Soft-Output Decision-Feedback Equalization with a Priori Information

Renato R. Lopes and John R. Barry

School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332-0250

Abstract — **Soft-output equalizers that exploit *a priori* information on the channel inputs play a central role in turbo equalization. Such equalizers are traditionally implemented with the forward-backward or BCJR algorithm, whose complexity is prohibitive for channels with large memory. Many reduced-complexity alternatives to the BCJR algorithm have been proposed that use a linear equalizer and use the *a priori* information to perform soft intersymbol interference cancellation. In this work, we propose a soft-feedback equalizer (SFE) that combines the equalizer output and the *a priori* information to improve interference cancellation. Also, by assuming a statistical model for the *a priori* information and the SFE output, we obtain an equalizer with linear complexity, as opposed to the quadratic complexity of some similar structures. Simulation results show that the SFE may perform within 1 dB of a system based on an BCJR equalizer, within 0.3 dB of quadratic complexity schemes, and consistently outperforms other linear complexity schemes.**

I. Introduction

Soft-output equalizers that exploit *a priori* information on the channel inputs are useful in a variety of applications. Most notably, such equalizers play a central role in turbo equalization, where soft equalizer outputs are fed to a soft-input channel decoder, and where soft decoder outputs are used by the equalizer as *a priori* information in subsequent iterations [1]. Traditionally, the soft outputs take the form of *a posteriori* probabilities (APP) for each transmitted symbol, given the channel outputs and the *a priori* information.

The APP may be computed exactly by the forward-backward or BCJR algorithm [2]. However, the computational complexity of BCJR is exponential in the channel memory, so it is not practical when the channel memory is large. This has motivated the development of reduced-complexity alternatives to the BCJR algorithm, such as the equalizers proposed in [3-8]. These structures use a linear filter to equalize the received sequence. The output of this linear filter contains residual intersymbol interference (ISI), which is estimated based on the *a priori* information, and then cancelled.

In this work, we propose the *soft-feedback equalizer* (SFE), which combines the soft equalizer outputs and the *a priori* information to form more reliable estimates of the residual ISI. A similar system is proposed in [7] that uses hard decisions on the equalizer output to help estimate the residual ISI. However, because hard decisions are used and because the *a priori* information is not combined with the equalizer output before a decision is made, the system with feedback of [7] performs worse than schemes without feedback.

This research was supported in part by NSF grant no. CCR-0082329 and CCR-0121565, and in part by CAPES-Brazil grant no. BEX 1376/96-2.

As in [5-8], the SFE coefficients are computed to minimize the mean squared error (MSE) between the equalizer output and the transmitted symbol. By assuming a statistical model for the *a priori* information and the equalizer output, we obtain a linear complexity equalizer, *i.e.*, the complexity is proportional to the number of taps. A similar statistical model is assumed in [9] to obtain a linear complexity hard-input hard-output equalizer with ISI cancellation (IC). The minimum-MSE (MMSE) schemes in [5-8] have quadratic complexity.

We will see that in special cases, the SFE reduces to a MMSE linear equalizer (LE), an MMSE-decision-feedback equalizer (DFE) or an IC. We will show that the SFE performs reasonably well when compared to the BCJR and quadratic-complexity algorithms, while it consistently outperforms other linear-complexity structures proposed in the literature.

II. Channel Model and Problem Statement

This paper considers the transmission of a sequence of symbols $\boldsymbol{a} = [a_0, \ldots, a_{L-1}]$ through a channel with output

$$r_k = \sum_{m=0}^{\mu} h_m a_{k-m} + n_k, \tag{1}$$

where the channel has memory μ and impulse response $\boldsymbol{h} = [h_0, \ldots, h_\mu]$, and where $n_k \sim \mathcal{N}(0, \sigma^2)$ is real white Gaussian noise. For notational ease, we restrict our presentation to a BPSK alphabet, with $a_k \in \{\pm 1\}$. The results can be extended to other alphabets using the techniques of [8].

In contrast to a classical equalizer, which assumes that the channel inputs are uniformly distributed, we assume that the receiver has *a priori* information about the channel inputs. For binary alphabets, this information is captured by the logarithm of the ratio of the *a priori* probabilities:

$$\lambda_k^p = \log\frac{\Pr(a_k = 1)}{\Pr(a_k = -1)}. \tag{2}$$

Given $\lambda^p = [\lambda_0^p, \ldots \lambda_{L-1}^p]$, the goal of a soft-output equalizer is to compute the APPs $\{p(a_k|\boldsymbol{r})\}$, where $\boldsymbol{r} = [r_0, \ldots r_{L+\mu-1}]$. For binary alphabets, this is equivalent to computing the log-APP ratio L_k, which we loosely refer to as a *log-likelihood ratio* (LLR), defined by:

$$L_k = \log\frac{\Pr(a_k = 1|\boldsymbol{r})}{\Pr(a_k = -1|\boldsymbol{r})}. \tag{3}$$

The LLR may be written as $L_k = \lambda_k + \lambda_k^p$, where λ_k is termed *extrinsic LLR* because it is not a function of the *a priori* information λ^p_k [1]. The concept of extrinsic information is crucial to turbo systems. Indeed, the components of a turbo system only exchange extrinsic information. This fact, and the independence between *a priori* information and extrinsic information, avoids positive feedback of information.

III. Prior Work on Soft ISI Cancellation

In this section, we describe the low-complexity algorithms proposed in [3-8] to compute approximate values of λ_k. The general structure of these algorithms is illustrated in Fig. 1. In this figure, the received signal is filtered by a linear filter $\boldsymbol{f}$. The *a priori* information is used to produce soft estimates $\{\tilde{a}_{l \neq k}\}$ of the interfering symbols $\{a_{l \neq k}\}$, according to:

$$\tilde{a}_l = \mathrm{E}[a_l \,|\, \lambda^p_l] = \tanh(\lambda^p_l/2)\,. \tag{4}$$

These estimates are then fed to a linear filter $\boldsymbol{g}$ whose output is an estimate of the residual ISI at the output of $\boldsymbol{f}$. Thus, the subtraction in Fig. 1 reduces this residual ISI. Notice that, since the equalizer output z_k is used to estimate a_k, the influence of a_k on z_k should not be cancelled. Hence, the zero-th coefficient of $\boldsymbol{g}$ is constrained to be zero.

Since the zero-th tap of $\boldsymbol{g}$ is zero, z_k is not a function of λ^p_k. Thus, z_k can only be used to produce extrinsic information, which can be done by writing

$$z_k = A\, a_k + v_k, \tag{5}$$

where A is a gain and v_k an equivalent noise with variance σ_v^2. Although this noise includes residual ISI, the computation of λ_k from z_k is easy when v_k is assumed to be Gaussian and independent of a_k. In this case, we find that

$$\lambda_k = 2A\, z_k / \sigma_v^2\,. \tag{6}$$

The equalizers proposed in [3] and [4], which are referred to as decision-aided equalizers (DAE), choose $\boldsymbol{f}$ under the assumption that $\tilde{a}_k = a_k$, which leads to the matched filter (MF) solution $f_k = h_{-k}$. Equalizers proposed in [5-8] choose $\boldsymbol{f}$ as an MMSE-LE that depends on the *a priori* information and must be computed for every transmitted symbol. The result is a time-varying equalizer (TVE) whose computational complexity is quadratic in the length of $\boldsymbol{f}$. Also proposed in [5-7] are approximations that yield time-invariant filters $\boldsymbol{f}$ and $\boldsymbol{g}$. In particular, the switched-equalizer (SE) strategy proposed in [7] chooses $\boldsymbol{f}$ as either an MF or a traditional MMSE-LE, depending on the quality of the *a priori* information. In all cases [3-8], the interference cancellation filter $\boldsymbol{g}$ is designed under the assumption that its inputs are equal to the transmitted symbols, yielding $g_k = \sum_l h_l f_{k-l}$ when $k \neq 0$.

IV. The Soft-Feedback Equalizer

We now propose the SFE, a soft-output equalization scheme that shares many similarities with the schemes of [3-8]. However, our approach differs in two substantial ways.

First, at time k, when computing z_k, the previous equalizer outputs $\{\lambda_{k-j}: j > 0\}$ are already known. With these values, we may compute the full LLR $L_{k-j} = \lambda^p_{k-j} + \lambda_{k-j}$, which provides a better estimate of a_{k-j} than λ^p_{k-j} alone. Thus, instead of using $\tilde{a}_{k-j}$ to cancel interference, we propose to use

$$\bar{a}_{k-j} = \mathrm{E}[a_{k-j} \,|\, L_{k-j}] = \tanh(L_{k-j}/2)\,, \tag{7}$$

for $j > 0$. This is similar to the principle behind a DFE. A DFE-based system is also proposed in [7]. However, this system feeds back hard decisions on the equalizer output, without combining them with the *a priori* information, and it performs worse than the schemes described in section III.

Second, as in [5-8], we pass $\bar{a}_k$ and $\tilde{a}_k$ through linear filters whose coefficients, along with $\boldsymbol{f}$, are computed to minimize the MSE $\mathrm{E}[|z_k - a_k|^2]$. However, following [9,10], we use a Gaussian approximation to λ^p_k and to z_k that leads to time-invariant coefficients, resulting in a complexity that is proportional to the length of $\boldsymbol{f}$.

Applying the above two changes to Fig. 1 leads to the proposed SFE structure shown in Fig. 2, where the filters $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are strictly anticausal and strictly causal, respectively, and the filters $\boldsymbol{f}$, $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are chosen to minimize the MSE. The thicker line in the feedback loop represents the only actual change from Fig. 1.

A. Computing the Coefficients

To find the MMSE values of $\boldsymbol{f}$, $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$, we write:

$$z_k = \boldsymbol{f}^{\mathrm{T}} \boldsymbol{r}_k - \boldsymbol{g}_1^{\mathrm{T}}\, \tilde{\boldsymbol{a}}_k - \boldsymbol{g}_2^{\mathrm{T}}\, \bar{\boldsymbol{a}}_k\,, \tag{8}$$

where $\boldsymbol{f} = [f_{-M_1}, \;\ldots\; f_{M_2}]^{\mathrm{T}}$, $\boldsymbol{r}_k = [r_{k+M_1}, \;\ldots\; r_{k-M_2}]^{\mathrm{T}}$, $\boldsymbol{g}_1 = [g_{-M_1}, \;\ldots\; g_{-1}]^{\mathrm{T}}$, $\boldsymbol{g}_2 = [g_1, \;\ldots\; g_{M_2+\mu}]^{\mathrm{T}}$, $\tilde{\boldsymbol{a}}_k = [\tilde{a}_{k+M_1}, \;\ldots\; \tilde{a}_{k+1}]^{\mathrm{T}}$, $\bar{\boldsymbol{a}}_k = [\bar{a}_{k-1}, \;\ldots\; \bar{a}_{k-M_2-\mu}]^{\mathrm{T}}$, the superscript T denotes transpose, and M_1 and M_2 determine the lengths of the filters. Also, we write:

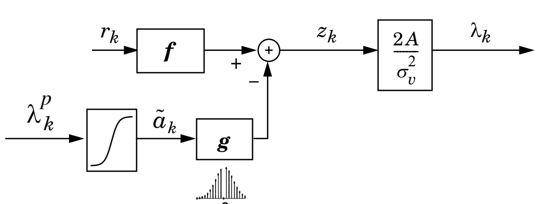

Fig. 1. Interference canceller with *a priori* information.

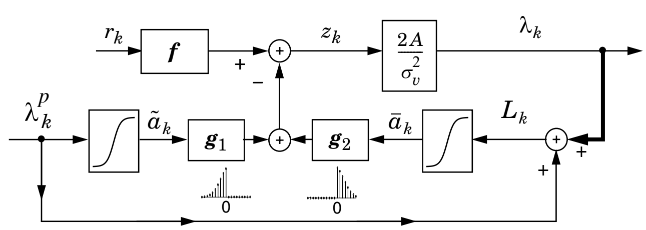

Fig. 2. The proposed SFE.

$$\boldsymbol{r}_k = \mathbf{H}\boldsymbol{a}_k + \boldsymbol{n}_k, \tag{9}$$

where $\boldsymbol{n}_k = [n_{k+M_1}, \ldots\ n_{k-M_2}]^{\mathrm{T}}$, $\boldsymbol{a}_k = [a_{k+M_1}, \ldots\ a_{k-M_2-\mu}]^{\mathrm{T}}$ and $\mathbf{H}$ is the $M \times (M + \mu)$ channel convolution matrix:

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \ldots & h_\mu & 0 & 0 & \ldots & 0 \\ 0 & h_0 & h_1 & \ldots & h_\mu & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & \ldots & h_0 & h_1 & \ldots & h_\mu \end{bmatrix}, \tag{10}$$

where $M = M_1 + M_2 + 1$. Note that, from (9), the output of $\boldsymbol{f}$ suffers the interference of a_{k-j}, $j = -M_1, \ldots\ M_2 + \mu$. This explains the index range of the IC vectors $\tilde{\boldsymbol{a}}_k$ and $\bar{\boldsymbol{a}}_k$.

Now, assume that $\mathrm{E}[\tilde{a}_k a_j] = \mathrm{E}[\bar{a}_k a_j] = \mathrm{E}[\tilde{a}_k \bar{a}_j] = 0$ when $k \neq j$. This is reasonable, since $\tilde{a}_k$ and $\bar{a}_k$ are both approximately a_k, and the transmitted symbols are uncorrelated. Using this assumption, it can be shown [11] that the filters that minimize $\mathrm{E}[|z_k - a_k|^2]$ are given by:

$$\boldsymbol{f} = (\mathbf{H}\mathbf{H}^{\mathrm{T}} - \frac{\alpha_1^2}{E_1}\mathbf{H}_1\mathbf{H}_1^{\mathrm{T}} - \frac{\alpha_2^2}{E_2}\mathbf{H}_2\mathbf{H}_2^{\mathrm{T}} + \sigma^2\mathbf{I})^{-1}\boldsymbol{h}_0, \tag{11}$$

$$\boldsymbol{g}_1 = (\alpha_1 / E_1)\ \mathbf{H}_1^{\mathrm{T}}\boldsymbol{f}, \tag{12}$$

$$\boldsymbol{g}_2 = (\alpha_2 / E_2)\ \mathbf{H}_2^{\mathrm{T}}\boldsymbol{f}, \tag{13}$$

where

$$E_1 = \mathrm{E}[|\tilde{a}_k|^2], \tag{14}$$

$$E_2 = \mathrm{E}[|\bar{a}_k|^2], \tag{15}$$

$$\alpha_1 = \mathrm{E}[\tilde{a}_k a_k], \tag{16}$$

$$\alpha_2 = \mathrm{E}[\bar{a}_k a_k]. \tag{17}$$

The vector $\boldsymbol{h}_0$ is the 0-th column of $\mathbf{H}$, where the columns of $\mathbf{H}$ are numbered as $\mathbf{H} = [\boldsymbol{h}_{-M_1}, \ldots\ \boldsymbol{h}_{M_2+\mu}]$. Also, $\mathbf{H}_1 = [\boldsymbol{h}_{-M_1}, \ldots\ \boldsymbol{h}_{-1}]$ and $\mathbf{H}_2 = [\boldsymbol{h}_1, \ldots\ \boldsymbol{h}_{M_2+\mu}]$. Note that $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are proportional to the strictly causal and anticausal portions of $\sum_l h_l f_{k-l}$.

B. Computing the Expected Values

We now compute the values of E_1, α_1, E_2 and α_2 needed in (11)-(13). Exploiting symmetries, it is not hard to see that these values may be computed by conditioning on $a_k = 1$. For instance, $E_1 = \mathrm{E}[|\tilde{a}_k|^2 | a_k = 1]$ and $\alpha_1 = \mathrm{E}[\tilde{a}_k | a_k = 1]$.

Now, assume, as in [10], that $\lambda_k^p = \gamma_p(a_k + w_k)$, where w_k is AWGN with variance σ_w^2, assumed to be independent of the transmitted sequence, the actual channel noise and the equalizer output, and where $\gamma_p = 2/\sigma_w^2$. Then, conditioning on $a_k = 1$, $\lambda_k^p \sim \mathcal{N}(\gamma_p, 2\gamma_p)$, so

$$\alpha_1 = \psi_1(\gamma_p), \tag{18}$$

$$E_1 = \psi_2(\gamma_p), \tag{19}$$

where

$$\psi_1(\gamma) = \mathrm{E}[\tanh(u/2)],\ u \sim \mathcal{N}(\gamma, 2\gamma), \tag{20}$$

$$\psi_2(\gamma) = \mathrm{E}[\tanh^2(u/2)],\ u \sim \mathcal{N}(\gamma, 2\gamma). \tag{21}$$

Unfortunately, there are no closed-form formulas for $\psi_1(\gamma)$ and $\psi_2(\gamma)$. However, these are well-behaved functions that may be tabulated or computed by simple numerical algorithms. Furthermore, since $\lambda_k^p \sim \mathcal{N}(\gamma_p, 2\gamma_p)$, the ML estimate of γ_p needed in (18) and (19) is given by

$$\hat{\gamma}_p = \sqrt{1 + \frac{1}{L}\sum_{k=0}^{L-1}|\lambda_k^p|^2} - 1\,. \tag{22}$$

Also, consider the Gaussian approximation to λ_k in (5) and (6), and let $\gamma_e = 2A^2/\sigma_v^2$. Then, since $L_k = \lambda_k + \lambda_k^p$,

$$L_k = (\gamma_p + \gamma_e)a_k + \gamma_p w_k + \gamma_e v_k, \tag{23}$$

so that, conditioning on $a_k = 1$, $L_k \sim \mathcal{N}(\gamma_p + \gamma_e, 2(\gamma_p + \gamma_e))$. Thus, E_2 and α_2 are given by

$$\alpha_2 = \psi_1(\gamma_p + \gamma_e), \tag{24}$$

$$E_2 = \psi_2(\gamma_p + \gamma_e). \tag{25}$$

In addition, it can be shown [11] that the value of γ_e needed in (24) and (25) is given by $\gamma_e = 2\boldsymbol{f}^{\mathrm{T}}\boldsymbol{h}_0/(1 - \boldsymbol{f}^{\mathrm{T}}\boldsymbol{h}_0)$. However, note that we need γ_e to compute $\boldsymbol{f}$, but we need $\boldsymbol{f}$ to compute γ_e. To solve this problem, we propose that, given an initial value of γ_e, we compute:

$$\boldsymbol{f} = (\mathbf{H}\mathbf{H}^{\mathrm{T}} - \frac{\alpha_1^2}{E_1}\mathbf{H}_1\mathbf{H}_1^{\mathrm{T}} - \frac{\psi_1^2(\gamma_p + \gamma_e)}{\psi_2(\gamma_p + \gamma_e)}\mathbf{H}_2\mathbf{H}_2^{\mathrm{T}} + \sigma^2\mathbf{I})^{-1}\boldsymbol{h}_0 \tag{26}$$

$$\gamma_e = 2\boldsymbol{f}^{\mathrm{T}}\boldsymbol{h}_0/(1 - \boldsymbol{f}^{\mathrm{T}}\boldsymbol{h}_0), \tag{27}$$

iteratively. It can be shown [11] that this procedure converges very quickly, often after three iterations with γ_e initialized to zero.

C. Special Cases

Note that $\gamma_p = 2/\sigma_w^2$ and $\gamma_e = 2A^2/\sigma_v^2$ are proportional to the SNR of the equivalent channels that generate λ_k^p and λ_k, respectively, and hence reflect the quality of these channels. Careful inspection of (11) – (13) reveals that, for certain values of γ_p and γ_e, the SFE behaves intuitively and reduces to well-known equalizers.

In the limit as γ_p and γ_e grow small, we have $\tilde{a}_k \to 0$ and $\bar{a}_k \to 0$, so that the SFE reduces to a linear equalizer $\boldsymbol{f}$. This is intuitively pleasing, since small values of γ_p and γ_e suggest low reliability, so no interference cancellation should be attempted. We also have that $\alpha_1^2/E_1 \to 0$ and $\alpha_2^2/E_2 \to 0$, so that $\boldsymbol{f}$ of (11) reduces to a traditional MMSE LE.

In the limit as $\gamma_p \to 0$ and $\gamma_e \to \infty$, the SFE reduces to a conventional MMSE-DFE. This is intuitive, since small γ_p implies unreliable *a priori* information, and hence no cancellation of precursor ISI should be performed. Furthermore, large γ_e implies reliable equalizer outputs, in which case postcursor ISI can be effectively cancelled using decision feedback.

In the limit as $\gamma_p \to \infty$, the SFE reduces to a traditional IC. This is intuitive because when γ_p is large, the equalizer has access to reliable estimates for all interfering symbols. In this case, the IC is known to be optimal.

V. Application to Turbo Equalization

Turbo equalizers [1] are a common application of soft-output equalizers with *a priori* information. In turbo equalizers, the transmitted signal is encoded and interleaved with an interleaver π before transmission. The receiver iteratively exchanges extrinsic information between the soft-output equalizer and the decoder, as seen in Fig. 3. The extrinsic information provided by the decoder is used as *a priori* information by the equalizer, and vice-versa. In this section, we discuss the application of the SFE in turbo equalization.

At the first turbo iteration, there is no *a priori* information available at the equalizer, so that $\gamma_p = 0$, yielding $E_1 = \alpha_1 = 0$. To avoid the indeterminate α_1 / E_1 in (11) and (12), we artificially set $E_1 = 1$, $\alpha_1 = 0$, resulting in an equalizer that is between an MMSE-LE and an MMSE-DFE, depending on γ_e. Algorithms based solely on IC have a problem at the first turbo iteration. For instance, to solve this problem, the DAE in [4] uses the BCJR algorithm in the first iteration.

Also, at the first turbo iteration, γ_e is computed with the iterative procedure described in (26) and (27). In later turbo iterations, the equalizer coefficients may be computed with the value of γ_e from the previous turbo iteration. An updated value of γ_e is then computed using (27) and passed on to the next turbo iteration. We have observed that performance does not improve if the iterative procedure in (26) and (27) is used in every turbo iteration to recompute γ_e.

Finally, we have observed that performance can be improved if γ_p and γ_e are estimated using the scalar channel estimator proposed in [12], instead of using (22) and (27). Given an initial estimate $\hat{\gamma}_e^{(1)}$, the estimator of [12] computes

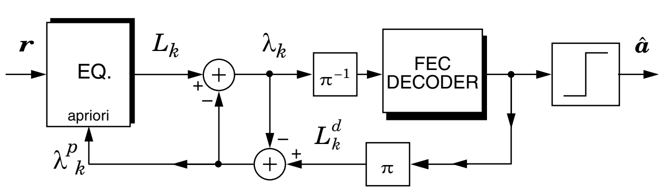

Fig. 3. A turbo equalizer.

$$\hat{A}_i = \frac{1}{L}\sum_{k=0}^{L-1} \tanh(\hat{\gamma}_e^{(i-1)} z_k / 2) z_k ,$$

$$\hat{\sigma}_i^2 = \frac{1}{L}\sum_{k=0}^{L-1} \left\| \hat{A}_i \mathrm{sign}(z_k) - z_k \right\|^2 ,$$

$$\hat{\gamma}_e^{(i)} = 2 \hat{A}_i / \hat{\sigma}_i^2 , \tag{28}$$

where the index $i > 1$ refers to the turbo iteration. If we replace z_k by λ^p_k in the equations above, we obtain an estimate for γ_p. The initial value $\hat{\gamma}_e^{(1)}$ is obtained from the iterative procedure in equations (26) and (27). Also, reflecting our initial belief that λ^p_k obeys the Gaussian approximation, we set $\hat{\gamma}_p^{(1)} = 1$.

VI. Simulation Results

In this section we compare the performance of turbo equalizers based on different soft-output equalizers. In all simulations, we use a rate-$1/2$ recursive systematic convolutional encoder with parity generator $(1+D^2)/(1+D+D^2)$ followed by a random interleaver whose length is equal to the block length. Channel knowledge is assumed. Furthermore, the primary application of the SFE is with channels with long memory, for which the BCJR equalizer is not practical. However, the BCJR equalizer is traditionally used as a benchmark. To facilitate comparison of the SFE with the BCJR equalizer, we only consider short channels in this section.

We begin with the simulation scenario of [7], in which $K = 2^{15}$ message are encoded and transmitted through the channel $\boldsymbol{h}$ = [0.227, 0.46, 0.688, 0.46, 0.227]. The equalizers use $M_1 = 9$, $M_2 = 5$, and the SNR per message bit is $E_b / N_0 = \|\boldsymbol{h}\|^2 / \sigma^2$. In Fig. 4, we show the BER performance of turbo equalizers based on the BCJR, TVE, SE and the SFE, after 14 turbo iterations, averaged over 100 trials. For comparison purposes, we also show the performance of the coded system for a channel that does not introduce ISI. As we can see, the SFE performs almost as well as the TVE (which has quadratic complexity), while its complexity is comparable

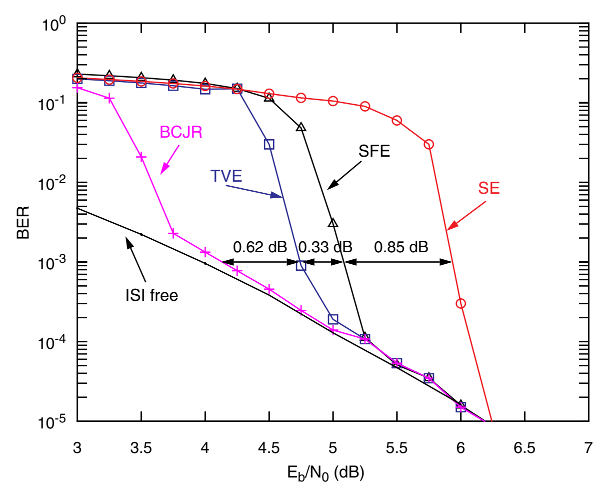

Fig. 4. BER performance for the simulation scenario of [7].

to that of the SE (which has linear complexity). It is interesting to point out that, for high enough E_b/N_0, the performance of all systems approaches that of the coded system in an ISI-free channel.

The performance gap between the different techniques is a strong function of the channel. To see this, we simulate the transmission of $K = 2^{11}$ encoded bits through $\boldsymbol{h}$ = [0.23, 0.42, 0.52, 0.52, 0.42, 0.23], which introduces severe ISI [13]. We used $M_1 = 15$ and $M_2 = 10$. For each E_b/N_0, the number of blocks detected in error was computed every 30 blocks. If this number exceeded 100, or the number of blocks exceeded 1,000, we would stop running the simulation for that E_b/N_0.

The performance of the turbo equalizers based on BCJR, DAE of [4], SE and the SFE, is shown in Fig. 5, where we plot the BER versus E_b/N_0 for the turbo equalizers. The maximum number of iterations shown for each scheme is that after which the equalizers stopped improving. For the DAE, error propagation is a problem for $E_b/N_0 < 10$ dB, as evidenced by its poor performance in this SNR range. Also the first turbo iteration of the DAE is done with a BCJR equalizer, which precludes its application to channels with long memory. We can also see in Fig. 5 that, for 6 iterations and a BER of 10^{-3}, the SFE is around 2.6 dB better than the SE. However, performance cannot be further improved with the SE. With the SFE, on the other hand, a gain of 0.65 dB is possible with 2 extra iterations, and a 1.4 dB gain is possible with 10 more iterations. One possible explanation for this performance gap is that all the zeros of the channel are on the unit circle, so that decision feedback structures such as the proposed algorithm tend to perform better than linear filters. The gap between the SFE- and BCJR-based systems is roughly 3 dB.

VII. Summary

We proposed the SFE, a low-complexity soft-output equalizer that exploits *a priori* information about the transmitted symbols to perform soft interference cancellation. The SFE achieves a compromise between linear equalization, decision feedback equalization and IC by choosing the equalizer coefficients according to the quality of the *a priori* information and of the equalizer output. Since the SFE exploits *a priori* information, it is well-suited for turbo equalization for channels with large memory.

The SFE differs from similar structures [3-8] in two ways. First, it successfully combines the soft equalizer outputs and the *a priori* information to improve IC. In contrast, the decision-feedback structure proposed in [7] uses only hard decisions on the equalizer output, and performs worse than its linear counterpart. Also, by assuming a statistical model for the *a priori* information, we obtain a time-invariant, linear complexity equalizer, as opposed to the quadratic complexity of the MMSE structures in [5-8]. Simulation results demonstrate that the SFE outperforms other structures of comparable complexity by as much as 3 dB at a BER of 10^{-3}.

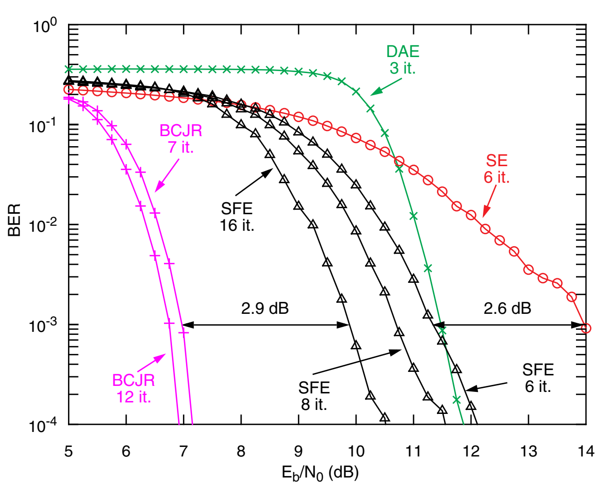

Fig. 5. BER performance of some turbo equalizers.

References

[1] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative Correction of Intersymbol Interference: Turbo-Equalization," *European Trans. on Telecomm.*, vol. 6, no.5, pp.507-511, Sept.-Oct. 1995.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Trans. on Info. Theory*, pp. 284-287, March 1974.

[3] A. Glavieux, C. Laot and J. Labat, "Turbo Equalization Over a Frequency Selective Channel," *International Symp. on Turbo Codes and Related Topics*, Brest, France, pp. 96-102, Sep 1997.

[4] Z. Wu and J. Cioffi, "Turbo Decision Aided Equalization for Magnetic Recording Channels," *IEEE Global Comm. Conf.*, pp. 733-738, Dec. 1999.

[5] D. Raphaeli and A. Saguy, "Reduced Complexity APP for Turbo Equalization," *IEEE Int. Conf. Comm.*, vol. 3, pp. 1940-1943, 2002.

[6] J. Rößler, W. Gerstacker, A. Lampe, and J. Huber, "Matched-Filter- and MMSE-Based Iterative Equalization with Soft Feedback for QPSK Transmission," *International Zurich Seminar on Broadband Comm.*, pp. 19-1 - 19-6, February 2002.

[7] 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, May 2002.

[8] A. Dejonghe and L. Vanderdorpe, "Turbo Equalization for Multilevel Modulation: an Efficient Low-Complexity Scheme," *IEEE International Conf. Comm.*, vol. 3, pp. 1863-1867, 2002.

[9] A. A. Chan and G. Wornell, "A Class of Block-Iterative Equalizers for Intersymbol Interference Channels: Fixed Channels Results," *IEEE Trans. on Comm.*, vol. 49, no. 11, pp. 1966-1976, Nov. 2001.

[10] S. ten Brink, "Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes," *IEEE Trans. on Comm.*, vol. 49, pp. 1727-1737, Oct. 2001.

[11] R. R. Lopes, *Iterative Estimation, Equalization and Decoding*, Ph.D. Thesis, Georgia Inst. Tech., Atlanta, 2003.

[12] R. R. Lopes and J. R. Barry, "Exploiting Error-Control Coding in Blind Channel Estimation," *IEEE Global Comm. Conf.*, vol. 2, pp. 1317-1321, November 2001.

[13] J. Proakis, *Digital Communications*, McGraw-Hill, 1995.