Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2006, Article ID 25686, Pages 1–12
DOI 10.1155/WCN/2006/25686

Low Complexity Turbo Equalization for High Data Rate Wireless Communications

Dimitris Ampeliotis and Kostas Berberidis

Computer Engineering and Informatics Department and CTI/R&D, University of Patras, 26500 Rio-Patras, Greece

Received 21 December 2005; Revised 3 July 2006; Accepted 21 July 2006

Recommended for Publication by Huaiyu Dai

Soft interference cancellers (SICs) have been proposed in the literature as a means for reducing the computational complexity of the so-called turbo equalization receiver architecture. Soft-input-soft output (SISO) equalization algorithms based on linear filters have a tremendous complexity advantage over trellis-diagram-based SISO equalizers, especially for high-order modulations and long-delay spread frequency selective channels. In this paper, we modify the way in which the SIC incorporates soft information. In existing literature the input to the cancellation filter is the expectation of the symbols based solely on the apriori probabilities coming from the decoder, whereas here we propose to use the conditional expectation of those symbols, given both the apriori probabilities and the received sequence. This modification results in performance gains at the expense of increased computational complexity, as compared to previous SIC-based schemes. However, by introducing an approximation to the aforementioned algorithm a linear complexity SISO equalizer can be derived. Simulation results for an 8-PSK constellation and hostile radio channels have shown the effectiveness of the proposed algorithms in mitigating the intersymbol interference (ISI).

1. INTRODUCTION

Turbo equalization [1] was motivated by the breakthrough of turbo codes [2] and has emerged as a promising technique for drastic reduction of the intersymbol interference in frequency selective wireless channels. Unfortunately, the trellis-diagram-based turbo equalizer of [1] can be a heavy computational burden to wireless systems with limited processing power, especially in cases the wireless channel has long-delay spread. Thus, a number of alternative, low complexity, equalization methods that can be properly incorporated in the generic turbo equalization scheme have been proposed, offering good complexity/performance trade-offs.

In this context, it was proposed in [3] to replace the trellis-diagram-based equalizer by an adaptive SIC of linear complexity. In [4], an improved extension of the algorithm of [3] was presented. In [5], an MMSE SIC for the receiver of a coded CDMA system was suggested. In [6] an MMSE-optimal equalizer based on linear filters was derived and it was proven that several other algorithms (such as the one in [3]) could be viewed as approximations of this one. In [7], the MMSE-optimal equalizer of [6] was used as a starting point for the derivation of two approximate equalizers. In particular, the so-called APPLE equalizer was derived in the case of "weak" a priori information, and the "matched filtering" equalizer in the case of "strong" a priori information. Moreover, a decision criterion was used for selecting among the aforementioned equalizers, leading to the so-called SWITCHED approach. In [8], a modified version of the sliding window algorithm of [6] was derived having similar performance with the original one while offering reduced computational complexity via the use of a Cholesky factorization technique. In [9], the authors modified the algorithm of [6] which involves complex valued matrices into an algorithm that uses augmented real valued matrices yielding better performance at approximately the same complexity. More recently, the authors of [10] derived the theoretical (time invariant) transfer function of an MMSE optimal equalizer and showed that this equalizer reduces to a linear equalizer in the case of no a priori information or to an MMSE SIC in the case of perfect a priori information. Their algorithm was shown to be identical to a low complexity algorithm derived in [11] in the case where the equalizer filters are restricted to finite length. In [12], the incorporation of channel output information in the computation of the input to the cancellation filter of the SIC was investigated.

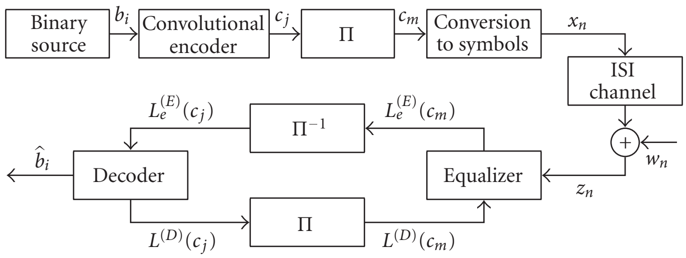

Figure 1: The model of transmission.

In the proposed turbo equalizer, we split the problem of a priori probabilities-based equalization into two distinct MMSE optimization problems. The first problem consists in the estimation of past and future symbols using a priori probabilities and channel output information, while the second problem is the estimation of the current symbol based on past and future symbols. The solution to the first problem is to use an MMSE equalizer similar to the one developed in [11], but modified appropriately so as to provide all the required symbols instead of computing only the current symbol estimate. For the second problem we suggest using an MMSE SIC which has been designed under the assumption that its input symbols are actually correct symbols (in practice they are provided by the aforementioned equalizer). As shown experimentally, the proposed approach, so-called conditional expectation-soft interference canceller (CE-SIC), exhibits similar performance to the exact MMSE solution of [11], at a similar computational cost. Although the exact implementation of the CE-SIC does not enjoy any advantage over the exact equalizer of [11], it leads to the derivation of an approximate version, so-called approximate conditional expectation-soft interference canceller (ACE-SIC), which has linear complexity. Simulation results have shown that the proposed algorithm exhibits very good performance characteristics that make it suitable for high data rate wireless communications.

The rest of this paper is organized as follows: in Section 2, the communication system model is formulated. In Section 3, the CE-SIC algorithm is derived. Then, an approximation to the exact algorithm is introduced and the ACE-SIC algorithm is formulated in Section 4. In Section 5, for comparison reasons, various SISO equalizers that are suitable for turbo equalization are categorized according to their computational complexity. Finally, in Section 6, simulation results verifying the performance of the proposed equalizers are provided and the work is concluded in Section 7.

2. SYSTEM MODEL

Let us consider the communication system depicted on Figure 1. A discrete memoryless source generates binary data b_i, $i = 1, \dots, S$. These data, in blocks of length S, enter a convolutional encoder of rate R, so that new blocks of S/R bits (c_j, $j = 1, \dots, S/R$) are created, where S/R is assumed integer and no trellis termination is assumed. The output of the convolutional encoder is then permuted by an interleaver, denoted as Π, so as to form the corresponding block of bits c_m, $m = 1, \dots, S/R$. The output of the interleaver is then grouped into groups of q bits each (with S/Rq also assumed integer) and each group is mapped into a 2^q-ary symbol from the alphabet $A = \{\alpha_1, \alpha_2, \dots, \alpha_{2^q}\}$. The resulting symbols x_n, $n = 1, \dots, S/Rq$, are finally transmitted through the channel.

We assume that the communication channel is frequency selective and constant during the packet transmission, so that the output of the channel (and input to the receiver) can be modeled as

$$z_n = \sum_{i=-L_1}^{L_2} h_i x_{n-i} + w_n, \quad (1)$$

where L_1, $L_2 + 1$ denote the lengths of the anticausal and causal parts, respectively, of the channel impulse response. The output of the multipath channel is corrupted by complex-valued additive white Gaussian noise (AWGN) w_n.

At the receiver, we employ an equalizer to compute soft estimates of the transmitted symbols. As a part of the equalizer is also a scheme that transforms the soft estimates of the symbols into soft estimates of the bits that correspond to those symbols. The output of the equalizer is the log-likelihood $L_e^{(E)}(c_m)$, $m = 1, \dots, S/R$, where the subscript stands for "extrinsic" and the superscript denotes that this log-likelihood ratio comes from the equalizer. The operator $L(\cdot)$ applied to a binary random variable y is defined as

$$L(y) = \ln\left(\frac{\Pr(y=1)}{\Pr(y=0)}\right). \quad (2)$$

In the sequel, the log-likelihood ratios $L_e^{(E)}(c_m)$ are deinterleaved and enter a soft convolutional decoder, implemented here as a MAP decoder. We stretch the fact that the convolutional decoder operates on the code bits c_j of the code and not on the information bits b_i. The log-likelihood ratios $L^{(D)}(c_j)$ at the output of the decoder are first interleaved and then enter the SISO equalizer as a priori probabilities information. These a priori probabilities are combined with the output of the channel via a SISO equalization algorithm which computes new soft estimates about the transmitted bits. Thus, the above mentioned procedure can be iterated a number of times. The authors of [13] have proposed three stoping criteria that can be used to terminate the iterative procedure when no further performance improvement is possible, thus reducing the computational complexity of the

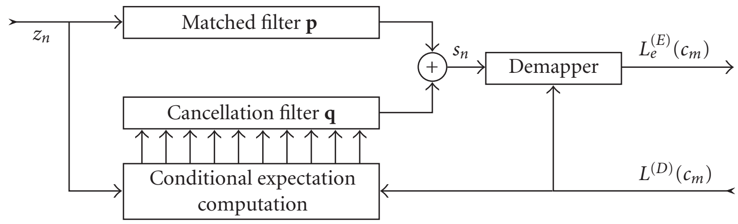

Figure 2: The proposed CE-SIC equalizer.

receiver. These stoping criteria consist in (a) using the cross entropy, (b) monitoring the hard decisions at the output of the decoder (whether they remain the same as in the previous iteration), and (c) evaluating a risk function that measures the reliability of the decisions at the output of the decoder. In any case, at the last iteration, the decoder operates on the information bits b_i and provides the hard estimates $\widehat{b}_i$. Although in our experiments we have used a fixed number of iterations, the above-mentioned stoping criteria could apply to our method as well.

It is interesting to note that, as it was also the case in [10, 14, 15], we observed that if the output of the MAP decoder is extrinsic then nonnegligible performance degradation occurs for high-order modulations. Thus, in this work we use the entire a posteriori probability information at the output of the decoder as input to the equalizer.

3. THE CONDITIONAL EXPECTATION SIC (CE-SIC)

The CE-SIC shown in Figure 2 is a device consisting of three distinct units, namely, an MMSE soft interference canceller, a conditional expectation computation unit that delivers symbol estimates to the cancellation filter of the SIC, and a Demapper. The conditional expectation computation unit provides estimates of the transmitted symbols given the a priori information coming from the decoder and the output of the channel. Based on these estimates the SIC forms an estimate s_n of the current symbol. Finally, the Demapper exploits the output of the SIC and the a priori bit probabilities to compute the corresponding a posteriori bit probabilities. In the following, we describe in detail each of these units.

3.1. MMSE soft interference cancellation

The SIC [3, 6] consists of two filters, that is, the matched filter

$$\mathbf{p} = \left[p_{-k} \cdots p_0 \cdots p_l\right]^T, \qquad M = k + l + 1 \tag{3}$$

and the cancellation filter

$$\mathbf{q} = \left[q_{-K} \cdots q_{-1} \;\; 0 \;\; q_1 \cdots q_N\right]^T. \tag{4}$$

The input to the filter $\mathbf{p}$ is the sampled output of the channel at the symbol rate, whereas the input to the cancellation filter consists of past and future symbols. The output s_n of the SIC is the sum of the outputs of the two filters, that is,

$$s_n = \mathbf{p}^H \mathbf{z}_n + \mathbf{q}^H \widetilde{\mathbf{x}}_n, \tag{5}$$

where $\mathbf{z}_n = [z_{n+k} \cdots z_n \cdots z_{n-l}]^T$ and $\widetilde{\mathbf{x}}_n = [\widetilde{x}_{n+K} \cdots \widetilde{x}_n \cdots \widetilde{x}_{n-N}]^T$. Minimizing the mean squared error $E[|s_n - x_n|^2]$ and assuming that the cancellation filter contains correct symbols, then the involved filters are given by the equations (see the appendix):

$$\begin{aligned} \mathbf{p} &= \frac{1}{\sigma_w^2 + E_h}\mathbf{H}\mathbf{d}, \\ \mathbf{q} &= -\mathbf{H}^H\mathbf{p} + \mathbf{d}\mathbf{d}^T\mathbf{H}^H\mathbf{p}, \end{aligned} \tag{6}$$

where $N = l + L_2$, $K = L_1 + k$, $E_h = \mathbf{d}^T\mathbf{H}^H\mathbf{H}\mathbf{d}$ is the energy of the channel and $\mathbf{H}$ is the $M \times (K+N+1)$ channel convolution matrix. $\mathbf{H}$ and $\mathbf{d}$ are defined as

$$\begin{aligned} \mathbf{H} &= \begin{bmatrix} h_{-L_1} & \cdots & h_{L_2} & 0 & \cdots & 0 \\ 0 & \ddots & h_{L_2-1} & h_{L_2} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{-L_1} & \cdots & h_{L_2} \end{bmatrix}, \\ \mathbf{d} &= \left[\mathbf{0}_{1\times K} \;\; 1 \;\; \mathbf{0}_{1\times N}\right]^T, \end{aligned} \tag{7}$$

respectively. From the above equations it is clear that the output s_n of the canceller does not depend on the symbol estimate $\widetilde{x}_n$ since the central tap (q_0) of the cancellation filter has been set to zero. At this point, it is convenient to define a function $\mathcal{T}(\mathbf{v}, L, C)$ which transforms the row vector $\mathbf{v}$ into a $L \times C$ Toeplitz matrix as

$$\mathcal{T}\left([v_1 v_2 \cdots v_d], L, C\right) = \underbrace{\begin{bmatrix} v_1 & \cdots & v_d & 0 & \cdots & 0 \\ 0 & \ddots & v_{d-1} & v_d & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & v_1 & \cdots & v_d \end{bmatrix}}_{C \text{ columns}} L \text{ rows}. \tag{8}$$

Thus, according to (8), the convolution matrix $\mathbf{H}$ can be written as

$$\mathbf{H} = \mathcal{T}\left(\mathbf{h}^T, M, K + N + 1\right), \tag{9}$$

where $\mathbf{h} = [h_{-L_1} \cdots h_0 \cdots h_{L_2}]^T$.

3.2. Conditional expectation computation

Let us first see how the mean and variance of the transmitted symbols may be computed based solely on a priori probabilities. If we define the function that maps the bits into symbols as

$$\alpha_i = \mathcal{A}(\beta_{i,1}, \beta_{i,2}, \dots, \beta_{i,q}), \tag{10}$$

where $\beta_{i,j} \in \{0,1\}$ and $\alpha_i \in A$, then the transmitted symbols are given by

$$x_n = \mathcal{A}(c_{(n-1)\cdot q+1}, c_{(n-1)\cdot q+2}, \dots, c_{(n-1)\cdot q+q}), \quad n = 1, \dots, \frac{S}{Rq}, \tag{11}$$

where $c_{(n-1)\cdot q+j}$ correspond to the output bits of the interleaver. Based on the assumption that these bits are mutually independent, we have

$$\Pr\{x_n = \alpha_i\} = \prod_{j=1}^{q} \Pr\{c_{(n-1)\cdot q+j} = \beta_{i,j}\}, \tag{12}$$

where the latter probabilities come from the decoder after converting the log-likelihood ratios to bit probabilities. Based on the above symbol probabilities we have

$$\begin{aligned} \overline{x}_n &= E[x_n] = \sum_{i=1}^{2^q} \alpha_i \Pr\{x_n = \alpha_i\}, \\ \sigma_{x_n}^2 &= E[|x_n|^2] - E[x_n]E[x_n^*] = 1 - |\overline{x}_n|^2 \end{aligned} \tag{13}$$

assuming unit average symbol power $E[|x_n|^2]$. The symbol $*$ denotes the complex conjugate operation. It should be made clear at this point that the operator $E[\cdot]$ is computed taking into account the a priori probabilities $\Pr\{x_n = \alpha_i\}$ at the output of the channel decoder. Thus, $\overline{x}_n$ is conditioned on the output of the decoder.

The conditional expectation computation unit sets the input to the cancellation filter of the SIC equal to

$$\widetilde{\mathbf{x}}_n = E[\mathbf{x}_n \mid \mathbf{z}'_n] \tag{14}$$

instead of $\overline{\mathbf{x}}_n = E[\mathbf{x}_n]$ as proposed in [3], which is only conditioned on the a priori probabilities at the output of the decoder. Vector $\mathbf{z}'_n$ is defined as

$$\mathbf{z}'_n = [z_{n+k+K} \cdots z_n \cdots z_{n-l-N}]^T \tag{15}$$

and its length is selected so that all elements of $\widetilde{\mathbf{x}}_n$ use information from a window of at least $M = k + l + 1$ samples of the sequence z_n. We may express vector $\mathbf{z}'_n$ in matrix form as

$$\mathbf{z}'_n = \mathbf{H}'\mathbf{x}'_n + \mathbf{w}'_n, \tag{16}$$

where $\mathbf{x}'_n = [x_{n+2K} \cdots x_n \cdots x_{n-2N}]^T$, vector $\mathbf{w}'_n$ contains the corresponding noise samples, and $\mathbf{H}'$ is the $(M' = K + M + N) \times (2(K+N)+1)$ channel convolution matrix defined similarly to matrix $\mathbf{H}$. Thus, [16, Theorem 10.3] concerning the Bayesian general linear model may be applied assuming that the symbols $\mathbf{x}'_n$ have a prior p.d.f $\mathcal{N}(\overline{\mathbf{x}}'_n, \mathbf{C}_{\mathbf{x}'_n})$, where

$$\mathbf{C}_{\mathbf{x}'_n} = \operatorname{diag}([\sigma_{x_{n+2K}}^2 \cdots \sigma_{x_n}^2 \cdots \sigma_{x_{n-2N}}^2]) \tag{17}$$

is the diagonal covariance matrix of the symbols based solely on a priori probabilities, and $\overline{\mathbf{x}}'_n = E[\mathbf{x}'_n]$. Thus,

$$\begin{aligned} \widetilde{\mathbf{x}}'_n &= E[\mathbf{x}'_n \mid \mathbf{z}'_n] \\ &= \overline{\mathbf{x}}'_n + \mathbf{C}_{\mathbf{x}'_n}\mathbf{H}'^H(\mathbf{H}'\mathbf{C}_{\mathbf{x}'_n}\mathbf{H}'^H + \mathbf{C}_{w'})^{-1}(\mathbf{z}'_n - \mathbf{H}'\overline{\mathbf{x}}'_n), \end{aligned} \tag{18}$$

where $\mathbf{C}_{w'} = \sigma_w^2\mathbf{I}_{M'}$ is the covariance matrix of the noise vector $\mathbf{w}'$. Finally, the required vector $\widetilde{\mathbf{x}}_n$ is extracted from $\widetilde{\mathbf{x}}'_n$ by simply keeping only the $K + N + 1$ required elements. At this point, it is interesting to mention that (18) is simply a "block" version of the linear MMSE equalizer proposed in [11], in the sense that instead of computing only one symbol estimate, it estimates a vector of symbols.

Now let us impose the extrinsic-information constraint on $\widetilde{\mathbf{x}}'_n$ which implies that this quantity should not depend on the a priori probabilities about symbol x_n. This modification yields

$$\begin{aligned} E^{(e)}[\mathbf{x}'_n] &= \overline{\mathbf{x}}'_n - \mathbf{D}'\overline{\mathbf{x}}'_n, \\ \mathbf{C}_{\mathbf{x}'_n}^{(e)} &= \mathbf{C}_{\mathbf{x}'_n} + (1 - \sigma_{x_n}^2)\mathbf{D}', \end{aligned} \tag{19}$$

where $\mathbf{D}' = \mathbf{d}'\mathbf{d}'^T$, $\mathbf{d}' = [\mathbf{0}_{1\times 2K} \; 1 \; \mathbf{0}_{1\times 2N}]^T$, and the superscript (e) stands for "extrinsic". If we define matrix $\mathbf{F}_n'^{(e)} = \mathbf{H}'^H(\mathbf{H}'\mathbf{C}_{\mathbf{x}'_n}^{(e)}\mathbf{H}'^H + \sigma_w^2\mathbf{I})^{-1}$, and substitute into (18), we get

$$\begin{aligned} \widetilde{\mathbf{x}}_n'^{(e)} &= E^{(e)}[\mathbf{x}'_n] + \mathbf{C}_{\mathbf{x}'_n}^{(e)}\mathbf{F}_n'^{(e)}(\mathbf{z}'_n - \mathbf{H}'E^{(e)}[\mathbf{x}'_n]) \\ &= \overline{\mathbf{x}}'_n - \mathbf{D}'\overline{\mathbf{x}}'_n + (\mathbf{C}_{\mathbf{x}'_n} + (1 - \sigma_{x_n}^2)\mathbf{D}') \\ &\quad \times \mathbf{F}_n'^{(e)}(\mathbf{z}'_n - \mathbf{H}'\overline{\mathbf{x}}'_n + \overline{x}_n\mathbf{H}'\mathbf{d}'). \end{aligned} \tag{20}$$

Now, keeping only the elements of $\widetilde{\mathbf{x}}_n'^{(e)}$ that are needed to feed the cancellation filter of the SIC, we have

$$\widetilde{\mathbf{x}}_n^{(e)} = \overline{\mathbf{x}}_n - \mathbf{D}\overline{\mathbf{x}}_n + (\mathbf{C}_{\mathbf{x}_n} + (1 - \sigma_{x_n}^2)\mathbf{D})\mathbf{F}_n^{(e)}(\mathbf{z}'_n - \mathbf{H}'\overline{\mathbf{x}}'_n + \overline{x}_n\mathbf{H}'\mathbf{d}'), \tag{21}$$

where

$$\mathbf{F}_n^{(e)} = \mathcal{C}(\mathbf{F}_n'^{(e)}, K+1, 2K+1+N) \tag{22}$$

denotes a matrix consisting of the "central" $K + 1 + N$ rows of $\mathbf{F}_n'^{(e)}$ (from row $K + 1$ to $2K + 1 + N$) and $\mathbf{D} = \mathbf{d}\mathbf{d}^T$. Substituting the above relation into (5), and taking into account that $\mathbf{q}^H\mathbf{d} = 0$, we finally get

$$\begin{aligned} s_n &= \mathbf{p}^H\mathbf{z}_n + \mathbf{q}^H\widetilde{\mathbf{x}}_n^{(e)} \\ &= \mathbf{p}^H\mathbf{z}_n + \mathbf{q}^H\overline{\mathbf{x}}_n + \mathbf{q}^H\mathbf{C}_{\mathbf{x}_n}\mathbf{F}_n^{(e)}(\mathbf{z}'_n - \mathbf{H}'\overline{\mathbf{x}}'_n + \overline{x}_n\mathbf{H}'\mathbf{d}'). \end{aligned} \tag{23}$$

From the above relation it is interesting to note that the suggested solution is, in fact, a soft interference canceller (consisting of the first two terms of (23)) plus a "correction" term to compensate for the fact that the cancellation filter of the SIC does not contain the correct symbols. Furthermore, for perfect a priori information ($\sigma_{x_n}^2 \to 0$), the third term of (23) vanishes and, in this case, the CE-SIC equalizer becomes equivalent to the exact linear MMSE equalizer of [11]. On

Input: $\mathbf{h}$, L_1, L_2, σ_w^2, $L^{(D)}(c_m)$, z_n, k, l	$n = 1,\dots,S/(Rq)$, $m = 1,\dots,S/R$
Output: $L_e^{(E)}(c_m)$	$m = 1,\dots,S/R$

Compute $\overline{x}_n$ and $\sigma_{x_n}^2$ from (13) $\quad n = 1,\dots,S/(Rq)$
$M = k+l+1$, $N = l+L_2$, $K = L_1+k$, $M' = K+M+N$
$\mathbf{H}' = \mathcal{T}\{\mathbf{h}^T, M', 2(K+N)+1\}$, $\mathbf{d}' = [\mathbf{0}_{1\times 2K} \quad 1 \quad \mathbf{0}_{1\times 2N}]^T$, $\mathbf{D}' = \mathbf{d}'\mathbf{d}'^T$
$\mathbf{H} = \mathcal{T}\{\mathbf{h}^T, M, K+N+1\}$, $\mathbf{d} = [\mathbf{0}_{1\times K} \quad 1 \quad \mathbf{0}_{1\times N}]^T$, $\mathbf{D} = \mathbf{d}\mathbf{d}^T$
$\mathbf{p} = \frac{1}{\sigma_w^2 + E_h}\mathbf{H}\mathbf{d}$ $(E_h = \mathbf{d}^T\mathbf{H}^H\mathbf{H}\mathbf{d})$
$\mathbf{q} = -\mathbf{H}^H\mathbf{p} + \mathbf{D}\mathbf{H}^H\mathbf{p}$
FOR $n = 1,\dots,S/(Rq)$
    $\mathbf{C}_{\mathbf{x}'_n} = \text{diag}([\sigma_{x_{n+2K}}^2 \cdots \sigma_{x_n}^2 \cdots \sigma_{x_{n-2N}}^2])$
    $\mathbf{C}_{\mathbf{x}'_n}^{(e)} = \mathbf{C}_{\mathbf{x}'_n} + (1-\sigma_{x_n}^2)\mathbf{D}'$
    $\mathbf{F}_n'^{(e)} = \mathbf{H}'^H(\mathbf{H}'\mathbf{C}_{\mathbf{x}'_n}^{(e)}\mathbf{H}'^H + \sigma_w^2\mathbf{I})^{-1}$
    $\mathbf{F}_n^{(e)} = \mathcal{C}(\mathbf{F}_n'^{(e)}, K+1, 2K+1+N)$
    $\mathbf{z}_n = [z_{n+k} \cdots z_n \cdots z_{n-l}]^T$, $\mathbf{z}'_n = [z_{n+k+K} \cdots z_n \cdots z_{n-l-N}]^T$
    $\overline{\mathbf{x}}_n = [\overline{x}_{n+K} \cdots \overline{x}_n \cdots \overline{x}_{n-N}]^T$, $\overline{\mathbf{x}}'_n = [\overline{x}_{n+2K} \cdots \overline{x}_n \cdots \overline{x}_{n-2N}]^T$
    $s_n = \mathbf{p}^H\mathbf{z}_n + \mathbf{q}^H\overline{\mathbf{x}}_n + \mathbf{q}^H\mathbf{C}_{\mathbf{x}_n}\mathbf{F}_n^{(e)}(\mathbf{z}'_n - \mathbf{H}'\overline{\mathbf{x}}'_n + \overline{x}_n\mathbf{H}'\mathbf{d}')$
    Compute $\mu_{i,n}$ and $\sigma_{i,n}^2$ from (24)
    FOR $j = 1,\dots,q$
        Compute $L_e^{(E)}(c_{(n-1)q+j})$ from (26)
    END
END

Algorithm 1: The CE-SIC equalizer.

the other hand, when a priori information is null, the linear MMSE equalizer of [11] reduces to a conventional linear equalizer and the CE-SIC reduces to an MMSE SIC whose cancellation filter is fed by the output of a conventional linear equalizer.

In order to transform the output of the CE-SIC into log-likelihood ratios, the mean and variance of s_n, given that a particular symbol α_i has been transmitted, must be computed. For these statistics, we get

$$\begin{aligned}\mu_{i,n} &= E[s_n \mid x_n = \alpha_i] = (\mathbf{p}^H\mathbf{H}\mathbf{d} + \mathbf{q}^H\mathbf{C}_{\mathbf{x}_n}\mathbf{F}_n^{(e)}\mathbf{H}'\mathbf{d}')\cdot\alpha_i,\\ \sigma_{i,n}^2 &= \mathbf{p}^H[\mathbf{H}(\mathbf{C}_{\mathbf{x}_n} - \sigma_{x_n}^2\mathbf{D})\mathbf{H}^H + \sigma_w^2\mathbf{I}_M]\mathbf{p}\\ &\quad + 2\,\text{Real}\,\{\mathbf{p}^H[\mathbf{H}(\mathbf{C}_{\mathbf{x}_n,\mathbf{x}'_n} - \sigma_{x_n}^2\mathbf{d}\mathbf{d}'^T)\mathbf{H}'^H\\ &\qquad + \mathbf{W}]\mathbf{F}_n^{(e)H}\mathbf{C}_{\mathbf{x}_n}^H\mathbf{q}\}\\ &\quad + \mathbf{q}^H\mathbf{C}_{\mathbf{x}_n}\mathbf{F}_n^{(e)}[\mathbf{H}'(\mathbf{C}_{\mathbf{x}'_n} - \sigma_{x_n}^2\mathbf{D}')\mathbf{H}'^H\\ &\qquad + \sigma_w^2\mathbf{I}_{M'}]\mathbf{F}_n^{(e)H}\mathbf{C}_{\mathbf{x}_n}^H\mathbf{q},\end{aligned} \tag{24}$$

where

$$\mathbf{W} = [\mathbf{0}_{M\times K} \quad \sigma_w^2\mathbf{I}_M \quad \mathbf{0}_{M\times N}] \tag{25}$$

and $\mathbf{C}_{\mathbf{x}_n,\mathbf{x}'_n}$ is the covariance matrix between $\mathbf{x}_n$ and $\mathbf{x}'_n$.

The computational complexity of this algorithm is $O(M'^3)$ since the most demanding operation is the matrix inversion involved in the computation of matrix $\mathbf{F}_n'^{(e)}$. A time recursive algorithm similar to the one developed in [6] can reduce this to $O(M'^2)$ by exploiting structural similarities between subsequent matrices. Moreover, in Section 4 the CE-SIC algorithm is used as a starting point to derive an $O(M)$ complexity algorithm.

3.3. Demapper

The required soft information for the output bits of the SIC, is computed as

$$\begin{aligned}L_e^{(E)}(c_m) &= L_e^{(E)}(c_{(n-1)\cdot q+j}) = \ln\left(\frac{\Pr\{c_{(n-1)\cdot q+j} = 1 \mid s_n\}}{\Pr\{c_{(n-1)\cdot q+j} = 0 \mid s_n\}}\right)\\ &= \ln\left(\frac{\sum_{\beta_{i,j}=1}\Pr\{x_n = a_i \mid s_n\}}{\sum_{\beta_{i,j}=0}\Pr\{x_n = a_i \mid s_n\}}\right)\\ &= \ln\left(\frac{\sum_{\beta_{i,j}=1}\Pr\{x_n = a_i\}p(s_n \mid x_n = a_i)/p(s_n)}{\sum_{\beta_{i,j}=0}\Pr\{x_n = a_i\}p(s_n \mid x_n = a_i)/p(s_n)}\right),\end{aligned} \tag{26}$$

where the term $p(s_n)$ can be eliminated from nominator and denominator. Note that when computing $\Pr\{x_n = a_i\}$ in the nominator and denominator we must set the probability of bit j equal to unity. Also, $p(s_n \mid x_n = a_i) = \mathcal{N}(\mu_{i,n}, \sigma_{i,n}^2)|_{s_n}$, with $\mu_{i,n}$ and $\sigma_{i,n}^2$ given from (24). The CE-SIC equalizer, as described in this section, is summarized in Algorithm 1.

4. APPROXIMATE IMPLEMENTATION

Although the CE-SIC developed in the previous section is less computationally demanding than the MAP equalization algorithm, it is still difficult to be implemented in a real-time system. Thus in this section we develop an approximate implementation of the CE-SIC equalizer, the so-called ACE-SIC, by modifying the unit that computes the conditional expectation of the transmitted symbols. Our design goal is to find an approximation that is well suited for low a priori information. The soft interference canceller that combines symbol estimates is left unchanged. The overall approximate equalizer will thus consist of a device optimized for low a priori information, and the SIC which is optimal for perfect a priori probabilities. Because these units cooperate, we expect that the overall scheme will have good performance for quite general a priori information.

In order to reduce complexity, we approximate matrix

$$\mathbf{F}_n'^{(e)} = \mathbf{H}'^H\left(\mathbf{H}'\mathbf{C}_{\mathbf{x}_n'}^{(e)}\mathbf{H}'^H + \sigma_w^2\mathbf{I}\right)^{-1} \tag{27}$$

by the matrix

$$\widehat{\mathbf{F}}'^{(e)} = \mathbf{H}'^H\left(\mathbf{H}'\mathbf{H}'^H + \sigma_w^2\mathbf{I}\right)^{-1} \tag{28}$$

assuming that $\mathbf{C}_{\mathbf{x}_n'}^{(e)} \to \mathbf{I}_{2K+1+2N}$, which is true when no a priori information is available. It should be noted that such an approximation has also been used in [7]. Furthermore, if we inspect the rows of matrix $\widehat{\mathbf{F}}'^{(e)}$, we can easily verify that each one corresponds to an MMSE linear equalizer designed for a corresponding output delay. In the previous section, we used only the $K+N+1$ "central" rows of matrix $\mathbf{F}_n'^{(e)}$. These rows correspond to linear equalizers estimating the symbols fed to the cancellation filter of the SIC. Each of these equalizer filters is designed to process a window of at least l past samples and k future samples of $\{z\}$, relatively to the corresponding output symbol. If the equalizer length $M = k + l + 1$ is sufficiently long, then any further increase of its length does not affect the existing taps while all new taps are equal to zero. In this case, the associated matrix $\widehat{\mathbf{F}}$ is given by

$$\widehat{\mathbf{F}} = \mathcal{T}\left\{\mathbf{d}^T\mathbf{H}^H\left(\mathbf{H}\mathbf{H}^H + \sigma_w^2\mathbf{I}_M\right)^{-1},\ K+1+N, K+M+N\right\}, \tag{29}$$

where the function $\mathcal{T}(\mathbf{v}, L, C)$ was defined earlier. Matrix $\widehat{\mathbf{F}}$ is an approximation of matrix $\mathbf{F}_n^{(e)}$, where the last matrix is defined by (22). This approximation is valid when the linear equalizer filter length is adequately large, so that two linear equalizers of equal length $K+M+N > M$, designed to provide estimates of symbols x_n and x_{n-i}, respectively, have equal taps but shifted by i places.

The above-suggested approximation of matrix $\mathbf{F}_n^{(e)}$ by $\widehat{\mathbf{F}}$ is expected to affect the performance of the ACE-SIC algorithm compared to the performance of its exact counterpart, the CE-SIC. In particular, the third term of (23) becomes suboptimal and thus, the past and future symbol estimates in the cancellation filter are not equal to their MMSE estimates. As a remedy to this performance degradation, we allow the (past and future) symbol estimates contained in the cancellation filter to depend on the a priori information about the current symbol x_n. Using the a priori information about x_n improves the computed past and future symbol estimates. On the other hand, as these estimates are subsequently combined for the computation of the output of the ACE-SIC, it turns out that the extrinsic information restriction has been relaxed. However, the output of the ACE-SIC depends only implicitly on the a priori information about x_n via the past and future estimates that use this information. This modification yields the following filtering equation:

$$\hat{s}_n = \mathbf{p}^H\mathbf{z}_n + \mathbf{q}^H\overline{\mathbf{x}}_n + \mathbf{q}^H\mathbf{C}_{\mathbf{x}_n}\widehat{\mathbf{F}}\left(\mathbf{z}_n' - \mathbf{H}'\overline{\mathbf{x}}_n'\right) \tag{30}$$

in which the vector multiplying $\widehat{\mathbf{F}}$ does not include the term $\overline{x}_n\mathbf{H}'\mathbf{d}'$, as opposed to (23). Simulation results verified that this modification leads to noticeable performance improvement.

Similarly to the CE-SIC, in order to transform the output of the algorithm into log-likelihood ratios the mean and variance of the output $\hat{s}_n$ must be estimated. For complexity reasons we assume that the required mean and variance remain fixed during each iteration, that is, they are computed once prior to each iteration. This can be achieved by keeping all symbol variances equal to σ^2 assuming all symbols to be equally reliable. It is interesting to note that a similar approximation has also been used in [7]. By using relations (24), which comply with the extrinsic information restriction, we get

$$\begin{aligned}\hat{\mu}_i &= \left(\mathbf{p}^H\mathbf{H}\mathbf{d} + \sigma^2\mathbf{q}^H\widehat{\mathbf{F}}\mathbf{H}'\mathbf{d}'\right)\cdot\alpha_i,\\ \hat{\sigma}_i^2 &= \mathbf{p}^H\left[\mathbf{H}\left(\sigma^2\mathbf{I}_{K+1+N} - \sigma^2\mathbf{D}\right)\mathbf{H}^H + \sigma_w^2\mathbf{I}_M\right]\mathbf{p}\\ &\quad + 2\sigma^2\ \mathrm{Real}\left\{\mathbf{p}^H\left[\mathbf{H}\left(\widehat{\mathbf{C}}_{\mathbf{x},\mathbf{x}'} - \sigma^2\mathbf{d}\mathbf{d}'^T\right)\mathbf{H}'^H + \mathbf{W}\right]\widehat{\mathbf{F}}^H\mathbf{q}\right\}\\ &\quad + \sigma^4\mathbf{q}^H\widehat{\mathbf{F}}\left[\mathbf{H}'\left(\sigma^2\mathbf{I}_{2K+1+2N} - \sigma^2\mathbf{D}'\right)\mathbf{H}'^H + \sigma_w^2\mathbf{I}_{M'}\right]\widehat{\mathbf{F}}^H\mathbf{q},\end{aligned} \tag{31}$$

where

$$\widehat{\mathbf{C}}_{\mathbf{x},\mathbf{x}'} = \left[\mathbf{0}_{K+N+1\times K}\quad \sigma^2\mathbf{I}_{K+N+1}\quad \mathbf{0}_{K+N+1\times N}\right]. \tag{32}$$

Concerning now the required parameter σ^2, in contrast to [7] where a time average over all $\sigma_{x_n}^2$ was used, here we suggest using

$$\sigma^2 = \max\left\{\sigma_{x_1}^2, \sigma_{x_2}^2, \ldots, \sigma_{x_{S/(Rq)}}^2\right\}. \tag{33}$$

This approximation is valid whenever all symbol variances are equal. We use the maximum symbol variance (i.e., the variance of the least reliable symbol), in place of the time average previously proposed, in order to assure that none of the symbols is treated as more reliable than it actually is, during the demapping operation.

At this point it is interesting to note that the ACE-SIC equalizer has a very attractive feature compared to other approaches. Apart from the Toeplitz-matrix approximation of (29), the ACE-SIC is identical to its exact counterpart (CE-SIC) both for perfect a priori information (i.e., $\sigma_{x_n}^2 \to 0$) and for no a priori information (i.e., $\sigma_{x_n}^2 \to 1$). Note that similar approximations in [6] resulted in equalizers satisfying only one of these two conditions, leading the authors to propose suitable decision criteria for selecting one out of two approximate algorithms (one designed for $\sigma_{x_n}^2 \to 0$, and the other for $\sigma_{x_n}^2 \to 1$) prior to each iteration [6, 7]. The performance of the ACE-SIC demonstrated in Section 6 justifies to some extend the approximations suggested above. Algorithm 2 summarizes the ACE-SIC equalizer.

Input: $\mathbf{h}$, L_1, L_2, σ_w^2, $L^{(D)}(c_m)$, z_n, k, l $\quad n = 1, \dots, S/(Rq)$, $m = 1, \dots, S/R$
Output: $L_e^{(E)}(c_m)$ $\quad m = 1, \dots, S/R$

Compute $\bar{x}_n$ and $\sigma_{x_n}^2$ from (13) $\quad n = 1, \dots, S/(Rq)$
$M = k + l + 1$, $N = l + L_2$, $K = L_1 + k$
For the first Iteration
$\quad \mathbf{H} = \mathcal{T}\{\mathbf{h}^T, M, K + N + 1\}$, $\mathbf{d} = [\mathbf{0}_{1\times K} \;\; 1 \;\; \mathbf{0}_{1\times N}]^T$, $\mathbf{D} = \mathbf{d}\mathbf{d}^T$
$\quad \mathbf{f} = (\mathbf{H}\mathbf{H}^H + \sigma_w^2 \mathbf{I}_M)^{-1}\mathbf{H}\mathbf{d}$
$\quad \mathbf{p} = \dfrac{1}{\sigma_w^2 + E_h}\mathbf{H}\mathbf{d}$ $(E_h = \mathbf{d}^T\mathbf{H}^H\mathbf{H}\mathbf{d})$
$\quad \mathbf{q} = -\mathbf{H}^H\mathbf{p} + \mathbf{D}\mathbf{H}^H\mathbf{p}$
$\quad$ Compute and store all terms of (31) using $\sigma^2 = 1$
FOR $n = 1, \dots, S/(Rq)$
$\quad \mathbf{z}_n = [z_{n+k} \cdots z_n \cdots z_{n-l}]^T$
$\quad \bar{\mathbf{x}}_n = [\bar{x}_{n+K} \cdots \bar{x}_n \cdots \bar{x}_{n-N}]^T$
$\quad \tilde{x}_n = \bar{x}_n + \sigma_{x_n}^2 \mathbf{f}^H(\mathbf{z}_n - \mathbf{H}\bar{\mathbf{x}}_n)$
END
$\sigma^2 = \max\{\sigma_{x_1}^2, \sigma_{x_2}^2, \dots, \sigma_{x_{S/(Rq)}}^2\}$
Compute $\hat{\mu}_i$ and $\hat{\sigma}_i^2$ from (31) using the stored terms
FOR $n = 1, \dots, S/(Rq)$
$\quad \mathbf{z}_n = [z_{n+k} \cdots z_n \cdots z_{n-l}]^T$
$\quad \tilde{\mathbf{x}}_n = [\tilde{x}_{n+K} \cdots \tilde{x}_n \cdots \tilde{x}_{n-N}]^T$
$\quad \hat{s}_n = \mathbf{p}^H\mathbf{z}_n + \mathbf{q}^H\tilde{\mathbf{x}}_n$
$\quad$ FOR $j = 1, \dots, q$
$\quad\quad$ Compute $L_e^{(E)}(c_{(n-1)q+j})$ from (26) using $\hat{\mu}_i$ and $\hat{\sigma}_i^2$
$\quad$ END
END

Algorithm 2: The ACE-SIC equalizer.

For the sake of simplicity, the algorithm demonstrated in Algorithm 2 appears to have two distinct filtering loops. The first one computes the input to the cancellation filter of the SIC while the second loop uses these estimates to cancel the interference. It should be noted however that these two loops could be combined to one: after the initialization phase where the first loop computes K estimates $\tilde{x}_1, \dots, \tilde{x}_K$ (of future symbols needed for cancellation of the interference), the two loops can run simultaneously, that is, at each time instant n an estimate of $\tilde{x}_{n+K}$ is first computed and then used for cancellation. This remark could be useful in applications with output delay restrictions.

5. COMPLEXITY ISSUES

Over the past decade, after turbo equalization was first proposed in [1], several attempts have been made towards reducing the computational complexity of the equalization algorithm involved in such a receiver architecture. As we have already mentioned, equalizers based on linear filters offer considerable complexity reduction as compared to equalizers based on trellis-diagrams. The various SISO equalizers that are based on linear filters can be classified into the following three categories in terms of their computational complexity.

(1) *Time-varying-filter algorithms.* This category includes algorithms whose filters are being updated each time a new output symbol is computed. Usually, a matrix inversion has to be computed, requiring a complexity order of $O(M^3)$ which can be reduced to $O(M^2)$ by using a time-recursive algorithm which exploits structural similarities between subsequent matrices. Typical examples of algorithms falling into this category are the MMSE exact algorithm of [11], the CE-SIC developed here and the algorithms presented in [8, 9].

(2) *Reoptimized prior every iteration.* This category includes algorithms whose filters are being updated prior to each iteration, but are kept fixed during the subsequent processing of the current data burst. This optimization involves only one matrix inversion before each iteration, and the required complexity is of order $O(M^2)$ when the involved channel convolution matrix has a Toeplitz structure. Typical examples of such algorithms are the approximate MMSE LE (I) developed in [11] and the equivalent IC LE developed in [10]. In the latter approach the complexity is reduced to $O(M\log_2(M))$ by approximating a Toeplitz matrix by a circulant matrix.

(3) *Optimized only once.* This category includes algorithms whose filters are optimized only at the first iteration, and turn out to be equal to the conventional MMSE equalizers, operating without using a priori probabilities. Examples of algorithms of this category are the APPLE, Matched Filtering (Soft Interference Canceller) and SWITCHED equalizers of [7] and the ACE-SIC developed here.

Table 1: Complexity order comparison of various SISO equalizers for the initialization phase (e.g., prior to filtering).

	Filter Computation		Statistics Computation	
Approach	First Iteration	Next Iterations	First Iteration	Next Iterations
CE-SIC	$O(M'^2)$	$O(M'^3)^\dagger$	$O(M'^2)$	—
MMSE LE [11]	$O(M^2)$	$O(M^3)^\dagger$	$O(M)$	—
IC-LE [10]	$O(M\log_2(M))$	$O(M\log_2(M))$	$O(M^2)$	$O(M^2)$
MMSE-LE (I) [11]	$O(M^2)$	$O(M^2)$	$O(M^2)$	$O(M^2)$
SWITCHED [7]	$O(M^2)$	—	$O(M^2)$	$O(1)$
ACE-SIC	$O(M^2)$	—	$O(M^2)$	$O(1)$

† We assume no "bootstrap" procedure as described in [11].

Equalizers of the third category, beyond their significant complexity savings, can be easily modified to derive adaptive counterparts that still have linear complexity. Since the filters of those equalizers are set equal to their conventional MMSE counterparts (computed without using a priori probabilities), a decision directed approach utilizing tentative decisions can be used in the update recursion. For example, in [3] the LMS algorithm was used to update the filters of a canceller whose initial estimates where obtained using a training sequence. Table 1 summarizes the complexity orders for the initialization phase of various SISO equalizers. It should be stressed that comparison of complexities is meaningful if the systems under comparison perform the same number of iterations.

6. SIMULATION RESULTS

To test the performance of the proposed equalizers we performed some typical experiments. Information bits were generated in bursts of $S = 6144$ bits. Then an R.S.C. code with generator matrix $G(D) = [1((1+D^2)/(1+D+D^2))]$ of rate $R = 1/2$ was applied, and the resulting bits were interleaved using a $\mathcal{S}$-random interleaver ($\mathcal{S} = 23$) [17]. The interleaved bits were mapped to an 8-PSK ($q = 3$) symbol alphabet using Gray code mapping. The 4096 symbols per burst were transmitted over a channel whose impulse response was set either $h_{-1} = 0.407$, $h_0 = 0.815$, $h_1 = 0.407$ (channel B of [18]) or $h_{-2} = 0.227$, $h_{-1} = 0.46$, $h_0 = 0.688$, $h_1 = 0.46$, $h_2 = 0.227$ (channel C of [18]). Figures 3 and 4 demonstrate the performance of various receivers performing turbo equalization for the aforementioned channels. The cases shown correspond to (a) conventional equalization and decoding executed once, and (b), (c), and (d) to 1, 2, and 8 turbo iterations, respectively. For all simulations, the filter lengths were computed using $k = l = 10$. The SNR of the system used in the simulations is defined as

$$\frac{E_b}{N_0} = \frac{E_s}{q \cdot R \cdot N_0}, \tag{34}$$

where E_s denotes the average energy per transmitted symbol and $N_0 = \sigma_w^2$.

From Figure 3, we notice that all equalizers exhibit similar performance. The MMSE equalizer of [6] has superior performance followed by the CE-SIC, the SWITCHED equalizer of [7] and the ACE-SIC. The performance of all algorithms is almost the same after eight iterations, and all algorithms have reached the performance bound that corresponds to the AWGN channel. Using a high complexity algorithm for the channel B does not seem very practical since the same performance can be obtained by the low complexity solutions.

On Figure 4, we notice that the ISI caused by the channel is quite severe so that none of the examined algorithms reaches the performance bound after eight iterations. The MMSE equalizer of [6], and the CE-SIC have almost the same performance. The MMSE I equalizer, proposed in [11] as a low cost alternative to the exact algorithm, offers better performance than the ACE-SIC equalizer but at a higher computational complexity, since its filters are reoptimized before every turbo iteration. It is interesting to note that the ACE-SIC equalizer exhibits better performance than the SWITCHED equalizer of [7] (approximately 1 dB less SNR is needed to achieve a BER of 10^{-3}). Therefore, for hostile channels, switching between equalizers optimized for the two extreme cases (no a priori and perfect a priori information) is a less efficient technique than using an algorithm that can smoothly adapt to the quality of the a priori information (such as the ACE-SIC). Also, the ACE-SIC equalizer, at medium SNRs, achieves a performance close to the performance of its exact counterpart.

7. CONCLUSION

In this work, a novel SISO equalizer of linear complexity was presented. This algorithm was derived as an approximate implementation (ACE-SIC) of a new two-step minimization algorithm (CE-SIC) which in turn was developed for the problem of equalization using a priori probabilities. Simulation results indicated that (a) the exact implementation has almost identical performance to the MMSE equalizer of [11], and (b) the approximate implementation offers very good performance at linear complexity. Thus, the latter low complexity equalization algorithm is suitable for high data-rate wireless communication systems with limited processing power.

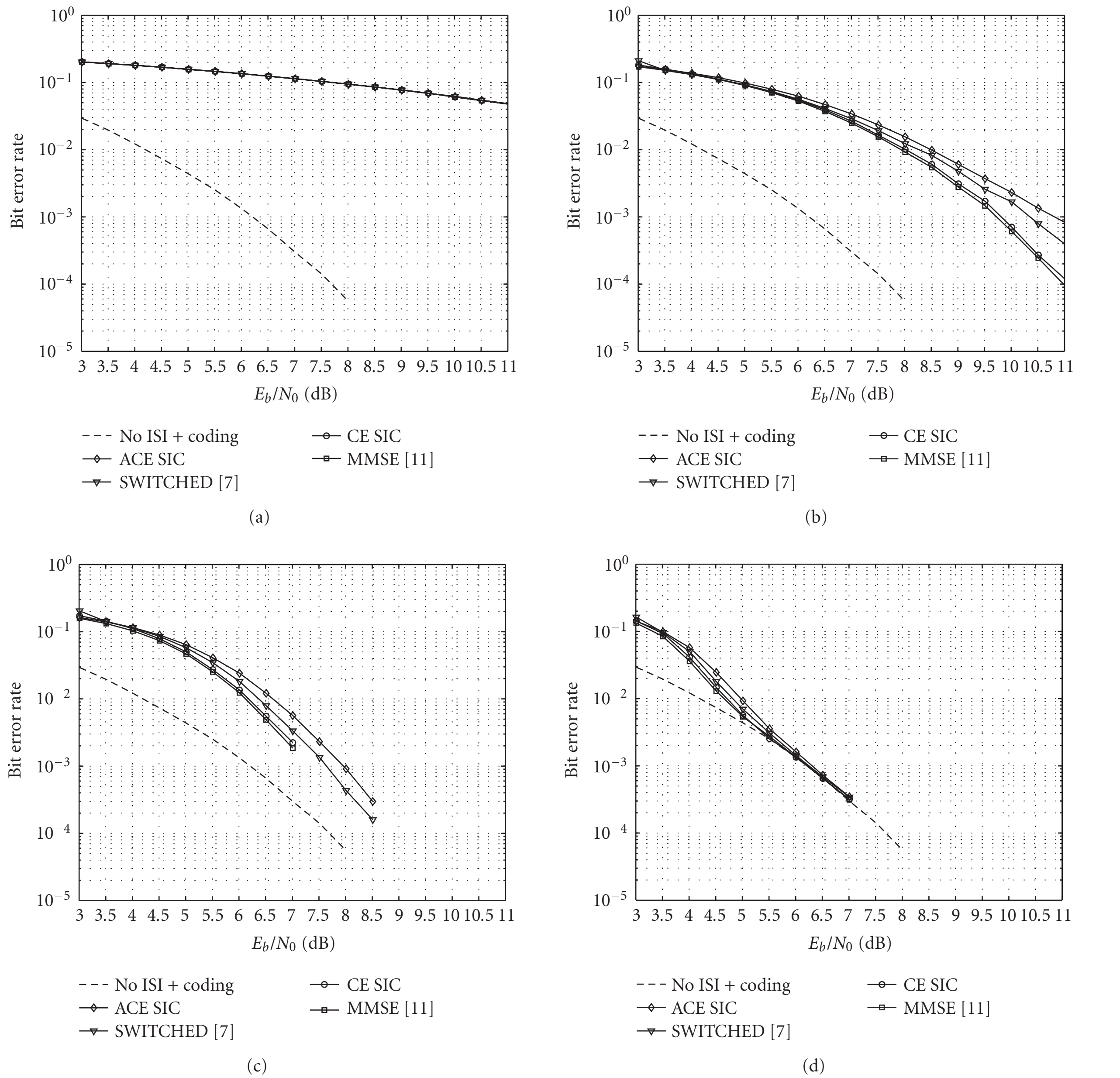

FIGURE 3: Bit error rate performance of various iterative receivers for channel B. (a) Equalization and decoding executed once, (b) one turbo iteration, (c) two turbo iterations, and (d) 8 turbo iterations.

APPENDIX

DERIVATION OF (6)

Let us define filter $\mathbf{p}$ of the soft interference canceller as in (3) and partition filter $\mathbf{q}$ as

$$\mathbf{q} = \left[q_{-K} \cdots q_{-1} \;\; 0 \;\; q_1 \cdots q_N\right]^T = \left[\mathbf{q}_{(f)}^T \;\; 0 \;\; \mathbf{q}_{(p)}^T\right]. \tag{A.1}$$

Let us now define a vector θ containing the coefficients of the above filters as $\theta = [\mathbf{p}^T \;\; \mathbf{q}_{(f)}^T \;\; \mathbf{q}_{(p)}^T]^T$ and the vector

$$\mathbf{u}_n = \left[z_{n+k} \cdots z_n \cdots \;\; z_{n-l} \;\; x_{n+K} \;\cdots\; x_{n+1} \;\; x_{n-1} \cdots x_{n-N}\right]^T \tag{A.2}$$

so that the output of the canceller at time index n is given by $s_n = \theta^H \mathbf{u}_n$. The vector θ_o that minimizes the mean squared error $E[|s_n - x_n|^2]$ will then satisfy

$$\mathbf{R}\theta_o = \mathbf{r}, \tag{A.3}$$

where $\mathbf{R} = E[\mathbf{u}_n \mathbf{u}_n^H]$ and $\mathbf{r} = E[x_n \mathbf{u}_n]$. We can now split matrix $\mathbf{R}$ into 9 submatrices as follows:

$$\mathbf{R} = E\left[\mathbf{u}_n \mathbf{u}_n^H\right] = \begin{bmatrix} \mathbf{R}_{zz} & \mathbf{R}_{zx}^{(f)} & \mathbf{R}_{zx}^{(p)} \\ \mathbf{R}_{xz}^{(f)} & \mathbf{R}_{xx}^{(f)} & \mathbf{R}_{xx}^{(f,p)} \\ \mathbf{R}_{xz}^{(p)} & \mathbf{R}_{xx}^{(p,f)} & \mathbf{R}_{xx}^{(p)} \end{bmatrix}, \tag{A.4}$$

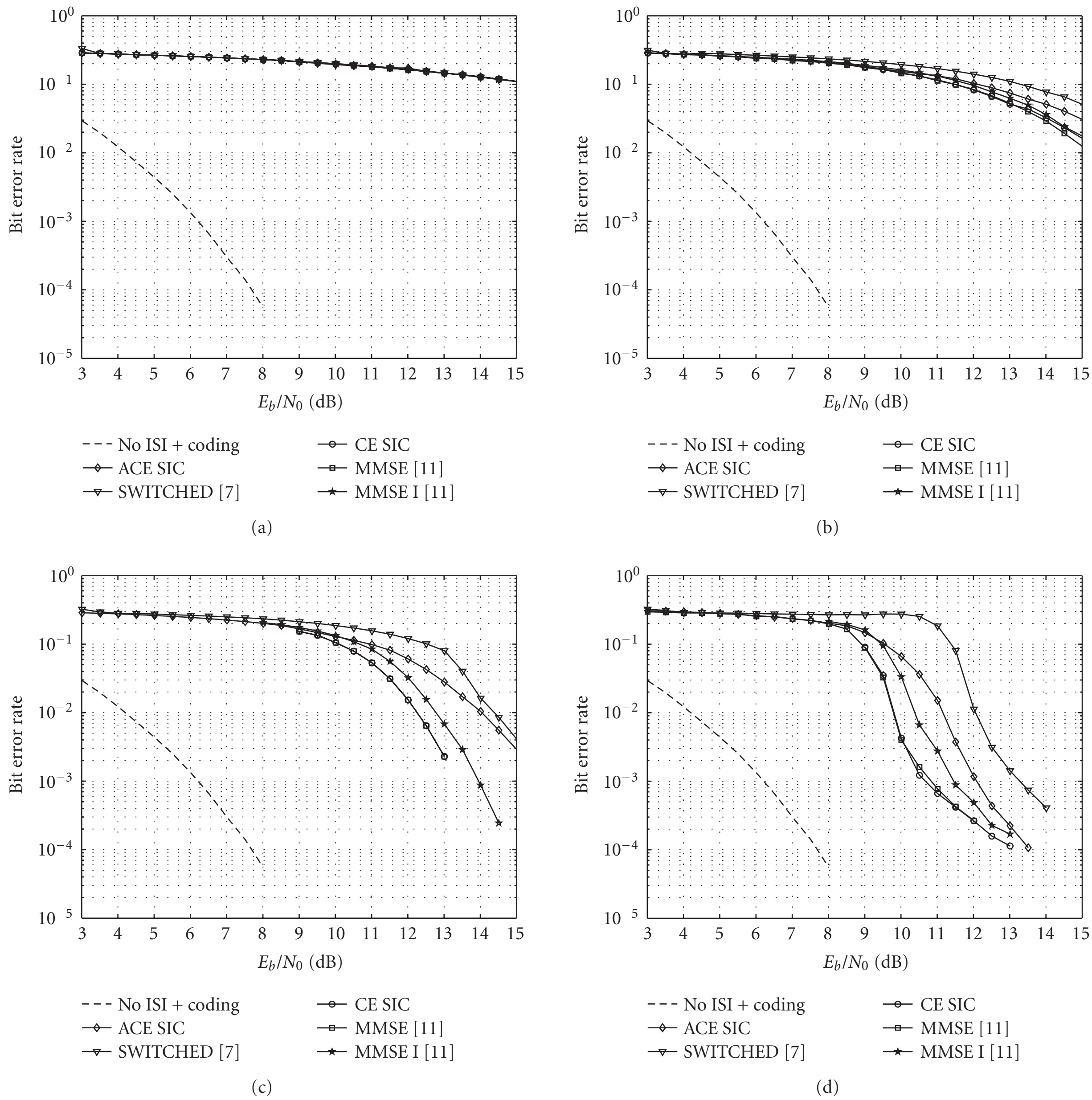

Figure 4: Bit error rate performance of various iterative receivers for channel C. (a) Equalization and decoding executed once, (b) one turbo iteration, (c) two turbo iterations, and (d) 8 turbo iterations.

where by using the fact that the symbols x_n are uncorrelated we have that

$$\begin{bmatrix} \mathbf{R}_{xx}^{(f)} & \mathbf{R}_{xx}^{(f,p)} \\ \mathbf{R}_{xx}^{(p,f)} & \mathbf{R}_{xx}^{(p)} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_K & \mathbf{0}_{K\times N} \\ \mathbf{0}_{N\times K} & \mathbf{I}_N \end{bmatrix}, \qquad \mathbf{r} = \begin{bmatrix} \mathbf{r}_{xz} \\ \mathbf{0}_{K\times 1} \\ \mathbf{0}_{N\times 1} \end{bmatrix}. \tag{A.5}$$

Now, it is possible to express the statistical quantities in the above expressions in terms of the channel impulse response. If we define as $\mathbf{H}$ the $M \times (K + 1 + N)$ channel convolution matrix, as $\mathbf{H}_A$ the matrix consisting of the first K columns of $\mathbf{H}$, and as $\mathbf{H}_B$ the matrix consisting of the last N columns of $\mathbf{H}$, it is easy to verify that

$$\begin{gathered} \mathbf{R}_{zz} = \mathbf{H}\mathbf{H}^H + \sigma_w^2\mathbf{I}_M, \\ \mathbf{R}_{zx}^{(f)} = \mathbf{H}_A, \qquad \mathbf{R}_{xz}^{(f)} = \mathbf{H}_A^H, \\ \mathbf{R}_{zx}^{(p)} = \mathbf{H}_B, \qquad \mathbf{R}_{xz}^{(p)} = \mathbf{H}_B^H, \\ \mathbf{r}_{xz} = \mathbf{H}\mathbf{d}, \quad \mathbf{d} = \begin{bmatrix} \mathbf{0}_{1\times k+L_1} & 1 & \mathbf{0}_{1\times l+L_2} \end{bmatrix}^T, \end{gathered} \tag{A.6}$$

where we have assumed $N = l + L_2$ and $K = L_1 + k$. Using the above expressions, we can split the initial linear system into three linear subsystems and express the filters of the MMSE canceller as

$$\mathbf{p}_o = \left(\mathbf{H}\mathbf{H}^H + \sigma_w^2\mathbf{I}_M - \mathbf{H}_B\mathbf{H}_B^H - \mathbf{H}_A\mathbf{H}_A^H\right)^{-1}\mathbf{H}\mathbf{d},$$
$$\mathbf{q}_{o,(f)} = -\mathbf{H}_A^H\mathbf{p}_o, \tag{A.7}$$
$$\mathbf{q}_{o,(p)} = -\mathbf{H}_B^H\mathbf{p}_o.$$

Substituting $\mathbf{H} = [\mathbf{H}_A \;\; \mathbf{H}\mathbf{d} \;\; \mathbf{H}_B]$ into (A.7) and using the matrix inversion lemma, we finally get

$$\mathbf{p}_o = \frac{1}{\sigma_w^2 + E_h}\mathbf{H}\mathbf{d}. \tag{A.8}$$

Finally, vector $\mathbf{q}_o$ can be expressed more compactly as

$$\mathbf{q}_o = \begin{bmatrix} \mathbf{q}_{o,(f)} \\ 0 \\ \mathbf{q}_{o,(p)} \end{bmatrix} = \begin{bmatrix} -\mathbf{H}_A^H\mathbf{p}_o \\ 0 \\ -\mathbf{H}_B^H\mathbf{p}_o \end{bmatrix} = -\mathbf{H}^H\mathbf{p}_o + \mathbf{d}\mathbf{d}^T\mathbf{H}^H\mathbf{p}_o, \tag{A.9}$$

where the last term is used to make the central coefficient of $\mathbf{q}_o$ equal to zero. Equations (A.8) and (A.9) conclude the proof.

Note at this point that the above equations are equivalent to equation (11) of [6] after proper manipulations. To verify this, let us first reproduce equation (11) of [6]:

$$\hat{x}_n = \mathbf{c}_{\text{MF}}^H\left(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s}\right), \tag{A.10}$$

where $\mathbf{c}_{\text{MF}}$ is the involved matched filter and $\mathbf{s} = \mathbf{H}\mathbf{d}$. Then,

$$\begin{aligned}\hat{x}_n &= \mathbf{c}_{\text{MF}}^H\mathbf{z}_n - \mathbf{c}_{\text{MF}}^H\mathbf{H}\overline{\mathbf{x}}_n + \mathbf{c}_{\text{MF}}^H\mathbf{H}\mathbf{d}\overline{x}_n \\ &= \underbrace{\mathbf{c}_{\text{MF}}^H}_{\mathbf{p}_o^H}\mathbf{z}_n + \underbrace{\left(-\mathbf{c}_{\text{MF}}^H\mathbf{H} + \mathbf{c}_{\text{MF}}^H\mathbf{H}\mathbf{d}\mathbf{d}^T\right)}_{\mathbf{q}_o^H}\overline{\mathbf{x}}_n.\end{aligned} \tag{A.11}$$

Therefore the involved filters are equivalent, since $\mathbf{c}_{MF}$ defined in [6] is equal to filter $\mathbf{p}_0$. Thus, it follows that for perfect a priori information the CE-SIC equalizer becomes equivalent to the MMSE linear equalizer of [6, 11].

REFERENCES

[1] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: turbo-equalization," *European Transactions on Telecommunications and Related Technologies*, vol. 6, no. 5, pp. 507–511, 1995.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit errorcorrecting coding and decoding: turbo codes," in *Proceedings of the IEEE International Conference on Communications (ICC '93)*, pp. 1064–1070, Geneva, Switzerland, May 1993.

[3] C. Laot, A. Glavieux, and J. Labat, "Turbo equalization: adaptive equalization and channel decoding jointly optimized," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 9, pp. 1744–1752, 2001.

[4] I. Fijalkow, A. Roumy, S. Ronger, D. Pirez, and P. Vila, "Improved interference cancellation for turbo-equalization," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '00)*, vol. 1, pp. 416–419, Istanbul, Turkey, June 2000.

[5] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 47, no. 7, pp. 1046–1061, 1999.

[6] M. Tüchler, R. Koetter, and A. C. Singer, "Turbo equalization: principles and new results," *IEEE Transactions on Communications*, vol. 50, no. 5, pp. 754–767, 2002.

[7] M. Tüchler and J. Hagenauer, "Linear time and frequency domain turbo equalization," in *Proceedings of IEEE 53rd Vehicular Technology Conference (VTC '01)*, vol. 2, pp. 1449–1453, Rhodes, Greece, May 2001.

[8] L. Liu and L. Ping, "An extending window MMSE turbo equalization algorithm," *IEEE Signal Processing Letters*, vol. 11, no. 11, pp. 891–894, 2004.

[9] S. Jiang, L. Ping, H. Sun, and C. S. Leung, "Modified LMMSE turbo equalization," *IEEE Communications Letters*, vol. 8, no. 3, pp. 174–176, 2004.

[10] C. Laot, R. Le Bidan, and D. Leroux, "Low-complexity MMSE turbo equalization: a possible solution for EDGE," *IEEE Transactions on Wireless Communications*, vol. 4, no. 3, pp. 965–974, 2005.

[11] M. Tüchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Transactions on Signal Processing*, vol. 50, no. 3, pp. 673–683, 2002.

[12] D. Ampeliotis and K. Berberidis, "A new turbo equalizer with linear complexity," in *Proceedings of 13th European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.

[13] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proceedings of 2nd European Personal Mobile Communications Conference (EPMCC '97)*, pp. 307–312, Bonn, German, September-October 1997.

[14] F. Vogelbruch and S. Haar, "Improved soft ISI cancellation for turbo equalization using full soft output channel decoder's information," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '03)*, vol. 3, pp. 1736–1740, San Francisco, Calif, USA, December 2003.

[15] M. Witzke, S. Bäro, F. Schreckenbach, and J. Hagenauer, "Iterative detection of MIMO signals with linear detectors," in *Proceedings of 36th Asilomar Conference on Signals Systems and Computers*, vol. 1, pp. 289–293, Pacific Groove, Calif, USA, November 2002.

[16] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice-Hall, Upper Saddle River, NJ, USA, 1993.

[17] C. Heegard and S. Wicker, *Turbo Coding*, Kluwer Academic, Boston, Mass, USA, 1999.

[18] J. Proakis, *Digital Communications*, McGraw-Hill, New York, NY, USA, 3rd edition, 1995.

Dimitris Ampeliotis was born in Athens, Greece, in 1979. He received the Diploma degree in computer engineering and informatics, and the Masters degree in signal processing from the University of Patras, Greece, in 2002 and 2004, respectively. He is currently pursuing the Ph.D. degree in signal processing and communications at the University of Patras. His research interests lie in the area of signal processing for communications. He is a Member of the Technical Chamber of Greece.

Kostas Berberidis received the Diploma degree in electrical engineering from DUTH, Greece, in 1985, and the Ph.D. degree in signal processing and communications from the University of Patras, Greece, in 1990. From 1986 to 1990, he was a Research Assistant at the Research Academic Computer Technology Institute (RACTI), Patras, Greece, and a Teaching Assistant at the Computer Engineering and Informatics Department (CEID), University of Patras. During 1991, he worked at the Speech Processing Laboratory of the National Defense Research Center. From 1992 to 1994 and from 1996 to 1997, he was a Researcher at RACTI. In the period 1994/1995 he was a Postdoctoral Fellow at CCETT, Rennes, France. Since December 1997, he has been with CEID, University of Patras, where he is currently an Associate Professor and Head of the Signal Processing and Communications Laboratory. His research interests include fast algorithms for adaptive filtering, and signal processing for communications. He has served as a Member of Scientific and Organizing Committees of several international conferences and he is currently serving as an Associate Editor of the IEEE Transactions on Signal Processing and the EURASIP Journal on Applied Signal Processing. He is also a Member of the Technical Chamber of Greece.