Turbo Equalization: Principles and New Results

Michael Tüchler, Ralf Koetter, and Andrew C. Singer

***Abstract*—We study the turbo equalization approach to coded data transmission over channels with intersymbol interference. In the original system invented by Douillard *et al.*, the data are protected by a convolutional code and the receiver consists of two trellis-based detectors, one for the channel (the equalizer) and one for the code (the decoder). It has been shown that iterating equalization and decoding tasks can yield tremendous improvements in bit error rate. We introduce new approaches to combining equalization based on linear filtering with decoding. Through simulation and analytical results, we show that the performance of the new approaches is similar to the trellis-based receiver, while providing large savings in computational complexity. Moreover, this paper provides an overview of the design alternatives for turbo equalization with given system parameters, such as the channel response or the signal-to-noise ratio.**

***Index Terms*—Decoding, equalization, intersymbol interference, iterative methods, turbo equalization.**

I. Introduction

MANY practical communication systems encounter the problem of data transmission over a channel with intersymbol interference (ISI). To protect the integrity of the data to be transmitted, a controlled amount of redundancy is added (encoding) using an error correction code (ECC). In this paper, we assume a coherent, symbol-spaced receiver front-end and precise knowledge of the signal phase and symbol timing, such that the channel can be approximated by an equivalent, discrete-time, baseband model, as shown in Fig. 1, where the transmit filter, the channel, and the receive filter are represented by a discrete-time *linear* filter, with the finite-length impulse response (FIR)

$$h[n] = \sum_{k=0}^{M-1} h_k \delta[n-k] \tag{1}$$

of length M. The coefficients h_k are assumed to be time-invariant and known to the receiver.

In a typical implementation of the system of Fig. 1, the received symbols are processed with an equalizer or detector to combat the effects of ISI. The equalizer produces estimates of the data and, for complexity reasons, typically consists of linear processing of the received signal [linear equalizer (LE)] and possibly past symbol estimates [decision feedback equalizer (DFE)] [1], [2]. The parameters of these filters can be selected using a variety of optimization criteria, such as zero forcing (ZF) or minimum mean squared error (MMSE) criteria [1], [2]. Optimal equalization methods for minimizing the bit error rate (BER) and the sequence error rate are nonlinear and are based on maximum-likelihood (ML) estimation, which turns into maximum *a posteriori* probability (MAP) estimation in the presence of *a priori* information about the transmitted data. Efficient algorithms exist for MAP/ML sequence estimation, e.g., the Viterbi algorithm (VA) [1], [3], [4], and MAP/ML symbol estimation, e.g., the BCJR algorithm [5]. We will refer to these estimation methods as MAP/ML equalization.

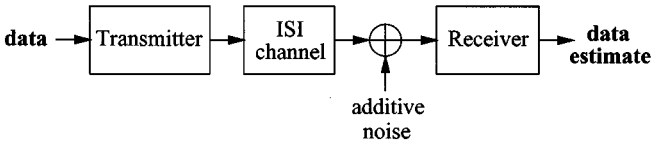

Fig. 1. Representation of a data transmission system.

Significant improvements in BER performance are possible with coded data transmission using an ECC. Communicating soft information between the equalizer and the decoder, instead of hard information (symbol estimates only), improves the BER performance but usually requires more complex decoding algorithms. State-of-the-art systems for a variety of communication channels employ convolutional codes and ML equalizers together with an interleaver after the encoder and a deinterleaver before the decoder [6], [7]. Interleaving shuffles symbols within a given time frame or block of data and thus decorrelates error events introduced by the equalizer between neighboring symbols. These error "bursts" are hard to deal with using a convolutional decoder alone. Some applications exploit coding to overcome deficiencies of the chosen equalizer, e.g., the use of a DFE and a high-rate code to deal with the effects of error propagation [8].

In the receiver, an optimal joint processing of the equalization and decoding steps is usually impossible due to complexity considerations. A number of iterative receiver algorithms repeat the equalization and decoding tasks on the same set of received data, where feedback information from the decoder is incorporated into the equalization process. This method, called turbo equalization, was originally developed for concatenated convolutional codes (turbo coding, [9]) and is now adapted to various communication problems, such as trellis coded modulation (TCM) [10], [11] and code division multiple access (CDMA) [12]. We refer to standard references [13]–[15] for an overview of turbo coding. Turbo equalization systems were first proposed in [16] and further developed in several articles [17], [18]. In

Paper approved by W. E. Ryan, the Editor for Modulation, Coding, and Equalization of the IEEE Communications Society. Manuscript received September 14, 2000; revised June 27, 2001, and October 22, 2001. This work was supported in part by the National Science Foundation under Grant CCR-0092598 (CAREER) and Grant CCR-9984515 (CAREER) and the Office Of Naval Research under Award N000140110117.

M. Tüchler is with the Institute for Communications Engineering, Technical University of Munich, D-80290 Munich, Germany (e-mail: michael.tuechler@ei.tum.de).

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

Publisher Item Identifier S 0090-6778(02)05108-5.

0090-6778/02$17.00 © 2002 IEEE

all these systems, MAP-based techniques, most often a VA producing soft output information [19], are used exclusively for both equalization and decoding [16], [17]. The more complex BCJR algorithm [5] was implemented in [17]. Combined turbo coding and equalization [20], [21] includes three or more layers: two or more coding layers as in conventional turbo coding applications and the channel equalizer.

The MAP/ML-based solutions often suffer from high computational load for channels with long memory or large constellation sizes (expensive equalizer) or convolutional codes with long memory (expensive decoder). This situation is exacerbated by the need to perform equalization and decoding several times for each block of data. A major research issue is thus the complexity reduction of such iterative algorithms. Ariyavisitakul and Li [22] proposed a joint coding-equalization approach, distinct from turbo equalization, working with convolutional coding and a DFE. Here, within the DFE, soft information from the DFE forward filter and tentative (hard) decisions from the decoder using the VA are fed back. Wang and Poor [12] proposed a turbo equalization-like system as part of a multiuser detector for CDMA. This iterative scheme is based on turbo equalization using an LE to reduce ISI and MAP decoding. The MAP equalizer is thus replaced with an LE, whose filter parameters are updated for every output symbol of the equalizer. In [23], the MAP equalizer in the turbo equalization framework is exchanged with a soft interference canceler based on linear filters with very low computational complexity, whose coefficients are obtained using a least-mean-square (LMS)-based update algorithm. This idea is enhanced in [24], where the filter coefficients are obtained using the LMS algorithm to match the output of a MAP equalizer. For varying signal-to-noise ratios (SNRs) and feedback information constellations, a linear estimate of the MAP equalizer is stored in a table and used for equalization in the receiver. The approach in [25] is similar to that of [23], but assumes a (known) impulse response of a partial response channel occurring in magnetic recording applications. The equalizer filter output is assigned a reliability measure enabling the receiver to decide whether the linear algorithm should be used instead of MAP equalization. Another common technique to decrease the complexity of the MAP equalizer is to reduce the number of states in the underlying trellis, which was applied to turbo equalization in [26]. The approaches in [23]–[25] and those proposed in this paper address a major shortcoming of the classical turbo equalization scheme [16]–[18], which is the exponentially increasing complexity of the equalizer for channels with a long impulse response or large signal alphabets. We replace the MAP equalizer with an LE and a DFE, where the filter parameters are updated using the MMSE criterion.

The paper is organized as follows. A brief definition of a coded data transmission system applying turbo equalization in the receiver is given in Sections II and III. In Section IV, we describe the general structure of a soft-in soft-out (SISO) equalizer based on MMSE equalization and derive four different implementations of this general approach using an LE or a DFE. Complexity considerations are explored in Section V. In Section VI, the two receiver components and the overall system are analyzed yielding estimates of the BER performance of the system and suggestions on how to select appropriate codes and SISO equalizers. These results are verified and compared to existing solutions [12], [16]–[18], [23]–[25] in Section VII and summarized in Section VIII.

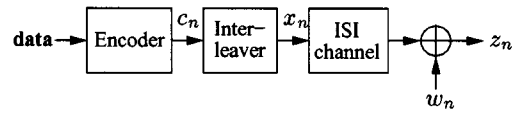

Fig. 2. Transmitter section of the data transmission system.

II. Notation and System Definition

To simplify the derivations, all systems to be investigated contain the same transmitter depicted together with the ISI channel in Fig. 2. The (binary) data is encoded with a (binary) convolutional encoder yielding the code symbols c_n, which are mapped to the alphabet $\mathcal{B}$ of the signal constellation. In this paper, for simplicity we assume binary phase shift keying (BPSK), i.e., $\mathcal{B} = \{+1, -1\}$, and that the channel impulse response coefficients h_k and the noise samples w_n are real valued. A framework to develop algorithms for higher order constellations and complex-valued h_k and w_n is presented in [27].

The transmission and receiving tasks are applied to blocks of data bits $b_i \in \{0, 1\}$ of length K_d. They are encoded to $K_c = K_d/R + K_o$ code symbols c_n, $n = 1, 2, \ldots, K_c$, $c_n \in \mathcal{B}$, where $R \in [0, 1]$ is the code rate and $K_o \geq 0$ is any overhead introduced by the encoder, e.g., a termination sequence. The interleaver permutes the c_n and outputs K_c symbols x_n, $n = 1, \ldots, K_c$, $x_n \in \mathcal{B}$, to be transmitted over the ISI channel. This operation is denoted $x_n = \Pi(c_n)$, where $\Pi(\cdot)$ is a fixed random permutation on K_c elements. For more information on $\Pi(\cdot)$ (see, for example, [15]). The permutation $\Pi^{-1}(\cdot)$, the deinterleaver, reverses the $\Pi(\cdot)$ operation. The noise is modeled as additive white Gaussian noise (AWGN), i.e., the noise samples w_n are independent and identically distributed (i.i.d.) with normal probability density function (pdf)

$$f_w(w) = \phi(w/\sigma_w)/\sigma_w \tag{2}$$

and independent of the data, where $\phi(x) = e^{-x^2/2}/\sqrt{2\pi}$. Given (1), the receiver input z_n is given by

$$z_n = \left(\sum_{k=0}^{M-1} h_k x_{n-k}\right) + w_n.$$

Before proceeding with a description of the different methods for turbo equalization, some frequently used notation is introduced. Vectors are written in bold letters and are considered to be column vectors. Matrices are specified by bold capital letters. Time-varying quantities are augmented with a time index n as the subscript. The $i \times j$ matrix $\mathbf{0}_{i\times j}$ contains all zeros, and $\mathbf{1}_{i\times j}$ contains all ones. $\mathbf{I}_i$ is the $i \times i$ identity matrix. The operator $\mathrm{E}(\cdot)$ is the expectation with respect to the joint pdf of x_n and w_n. The covariance operator $\mathrm{Cov}(\mathbf{x}, \mathbf{y})$ is given by $\mathrm{E}(\mathbf{x}\,\mathbf{y}^H) - \mathrm{E}(\mathbf{x})\mathrm{E}(\mathbf{y}^H)$, where $()^H$ is the Hermitian oper-

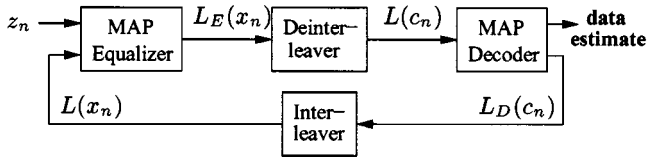

Fig. 3. A receiver performing turbo equalization.

ator. The L-value operator $L(x)$ is applied to quantities $x \in \{+1, -1\}$ and is given by

$$L(x) \triangleq \ln(P(x=+1)/P(x=-1))$$

i.e., the log likelihood ratio (LLR). The operator $\mathrm{Diag}(\cdot)$ to be applied to a length L vector returns an $L \times L$ square matrix with the vector elements along the main diagonal.

III. Basic Principle of Turbo Equalization

Fig. 3 depicts the receiver structure for turbo equalization pioneered in [16]. All other approaches presented here use the same structure and vary only in the type of equalizer. For decoding, we consider only the BER-optimal MAP approach. The MAP equalizer suitable for turbo equalization, which was shown to perform best in simulations among the trellis-based detection schemes [17], computes the *a posteriori* probabilities (APP's), $P(x_n = x | z_1, \ldots, z_{K_c})$, $x \in \mathcal{B}$, given K_c received symbols z_n, $n = 1, 2, \ldots, K_c$, and outputs the *a posteriori* LLR minus the *a priori* LLR

$$L_E(x_n) \triangleq \ln \frac{P(x_n = +1 | z_1, \ldots, z_{K_c})}{P(x_n = -1 | z_1, \ldots, z_{K_c})} - \ln \frac{P(x_n = +1)}{P(x_n = -1)}. \tag{3}$$

The *a priori* LLR, which is $L(x_n)$, represents prior information on the occurrence probability of x_n and is provided by the decoder. For the initial equalization step, no *a priori* information is available and hence we have $L(x_n) = 0, \forall n$. We emphasize that $L_E(x_n)$ is independent of $L(x_n)$. This and the concept of treating feedback as *a priori* information are the two essential features of any system applying the turbo principle [13] and turbo equalization in particular [16], [17]. The MAP decoder computes the APPs $P(c_n = x \,|\, L(c_1), \ldots, L(c_{K_c}))$, $x \in \mathcal{B}$, given K_c code bit LLRs $L(c_n)$, $n = 1, 2, \ldots, K_c$, and outputs the difference

$$L_D(c_n) \triangleq \ln \frac{P(c_n = +1 | L(c_1), \ldots, L(c_{K_c}))}{P(c_n = -1 | L(c_1), \ldots, L(c_{K_c}))} - \ln \frac{P(c_n = +1)}{P(c_n = -1)}$$

where the equalizer output $L_E(x_n)$ is considered to be *a priori* LLR $L(c_n)$ for the decoder. The interleaver $\Pi(\cdot)$ and the deinterleaver $\Pi^{-1}(\cdot)$ provide the correct ordering of the LLRs $L(c_n) = \Pi^{-1}(L_E(x_n))$ and $L(x_n) = \Pi(L_D(c_n))$, which are input to the equalizer and decoder, respectively. The MAP decoder also computes the data bit estimates

$$\hat{b}_i \triangleq \operatorname*{argmax}_{b \in \{0,1\}} P(b_i = b \,|\, L(c_1), \ldots, L(c_{K_c})). \tag{4}$$

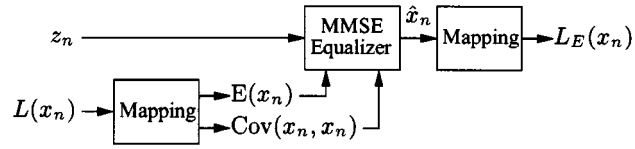

Fig. 4. A SISO equalizer based on MMSE equalization.

Applying the turbo principle, after an initial detection of a block of received symbols, blockwise decoding and equalization operations are performed on the same set of received data. A suitably chosen termination criterion stops the iterative process.

IV. Turbo Equalization Using MMSE Equalization

The MAP equalizer in Fig. 3 can be replaced with the SISO equalizer shown in Fig. 4. The depicted structure leads to a rather general class of SISO equalizers consisting of an MMSE equalizer, which computes estimates $\hat{x}_n$ of the transmitted symbols x_n from the received symbols z_n by minimizing the cost function $\mathrm{E}(|x_n - \hat{x}_n|^2)$. We will apply linear and nonlinear (including decision feedback) MMSE equalizers. We furthermore show that some instances of the SISO equalizer algorithms recover algorithms by Glavieux *et al.* [23] and Wu and Cioffi [25]. The SISO equalizer output $L_E(x_n)$ is obtained using the estimate $\hat{x}_n$

$$\begin{aligned} L_E(x_n) &\triangleq \ln \frac{P(x_n = +1 \,|\, \hat{x}_n)}{P(x_n = -1 \,|\, \hat{x}_n)} - \ln \frac{P(x_n = +1)}{P(x_n = -1)} \\ &= \ln \frac{p(\hat{x}_n \,|\, x_n = +1)}{p(\hat{x}_n \,|\, x_n = -1)} \end{aligned} \tag{5}$$

instead of z_n, $n = 1, \ldots, K_c$, which requires the knowledge of the distribution $p(\hat{x}_n \,|\, x_n = x)$ of $\hat{x}_n$ conditioned on $x_n = x$, $x \in \mathcal{B}$.

To perform MMSE estimation, the statistics $\overline{x}_n \triangleq \mathrm{E}(x_n)$ and $v_n \triangleq \mathrm{Cov}(x_n, x_n)$ of the symbols x_n are required. Usually, the x_n are assumed to be equiprobable and i.i.d., which corresponds to $L(x_n) = 0, \forall n$, and yields $\overline{x}_n = 0$ and $v_n = 1$. For general $L(x_n) \in \mathbb{R}$ (the x_n are not equiprobable), $\overline{x}_n$ and v_n are obtained as

$$\begin{aligned} \overline{x}_n &= \sum_{x \in \mathcal{B}} x \cdot P(x_n = x) = P(x_n = +1) - P(x_n = -1) \\ &= \frac{e^{L(x_n)}}{1 + e^{L(x_n)}} - \frac{1}{1 + e^{L(x_n)}} = \tanh(L(x_n)/2) \\ v_n &= \sum_{x \in \mathcal{B}} |x - \mathrm{E}(x_n)|^2 \cdot P(x_n = x) = 1 - |\overline{x}_n|^2. \end{aligned}$$

After MMSE equalization, we assume that the pdfs $p(\hat{x}_n \,|\, x_n = x)$, $x \in \mathcal{B}$, are Gaussian with the parameters $\mu_{n,x} \triangleq \mathrm{E}(\hat{x}_n | x_n = x)$ and $\sigma_{n,x}^2 \triangleq \mathrm{Cov}(\hat{x}_n, \hat{x}_n | x_n = x)$ [12]

$$p(\hat{x}_n \,|\, x_n = x) \approx \phi((\hat{x}_n - \mu_{n,x})/\sigma_{n,x})/\sigma_{n,x}. \tag{6}$$

This assumption tremendously simplifies the computation of the SISO equalizer output LLR $L_E(x_n)$. We emphasize that $L_E(x_n)$ should not depend on the particular *a priori* LLR $L(x_n)$. Therefore, we require that $\hat{x}_n$ does not depend on $L(x_n)$, which affects the derivation of the MMSE equalization algorithms.

A. Turbo Equalization Using Linear MMSE Equalization

1) Exact Implementation: The MMSE equalizer for this novel approach is an LE consisting of a length N filter with *time-varying* coefficients $c_{n,k}$, $k = -N_1, 1-N_1, \ldots, N_2$, where $N = N_1 + N_2 + 1$, which are defined by the linear MMSE estimate $\hat{x}_n$ [28] of x_n given the observation $\mathbf{z}_n \triangleq [z_{n-N_2} z_{n-N_2+1} \cdots z_{n+N_1}]^T$:

$$\hat{x}_n = \mathrm{E}(x_n) + \mathrm{Cov}(x_n, \mathbf{z}_n)\mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1}(\mathbf{z}_n - \mathrm{E}(\mathbf{z}_n)). \quad (7)$$

For data transmission over an ISI channel, this becomes

$$\hat{x}_n = \overline{x}_n + v_n \mathbf{s}^H(\sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H)^{-1}(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n)$$

where $\mathbf{H}$ is the $N \times (N+M-1)$ channel convolution matrix

$$\mathbf{H} \triangleq \begin{bmatrix} h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & & 0 \\ 0 & h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & 0 \\ & & & \ddots & & & & \\ 0 & & & \cdots & 0 & h_{M-1} & h_{M-2} \cdots & h_0 \end{bmatrix}$$

and

$$\begin{aligned} \overline{\mathbf{x}}_n &\triangleq [\overline{x}_{n-M-N_2+1}\ \overline{x}_{n-M-N_2+2} \cdots \overline{x}_{n+N_1}]^T \\ \mathbf{V}_n &\triangleq \mathrm{Diag}(v_{n-M-N_2+1}\ v_{n-M-N_2+2} \cdots v_{n+N_1}) \\ \mathbf{s} &\triangleq \mathbf{H}[\,\mathbf{0}_{1\times(N_2+M-1)} \quad 1 \quad \mathbf{0}_{1\times N_1}]^T. \end{aligned}$$

However, $\hat{x}_n$ depends on $L(x_n)$ via $\overline{x}_n$ and v_n. In order that $\hat{x}_n$ be independent of $L(x_n)$, we set $L(x_n)$ to 0 while computing $\hat{x}_n$, yielding $\overline{x}_n = 0$ and $v_n = 1$. This changes (7) to

$$\begin{aligned} \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n) &= (\sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H), \\ \hat{x}_n &= \mathbf{s}^H \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1}(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + (\overline{x}_n - 0)\mathbf{s}). \end{aligned}$$

Writing the MMSE LE output as

$$\hat{x}_n = \sum_{k=-N_1}^{N_2} c_{n,k}(z_{n-k} - \mathrm{E}(z_{n-k}))$$

where $\mathrm{E}(z_n) = \sum_{k=0}^{M-1} h_k \overline{x}_{n-k}$, the vector of the coefficients $\mathbf{c}_n \triangleq [c_{n,N_2}^*\ c_{n,N_2-1}^* \cdots c_{n,-N_1}^*]^T$ is consequently set to

$$\mathbf{c}_n \triangleq (\sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H)^{-1}\mathbf{s}.$$

This yields the final expression $\hat{x}_n = \mathbf{c}_n^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})$, from which the statistics $\mu_{n,x}$ and $\sigma_{n,x}^2$ of $\hat{x}_n$ are computed

$$\begin{aligned} \mu_{n,x} &= \mathbf{c}_n^H(\mathrm{E}(\mathbf{z}_n | x_n = x) - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s}) = x \cdot \mathbf{c}_n^H\mathbf{s} \\ \sigma_{n,x}^2 &= \mathbf{c}_n^H \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n | x_n = x)\mathbf{c}_n \\ &= \mathbf{c}_n^H(\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H - v_n\mathbf{s}\mathbf{s}^H)\mathbf{c}_n \\ &= \mathbf{c}_n^H\mathbf{s}(1 - \mathbf{s}^H\mathbf{c}_n). \end{aligned}$$

The output LLR $L_E(x_n)$ follows as

$$\begin{aligned} L_E(x_n) &= \ln \frac{\phi((\hat{x}_n - \mu_{n,+1})/\sigma_{n,+1})/\sigma_{n,+1}}{\phi((\hat{x}_n - \mu_{n,-1})/\sigma_{n,-1})/\sigma_{n,-1}} = \frac{2\hat{x}_n\mu_{n,+1}}{\sigma_{n,+1}^2} \\ &= 2\mathbf{c}_n^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})/(1 - \mathbf{s}^H\mathbf{c}_n). \end{aligned} \quad (8)$$

When $L(x_n) = 0, \forall n$, e.g., for the initial equalization step, we have $\overline{x}_n = 0$ and $v_n = 1, \forall n$, yielding a time-invariant coefficient vector $\mathbf{c}_n = \mathbf{c}_{\mathrm{NA}}$ (NA stands for no *a priori* information), the usual MMSE LE solution [1]

$$\begin{aligned} \mathbf{c}_{\mathrm{NA}} &\triangleq (\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H)^{-1}\mathbf{s}|_{L(x_n)=0} \\ &= (\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{H}^H)^{-1}\mathbf{s}. \end{aligned}$$

The corresponding output LLR is given by

$$L_E(x_n) = 2\mathbf{c}_{\mathrm{NA}}^H\mathbf{z}_n/(1 - \mathbf{s}^H\mathbf{c}_{\mathrm{NA}}). \quad (9)$$

2) Approximate Implementation I: Computing $\mathbf{c}_n$ for each time step n causes a high computational load for computing $L_E(x_n)$, since an $N \times N$ matrix has to be inverted for each n. A recursive algorithm to compute $\mathbf{c}_n$ from $\mathbf{c}_{n-1}$ devised in [27] reduces this load tremendously. A further reduction is possible by using time-invariant coefficients. We propose a low-complexity alternative, motivated as an approximation to the exact MMSE solution, which uses the time-invariant coefficient vector $\mathbf{c}_{\mathrm{NA}}$ to compute $\hat{x}_n$ given a general $L(x_n) \in \mathbb{R}$

$$\hat{x}_n = \mathbf{c}_{\mathrm{NA}}^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s}).$$

The use of this approximation will be justified by the excellent complexity/performance tradeoff shown in Sections V and VI. Given a set of general, nonzero, $L(x_n)$, while the MMSE-optimal coefficients must vary with n, if we restrict complexity such that the filter coefficients are time-invariant, then after subtracting the nonzero means, this set of coefficients can be viewed as MMSE optimal by simply ignoring (i.e., setting to zero) the $L(x_n)$. From the statistics

$$\begin{aligned} \mu_{n,x} &= \mathbf{c}_{\mathrm{NA}}^H(\mathrm{E}(\mathbf{z}_n | x_n = x) - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s}) = x \cdot \mathbf{c}_{\mathrm{NA}}^H\mathbf{s} \\ \sigma_{n,x}^2 &= \mathbf{c}_{\mathrm{NA}}^H \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n | x_n = x)\mathbf{c}_{\mathrm{NA}} \\ &= \mathbf{c}_{\mathrm{NA}}^H(\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H - v_n\mathbf{s}\mathbf{s}^H)\mathbf{c}_{\mathrm{NA}} \end{aligned}$$

for this approach, the output LLR is given by

$$L_E(x_n) = \frac{2\hat{x}_n\mu_{n,+1}}{\sigma_{n,+1}^2} = \frac{2\mu_{n,+1}\mathbf{c}_{\mathrm{NA}}^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})}{\sigma_{n,+1}^2}.$$

Computing $\sigma_{n,+1}^2$ for each n is computationally expensive. A possible simplification is to approximate $\sigma_{n,+1}^2$ with the time average $\overline{\sigma}^2 \triangleq (1/K_c)\sum_{n=1}^{K_c} \sigma_{n,+1}^2$

$$\overline{\sigma}^2 = \mathbf{c}_{\mathrm{NA}}^H\left(\sigma_w^2\mathbf{I}_N + \frac{1}{K_c}\sum_{n=1}^{K_c}(\mathbf{H}\mathbf{V}_n\mathbf{H}^H - v_n\mathbf{s}\mathbf{s}^H)\right)\mathbf{c}_{\mathrm{NA}}$$

which can be further simplified to

$$\overline{\sigma}^2 \approx \mathbf{c}_{\mathrm{NA}}^H\left(\sigma_w^2\mathbf{I}_N + \left(\frac{1}{K_c}\sum_{n=1}^{K_c} v_n\right)(\mathbf{H}\mathbf{H}^H - \mathbf{s}\mathbf{s}^H)\right)\mathbf{c}_{\mathrm{NA}} \quad (10)$$

using the approximation $\sum_{n=1}^{K_c}\mathbf{H}\mathbf{V}_n\mathbf{H}^H \approx (\sum_{n=1}^{K_c} v_n)\mathbf{H}\mathbf{H}^H$. Simulation results not shown in this paper reveal that using $\overline{\sigma}^2$ instead of $\sigma_{n,+1}^2$ does not sacrifice much performance.

When $L(x_n) = 0, \forall n$, the exact (Section IB-A-1) and the approximate implementation I of the MMSE LE are identical, yielding the same output LLR $L_E(x_n)$.

3) Approximate Implementation II: Another way to yield a time-invariant coefficient vector is to let $|L(x_n)| \to \infty, \forall n$, i.e., for perfect *a priori* information, yielding $\overline{x}_n = x_n$ and $v_n = 0$

$$\begin{aligned}\mathbf{c}_{\mathrm{MF}} &\triangleq (\sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H)^{-1}\mathbf{s}|_{|L(x_n)|\to\infty} \\ &= (\sigma_w^2\mathbf{I}_N + \mathbf{s}\mathbf{s}^H)^{-1}\mathbf{s} = \left(\sigma_w^{-2}\mathbf{I}_N - \frac{\sigma_w^{-2}\mathbf{s}\mathbf{s}^H\sigma_w^{-2}}{1+\sigma_w^{-2}\mathbf{s}^H\mathbf{s}}\right)\mathbf{s} \\ &= 1/(\sigma_w^2 + \mathbf{s}^H\mathbf{s})\cdot\mathbf{s}\end{aligned}$$

which is the matched filter response to the channel impulse response $h[n]$ normalized by $(\sigma_w^2 + \mathbf{s}^H\mathbf{s})^{-1}$, where $\mathbf{s}^H\mathbf{s}$ is the energy $E_h \triangleq \sum_{k=0}^{M-1}|h_k|^2$ of the channel. The estimates $\hat{x}_n$ are computed using $\mathbf{c}_{\mathrm{MF}}^H$ given general $L(x_n) \in \mathbb{R}$ as follows:

$$\hat{x}_n = \mathbf{c}_{\mathrm{MF}}^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s}) \tag{11}$$

yielding the statistics

$$\begin{aligned}\mu_{n,x} &= x\cdot\mathbf{c}_{\mathrm{MF}}^H\mathbf{s} = x\cdot E_h/(\sigma_w^2 + E_h) \\ \sigma_{n,x}^2 &= \mathbf{c}_{\mathrm{MF}}^H(\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H - v_n\mathbf{s}\mathbf{s}^H)\mathbf{c}_{\mathrm{MF}} \\ &= (E_h\sigma_w^2 + \mathbf{s}^H\mathbf{H}\mathbf{V}_n\mathbf{H}^H\mathbf{s} - v_nE_h^2)/(\sigma_w^2 + E_h)^2\end{aligned}$$

and the output LLR

$$L_E(x_n) = \frac{2\hat{x}_n\mu_{n,+1}}{\sigma_{n,+1}^2} = \frac{2E_h\mathbf{s}^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})}{E_h\sigma_w^2 + \mathbf{s}^H\mathbf{H}\mathbf{V}_n\mathbf{H}^H\mathbf{s} - v_n E_h^2}.$$

It turns out that $\hat{x}_n$ is identical to that for the linear SISO equalizer in [23] (in case the assumption $|L(x_n)| \to \infty, \forall n$, actually holds) and [25]. Thus, the algorithms in [23] and [25] are instances of the MMSE LE derived in Section IV-A1 under the constraint that $|L(x_n)| \to \infty, \forall n$. As in Section IV-A3, the variance $\sigma_{n,+1}^2$ can be approximated using the time average

$$\begin{aligned}\overline{\sigma}^2 &= \frac{1}{K_c}\sum_{n=1}^{K_c}(E_h\sigma_w^2 + \mathbf{s}^H\mathbf{H}\mathbf{V}_n\mathbf{H}^H\mathbf{s} - v_nE_h^2)/(\sigma_w^2 + E_h)^2 \\ &\approx \left(E_h\sigma_w^2 + \left(\frac{1}{K_c}\sum_{n=1}^{K_c}v_n\right)(\mathbf{s}^H\mathbf{H}\mathbf{H}^H\mathbf{s} - E_h^2)\right)\Big/(\sigma_w^2 + E_h)^2.\end{aligned} \tag{12}$$

When $L(x_n) = 0, \forall n$, for which this approach, as shown later, is certainly not suited, the output LLR becomes

$$L_E(x_n) = 2E_h\mathbf{s}^H\mathbf{z}_n/(E_h\sigma_w^2 + \mathbf{s}^H\mathbf{H}\mathbf{H}^H\mathbf{s} - E_h^2).$$

B. Turbo Equalization Using MMSE Decision Feedback Equalization

The MMSE equalizer for the second novel approach introduced in this paper is a DFE consisting of a length N feedforward filter with *time-varying* coefficients $c_{n,k}$, $k = -N_1, 1-N_1, \ldots, N_2$, $N = N_1 + N_2 + 1$, and a strictly causal length N_b feedback filter with *time-varying* coefficients $c_{n,k}^b$, $k = 1, 2, \ldots, N_b$. It is natural to address the postcursor ISI with the feedback filter and the precursor ISI with the feedforward filter yielding the choice

$$N_2 = 0, \quad N_b = M-1, \quad N_1 = N-1$$

for the filter length parameters [1]. A general expression for the MMSE DFE output is therefore

$$\begin{aligned}\hat{x}_n = \overline{x}_n &+ \left(\sum_{k=-(N-1)}^{0} c_{n,k}(z_{n-k} - \mathrm{E}(z_{n-k}))\right) \\ &- \left(\sum_{k=1}^{M-1} c_{n,k}^b(\hat{x}_{n-k}^d - \mathrm{E}(\hat{x}_{n-k}^d))\right)\end{aligned} \tag{13}$$

where the $\hat{x}_n^d$ are past decided estimates $\hat{x}_n$ obtained using an appropriate decision function, e.g.,

$$\hat{x}_n^d \triangleq \begin{cases}1, & \hat{x}_n \geq 0 \\ -1, & \hat{x}_n < 0\end{cases}$$

for BPSK. We assume that the DFE is error-free, i.e., $\hat{x}_n^d = x_n, \forall n$. Using the relation $c_{n,k}^b = \sum_{l=0}^{M-1} h_l c_{n,k-l}$, $k = 1, \ldots, M-1$, between the feedback coefficients $c_{n,k}^b$ and the feedforward coefficients $c_{n,k}$ [1], which also holds in the presence of *a priori* information $L(x_n)$ [29], the output equation (13) becomes

$$\begin{aligned}\overline{\mathbf{x}}_n &\triangleq [\hat{x}_{n-M+1}^d \cdots \hat{x}_{n-1}^d \overline{x}_n \overline{x}_{n+1} \cdots \overline{x}_{n+N-1}]^T \\ \hat{x}_n &= \overline{x}_n + \mathbf{c}_n^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n)\end{aligned}$$

where $\mathbf{H}$ is the $N \times (N+M-1)$ channel convolution matrix, $\mathbf{c}_n \triangleq [c_{n,0}^* c_{n,-1}^* \cdots c_{n,-(N-1)}^*]^T$, $\mathbf{z}_n \triangleq [z_n z_{n+1} \cdots z_{n+N-1}]^T$. Applying (7), i.e., $\mathbf{c}_n^H = \mathrm{Cov}(x_n, \mathbf{z}_n)\mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1}$, we find

$$\begin{aligned}\mathbf{V}_n &\triangleq \mathrm{Diag}\left(\mathbf{0}_{1\times(M-1)} v_n \; v_{n+1} \cdots v_{n+N-1}\right), \\ \mathbf{s} &\triangleq \mathbf{H}\left[\mathbf{0}_{1\times(M-1)} \quad 1 \quad \mathbf{0}_{1\times(N-1)}\right]^T \\ \hat{x}_n &= \overline{x}_n + v_n\mathbf{s}^H(\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H)^{-1}(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n)\end{aligned}$$

where the zero row in $\mathbf{V}_n$ follows because the transmitted symbols $x_{n'}$, $n' = n-1, \ldots, n-M+1$, are assumed to be known due to the available decided estimates $\hat{x}_{n'}^d$ yielding $v_{n'} = 0$. The derivation of the MMSE DFE is similar to MMSE estimation using (7) given the observation [29]

$$[\hat{x}_{n-M+1}^d \cdots \hat{x}_{n-1}^d \,|\, z_n z_{n+1} \cdots z_{n+N-1}]^T.$$

As in Section IV-A1, to ensure that $\hat{x}_n$ is independent from $L(x_n)$, we replace $\overline{x}_n$ and v_n with 0 and 1, respectively, while computing $\hat{x}_n$:

$$\begin{aligned}\hat{x}_n = 0 + 1\cdot\mathbf{s}^H(\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H)^{-1} \\ \cdot(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + (\overline{x}_n - 0)\mathbf{s}).\end{aligned}$$

With the final expressions for the feedforward filter coefficients, $\mathbf{c}_n \triangleq (\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^H + (1-v_n)\mathbf{s}\mathbf{s}^H)^{-1}\mathbf{s}$, and the MMSE estimate, $\hat{x}_n = \mathbf{c}_n^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})$, the statistics

$$\mu_{n,x} = x\cdot\mathbf{c}_n^H\mathbf{s}, \qquad \sigma_{n,x}^2 = \mathbf{c}_n^H\mathbf{s}(1 - \mathbf{s}^H\mathbf{c}_n)$$

TABLE I
NUMBER OF REQUIRED OPERATIONS PER RECEIVED SYMBOL PER ITERATION USING VARYING SISO EQUALIZATION ALGORITHMS. M: CHANNEL IMPULSE RESPONSE LENGTH; N: EQUALIZER FILTER LENGTH; 2^m: ALPHABET SIZE OF THE SIGNAL CONSTELLATION

section	approach	real multiplications	real additions
-	MAP equalizer	$3\cdot 2^{mM}+2\,m\,2^{m(M-1)}$	$3\cdot 2^{mM}+2(m-1)2^{m(M-1)}$
IV-A.1	exact MMSE LE	$16N^2+4M^2+10M-4N-4$	$8N^2+2M^2-10N+2M+4$
IV-A.2	approximate MMSE LE (I)	$4N+8M$	$4N+4M-4$
IV-A.3	approximate MMSE LE (II)	$10M$	$10M-2$
IV-B	MMSE DFE	$16N^2+4M^2+10M-4N-4$	$8N^2+2M^2-10N+2M+4$

and the output LLR

$$L_E(x_n) = 2\mathbf{c}_n^H(\mathbf{z}_n - \mathbf{H}\overline{\mathbf{x}}_n + \overline{x}_n\mathbf{s})/(1-\mathbf{s}^H\mathbf{c}_n) \qquad (14)$$

can be computed as in Section IV-A1.

When $L(x_n)=0, \forall\, n$, $\mathbf{c}_n$ is time invariant and equal to the usual MMSE DFE solution [1]

$$\mathbf{c}_{\mathrm{NA}} \triangleq (\sigma_w^2\mathbf{I}_N + \mathbf{H}\mathrm{Diag}(\mathbf{0}_{1\times(M-1)}\mathbf{1}_{1\times N})\mathbf{H}^H)^{-1}\mathbf{s}.$$

The corresponding output LLR is given by

$$L_E(x_n) = 2\mathbf{c}_{\mathrm{NA}}^H\mathbf{z}_n/(1-\mathbf{s}^H\mathbf{c}_{\mathrm{NA}}).$$

A recursive algorithm to efficiently compute $\mathbf{c}_n$ from $\mathbf{c}_{n-1}$ is given in [29]. It is also possible to derive approximate implementations similar to the approaches in Sections IV-A2 and IV-Q3. However, the resulting suboptimal algorithms exhibit inferior performance to the exact implementation [29], which in turn is inferior to the LE-based methods (see Section VII). We therefore omit the corresponding derivations here.

V. COMPLEXITY COMPARISON

An important aspect of these SISO equalizer algorithms is their computational complexity. We consider the MAP equalizer, the exact MMSE LE (Section IV-A1), the approximate MMSE LE (I) (Section IV-A2) and (II) (Section IV-A3), and the MMSE DFE (Section IV-B) for comparison.

We assume that the statistics $\overline{x}_n$ and v_n of x_n are available for all n and do not consider the computation of $L_E(x_n)$ [both mappings $\overline{x}_n,\, v_n \rightarrow L(x_n)$ and $\hat{x}_n \rightarrow L_E(x_n)$ strongly depend on $\mathcal{B}$]. Any overhead due to initialization (one-time operations for all iterations), e.g., to compute $\mathbf{c}_{\mathrm{NA}}$, is neglected. All quantities are assumed to be complex for this comparison. Table I shows the required number of real multiplications and additions per received symbol z_n per iteration, given general *a priori* information $L(x_n)\in\mathbb{R}$. The numbers for the MMSE-based approaches follow from the detailed algorithm derivations in [29]. The MAP equalizer uses the BCJR algorithm [5], where we considered only the computation of the path metrics (the γ quantity) for all trellis sections and the α-, β-recursions. Efficient implementations of the BCJR algorithm in the context of the turbo principle are presented, for example, in [13] and [17]. The algorithms for the approximate MMSE LE's (I) and (II) can be implemented in the frequency domain [30], which further decreases the computational load for specific systems.

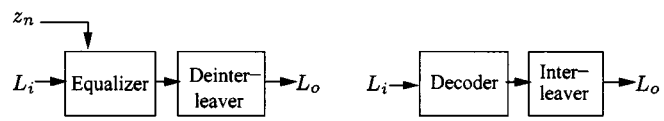

Fig. 5. The two basic receiver components.

VI. ANALYSIS

A. The EXIT Chart

We describe in this paper an analysis tool called an EXIT chart to compare the performance of the approaches described in Section IV and to ease the selection of system parameters, such as the generator of the convolutional code or the equalizer filter lengths.

A large body of research has been undertaken to provide tools for choosing design options for turbo codes, e.g., by analyzing the effects of the interleaver and bounds on the BER [31], [32]. Another analysis approach is to consider the turbo decoder as a high-dimensional nonlinear dynamic system [33], [34] whose convergence behavior can be characterized by its fixed points. These results have proven useful for determining the SNR regions, where the iterative algorithm provides improvement. We use the approach of ten Brink for analysis of TCM [35] and parallel concatenated convolutional codes (PCCCs) [36] using extrinsic information transfer (EXIT) charts for turbo equalization. For the EXIT analysis, the receiver components are modeled as devices mapping a sequence of observations z_n (equalizer only) and LLRs L_i, the *a priori* information, to a new sequence of LLRs L_o as shown in Fig. 5.

The sequence of random variables (r.v.s) L_i is assumed to be i.i.d. according to a single parameter conditional pdf $f_L(l\,|\,X=x)$ for a suitably chosen parameter σ_i^2

$$f_L(l\,|\,X=x) \triangleq \phi((l-x\sigma_i^2/2)/\sigma_i)/\sigma_i, \qquad x\in\mathcal{B}. \qquad (15)$$

We denote $f_L(l\,|\,X=x)$ briefly as $f_L(l\,|\,x)$. This pdf is motivated by the fact that an LLR $L(y|x)$ computed from the output $y=x+w$ of an AWGN channel with noise power $\mathrm{E}(w^2)=\sigma_w^2$ and input $x\in\mathcal{B}$ has a distribution according to (15), where $\sigma_i^2 = 4/\sigma_w^2$

$$L(y|x) = \ln(p(y|x=+1)/p(y|x=-1)) = 2/\sigma_w^2\cdot y.$$

The crucial observation in ten Brink's analysis is that the sequence of the output LLRs L_o is reasonably well approximated by a single parameter normal distribution of type (15) for a second parameter σ_o^2. A MAP equalizer and a MAP decoder are

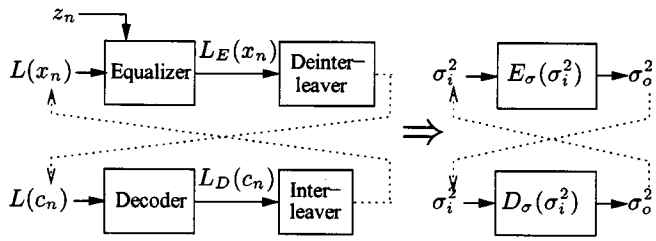

Fig. 6. Model for analysis of the iterative receiver algorithm.

assumed to match this assumption, which was shown to be accurate in [37] for sum-product decoding algorithms in the general setup of decoding of graph-based codes.

The SISO equalizer algorithms derived in this paper use the Gaussian model (6) to compute the output LLR $L_E(x_n) = 2\hat{x}_n\mu_{n,+1}/\sigma_{n,+1}^2$ using (5). Describing the $L_E(c_n)$ (after deinterleaving) with the r.v. L_o reveals that

$$\begin{aligned}\mathrm{Cov}(L_o, L_o \,|\, x_n = x) &= \frac{4\mu_{n,+1}^2}{\sigma_{n,+1}^4}\,\mathrm{Cov}(\hat{x}_n, \hat{x}_n \,|\, x_n = x)\\ &= \frac{4\mu_{n,+1}^2}{\sigma_{n,+1}^4}\sigma_{n,+1}^2 = \frac{4\mu_{n,+1}^2}{\sigma_{n,+1}^2}\\ \mathrm{E}(L_o \,|\, x_n = x) &= \frac{2\mu_{n,+1}}{\sigma_{n,+1}^2}\mathrm{E}(\hat{x}_n \,|\, x_n = x)\\ &= x\frac{2\mu_{n,+1}^2}{\sigma_{n,+1}^2}, \qquad x \in \mathcal{B}. \end{aligned} \tag{16}$$

The distribution of $L_E(c_n)$ exhibits the property that the variance is twice the magnitude of the mean similar to LLRs distributed with $f_L(l \,|\, x)$. However, using (15) to analyze the SISO equalizer output is still an approximation, since (6) is an assumption and the statistics $\mu_{n,+1}$ and $\sigma_{n,+1}^2$ are not constant in n, as is assumed in (15).

Using the definitions above, it is possible to define a single parameter transfer function $\sigma_o^2 = E_\sigma(\sigma_i^2)$ of the equalizer and $\sigma_o^2 = D_\sigma(\sigma_i^2)$ of the decoder. Fig. 6 illustrates how these transfer functions appear in the iterative decoding process.

Besides the constraint on the pdf of the output LLRs $L_E(x_n)$ and $L_D(c_n)$ of the receiver components, another assumption within the EXIT analysis is that the input LLRs $L(x_n)$ and $L(c_n)$ conditioned on x_n and c_n, respectively, are i.i.d. samples of the r.v. L_i. This is also taken into account in the derivation of the equalizer algorithms, where $\mathrm{Cov}(x_n, x_m) = 0$, $n \neq m$, is assumed. In the receiver, where $L(x_n)$ is provided by the decoder, this assumption is plausible for large interleaver block lengths, at least for several iterations. Also the decoding algorithm assumes the input code symbol LLR's $L(c_n)$ to be independent given c_n, which is again plausible only in presence of the deinterleaver, since the $L_E(x_n)$ given x_n are dependent due to the colored noise disturbing the MMSE equalizer output $\hat{x}_n$. The interleaving and deinterleaving process is thus the crucial step to approximately provide independence requirements at least locally between neighboring samples, and for several iterations.

For the EXIT charts, *a priori* information $L(x_n)$ or $L(c_n)$, respectively, is generated from L_i perfectly matching the two requirements [i.i.d. according to $f_L(l \,|\, x)$] and presented to each receiver component (equalizer, decoder) separately. This analysis is thus asymptotic in that the independence assumptions are assumed to hold over an infinite number of iterations, which is possible only for an infinite length block length ($K_c \to \infty$) and an ideal interleaver $\Pi(\cdot)$. For a finite block length K_c, the EXIT chart analysis is still useful over several iterations. While the independence assumption can be satisfied to arbitrary accuracy, we emphasize that the assumed distribution (15) is only approximately satisfied even for large interleaving. The main justification for the proposed analysis is the apparent usefulness of the method demonstrated in the sequel.

Similar analysis tools for turbo equalization [38], turbo codes [39], low-density-parity-check codes [40], or TCM [41] are available, which are all based on (15) but differ in the observed parameter. For example, the approach in [41] considers the SNR of the LLRs for the analysis, which is to observe transfer functions of type

$$\frac{(\mathrm{E}(L_i|x_n = x))^2}{\mathrm{Cov}(L_i, L_i|x_n = x)} \to \frac{(\mathrm{E}(L_o|x_n = x))^2}{\mathrm{Cov}(L_o, L_o|x_n = x)}$$

obtained by passing *a priori* LLRs L_i at some input SNR through the equalizer and decoder yielding an output SNR of the LLRs L_o. Ten Brink's approach computes the mutual information

$$I(L_i;\ X) = \tfrac{1}{2}\sum_{x\in\mathcal{B}}\int_{-\infty}^{\infty} f_L(l \,|\, x) \cdot \log_2 \frac{2f_L(l \,|\, x)}{f_L(l \,|\, +1) + f_L(l \,|\, -1)}\,\mathrm{d}l \tag{17}$$

where $I(L_i;\ X) \in [0, 1]$, between the r.v.s L_i and X. After passing samples of L_i through the equalizer or the decoder, at the output the mutual information $I(L_o;\ X)$ between the r.v.s L_o and X is obtained by applying (17) using the distribution of L_o. This is done in [35] and [36] by observing the histogram of the outputted LLRs to estimate the pdf of L_o and then computing $I(L_o;\ X)$ numerically using (17). Thus, even though the pdf of L_o is assumed to be of type (15), which is crucial to model the input LLRs of the following receiver component, $I(L_o;\ X)$ is obtained without this assumption.

We denote $I(L_i;\ X)$ briefly as I_i and $I(L_o;\ X)$ briefly as I_o. From the definition of $f_L(l \,|\, x)$, we see that I_i is only a function of σ_i^2 with the two extremal values $I_i(\sigma_i^2 = 0) = 0$ and $I_i(\sigma_i^2 \to \infty) = 1$ for no and perfect *a priori* information. The EXIT chart is the transfer function $I_o = E_{\mathrm{MI}}(I_i)$ of the equalizer or $I_o = D_{\mathrm{MI}}(I_i)$ of the decoder mapping the input variable $I_i \in [0, 1]$ to the output variable $I_o \in [0, 1]$. The range $[0, 1]$ of I_o compactly describes the quality of the output LLRs of a receiver component and is more convenient than $[0, \infty)$ for the approaches based on σ_i^2 or the SNR.

B. EXIT Chart of a MAP Decoder

Fig. 8 depicts EXIT charts of a MAP decoder using several optimal (with respect to (w.r.t.) the distance spectrum) rate $R = 1/2$ codes including rate $R = 2/3$ punctured versions. The octal notation $G = [a\ b]$ of the code generators is taken from [3]. To obtain the EXIT charts, $K_c = 10^7$ code symbols c_n generated from randomly chosen equiprobable information bits b_i and K_c

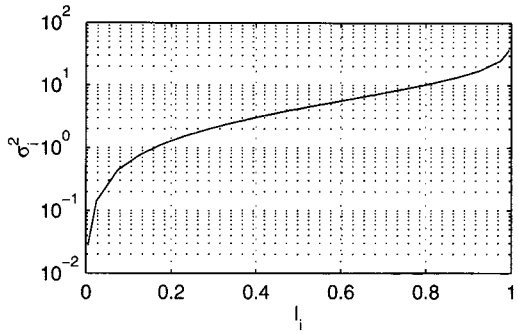

Fig. 7. Mapping between the parameters σ_i^2 and I_i.

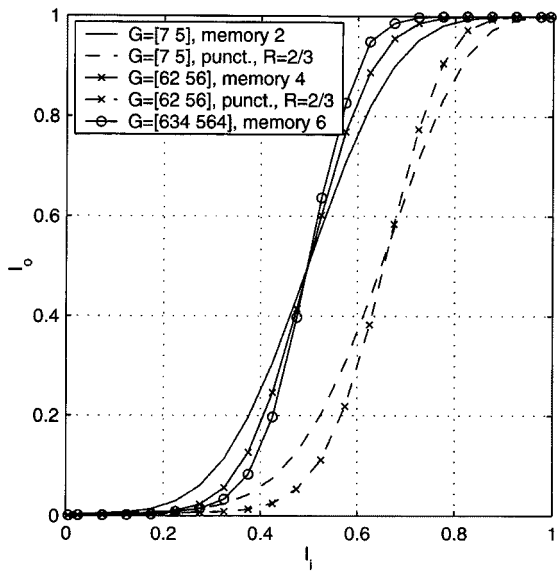

Fig. 8. EXIT charts of a map decoder.

corresponding *a priori* LLRs $L(c_n)$ distributed with $f_L(l\,|\,x)$ given a preset σ_i^2 are used. This is done by randomly generating K_c LLRs $L(c_n)$ with the pdf $f_L(l\,|\,+1)$ and flipping the sign of all $L(c_n)$ where $c_n = -1$ yielding the pdf $f_L(l\,|\,-1)$ for those LLRs. For the chosen σ_i^2, the mutual information I_i is computed numerically using (17), which is the fixed function depicted in Fig. 7. To obtain I_o, in [35] and [36] the pdfs of the output LLRs $L_D(c_n)$ are estimated by splitting the K_c LLRs $L_D(c_n)$ into two groups, where $c_n = +1$ and $c_n = -1$, respectively. A histogram of the samples in each group approximates $f_L(l\,|\,x)$, $x \in \mathcal{B}$, which is therefore used in (17) to obtain I_o. The EXIT chart is constructed by repeating the procedure above for several values of σ_i^2 yielding pairs (I_i, I_o).

C. EXIT Chart of a SISO Equalizer

Fig. 9 depicts EXIT charts of the equalizer using MAP equalization and MMSE equalization. We selected the length-5 ISI channel

$$h[n] = 0.227\delta[n] + 0.46\delta[n-1] + 0.688\delta[n-2] + 0.46\delta[n-3] + 0.227\delta[n-4] \quad (18)$$

from [1] for the EXIT chart analysis and the simulations. This channel causes severe ISI, enabling turbo equalization to yield large performance gains. However, the performance enhancements observed for this channel are representative and qualitatively similar to those obtained for a wide variety of fixed and fading channels [29]. The noise variance σ_w^2 is determined according to the SNR E_s/N_0 defined as

$$\frac{E_s}{N_0} \triangleq \frac{\mathrm{E}(|z_n|^2)}{N_0} = \frac{\mathrm{E}(|x_n|^2)\cdot E_h}{N_0} = \frac{E_h}{2\sigma_w^2} \quad (19)$$

which we set to 4 dB for the chart in Fig. 9. The filter parameters for the exact MMSE LE and the approximate MMSE LE (I) were set to $(N_1 = 9,\, N_2 = 5,\, N = 15)$ and for the MMSE DFE to $(N = 15,\, N_b = 4)$.

For the EXIT charts, 10^7 randomly chosen equiprobable symbols $x_n \in \mathcal{B}$ were generated and transmitted over the ISI channel. The corresponding *a priori* LLRs $L(x_n)$ given a preset σ_i^2 were generated as in Section VI-B. The equalizer processed the received symbols z_n together with $L(x_n)$. The quantity I_i is computed from σ_i^2 and I_o is computed from the output LLRs $L_E(x_n)$ as in Section VI-B.

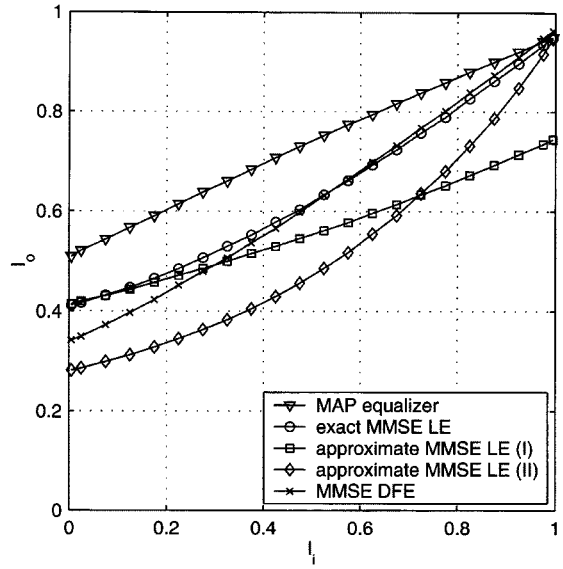

Fig. 9. EXIT charts of the equalizer at 4-dB SNR.

D. BER Estimation

In [36], the EXIT analysis is also used to estimate the information BER $P(b_i \neq \hat{b}_i)$, where $\hat{b}_i$ is given by (4), of the MAP decoders as part of a PCCC system considered there. This estimation is based on the Gaussian assumption (15) yielding a unique $P(b_i \neq \hat{b}_i)$ corresponding to a parameter σ_o^2 or I_o, respectively, of the decoder output LLRs $L_D(c_n)$. For more information on how to estimate this error probability, we refer to [36].

We use simulation to assign a BER $P(b_i \neq \hat{b}_i)$ to each $I_o \in [0,\, 1]$ obtained for the decoder EXIT chart, since the chart itself is based on simulation yielding the BER as a side product. Fig. 10 displays the BER as a function of the expression $1 - I_o$ on logarithmic scales for the rate $R = 1/2$ codes from Section VI-B using the same simulation parameters. The curves for several code memories and that for the punctured rate $R = 2/3$ codes (not shown here) were almost identical, which verifies the

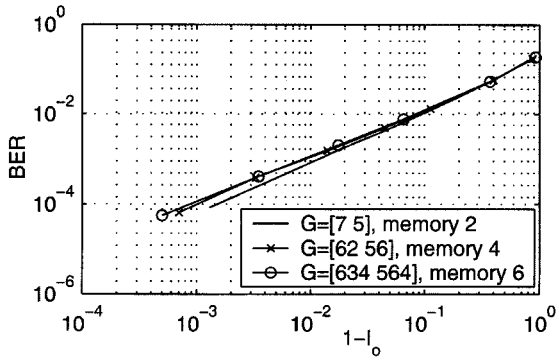

Fig. 10. Information BER of MAP decoding as a function of I_o.

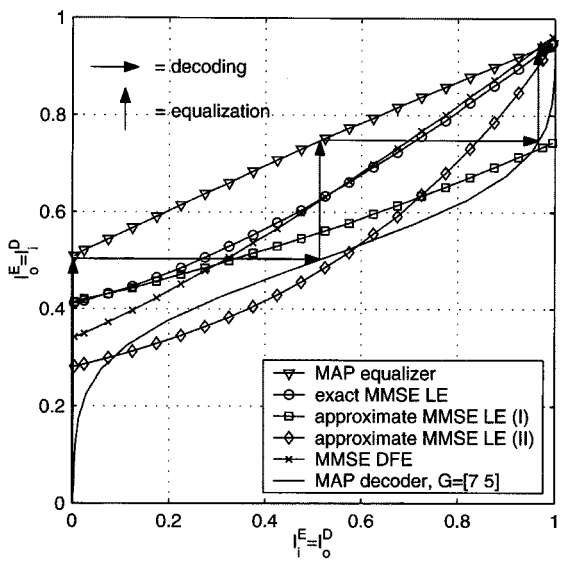

Fig. 11. Receiver EXIT chart at 4 dB E_s/N_0.

result in [36] that I_o alone can be used to estimate $P(b_i \neq \hat{b}_i)$ independently of the chosen code.

E. Application to the Iterative Receiver Algorithm

In the turbo equalization-based receiver, equalization and decoding steps are iterated by passing the LLRs $L_E(x_n)$ and $L_D(c_n)$ between the receiver components. Using our analysis, we can describe this mechanism just by the evolution of the distribution of the r.v.s L_i and L_o describing the LLRs $L_E(x_n)$ and $L_D(c_n)$. Using (15), this density evolution is completely described by the change of σ_i^2 to σ_o^2 or I_i to I_o, respectively. To distinguish between equalizer and decoder, the quantities I_i and I_o are augmented with the superscripts E (equalizer) and D (decoder).

The iterative process starts with an initial equalization, where $L(x_n) = 0, \forall n$, and therefore $I_i^E = 0$. Next, the output LLRs $L_E(x_n)$ described by $I_o^E = I_i^D$ are fed into the decoder yielding LLRs $L_D(c_n)$ described by $I_o^D = I_i^E$ which are fed back to the equalizer and so forth. This procedure is described with a single *receiver* EXIT chart combining Figs. 8 and 9 as shown in Fig. 11, where the decoder chart is flipped along the $I_i^D = I_o^D$ line. The iteration process is a trace between the transfer curves of the two receiver components.

Fig. 11 shows such a receiver EXIT chart at 4 dB E_s/N_0, where the receiver uses a MAP equalizer or one of the MMSE-based equalizers and a MAP decoder for the code $G = [7\ 5]$. In the example trace, a turbo equalization system using a MAP equalizer needs only two iterations (equalization/decoding tasks after one initial equalization/decoding task) to nearly reach $I_o^D = 1$ at the decoder output, which corresponds to a zero BER. Surprisingly, with the exact MMSE LE the same performance is achieved, only after four iterations (for clarity, the trace is not shown here). The approximate MMSE LE (II) is also capable of achieving the MAP equalizer performance for large values of I_i^E, but it has a very poor gain in I_o^E for small values of I_i^E. The severe ISI caused by the channel (18), which acts as noise disrupting the output estimates of the equalizer, is not reduced at all for *a priori* LLRs $L(x_n)$ close to 0, yielding the poor values of I_i^E. The system thus cannot find a trajectory leading to a large I_o^D using the code $G = [7\ \ 5]$ and stops improving the performance after around three iterations and remains in the fixed point $I_o^D = I_i^E = 0.095$. The approximate MMSE LE (I) starts with the same performance as the MMSE LE [both algorithms are identical for $L(x_n) = 0, \forall n$], but shows poor performance for large values of I_i^E. The fixed point of the receiver is here $I_o^D = 0.95$. The MMSE DFE shows a poorer performance than the linear counterpart (exact MMSE LE) for small values of I_i^E and becomes better for $I_i^E \geq 0.55$. The performance is even better than that of a MAP equalizer for $I_i^E \geq 0.95$, which likely stems from the hard decision element in this algorithm.

Fig. 11 also reveals that a MAP equalizer, the exact MMSE LE, and the approximate MMSE LE (II) yield the same value $I_o^E < 1$ for perfect *a priori* information $L(x_n)$, which corresponds to $I_i^E = 1$. This is due to (3) and (5) to compute the output LLRs $L_E(x_n)$, which prohibit that $L(x_n)$ itself is used to compute $L_E(x_n)$. It follows that for $|L(x_n)| \to \infty$ the ISI disturbing a received symbol z_n can be removed completely, since all $x_{n'}$, $n' \neq n$, are known. According to (11), the equalized estimate

$$\hat{x}_n = (E_h \cdot x_n + w_n)/(E_h + \sigma_w^2)$$

remains, yielding the output LLR

$$L_E(x_n) = \ln \frac{\phi((\hat{x}_n - E_h)/\sigma_w)/\sigma_w}{\phi((\hat{x}_n + E_h)/\sigma_w)/\sigma_w} = \frac{2E_h\hat{x}_n}{\sigma_w^2}.$$

The variance $\sigma_o^2 = \mathrm{Cov}(L_o, L_o \,|\, x_n = x) = 4E_h^2/\sigma_w^2$ of the r.v. L_o describing $L_E(x_n)$ and hence I_o^E derived from σ_o^2 are a function of σ_w^2 and E_h, only. Using Fig. 7, the upper bound on I_o^E attained for $I_i^E = 1$ at 4 dB E_s/N_0 is $I_o^E \leq 0.95$. Thus, the MAP decoder receives LLRs $L(c_n)$ with at most $I_i^D = 0.95$, which is the matched filter bound for the turbo equalization system considered here, i.e., the asymptotic performance of the receiver is at best the performance of the MAP decoder alone for transmission over an AWGN channel. This bound is overcome by adding a rate-1 recursive precoder to the transmitter [42], which can improve the system performance tremendously [38], [43].

TABLE II
EXPECTED BER AFTER 0, 1, 2, OR ∞ ITERATIONS OF A RECEIVER USING VARIOUS SISO EQUALIZERS AND A MAP DECODER

SISO equalizer	0th	1st	2nd	∞
MAP equalizer	0.070	$3.5 \cdot 10^{-3}$		
exact MMSE LE	0.150	0.070	0.020	
approximate MMSE (I)	0.150	0.090	0.050	$4.0 \cdot 10^{-3}$
approximate MMSE LE (II)	0.230	0.220	0.210	0.210
hybrid approx. MMSE LE (I)/(II)	0.150	0.090	0.050	
MMSE DFE	0.180	0.130	0.095	

The receiver EXIT chart in Fig. 11 suggests a hybrid approach for equalization, which is to use one of the two approximate MMSE LEs (I) or (II) depending on the *a priori* information constellation, i.e., the value of I_i^E. At 4 dB E_s/N_0, using (I) as an equalization algorithm for at least three initial iterations followed by (II) for the successive iterations gives a trace that remarkably reaches the performance of the MAP equalizer requiring a very low computational complexity (see Table I). With the hybrid approach, we can combine the good starting behavior of (I) (small I_i^E) and the good performance of (II) for large values of I_i^E. A simple criterion to optimally switch between different SISO equalizers is to pick the equalizer yielding the largest I_o^E given a specific I_i^E. Since I_o is monotonically increasing in σ_o^2 (see Fig. 7), the equalizer yielding the largest σ_o^2, where $\sigma_o^2 = 4\mu_{n,+1}^2/\sigma_{n,+1}^2$ for the SISO equalizers based on MMSE equalization (16), should be selected. For the approximate MMSE LE (I) and (II), the mean $\mu_{n,+1}^2$ is constant and the variance $\sigma_{n,+1}^2$ varies in n. Consequently, the criterion changes to selecting the equalizer maximizing the average variance

$$\overline{\sigma}_o^2 = 4\mu_{n,+1}^2 \cdot \frac{1}{K_c} \sum_{n=1}^{K_c} 1/\sigma_{n,+1}^2 \tag{20}$$

given K_c estimates $\hat{x}_n$, $n = 1, \ldots, K_c$. This average variance can be approximated with the lower bound

$$\overline{\sigma}_o^2 \geq 4\mu_{n,+1}^2 \Bigg/ \left(\frac{1}{K_c} \sum_{n=1}^{K_c} \sigma_{n,+1}^2 \right)$$

where the denominator is the time average $\overline{\sigma}_o^2$ of the variance $\sigma_{n,+1}^2$ computed in (10) and (12).

Following the traces of the receiver for all SISO equalizers yields values for I_o^D after each iteration. Using Fig. 10, we can compute estimates of the BER $P(b_i \neq \hat{b}_i)$ of the iterative receiver algorithm given in Table II.

The receiver EXIT chart also helps to choose a suitable code for the system. Fig. 12 displays the tradeoff in selecting a code, which ideally enables the iterative algorithm to converge at SNRs as low as possible to a BER as low as possible. The transfer curve of the MAP equalizer in Fig. 12 touches the curve of the code $G = [62\ 56]$ at $\approx$0.8 dB E_s/N_0 and therefore prohibits the iterative system to converge, whereas there is still a path through the bottleneck using the code $G = [7\ 5]$.

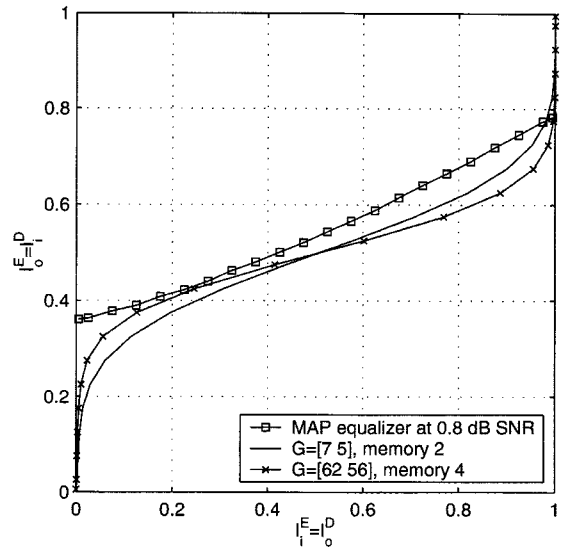

Fig. 12. Receiver EXIT chart at 0.8 dB E_s/N_0 for various codes.

TABLE III
SNR THRESHOLD IN E_s/N_0 FOR SIGNIFICANT PERFORMANCE IMPROVEMENT OF A RECEIVER USING DIFFERENT SISO EQUALIZERS AND A MAP DECODER

SISO equalizer	SNR threshold
MAP equalizer	0.4 dB
MMSE LE	1.0 dB
approximate MMSE LE (I)	2.3 dB
approximate MMSE LE (II)	∞ dB
hybrid approx. MMSE LE (I)/(II)	2.3 dB
MMSE DFE	2.4 dB

However, the code $G = [62\ 56]$ yields a lower error floor after convergence, since the transfer curve approaches $I_o^D = 1$ faster. A similar problem, i.e., to find convolutional codes with a good tradeoff between early convergence (at low SNRs) and a small error floor after convergence, exists in turbo coding. Here, techniques such as choosing a generator with feedforward polynomials of degree much larger than that of the feedback polynomial [44] or systematic doping [45] are available and can be applied to turbo equalization also.

Table III gives SNR thresholds in E_s/N_0 for significant performance improvement of a receiver using different SISO equalizers and a MAP decoder for the code $G = [7\ 5]$. The thresholds were obtained by generating equalizer EXIT charts at varying SNRs until the transfer curve touched or intersected the flipped decoder transfer curve at $I_o^D = 0.5$ or less, which corresponds to a fixed point of the system with a BER $P(b_i \neq \hat{b}_i)$ of 0.08 or more (using Fig. 10). Since the equalizer and the flipped decoder transfer curve have a similar slope, increasing the SNR, which nearly results in a parallel upshift of the equalizer transfer curve, yields a quickly moving fixed point in the area around $I_o^D = 0.5$. The corresponding BER $P(b_i \neq \hat{b}_i)$ changes quickly too, which is the usually observed "water fall" region in the performance plot (BER versus E_s/N_0) of a turbo equalization system.

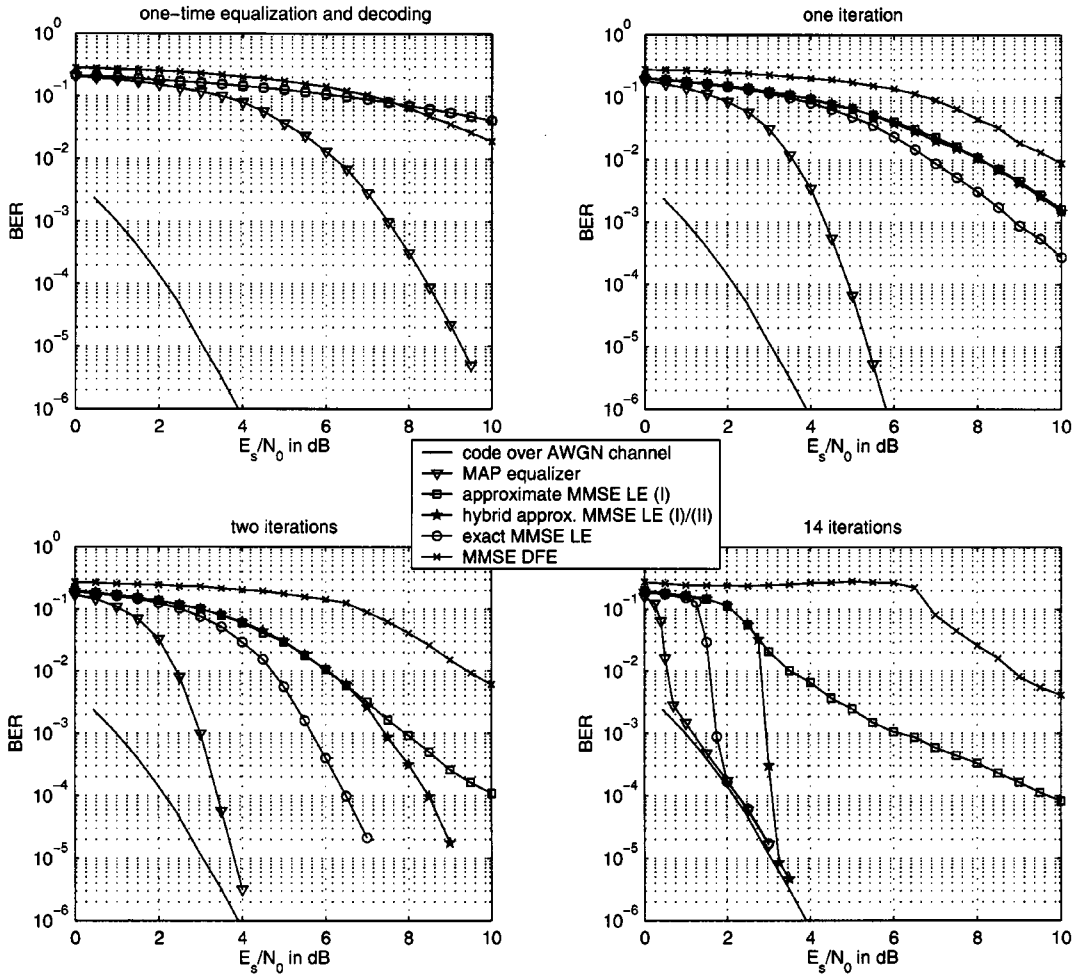

Fig. 13. BER performance of receivers applying turbo equalization.

VII. Verification of the Analytical Results

We present BER performance results by simulating data transmission using the receiver introduced in Section III and analyzed in Section VI. The transmitter (see Fig. 2), the ISI channel (18), the decoder (MAP decoding), and the interleaver/deinterleaver $\Pi(\cdot)/\Pi^{-1}(\cdot)$ are identical for all systems. They differ only in the SISO equalizer, which is either a MAP equalizer, an exact MMSE LE (derived in Section IV-A1), an approximate MMSE LE (I) (Section IV-A-2) or (II) (Section IV-A-2), or an MMSE DFE (Section IV-B).

We use a recursive, systematic encoder for the code $G = [7\ 5]$, where the first generator is the feedback part, to encode length K_d blocks of data bits b_i. The K_c code symbols c_n are interleaved using a permutation $\Pi(\cdot)$ obtained by an S-random construction [15] with $S = 0.5\sqrt{0.5 \cdot K_c}$. Both the channel impulse response (18) and the noise variance σ_w^2 determined from the SNR E_s/N_0 defined in (19) are known to the receiver, which facilitates turbo equalization. As suggested in Section VI-E, we include a hybrid approach using the approximate MMSE LEs (I) and (II) and switch between them by evaluating the average variance $\overline{\sigma}_o^2$ as in (20) prior to each equalization step.

The number of performed iterations (one equalization/decoding task after one initial equalization/decoding task) is 14 for all systems and SNRs, which is a conservative choice. For example, the analysis in Section VI-E predicts convergence after two to three iterations using a MAP equalizer at 4 dB E_s/N_0 (see Fig. 11). The number of iterations after which the BER eventually ceases to improve is a function of the block length K_c, the interleaver $\Pi(\cdot)$, the SNR, and the used equalization and decoding algorithms. Some suitable stopping criteria for the iterative process have been considered by Bauch for ML/MAP-based algorithms [46], based on the change of the hard decisions $\hat{b}_i$, the cross entropy of the LLRs communicated between the equalizer and the decoder, or the reliability of the information bit decisions.

Fig. 13 shows the BER performance for the block length $K_d = 2^{15}$. The upper left plot shows the performance of separate or one-time equalization and decoding, which exhibits the expected behavior: a system using a MAP equalizer is best, followed by the MMSE DFE (for E_s/N_0 above 7.7 dB) and the MMSE LE. These results are the same as would be obtained using the SISO equalizers presented here in the absence of *a priori* information, i.e., $L(x_n) = 0, \forall n$. With this constraint, the exact MMSE LE and the approximate MMSE LE (I) are identical. The approximate MMSE LE (II) was not considered. A lower bound on the BER performance is that of coded data transmission over an AWGN channel at the same E_s/N_0 as shown in Section VI-E.

Using iterative equalization and decoding, after 1 (top right), 2 (bottom left), and 14 iterations (bottom right), the

TABLE IV
BER AFTER 0, 1, 2, OR 14 ITERATIONS OF A RECEIVER USING DIFFERENT SISO EQUALIZERS AND A MAP DECODER

SISO equalizer	0th	1st	2nd	14th
MAP equalizer	0.074	$2.8 \cdot 10^{-3}$	$3.2 \cdot 10^{-6}$	$9.9 \cdot 10^{-7}$
exact MMSE LE	0.146	0.082	0.028	
approximate MMSE LE (I)	0.146	0.093	0.060	$6.5 \cdot 10^{-3}$
approximate MMSE LE (II) (alone)	0.254	0.243	0.240	0.240
hybrid approximate MMSE LE (I)/(II)	0.146	0.093	0.060	
MMSE DFE	0.204	0.200	0.205	0.267

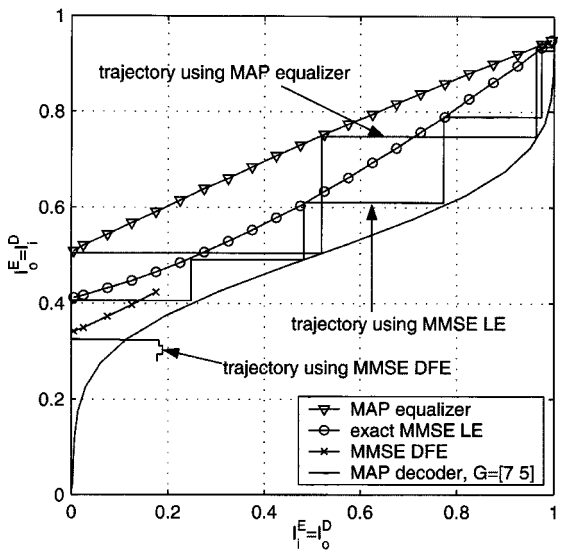

Fig. 14. Receiver EXIT charts with traces of iterative algorithms.

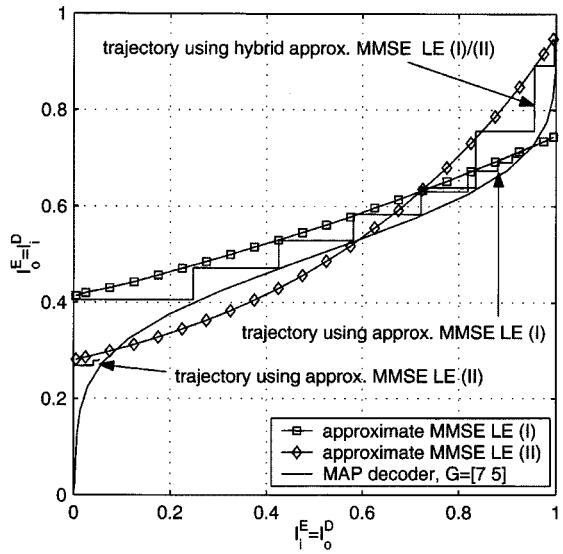

Fig. 15. Receiver EXIT charts with traces of a receiver using hybrid linear MMSE equalization.

performance order changes drastically. A system using a MAP equalizer is still best, followed by the exact MMSE LE and the hybrid approximate MMSE LE (I)/(II). The receiver using the MMSE DFE provides only a minor performance improvement. The approximate MMSE LE (II) alone was not considered due to its poor starting behavior. Using the approximate MMSE LE (I) alone provides good performance, however, with the hybrid approach, which does not alter the computational complexity, the performance is rather remarkable, approaching the AWGN bound.

Since the receiver performed only 14 iterations and the block length K_d is finite, the SNR thresholds from Table III are somewhat too optimistic. However, for the MAP equalizer, the exact MMSE LE, the approximate MMSE LE (I), and the hybrid approach the derived SNR thresholds and match exactly (within 0.1 dB) the actual threshold for which $BER < 0.08$ holds after performing all 14 iterations. The analysis fails for the MMSE DFE.

The BERs over successive iterations for each receiver algorithm at 4 dB E_s/N_0 are shown in Table IV and indicate the degree to which the BER analysis matches the actual performance. The predicted BER values in Table II closely match the simulated results for all SISO equalizers except for the MMSE DFE. This discrepancy becomes most obvious by looking at receiver EXIT charts at 4 dB SNR including the trace of the iteration process. The traces were obtained by computing I_o^E or I_o^D from 10^7 equalizer/decoder output LLRs $L_E(x_n)$ or $L_D(c_n)$, respectively, over several blocks as inSection IV-B. Figs. 14 and 15 depict receiver EXIT charts together with the simulated traces of the corresponding iterative algorithm at 4 dB E_s/N_0. The trace of the system using an MMSE DFE does not follow the transfer curves of equalizer and decoder alone. Therefore, the EXIT chart analysis cannot be applied to this algorithm. Moreover, the performance is worse than that of the linear counterpart, the exact MMSE LE, at the same computational complexity, a result which was obtained for a number of different channels. The only difference between the equations used to obtain the output LLRs $L_E(x_n)$ with the exact MMSE LE (8) or the MMSE DFE (14) is to replace the means $\overline{x}_{n-i}$, $i = 1, \ldots, N_b$, with the past decided estimates $\hat{x}_{n-i}^d$ while computing $\mathbf{c}_n$ and $\overline{\mathbf{x}}_n$ (when N_1 and N_2 are set appropriately for the MMSE LE). While this step often provides performance gains for *separate* equalization and decoding, the early use of hard decisions appears to be detrimental in an iterative algorithm.

Figs. 16 and 17 depict the performance of the receiver for varying block lengths K_d. For short K_d, repeated interleaving causes short cycles leading to dependencies within

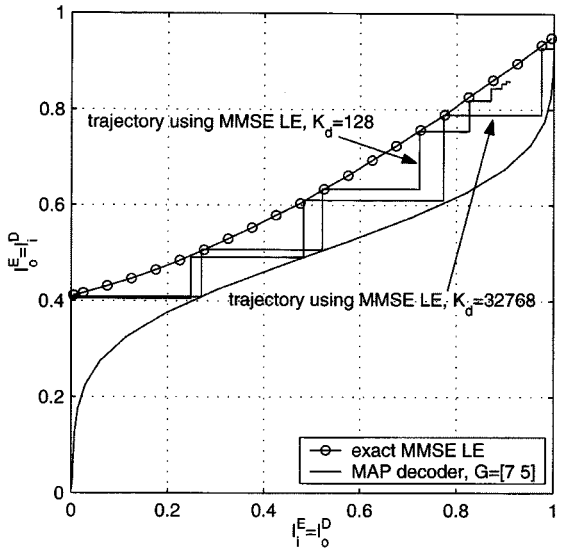

Fig. 16. Receiver EXIT charts with traces of a receiver using linear MMSE equalization at two block lengths K_d.

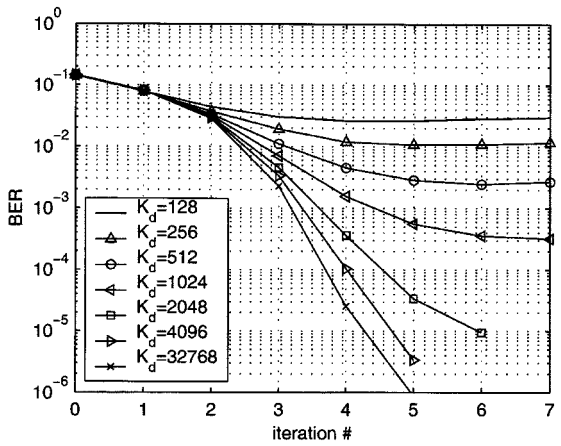

Fig. 17. BER of the receiver at 4 dB E_s/N_0 using exact MMSE linear equalization for several iterations and block lengths K_d.

the LLRs communicated between the equalizer and decoder. For example, we looked at the BER of a receiver using the exact MMSE LE at 4 dB E_s/N_0 for block lengths $K_d \in \{2^i \mid i = 7, 8, 9, 10, 11, 12, 15\}$ (Fig. 17) and showed the trace of the iteration process for $K_d = 2^{15}$ and $K_d = 2^7$ (Fig. 16). The EXIT chart analysis is still valid (the trace follows the transfer curves of the receiver components), but only for the number of iterations for which the independence requirements are matched. After that, the gains in I_o^E and I_o^D decline and the BER stops to decrease over the iterations. Finally, the system trajectory rests in a fixed point within the area between the two transfer curves occurs.

VIII. Conclusion

The preceding sections introduced and analyzed several algorithms for turbo equalization, which are applied to coded data transmission over ISI channels. In addition to the original ML/MAP-based method, several new approaches based on MMSE equalization have been developed. The approaches using linear MMSE equalization, especially a novel hybrid approach using two approximate versions of linear MMSE equalization, perform as well as the BER-optimal MAP equalizer, only requiring a few more iterations to achieve a similar BER performance. However, the results also showed that a receiver using a MAP equalizer can achieve considerable performance improvement through iteration over SNR ranges where all other approaches fail. The system employing decision feedback MMSE equalization did not provide satisfactory results. The BER performance of the decoder for transmission over an AWGN channel is a lower bound on the performance of the receiver performing turbo equalization. This bound can be overcome by introducing a precoder [42], which makes the "inner code," i.e., the conactenation of the precoder plus the ISI channel, recursive.

The EXIT chart analysis tool was shown to provide significant insight into the behavior of these iterative algorithms. This analysis also enables the selection of appropriate equalization methods and error correction codes for a given scenario, such as the channel response, the SNR, and complexity constraints.

References

[1] J. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
[2] S. Haykin, *Communication Systems*, 3rd ed. New York: Wiley, 1994.
[3] S. Lin and J. J. Costello, *Error Control Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
[4] G. Forney, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 363–378, May 1972.
[5] L. R. Bahl *et al.*, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.
[6] Y. Li, B. Vucetic, and Y. Sato, "Optimum soft-ouput detection for channels with intersymbol interference," *IEEE Trans. Inform. Theory*, vol. 41, pp. 704–713, May 1995.
[7] W. Koch and A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time varying intersymbol interference," in *Proc. of the IEEE Global Telecomm. Conf.*, Dec. 1990, pp. 1679–1684.
[8] D. Yellin, A. Vardy, and O. Amrani, "Joint equalization and coding for intersymbol interference channels," *IEEE Trans. Inform. Theory*, vol. 43, pp. 409–425, Mar. 1997.
[9] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Int. Conf. on Communications*, Geneva, Switzerland, May 1993.
[10] S. Benedetto *et al.*, "Serial concatenated trellis coded modulation with iterative decoding: Design and performance," in *Proc. IEEE Global Telecomm. Conf.*, Nov. 1997.
[11] S. Benedetto *et al.*, "Parallel concatenated trellis coded modulation," in *Proc. IEEE Int. Conf. Communications*, June 1996.
[12] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July 1999.
[13] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. on Turbo Codes*, Brest, France, Sept. 1997, pp. 1–11.
[14] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271, Oct. 1996.
[15] C. Heegard and S. Wicker, *Turbo Coding*. Boston, MA: Kluwer, 1999.
[16] C. Douillard *et al.*, "Iterative correction of intersymbol interference: Turbo equalization," *Eur. Trans. Telecommun.*, vol. 6, pp. 507–511, Sept.–Oct. 1995.
[17] G. Bauch and V. Franz, "A comparison of soft-in/soft-out algorithms for 'turbo detection'," in *Proc. Int. Conf. Telecomm.*, June 1998, pp. 259–263.
[18] A. Anastasopoulos and K. Chugg, "Iterative equalization/decoding for TCM for frequency-selective fading channels," in *Conf. Rec. 31th Asilomar Conf. Signals, Systems & Computers*, vol. 1, Nov. 1997, pp. 177–181.

[19] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE Global Telecomm. Conf.*, 1989, pp. 1680–1686.
[20] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," in *Proc. IEEE Global Telecomm. Conf.*, vol. 2, Nov. 1997, pp. 639–643.
[21] M. Toegel, W. Pusch, and H. Weinrichter, "Combined serially concatenated codes and turbo-equalization," in *Proc. 2nd Int. Symp. on Turbo Codes*, Brest, France, Sept. 2000, pp. 375–378.
[22] S. Ariyavisitakul and Y. Li, "Joint coding and decision feedback equalization for broadband wireless channels," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1670–1678, Dec. 1998.
[23] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes*, Brest, France, Sept. 1997, pp. 96–102.
[24] D. Raphaeli and A. Saguy, "Linear equalizers for Turbo equalization: A new optimization criterion for determining the equalizer taps," in *Proc. 2nd Int. Symp. Turbo codes*, Brest, France, Sept. 2000, pp. 371–374.
[25] Z. Wu and J. Cioffi, "Turbo decision aided equalization for magnetic recording channels," in *Proc. Global Telecomm. Conf.*, Dec. 1999, pp. 733–738.
[26] A. Berthet, R. Visoz, and P. Tortelier, "Sub-optimal Turbo-detection for coded 8-PSK signals over ISI channels with application to EDGE advanced mobile system," in *Proc. 11th Int. Symp. Personal, Indoor, and Mobile Radio Communications*, vol. 1, Sept. 2000, pp. 151–157.
[27] M. Tüchler, A. Singer, and R. Koetter, "Minimum mean square error equalization using *a priori* information," *IEEE Trans. Signal Processing*, vol. 50, pp. 673–683, Mar. 2002.
[28] H. Poor, *An Introduction to Signal Detection and Estimation*, 2nd ed. New York: Springer-Verlag, 1994, pp. 221–229.
[29] M. Tüchler, "Iterative equalization using priors," Master's thesis, University of Illinois, Urbana-Champaign, IL, 2000.
[30] M. Tüchler and J. Hagenauer, "Turbo equalization using frequency domain equalizers," in *Proc. Allerton Conf.*, Monticello, IL, Oct. 2000, pp. 1234–1243.
[31] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollard, "Serial concatenation of interleaved codes: Performance analysis design, and iterative decoding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 909–926, May 1998.
[32] S. Benedetto and G. Mondorsi, "Unveiling Turbo codes: Some results on parallel concatenated coding schemes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 409–428, Mar. 1996.
[33] T. Richardson, "The geometry of Turbo decoding dynamics," *IEEE Trans. Inform. Theory*, vol. 46, pp. 9–23, Jan. 2000.
[34] D. Agrawal and A. Vardy, "The Turbo decoding algorithm and its phase trajectories," *IEEE Trans. Inform. Theory*, vol. 47, pp. 699–722, Feb. 2001.
[35] S. ten Brink, "Convergence of iterative decoding," *Electron. Lett.*, vol. 35, pp. 806–808, May 1999.
[36] ——, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 40, pp. 1727–1737, Oct. 2001.
[37] T. Richardson and R. Urbanke, "The capacity of low density parity-check codes under message passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.
[38] K. Narayanan, "Effect of precoding on the convergence of Turbo equalization for partial response channels," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 686–698, Apr. 2001.
[39] H. Gamal and A. Hammons, Jr., "Analyzing the Turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 671–686, Feb. 2001.
[40] S. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density-parity-check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 657–670, Feb. 2001.
[41] D. Divsalar, S. Dolinar, and F. Pollara, "Serial Turbo trellis coded modulation with rate-1 inner code," in *Proc. Int. Symp. Information Theory*, 2000, p. 194.
[42] I. Lee, "The effect of a precoder on serially concatenated coding systems with an ISI channel," *IEEE Trans. Commun.*, vol. 49, pp. 1168–1175, July 2001.
[43] R. Ramamruthy and W. Ryan, "Convolutional double accumulate codes (or double Turbo DPSK)," *IEEE Commun. Lett.*, vol. 5, pp. 157–159, Apr. 2001.
[44] P. Massey, O. Takeshita, and D. Costello, "Contradicting a myth: Good Turbo codes with large memory order," in *Proc. Int. Symp. Information Theory*, 2000, p. 122.
[45] S. ten Brink, "Iterative decoding trajectories of parallel concatenated codes," in *Proc. 3rd IEEE/ITG Conf. Source and Channel Coding*, Munich, Germany, Jan. 2000, pp. 75–80.
[46] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proc. 2nd Eur. Personal Mobile Comm. Conf.*, Sept./Oct. 1997, pp. 307–312.

Michael Tüchler was born on October 6, 1975, in Hildburghausen, Germany. He studied electrical engineering from 1995 to 1997 at the Technical University Ilmenau, Germany, and from 1997 to 1998 at the Technical University Munich, Germany. He received the M.S. degree from the University of Illinois at Urbana-Champaign, in 2000. He is currently working toward the Ph.D. degree at the Institute for Communications Engineering at the Technical University Munich.

In 2001, he was a Visiting Scientist at the IBM Zurich Research Laboratory, Switzerland. His research interests include coding theory and equalization with application to wireless and data storage communication systems.

Ralf Koetter was born on October 10, 1963, in Königstein, Germany. He received the Diploma in electrical engineering from the Technical University Darmstadt, Germany in 1990 and the Ph.D. degree from the Department of Electrical Engineering at Linköping University, Sweden.

From 1996 to 1997, he was a Visiting Scientist at the IBM Almaden Research Laboratory, San Jose, CA. He was a Visiting Assistant Professor at the University of Illinois at Urbana/Champaign and Visiting Scientist at CNRS in Sophia Antipolis, France, during the years 1997 to 1998. He joined the faculty of the University of Illinois at Urbana/Champaign in 1999 and is currently an Assistant Professor at the Coordinated Science Laboratory at the University. His research interests include coding and information theory and their application to communication systems.

Dr. Koetter served as Associate Editor for Coding Theory and Techniques from 1999 to 2001 for the IEEE TRANSACTIONS ON COMMUNICATIONS. In 2000, he started a term as Associate Editor for Coding Theory of the IEEE TRANSACTIONS ON INFORMATION THEORY. He received an IBM Invention Achievement Award in 1997 and an NSF CAREER Award in 2000, and an IBM Partnership Award in 2001.

Andrew C. Singer 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, Cambridge, in 1990, 1992, and 1996, respectively.

Since 1998, he has been on the faculty of the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign, where he is currently an Assistant Professor in the Electrical and Computer Engineering Department, and a Research Assistant Professor in the Coordinated Science Laboratory. During the academic year 1996, he was a Post-Doctoral Research Affiliate in the Research Laboratory of Electronics at MIT. From 1996 to 1998, he was a Research Scientist at Sanders, A Lockheed Martin Company, Manchester, NH. His research interests include statistical signal processing and communication, universal prediction and data compression, and machine learning.

Prof. 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.