

RESEARCH

Open Access

Lattice implementation of adaptive channel shortening in multicarrier transmission over IIR channels

Emna Ben Salem^{1*}, Sofiane Cherif¹, Hichem Besbes¹ and Roberto López-Valcarce²

Abstract

Time-domain equalization is crucial in reducing channel dispersion and canceling interference in multicarrier systems. The equalizer is a finite impulse response (FIR) filter with the purpose that the delay spread of the combined channel-plus-equalizer IR is not longer than the cyclic prefix length. In this paper, a specific framework of long FIR channel-shortening problem is studied. In fact, approximated by a stable pole-zero model, we show that the channel transfer function poles introduce interference. Hence, to cancel bad poles, we propose the use of lattice structure to implement the channel shortener which places their zeros very close to critical channel poles and cancels them out. For low complexity implementation, we adopt adaptive algorithms to design the lattice channel shorteners. This paper analyzes the lattice structure performances of two blind adaptive channel shorteners: sum-squared autocorrelation minimization and multicarrier equalization by restoration of redundancy algorithms. The proposed implementation performances are given in terms of bit rate, and the simulation results are studied in the context of asymmetric digital subscriber line system.

1 Introduction

Multicarrier (MC) modulation has various advantages that make it useful for a wide variety of digital communication systems [1]. Actually, it has been chosen as the physical layer standard for a diversity of basic systems such as digital transmission over telephone lines, applications in broadcasting, and in wireless networks [2-6]. The most important advantage of the MC system is its robustness against interferences. In fact, cyclic prefix (CP) insertion through MC symbols provides higher immunity against channel delay spread. Therefore, as long as channel dispersion is not longer than the CP, system performance does not degrade. However, a highly time-dispersive channel leads to a significant reduction of the transmission data rate since the received signal is corrupted by both inter-carrier and inter-symbol interferences. To avoid such a performance degradation, a channel-shortening technique, commonly referred to as time-domain equalizer (TEQ), is introduced at the

receiver front end. Generally, the TEQ is defined as a transversal filter with the main purpose of keeping the delay spread of the combined channel-plus-equalizer impulse response (IR) not longer than the CP length [7]. Yet, by being finite IR (FIR) filters, all of the proposed TEQs are designed for both FIR and/or infinite IR (IIR) channel shortening which make all of them solve the channel-shortening problems without taking into account the transmission channel models.

Otherwise, we must stress the fact that the transmission characteristics of the channel determine directly the performance of communication systems. Therefore, several channel models were proposed to accurately simulate the effect of transmission of the MC signal through the channel. Particularly, it is proven that in twisted pair lines, the channel is well modeled by a recursive filter with a slowly decaying IR [8,9]. This means that the channel transfer function presents poles close to the unit circle (UC). By exploiting this result, we propose in this paper to develop a specific framework for slowly decaying recursive channel shortening. Indeed, approximated by a stable pole-zero model, we need to show that the channel poles introduce a term of interference in each received MC symbols. Thus, in order to shorten the recursive channel, an

*Correspondence: emna.bensalem@supcom.rnu.tn

¹ Cosim laboratory, High School of Communications of Tunis (SUPCOM), Ariana 2088, Tunisia

Full list of author information is available at the end of the article

effective TEQ will place their zeros on the critical poles to cancel them out. Yet, it is worth noting that inaccurate zeros position results in a limited channel-shortening performance, which motivates the use of lattice structure to cancel channel poles. Therefore, this can be effectively done only by adopting the lattice structure which must be conceived by optimizing appropriate channel-shortening criteria.

Furthermore, in the literature, most of the supervised channel-shortening criteria formulations, such as MSSNR [10], MGSNR [11], min-ISI [12], MBR [13] and SINR-max [14], are mainly based on the combined channel-plus-equalizer IR expression and their optimum TEQs which are perceived as solutions of algebra linear equations. However, when lattice structure is put into service, all of these criteria deal with non-linear formulation in terms of lattice TEQ reflection coefficients, whereas the criteria which are based on lattice filter output signal present linear formulation in terms of reflection coefficients. Thus, for linear algebra resolution and low complexity lattice implementation, we need to adopt the last kind of criteria, from which we quote the blind multicarrier equalization by restoration of redundancy (MERRY) and sum-squared autocorrelation minimization (SAM) algorithms [15,16].

Accordingly, we tend to present through this paper the lattice structure performances of the adaptive MERRY and SAM algorithms. Indeed, the proposed implementation performances are given in terms of bit rate, and the simulation results are studied in the context of digital subscriber line (DSL) systems. Initially, Section 2 discusses the problem statement of long FIR channel shortening and motivates the use of the lattice structure as a way to implement the TEQ. Section 4 then develops the steepest descent implementation of the lattice-based SAM algorithm, whereas Section 5 gives the recursive least squares (RLS) implementation of the lattice-based MERRY algorithm. Further, Section 6 presents comparative simulation results, and conclusions are given in Section 7.

The matrices and vectors are denoted with upper and lower case boldface letters (e.g., \mathbf{M} and \mathbf{v}), the superscripts $*$, T , H , and -1 denote the conjugate, the transpose, the Hermitian (conjugate transpose), and the inverse of a

matrix, respectively; $\mathbf{0}_{N \times L}$ and \mathbf{I}_N denote the $N \times L$ null matrix and the $N \times N$ identity matrix. For any vector \mathbf{v} , $\|\mathbf{v}\|$ denotes the Euclidean norm, and $\mathbf{M} = \text{diag}(\mathbf{v})$ is the diagonal matrix with diagonal elements equal to the vector \mathbf{v} elements. The notation $s_{k,n}$ is used to present the n th MC sample transmitted or received at k th MC symbol period.

2 Channel interference analysis for MC transmission

Actually, the ultimate goal of this section is to show that if the MC transmission channel is modeled by slowly decaying recursive filter, then its partial equalization can be done only through canceling its transfer function poles, causing interference. Thus, to achieve this purpose, the filter equalizer must place the zeros in critical poles as a way to cancel them out.

2.1 MC system model

MC modulation divides the transmission bandwidth into N parallel subchannels by means of inverse fast Fourier transform (IFFT). A CP is appended to each symbol to ensure subchannel orthogonality after propagation through the time-dispersive channel. Demodulation of the received signal is performed by an FFT operation. The simplified baseband equivalent MC system model is shown in Figure 1. At the transmitter and after modulation, the data sequence is converted into N parallel subsequences, $z_{k,n}$, where $n \in \{1, 2, \dots, N\}$ refers to the subcarrier number and k is the discrete time index. The block $\mathbf{z}_k \doteq [z_{k,1}, \dots, z_{k,n}, \dots, z_{k,N}]^T$ is used to modulate the different subcarriers by means of an IFFT, the result block vector is denoted by $\mathbf{x}_k \doteq [x_{k,1}, \dots, x_{k,n}, \dots, x_{k,N}]^T$ and is expressed by $\mathbf{x}_k = \mathbf{F}\mathbf{z}_k$, where \mathbf{F} represents the unitary symmetric IFFT matrix, $\{\mathbf{F}\}_{n_1, n_2} \doteq \frac{1}{\sqrt{N}} e^{j2\pi \frac{n_1 n_2}{N}}$, $n_1, n_2 \in \{0, 1, \dots, N-1\}$, and $j^2 = -1$. To ensure that the subcarriers remain orthogonal after propagation through the channel, the last P samples (corresponding to the CP) of \mathbf{x}_k are copied and added to the beginning of this block to form the k th MC block, $\tilde{\mathbf{x}}_k$, of length $M = N + P$, where $\tilde{\mathbf{x}}_k \doteq [\tilde{x}_{kM+1}, \dots, \tilde{x}_{(k+1)M}]^T = [x_{k, N-P+1}, \dots, x_{k, N}, x_{k, 1}, \dots, x_{k, N}]^T$.

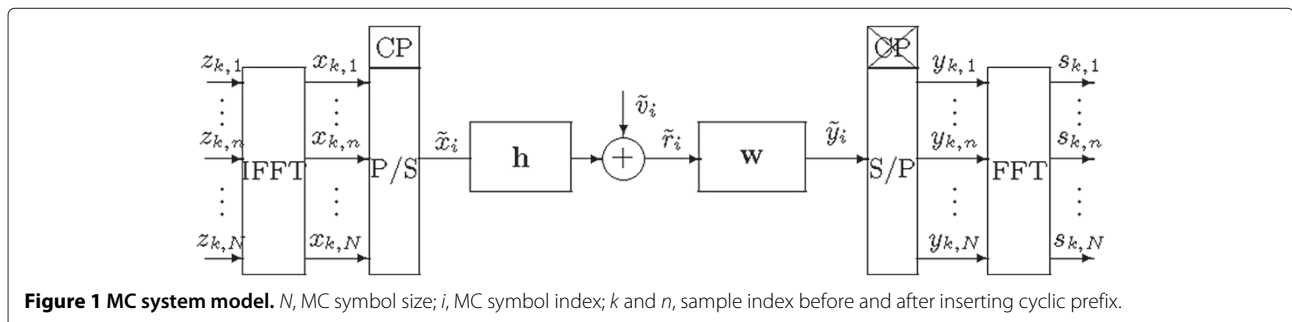


Figure 1 MC system model. N , MC symbol size; i , MC symbol index; k and n , sample index before and after inserting cyclic prefix.

Let \tilde{x}_i , where $i \doteq kM + m$ and $m \in \{1, \dots, M\}$, be the source sequence to be transmitted, after P/S operation. The process \tilde{x}_i is modeled as zero mean, wide sense stationary with unit variance, and white (this property holds provided that sampling is done at twice the signal bandwidth). The noise \tilde{v}_i is a zero mean white Gaussian process with variance σ_v^2 uncorrelated with the transmitted signal.

2.2 Channel pole-zero model

In order to analyze the channel-shortening problem, we propose to approximate the transmission channel with a lower order pole-zero model. In fact, when the channel is characterized by its long tail, it can be approximated by a reduced-parameter model. Actually, the problem of approximating a long FIR filter by a pole-zero filter with a small total number of coefficients had been investigated by many researchers [17,18], some of them had especially developed a stable pole-zero model of the DSL loop as a way to reduce the implementation complexity of the DSL channel equalization [9]. Thus, the transfer function of long FIR channel can be as follows:

$$H(z) = \frac{\sum_{q=0}^{L_a} a_q z^{-q}}{1 + \sum_{p=1}^{L_b} b_p z^{-p}}, \quad (1)$$

where L_a and L_b are the channel numerator and denominator orders, respectively. Moreover, the direct form channel model can be characterized in the time domain by the following difference equation:

$$\tilde{r}_i = \sum_{q=0}^{L_a} a_q \tilde{x}_{i-q} - \sum_{p=1}^{L_b} b_p \tilde{r}_{i-p} + \tilde{v}_i, \quad (2)$$

with \tilde{r}_i as the received signal.

2.3 Channel interference analysis

In the following, we detail the necessity of channel shortening in the case of MC long FIR channels. Hence, we will show that the presence of at least one pole in the channel transfer function introduces a term of interference in the received MC block. In order to simplify this analysis, we need to state the following explanation by assuming that the channel had one pole, i.e., $L_b = 1$, the numerator order is less than the CP, i.e., $L_a \leq P$, the equalizer filter is transparent, i.e., $\tilde{y}_i = \tilde{r}_i$, and no channel delay is considered, i.e., $\delta = 0$.

At the receiver, the channel output, \tilde{y}_i , is converted into M parallel substreams and the cyclic prefix is removed from the received block in order to obtain the k th MC symbol vector $\mathbf{y}_k = [\tilde{y}_{kM+P+1}, \tilde{y}_{kM+P+2}, \dots, \tilde{y}_{(k+1)M}]^T$.

According to the relation (2), we can state the matrix-vector form of the received MC block as follows:

$$\mathbf{y}_k = \mathbf{A}\mathbf{x}_k - \mathbf{B}\mathbf{y}_k^{(1)} + \tilde{\mathbf{v}}_k, \quad (3)$$

where \mathbf{A} is a $N \times N$ circulant matrix, where the first column and row are, respectively, disposed by $[a_0, \dots, a_{L_a}, 0, \dots, 0]^T$ and $[a_0, 0, \dots, 0, a_{L_a}, \dots, a_1]$, $\tilde{\mathbf{v}}_k \doteq [\tilde{v}_{kM+P+1}, \dots, \tilde{v}_{(k+1)M}]^T$ is the noise vector, $\mathbf{y}_k^{(1)} \doteq [\tilde{y}_{kM+P}, \dots, \tilde{y}_{(k+1)M-1}]^T$ and $\mathbf{B} \doteq b_1 \mathbf{I}_N$. In another aspect, for detailing the signal components received at k th symbol period, we denote by $\mathbf{y}_k^{(j)}$ the vector $[\tilde{y}_{kM+P+1-j}, \dots, \tilde{y}_{(k+1)M-j}]^T$ and by $\tilde{\mathbf{v}}_k^{(j)}$ the vector $[\tilde{v}_{kM+P+1-j}, \dots, \tilde{v}_{(k+1)M-j}]^T$. The last notation is omitted for $j = 0$, i.e., $\mathbf{y}_k^{(0)} = \mathbf{y}_k$ and $\tilde{\mathbf{v}}_k^{(0)} = \tilde{\mathbf{v}}_k$. As it can be shown, for $j \leq P - L_a$ and $l \leq P$, the vectors $\mathbf{y}_k^{(j)}$ and $\mathbf{x}_k^{(l)}$ are recursively computed according to the following relations:

$$\mathbf{y}_k^{(j)} = \mathbf{A}\mathbf{x}_k^{(j)} - \mathbf{B}\mathbf{y}_k^{(j+1)} + \tilde{\mathbf{v}}_k^{(j)}, \quad (4)$$

$$\mathbf{x}_k^{(l+1)} = \tilde{\mathbf{J}}\mathbf{x}_k^{(l)}. \quad (5)$$

Here, $\tilde{\mathbf{J}}$ is a shifting matrix, $\tilde{\mathbf{J}} \doteq [\tilde{\mathbf{J}}, \mathbf{e}]$ where $\tilde{\mathbf{J}} \doteq [\mathbf{0}_{(N-1) \times 1}, \mathbf{I}_{N-1}]^T$ and $\mathbf{e} \doteq [1, \mathbf{0}_{1 \times (N-1)}]^T$, and then, by exploiting the recursive relations (4) and (5), we can write for $L = P - L_a$:

$$\begin{aligned} \mathbf{y}_k &= \sum_{j=0}^L (-1)^j \mathbf{B}^j \tilde{\mathbf{A}}^j \mathbf{x}_k + (-1)^{L+1} \mathbf{B}^{L+1} \mathbf{y}_k^{L+1} \\ &\quad + \sum_{j=0}^L (-1)^j \mathbf{B}^j \tilde{\mathbf{v}}_k^{(j)}. \end{aligned} \quad (6)$$

Let $\mathbf{D} \doteq \sum_{j=0}^L (-1)^j \mathbf{B}^j \tilde{\mathbf{A}}^j$; we can easily show that \mathbf{D} is a circulant matrix. Further, an FFT operation demodulates the received MC symbol \mathbf{y}_k , and the resulting block which is denoted by \mathbf{s}_k is given by the following:

$$\begin{aligned} \mathbf{s}_k &= \underbrace{\mathbf{F}^H \mathbf{D} \mathbf{F} \mathbf{z}_k}_{\text{desired symbol}} + \underbrace{(-1)^{L+1} \mathbf{F}^H \mathbf{B}^{L+1} \mathbf{y}_k^{L+1}}_{\text{interference term}} \\ &\quad + \underbrace{\sum_{j=0}^L (-1)^j \mathbf{F}^H \mathbf{B}^j \tilde{\mathbf{v}}_k^{(j)}}_{\text{noise term}}. \end{aligned} \quad (7)$$

As a result, we notice that the received MC symbol consists of three major components: the desired signal component which is obtained through the diagonal matrix $\mathbf{F}^H \mathbf{D} \mathbf{F}$, the interference component which is expressed by the term $(-1)^{L+1} \mathbf{F}^H \mathbf{B}^{L+1} \mathbf{y}_k^{L+1}$, and the noise component $\sum_{j=0}^L (-1)^j \mathbf{F}^H \mathbf{B}^j \tilde{\mathbf{v}}_k^{(j)}$. Notably, the interference component is analyzed by expressing \mathbf{y}_k^{L+1} as follows:

$$\mathbf{y}_k^{L+1} = \tilde{\mathbf{A}} \mathbf{J}^{L+1} \mathbf{x}_k - \mathbf{B} \mathbf{y}_k^{L+2} + \tilde{\mathbf{v}}_k^{L+1}, \quad (8)$$

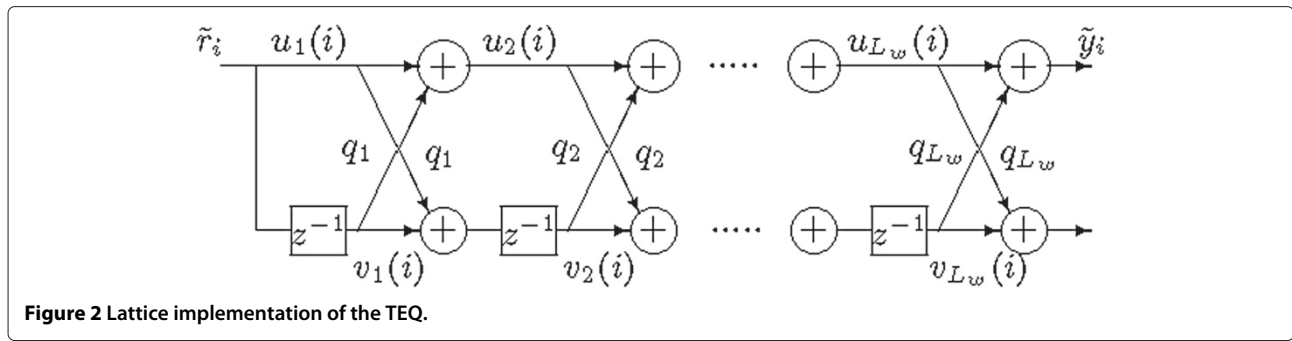


Figure 2 Lattice implementation of the TEQ.

where $\tilde{\mathbf{A}} \triangleq [\mathbf{a}, \mathbf{A}^T(:, 2 : N)]^T$, and $\mathbf{a} \triangleq [a_0, 0, \dots, 0, a_{L_a-1}, \dots, a_1]^T$. Therefore, we can observe that the matrix $\mathbf{B}^{L+1} \tilde{\mathbf{A}} \mathbf{J}^{L+1}$ cannot be circulant because $\tilde{\mathbf{A}}$ is non-circulant. Hence, it cannot be diagonalized by Fourier transform which makes this quotation produce an undesired interference signal in the received MC symbol \mathbf{s}_k .

Finally, we notice that, independent of the channel zero number, the channel poles introduce a term of interference with power depending on their locations inside the UC. Therefore, with the aim to cancel interference, it is proposed to insert a TEQ at the receiver front end to equalize poles and produce a combined channel-plus-equalizer IR not longer than the CP length. Next, we will discuss the lattice structure derived to implement the TEQ to shorten the recursive channel model.

3 Lattice structure of channel equalizer

There is no doubt that the equalizer coefficients depend mainly on the characteristics of the channel that are similarly determined from the measurements obtained by transmitting signals through the physical media. Such filters, with adjustable parameters, are usually called adaptive, especially when they are incorporated with algorithms which allow the filter coefficients to be adapted to the changes in the signal statistics. Adaptive technique can also lead to a reducing complexity.

Under their direct transversal shape, all of the proposed adaptive channel shorteners are designed for both FIR and/or IIR channel shortening. Therefore, in this paper, we propose a specific framework for channel shortening where we consider the channel model as a slowly decaying IIR filter. In order to ensure the best algorithm convergence to TEQ zeros, canceling poles and causing the tail, we suggest working with the lattice structure for the adaptive TEQ implementation. We admit the adaptive blind MERRY and SAM algorithms.

As it will be illustrated in the following sections, the choice of the TEQ filter structure has a profound effect on the operation of the channel-shortening algorithms. In fact, we will show that when the adaptive TEQs are implemented using the lattice structure, the TEQ coefficients converge very close to the optimal values.

Therefore, we need to distinguish the most popular filter structures: the transversal and lattice structures. Basically, the transversal structure is the most common structure used in implementing channel shorteners, i.e., the function of the latter is to adjust the set of $L_w + 1$ filter coefficients $w_j, j = 0, 1, \dots, L_w$ (tap weights) which makes the output \tilde{y}_i to be close as possible to a desired signal. The output filter is calculated as a linear combination of the input sequence.

The lattice TEQ filter is perceived as modular in structure so that it consists of a number of individual stages, each one of them has the appearance of a lattice, as it is stated in Figure 2. The transfer function of the lattice filter is determined by the reflection coefficients $q_p, p = 1, 2, \dots, L_w$. The stage outputs are obtained as follows:

$$u_1(i) = v_1(i + 1) = \tilde{r}_i, \quad (9)$$

$$v_p(i + 1) = q_{p-1}u_{p-1}(i) + v_{p-1}(i), \quad (10)$$

$$u_p(i) = q_{p-1}v_{p-1}(i) + u_{p-1}(i), \quad (11)$$

and the output TEQ is given by the following:

$$\begin{aligned} \tilde{y}_i &= \tilde{r}_i + \sum_{p=1}^{L_w} q_p v_p(i), \\ &= \tilde{r}_i + \mathbf{q}^T \mathbf{v}_i, \end{aligned} \quad (12)$$

where $\mathbf{q} = [q_1, q_2, \dots, q_{L_w}]^T$ and $\mathbf{v}_i = [v_1(i), v_2(i), \dots, v_{L_w}(i)]^T$.

The mapping from transversal to lattice form can be achieved via the Levinson-Durbin recursions [19]. Note that the lattice structure effectively imposes a monic constraint on the TEQ, i.e., the first tap is always 1. Hence, the equivalent filter length is equal to $L_w + 1$.

Hence, by adopting the lattice structure, we need to optimize the TEQ coefficients by implementing adaptive algorithms. We suggest working with the blind MERRY and SAM algorithms.

4 Lattice implementation of the adaptive SAM algorithm

The short autocorrelation of the effective channel is a property that is degraded by a long channel IR. Uniquely,

SAM is a blind adaptive channel-shortening algorithm that attempts to restore this short autocorrelation property.

4.1 SAM cost function properties

SAM algorithm performs a gradient descent of a specific cost function which is defined as the sum of the squares of autocorrelation coefficients of all lags greater than the desired channel memory. This cost function could be stated as follows:

$$J_{\text{SAM}}(\mathbf{w}) \doteq \sum_{l=P+1}^{L_c} |R_{cc}(l)|^2, \quad (13)$$

where $R_{cc}(l) = \sum_{i=0}^{L_c} c_i c_{i-l}$ is the autocorrelation of the overall IR c_i ($c_i = \sum_{l=0}^{L_w} w_l h_{i-l}$). Note that if $c_i = 0$ for $i \geq P+1$, then $R_{cc}(l) = 0$ for $l \geq P+1$ so that J_{SAM} will be zero. In other words, a short channel implies a short autocorrelation. Therefore, we must specify that the converse is not true: for example, let us consider the IR as follows:

$$c_i = \begin{cases} \alpha, & i = 0, \\ \alpha^2 - 1, & i = 1, \\ \alpha(\alpha^2 - 1), & 2 \leq i \leq L_c, \end{cases} \quad (14)$$

with $|\alpha| < 1$. For sufficiently large L_c , one has $R_{cc}(l) \approx \delta(l)$, whereas the IR cannot be said to be 'short' (just take $|\alpha|$ close to 1). This, normally, is due to the fact that c_i in Equation 14 resembles the IR of an all-pass system. Nevertheless, shortening $R_{cc}(l)$ seems to be a useful way that aims to attempt channel shortening in practical scenarios [16].

Another important observation regarding the SAM cost (13) is its invariance to flipping any of the zeros of the overall transfer function $C(z) = \sum_{i=0}^{L_c} c_i z^{-i}$ with respect to the unit circle, since this operation leaves $R_{cc}(l)$ unchanged. Since any zero of the TEQ transfer function $W(z) = \sum_{i=0}^{L_w} w_i z^{-i}$ is a zero of $C(z)$, flipping the TEQ zeros leaves the cost J_{SAM} unaltered. As a result, by considering any point on the cost surface, there will be 2^{L_w} points giving identical cost elsewhere, and in particular, any minimum (local or global) will be repeated 2^{L_w} times. These minima in reality may or may not yield a good performance in terms of shortening the effective channel IR. For example, let us consider a channel with IR h_i given by Equation 14, and then the two-tap TEQs $\mathbf{w}_1 = [1, -\alpha]^T$ and $\mathbf{w}_2 = [-\alpha, 1]^T$ result in identical short autocorrelation. However, while \mathbf{w}_1 yields good performance in terms of overall IR shortening, \mathbf{w}_2 does not.

As a way to ensure SAM algorithm convergence to the good TEQ zeros, canceling poles and causing the tail, we suggest a minimum phase constraint be imposed on the TEQ. This can be effectively implemented using the lattice structure. On the other hand, there is a necessary and

sufficient condition for this filter which allows it to be in a minimum phase: all reflection coefficients have magnitude less than unity. We propose to check this condition as the algorithm progresses.

4.2 Lattice SAM algorithm

In [16], it has been showed that the SAM cost function is merely approximated by the sum-squared autocorrelation of the TEQ output sequence:

$$J_{\text{SAM}} \approx \sum_{l=P+1}^{L_c} |R_{\tilde{y}\tilde{y}}(l)|^2. \quad (15)$$

Given the lattice TEQ output signal \tilde{y}_i , the purpose of the SAM algorithm is to update \mathbf{q} as a way to minimize Equation 15. This can be done by means of the following well-known gradient descent:

$$\mathbf{q}(i+1) = \mathbf{q}(i) - \mu \nabla_{\mathbf{q}} J_{\text{SAM}}, \quad (16)$$

where $\nabla_{\mathbf{q}} J_{\text{SAM}}$ is the gradient of J_{SAM} evaluated for $\mathbf{q} = \mathbf{q}(i)$ and μ is the step size. However, in order to implement Equation 16, we define an instantaneous cost function by replacing the expectation operation with a moving average over a defined window of length ν :

$$J_{\text{SAM}} \approx \sum_{l=P+1}^{L_c} \left| \frac{\xi(i, l)}{\nu} \right|^2, \quad \text{with } \xi(i, l) \doteq \sum_{j \in I_i} \tilde{y}_j \tilde{y}_{j-l}^* \quad (17)$$

where $I_i \doteq \{i, i-1, \dots, i-\nu+1\}$. The partial derivative of J_{SAM} with respect to lattice-TEQ coefficient, q_p , is denoted as follows:

$$\frac{\partial J_{\text{SAM}}}{\partial q_p} = \frac{1}{\nu^2} \sum_{l=P+1}^{L_c} \left(\xi(i, l) \frac{\partial \xi^*(i, l)}{\partial q_p} + \xi^*(i, l) \frac{\partial \xi(i, l)}{\partial q_p} \right). \quad (18)$$

Now, one has

$$\frac{\partial \xi^*(i, l)}{\partial q_p} \approx \nu_p^*(i) \tilde{y}_{i-l}, \quad (19)$$

$$\frac{\partial \xi(i, l)}{\partial q_p} \approx \nu_p^*(i-l) \tilde{y}_i. \quad (20)$$

In Equations 19 and 20, we have neglected the dependence of the signals $\nu_p(i)$ with the reflection coefficients (see Equation 12). In spite of that, in the summation, we have neglected all but one term in order to keep the computational complexity at bay. Let us define the vectors

$$\tilde{\mathbf{y}}_i \doteq [\tilde{y}_{i-P-1}, \tilde{y}_{i-P-2}, \dots, \tilde{y}_{i-L_c}]^T, \quad (21)$$

$$\Xi_i \doteq [\xi(i, P+1), \xi(i, P+2), \dots, \xi(i, L_c)]^T, \quad (22)$$

and the matrix

$$\tilde{\mathbf{V}}_i \doteq [\mathbf{v}_{i-P-1}, \mathbf{v}_{i-P-2}, \dots, \mathbf{v}_{i-L_c}]^T. \quad (23)$$

After absorbing the factor $\frac{1}{\sqrt{\nu}}$ into the step-size μ , the proposed update rule can then be written as follows:

$$\mathbf{q}(i+1) = \mathbf{q}(i) - \mu \left(\mathbf{v}_i^* \Xi_i^T \tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_i \tilde{\mathbf{V}}_i^T \Xi_i^* \right). \quad (24)$$

In other respects, the lattice version of SAM (Equation 24) requires on the order of $L_w(L_c - P)$ multiplications and additions per update, which is comparable to that of the original transversal implementation.

4.3 Normalized lattice SAM algorithm

As a way to improve the convergence rate of the lattice SAM algorithm, there is a variable step size that can be used in the update rule as follows:

$$q_p(i+1) = q_p(i) - \mu_i \gamma_p(i), \quad 1 \leq p \leq L_w, \quad (25)$$

where $\gamma_p(i) = v_p^*(i) \Xi_i^T \tilde{\mathbf{y}}_i + \tilde{\mathbf{y}}_i \tilde{\mathbf{V}}_i^T (p, :) \Xi_i^*$.

In order to determine the optimal value of μ_i , we propose to minimize, at each i , the *a posteriori* cost function $J_{\text{SAM}}^{\text{pos}}$ which mainly depends on the updated TEQ output signal $\tilde{\mathbf{y}}_i^{\text{pos}}$. Let us define *a posteriori* output signal as follows:

$$\tilde{\mathbf{y}}_i^{\text{pos}} = \tilde{\mathbf{r}}_i + \sum_{p=1}^{L_w} q_p(i+1) v_p^{\text{pos}}(i). \quad (26)$$

Thus, to determine $v_p^{\text{pos}}(i)$, we exploit the recursive relations in Equation 9, at *a posteriori* time. By neglecting the terms of proportional to μ^m for $m \geq 2$, it can be verified that $v_p^{\text{pos}}(i)$ and $u_p^{\text{pos}}(i)$ are recursively computed as follows:

$$v_p^{\text{pos}}(i) = v_p(i) - \mu_i t_p(i), \quad (27)$$

$$u_p^{\text{pos}}(i) = u_p(i) - \mu_i s_p(i), \quad (28)$$

where $t_p(i)$ and $s_p(i)$ are determined according to the following:

$$\begin{aligned} t_{p+1}(i) &= t_p(i-1) + q_p(i) s_p(i-1) + \gamma_p(i) u_p(i-1), \\ s_{p+1}(i) &= q_p(i) t_p(i) + \gamma_p(i) v_p(i) + s_p(i). \end{aligned} \quad (29)$$

By neglecting again terms in μ^m for $m \geq 2$ and replacing Equations 27 and 28 into Equation 26, the *a posteriori* TEQ output yields in the following notation:

$$\tilde{\mathbf{y}}_i^{\text{pos}} = \tilde{\mathbf{y}}_i - \mu_i \left(\sum_{p=1}^{L_w} q_p(i) t_p(i) + v_p(i) \gamma_p(i) \right). \quad (30)$$

We may define the *a posteriori* cost function at the i th iteration as follows:

$$J_{\text{SAM}}^{\text{pos}} \doteq \sum_{l=P+1}^{L_c} \left| \sum_{j=i-\nu+1}^{i-1} \tilde{y}_j \tilde{y}_{j-l}^* + \tilde{y}_i^{\text{pos}} \tilde{y}_{i-l}^* \right|^2. \quad (31)$$

By admitting that $\kappa(i) = \sum_{p=1}^{L_w} q_p(i) t_p(i) + v_p(i) \gamma_p(i)$. Through replacing the expression of $\tilde{\mathbf{y}}_i^{\text{pos}}$ from Equation 30 into Equation 31, it follows that

$$J_{\text{SAM}}^{\text{pos}} = J_{\text{SAM}} + \mu_i^2 |\kappa(i)|^2 \sum_{l=P+1}^{L_c} |\tilde{y}_{i-l}|^2 - 2\mu_i \Re \left(\Xi_i^H \tilde{\mathbf{y}}_i^* \kappa(i) \right).$$

The value of μ_i minimizing $J_{\text{SAM}}^{\text{pos}}$ is found to be

$$\mu_i^{\text{opt}} = \frac{\Re \left(\Xi_i^H \tilde{\mathbf{y}}_i^* \kappa(i) \right)}{|\kappa(i)|^2 \sum_{l=P+1}^{L_c} |\tilde{y}_{i-l}|^2}. \quad (32)$$

The updating rule for the lattice version of SAM will be disposed by the following:

$$q_p(i+1) = q_p(i) - \alpha \mu_i^{\text{opt}} \gamma_p(i), \quad 1 \leq p \leq L_w, \quad (33)$$

where $0 < \alpha \leq 2$ is a fixed stability factor.

5 Lattice implementation of the adaptive MERRY algorithm

Inspired by the behavior of the lattice implementation of the SAM cost function, we examine throughout this section the benefit that can be accomplished by implementing MERRY with the same structure. Therefore, we develop an adaptive algorithm relying on the lattice RLS MERRY implementation.

In the following, we seek to reformulate the minimization of the lattice MERRY cost function into least squares approach to adaptively implement the RLS algorithm. That is why we define a weighted least squares cost function as follows:

$$J_{\text{MERRY}} \doteq \sum_{l \in I_k} \lambda^{k-l} |\tilde{y}_{lM+P+\delta} - \tilde{y}_{lM+N+P+\delta}|^2, \quad (34)$$

where $I_k \doteq \{0, 1, \dots, k\}$, $0 < \lambda \leq 1$ is an exponential weighting factor which effectively limits the number of symbols based on which the cost function is minimized. Therefore, the RLS algorithm is detailed in the following quotation by expressing the gradient of the LS function:

$$J_{\text{MERRY}} = \sum_{l \in I_k} \lambda^{k-l} |\tilde{\mathbf{r}}_d(l) + \mathbf{q}^T \mathbf{v}_d(l)|^2, \quad (35)$$

where $\mathbf{v}_d(l) \doteq [v_{d1}(l), \dots, v_{dp}(l), \dots, v_{dL_w}(l)]^T$ with $v_{dp}(l) \doteq v_p(lM + P + \delta) - v_p(lM + N + P + \delta)$ and $\tilde{\mathbf{r}}_d(l) \doteq \tilde{r}_{lM+P+\delta} - \tilde{r}_{lM+N+P+\delta}$. If we take the gradient, evaluated at $(k+1)$ th MC symbol for $\mathbf{q} = \mathbf{q}(k)$, we have

$$\begin{aligned} \nabla_{\mathbf{q}} J_{\text{MERRY}} &= \mathbf{q}^T(k) \sum_{l \in I_k} \lambda^{k-l} \mathbf{v}_d(l) \mathbf{v}_d^H(l) \\ &+ \sum_{l \in I_k} \lambda^{k-l} \tilde{\mathbf{r}}_d(l) \mathbf{v}_d^H(l), \end{aligned} \quad (36)$$

when the cost function gradient is equal to zero, it results in the following:

$$\mathbf{q}^T(k) \sum_{l \in I_k} \lambda^{k-l} \mathbf{v}_d(l) \mathbf{v}_d^H(l) = - \sum_{l \in I_k} \lambda^{k-l} \tilde{r}_d(l) \mathbf{v}_d^H(l). \quad (37)$$

We then emphasize the correlation matrix as $\mathbf{V}_d(k) \doteq \sum_{l \in I_k} \lambda^{k-l} \mathbf{v}_d(l) \mathbf{v}_d^H(l)$ and the cross-correlation vector as $\mathbf{r}_v(k) \doteq \sum_{l \in I_k} \lambda^{k-l} \tilde{r}_d^*(l) \mathbf{v}_d(l)$; the LS solution for time instant k is then stated as follows:

$$\mathbf{q}^T(k) = -\mathbf{r}_v^H(k) \mathbf{V}_d(k)^{-1}. \quad (38)$$

We start the derivation of the recursive algorithm by expressing the deterministic correlations matrix, $\mathbf{V}_d(k)$, and the deterministic cross-correlation vector, $\mathbf{r}_v(k)$, in their recursive forms:

$$\begin{aligned} \mathbf{V}_d(k) &= \lambda \mathbf{V}_d(k-1) + \mathbf{v}_d(k) \mathbf{v}_d^H(k) \\ \mathbf{r}_v(k) &= \lambda \mathbf{r}_v(k-1) + \tilde{r}_d^*(k) \mathbf{v}_d(k). \end{aligned} \quad (39)$$

In order to generate the coefficient vector of Equation 38, we are interested in the inverse of the deterministic autocorrelation matrix $\mathbf{V}_d(k)$. For that task, the matrix inversion lemma comes in handy. Let us denote by $\mathbf{P}(k)$ the inverse of the matrix $\mathbf{V}_d(k)$ then

$$\mathbf{P}(k) = \lambda^{-1} \mathbf{P}(k-1) - \frac{\lambda^{-2} \mathbf{P}(k-1) \mathbf{v}_d(k) \mathbf{v}_d^H(k) \mathbf{P}(k-1)}{1 + \lambda^{-1} \mathbf{v}_d^H(k) \mathbf{P}(k-1) \mathbf{v}_d(k)}. \quad (40)$$

Through the equations which were already seen, the RLS algorithm becomes easy to follow and to implement without the need of matrix inversion. The forgetting factor λ presents the contribution of the previous samples; this makes the filter sensitive or not to the recent samples. The matrix $\mathbf{P}(k)$ is typically initialized as a scaled identity matrix $\rho \mathbf{I}_{L_w}$, where ρ is a large positive constant. The discussion resulted in a single equation to determine the lattice coefficient vector which minimizes the following cost function:

$$\mathbf{q}(k) = \left[\mathbf{I}_{L_w} - \tilde{\mathbf{P}}^T(k-1) \right] \mathbf{q}(k-1) - \mathbf{P}^T(k) \mathbf{v}_d^*(k) \tilde{r}_d(k), \quad (41)$$

where $\tilde{\mathbf{P}}(k-1) = \frac{\mathbf{v}_d(k) \mathbf{v}_d^H(k) \mathbf{P}(k-1)}{\lambda + \mathbf{v}_d^H(k) \mathbf{P}(k-1) \mathbf{v}_d(k)}$. The computational cost of this algorithm is approximately $2L_w^2$ multiplications and accumulate operations.

6 Simulation results

Accordingly, we tend to observe through this section the lattice structure performances of the blind MERRY and SAM algorithms. Indeed, the proposed implementation performances are given in terms of convergence properties of the algorithms and simulation results in the DSL environment.

6.1 SAM convergence properties

In the following, we propose to compare the cost surface behavior of the lattice to the transversal TEQs adjusted by the SAM criterion. Let us consider an example of the two-pole channel model where the transfer function is stated as follows:

$$H(z) = \frac{1 - 0.5z^{-1}}{(1 - 0.8e^{j\pi/2}z^{-1})(1 - 0.8e^{-j\pi/2}z^{-1})}. \quad (42)$$

The CP is composed of two samples, the lattice shortener has three taps ($L_w = 2$), whereas no noise is considered. Three-dimensional and contour plots of the SAM cost function are shown in Figures 3 and 4, in terms of the TEQ reflection coefficients q_1 and q_2 for the lattice implementation of spherical coordinates for direct transversal implementation. Nevertheless, we observe that in the working range $|q_p| < 1$, the cost of the lattice SAM through this example is convex and has a unique

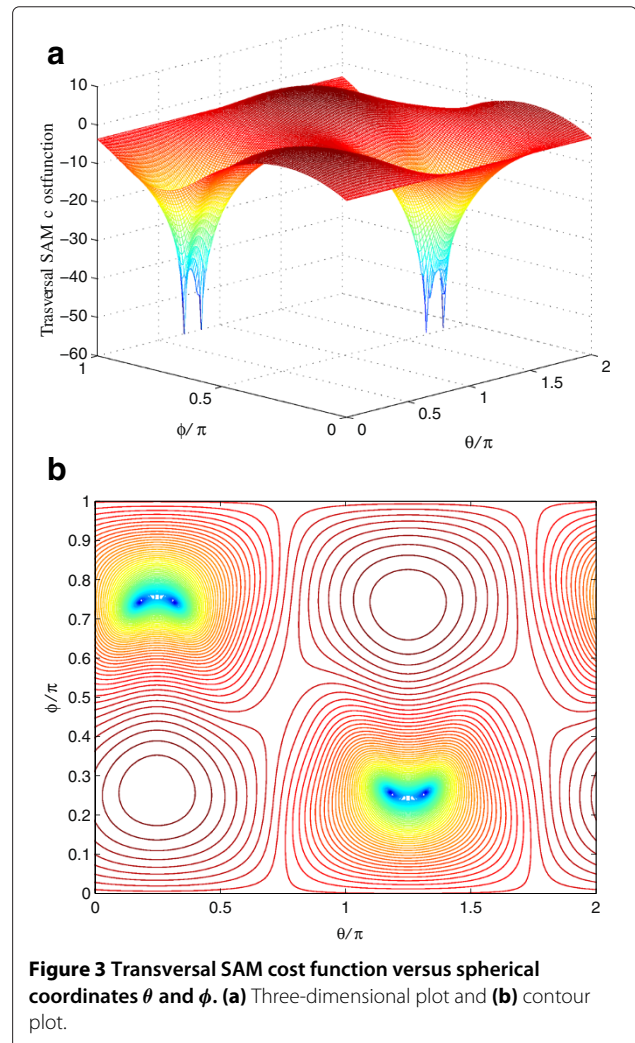
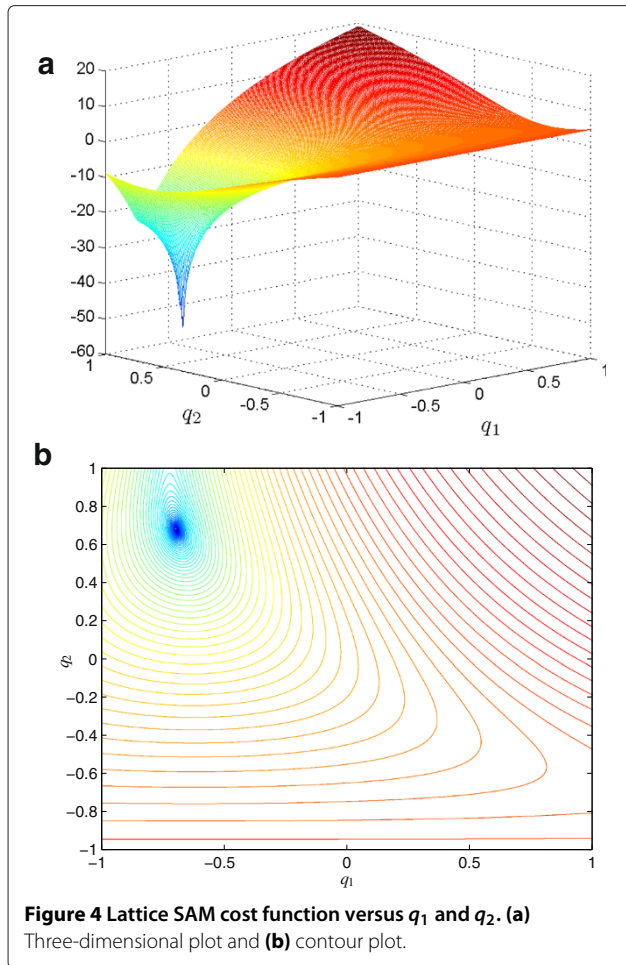


Figure 3 Transversal SAM cost function versus spherical coordinates θ and ϕ . (a) Three-dimensional plot and (b) contour plot.



minimum, while for the transversal structure there are four minima having equivalent values of the SAM cost.

6.2 Convergence properties of the RLS-MERRY algorithms

Furthermore, we aim to compare the convergence properties of the proposed lattice RLS-MERRY algorithm to their homology in the transversal implementation. The transversal RLS-MERRY algorithm is deeply detailed in [20]. Let us consider the same example of the channel transfer function given in the previous paragraph. Thus, to examine the RLS-MERRY algorithms convergence to the optimal solution, the mean square deviation (MSD) performance metric is used. The MSD with respect to the optimal coefficients corresponds to the minimum of the MERRY cost function:

$$\text{MSD} \doteq \frac{\|\mathbf{w}(k) - \mathbf{w}_\infty\|^2}{\|\mathbf{w}_\infty\|^2}. \quad (43)$$

The optimal TEQ, \mathbf{w}_∞ , is obtained by averaging the filter coefficient, determined after the algorithm

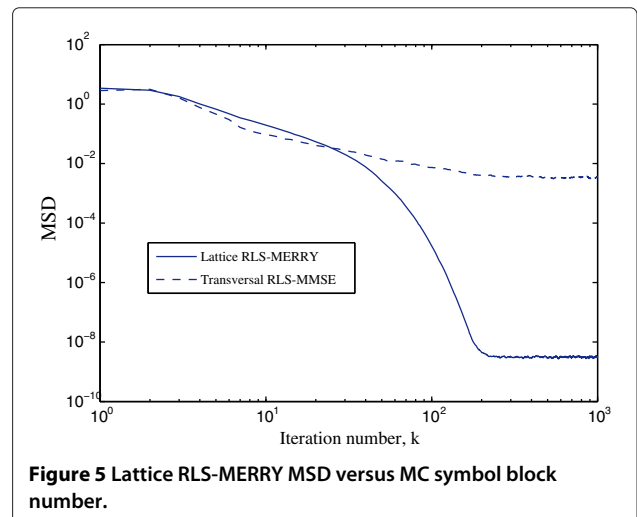
convergence, over 100 independent runs. In the case of lattice structure, the RLS-MERRY reflection coefficients are computed using the update rule (Equation 41), then the mapping from lattice to transversal is made through the Levinson-Durbin recursions. Figure 5 shows the MSD variation during lattice RLS-MERRY algorithm compared with the transversal RLS-MERRY TEQ. The MSD curves are obtained by averaging 100 independent runs. Notice that, for the two RLS-MERRY algorithms, the forgetting factor is $\lambda = 0.99$ and the scaled coefficient is $\rho = 100$.

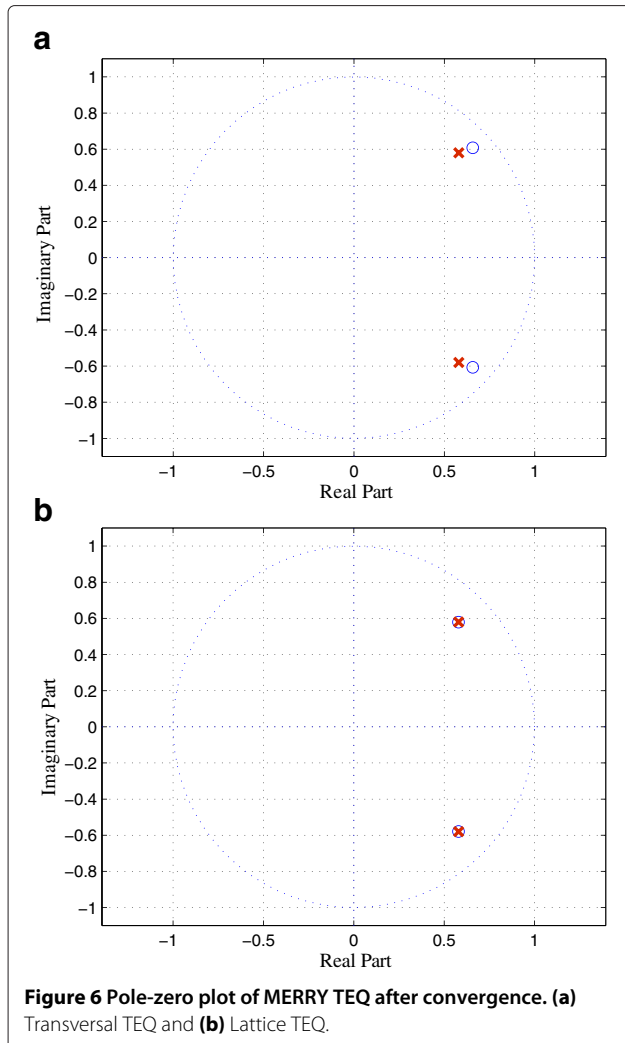
Based on the results shown in the MSD plot in Figure 5, we may observe that for the same rate convergence, the steady-state value of the MSD produced by the lattice implementation is much smaller than in the case of transversal implementation. To confirm this result, we show in Figure 6 the TEQ zero locations with transversal and lattice structures (a) and (b). We remark that when lattice structure is implemented, the TEQ zeros are very close to the channel poles.

6.3 DSL environment simulation

Throughout this section, we compare the performance of the different studied TEQ implementations, specifically in the considered DSL environment where the transmission channel is modeled as an IIR channel. Indeed, it is shown that, in twisted pair lines, the channel is well modeled by an IIR filter with a slowly decaying IR [8]. This means that the channel transfer function presents poles very close to the UC.

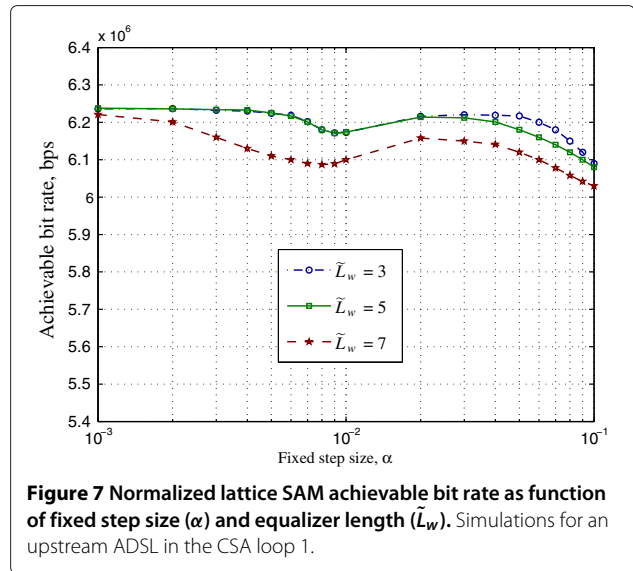
To shorten the DSL channel, an effective TEQ will place zeros on the critical poles to cancel them out, which motivates the use of the lattice structure to implement a minimum phase TEQ and to ensure convergence of the multimodal SAM algorithm to a good stationary points. Further, the lattice structure is adopted to implement the blind MERRY algorithm.





To compare the performance of different implementations of adaptive equalizers, simulation results are presented for a set of standard ADSL channels. The transfer functions for the different channels include the model of the copper wire itself as well as the digital and analog transmit and receive filters. Hence, we consider carrier serving area (CSA) test loops combined with a plain old telephone service splitter. Actually, the parameters were chosen to match the downstream ADSL standard system: the cyclic prefix is 32 samples, and the FFT size is 512. The external noise consists of -140 dBm/Hz additive white Gaussian noise as given in [21], and no crosstalk is considered. In the following simulations, we will focus only on equalization issues without optimizing the power allocation of the different subcarriers. We suppose that 16-QAM signaling is used on all of the subchannels, and the sampling frequency is 2.208 MHz.

By analyzing the lattice implementation performances of SAM and RLS-MERRY algorithms, we disposed, as



performance metric, the bit rate. The achievable bit rate of the multicarrier channel can be written as the sum of the capacities of AWGN subchannels

$$R \doteq \sum_{n \in V} \log_2 \left(1 + \frac{\text{SNR}_n}{\Gamma} \right), \quad (44)$$

where V is the index set of used subchannels and Γ is the SNR gap to Shannon capacity, which is assumed to be constant over all subchannels. Actually, the SNR gap is a function of the bit error probability aimed at [22], we take $\Gamma = 9.8$ dB. The subchannel SNR $_n$, as defined in [13], incorporates both interference and noise distortion. The bit rate was measured and averaged over 100 realizations of each channel-shortening procedure.

To begin the implementation of the normalized version of lattice SAM algorithm, we consider the moving average window of length $\nu = 100$. Therefore, Figure 7 indicates the normalized lattice SAM achievable bit rate versus the fixed step size, α , for different equalizer length $\tilde{L}_w = L_w + 1$. For the other CSA loops, the optimal values of TEQ order and fixed step size as well as the achievable bit rate are summarized in Table 1. Also, Figure 8 shows the bit rate evolution over time (sample index), by comparing the normalized lattice and the original transversal implementations of SAM algorithm. From the last figure, we

Table 1 Normalized lattice SAM achievable bit rate

CSA loop number	1	2	3	4	5	6	7	8
\tilde{L}_w	3	7	11	21	11	15	20	27
α_{opt}	0.05	0.1	0.05	0.05	0.05	0.05	0.001	0.001
R (Mbps)	6.24	7.72	6.62	6.38	6.53	7.17	5.52	4.89

It is a function of optimal values of equalizer order, \tilde{L}_w , and fixed step size, α , for the eight CSA loops.

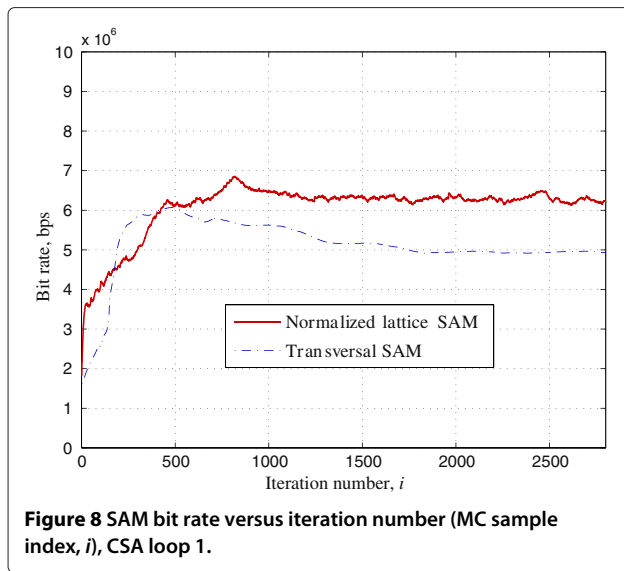


Figure 8 SAM bit rate versus iteration number (MC sample index, i), CSA loop 1.

can see that the normalized lattice SAM equalizer outperforms the transversal SAM algorithm (which is updated with a moving average implementation as well).

Notably, the RLS-MERRY channel shortener algorithm depends on parameter synchronization δ . Hence, Figure 9 shows the RLS-lattice MERRY achievable bit rate as a function of the delay parameter, δ , and the TEQ order, \tilde{L}_w , for CSA loop 1. For the other CSA loops, the optimal values of delay parameter and equalizers' length are given in Table 2 where the achievable bit rate of RLS-transversal MERRY and RLS-lattice MERRY algorithms are also computed. However, Figure 10 displays the bit rate evolution over iteration number (MC symbol number), for comparing the RLS lattice and the original RLS

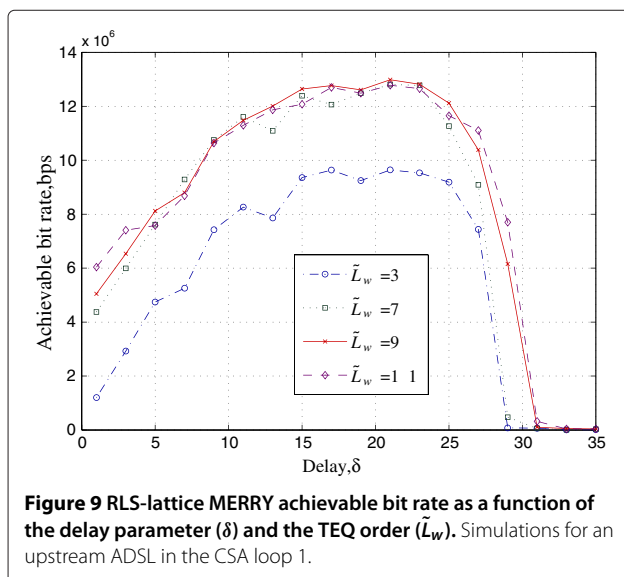


Figure 9 RLS-lattice MERRY achievable bit rate as a function of the delay parameter (δ) and the TEQ order (\tilde{L}_w). Simulations for an upstream ADSL in the CSA loop 1.

Table 2 Achievable bit rate as a function of optimal values of equalizer order \tilde{L}_w and delay parameter δ

CSA loop number	Transversal RLS-MERRY			Lattice RLS-MERRY		
	\tilde{L}_w	δ_{opt}	R (Mbps)	\tilde{L}_w	δ_{opt}	R (Mbps)
1	7	20	5.59	9	21	12.96
2	9	18	7.03	11	18	13.54
3	13	22	5.42	17	25	11.02
4	11	20	5.15	20	21	10.76
5	5	18	5.35	19	23	11.12
6	3	18	4.92	16	18	10.25
7	11	30	5.15	21	28	10.19
8	9	30	5.06	21	30	10.61

This is for the eight CSA loops shortened with the transversal RLS-MERRY and lattice RLS-MERRY algorithms.

transversal implementations of MERRY. Easily, we can see that the lattice structure results in a noticeable improvement in terms of achievable bit rate. Be reminded that the unit first tap constraint is imposed by the lattice structure for all studied algorithms. Also, we can see that the lattice implementation of RLS-MERRY equalizer outperform the lattice SAM TEQ for the eight CSA loops.

7 Conclusions

Specific framework of MC transmission channel has been studied in this paper. We have considered slowly decaying recursive channel model. Therefore, on the basis of the pole-zero model, we have shown that the channel poles introduce a term of interference in each received MC symbols; thus, to shorten this kind of channel, an effective TEQ will place their zeros on the critical poles to cancel them out. However, inaccurate zero location may result in limited channel-shortening performance, which

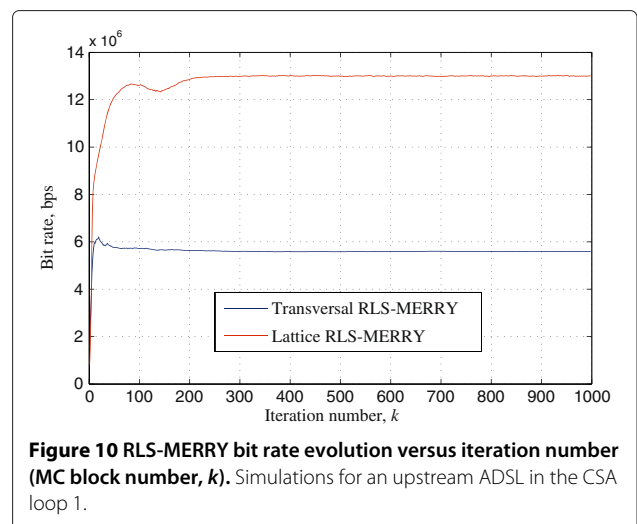


Figure 10 RLS-MERRY bit rate evolution versus iteration number (MC block number, k). Simulations for an upstream ADSL in the CSA loop 1.

motivates the use of the lattice TEQ structure aiming to place zeros very close to the channel poles.

Furthermore, for linear algebra resolution and low complexity reasons, we have adopted only adaptive algorithms to update the TEQ which is designed using the lattice structure. Hence, an implementation of the lattice SAM and MERRY adaptive algorithms has been proposed. With computational complexity similar to that of the original versions of transversal algorithms, the convergence of lattice algorithms has been observed. In addition, we have showed that when it is compared to the transversal direct implementation, the lattice TEQ algorithms achieve higher performance in terms of achievable bit rate.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

The work of R. Lopez-Valcarce is supported by the Spanish Government, ERDF funds (TEC2010-21245-C02-02/TCM DYNACS, CONSOLIDER-INGENIO 2010 CSD2008-00010 COMONSENS), and the Galician Regional Government (CN 2012/260 AtlantTIC).

Author details

¹Cosim Laboratory, High School of Communications of Tunis (SUPCOM), Ariana 2088, Tunisia. ²Department of Signal Theory and Communications, University of Vigo, Vigo 36310, Spain.

Received: 18 January 2013 Accepted: 30 August 2013

Published: 19 November 2013

References

1. JA Bingham, Multicarrier modulation for data transmission: an idea whose time has come. *IEEE Commun. Mag.* **28**, 5–14 (1990)
2. T Starr, JM Cioffi, PJ Silvermann, *Understanding Digital Subscriber Line Technology* (Prentice Hall, Englewood Cliffs, 1999)
3. IEEE, IEEE 802.11 wireless local area networks: the working group WLAN standards (2010). <http://grouper.ieee.org/groups/802/11>. Accessed 10/07/2012
4. ETSI Normalization Committee, High performance radio local area networks (HIPERLAN) type 2 (1995). <http://www.etsi.org>. Accessed 1 Sept 2012
5. ETSI Normalization Committee, Radio broadcasting systems; digital audio broadcasting (DAB) to mobile, portable and fixed receivers (2006). <http://www.etsi.org>. Accessed 1 Sept 2012
6. ETSI Normalization Committee, Digital video broadcasting (DVB): framing structure, channel coding and modulation for digital terrestrial television (2004). <http://www.etsi.org>. Accessed 1 Sept 2012
7. RK Martin, K Vanbleu, M Ding, G Ysebaert, M Milosevic, BL Evans, M Moonen, CR Johnson, Jr., Unification and evaluation of equalization structures and design algorithms for discrete multitone modulation systems. *IEEE Trans. Signal Process.* **53**, 3880–3894 (2005)
8. WY Chen, *DSL Simulation Techniques and Standard Development for Digital Subscriber Line Systems* (McMillan Technical Publishing, Indianapolis, 1998)
9. N Al-Dhahir, AH Sayed, JM Cioffi, Stable pole-zero modeling of long FIR filters with application to the MMSE-DFE. *IEEE J. Selected Areas Comm.* **45**, 508–513 (1997)
10. PJW Melsa, RC Younce, CE Rohrs, Impulse response shortening for discrete multitone transceivers. *IEEE Trans. Commun.* **44**, 1662–1672 (1996)
11. N Al-Dhahir, JM Cioffi, Bandwidth-optimized reduced-complexity equalized multicarrier transceiver. *IEEE J. Selected Areas Comm.* **45**, 948–956 (1997)
12. G Arslan, BL Evans, S Kiaei, Optimum Channel Shortening for Discrete Multitone Transceivers. *Int. Conf. Acoustics, Speech, Signal Process.* **5**, 2965–2968 (2000)
13. G Arslan, BL Evans, S Kiaei, Equalization for discrete multitone receivers to maximize bit rate. *IEEE Trans. Signal Process.* **49**, 3123–3135 (2001)
14. E Ben Salem, C Sofiane, H Sofiane, H Besbes, R López-Valcarce, Fractionally spaced channel shortening by subchannels SINR maximization in multicarrier system. *Proceedings of 6th International Symposium on Image and Signal Processing and Analysis (ISPA 2009)* (IEEE, Piscataway, 2009), pp. 53–58
15. RK Martin, J Balakrishnan, WA Sethares, CR Johnson, A blind adaptive TEQ for multicarrier systems. *IEEE Signal Process. Lett.* **9**, 341–343 (2002)
16. J Balakrishnan, RK Martin, CR Johnson, Jr., Blind, adaptive channel shortening by sum-squared autocorrelation minimization (SAM). *IEEE Trans. Signal Process.* **51**, 3086–3093 (2003)
17. A Betsler, E Zeheb, Reduced order IIR approximation to FIR digital filters. *IEEE Trans. Signal Process.* **39**, 2540–2544 (1991)
18. P Crespo, M Honig, Pole-Zero decision feedback equalization with a rapidly converging adaptive IIR algorithm. *IEEE J. Selected Areas Comm.* **6**, 817–829 (1991)
19. M Bellanger, *Adaptive Digital Filters and Signal Processing* (Marcel Dekker Inc., New York, 1987)
20. RK Martin, Fast-converging blind adaptive channel-shortening and frequency-domain equalization. *IEEE Trans. Signal Process.* **55**, 102–110 (2007)
21. K Sistanizadeh, *Loss characteristics of the proposed canonical, ADSL loops with 100-ohm termination at 70, 90, and 120 F* (ANSI T1E1.4 Committee Contribution 161, Washington, DC, 1991)
22. GD Forney, MV Eyuboglu, Combined equalization and coding using precoding. *IEEE Commun. Mag.* **12**, 25–34 (1991)

doi:10.1186/1687-6180-2013-174

Cite this article as: Ben Salem et al.: Lattice implementation of adaptive channel shortening in multicarrier transmission over IIR channels. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:174.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com