

Linear Turbo Equalization Analysis via BER Transfer and EXIT Charts

Seok-Jun Lee, Andrew C. Singer, *Member, IEEE*, and Naresh R. Shanbhag, *Senior Member, IEEE*

Abstract—In this paper, two analytical methods are presented to investigate the soft information evolution characteristics of a soft-input soft-output (SISO) linear equalizer and its application to the design of turbo equalization systems without the reliance on extensive simulation. Given the channel and a SISO equalization algorithm, one method explored is to analytically compute the bit-error rate (BER) of the SISO equalizer in two extreme cases (no and perfect *a priori* information) from which a BER transfer characteristic is estimated. The second approach is to compute the mutual information [a key parameter of the extrinsic information transfer (EXIT) chart] at the two end points of the EXIT function. Then, by modeling the SISO equalizer BER transfer and EXIT functions as linear, some of the behavior of linear turbo equalization, such as how the BER performance can be improved as iterations proceed, can be predicted. Further, soft information evolution characteristics of different linear SISO equalizers can be compared and the usefulness of iterative methods such as linear turbo equalization for a given channel can be determined. Compared with existing methods for generating EXIT functions, these predictive methods provide insight into the iterative behavior of linear turbo equalizers with substantial reduction in numerical complexity.

Index Terms—Bit-error rate (BER) analysis, extrinsic information transfer (EXIT) chart, iterative decoding, linear equalizer, soft-input soft-output (SISO) equalizer, turbo equalizer.

I. INTRODUCTION

A wide variety of communication systems encounter inter-symbol interference (ISI) during transmission of digital information. To combat ISI, receivers are often equipped with an equalizer and use error correction codes to remove residual errors after equalization. In conventional solutions, equalization and decoding are often disjoint (see Fig. 1), which is usually suboptimal in terms of minimizing bit-error rate (BER). After the discovery of turbo codes [1], the turbo principle has been adopted in a variety of communication systems to solve these problems jointly [2]–[4]. The result, *turbo equalization*,

was proposed [5] as a method for joint equalization and decoding, where soft-input soft-output (SISO) equalization and decoding operations are iteratively combined through the exchange of soft information. The soft information can be in the form of probabilities, *a priori* log-likelihood ratios (LLRs) defined as $L(X) \triangleq \ln(\Pr(X = +1)/\Pr(X = -1))$ for $X \in \{+1, -1\}$, or functions of these, such as soft symbols defined as $\bar{X} \triangleq \Pr(X = +1) - \Pr(X = -1)$. This iterative exchange of soft information enables substantial gains beyond those obtained from separate equalization and decoding [5]–[13].

Recently, successful application of the turbo principle has motivated a number of results on the analysis of turbo decoding and turbo equalization [8], [14]–[19]. Many such analysis methods have investigated the convergence threshold of iterative decoding algorithms in terms of signal-to-noise ratio (SNR) by monitoring a single parameter across multiple iterations. This parameter is assumed to capture the salient characteristics of the behavior of the SISO blocks which are employed in turbo receivers. Such parameters include mutual information between a given transmitted symbol and the associated LLR computed in the SISO decoders [8], [14], [15], estimated noise variance [16], the probability density function of the LLR [17]–[19], and a number of other related metrics [15].

A major drawback of such analysis methods is the reliance on extensive computer simulations in order to obtain the necessary single parameter mapping functions. In this paper, the evolution characteristics of two such parameters (BER and the mutual information between the transmitted symbol and its associated LLR) in linear SISO equalization are estimated without running these extensive simulations. Given channel knowledge and a time-invariant set of equalizer coefficients, the BER of the linear SISO equalizer is derived in the two cases where no and perfect *a priori* LLRs are available to the SISO equalizer. Another approach we consider is to analytically compute the mutual information between the transmitted data and their LLRs at the two end points of an extrinsic information transfer (EXIT) function. By using the end-point values, the SISO equalizer BER transfer and EXIT functions are approximated as linear [8], [15] and the evolution characteristics are predicted for all iterations. Based on these approaches, convergence behavior of the turbo equalizer can be investigated, different linear SISO equalizers can be compared in terms of BER and mutual information transfer characteristics, and the efficacy of linear turbo equalization can be determined for a given channel without running long simulations. Hence, the presented methods are useful in designing a practical linear turbo equalizer for a given channel.

This paper is organized as follows. We begin with a brief review of linear turbo equalization and a single parameter tracking

Manuscript received March 6, 2004; revised September 27, 2004. This work was supported by the National Science Foundation under Grants CCR 99-79381, CCR-0092598, and ITR 00-85929. This paper was presented in part at the 37th Asilomar Conference on Signals, Systems, and Computers, Monterey, CA, November 2003, and in part at the IEEE 2003 Global Communications Conference, San Francisco, CA, December 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Vikram Krishnamurthy.

S.-J. Lee is with the Communication Systems Laboratory, DSP Solutions R&D Center, Texas Instruments, Dallas, TX 75243 USA (e-mail: s-lee8@ti.com).

A. C. Singer and N. R. Shanbhag are with the Coordinated Science Laboratory, Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: acsinger@uiuc.edu; shanbhag@uiuc.edu).

Digital Object Identifier 10.1109/TSP.2005.850375

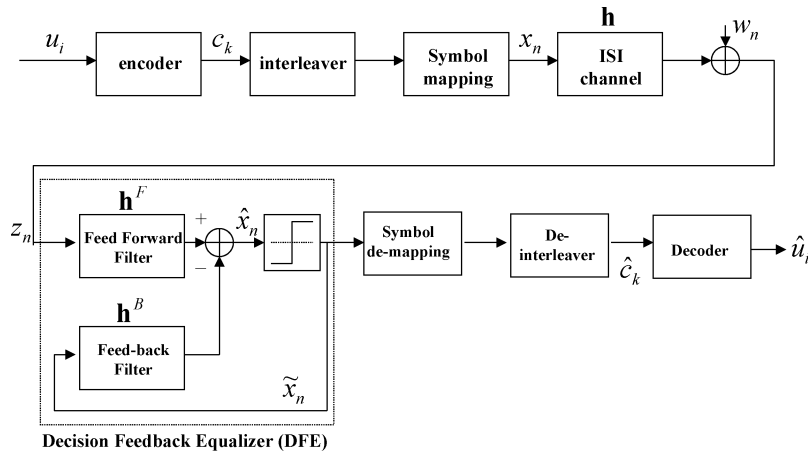


Fig. 1. Conventional communication system, in which the equalization algorithm (shown as a DFE) and the decoding algorithm operate separately, and \hat{x}_n , \hat{c}_k , and \hat{u}_i are the estimates of x_n , c_k , and u_i , respectively.

method for analyzing iterative equalization and decoding algorithms, with an example EXIT chart in Section II. In Section III, methods for investigating the BER and mutual information evolution characteristics of the linear SISO equalizer are presented. We then present the usefulness of the proposed methods in choosing turbo equalization parameters, and simulation results are provided in Section IV.

II. LINEAR TURBO EQUALIZATION

This section describes linear turbo equalization and the single parameter-based analysis tool, or the so-called EXIT chart [14]. Throughout this paper, we assume binary phase shift keying (BPSK) modulation, but extension to higher order modulation is straightforward and described in greater detail in [8] and [9].

The original turbo equalization scheme [5] used a maximum *a posteriori* probability (MAP) detector for channel equalization with a MAP decoder for the error control code. However, the complexity of such MAP-based turbo equalization can be prohibitive for modulation formats with modest spectral efficiency and large delay spreads [6]–[9]. To overcome this problem, several low complexity approaches [6]–[9] have also been proposed by replacing the MAP SISO equalizer with a lower complexity linear SISO equalizer such as an ISI canceller [6], [7] or minimum mean-square error (MMSE) equalizer or approximations to these [8], [9]. This new class of iterative equalization and decoding algorithms has been called linear turbo equalization, which can combat ISI with a higher degree of computational efficiency, even in the presence of long channel memory and high spectral efficiency modulation schemes and, hence, is considered in this paper.

A. Linear Turbo Equalization Algorithm

Consider the communication system depicted in Fig. 1, in which the receiver contains a decision feedback equalizer (DFE) and a separate decoder. The transmission and receiving tasks are applied to blocks of data bits $u_i \in \{0, 1\}$ of length N_u . The data bits are encoded yielding the coded sequence c_k of length $N_c = N_u \cdot (1/R)$, where R is the code rate of the encoder that the interleaver permutes in time. The mapper maps these code bits to BPSK modulated symbols $x_n \in \mathcal{B} = \{+1, -1\}$ that

are transmitted over an ISI channel corrupted by additive white Gaussian noise (AWGN). The channel output z_n is given by

$$z_n = \sum_{k=-l_1}^{l_2} h_k \cdot x_{n-k} + w_n \quad (1)$$

for a linear, time-invariant channel with impulse response h_k of length $l_1 + l_2 + 1$ and additive noise sequence w_n of variance σ_w^2 . In a conventional DFE-based receiver, where equalization and decoding are disjoint, the DFE computes the estimates \hat{x}_n based on the channel observations z_n and the past hard decisions \tilde{x}_n from the slicer (decision device) as follows:

$$\hat{x}_n = \sum_{k=-L_1}^{L_2} h_k^F \cdot z_{n-k} - \sum_{k=1}^{K_2} h_k^B \cdot \tilde{x}_{n-k} \quad (2)$$

where $L_1 + L_2 + 1$ and K_2 are the number of taps in the feed-forward and feedback filters, respectively, and h_k^F and h_k^B are their respective filter coefficients. The slicer outputs are then processed to yield bit estimates \hat{u}_i by the channel decoder. Here, the equalizer coefficients can be computed via many design criteria, including the MMSE or least mean-square (LMS) criteria [20], for example.

We now consider a general structure for a linear turbo equalizer, shown in Fig. 2, which can be used to describe a number of different turbo equalization algorithms by a suitable choice of (possibly time varying) filter coefficients h_k^F and h_k^B and soft symbols \bar{x}_n [6]–[9]. In comparison with a conventional DFE-based receiver, an important difference in the linear turbo equalizer of Fig. 2 is the exchange of soft information (log-likelihood ratios or functions of these) between the linear SISO equalizer and the SISO decoder and subsequent iterative processing in this feedback loop. The SISO equalizer produces the LLR $L_o^E(x_n)$ on each symbol as

$$L_o^E(x_n) = \ln \frac{f_{\hat{x}_n|x_n}(\hat{x}_n|x_n = +1)}{f_{\hat{x}_n|x_n}(\hat{x}_n|x_n = -1)} \quad (3)$$

where $f_{\hat{x}_n|x_n}$ is the conditional probability density of the estimate \hat{x}_n . The LLRs $L_o^E(x_n)$, $n = 0, 1, \dots, N_c - 1$ are then passed through the de-interleaver and input to the SISO decoder

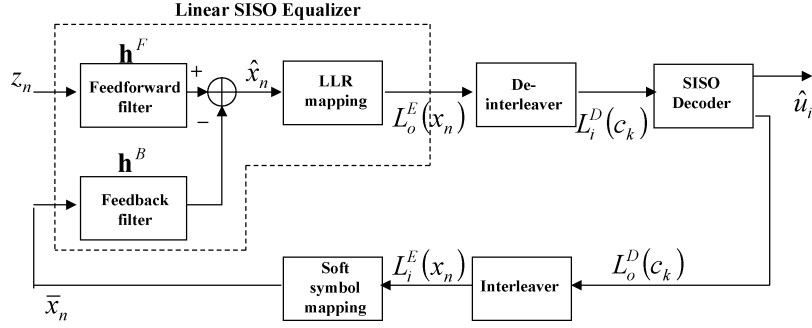


Fig. 2. Linear turbo-equalizer block diagram, in which soft information is exchanged between the SISO equalizer and the SISO decoder. The superscripts E and D imply equalization and decoding, respectively, and the subscripts i and o refer to input and output, respectively.

as *a priori* LLRs $L_i^D(c_k)$, $k = 0, 1, \dots, N_c - 1$, on each coded bit

$$L_i^D(c_k) = \ln \frac{\Pr(c_k = 1)}{\Pr(c_k = 0)} \quad (4)$$

where $x_n = 2\mathbf{\Pi}(c_k) - 1$, $L_o^E(x_n) = \mathbf{\Pi}(L_i^D(c_k))$ and $\mathbf{\Pi}(\cdot)$ refers to the interleaving operation of Fig. 2. In turn, the SISO decoder attempts to further improve the LLRs on the coded bits [7]. If $\mathbf{L}_i^D(\mathbf{c})$ is the updated LLR sequence from the SISO equalizer, then the *a posteriori* LLR $\Lambda^D(c_k)$ for the k th bit is given by

$$\Lambda^D(c_k) = \ln \frac{\Pr(c_k = 1 | \mathbf{L}_i^D(\mathbf{c}))}{\Pr(c_k = 0 | \mathbf{L}_i^D(\mathbf{c}))}. \quad (5)$$

Several algorithms exist for computing or approximating this output *a posteriori* LLR; see [21] for details.

The decoder output LLR $L_o^D(c_k)$, which is passed back to the SISO equalizer to form the iterative loop, can be computed by subtracting the *a priori* LLR from $\Lambda^D(c_k)$

$$L_o^D(c_k) = \Lambda^D(c_k) - \ln \frac{\Pr(c_k = 1)}{\Pr(c_k = 0)} = \Lambda^D(c_k) - L_i^D(c_k). \quad (6)$$

In the so-called “turbo principle” [4], the term on the left $L_o^D(c_k)$ is called “extrinsic information,” and has the *a priori* information removed from it, since this information was already known to the equalizer. Such processing is well known in turbo algorithms and prevents early limit-cycle behavior. The updated $L_o^D(c_k)$ is passed through an interleaver and used as input to the soft symbol mapping block in Fig. 2. The LLRs updated by a SISO decoder in the previous iteration are converted to the soft symbol values \bar{x}_n by taking a statistical expectation under the assumption of independent LLRs. The interleaving allows for this independence assumption, assuming efficient interleaving [8] and large block size.

In analogy to the conventional DFE and since the mapping is invertible, we compute soft symbols, given by

$$\bar{x}_n = \Pr(x_n = +1) - \Pr(x_n = -1) = \frac{e^{L_i^E(x_n)} - 1}{1 + e^{L_i^E(x_n)}} \quad (7)$$

which are fed back to the linear SISO equalizer, rather than the hard decisions \hat{x}_n (produced by quantizing \hat{x}_n) used in the traditional DFE. Doing so enables additional information exchange

(or cooperation) between these two SISO blocks. As soft information is circulated via this algorithmic loop, more reliable soft information produced in one SISO block helps the other to improve. Thus, as iterations proceed, better performance can be achieved in terms of BER and the process can be terminated using one of a number of possible termination criteria [22]–[25].

In the linear SISO equalizer, the channel outputs z_n and soft symbols \bar{x}_n are processed as follows:

$$\hat{x}_n = \sum_{k=-L_1}^{L_2} h_k^F \cdot z_{n-k} - \sum_{k=-K_1}^{K_2} h_k^B \cdot \bar{x}_{n-k} \quad (8)$$

where h_0^B is usually set to zero when generating the estimated symbol \hat{x}_n to avoid the short feedback cycle of soft information; this is analogous to the so-called extrinsic information used in turbo decoding and MAP-based turbo equalization [5], [8]. Note that the feedback filter can have both causal and noncausal taps where the noncausal taps can be used starting from the second iteration of the turbo equalization procedure, since it operates in a batch mode, as described. There are several algorithms in computing \mathbf{h}^F and \mathbf{h}^B ; see [6]–[9], [26], and [27] for details.

B. EXIT Charts

In this section, we introduce a useful analysis method for turbo receivers, which traces the evolution of a single parameter through multiple iterations. The EXIT chart [14] is one such method, which attempts to track the mutual information between the transmitted symbol and its LLR computed in SISO blocks [8], [14], [15]. In ten Brink’s original approach [14], empirical density functions are used to compute the mutual information I_i and $I_o \in [0, 1]$ between input/output LLR and x_n (for equalization) or c_n (for decoding), respectively. Mutual information has been shown to predict the convergence behavior better than a variety of other proposed measures [8], [14], [15].

In EXIT chart analysis [8], each constituent SISO block can be modeled as a transfer function for this single parameter. In order to obtain the transfer functions, the LLR of each bit is modeled as a collection of independent and identically distributed (i.i.d.) random variables whose values at each bit position form an independent realization [8], [14], [17], [18]. The LLRs $L_i^E(x_n)$, $L_o^E(x_n)$, $L_i^D(c_k)$, and $L_o^D(c_k)$ are i.i.d. samples of random variables L_i^E , L_o^E , L_i^D , and L_o^D , respectively. These

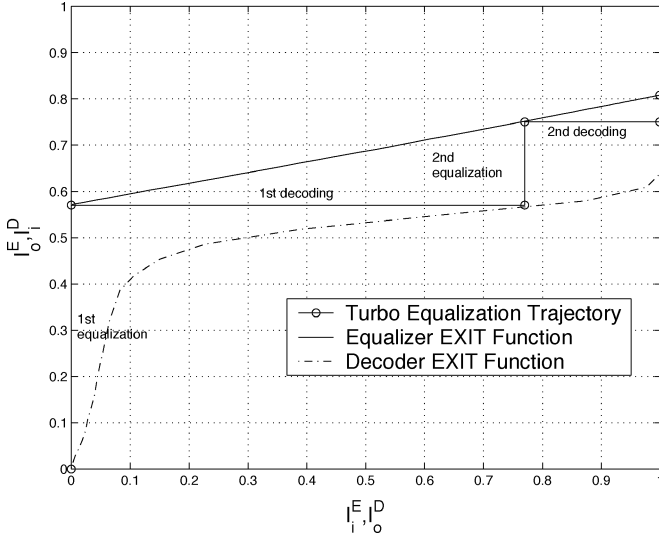


Fig. 3. Example turbo equalization EXIT chart where the iterative trajectory can be visualized given a SISO equalizer EXIT function and a SISO decoder EXIT function. The transfer functions $I_o^E = T_E(I_i^E = I_o^D)$ and $I_o^D = T_D(I_i^D = I_o^E)$ can be determined through extensive simulations [8].

random variables are reasonably well approximated (and often modeled) as i.i.d. with a Gaussian density

$$f_L(l|x_n) \triangleq f_L(l|X = x_n) = \frac{1}{\sqrt{2\pi}\sigma_L} \exp\left(-\frac{\left(l - \frac{x_n\sigma_L^2}{2}\right)^2}{2\sigma_L^2}\right) \quad (9)$$

where σ_L^2 is the variance of the LLR random variables [18]. Since the conditional probability density functions (PDFs) of L_o^E and L_o^D are each assumed to be a function of a single parameter σ_L , the equalization and decoding blocks can be depicted as a single-parameter transfer function of L_i^E or L_i^D , as shown in [8, Fig. 6]. Thus, in existing analysis methods based on EXIT charts [8], [14], the random variables of input LLR L_i are assumed to have distribution $f_{L_i}(l|x) \triangleq (1/\sqrt{2\pi}\sigma_{L_i})\exp(-(l - (x_n\sigma_{L_i}^2/2))^2/2\sigma_{L_i}^2)$. Then, for a range of values of σ_{L_i} and for each distribution empirically generated and determined by σ_{L_i} , a histogram of the output LLRs $P_{L_o}(l|x)$ is generated based on extensive simulations, where the step size in each such histogram measurement must be sufficiently small. The PDF of the random variables L_o of output LLRs is then estimated for the SISO equalizer and decoder from this histogram. Mutual information I_i and I_o can then be computed numerically using

$$I_i = \frac{1}{2} \sum_{x \in \mathcal{B}} \int_{-\infty}^{\infty} f_{L_i}(l|x) \log_2 \frac{2f_{L_i}(l|x)}{f_{L_i}(l+1) + f_{L_i}(l-1)} dl$$

$$I_o = \frac{1}{2} \sum_{x \in \mathcal{B}} \sum_l P_{L_o}(l|x) \log_2 \frac{2P_{L_o}(l|x)}{P_{L_o}(l+1) + P_{L_o}(l-1)}. \quad (10)$$

Note that $I_i = 0$ and $I_i = 1$, imply no, and perfect, *a priori* LLRs, respectively, and $I_o = 0$ and $I_o = 1$ indicate the least and the most reliable output LLRs, respectively. Thus, when $I_o = 1$, no further iteration is warranted.

Fig. 3 shows an example of a turbo equalization EXIT chart, where we have selected the channel with system function

$H(z) = 0.407z^{-1} + 0.815 + 0.407z$ and used a recursive systematic convolutional (RSC) encoder with generator polynomial $(23, 35)_8$. Note that the output LLR $L_o^E(x_n)$ becomes more reliable (I_o^E increases) as the input LLR $L_i^E(x_n)$ becomes more reliable (I_i^E increases) in the SISO equalizer. Considering both the equalizer and decoder EXIT functions jointly in a single EXIT chart, the iteration process can be visualized as a trace between the two EXIT functions by setting $I_o^E \rightarrow I_i^D$ and $I_o^D \rightarrow I_i^E$. For this example, if the equalizer generates LLRs with $I_o^E = 0.57$ in the first iteration, then the decoder improves output LLRs with $I_o^D = 0.77$. This computation can be traced by the arrow-line staircase plot in Fig. 3, and, after three iterations, the $I_o^D = 0.99$ condition, and corresponding algorithmic convergence, is nearly achieved. Note that in order for this graphical depiction of convergence to succeed in achieving the $I_o^D = 1$ point, a “tunnel” between the two EXIT functions of the equalizer and decoder must appear and guide the iterative trajectory to $I_o^D = 1$ point.

The conventional method for obtaining an equalizer EXIT function as in Fig. 3 involves extensive open-loop simulation(or sampling) of the input–output relationship for the SISO devices. In Fig. 4, we depict the procedure explained in [8] for sampling this input–output relationship in which samples of $L_i^E(x_n)$ are drawn along with samples of z_n and the resulting measured PDF of L_o^E is used in numerically computing (10). If N_S samples of L_o^E are generated for N_E points of an EXIT function, the required number of computations is proportional to $N_E \cdot (N_S \cdot \text{equalization} + \text{histogram measure})$. To be statistically sufficient, N_S is usually much greater than $Q^{l_1+l_2+L_1+L_2+1}$, where Q is the alphabet size. Further, given a channel impulse response, the SISO equalizer EXIT function must be recomputed (through numerical integration of the extensive simulation-based histograms) for different equalization algorithms or for varying channel signal-to-noise ratio (SNR). Hence, the computational complexity will be $O(N_E \cdot N_S) \gg Q^{l_1+l_2+L_1+L_2+1}$. In the next section, we present two analytical methods to investigate the evolution characteristics of LLRs of the linear SISO equalizer in terms of BER and mutual information with approximately $Q^{l_1+l_2+L_1+L_2+1}$ complexity, implying a substantial reduction in the required computational complexity for the generation of the EXIT functions.

III. ANALYSIS

Fig. 5 depicts the two analytical approaches we explore for investigating the evolution characteristics of BER and mutual information in SISO linear equalization. We first describe a method to derive the BER that would be achieved from quantizing the estimates \hat{x}_n . We then derive the PDF of the output LLR $L_o^E(x_n)$ and compute the equalizer output mutual information I_o^E . In order to derive the BER and I_o^E at the two extremes corresponding to no *a priori* and perfect *a priori* LLR cases, an equalized channel model is considered as shown in Figs. 6 and 7. In the first iteration, no *a priori* LLR is available ($I_i^E = 0$). Hence, $L_i^E(x_n) = 0$ leads to $\bar{x}_n = 0$ so that

¹Where $F(x) = O(G(x))$ implies $\lim_{x \rightarrow \infty} (F(x)/G(x)) = c$, $0 < c < \infty$.

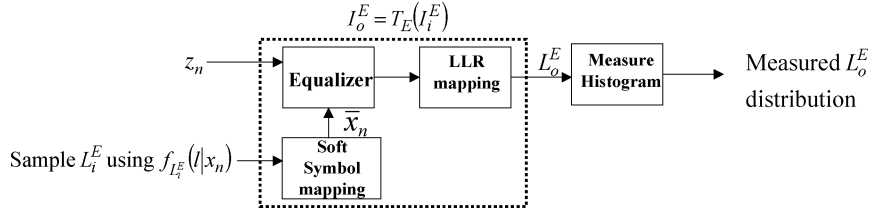


Fig. 4. Conventional method for empirically obtaining an equalizer EXIT function.

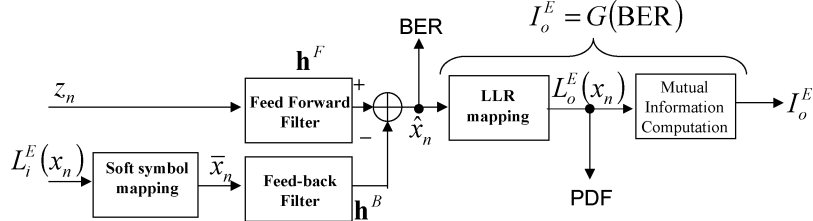


Fig. 5. Two analysis methods are proposed in this paper. One computes the BER at the equalizer output at the two end points of a BER transfer curve. The other derives a PDF at the LLR mapping output for analytically computing mutual information between the transmitted symbol and its associated soft information at the two end points of an EXIT chart.

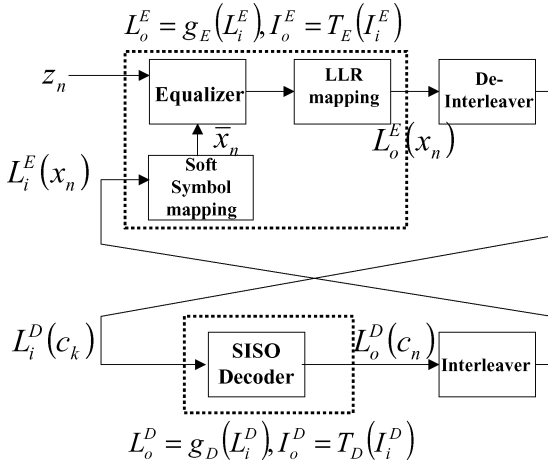


Fig. 6. Equalized channel model for the first iteration, where $L_i^E(x_n)$ is assumed to be zero and $H^F(e^{j\omega})$ and $H(e^{j\omega})$ are the frequency response of the feedforward and channel impulse responses, respectively. The equalized channel response is $H_{eq}(e^{j\omega})$.

the feedback path can be neglected, and we have an equalized channel as in Fig. 6. After convergence, perfect *a priori* LLR is supposed to be available to the receiver, leading to the condition $\bar{x}_n = x_n$. Thus, the equalized channel can be depicted as in Fig. 7. Using these two equalized channel models, we derive the BER and compute the mutual information I_o^E at these two end points.

A. BER Derivation

For a given channel impulse response and equalization algorithm, i.e., a time-invariant set of filter coefficients, Fig. 6 depicts the equalized channel model in the first iteration. The responses of the channel and the equalizer are combined such that the output power spectral density $S_{\hat{x}\hat{x}}(e^{j\omega})$ can be written

$$S_{\hat{x}\hat{x}}(e^{j\omega}) = |\mathbf{H}(e^{j\omega})|^2 |\mathbf{H}^F(e^{j\omega})|^2 S_{xx}(e^{j\omega}) + |\mathbf{H}^F(e^{j\omega})|^2 S_{ww}(e^{j\omega}) \quad (11)$$

where $S_{xx}(e^{j\omega})$ is the input power spectral density and $S_{ww}(e^{j\omega})$ is the power spectral density of the noise process w_n , which is assumed white, i.e., $S_{ww}(e^{j\omega}) = \sigma_w^2$. The estimated symbol \hat{x}_n is then given by

$$\hat{x}_n = (\mathbf{h} * \mathbf{h}^F)^T \cdot \mathbf{x}_n + v_n = (\mathbf{h}^{eq})^T \cdot \mathbf{x}_n + v_n \quad (12)$$

where $\mathbf{h} * \mathbf{h}^F$ implies the convolution of two sequences whose nonzero elements comprise the entries of two single column vectors (\mathbf{h} and \mathbf{h}^F), $\mathbf{h} = [h_{-l_1}, \dots, h_{-1}, h_0, h_{+1}, \dots, h_{+l_2}]^T$, \mathbf{h}^{eq} denotes the equalized channel response, the filtered noise sequence v_n has variance $\sigma_v^2 = \sigma_w^2 \sum_k |h_k^F|^2$, $\mathbf{x}_n = [x_{n+M_1}, \dots, x_{n+1}, x_n, x_{n-1}, \dots, x_{n-M_2}]^T$, $M_1 = L_1 + l_1$, and $M_2 = L_2 + l_2$. By defining

$$\mathbf{h}^{eq0} \triangleq [h_{-M_1}^{eq}, \dots, h_{-1}^{eq}, 0, h_{+1}^{eq}, \dots, h_{M_2}^{eq}]^T$$

the estimated symbol can be expressed

$$\hat{x}_n = h_0^{eq} \cdot x_n + (\mathbf{h}^{eq0})^T \cdot \mathbf{x}_n + v_n = h_0^{eq} \cdot x_n + e_n + v_n \quad (13)$$

where e_n denotes the ISI caused by neighboring symbols and is a sequence of i.i.d samples of the random variable ϵ with $\Pr(\epsilon = \epsilon_j) = (1/M)$ for $j = 1, 2, \dots, M (= 2^{M_1+M_2})$ for BPSK, the cardinality of \mathbf{h}^{eq0} , since x_n is assumed to be i.i.d. with $\Pr(x_n = +1) = \Pr(x_n = -1)$.

Assuming the transmitted symbols are equiprobable, the BER, P_e , can then be expressed for a memoryless decision device as

$$P_e = \frac{1}{2} \{ \Pr(\hat{x}_n < 0 | x_n = 1) + \Pr(\hat{x}_n > 0 | x_n = -1) \} = \frac{1}{2} \left\{ \Pr \left(h_0^{eq} + \sum_{k \neq 0} h_k^{eq} x_{n-k} + v_n < 0 \right) + \Pr \left(-h_0^{eq} + \sum_{k \neq 0} h_k^{eq} x_{n-k} + v_n > 0 \right) \right\}. \quad (14)$$

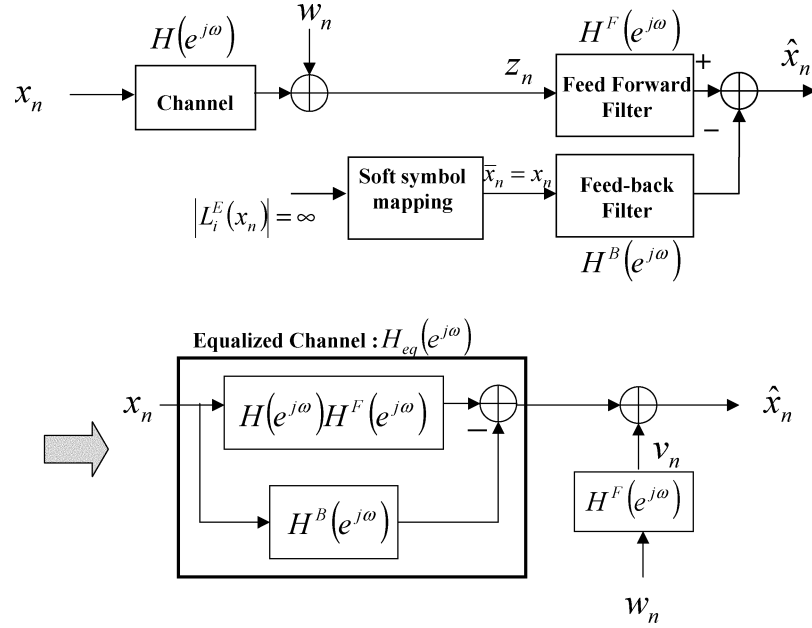


Fig. 7. Equalized channel model after convergence, where the soft symbols are assumed to be equal to the transmitted symbols and $H^B(e^{j\omega})$ is the frequency response of the feedback filter.

Using $\mathbf{x}_n^0 \triangleq [-x_{n-M_1}, \dots, -x_{n-1}, 0, -x_{n+1}, \dots, -x_{n+M_2}]$, then

$$P_e = \frac{1}{2M} \left\{ \sum_{\forall \mathbf{x}_n^0} \Pr(v_n < (\mathbf{h}^{eq})^T \cdot \mathbf{x}_n^0 - h_0^{eq}) + \sum_{\forall \mathbf{x}_n^0} \Pr(v_n > (\mathbf{h}^{eq})^T \cdot \mathbf{x}_n^0 + h_0^{eq}) \right\} = \frac{1}{2M} \sum_{\forall j} \Pr(|v_n - \epsilon_j| > h_0^{eq}) \quad (15)$$

where v_n has a Gaussian density, $f(v) = (1/\sqrt{2\pi}\sigma_v)\exp(-v^2/2\sigma_v^2)$. The error region is shifted depending on the sign of the residual ISI (e_n). Thus, the BER after the first iteration (no *a priori* LLR available) can be computed using (15).

We use a similar approach to find the BER of a linear SISO equalizer for the perfect *a priori* LLR case ($|L_i^E(x_n)| \rightarrow \infty, \forall n$). If \hat{x}_n is the equalizer output, we have

$$\hat{x}_n = \sum_{k=-L_1}^{L_2} h_k^F z_{n-k} - \sum_{k=-K_1}^{K_2} h_k^B \bar{x}_{n-k}. \quad (16)$$

We can express the BER as

$$P_e = \frac{1}{2M} \sum_{\forall \mathbf{x}_n^0, \bar{\mathbf{x}}_n^B} \Pr\left(|v_n - (\mathbf{h}^{eq})^T \mathbf{x}_n^0 - (\mathbf{h}^B)^T \bar{\mathbf{x}}_n^B| > h_0^{eq}\right) = \frac{1}{2M} \sum_{\forall j} \Pr(|v_n - \epsilon_j| > h_0^{eq}) \quad (17)$$

where $\bar{\mathbf{x}}_n^B = [x_{n+K_1}, \dots, x_n, \dots, x_{n-K_2}]$ is a vector of transmitted symbols since perfect *a priori* LLR is available to the SISO equalizer and $e_n = (\mathbf{h}^{eq})^T \mathbf{x}_n - (\mathbf{h}^B)^T \bar{\mathbf{x}}_n^B$. It should be noted that ϵ_j of (15) has different sample values from that of (17). Given a symbol alphabet size Q , the computation of

(15) and (17) needs $Q^{M_1+M_2}$ operations. Thus, as the lengths of the channel response and equalizer filter are increased, the complexity may be large. However, in many practical applications, most of the energy of the equalized channel response \mathbf{h}^{eq} will be concentrated around a single tap of \mathbf{h}^{eq} . Then, by considering only a few taps around the main cursor which contains, say, more than 95% energy of \mathbf{h}^{eq} , the complexity may be greatly reduced with a small loss in precision. More discussion on this complexity issue and simulations will be presented in Section IV to demonstrate this method.

Once the BER is computed using (15) for the first equalization pass and (17) for perfect *a priori* LLRs, we note that the BER transfer function can be assumed to be an increasing function of the input BER provided by the previous SISO decoder iteration, and, hence, the transfer function can be approximated via linear interpolation between the two end points as shown in Fig. 8, where it should be noted that both axes are shown in log scale. By projecting this interpolated transfer function onto an empirically determined decoder BER transfer function, the convergence behavior can be investigated. As shown in Fig. 8, where $\text{SNR} = \log_{10}(x_n^2/w_n^2)$, the decoder output BER at the convergence point is about 10^{-4} for $\text{SNR} = 3$ dB, which enables us to measure SNR required to achieve a given BER (simulation results will be demonstrated later). Further, we can conclude that at least three iterations are needed for reaching the convergence point. We note that due to the linear approximation, there is a small discrepancy between the conventional (simulation based) and the predicted (linearized) BER transfer functions, where offsets are most noticeable in the intermediate iterations. Thus, the predicted BERs at intermediate iterations may be optimistic. We note that the deviation from the conventional BER transfer function to the linearized one is relatively small and actually smaller than the deviation from the closed-loop iterations to the open loop trajectory computation. Therefore, most of this deviation arises from the inaccuracy of the Gaussian assumption of (9) on LLRs as opposed to the inaccuracy of the

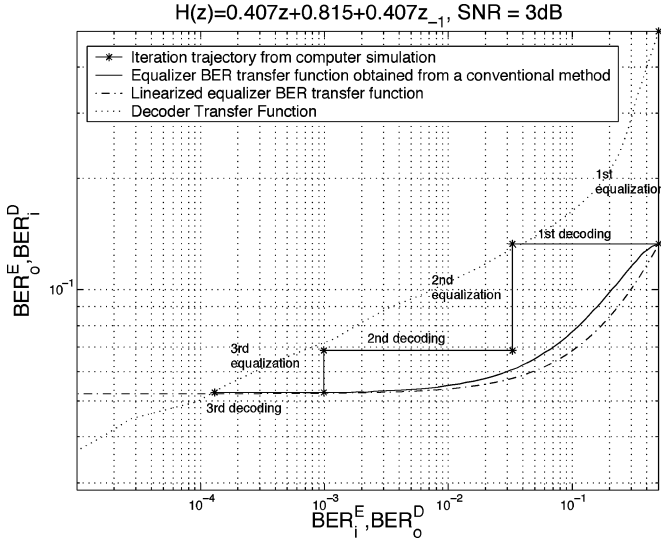


Fig. 8. For an RSC encoder with generator polynomial $(23, 35)_8$, an example linearized BER transfer chart is plotted using the derived BER formula and linear interpolation for a block size of 65536, where the superscripts E and D refer to equalization and decoding and the subscripts i and o denote input and output, respectively. Note that, prior to the first iteration, the BER given in the *a priori* LLR case is equal to 0.5.

linearization. Comparisons with simulation results are also provided in Section IV. In Fig. 8, note that while the BER is monotonically decreasing for each constituent device (equalizer/decoder), the equalizer output BER may be larger than the equalizer input BER (the decoder output BER at the previous iteration). This is not a problem since the improved decoder BER helps the equalizer to lower its next iteration output BER as shown in Fig. 8.

B. Mutual Information Computation

To investigate the evolution characteristics of the mutual information between x_n and $L_o^E(x_n)$, the two end points of EXIT functions, $I_i^E = 0$ ($L_i^E(x_n) = 0, \forall n$) and $I_i^E = 1$ ($|L_i^E(x_n)| \rightarrow \infty, \forall n$), respectively, are considered. As shown in Fig. 6, for the first iteration, the estimated symbol after equalization can be expressed as in (13). Thus, the equalized channel can be modeled as a scalar gain and scalar additive distortion as in Fig. 9. By approximating the LLR mapping as a functional form $L(\hat{x}_n) \triangleq (2\hat{x}_n/\sigma_n^2)$ as would be for the case of a memoryless channel with AWGN or in the case of perfect equalization [14], we can estimate the mutual information at these two end points. Since perfect equalization is approached as iterations proceed, the approximation improves. Using this mapping, the LLRs of the linear SISO equalizer outputs become

$$\begin{aligned} L_o^E(x_n) &= \ln \frac{f_{\hat{x}_n|x_n}(\hat{x}_n|x_n = +1)}{f_{\hat{x}_n|x_n}(\hat{x}_n|x_n = -1)} \\ &= \frac{2\hat{x}_n}{\sigma_v^2} \\ &= \frac{2(h_0^{eq}x_n + e_n + v_n)}{\sigma_v^2} \end{aligned} \quad (18)$$

where the LLR values are identical to those in the AWGN channel case if $h_0^{eq} = 1$, $e_n = 0$, or equivalently perfect equal-

ization. The PDF of L_o^E is assumed to also have a Gaussian distribution with mean and variance

$$\begin{aligned} \mu_{L_o^E} &= E\{L_o^E\} = \frac{2(h_0^{eq} \cdot x_n + e_n)}{\sigma_v^2} \\ \sigma_{L_o^E}^2 &= E\{(L_o^E)^2\} - \mu_{L_o^E}^2 = \frac{4}{\sigma_v^2} \end{aligned} \quad (19)$$

where L_o^E denotes the random variables whose samples are $L_o^E(x_n)$. Note that, as iterations proceed, (19) provides more consistent LLRs. This is because the variance becomes close to twice the mean as the conditions $h_0^{eq} \approx 1$ and $e_n \approx 0$ are achieved. However, in the low SNR or in the first few iterations, consistent LLRs are not generated by SISO equalizers. The assumed Gaussian distribution leads to the following approximation to the distribution of L_o^E conditioned on \mathbf{x}_n

$$f_{L_o^E}(l|\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma_{L_o^E}} \exp\left(-\frac{(l - \mu_{L_o^E})^2}{2\sigma_{L_o^E}^2}\right). \quad (20)$$

Since ϵ is the function of neighboring symbols of x_n and ϵ is independent of x_n , (20) can be written

$$f_{L_o^E}(l|x, \mathbf{x}^0) = f_{L_o^E}(l|x, \epsilon) = \frac{1}{\sqrt{2\pi}\sigma_{L_o^E}} \exp\left(-\frac{(l - \mu_{L_o^E})^2}{2\sigma_{L_o^E}^2}\right). \quad (21)$$

We desire the mutual information between x_n and the LLR $L_o^E(x_n)$

$$I(X; L) \triangleq \sum_{x \in \mathcal{B}_{-\infty}} \int f(x, l) \log \frac{f(x, l)}{\Pr(x)f(l)} dl \quad (22)$$

in terms of $f_{L_o^E}(l|x, \epsilon)$. To this end, we have

$$f(x, l) = \sum_{j=1}^M f(x, l, \epsilon_j) = \sum_{j=1}^M \frac{f(x, l, \epsilon_j)}{\Pr(x)} \Pr(x). \quad (23)$$

Since x and ϵ are independent, $\Pr(x = +1) = \Pr(x = -1) = 1/2$, and $\Pr(\epsilon = \epsilon_j) = (1/M) \forall j$, we can express $f(x, l)$ in terms of $f(l|x, \epsilon = \epsilon_j)$

$$f(x, l) = \frac{1}{M} \sum_{j=1}^M \frac{f(x, l, \epsilon = \epsilon_j)}{\Pr(\epsilon = \epsilon_j, x)} \Pr(x) = \frac{1}{2M} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j). \quad (24)$$

Using (24) in (22), we obtain

$$\begin{aligned} I(X; L) &= \sum_{x \in \mathcal{B}_{-\infty}} \int \left\{ \frac{1}{2M} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j) \right\} \\ &\quad \times \log \frac{\frac{1}{2M} \sum_{k=1}^M f(l|x, \epsilon = \epsilon_k)}{\Pr(x)f(l)} dl. \end{aligned} \quad (25)$$

The term $f(l)$ in (25) can be written as

$$\begin{aligned} f(l) &= \sum_{x \in \mathcal{B}} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j) \Pr(x, \epsilon = \epsilon_j) \\ &= \frac{1}{2M} \sum_{x \in \mathcal{B}} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j) \end{aligned} \quad (26)$$

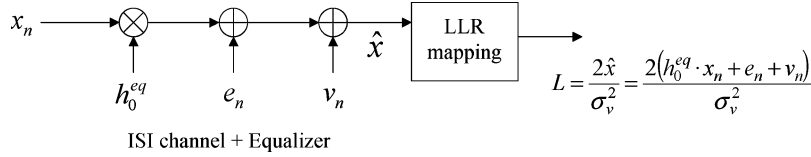


Fig. 9. Equalized channel model for computing the mutual information between the transmitted symbols and their LLRs, where h_0^{eq} is the dominant tap (decision point) in the equalized channel response and e_n comprises the ISI caused by neighboring symbols of x_n .

and, hence, (25) becomes

$$\begin{aligned}
 I(X;L) &= \sum_{x \in \mathcal{B}_{-\infty}} \int \left\{ \frac{1}{2M} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j) \right\} \\
 &\times \log \frac{\frac{1}{2M} \sum_{k=1}^M f(l|x, \epsilon = \epsilon_k)}{\frac{1}{4M} \sum_{x \in \mathcal{B}} \sum_{i=1}^M f(l|x, \epsilon = \epsilon_i)} dl \\
 &= \sum_{x_n \in \mathcal{B}_{-\infty}} \int \left\{ \frac{1}{2M} \sum_{j=1}^M f(l|x, \epsilon = \epsilon_j) \right\} \\
 &\times \log \frac{2 \sum_{k=1}^M f(l|x, \epsilon = \epsilon_k)}{\sum_{i=1}^M \{f(l|x = -1, \epsilon = \epsilon_i) + f(l|x = +1, \epsilon = \epsilon_i)\}} dl \quad (27)
 \end{aligned}$$

which can be used to compute the mutual information at one end point, i.e., at the *a priori* LLR point. In order to compute the mutual information at the other end point (for perfect *a priori* LLRs), \bar{x}_n is equal to x_n . Therefore, by replacing $H_{eq}(e^{j\omega})$ of Fig. 6 by $H(e^{j\omega})H^F(e^{j\omega}) - H^B(e^{j\omega})$ of Fig. 7, the mutual information can also be computed using (27). Note that the ISI samples ϵ_j are different at these two end points (in case of no *a priori* LLR, $e_n = (\mathbf{h}^{eq0})^T \cdot \mathbf{x}_n$, and, in the other case, $e_n = (\mathbf{h}^{eq})^T \mathbf{x}_n - (\mathbf{h}^B)^T \bar{\mathbf{x}}_n^B$) because the overall channel responses after equalization are different. In order to evaluate (27), $Q^{M_1+M_2}$ operations are required for an alphabet size of Q . As mentioned in Section III-A, by considering only a few taps containing, say, more than 95% energy of \mathbf{h}^{eq} , the computational complexity may be reduced with negligible error, as will be demonstrated in Section IV.

Once the mutual information I_o^E is computed at the two end points ($I_i^E = 0$ and $I_i^E = 1$), we note that the EXIT functions for linear SISO equalizers appear to be well modeled as linear [8], [15], [19], and, hence, the equalizer EXIT function can be interpolated linearly between these two values. This estimated equalizer EXIT function is projected over the decoder EXIT function, which is determined numerically given the encoder polynomial [14]. Hence, given channel knowledge, linear equalizer coefficients, and the empirically determined decoder EXIT function, the convergence behavior of a linear turbo equalizer can be investigated without the need for extensive simulations of the equalizer. By tracing the equalizer and decoder EXIT functions as explained in Section II-B, we can see whether the turbo equalization procedure will reach the desired operating point or not. For example, as shown in Fig. 10, where the desired operating point is assumed to be $I_o^D = 0.99$, the iterative procedure is shown to reach the desired point with three iterations, where simulation and analysis results are close but do not match

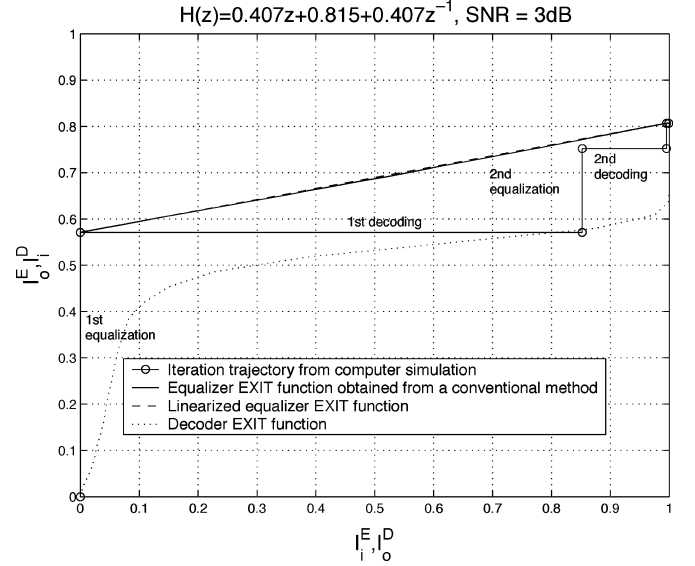


Fig. 10. For an RSC encoder with generator polynomial $(23, 35)_8$, an example EXIT chart is plotted using the proposed analysis method for a block size of 65 536.

perfectly at the second iteration (the BER transfer chart in Section III-A has the same phenomenon) due to the linearization errors.

C. Discussion

For purposes of discussion, we considered three static channel models (channel **A**, **B**, and **C**) throughout this paper

$$\mathbf{H}_A(z) = 0.04z^5 - 0.05z^4 + 0.07z^3 - 0.21z^2 - 0.5z + 0.72 + 0.36z^{-1} + 0.21z^{-3} + 0.03z^{-4} + 0.07z^{-5}$$

$$\mathbf{H}_B(z) = 0.407z + 0.815 + 0.407z^{-1}$$

$$\mathbf{H}_C(z) = 0.227z^2 + 0.46z + 0.688 + 0.46z^{-1} + 0.227z^{-2}.$$

$\mathbf{H}_A(z)$ is a “good” channel with only mild ISI, $\mathbf{H}_B(z)$ has “medium” ISI (strong spectral null near $\omega = \pi$), and $\mathbf{H}_C(z)$ has “severe” ISI (strong spectral null near $\omega = 0.6\pi$) [20]. An RSC encoder with generator polynomial $(23, 35)_8$ and a random interleaver with $N_c = 65\,536$ are considered.

Fig. 11 shows linearized equalizer BER transfer and EXIT functions for three different channels. We note that the slope of the two linearized BER transfer and EXIT functions of a linear SISO equalizer provides a measure of potential improvement via iterative turbo equalization in terms of BER and mutual information. When the slope of the transfer function is steep, there is more room for improvement as iterations proceed (e.g., channels **B** and **C**). This is because more reliable input *a priori* LLRs $L_i^E(x_n)$ (greater I_i^E value and smaller BER_i^E) produces more reliable output *a posteriori* LLRs $L_o^E(x_n)$ (greater I_o^E value and

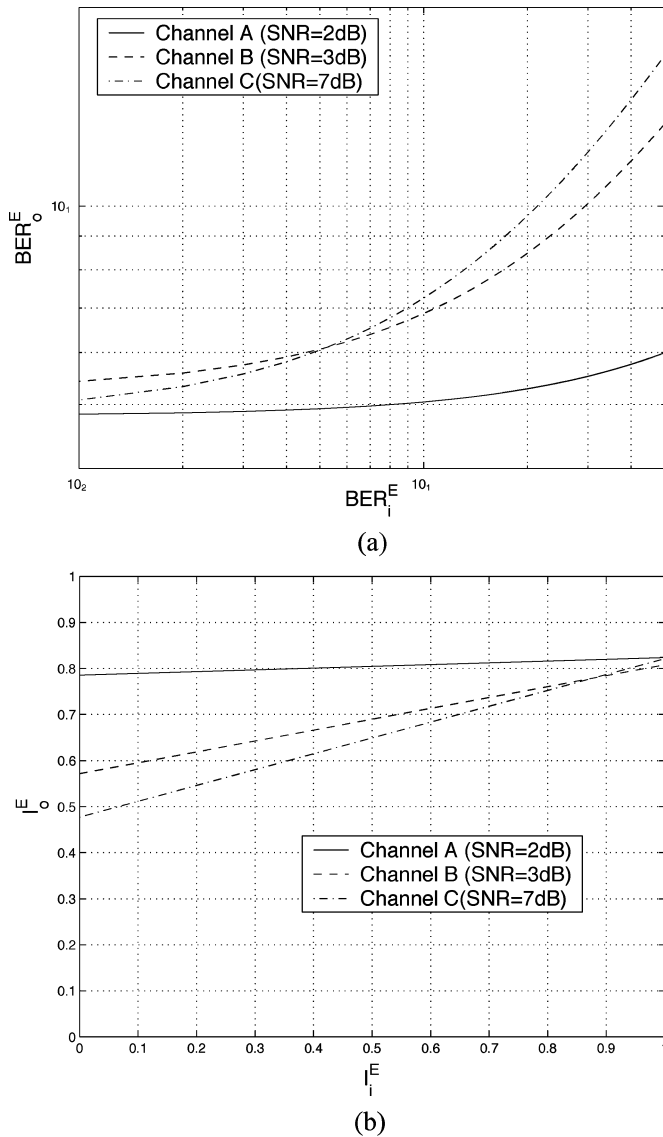


Fig. 11. Interpolated transfer functions for the linear SISO equalizer for each of the three channels, where MMSE-NA equalizers [8] are employed. The slopes of these transfer functions determine the utility of the iterative algorithm. (a) Linearized BER transfer function. (b) Linearized EXIT function.

smaller BER_o^E). However, for a “flat” equalizer EXIT function (channel **A**), better input *a priori* LLRs make little difference in generating output posteriori LLRs. Thus, simply by estimating two end points of these BER transfer and EXIT charts, we can estimate the potential effectiveness of a turbo equalizer for a given channel.

For channels **B** and **C**, more detailed analyses are carried out. First, Fig. 12(a) shows linearized BER transfer functions for channel **B** for different SNR values. By observing the point at which the equalizer and decoder transfer functions cross, the equalizer or decoder BERs after convergence can be predicted. As shown in Fig. 12(b), we see that the decoder BER gradually decreases as the channel SNR is increased. Note that this behavior is different from that of iterative decoding of turbo codes, where the BER performance curve has a “waterfall” shape after a certain SNR threshold is reached [1], [14]. In turbo decoding of parallel concatenated codes, two decoder EXIT functions are

jointly plotted in an EXIT chart, and, hence, if the SNR is greater than a certain threshold, the output BER sharply drops and creates a waterfall region since the tunnel between those two decoder EXIT functions becomes sufficiently wide and the convergence point reaches $I_o^D = 1$ (as shown in [14, Fig. 6]).

However, in the case of turbo equalization, the slopes of the equalizer BER transfer and EXIT functions are not as steep as that of the decoder transfer function, leading to noticeably different behavior as a function of SNR. For example, there may or may not be a similar waterfall region depending upon the channel impulse response. As such, the notion of an SNR threshold may not be appropriate for turbo equalization. However, SNR required for achieving a given BER may be more appropriate measure. As shown in Fig. 12(a), the decoder function sweeps from 0.5 to 10^{-6} , but the equalizer falls from 0.2 to 0.02. Thus, if these two different characteristics are jointly considered in a BER transfer or EXIT chart as shown in Fig. 12, the BER will only gradually improve as SNR is increased.

The simulated BER performance supports this predicted BER behavior as shown in Fig. 12(b), with some slight differences in the low SNR region. This mismatch may be caused by deviations in the PDF of $L_o^E(x_n)$ from that assumed while the decoder BER transfer function is empirically obtained under the Gaussian assumption of (9). Fig. 12(c) compares two cumulative density functions when $x_n = 1$. The solid line refers to the CDF numerically obtained from computer simulations. The dashed line refers to the CDF of a Gaussian distribution given the mean of L_o^E . As shown in Fig. 12(c), there are mismatches between the assumed and simulated CDF, which may contribute to the prediction errors observed in the low SNR region of Fig. 12(b). The linearized EXIT functions are also presented in Fig. 12(d). Note that the mutual information (which is a function of BER) after convergence changes rapidly at around 0 dB and then is gradually improved for $SNR > 0$ dB, which is supported by Fig. 12(b). Although there is a long tunnel between the equalizer and decoder EXIT functions for $SNR > 0$ dB, the points at convergence does not approach $I_o^D = 1$, as observed in turbo decoding [14]. Hence, the BER is only gradually improved as shown in Fig. 12(b). Note that the linearized EXIT analysis approaches provide less insight into behavior near convergence than the linearized BER transfer chart does since the linearized BER transfer chart can identify the resulting BER at convergence directly. Using a mapping from mutual information to BER as in [8] may provide a similar, though less intuitive, approach.

For channel **C**, Fig. 13(c) shows the linearized EXIT functions for different SNR values. We can predict three distinct regions of BER performance. When $SNR < 4$ dB, the linearized EXIT chart converges at a small value of I_o^D , which implies high BER. When $4 \text{ dB} < SNR < 5$ dB, a narrow tunnel guides the convergence point to approach $I_o^D = 1$, which implies BER is improved sharply, creating a waterfall region as in the turbo decoding case. For $SNR > 5$ dB, the BER is improved gradually again as the SNR is increased. This analysis is well in line with the simulated BER curve shown in Fig. 13(b), where for $5 \text{ dB} < SNR < 6$ dB, the BER drops sharply. This behavior is different from that observed for channel **B** since the slope of an equalizer transfer function is different. The predicted BER

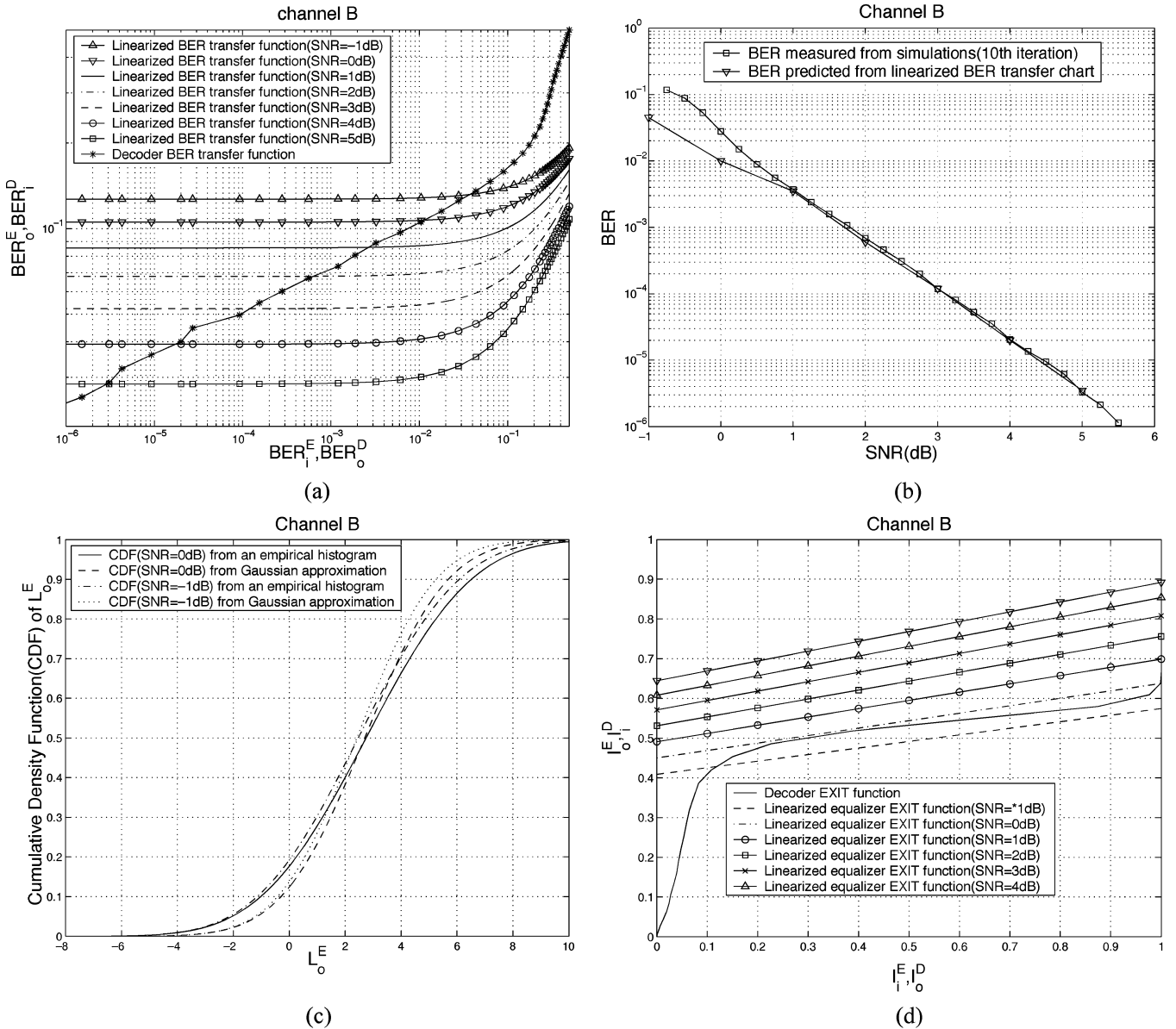


Fig. 12. Application of the linearized BER transfer and EXIT chart analyses to channel B. (a) Linearized BER transfer chart analysis. (b) Comparison of decoder BER. (c) Cumulative density function (CDF). (d) Linearized EXIT chart analysis.

performance after convergence can also be predicted from the linearized BER chart shown in Fig. 13(a) and is plotted along with simulated BER in Fig. 13(b), where mismatches are again noticeable in the low SNR region. Since these linearized transfer function analyses are asymptotic (in that the independence assumptions on LLRs become valid for an infinite-length ideal interleaver [8], [14]) and approximate [in that L_o is assumed to follow (9)], we expect such deviations for finite block sizes in practice.

We observe that similar to SNR threshold behavior for turbo codes, SNR requirements can be estimated through these linearized BER transfer and EXIT charts, such that given a desired decoder (equalizer) BER, the required channel SNR to obtain it can be determined. For example, if the desired decoder BER after convergence is 10^{-5} , the required SNR for channel C would be 9 dB as shown in Fig. 13(a), which is verified in simulation results [Fig. 13(b)].

IV. APPLICATIONS

In this section, the efficacy of the linearized BER transfer and EXIT chart analysis methods are presented and results of their application to specific scenarios are verified through computer simulations. For computer simulations, we employ an RSC encoder at the transmitter with generator polynomial $(23, 35)_8$. The coded bit stream is first passed through a random interleaver followed by a BPSK modulation. We use random interleavers ($N_c = 65\,536$) and the two SISO equalization algorithms—MMSE-NA and MMSE-MF—which are based on the MMSE criterion and are presented in [8], are employed to determine the linear equalizer coefficients. The MMSE-NA method computes the MMSE set of coefficients assuming no *a priori* (NA) LLRs while the MMSE-MF method minimizes the MSE assuming perfect *a priori* LLRs in which the coefficients approach to those of a matched filter (MF). The log-domain MAP

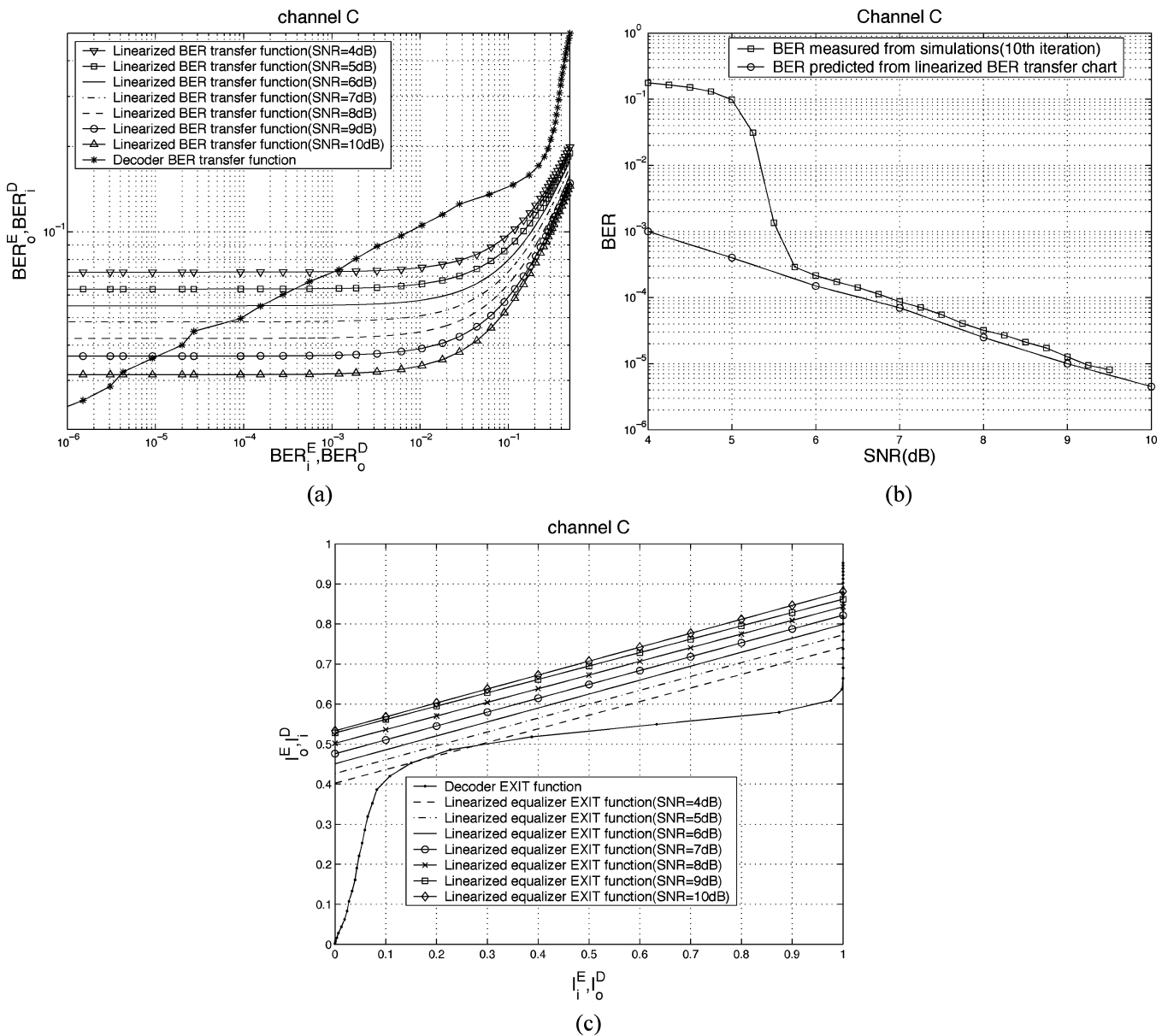


Fig. 13. Application of the linearized BER transfer and EXIT chart analyses to channel C. (a) Linearized BER transfer chart analysis. (b) Comparison of decoder BERs. (c) Linearized EXIT chart analysis.

decoder, presented in [28], is employed and ten iterations are carried out.

The number of taps used in the feedforward and feedback filters for given channel conditions are often determined empirically in a brute-force manner, since no method for optimal filter length (for most operating conditions) are known other than exhaustive search. However, in this paper, the linearized BER transfer and EXIT chart methods are applied to aid the search for such an optimal number of taps such that the mutual information measured from the equalizer output LLRs is maximized, or the BER measured from the equalizer outputs is minimized. Fig. 14(a) shows the output mutual information versus the number of taps in the feedforward and feedback filters of MMSE-NA equalization for channel B. Thus, we can determine the expected marginal benefit of additional filter taps by observing the improvement in the associated mutual information, which leads to a reduction in BER. Although it is not

shown here, the BER transfer functions can be employed for these applications. By following this approach, the number of taps determined in the feedforward path and the feedback path are summarized in Table I. Fig. 14(b) shows the BER measured from computer simulations for different numbers of feedforward and feedback filters, N_F and N_B , respectively. As predicted in Fig. 14(a), $N_F = 7$ and $N_B = 5$ show the best BER for channel B and the simulated BER well supports this predictive analysis.

The measured BER and mutual information are compared in Table II, where an MMSE-NA SISO equalizer is employed [8]. The BER and mutual information at the two end points are computed using the derived formulae (presented in Section III-A and III-B). We see that our BER analysis results match the simulation results well at the two end points. Hence, by computing these two values, the potential iterative processing gain can be determined in terms of BER and mutual information, without re-

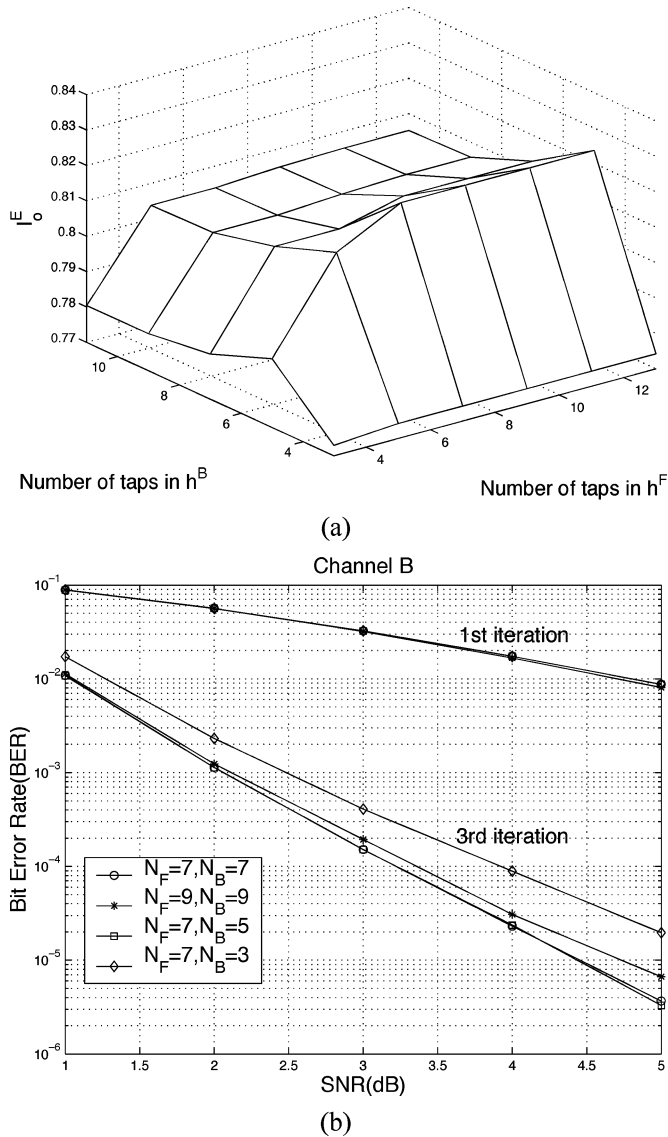


Fig. 14. Determination of the number of taps. (a) The number of taps in the feedforward and feedback filters of an MMSE-NA equalizer are shown versus the output mutual information after convergence and (b) decoder BER obtained from computer simulations from different numbers of filter taps.

TABLE I
NUMBER OF TAPS IN SISO EQUALIZATION

channel	Feedforward filter(N_F)	Feedback filter(N_B)
A	11	11
B	7	5
C	9	7

liance on extensive computer simulations of the equalizer. Furthermore, it is shown in Table II that if the slope is steep [as in $\mathbf{H}_B(z)$ and $\mathbf{H}_C(z)$], BER performance improves with iteration. However, when the slope is flat [as in $\mathbf{H}_A(z)$], more reliable feedback information from the SISO decoder makes little difference in soft outputs. Hence, after one or two iterations, BER improvement ceases as shown in Table II. For example, it would be concluded that turbo equalization is more useful for channels B and C. This is in line with the results in Section III-C

(Fig. 11). We also estimated the BER and mutual information by considering a truncated impulse response, containing about 99% energy of \mathbf{h}^{eq} , and compared the results in Table III. With less than 4% error, we can achieve an additional computational complexity reduction as shown in Table IV, where 20 sample points in the X axis of an EXIT chart and 10^7 coded bits are considered for an empirical method [8]. As mentioned in Section II-B, the linearized BER transfer and EXIT analyses save computations.

The linearized BER transfer charts can be also employed to estimate the BER that would be obtained during intermediate iterations. As summarized in Table II, the estimated BERs are close to these obtained through simulations but are underestimated due to the linear approximation. The mismatches are more noticeable for channel $\mathbf{H}_C(z)$, since the linear interpolation appears to predict BER transfer characteristics optimistically. Further, the required number of iterations for convergence, where the BER is not significantly further improved, can be also predicted. Table V compares simulation results with these predicted.

Fig. 15 shows two different SISO equalizer EXIT functions (MMSE-NA and MMSE-MF presented in [8]) over the channel C. For small I_i^E (unreliable soft information), MMSE-NA equalizers produce more reliable output LLRs, but MMSE-MF produces more reliable output LLRs near convergence. These results are in line with assumptions made in computing approximate MMSE solutions in [8]. The MMSE-NA equalizer ignores $L_i^E(x_n)$ by setting $L_i^E(x_n) = 0, \forall n$, and, hence, performs better in the unreliable *a priori* LLR region by de-emphasizing $L_i^E(x_n)$. The MMSE-MF assumed $L_i^E(x_n)$ to be perfect and, hence, produces more reliable $L_o^E(x_n)$ near $I_o^E = 1$, but in the unreliable region, poor $L_i^E(x_n)$ degrades the quality of $L_o^E(x_n)$ more than in MMSE-NA as shown in Fig. 15. Thus, the iterative behavior of these two equalization algorithms can be compared without reliance on simulations. Further, these different evolution characteristics motivated an adaptive turbo equalization approach, where the choice of the equalization algorithm (where MMSE-NA and MMSE-MF are available) is adapted to the state of the iterative procedure [8], [29]–[31]. Our linearized EXIT chart analysis can be used to predict the switching point, where the two equalizer EXIT functions cross [31]. Thus, in Fig. 15, switching from MMSE-NA to MMSE-MF is expected after the third iteration. With this predicted switching point, the resulting BER for “switching turbo equalization” [29]–[31] is demonstrated in Fig. 16, where BF denote a brute-force switch method and the number in the parenthesis is the number of executed iterations. In a BF switching, the switch to MMSE-MF occurs after turbo equalization with MMSE-NA has converged. As shown in Fig. 16, the switching turbo equalization with the predicted switching point demonstrates the best BER after the sixth iteration. After the tenth iteration, in the middle of SNR range, the BF method shows the better BER, but for high SNR values, the switching method predicted from our analysis show the best BER even with six iterations. Hence, the predicted switching scheme shows better BERs than MMSE-NA and MMSE-MF turbo equalization and faster convergence than a brute-force scheme.

TABLE II
COMPARISON BETWEEN SIMULATION AND ANALYSIS, WHERE BER AND MUTUAL INFORMATION ARE MEASURED FROM SISO EQUALIZER OUTPUTS

A	BER Transfer Chart								EXIT Chart			
	Simulation				Analysis				Simulation		Analysis	
SNR	1st	2nd	3rd	10th	$BER_i^E = 0.5$	2nd	3rd	$BER_i^E = 0$	1st	10th	$I_i^E = 0$	$I_i^E = 1$
1 dB	0.0795	0.0668	0.0668	0.0668	0.0787	0.0658	0.0658	0.0658	0.718	0.759	0.721	0.763
2 dB	0.0599	0.0481	0.0481	0.0481	0.0591	0.0473	0.0473	0.0473	0.782	0.822	0.785	0.825
3 dB	0.0433	0.0329	0.0329	0.0329	0.0424	0.0322	0.0322	0.0322	0.838	0.874	0.843	0.879
4 dB	0.0296	0.0212	0.0212	0.0212	0.0288	0.0205	0.0205	0.0205	0.886	0.916	0.891	0.921
B	Simulation				Analysis				Simulation		Analysis	
SNR	1st	2nd	3rd	10th	$BER_i^E = 0.5$	2nd	3rd	$BER_i^E = 0$	1st	10th	$I_i^E = 0$	$I_i^E = 1$
1 dB	0.1612	0.1243	0.0966	0.0871	0.161	0.1125	0.092	0.0856	0.490	0.693	0.491	0.699
2 dB	0.1467	0.0939	0.0709	0.0683	0.1464	0.0875	0.069	0.0679	0.531	0.754	0.531	0.756
3 dB	0.1330	0.0678	0.0531	0.0526	0.1326	0.0638	0.0523	0.0523	0.57	0.806	0.571	0.808
4 dB	0.1201	0.0476	0.0395	0.0395	0.1197	0.0442	0.0392	0.0392	0.607	0.851	0.608	0.854
C	Simulation				Analysis				Simulation		Analysis	
SNR	1st	2nd	3rd	10th	$BER_i^E = 0.5$	2nd	3rd	$BER_i^E = 0$	1st	10th	$I_i^E = 0$	$I_i^E = 1$
6 dB	0.1791	0.1549	0.1102	0.0555	0.1791	0.1025	0.0625	0.0550	0.451	0.796	0.451	0.799
7 dB	0.1689	0.1291	0.0755	0.0484	0.1692	0.0875	0.0495	0.0483	0.476	0.821	0.476	0.822
8 dB	0.1586	0.1039	0.0515	0.0424	0.1587	0.0719	0.0435	0.0422	0.502	0.842	0.502	0.843
9 dB	0.1482	0.0809	0.0390	0.0368	0.1483	0.0605	0.0366	0.0366	0.528	0.862	0.528	0.863

TABLE III
APPROXIMATED BER AND MUTUAL INFORMATION COMPUTATION FOR CHANNEL C

SNR	BER		Mutual Information	
	100%	$\geq 95\%$	100%	$\geq 95\%$
6 dB	0.0550	0.0532	0.799	0.806
7 dB	0.0483	0.0464	0.822	0.829
8 dB	0.0422	0.0405	0.843	0.849
9 dB	0.0366	0.0351	0.863	0.868

TABLE IV
COMPUTATIONAL COMPLEXITY COMPARISON WITH AN EXPERIMENT SETUP OF [8]

Channel	Direct Computation	Approximation	Conventional Method [8]
B	$O(2^9)$	$O(2^5)$	$O(2^{31})$
C	$O(2^{13})$	$O(2^7)$	$O(2^{32})$

TABLE V
NUMBER OF ITERATIONS FOR CONVERGENCE (CHANNEL B)

SNR	Simulated results	Predicted results
1 dB	5	4
2 dB	4	4
3 dB	3	3
4 dB	3	3

V. SUMMARY AND CONCLUSION

Linearized BER transfer and EXIT function estimation methods are proposed to investigate the iterative behavior of a linear turbo equalization system with dramatically reduced

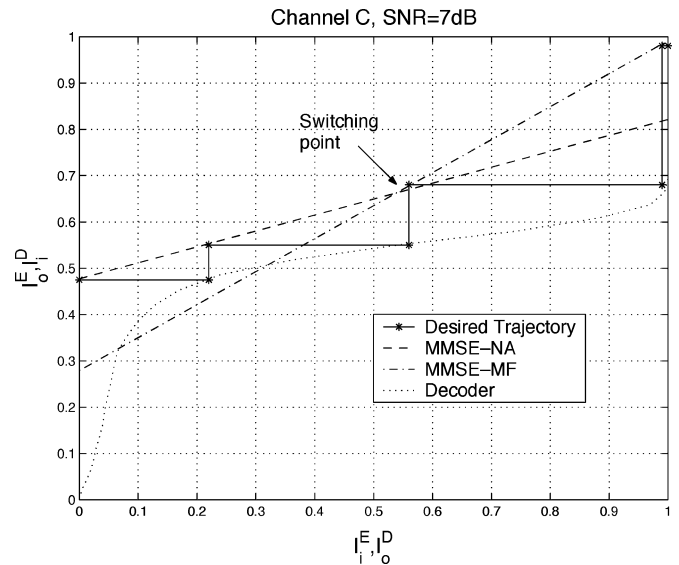


Fig. 15. Comparison of different SISO equalizer algorithms (MMSE-NA and MMSE-MF), where MMSE-NA and MMSE-MF transfer functions are obtained via linear interpolation.

complexity in comparison with conventional methods for generating such EXIT functions. Estimation of the BER evolution of the symbol estimates is accomplished by linearizing the BER transfer characteristics. Similarly, evolution of the mutual information between the transmitted symbols and their LLRs can be estimated by approximating the EXIT functions as linear. These linearized BER transfer and EXIT function methods enable us to determine the efficacy of turbo equalization systems for a given application, such as a given channel impulse response and SNR regime and to predict the iterative BER behavior and the

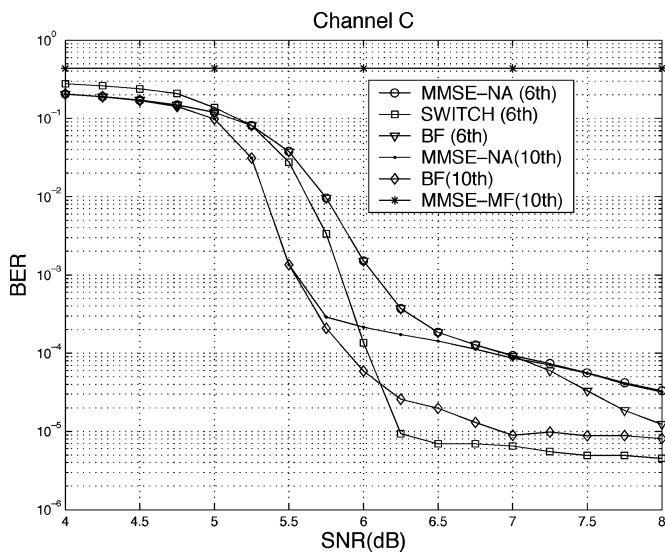


Fig. 16. Switching equalization BER.

SNR required to achieve a desired BER. Some applications are demonstrated and verified through computer simulations. These linearized methods may serve to aid a communication system designer to appropriately choose equalization parameters in implementing turbo equalization systems without reliance on extensive computer simulations.

REFERENCES

- [1] C. Berrou, P. Adde, E. Angui, and S. Faudeil, "Near Shannon limit error-correcting coding and decoding: turbo-codes," in *Proc. IEEE Int. Conf. Communication*, Geneva, Switzerland, May 1993, pp. 1064–1070.
- [2] P. H. Siegel, D. Divsalar, E. Eleftheriou, J. Hagenauer, and D. Rowitch, "Guest editorial the turbo principle: from theory to practice," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 793–797, May 2001.
- [3] —, "Guest editorial the turbo principle: From theory to practice II," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 9, pp. 1657–1661, Sep. 2001.
- [4] J. Hagenauer, "Turbo principle: tutorial introduction and state of the art," in *Proc. Int. Symp. Turbo Codes and Related Topics*, Bretagne, France, 1997, pp. 1–11.
- [5] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: turbo-equalization," *Eur. Trans. Telecommun.*, vol. 6, pp. 507–511, 1995.
- [6] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes Related Topics*, Sep. 1997, pp. 96–102.
- [7] C. Laot, A. Glavieux, and J. Labat, "Turbo equalization: adaptive equalization and channel decoding jointly optimized," *IEEE J. Sel. Areas Commun.*, vol. 19, pp. 1744–1752, Sep. 2001.
- [8] M. Tuchler, R. Koetter, and A. Singer, "Turbo-equalization: principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.
- [9] M. Tuchler, A. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 673–683, Mar. 2002.
- [10] Z. Wu and J. M. Cioffi, "Turbo decision aided equalization for magnetic recording channels," in *Proc. IEEE Global Telecommunication Conf.*, 1999, pp. 733–738.
- [11] A. Roumy, I. Fijalkow, and D. Pirez, "Joint equalization and decoding: why choose the iterative solution?," in *Proc. IEEE Vehicular Technology Conf.*, vol. 5, 1999, pp. 733–738.
- [12] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.
- [13] R. Koetter, A. C. Singer, and M. Tuchler, "Turbo equalization," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 67–80, Jan. 2004.
- [14] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 1999.
- [15] M. Tuchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proc. 4th Int. ITG Conf. Source and Channel Coding*, Berlin, Germany, Jan. 2002, pp. 53–60.
- [16] A. Roumy, A. J. Grant, I. Fijalkow, P. D. Alexander, and D. Pirez, "Turbo-equalization: convergence analysis," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, 2001, pp. 2645–2648.
- [17] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 891–907, May 2001.
- [18] H. E. Gamal and A. R. Hammons, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.
- [19] J. Nelson, A. Singer, and R. Koetter, "Linear turbo equalization for parallel ISI channels," *IEEE Trans. Commun.*, vol. 51, no. 6, pp. 860–864, Jun. 2003.
- [20] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
- [21] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. IEEE Int. Conf. Communication*, 1995, pp. 1009–1013.
- [22] A. Matache, S. Dolinar, and F. Pollara, "Stopping Rules for Turbo Decoders," Jet Propulsion Lab., Pasadena, CA, TDA Progress Rep. 42-142, Aug. 2000.
- [23] R. Y. Shao, S. Lin, and M. P. C. Fossorier, "Two simple stopping criteria for turbo decoding," *IEEE Trans. Commun.*, vol. 47, no. 8, pp. 1117–1120, Aug. 1999.
- [24] S. Lee, N. Shanbhag, and A. Singer, "Low-power turbo equalizer architecture," in *Proc. IEEE Signal Processing Systems(SiPS): Design and Implementation*, San Diego, CA, Oct. 2002, pp. 33–38.
- [25] G. Bauch, H. Khorrarn, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proc. Eur. Personal Mobile Communication Conf.*, Oct. 1997.
- [26] I. Fijalkow, A. Roumy, S. Ronger, D. Pirez, and P. Vila, "Improved interference cancellation for turbo-equalization," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, 2000, pp. 416–419.
- [27] D. Raphaeli and A. Sauly, "Reduced complexity APP for turbo equalization," in *Proc. IEEE Int. Conf. Communication*, May 2002, pp. 1940–1943.
- [28] A. J. Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 260–264, Feb. 1998.
- [29] M. Tuchler, R. Koetter, and A. Singer, "Hybrid equalization strategies for iterative equalization and decoding," presented at the Int. Symp. Information Theory, 2001, p. 267.
- [30] S. Lee, A. C. Singer, and N. R. Shanbhag, "Switching LMS linear turbo equalization," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, May 2004, pp. 641–644.
- [31] S. Lee, N. R. Shanbhag, and A. C. Singer, "Switching methods for linear turbo equalization," in *Proc. IEEE Int. Symp. Circuits Systems*, vol. 3, May 2004, pp. 601–604.



Seok-Jun Lee was born in Pusan, Korea, in March, 1973. He received the B.S. and M.S. degrees from the School of Electrical Engineering, Seoul National University, Seoul, Korea, in 1996 and 1998, respectively.

From August 2000 to August 2004, he was in the graduate program of the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, where he was a Research Assistant at the Coordinated Science Laboratory. During the summers of 2001 and 2002,

he was with Texas Instruments, Inc., Dallas, TX, working in the Communication Systems Laboratory, DSPS R&D Center, where he was involved in IEEE 802.11 and 802.16 modem development. Since September 2004, he has been a System Engineer at the Communication Systems Laboratory, DSPS R&D Center, Texas Instruments, Inc. His research interests include VLSI architectures for communication and signal processing systems.

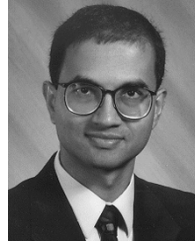


Andrew C. Singer (M'96) was born in Akron, OH, in 1967. He received the S.B., S.M., and Ph.D. degrees, all in electrical engineering and computer science, from the Massachusetts Institute of Technology (MIT), Cambridge, in 1990, 1992, and 1996, respectively.

Since 1998, he has been with the faculty of the Department of Electrical and Computer Engineering (ECE), University of Illinois, Urbana-Champaign, where he is a Willett Faculty Scholar and an Associate Professor with ECE and a Research Associate

Professor with the Coordinated Science Laboratory. In 1996, he was a Post-doctoral Research Affiliate with the Research Laboratory of Electronics, MIT. From 1996 to 1998, he was a Research Scientist with Sanders, A Lockheed Martin Company, Manchester, NH. His research interests include statistical signal processing and communication, universal prediction and data compression, and machine learning.

Dr. Singer was a Hughes Aircraft Masters Fellow and was the recipient of the Harold L. Hazen Memorial Award for excellence in teaching in 1991. In 2000, he received the National Science Foundation CAREER Award, and in 2001, he received the Xerox Faculty Research Award. He is currently a member of the MIT Educational Council, Eta Kappa Nu, and Tau Beta Pi.



Naresh R. Shanbhag (M'93–SM'03) received the B. Tech. degree from the Indian Institute of Technology, New Delhi, in 1988, the M.S. degree from the Wright State University, Dayton, OH, in 1990, and the Ph.D. degree from the University of Minnesota, Minneapolis, in 1993, all in electrical engineering.

From July 1993 to August 1995, he was with AT&T Bell Laboratories, Murray Hill, NJ, where he was responsible for the development of VLSI algorithms, architectures, and implementation of broadband data communications transceivers. In

particular, he was the lead chip architect for AT&T's 51.84 Mb/s transceiver chips over twisted-pair wiring for asynchronous transfer mode (ATM)-LAN and broadband access chip sets. Since August 1995, he has been with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, where he is presently a Professor and the Director of the Illinois Center for Integrated Microsystems. At the University of Illinois, he founded the VLSI Information Processing Systems (ViPS) Group, whose charter is to explore issues related to low-power, high-performance, and reliable integrated circuit implementations of broadband communications and digital signal processing systems. He has published more than 90 journal articles/book chapters/conference publications in this area and holds three U.S. patents. He is also a coauthor of the research monograph entitled "Pipelined Adaptive Digital Filters" (Norwell, MA: Kluwer, 1994).

Dr. Shanbhag received the 2001 IEEE Transactions on VLSI Systems Best Paper Award, the 1999 IEEE Leon K. Kirchmayer Best Paper Award, the 1999 Xerox Faculty Award, the National Science Foundation CAREER Award in 1996, and the 1994 Darlington Best Paper Award from the IEEE Circuits and Systems Society. From July 1997 to 2001, he was a Distinguished Lecturer for the IEEE Circuits and Systems Society. From 1997 to 1999, he served as an Associate Editor for the IEEE TRANSACTION ON CIRCUITS AND SYSTEMS: PART II. He is presently an Associate Editor for the IEEE TRANSACTIONS ON VLSI SYSTEMS. He has served on the technical program committee of various international conferences. He was the Technical Program Chair of the 2002 IEEE Workshop on Signal Processing Systems.