CDMA system. We show that the proposed Complementary Pair Variable Step-Size LMS algorithms highly increase the speed of convergence while reducing the trade-off between the convergence speed and the output error.

Keywords: Code-division multiple access, adaptive filters, least mean square, variable step-size.

1 Introduction

Code-Division Multiple-Access (CDMA) using the direct-sequence (DS) spread-spectrum signaling has gained increased interest for application in telecommunication systems. Some of the main advantages of the DS/CDMA technique are: the ability of asynchronous operation, a better channel usage compared with other

---

Manuscript received February 3, 2004. An earlier version of this paper was presented at the 6th Int. Conf. on Telecommunications in Modern Satellite, Cable and Broadcasting Services, TELSIKS 2003, October 1-3, 2003, 18 000 Niš, Serbia.

The authors are with the Institute of Signal Processing, Tampere University of Technology, Korkeakoulunkatu 1, 33720, Tampere, Finland, (e-mail: `[karen,pqo,bilcu]@cs.tut.fi`).

techniques that allow a single user to be transmitted over the channel at a certain time and their ability to operate in the presence of narrow band communication systems. When a given user is demodulated in a DS/CDMA system, two types of interferences must be minimized, namely the wide band multiple access interference (MAI) and the Narrow Band Interference (NBI), as well as, the channel noise. The MAI is caused by other spread spectrum users into the channel while the NBI interference is caused by other conventional communication systems.

Among other demodulation techniques, the adaptive methods has been successfully applied to reduce both the MAI and NBI interferences in DS/CDMA systems. When the spreading code and the channel parameters of the desired user are known or can be estimated, the blind adaptive detectors can be easily used [1]- [7], whereas in absence of these information the trained based implementations are preferred [8]-[10].

In the case of the trained based systems a known training sequence is transmitted which is used to tune the coefficients of the adaptive filter before the actual data is send. The well known adaptive algorithm used in both blind and training based demodulators is the Least Mean Square algorithm which has the advantage of having a simple implementation and low computational complexity. However, the main disadvantages of the LMS algorithm are its slow convergence when operate with highly correlated input signals and the tradeoff between the convergence speed and the output error [12]. In order to reduce these disadvantages many of its variants where introduced in the open literature, such as, the class of Variable Step-Size LMS (VSSLMS) algorithms [13]- [15].

In this paper, we analyze the behavior of different VSSLMS adaptive algorithms for the problem of multiuser detection in synchronous CDMA systems. We show, by means of simulations, that the Complementary Variable Step-Size LMS (CPVSLMS) adaptive algorithm proposed by the authors in [15] possess a faster convergence speed than other known algorithms, while reducing or eliminating the tradeoff between convergence speed and steady-state output error.

The paper is organized as follows: in Section 2 the DS/CDMA signal model is summarized and the adaptive trained based single user detector using the LMS adaptive filter is shortly reviewed, in Section 3 some variable step-sizes adaptive algorithms are described, the simulation results are presented in Section 4 and Section 5 concludes the paper.

2 Theoretical Considerations

For the sake of simplicity we consider a synchronous CDMA system in which a number of K users transmit over a single-path time-invariant channel. The processing gain is denoted by N, the attenuation of each user data are denoted by a_k and the data symbols transmitted by all users are aligned in time. The received signal sampled at chip rate can be written in vector form as follows:

$$\mathbf{r}(n) = \mathbf{SAd}(n) + \mathbf{v}(n), \tag{1}$$

where the j^{th} column of $\mathbf{S}$ represents the received spreading code of the j^{th} user, the vector $\mathbf{d}(n) = [d_1(n), \ldots d_K(n)]^T$ contains the data symbols transmitted by all users at the time instant n, the $N \times 1$ vector $\mathbf{v}$ is the sampled channel noise and the $K \times K$ matrix $\mathbf{A}$ is given by:

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & 0 & \ldots & 0 \\ 0 & a_2 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & a_K \end{bmatrix}$$

Assuming that the desired user is user 1, a block diagram of a trained based detector using the standard LMS adaptive algorithm is depicted in Fig. 1, where $\mathbf{r}(n)$ is the input vector described in Eq. (1), $\hat{\mathbf{h}}(n) = \left[\hat{h}_1(n), \ldots, \hat{h}_N(n)\right]^T$ is the $N \times 1$ vector containing the coefficients of the demodulator, $d_1(n)$ is the known desired sequence that is the same as the data sequence transmitted by the user 1 and $e(n)$ is the output error.

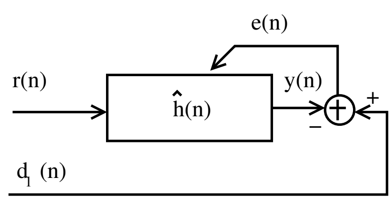

Fig. 1. Block diagram of an adaptive detector using the LMS algorithm.

The LMS adaptive algorithm used to train the coefficients of the adaptive filter $\hat{\mathbf{h}}(n)$ can be described by the following steps:

1. Compute the output of the adaptive filter $\hat{\mathbf{h}}(n)$:

$$y(n) = \hat{\mathbf{h}}^T(n)\mathbf{r}(n) = \sum_{i=1}^{N} \hat{h}_i(n) r_i(n), \tag{2}$$

where $r_i(n)$ is the i^{th} element of the vector $\mathbf{r}(n)$ in Eq. (1).

2. Compute the output error:

$$e(n) = d_1(n) - y(n), \tag{3}$$

3. Update the coefficients of the adaptive demodulator:

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + \mu e(n)\mathbf{r}(n). \tag{4}$$

   where μ is a constant parameter called step-size, which controlls the steady-state error and the convergence speed.

For the training based detector the convergence speed is governed by the eigenvalue spread of the input autocorrelation matrix defined as:

$$\mathbf{R} = E\left[\mathbf{r}(n)\mathbf{r}^T(n)\right] \quad = E\left[\mathbf{SAd}(n)\mathbf{d}^T(n)\mathbf{A}^T\mathbf{S}^T\right] + E\left[\mathbf{v}(n)\mathbf{v}^T(n)\right], \tag{5}$$

where $E[\bullet]$ represents the expected value of the quantity inside the squared brackets and we have assumed that the elements of the vector $\mathbf{v}$ are random zero-mean and independent from $\mathbf{S}$, $\mathbf{A}$ and $\mathbf{d}(n)$.

It is clear from Eq. (5) that the eigenvalue spread of the input autocorrelation matrix $\mathbf{R}$ can be far from unity and an adaptive demodulator using the standard LMS algorithm will have a very slow convergence. Since in the case of training based detectors, during the adaptation period no data sequences can be transmitted this slow convergence will decrease also the transmission rate. Therefore, in practical applications, the convergence speed of the detector has to be increased while maintaining a small steady-state error. Besides a slow convergence, the standard LMS algorithm has also the disadvantage of a trade-off between speed and steady-state output error. Indeed from Eq. (4) we can see that in order to obtain a small steady-state error, one has to chose a small step-size, but a small value of μ decreases the speed of convergence of the algorithm.

3 Variable Step-Size Least Mean Square Algorithm

In order to improve the performances of the LMS algorithm, the class of VSSLMS algorithms was introduced. In this paper, we analyze and compare the behavior of three algorithms belonging to this class. The comparison is done in terms of convergence speed, computational complexity and memory load.

The first compared algorithm is the Variable Step-Size LMS algorithm introduced in [13]. This algorithm uses a time-variable step-size in Eq. (4) which is adjusted as follows:

$$\mu'(n+1) = \alpha\mu(n) + \gamma e^2(n), \tag{6}$$

where α and γ are two positive constant parameters, $e(n)$ is the output error at time instant n and

$$\mu(n+1) = \begin{cases} \mu_{max} & \text{if} \quad \mu'(n+1) > \mu_{max} \\ \mu_{min} & \text{if} \quad \mu'(n+1) < \mu_{min} \\ \mu'(n+1) & \text{otherwise} \end{cases} \tag{7}$$

μ_{min} and μ_{max} being the minimum and respectively the maximum values allowed for the step-size.

In [13] the transient and steady-state analysis of the VSSLMS is given and the theoretical missadjustment is derived for both stationary and non-stationary cases. However, from the analysis presented in [13] the value of the missadjustment and the convergence speed depend on both coefficients α and γ. Therefore, we can conclude that the VSSLMS increases the convergence speed but still has the drawback between a fast convergence and a small steady-state error.

Another adaptive algorithm with time-varying step-size was introduced in [14] in order to improve the performances of the VSSLMS algorithm from [13] at the steady-state. The step-size update of the Robust Variable Step-Size LMS algorithm of [14] is described by the following equations:

$$\mu'(n+1) = \alpha\mu(n) + \gamma p^2(n), \tag{8}$$

$$\mu(n+1) = \begin{cases} \mu_{max} & \text{if} \quad \mu'(n+1) > \mu_{max} \\ \mu_{min} & \text{if} \quad \mu'(n+1) < \mu_{min} \\ \mu'(n+1) & \text{otherwise} \end{cases} \tag{9}$$

where

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1). \tag{10}$$

Using an approximation of the error autocorrelation $p(n)$ in the step-size update, the influence of the measurement noise is reduced and the algorithm performs better at the steady-state. However, also in the case of this algorithm the steady-state missadjustment depend on all three parameters (α, β and γ), so the dependence between the convergence speed and the steady-state error still exist.

In order to eliminate the dependence between the convergence speed and the steady-state error and in the same time to speed up the convergence, a new algorithm was introduced by the authors [15]. The new algorithm called Complementary Pair Variable Step-Size LMS uses two adaptive filters that works in parallel as shown in Fig. 2. In this figure $\mathbf{r}(n)$ is the $N \times 1$ input vector that is common for both adaptive filters, $\hat{\mathbf{h}}_1(n)$ and $\hat{\mathbf{h}}_2(n)$ are two $N \times 1$ vectors containing the coefficients of the accuracy and speed mode filter respectively, $e_1(n)$, $e_2(n)$, $y_1(n)$ and $y_2(n)$ are the output errors and the output sequences of the adaptive filters and $d_1(n)$ is the known transmitted sequence of the first user (we assume that the user 1 is the user of interest).

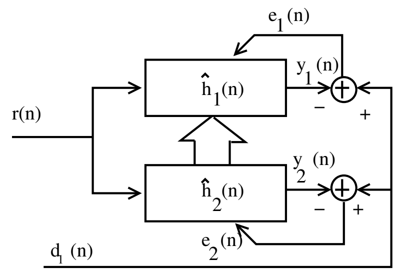

Fig. 2. Block diagram of the CPVSLMS.

The speed mode filter $\hat{\mathbf{h}}_2(n)$ uses an adaptive LMS algorithm with fixed and large step-size $\mu_2 = \mu_{max}$, while the filter $\hat{\mathbf{h}}_1(n)$ uses a variable step-size $\mu_1(n)$. The speed mode filter is used in our algorithm just to speed-up the convergence. The accuracy mode filter $\hat{\mathbf{h}}_1(n)$ is the filter of interest and from its output sequence the transmitted data is reconstructed. The CPVSLMS algorithm is described by the following steps:

1. Compute the output sequences and the output errors:

$$\begin{aligned} y_1(n) &= \hat{\mathbf{h}}_1(n)\mathbf{r}(n), \\ y_2(n) &= \hat{\mathbf{h}}_2(n)\mathbf{r}(n), \\ e_1(n) &= d_1(n) - y_1(n), \\ e_2(n) &= d_1(n) - y_2(n) \end{aligned} \tag{11}$$

2. Update the coefficients of the speed mode filter:

$$\hat{\mathbf{h}}_2(n+1) = \hat{\mathbf{h}}_2(n) + \mu_{max} e_2(n)\mathbf{r}(n), \tag{12}$$

3. Update the coefficients of the accuracy mode filter:

$$\hat{\mathbf{h}}_1(n+1) = \begin{cases} \hat{\mathbf{h}}_2(n+1), & \text{if } \begin{cases} \sum\limits_{i=n-T+1}^{n} e_2^2(i) < \sum\limits_{i=n-T+1}^{n} e_1^2(i) \\ \text{and } n = T, 2T, \ldots \end{cases} \\ \hat{\mathbf{h}}_1(n) + \mu_1(n) e_1(n)\mathbf{X}(n), & \text{otherwise} \end{cases} \tag{13}$$

4. Update the step-size of the accuracy mode filter:

$$\mu_1(n+1) = \begin{cases} \dfrac{\mu_1(n)+\mu_{max}}{2}, & \text{if } \begin{cases} \sum\limits_{i=n-T+1}^{n} e_2^2(i) < \sum\limits_{i=n-T+1}^{n} e_1^2(i) \\ \text{and } n = T, 2T, \ldots \end{cases} \\ \max\left\{\alpha\mu_1(n), \mu_{min}\right\}, & \text{if } \begin{cases} \sum\limits_{i=n-T+1}^{n} e_2^2(i) \geq \sum\limits_{i=n-T+1}^{n} e_1^2(i) \\ \text{and } n = T, 2T, \ldots \end{cases} \\ \mu_1(n), & \text{otherwise} \end{cases} \tag{14}$$

where T and $\alpha \in [0,1]$ are two constant parameters.

The behavior of the proposed CPVSLMS algorithm can be described as follows: during the training period the speed mode filter performs similar as an adaptive filter with fixed and large step-size. The accuracy mode filter performs also as an adaptive filter with fixed step-size for a number of T consecutive iterations (test interval of length T). At the end of the test interval, the sum of squared errors of the speed mode filter and accuracy mode filter are computed. If the sum of the squared error of the accuracy mode filter is larger than the sum of squared error of the speed mode filter, it means that the speed mode filter performed better that the accuracy mode filter during the last T iterations and it is closer to the Wiener solution. So, in this case the coefficients of the accuracy mode filter are updated with the coefficients of the speed mode filter (because they are closer to the Wiener solution) and also the step-size is increased (since a larger step-size had a better behavior). This situation appears at early stages of the adaptation when both adaptive filters are far from the optimum. When the speed mode filter is near its steady-state its sum of squared error will be larger that the sum of squared error of the accuracy mode filter (larger step-size means larger steady-state MSE) and in this case the step-size of the accuracy mode filter is decreased in order to obtain the desired steady-state missadjustment. The value of the step-size $\mu_1(n)$ at the steady-state will be equal (or very close) to μ_{min}.

This operation of the CPVSLMS increases the convergence speed by increasing the step-size $\mu_1(n)$ when the adaptive filters are far from the optimum. Also the trade-off between the steady-state error and convergence speed is eliminated since the missadjustment of the accuracy mode filter will be given only by the value of μ_{min}. In the case of CPVSLMS algorithm there are some fixed parameters that has to be chosen by the user. The first parameter that controls the adaptation of the step-size and the convergence speed is α. For a small value of α, the step-size $\mu_1(n)$ is decreased too fast and the convergence of the algorithm is decreased. Therefore, we have used $\alpha = 0.9$ that gives good results in all our experiments. The length of the test interval T also controls the convergence speed of the algorithm. If T is chosen to be too large then the adaptation of the step-size of the accuracy mode filter is lost. For very large values of T the speed mode filter might converge inside the test interval and the step-size $\mu_1(n)$ is not enough increased. In this case, the speed of convergence will be very low. If T is too small, the step-size $\mu_1(n)$ will have large variations at the steady-state (even at the steady-state for a very small number of consecutive iterations the speed mode filter might have smaller error). In all our experiments we have used $T = 50$ and we have obtained good results. The parameter μ_{max}, must be chosen close to the stability limit in order to have a fast convergence. The steady-state value of $\mu_1(n)$ will be equal with μ_{min} (or very close to it) so μ_{min} will control the level of the steady-state missadjustment.

Table 1. Computational complexity and memory load for the compared algorithms.

Algorithm	Mult. and Div.	Add. and Sub.	Memory load
LMS	2N+1	2N	2N+4
CPVSLMS	4N+5	4N+5	3N+10
VSSLMS	2N+4	2N+1	2N+6
RVSSLMS	2N+9	2N+3	2N+9

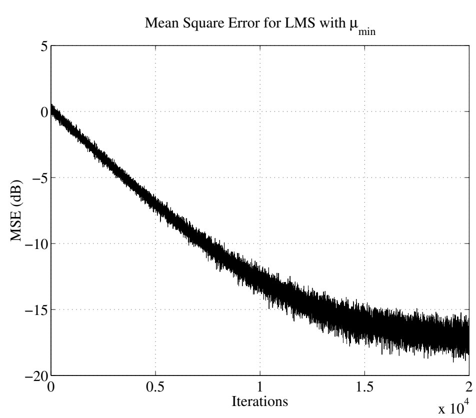

Fig. 3. Output Mean Square Error for the LMS with $\mu_{min} = 3 \times 10^{-4}$.

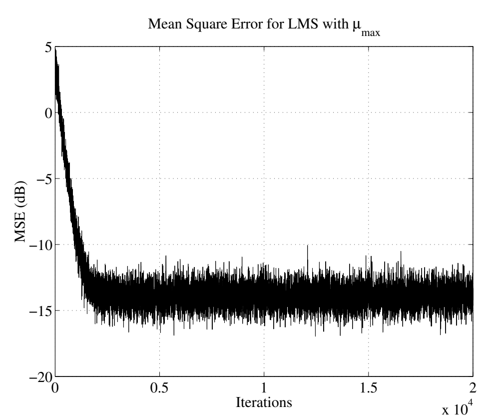

Fig. 4. Output Mean Square Error for the LMS with $\mu_{max} = 3 \times 10^{-3}$.

In Table 1 the computational complexity and memory load (the number of memory locations needed to store the variables and the parameters) of the compared algorithms are given. From this table we can see that the computational complexity and memory load of the proposed CPVSLMS algorithm are almost double compared with the other algorithms. However the benefit of the proposed algorithm the increased convergence speed and the fact that the dependence between the speed of convergence and the steady-state error is eliminated. Indeed the steady-state missadjustment of the CPVSLMS is given by its steady-state step-size which is $\mu_1(\infty) = \mu_{min}$ whereas the speed of convergence can be tuned by chosen the other parameters such as α, μ_{max} and T. In the case of VSSLMS and RVSSLMS the equations that gives the values of the parameters, provided in [13] and [14] are sometimes difficult to be used.

Table 2. Mean Square Error and the parameters for the compared algorithms.

Algorithm	Parameters	Steady-State MSE (dB)
LMS	$\mu_{min} = 3\times10^{-4}$	-17.4256
LMS	$\mu_{max} = 3\times10^{-3}$	-14.0252
CPVSLMS	$\mu_{min} = 3\times10^{-4}$, $\mu_2 = 3\times10^{-3}$, $\alpha = 0.9$, $T = 50$	-17.4251
VSSLMS	$\mu_{min} = 3\times10^{-4}$, $\mu_{max} = 3\times10^{-3}$, $\alpha = 0.9$, $\gamma = 0.002$	-17.2241
RVSSLMS	$\mu_{min} = 3\times10^{-4}$, $\mu_{max} = 3\times10^{-3}$, $\alpha = 0.97$, $\beta = 0.99$ $\gamma = 2$	-17.2643

4 Simulations and Results

In this section, computer simulations showing the performances of the CPVSLMS algorithm are presented. The CPVSLMS is compared with the LMS with fixed step-size, VSSLMS of [13] and MVSS of [14] in CDMA multiuser detection framework.

The signal model is given in Eq. (1), the number of users was $K = 4$ with the first user being the user of interest. The attenuation of the first user was 10 dB below the attenuation of the other three users. The spreading codes were chosen from a set of Gold sequences of length $N = 31$, the channel noise $v(n)$ was white Gaussian with zero mean and variance $\sigma_v^2 = 10^{-2}$. The transmitted data (the elements of the vector $d(n)$ in (1)) were equiprobable bipolar sequences with values in $\{-1,+1\}$.

The parameters of all the tested algorithms are presented in Table 2 together with the corresponding values of the steady-state missadjustments. The learning curves (the value of the Mean Square Error during the adaptation) for all algorithms are presented in Fig. 3, Fig. 4, Fig. 5, Fig. 6 and Fig. 7. These results were obtained by averaging a number of 100 Monte-Carlo simulations of length 4×10^4 iterations. From these figures, we can see, that the CPVSLMS has faster convergence compared with the VSSLMS, RVSSLMS and the LMS having a small step-size while their steady-state MSE were comparable.

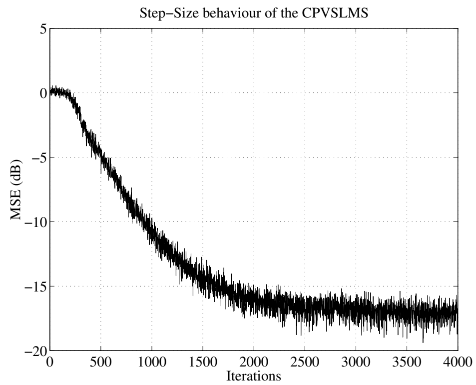

Fig. 5. Output Mean Square Error for CPVSSLMS.

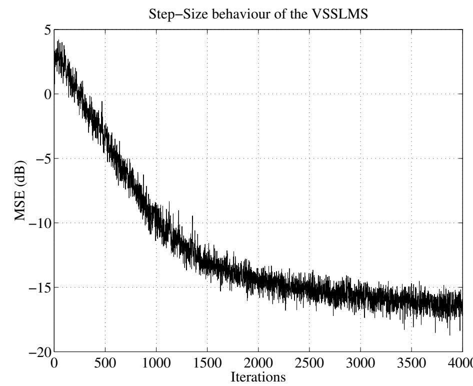

Fig. 6. Output Mean Square Error for VSSLMS.

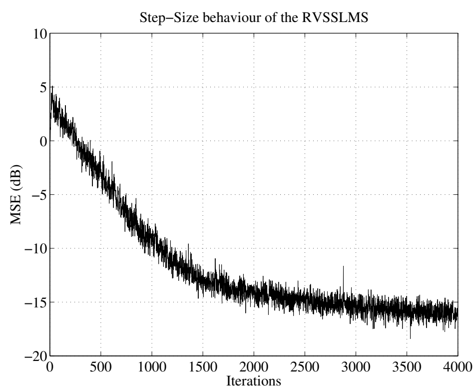

Fig. 7. Output Mean Square Error for RVSSLMS.

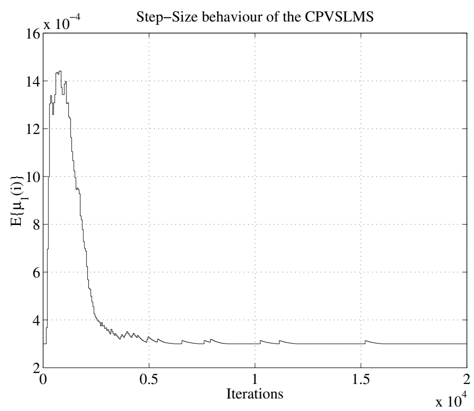

Fig. 8. Step-Size behavior for CPVSSLMS.

In order to have a more clear insight of the behavior of the compared algorithms in Fig. 8, Fig. 9 and Fig. 10 the expected value of the step-size during the adaptation for the CPVSLMS, VSSLMS and RVSSLMS respectively are plotted. From these figures, we can see, that the step-size of the CPVSLMS algorithm has the smallest variations at the steady-state and also its value is very close to $\mu_{min} = 3 \times 10^{-4}$. These results proofs our theoretical considerations that the steady-state missadjustment of the CPVSLMS is given by μ_{min}.

5 Conclusion

In this paper, we have compared the behavior of three different Variable Step-Size LMS adaptive algorithms for the problem of multiuser detection in DS/CDMA

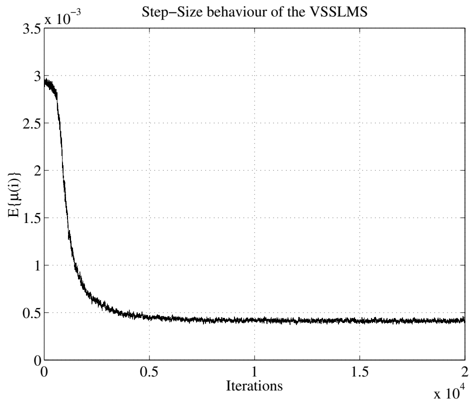

Fig. 9. Step-Size behavior for VSSLMS.

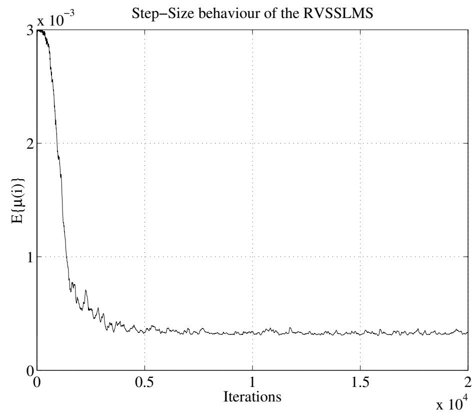

Fig. 10. Step-Size behavior for RVSSLMS.

systems. All the compared algorithms uses a time-variable step-size adapted by the output error to increase the adaptation speed. While two of the algorithms have smaller computational complexity and memory load they still suffer from the fact that their steady-state error and the speed of convergence depend on the same parameters. More that that, the equations used to compute the parameters of the VSSLMS and RVSSLMS are sometimes difficult to be used. The Complementary Pair Variable Step-Size LMS algorithm introduced by the authors in [15], although has an increased computational complexity and memory load, it has better speed performance and more simple parameters setup which are very important in practical applications.

References

[1] H. V. Poor and X. Wang, "Code-Aided Interference Suppression for DS/CDMA Communications-Part I: Interference Suppression Capability," *IEEE Trans. on Communications*, vol. 45, no. 9, pp. 1101–1111, Sept. 1997.

[2] ——, "Code-Aided Interference Suppression for DS/CDMA Communications-Part II: Parallel Blind Adaptive Implementations," *IEEE transactions on Communications*, vol. 45, no. 9, pp. 1112–1122, Sept. 1997.

[3] Y. Gong, T. J. Lim, and B. F. Boroujeny, "Performance Analysis of the LMS Blind Minimum-Output-Energy CDMA Detector," in *Proc. of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 5, Istanbul, Turkey, 2000, pp. 2473–2476.

[4] S. V. M. Honig, U. M, "Blind Adaptive Multiuser Detection," *IEEE Trans. on Information Theory*, vol. 41, no. 4, July 1995.

[5] K. W. Wong and T. O'Farell, "Blind Adaptive Single User Detection with Adjustable Step Size for DS/CDMA Communication over a Rayleyght Fading Mobile Channel," in *Proc. of IEE Third International Conference on 3G Mobile Communication Technologies*, May May 8-10, 2002, pp. 180–184.

[6] V. Krishnamurthy and G. Yin, "Adaptive Step-Size Algorithms for Blind Interference Suppression in DS/CDMA Systems," *Trans. on Signal Processing*, vol. 49, no. 1, Jan. 2001.

[7] Y. Gong, B. F. Boroujeny, and T. J. Lim, "Variable Step-Size LMS Blind CDMA Multiuser Detector," in *Proc. of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 4, Salt Lake City, 2001, pp. 2285–2288.

[8] M. Latva-aho and M. J. Juntti, "LMMSE Detection for DS-CDMA Systems in Fading Channels," *IEEE Trans. on Communications*, vol. 48, no. 2, pp. 194–199, Feb. 2000.

[9] T. J. Lim, Y. Gong, and B. Farhang-Boroujeny, "Convergence Analysis of LMS Multiuser CDMA Detectors," in *Proc. of Wireless Communications and Networking Conference*, New Orleans, USA, 1999, pp. 75–79.

[10] S. L. Miller, "Training Analysis of Adaptive Interference Suppression for Direct-Sequence Code-Division Multiple-Access Systems," *IEEE Trans. on Communications*, vol. 44, no. 4, pp. 488–495, Apr. 1996.

[11] S. Verdu, *Multiuser Detection*. New York: Cambridge University Press, 1998.

[12] S. Haykin, *Adaptive Filter Theory, 3-rd edition*. New Jersey: Prentice-Hall, 1996.

[13] R. H. Kwong and E. W. Johnston, "A Variable Step Size LMS Algorithm," *IEEE Trans. on Signal Processing*, vol. 40, no. 7, pp. 1633–1642, July 1992.

[14] T. Aboulnasr and K. Mayyas, "A Robust Variable Step-Size LMS-Type Algorithm: Analysis and Simulations," *IEEE Trans. on Signal Processing*, no. 3, pp. 631–639, Mar. 1997.

[15] R. C. Bilcu, P. Kuosmanen, and C. Rusu, "Improving Performances of Complementary Pair LMS Algorithm," in *Proc. of X European Signal Processing Conference, EUSIPCO 2000*, Tampere, Finland, Sept. 4–8, 2000.