Fast Convergence Least-Mean-Square Algorithms for MMSE Receivers in DS-CDMA Systems

Constantin Paleologu[1], *Călin Vlădeanu*[1], *and Safwan El Assad*[2]

[1]Department of Telecommunications, University Politehnica of Bucharest
Bucharest, Romania
{pale, calin}@comm.pub.ro
[2]IREENA, Ecole Polytechnique de l'Université de Nantes
Nantes, France
safwan.elassad@polytech.univ-nantes.fr

Abstract—**This paper considers a minimum mean-squared error (MMSE) single user adaptive receiver for the asynchronous direct-sequence code-division multiple-access (DS-CDMA) system, based on the least-mean-square (LMS) algorithm. It is known that in this context the adaptive algorithm can be iterated several times during the same bit interval in order to achieve a faster convergence rate, which further reduces the length of the training sequence. The objective of this paper is twofold. First, instead of using such multiple iterations, we propose a single equivalent formula for updating the receiver coefficients, saving significant time processing. Secondly, in order to further increase the convergence rate, a division-free version of the gradient adaptive lattice (GAL) algorithm is proposed. Since the lattice predictor orthogonalizes the input signals, this algorithm achieves a faster convergence rate than the transversal LMS algorithm.**

Keywords: Code-division multiple-access (CDMA) system; gradient adaptive lattice (GAL) algorithm; least-mean-square (LMS) algorithm; minimum mean-squared error (MMSE) receiver.

I. INTRODUCTION

There are a lot of mobile communications systems that employ the code-division multiple-access (CDMA) technique, where the users transmit simultaneously within the same bandwidth by means of different code sequences. This technique has been found to be attractive because of such characteristics as potential capacity increases over competing multiple access methods, anti-multipath capabilities, soft capacity, narrow-bandwidth anti-jamming, and soft handoff. In the Direct Sequence CDMA (DS-CDMA) system [1], each code sequence is used to spread the user data signal over a larger bandwidth and to encode the information into a random waveform. A simple multiplication between the data signal and the code sequence waveform is needed, and the resulted signal inherits its spectral characteristics from the spreading sequence. Due to its linear signal processing function this scheme may be a subject for possible performance improvements by developing new signal processing techniques for the receiver.

In DS-CDMA systems the conventional matched filter receiver distinguishes each user's signal by correlating the received multi-user signal with the corresponding signature waveform. The data symbol decision for each user is affected by multiple-access interference (MAI) from other users and by channel distortions. Hence, the conventional matched filter receiver performances are limited by its original purpose. It was designed to be optimum only for a single user channel where no MAI is present and to be optimum for a perfect power control, so it suffers from the near-far problem. Motivated by these limitations, adaptive minimum mean-squared error (MMSE) receivers have been introduced [2], [3]. The principle consists of a single user detector that works only with the bit sequence of that user. In this case, the detection process is done in a bit by bit manner, and the final decision is taken for a single bit interval from the received signal. The complexity of an adaptive MMSE receiver is slightly higher than that of a conventional receiver, but with superior performance. Besides its facile implementation the adaptive MMSE receiver has the advantage that it needs no supplementary information during the detection process.

The "brain" of an adaptive MMSE receiver is the adaptive algorithm. There are two major categories of such algorithms [4]. The first one contains the algorithms based on the mean square error minimization, whose representative member is the least-mean-square (LMS) algorithm. The second category of algorithms uses an optimization procedure in the least-squares (LS) sense, and its representative is the recursive-least-squares (RLS) algorithm. The LMS algorithm with its simple implementation suffers from slow convergence, which implies long training overhead with low system throughput. On the other hand, LS algorithms offer faster convergence rate, paying with increased computational complexity and numerical stability problems. Due to these reasons, LMS based algorithms are still preferred in practical implementations of adaptive MMSE receivers. Lattice structures have also been considered for this type of applications [5], [6], [7]. Since the lattice predictor orthogonalizes the input signals, the gradient adaptation algorithms using this structure are less dependent on the eigenvalues spread of the input signal and may converge faster than their transversal counterparts. The computational complexity of these algorithms is between transversal LMS and LS algorithms. In addition, several simulation examples and also numerical comparison of the analytical results have shown

This work was supported by the UEFISCSU Romania under Grant PN-II-PCE no. 331/01.10.2007.

that adaptive lattice filters have better numerical properties than their transversal counterparts [8], [9]. Moreover, stage-to-stage modularity of the lattice structure has benefits for efficient hardware implementations.

A solution for increasing the convergence rate of the MMSE receiver is to adjust the filter tap weights iteratively several times every transmitted bit interval [10]. Moreover, this procedure can be combined with a parallel interference cancellation (PIC) mechanism for further reducing the multiple access interference (MAI) [11]. A drawback of these approaches is that they are time consumers. During a bit interval, every single iteration has to "wait" the result from the previous one, which is the natural function mode for every iterative process. Anyway, from the time processing reason, it would be more convenient to use a single formula instead of those multiple iterations. A first objective of this paper is to propose a relation for updating the LMS adaptive filter coefficients, which is equivalent with the multiple iterations algorithm [12]. We will demonstrate that the multiple iterations process is equivalent with an unique iteration with a particular step size of the algorithm. Secondly, for further increasing the convergence rate, a lattice MMSE receiver based on a division-free gradient adaptive lattice (GAL) algorithm [13] is developed.

The paper is organized as follows. In Section II we briefly describe the asynchronous DS-CDMA system model. The analytical expression equivalent with the multiple iterations of the LMS algorithm is developed in Section III. Section IV is focused on the lattice receiver, revealing in this context the division-free GAL algorithm. The experimental results are presented in Section V. Finally, Section VI concludes this work.

II. DS-CDMA SYSTEM MODEL

In the transmitter part of the DS-CDMA system, each user data symbol is modulated using a unique signature waveform $a_i(t)$, with a normalized energy over a data bit interval T, $\int_0^T \|a_i(t)\|^2 \, dt = 1$, given by $a_i(t) = \sum_{j=1}^{N} a_i(j) p_c(t - jT_c)$ [1], with $i = 1, \ldots, K$, where K is the number of users in the system. Parameter $a_i(j)$ represents the jth chip of the ith user's code sequence and $p_c(t)$ is the chip pulse waveform defined over the interval $[0; T_c)$, with T_c as the chip duration (it is related to the bit duration through the processing gain N by $T_c=T/N$). In the following analysis we consider binary-phase shift keying (BPSK) transmission. The ith user transmitted signal is [1]

$$s_i(t) = \sqrt{2P_i} b_i(t) a_i(t) \cos(\omega_0 t + \theta_i)\,, \; i = \overline{1, \; K} \tag{1}$$

where $b_i(t) = \sum_{m=1}^{N_b} b_i(m) p(t - mT)$ is the binary data sequence for ith user (N_b is the number of received data bits), with $b_i(m) \in \{-1, \; +1\}$, P_i is the ith user bit power, ω_0 and θ_i represent the common carrier pulsation and phase, respectively.

After converting the received signal to its baseband form using a down converter, the received signal is given by [1]:

$$r(t) = \sqrt{\frac{P_i}{2}} \sum_{i=1}^{K} b_i(t - \tau_i) a_i(t - \tau_i) \cos(\theta_i) + n(t) \cos(\omega_0 t) \tag{2}$$

where $n(t)$ is the two-sided power spectral density $N_0/2$ additive white Gaussian noise. The asynchronous DS-CDMA system consists of random initial phases of the carrier $0 \le \theta_i < 2\pi$ and random propagation delays $0 \le \tau_i < T$ for all the users $i = \overline{1, \; K}$. There is no loss of generality to assume that $\theta_k = 0$ and $\tau_k = 0$ for the desired user k, and to consider only $0 \le \tau_i < T$ and $0 \le \theta_i < 2\pi$ for any $i \ne k$ [2]. Assuming perfect chip timing at the receiver, the received signal from (2) is passed through a chip-matched filter followed by sampling at the end of each chip interval to give for the mth data bit interval:

$$r_{m,\,l} = \int_{mT+lT_c}^{mT+(l+1)T_c} r(t) p(t - lT_c) dt, \quad l = 0, \; 1, \; \ldots, \; N-1 \tag{3}$$

where $p(t)$ is the chip pulse shape, which is taken to be a rectangular pulse with amplitude $1/\sqrt{N}$. Using (3) and taking the kth user as the desired one, the output of the chip matched filter after sampling for the mth data bit is given by:

$$\begin{aligned} r_{m,l} &= \sqrt{\frac{P_k}{2N}} T_c b_k(m) a_k\left(l\right) + \\ &+ \frac{1}{\sqrt{2N}} \sum_{\substack{i=1 \\ i \ne k}}^{K} \sqrt{P_i} \cos\theta_i b_i(m) I_{i,k}\left(m,l\right) + n(m,l) \end{aligned} \tag{4}$$

where

$$\begin{aligned} & I_{i,k}\left(m,l\right) = \\ & = \begin{cases} b_i\left(m-1\right)[\varepsilon_i a_i\left(N-1-N_i+l\right) + \\ \quad +\left(T_c - \varepsilon_i\right) a_i\left(N - N_i + l\right)], & 0 \le l \le N_i - 1 \\ b_i\left(m-1\right) \varepsilon_i a_i\left(N-1\right) + \\ \quad + b_i\left(m\right)\left(T_c - \varepsilon_i\right) a_i\left(0\right), & l = N_i \\ b_i\left(m\right)[\varepsilon_i a_i\left(l - N_i - 1\right) + \\ \quad +\left(T_c - \varepsilon_i\right) a_i\left(l - N_i\right)], & N_i + 1 \le l \le N-1 \end{cases} \end{aligned} \tag{5}$$

with $\tau_i = N_i T_c + \varepsilon_i$, $0 \le N_i \le N-1$, $0 < \varepsilon_i < T_c$.

A block diagram of the transversal MMSE receiver structure is depicted in Fig. 1. In the training mode, the receiver adapts its coefficients using a short training sequence employing an adaptive algorithm. After training is acquired, the receiver switches to the decision-directed mode and continues to adapt and track channel variations.

Let us consider the following vectors:

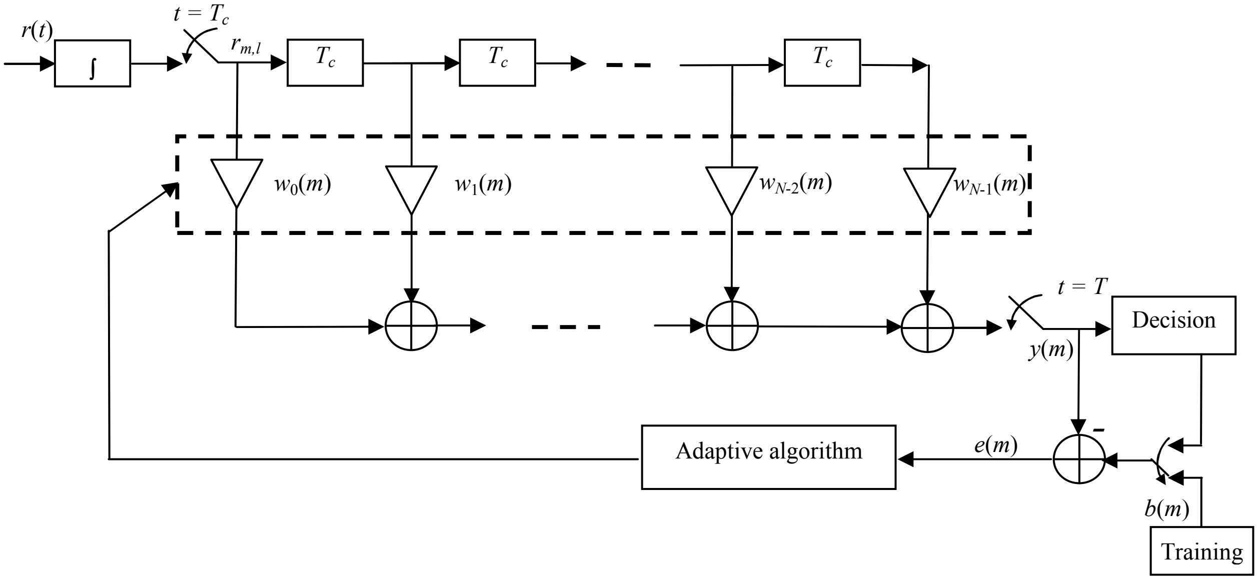

Figure 1. Transversal MMSE receiver scheme

$$\begin{aligned}\mathbf{w}(m) &= [w_0(m), w_1(m), ..., w_{N-1}(m)]^T \\ \mathbf{r}(m) &= [r_{m,0}, r_{m,1}, \ldots, r_{m,N-1}]^T\end{aligned} \quad , \qquad (6)$$

with $r_{m,l}$ given by (4). The output signal $y(m)$ will be an estimate of $b(m)$. For the estimation of $\mathbf{w}(m)$ a stochastic-gradient approach based on the LMS adaptive algorithm is used. The output signal is $y(m)=\mathbf{w}^T(m)\mathbf{r}(m)$. The receiver forms an error signal $e(m) = b(m) - y(m)$ and a new filter tap weight vector is estimated according to

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \mu\mathbf{r}(m)e(m) \, . \qquad (7)$$

The parameter μ represents the adaptation step size of the algorithm; its value is based on a compromise between fast convergence rate and low mean squared error [4].

A solution to increase the overall performances of the MMSE receiver is to adjust the filter tap weights iteratively several times every transmitted bit interval [10]. The coefficients obtained after the Gth iteration of the mth data bit is used by the algorithm in the first iteration of the $(m+1)$th data bit. It is obvious that this multiple iterations process will increase the computational complexity. This procedure can be combined with a PIC mechanism for further reducing the MAI [11]. This approach also makes use of the available knowledge of all users' training sequences at the base-station receiver to jointly cancel MAI and adapts to the MMSE optimum filter taps using the combined adaptive MMSE/PIC receiver.

III. Multiple Iterations LMS Algorithm

It can be noticed that the LMS adaptive algorithm used for MMSE receiver does not work in a sample-by-sample manner (chip-by-chip), as usually adaptive filters works, but in a block-by-block mode (bit-by-bit). Therefore, by using multiple iterations for every input data bit the converge rate of the adaptive algorithm increases. Nevertheless, this process is time consumer and it would be more convenient to use a one step formula instead of those multiple iterations per bit. Using the notations from the previous section, the multiple iterations LMS algorithm (without PIC mechanism) for the mth input data bit can be resume as follows:

```
for $g = 1, 2, \ldots, G$ ($G$ is the number of iterations per bit)
    $y^{(g)}(m) = \mathbf{w}^{(g)^T}(m)\mathbf{r}(m)$
    $e^{(g)}(m) = b(m) - y^{(g)}(m)$
    $\mathbf{w}^{(g+1)}(m) = \mathbf{w}^{(g)}(m) + \mu\mathbf{r}(m)e^{(g)}(m)$
end
```

As we have mentioned in the previous section, the first iteration for the $(m+1)$th data bit begins with the initial values of the filter coefficients given by $\mathbf{w}^{(1)}(m+1) = \mathbf{w}^{(G+1)}(m)$. In order to simplify the presentation, we renounce at the temporal index m and we will use the following notations: $\mathbf{r}(m) = \mathbf{r}$, $b(m) = b$, $\mathbf{w}^{(1)}(m) = \mathbf{w}_0$, $y_0 = \mathbf{w}_0^T\mathbf{r}$, $e^{(1)}(m) = b - y_0 = e_0$, $s = \mathbf{r}^T\mathbf{r}$. In the first three iteration we have:

$$\begin{aligned} g = 1: \quad & e^{(1)}(m) = b - y_0 = e_0 \\ & \mathbf{w}^{(2)} = \mathbf{w}_0 + \mu\mathbf{r}e_0 \end{aligned}$$

$$\begin{aligned} g = 2: \quad & e^{(2)} = b - \mathbf{w}^{(2)^T}\mathbf{r} = b - (\mathbf{w}_0 + \mu\mathbf{r}e_0)^T\mathbf{r} = e_0(1 - \mu s) \\ & \mathbf{w}^{(3)} = \mathbf{w}^{(2)} + \mu\mathbf{r}e^{(2)} = \mathbf{w}_0 + \mu\mathbf{r}e_0(2 - \mu s) \end{aligned}$$

$$g=3:\quad \begin{aligned} e^{(3)} &= b-\mathbf{w}^{(3)^T}\mathbf{r} = b-\left(\mathbf{w}_0+\mu\mathbf{r}e_0\left(2-\mu s\right)\right)^T\mathbf{r} = \\ &= e_0\left(1-2\mu s+\mu^2 s^2\right) \\ \mathbf{w}^{(4)} &= \mathbf{w}^{(3)}+\mu\mathbf{r}e^{(3)} = \mathbf{w}_0+\mu\mathbf{r}e_0\left(3-3\mu s+\mu^2 s^2\right) \end{aligned}$$

By performing simple polynomial operations we obtain in a similar manner:

$$\begin{aligned} \mathbf{w}^{(5)} &= \mathbf{w}_0+\mu\mathbf{r}e_0\left(4-6\mu s+4\mu^2 s^2-\mu^3 s^3\right) \\ \mathbf{w}^{(6)} &= \mathbf{w}_0+\mu\mathbf{r}e_0\left(5-10\mu s+10\mu^2 s^2-5\mu^3 s^3+\mu^4 s^4\right) \\ \mathbf{w}^{(7)} &= \mathbf{w}_0+\mu\mathbf{r}e_0\left(6-15\mu s+20\mu^2 s^2-15\mu^3 s^3+6\mu^4 s^4-\mu^5 s^5\right) \end{aligned}$$

Let us denote $-\mu s=\alpha$. It can be noticed that

$$\mathbf{w}^{(G+1)} = \mathbf{w}_0+\mu\mathbf{r}e_0 F\left(\alpha\right), \tag{8}$$

where

$$F\left(\alpha\right) = \sum_{k=0}^{G-1} f_k^{(G)}\alpha^k = \mathbf{f}^{(G)^T}\boldsymbol{\alpha}^{(G)}. \tag{9}$$

The column vectors from (9) are

$$\mathbf{f}^{(G)} = \left[f_0^{(G)}, f_1^{(G)}, \ldots, f_{G-1}^{(G)}\right]^T, \tag{10}$$

$$\boldsymbol{\alpha}^{(G)} = \left[1, \alpha, \alpha^2, \ldots, \alpha^{G-1}\right]^T. \tag{11}$$

The first element of the vector from (10) is $f_0^{(G)} = G$. Comparing (8) with (7) we can conclude that the LMS algorithm with multiple iterations per bit is equivalent with a LMS algorithm with a single iteration per bit but using a particular step size parameter given by

$$\mu^{(G)} = \mu F\left(\alpha\right). \tag{12}$$

To compute this parameter we need for the elements of the vector from (10). Let us consider the row vectors:

$$\underline{\mathbf{f}}^{(g)} = \left[\mathbf{f}^{(g)^T}, \mathbf{0}_{1\times(G-g)}\right], \qquad g=\overline{1,G}. \tag{13}$$

It can be noticed that $\underline{\mathbf{f}}^{(G)} = \mathbf{f}^{(G)^T}$. Using the vectors from (13) we can obtain the matrix

$$\mathbf{F}^{(G)} = \begin{bmatrix} \underline{\mathbf{f}}^{(1)} \\ \underline{\mathbf{f}}^{(2)} \\ \vdots \\ \underline{\mathbf{f}}^{(G)} \end{bmatrix}. \tag{14}$$

Each row of this matrix corresponds to a specific iteration. The elements of this matrix can be computed using

$$\mathbf{F}^{(G)}\left(i,j\right) = \mathbf{F}^{(G)}\left(i-1,j-1\right)+\mathbf{F}^{(G)}\left(i-1,j\right), \quad i,j=\overline{2,G}\ .(15)$$

It is obvious that $\mathbf{F}^{(G)}\left(j,1\right) = f_0^{(j)} = j$, with $j=\overline{1,G}$, and $\underline{\mathbf{f}}^{(1)} = \left[1, \mathbf{0}_{1\times(G-1)}\right]$. Once we have set the value of parameter *G*, this matrix can be computed a priori and we will use only the elements of its last row. For example, assuming that we choose *G* = 10, the matrix from (14) results

$$\mathbf{F}^{(10)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 10 & 10 & 5 & 1 & 0 & 0 & 0 & 0 & 0 \\ 6 & 15 & 20 & 15 & 6 & 1 & 0 & 0 & 0 & 0 \\ 7 & 21 & 35 & 35 & 21 & 7 & 1 & 0 & 0 & 0 \\ 8 & 28 & 56 & 70 & 56 & 28 & 8 & 1 & 0 & 0 \\ 9 & 36 & 84 & 126 & 126 & 84 & 36 & 9 & 1 & 0 \\ 10 & 45 & 120 & 210 & 252 & 210 & 120 & 45 & 10 & 1 \end{bmatrix}.$$

The last row of this matrix contains the elements of $\mathbf{f}^{(10)}$ which will be a priori computed in this manner. Of course, there is a need for the elements of the vector from (11), which have to be computed for each input data bit. After that, the parameter from (9) has to be computed, which requires *G* multiplication operations and (*G* – 1) addition operations. Finally, we update the filter coefficients according to (8). From the computational complexity point of view our approach requires almost the same number of operations as the multiple iterations per bit algorithm. Nevertheless, there is a significant time processing gain because our final result from (8) does not “wait” for the previous *G* iterations results, as in the multiple iterations process.

According to the theory of stability [4], the LMS adaptive algorithm is stable if the value of the step size parameter (which is a positive constant) is smaller than $\lambda_{\max}$, which is the largest eigenvalue of the correlation matrix of the input signal (i.e., the input data bit). In the case of the multiple iterations LMS algorithm, the stability condition has to take into account the number of iteration per bit. In our approach this is very facile because we use a single iteration as in (8), with the step size parameter given by (12). Consequently, the stability condition become

$$\mu^{(G)} < \frac{2}{\lambda_{\max}} . \tag{16}$$

Even if the multiple iterations algorithm is used, the previous condition can be used to determine the stability upper bound value of the step size parameter μ. According to (12) and (16)

$$\mu < \frac{2}{\lambda_{\max} F\left(\alpha\right)} . \tag{17}$$

An interesting situation appears if we use the Normalized LMS (NLMS) adaptive algorithm [4], which is another member of the stochastic gradient family. In this case, the algorithm step size parameter is computed as

$$\mu_{NLMS} = \frac{\overline{\mu}}{\mathbf{r}^T \mathbf{r}} = \frac{\overline{\mu}}{s} . \tag{18}$$

where $\overline{\mu}$ is a positive constant. In the case of the classical NLMS algorithm the previous constant has to be smaller than 2, in order to assure the stability of the algorithm [4]. Following the same procedure, it will result that

$$\mathbf{w}^{(G+1)} = \mathbf{w}_0 + \mu_{NLMS} \mathbf{r} e_0 F\left(\overline{\mu}\right) , \tag{19}$$

where

$$F\left(\overline{\mu}\right) = \sum_{k=0}^{G-1} f_k^{(G)} \left(-\overline{\mu}\right)^k = \mathbf{f}^{(G)^T} \boldsymbol{\beta}^{(G)} , \tag{20}$$

$$\boldsymbol{\beta}^{(G)} = \left[1, -\overline{\mu}, \left(-\overline{\mu}\right)^2, \ldots, \left(-\overline{\mu}\right)^{G-1} \right]^T . \tag{21}$$

Similarly, the NLMS algorithm with multiple iterations per bit is equivalent with a NLMS algorithm with a single iteration per bit but using a step size parameter given by

$$\mu_{NLMS}^{(G)} = \mu_{NLMS} F\left(\overline{\mu}\right) = \frac{\overline{\mu} F\left(\overline{\mu}\right)}{\mathbf{r}^T \mathbf{r}} \tag{22}$$

The following condition has to be satisfied for the stability:

$$0 < \overline{\mu} F\left(\overline{\mu}\right) < 2 \tag{23}$$

Because it is more facile to compute the elements of the vector from (21), as compared with the elements of the vector from (11), the NLMS algorithm could be an attractive alternative to the LMS algorithm. Moreover, it is a more robust algorithm because it overcomes in some sense the gradient noise amplification problem associated with the LMS algorithm [4]. Nevertheless, a division operation is required for computing the step size parameter from (22), which could be a difficulty and a source of numerical errors especially in a fixed-point implementation.

IV. GAL ALGORITHM FOR MMSE RECEIVER

Expected advantages of the adaptive lattice filters over the conventional LMS transversal filters include faster convergence rates with spectrally deficient inputs, automatic determination of the system's order, stage-to-stage modularity for efficient hardware implementations, and better data tracking abilities.

The (N – 1)-th-order multistage lattice predictor from Fig. 2 is specified by the recursive equations

$$\begin{aligned} e_p^f(l) &= e_{p-1}^f(l) + k_p^*(l) e_{p-1}^b(l-1) \\ e_p^b(l) &= e_{p-1}^b(l-1) + k_p(l) e_{p-1}^f(l) \end{aligned} \tag{24}$$

with p=1,…,N–1. We denoted by $e_p^f(l)$ the forward prediction error, by $e_p^b(l)$ the backward prediction error, and by $k_p(l)$ the reflection coefficient at the pth stage and chip-time l. The initial prediction errors are $e_0^f(l) = e_0^b(l) = r_{m,l}$.

The cost function used for the estimation of $k_p(l)$ is [4]

$$J_p(l) = \frac{1}{2} E\left\{ \left| e_p^f(l) \right|^2 + \left| e_p^b(l) \right|^2 \right\} \tag{25}$$

where E is the statistical expectation operator. Substituting (24) into (25), then differentiating the cost function $J_p(l)$ with respect to the complex-valued reflection coefficient $k_p(l)$ and imposing the gradient equal to zero, the optimum value of the reflection coefficient for which the cost function is minimum results

$$k_p^{opt} = -\frac{2E\left\{ e_{p-1}^b(l-1) e_{p-1}^{f*}(l) \right\}}{E\left\{ \left| e_{p-1}^f(l) \right|^2 + \left| e_{p-1}^b(l-1) \right|^2 \right\}} \tag{26}$$

Assuming that the input signal is ergodic, the expectations can be substituted by time averages, resulting the Burg estimate for the reflection coefficient k_p^{opt} for stage p:

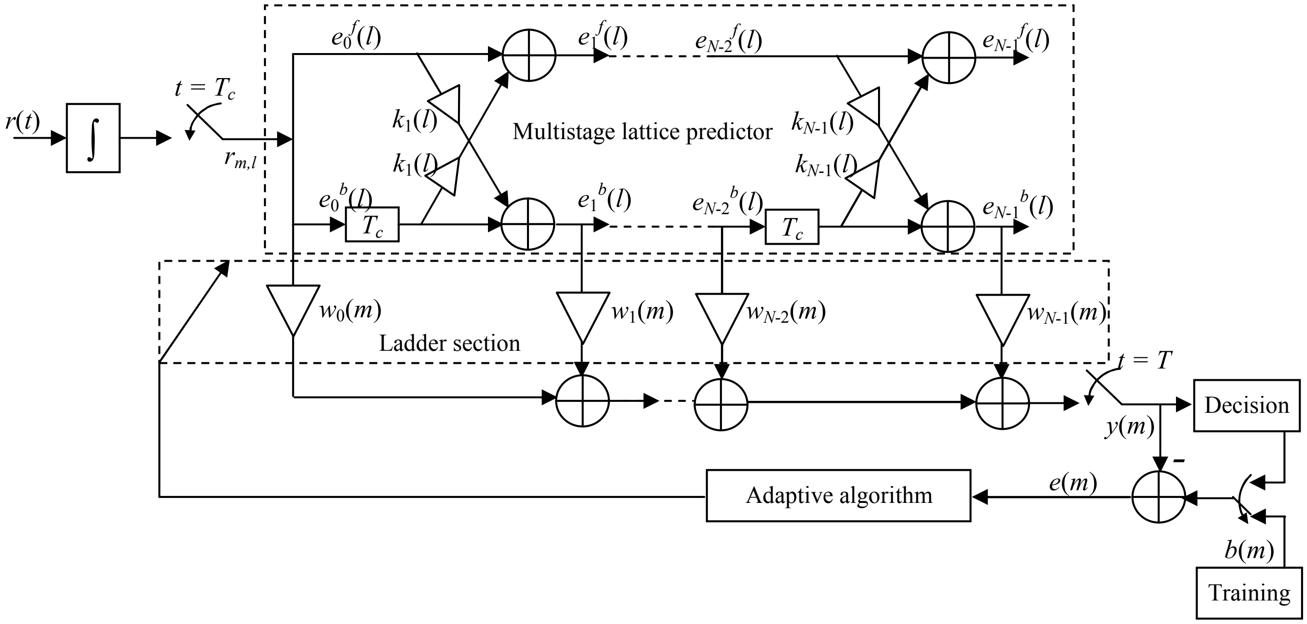

Figure 2. Lattice MMSE receiver scheme.

$$k_p(l) = -\frac{2\sum_{q=1}^{l} e_{p-1}^{b}(q-1)e_{p-1}^{f*}(q)}{\sum_{q=1}^{l}\left[\left|e_{p-1}^{f}(q)\right|^2+\left|e_{p-1}^{b}(q-1)\right|^2\right]} \tag{27}$$

Let us denoted by $W_{p-1}(l)$ the total energy of both the forward and backward prediction errors at the input of the pth lattice stage. It is expressed as:

$$\begin{aligned} W_{p-1}(l) &= \sum_{q=1}^{l}\left[\left|e_{p-1}^{f}(q)\right|^2+\left|e_{p-1}^{b}(q-1)\right|^2\right]= \\ &= W_{p-1}(l-1)+\left|e_{p-1}^{f}(l)\right|^2+\left|e_{p-1}^{b}(l-1)\right|^2 \end{aligned} \tag{28}$$

It can be demonstrated [4] that the GAL algorithm updates the reflection coefficients using

$$k_p(l+1) = k_p(l) - \frac{\eta}{W_{p-1}(l)}\cdot\left[e_p^{f*}(l)e_{p-1}^{b}(l-1)+e_p^{b*}(l)e_{p-1}^{f}(l)\right] \tag{29}$$

where the constant η controls the convergence of the algorithm. For a well-behaved convergence of the GAL algorithm, it is recommended to set $\eta < 0.1$. In practice, a minor modification is made to the energy estimator from (28) by writing it in the form of a single-pole average of squared data:

$$W_{p-1}(l) = \beta W_{p-1}(l-1) + (1-\beta)\cdot\left[\left|e_{p-1}^{f}(l)\right|^2+\left|e_{p-1}^{b}(l-1)\right|^2\right] \tag{30}$$

where $0 < \beta < 1$. The introduction of parameter β in (30) provides the GAL algorithm with a finite memory, which helps it to deal better with statistical variations when operating in a nonstationary environment. As it was reported in [8] and it was demonstrated in [14], the proper choice is $\beta = 1 - \eta$.

As we see in Fig. 2, the basic structure for the estimation of the user desired response $b(m)$, is based on a multistage lattice predictor that performs both forward and backward predictions, and an adaptive ladder section. We have an input column vector of the backward prediction errors

$$\mathbf{e}_N^b(m) = [e_0^b(m), e_1^b(m), \ldots, e_{N-1}^b(m)]^T \tag{31}$$

and a corresponding column vector $\mathbf{w}(m)$ containing the N coefficients of the ladder section of the adaptive filter. For the estimation of $\mathbf{w}(m)$, we may use a stochastic-gradient approach. The discrete output signal $y(m)$ is given by:

$$y(m) = \mathbf{w}^T(m)\mathbf{e}_N^b(m) \tag{32}$$

The receiver forms the error signal $e(m)$ and a new filter tap weight vector is estimated according to:

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \tilde{\mu} e(m)\mathbf{e}_N^b(m) \qquad (33)$$

The parameter $\tilde{\mu}$ is the ladder structure adaptation step size, chosen to optimize both the convergence rate and the misadjustment of the algorithm.

Summarizing, we will use (24), (30), and (29) for the lattice predictor part of the scheme, together with (32) and (33) for the ladder section. As compared to its transversal counterpart based on the LMS algorithm, the lattice MMSE receiver implies an increased computational complexity due to the multistage lattice predictor. Nevertheless, due to the fact that the lattice predictor orthogonalizes the input signals, a faster convergence rate is expected.

It can be noticed that in the lattice predictor part of the classical GAL algorithm a division operation per stage is used, which significantly grows the computational complexity in a fixed-point implementation context. Due to cost considerations, equipment manufactures generally prefer the use of fixed-point Digital Signal Processors (DSPs) over floating-point ones in their products. In fixed point DSPs, every division operation requires a number of iterations equal to the word length (i.e., the number of representation bits), while the multiplication and addition operations can be performed in a single iteration. In the following, we will propose an approximate version of the GAL algorithm that replaces the division operation using three multiplication operations and one addition operation instead. This is much more convenient from the computational complexity point of view in a fixed-point DSP implementation.

We start from equation (30) and we may write:

$$\begin{aligned}
&\frac{1}{W_{p-1}(l)} = \\
&= \frac{1}{\beta W_{p-1}(l-1) + (1-\beta)\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2} = \\
&= \frac{1}{\beta W_{p-1}(l-1)} \cdot \frac{1}{1 + \dfrac{1-\beta}{\beta} \cdot \dfrac{\left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2}{W_{p-1}(l-1)}}
\end{aligned} \qquad (34)$$

Let us denote

$$T_{p-1}(l-1) = \frac{1}{W_{p-1}(l-1)} \qquad (35)$$

and

$$c_{p-1}(l) = \left|e_{p-1}^f(l)\right|^2 + \left|e_{p-1}^b(l-1)\right|^2 \qquad (36)$$

Using the previous notations we may rewrite equation (34) as follows:

$$T_{p-1}(l) = \frac{1}{\beta} T_{p-1}(l-1) \cdot \frac{1}{1 + \dfrac{1-\beta}{\beta} c_{p-1}(n) T_{p-1}(l-1)} \qquad (37)$$

Supposing that the sum of the squared prediction errors at time l, i.e., $c_{p-1}(l)$, is much smaller than the sum of all squared prediction errors until that moment of time, $W_{p-1}(l-1)$, we may write that

$$\left|c_{p-1}(l) T_{p-1}(l-1)\right| << 1 \qquad (38)$$

Taking into account that

$$\frac{1}{1+x} = \sum_{n=0}^{\infty} (-1)^n x^n \qquad \text{for} \qquad |x| < 1 \qquad (39)$$

we may use the following approximate relation instead of equation (37):

$$T_{p-1}(l) \cong \frac{1}{\beta} T_{p-1}(l-1) \cdot \left(1 - \frac{1-\beta}{\beta} c_{p-1}(l) T_{p-1}(l-1)\right) \qquad (40)$$

In order to prevent any unwanted situations that can affect the supposition made in equation (38) (e.g., impulse perturbation of the input signal) we may compute the following minimum value:

$$t_{p-1}(l) = \min\left(\lambda, \frac{1-\beta}{\beta} c_{p-1}(l) T_{p-1}(l-1)\right) \qquad (41)$$

where $\lambda < 1$ is a positive constant, and than rewrite the equation (40) as follows:

$$T_{p-1}(l) \cong \frac{1}{\beta} T_{p-1}(l-1)(1 - t_{p-1}(l)) \qquad (42)$$

Finally, the reflection coefficients are updated using

$$\begin{aligned}
&k_p(l) = k_p(l-1) - \eta T_{p-1}(l) \cdot \\
&\cdot \left[e_p^{f*}(l) e_{p-1}^b(l-1) + e_p^{b*}(l) e_{p-1}^f(l) \right]
\end{aligned} \qquad (43)$$

The new step-size parameter of the division-free GAL algorithm is $\eta T_{p-1}(l)$ and it acts similar to $\eta / W_{p-1}(l)$ from the classical GAL algorithm. It can be noticed that the "unwanted" division operation from the classical GAL algorithm is replaced by three multiplication operations and one addition operation, which is more suitable in a fixed-point implementation context.

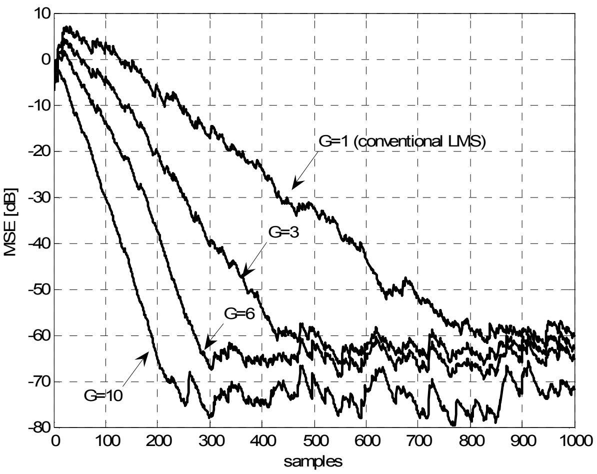

Figure 3. Convergence for the multiple iterations LMS algorithms.

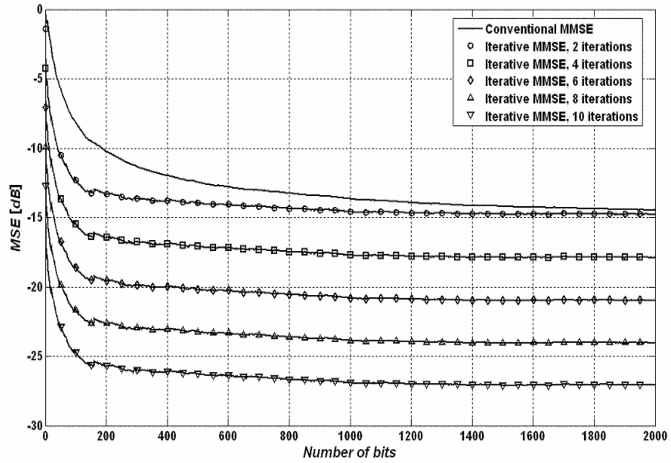

Figure 5. Convergence of the MMSE receiver based on the multiple iterations LMS algorithm. $K = 8$, $N = 16$, SNR = 15dB.

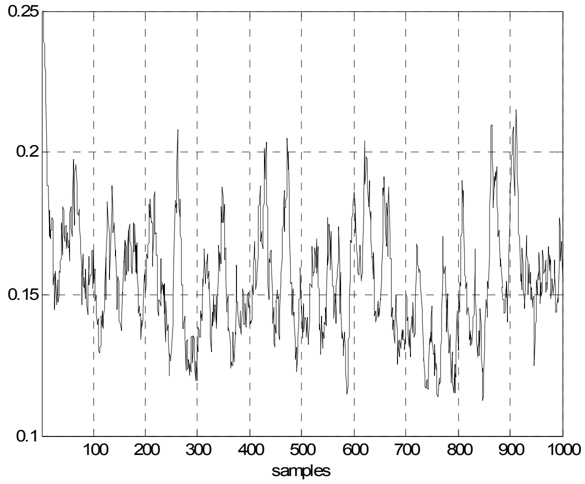

Figure 4. Evolution of the step size parameter $\mu^{(10)}$ given by (12).

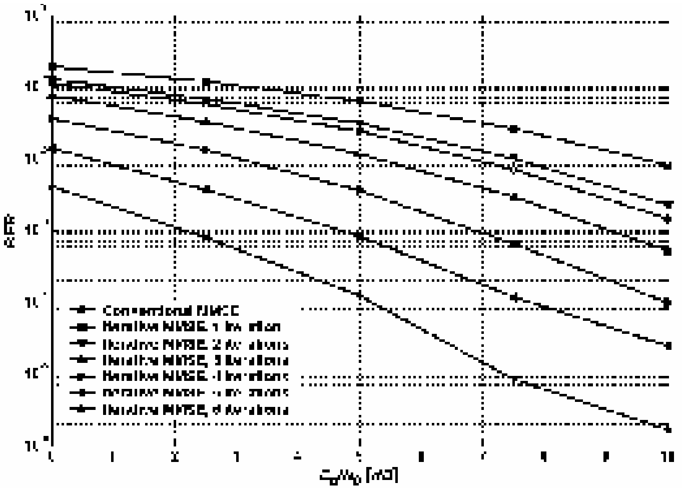

Figure 6. BER performance of the MMSE receiver based on the multiple iterations LMS algorithm. Other conditions as in Fig. 5.

V. Simulation Results

In order to analyze the convergence of the multiple-iteration LMS algorithm, a first set of experimental tests is performed in a simple "system identification" configuration. In this class of applications, an adaptive filter is used to provide a linear model that represents the best "fit" to an unknown system. The adaptive filter and the unknown system are driven by the same input. The unknown system output supplies the desired response for the adaptive filter. These two signals are used to compute the estimation error, in order to adjust the filter coefficients. The input signal is a random sequence with an uniform distribution on the interval (–1;1). In this first set of experiments the adaptation process is performed in a sample-by-sample manner. The mean square error (MSE) is estimated by averaging over 100 independent trials. In Fig. 3 are depicted the convergence curves for the multiple iterations LMS algorithm and the proposed equivalent algorithm, for different values of G. As expected, the curves for these two algorithms are perfectly matched, due to the mathematical equivalency between them. It can be notice that the convergence rate of the algorithms increases with the value of G and the MSE decreasing. In Fig. 4 we plot the evolution of the step size parameter of the proposed algorithm, given by (12), for G =10.

A second set of simulations employs the asynchronous DS-CDMA system, using the transversal MMSE iterative receiver. A BPSK transmission in a training mode scenario was considered. The binary spreading sequences are pseudo-random. The system simulation parameters are N = 16 and K = 8. The signal-to-noise ratio (SNR) is 15 dB. The LMS adaptive algorithm is iterated several times for each data bit using the proposed equivalent relation from (8). The adaptive process works now in a block-by-block manner. The MSE is also estimated by averaging over 100 independent trials. The results are presented in Fig. 5. It can be noticed that the MSE is decreased every new iteration.

Next, we investigated the steady-state BER (Bit Error Rate) performance as a function of the energy per bit to noise power

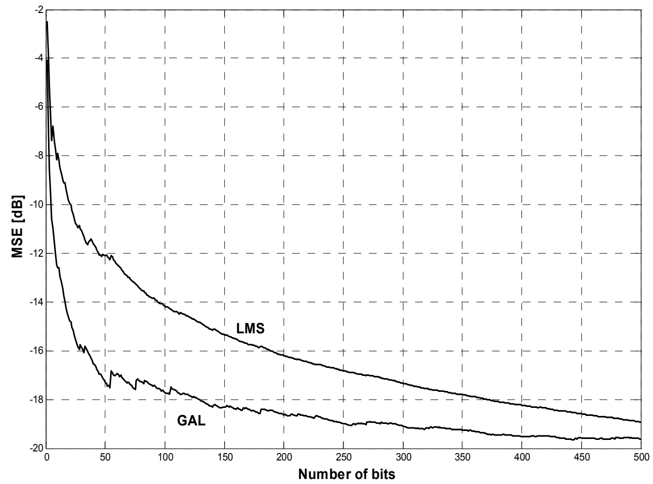

Figure 7. Convergence of the MMSE receivers. $K = 32$, $N = 64$, SNR = 15dB.

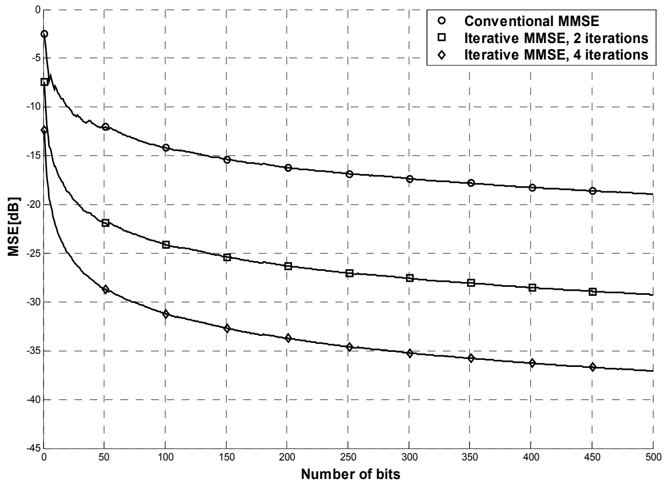

Figure 9. Convergence of the iterative LMS receiver. Other conditions as in Fig. 7.

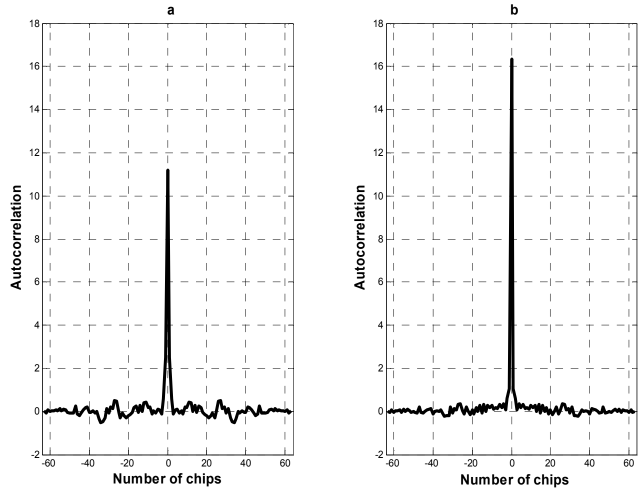

Figure 8. Autocorrelation functions for (a) $\mathbf{r}(m)$ - LMS input, (b) $\mathbf{e}^b_N(m)$ - GAL ladder section input. Other conditions as in Fig. 7.

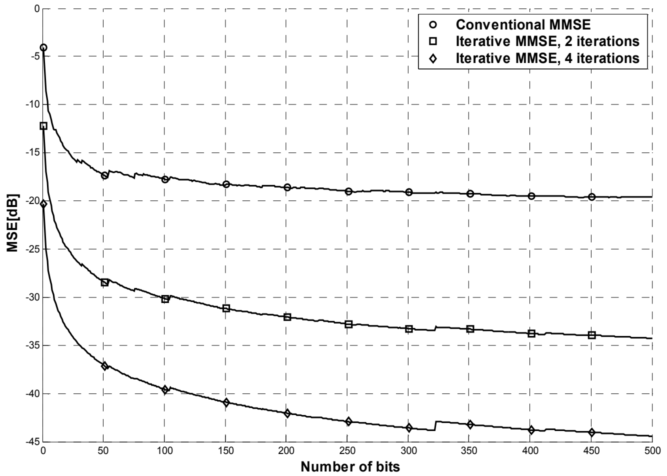

Figure 10. Convergence of the iterative GAL receiver. Other conditions as in Fig. 7.

spectral density ratio (E_b/N_0) of the iterative MMSE receiver considered above. These results are shown in Fig. 6, where we compared the performance of the iterative MMSE receiver using 1 to 6 additional iterations (i.e., G = 2 to 7) with the conventional adaptive LMS receiver, using one iteration per bit (i.e., G =1). The simulation results were obtained using 2000 bits training period for each value of E_b/N_0, in order to assure that the steady-state is reached. It is very important to note that under the same conditions the BER is improved every new iteration. Nevertheless, one should make a compromise between the computational complexity and the BER performances.

For further increasing the convergence rate, the lattice MMSE receiver based on the division-free GAL algorithm is tested as compared with its transversal counterpart based on the LMS algorithm. The simulation parameters were fixed as follows. The processing gain is $N = 64$, the number of users is $K = 32$, and SNR = 15 dB. The MSE is estimated by averaging over 100 independent trials. First, the conventional MMSE receivers are compared (Fig. 7). The superior convergence rate achieved by the GAL algorithm as compared to the transversal LMS algorithm is obvious. This faster convergence rate of the GAL algorithm over the transversal LMS algorithm can be explained by the fact that the lattice predictor orthogonalizes the input signals. Hence, the gradient adaptation algorithm using this structure is less dependent on the eigenvalues spread of the input signal. In order to support these remarks, in Fig. 8 the mean autocorrelation function is depicted for both inputs, i.e. $\mathbf{r}(m)$ (used as the direct input for the transversal LMS receiver) and the sequence of backward prediction errors $\mathbf{e}_N^b(m)$ (the input of the ladder section of the GAL receiver). It can be noticed that the input of the GAL ladder section has a higher variance as compared to the LMS input sequence. We should note that the faster convergence rate offered by the iterative MMSE receivers over the conventional receivers is mainly due to the interference rejection capability of the multiple iterations procedure within the adaptive algorithm. This improvement leads to a reduced level of MAI, which translates into a convergence rate close to the single user case.

Finally, a last set of simulations is performed in order to compare the multiple iterations MMSE receivers based on both LMS and GAL algorithms. Both algorithms are iterated for $G = 4$ within each data bit. The results are presented in Figs. 9 and 10. In both cases it can be noticed that MSE is decreased every new iteration. Also, the division-free GAL algorithm outperforms the LMS algorithm in terms of convergence rate.

VI. Conclusion and Perspectives

In this paper we first propose a relation for updating the LMS adaptive filter coefficients, which is equivalent with a multiple iterations LMS algorithm used in MMSE receivers for DS-CDMA communications systems. It was demonstrated that the multiple iterations process is equivalent with an unique iteration with a particular step size of the algorithm. Our proposed solution requires almost the same number of operations as the multiple iterations per bit algorithm but a significant time processing gain is achieved. The stability condition for the proposed algorithm was also derived. The simulation results proved the equivalency between the classical multiple iteration LMS algorithm and our proposed approach.

The MMSE iterative receiver considered in this paper was shown to improve the asynchronous DS-CDMA system performances. Thus, the MSE is decreased every new iteration by reducing MAI. This decrease offers a faster training mode for the receiver. A very important result is that the BER is considerably improved every new iteration.

The lattice MMSE receiver based on the division-free GAL algorithm improves the convergence rate as compared to the transversal LMS receiver. The lattice predictor orthogonalizes the input signals, so that the GAL algorithm is less dependent on the eigenvalues spread of the input signal and it converges faster than their transversal counterpart, the LMS algorithm. As a practical consequence, the lattice receiver will require a shorter training sequence as compared to the transversal one. Also, the GAL iterative receiver was shown to improve the asynchronous DS-CDMA system performances. The MSE decrease offers a faster training mode for the receiver. Hence, the designing procedure may consider two aspects, i.e., to shorten the training sequence for maintaining the same MAI in the system or to strongly reduce the MAI by keeping the same length of the training sequence. An analytical estimation of BER for the GAL receiver will be considered in perspective.

References

[1] S. Glisic and B. Vucetic, *CDMA Systems for Wireless Communications*, Artech House, 1997.

[2] S. Miller, “An adaptive direct-sequence code-division multiple-access receiver for multi-user interference rejection,” *IEEE Trans. Communications*, vol. 43, pp. 1746-1755, Apr. 1995.

[3] P. Rapajic and B. Vucetic, “Adaptive receiver structures for asynchronous CDMA systems,” *IEEE J. Select. Areas Communic.*, vol. 12, no. 4, pp. 685-697, May 1994.

[4] A. H. Sayed, *Adaptive Filters*, Wiley, 2008.

[5] F. Takawira, “Adaptive lattice filters for narrowband interference rejection in DS spread spectrum systems,” *Proc. IEEE South African Symp. Communications and Signal Processing*, 1994.

[6] J. Wang and V. Prahatheesan, “Adaptive lattice filters for CDMA overlay,” *IEEE Trans. Commun.*, vol. 48, no. 5, May 2000, pp. 820-828.

[7] C. Paleologu, C. Vlădeanu, and A.A. Enescu, “Lattice MMSE single user receiver for asyncronous DS-CDMA systems,” *Proc. of IEEE Int. Symp. ETc 2006*, pp. 97-102.

[8] V. J. Mathews and Z. Xie, “Fixed-point error analysis of stochastic gradient adaptive lattice filters,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 38, no. 1, January 1990, pp. 70-79.

[9] R. C. North, J. R. Zeidler, W. H. Ku, and T. R. Albert, “A floating-point arithmetic error analysis of direct and indirect coefficient updating techniques for adaptive lattice filters,” *IEEE Trans. on Signal Processing*, vol. 41, no. 5, May 1993, pp. 1809-1823.

[10] W. Hamouda and P. McLane, “Multiuser interference cancellation aided adaptation of a MMSE receiver for direct-sequence code-division multiple-access systems,” *Proc. IEEE Communications Theory Mini-Conf. (GLOBECOM)*, 2001.

[11] W. Hamouda and P. McLane, “A fast adaptive algorithm for MMSE receivers in DS-CDMA systems,” *IEEE Sign Proc. Letters*, vol. 11, no. 2, pp. 86-89, Feb. 2004.

[12] C. Paleologu and C. Vlădeanu, “On the behavior of LMS adaptive algorithm in MMSE receivers for DS-CDMA systems,” *Proc. of ICCGI 2007*, pp.12-17.

[13] C. Paleologu, S. Ciochină, A. A. Enescu, and C. Vlădeanu, “Gradient adaptive lattice algorithm suitable for fixed point implementation,” *Proc. of ICDT 2008*, pp. 41-46.

[14] C. Paleologu, S. Ciochină, and A.A. Enescu, “Modified GAL algorithm suitable for DSP implementation,” *Proc. of IEEE Int. Symp. ETc 2002*, vol. 1, pp. 2-7.