Turbo Equalization via Constrained-Delay APP Estimation With Decision Feedback

Jaekyun Moon, *Fellow, IEEE,* and Farshid Rafiee Rad, *Student Member, IEEE*

***Abstract*—We consider turbo equalization for intersymbol interference (ISI) channels, wherein soft symbol decisions generated by the channel detector are iteratively exchanged with the outer error-correction decoder based on the turbo principle. Our work is based on low-complexity suboptimal soft-output channel detection using a constrained-delay (CD) *a posteriori* probability (APP) algorithm. Central to the proposed idea is the incorporation of effective decision-feedback schemes, which significantly reduce complexity while providing immunity against error propagation that typically plagues decision-feedback schemes. We observe that the effect of decision feedback is quite different on turbo equalization versus traditional, hard-decision-generating and noniterative equalization. In particular, we demonstrate that when the feedback scheme applied is inadequate for the given equalizer parameters and ISI condition, the extrinsic information generated by the equalizer becomes distinctly non-Gaussian, and the quality of soft information, as monitored by the trajectory of mutual information, fails to improve in the iterative process. We identify parameters of feedback-based CD-APP schemes that offer favorable complexity/performance tradeoffs, compared with existing turbo-equalization techniques.**

***Index Terms*—Equalization, intersymbol interference (ISI), iterative methods, maximum *a posteriori* (MAP) detection, turbo equalization.**

I. Introduction

INTERSYMBOL interference (ISI) arises in bandlimited data transmission. Channel-equalization techniques to combat ISI have been fairly well established, ranging from the maximum-likelihood sequence detector (MLSD) [1] to linear and decision-feedback equalization (DFE) techniques [2], [3], as well as various reduced-complexity MLSD approximations [4]–[6]. ISI equalization had traditionally focused on generating hard decisions, which then typically get passed to the outer decoder that attempts to correct erroneous bits that were made in the equalization/detection process. Notable exceptions to this general trend were the Bahl–Cocke–Jelinek–Raviv (BCJR) algorithm of [7] and, more recently, the soft-output Viterbi algorithm (SOVA) of [8], both of which were designed to provide "soft" decisions in the form of *a posteriori* probabilities (APPs) of transmitted symbols.

Paper approved by X. Wang, the Editor for Wireless Spread Spectrum of the IEEE Communications Society. Manuscript received October 29, 2003; revised June 15, 2004 and January 19, 2005. This paper was presented in part at the IEEE International Conference on Communications, Seoul, Korea, May 2005.

The authors are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: moon@ece.umn.edu; farshid@ece.umn.edu).

Digital Object Identifier 10.1109/TCOMM.2005.860057

With the emergence of turbo coding [9], and the subsequent interest in the broader turbo principle [10], [11], where the iterative processing of soft information takes the center stage, the detector's ability to generate soft decisions has become an important system design requirement. The turbo principle has first been applied to the channel ISI equalization problem in [12], where the channel detector and the outer error-correction coding (ECC) decoder exchange soft symbol decisions in an iterative fashion in an effort to improve the quality of the soft information. The techniques to combat ISI by generating and iteratively processing soft decisions in conjunction with an outer soft-input soft-output (SISO) decoder have since been dubbed collectively as turbo equalization. The performance results presented in [12] pointed to a great potential for this type of approach in ISI equalization. Complexity of the trellis-based soft-output detectors, such as the BCJR and SOVA algorithms, however, grows exponentially with the length of ISI. Thus, for channels with long ISI, a need arises to compromise performance with complexity, and a resort should be made to some low-complexity methods.

A particularly effective approach to low-complexity turbo equalization is discussed in [13]. There, a turbo-equalization technique based on soft output minimum mean-squared error (MMSE) linear equalization (LE) is discussed, leading to a time-varying filter structure. To avoid the high implementation complexity associated with a time-varying filter, the authors then suggest two time-invariant approximations that can be used in combination. The resulting equalizer performance and properties have been analyzed using the extrinsic information transfer (EXIT) chart analysis of [14]. A SISO version of the DFE is discussed in [15] and also in [13], but its performance is shown to be inferior to soft-output LE schemes [13]. It is also noted that an approach similar to turbo equalization is taken in [16] for interference cancellation in code-division multiple-access (CDMA) multiuser communication systems. Instantaneous soft-output MMSE filtering is used to reduce the complexity of maximum *a posteriori* (MAP) probability detection based on the BCJR algorithm.

The work presented in this paper is also an attempt to develop a low-complexity SISO detector for ISI channels that can be used in any turbo-equalization setting. Our proposed approach is based on derivation of a constrained-delay (CD) APP detector with decision feedback (DF). The notion of a CD is similar to that of the finite or sliding-window BCJR algorithms [17], [18], but the main conceptual difference is that the derivation of the CD-APP detector starts with a hard and explicit constraint of a fixed decision delay. Also, while a truncation of the processing window in the BCJR algorithm effectively fixes the decision delay, it does not reduce the number of trellis nodes, whereas

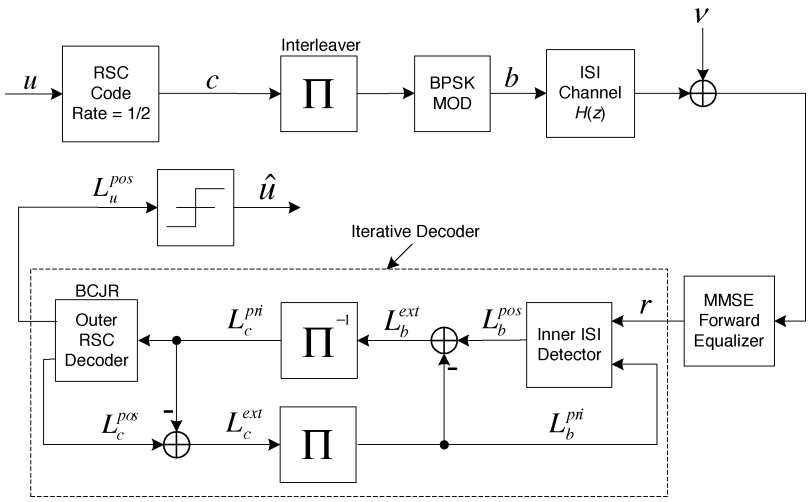

Fig. 1. Coded data-transmission system model.

in our proposed setting, the decision-delay parameter directly affects the size of the trellis. Abend and Fricthman appear to be the first to have derived a MAP detector with a prescribed decision-delay constraint [19]. The work presented in [20] and [21] discusses a CD MAP technique based on tree search and hard DF (HDF). The tree formulation is used to simplify the arbitration process in the presence of certain code constraints, as well as to point to some interesting implementation options when the goal is to release hard decisions. Soft DF (SDF) in conjunction with a CD-MAP scheme is discussed in a nonturbo setting in [22]. The method presented in [23] can also be viewed as a truncated APP technique formulated in the context of multiuser detection/layered space–time processing, with SDF applied in the spatial domain. We also mention that the notion of "path partitioning" has been introduced in [24], and illustrates relationships among some of exiting trellis-based soft- and hard-decision algorithms. For the interested reader, an extensive list of references is also provided in [24] on soft-decision algorithms available in the literature.

In this paper, we first provide a conceptually simple, trellis-based formulation of an optimal CD-APP algorithm, which has exponential complexity in decision delay rather than the ISI length. We are mainly interested in situations where decision delay is smaller than the extent of ISI, so that the resulting complexity is lower than the optimal BCJR algorithm. We emphasize that in a turbo-equalization setting, a comfortably small value of decision delay (which can be significantly smaller than the ISI length) can provide an excellent complexity/performance tradeoff, provided that an appropriate DF scheme is employed. In noniterative systems where soft *a priori* information does not improve through successive iterations, a considerable performance degradation would result if decision delay is chosen to be less than the extent of ISI, as verified in [25].

Some sort of DF is necessary to compensate for tail ISI if the decision delay is shorter than the extent of ISI. To this end, we proceed to investigate various DF schemes used in conjunction with CD-APP, in the context of turbo equalization. We observe that the EXIT chart analysis based on the Gaussianity assumption of *a priori* probability entering a constituent equalizer/decoder may not predict the performance of DF-based turbo equalizers for certain feedback schemes. In such cases, we note that the trajectory of mutual information, which depicts how the quality of soft information evolves through the long chain of successive equalization and decoding steps in the iterative process, fails to move up and follow the path outlined by the EXIT charts. We further observe that when this happens, the distribution of the soft information becomes distinctly non-Gaussian. We quantify this effect using a normalized kurtosis. Using average trajectories and kurtosis, we evaluate different feedback strategies for ISI channels with varying degrees of ISI strength. The feedback schemes we investigated range from simple HDF to "full-blown" SDF with exhaustive enumeration of uncompensated ISI patterns incorporating their probability measures. We shall also compare feedback-based CD-APP algorithms with the LE approach of [13], as well as the optimal BCJR algorithm.

The organization of the paper is as follows. In Section II, we describe the communication system model used in this paper. This section briefly touches upon the turbo-equalization concept. Section III includes the trellis-based derivation of a CD detector, as well as development of SDF schemes. Section IV provides a brief review of the EXIT chart analysis. Section V presents numerical results for two ISI channels with different levels of severity, highlighting improved performance/complexity tradeoffs associated with the proposed scheme in relation to existing approaches. Finally, conclusions are drawn in Section VI.

II. Communication System Model

The communication system model is shown in Fig. 1. First an outer recursive systematic convolutional (RSC) code encodes the source information bits $\boldsymbol{u}$ (sequences are sometimes represented as vectors and denoted by bold letters). Next, an interleaver scrambles the coded bit sequence $\boldsymbol{c}$ and removes the memory introduced by the encoder. The binary phase-shift keying (BPSK)-modulated output of the interleaver, $\boldsymbol{b}$, is then transmitted through the ISI channel. Finally, additive white

Gaussian noise (AWGN) with variance $N_0/2$ gets added. The received signal at time k is modeled as

$$r_k = \sum_{i=0}^{I} h_i b_{k-i} + \nu_k \tag{1}$$

where h_i denotes the impulse response of the ISI channel whose memory length is I, and where ν_k represents samples of AWGN. Note that the block length of all the sequences specified in Fig. 1 is dictated by the interleaver length.

At the receiver side, a (reasonably short) MMSE forward equalizer, very much like that of a DFE system, is first used to phase-equalize the channel and push its energy to the front [26]. If the given ISI channel is already minimum phase, there is certainly no need for phase equalization. At the output of the forward equalizer, we have an effective channel response which is (nearly) minimum phase, so that a small value of decision delay in the subsequent CD-APP detector covers a good proportion of the total ISI energy. In essence, we trade the small linear complexity of a short-phase equalizer for a large saving in the exponential complexity of the subsequent CD-APP detector. The output of the forward equalizer, $\boldsymbol{r}$, feeds the block labeled "iterative decoder" which consists of an inner ISI detector (based on the CD-APP detector) and an outer RSC decoder (based on the BCJR algorithm), connected through an interleaver and a deinterleaver. The quantities $\boldsymbol{L}^{\text{pri}}$, $\boldsymbol{L}^{\text{pos}}$, and $\boldsymbol{L}^{\text{ext}}$ refer to the a priori, a posteriori, and extrinsic log-likelihood ratios (LLRs), respectively. Note that there is a one-to-one correspondence between bit sequences in the transmitter and LLR sequences in the receiver. This is reflected in the subscripts of the LLRs, e.g., $\boldsymbol{L}_b^{\text{ext}}$ refers to the extrinsic LLR sequence corresponding to bit sequence $\boldsymbol{b}$.

Based on the channel observations $\boldsymbol{r}$ and the *a priori* information $\boldsymbol{L}_b^{\text{pri}}$ coming from outer decoder, the inner detector computes the *a posteriori* values $\boldsymbol{L}_b^{\text{pos}}$. What is passed along to the outer decoder is $\boldsymbol{L}_b^{\text{ext}}$, which is $\boldsymbol{L}_b^{\text{pos}}$ less the $\boldsymbol{L}_b^{\text{pri}}$ to prevent the *a priori* information from cycling around. Since the outer decoder does not have access to direct channel observations, we do not subtract out their effect from $\boldsymbol{L}_b^{\text{pos}}$. More formally, at time $k-\tau$, we can write

$$L_{b_{k-\tau}}^{\text{pos}} = \log \frac{P(b_{k-\tau}=+1|r_1,\ldots,r_k)}{P(b_{k-\tau}=-1|r_1,\ldots,r_k)}. \tag{2}$$

The LLR in (2) reflects the motivation behind a CD detector with decision delay τ: *Given samples $r_1, r_2, \ldots, r_k$, what is the best we can say about the transmitted bit at time $k-\tau$?* Using Bayes' rule, (2) can be rewritten as

$$L_{b_{k-\tau}}^{\text{pos}} = \log \frac{P(r_1,\ldots,r_k|b_{k-\tau}=+1)}{P(r_1,\ldots,r_k|b_{k-\tau}=-1)} + \log \frac{P(b_{k-\tau}=+1)}{P(b_{k-\tau}=-1)} \tag{3}$$

where the first and the second terms are denoted by $\boldsymbol{L}_{b_{k-\tau}}^{\text{ext}}$ and $\boldsymbol{L}_{b_{k-\tau}}^{\text{pri}}$, respectively. After being deinterleaved, $\boldsymbol{L}_b^{\text{ext}}$ becomes $\boldsymbol{L}_c^{\text{pri}}$ and serves as the input to the outer decoder. As in the case of the ISI channel detector, *a priori* information $\boldsymbol{L}_c^{\text{pri}}$ is subtracted out from $\boldsymbol{L}_c^{\text{pos}}$ for the same reason.

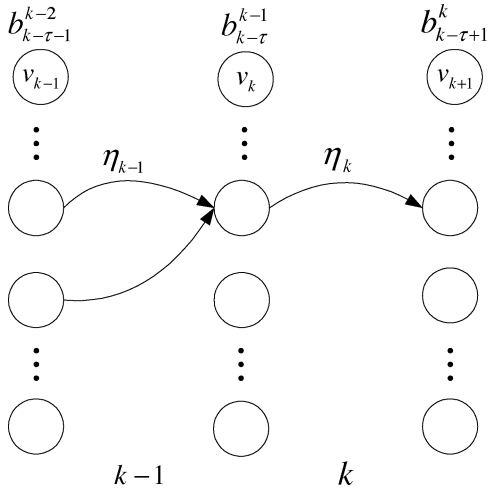

Fig. 2. States and state transitions of a generic CD trellis at time units $k-1$ and k.

Similar to (3), we can decompose $L_{c_k}^{\text{pos}}$ as

$$L_{c_k}^{\text{pos}} = \log \frac{P\left(L_{c_1}^{\text{pri}},\ldots,L_{c_N}^{\text{pri}}|c_k=1\right)}{P\left(L_{c_1}^{\text{pri}},\ldots,L_{c_N}^{\text{pri}}|c_k=0\right)} + \log \frac{P(c_k=1)}{P(c_k=0)} \tag{4}$$

where N is the length of the transmitted sequence. Note that in (3), we look ahead a finite number of symbols, while in (4), we require reception of the whole sequence. This is essentially the difference between constrained and unconstrained MAP detection.

III. Constrained-Delay Detector Design

In this section, we present a CD-APP detector with various DF schemes. In the following derivations, we assume BPSK modulation. Extension to larger constellations is straightforward.

A. Algorithm Description

A simple description of a CD-APP detector is possible by constructing a trellis diagram with states v_k and state transitions η_k, as follows:

$$v_k \triangleq b_{k-\tau}^{k-1} \tag{5}$$

$$\eta_k \triangleq \{v_k, v_{k+1}\} = b_{k-\tau}^{k} \tag{6}$$

where b_j^i denotes the transmitted bits $b_j, b_{j+1}, \ldots, b_i$. Fig. 2 illustrates the states and state transitions of a generic CD trellis at time units $k-1$ and k. The goal is to find the joint probability density $f(b_{k-\tau}, r_1^k)$, which can be stated as the following sum:

$$f\left(b_{k-\tau}=+1, r_1^k\right) = \sum_{\eta_k \in \mathcal{B}^+} f\left(\eta_k, r_1^k\right) \tag{7}$$

where $\mathcal{B}^+$ denotes the set of all transitions η_k such that $b_{k-\tau}=+1$. The joint probability density $f(b_{k-\tau}=-1, r_1^k)$ is similarly defined, with $\mathcal{B}^+$ replaced by $\mathcal{B}^-$, the set of all transitions η_k

such that $b_{k-\tau}=-1$. We now recursively express $f(\eta_k, r_1^k)$ in terms of $f(\eta_{k-1}, r_1^{k-1})$

$$\begin{aligned} f\left(\eta_k, r_1^k\right) &= f\left(v_k, b_k, r_1^{k-1}, r_k\right) \\ &= f\left(r_k, b_k | v_k, r_1^{k-1}\right) f\left(v_k, r_1^{k-1}\right) \\ &= f\left(r_k | b_k, v_k, r_1^{k-1}\right) P\left(b_k | v_k, r_1^{k-1}\right) f\left(b_{k-\tau}^{k-1}, r_1^{k-1}\right) \\ &= P(b_k) l(\eta_k) \sum_{\eta_{k-1} \in \mathcal{D}(v_k)} f\left(\eta_{k-1}, r_1^{k-1}\right) \end{aligned} \tag{8}$$

where $\mathcal{D}(v_k)$ is the set of all η_{k-1} leading to v_k, and $l(\eta_k)$ is the likelihood function, defined as

$$l(\eta_k) \triangleq f\left(r_k | \eta_k, r_1^{k-1}\right). \tag{9}$$

Note that we assumed independent b_k's in obtaining the last equality in (8). This is justified because the bit sequence $\boldsymbol{b}$ is a "randomized" version of the bit sequence $\boldsymbol{c}$. In turbo equalization, $P(b_k)$ in (8) is to be replaced by the *a priori* information passed back from the outer soft-output decoder. A similar recursion equation can be derived on the joint probability density of the state variable and the up-to-date observation sequence, but it can be shown that the size of the trellis needs to be doubled, requiring twice more likelihood-function calculations. The algorithm outlined in [22] can be viewed as the recursive computation of the joint probability densities of state variables.

When $\tau \geq I$, the algorithm of (8) and (9) can be viewed as a recasting of the optimal CD APP detector of [19] as a trellis-based algorithm. With $\tau \geq I$, η_k completely specifies b_{k-I}^k and, thus, for signal-independent noise, r_k is independent of r_1^{k-1}. Accordingly, we can drop r_1^{k-1} in (9). It is also easy to see that for the special case of $\tau = I$, this algorithm coincides with the forward processing of the BCJR algorithm. We do not consider this algorithm to be very useful for $\tau \geq I$, since the size of the resulting trellis (2^τ) is no smaller than that of the BCJR algorithm (2^I). This algorithm will be interesting, and have a chance to provide favorable performance/complexity tradeoffs only when $\tau < I$.

For $\tau < I$, however, exact computation of $l(\eta_k)$ in (9) is problematic, as the ISI terms are not completely specified in η_k. Past hard decisions can be used in completely defining the signal portion of r_k, as suggested in [19]. If we do assume correct past hard decisions, then the computation of the likelihood function at time k can be carried out, under Gaussian noise, as

$$l(\eta_k) = \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{(r_k - \mu_k)^2}{N_0}\right) \tag{10}$$

$$\mu_k = \sum_{i=0}^{\tau} h_i b_{k-i} + \sum_{i=\tau+1}^{I} h_i \hat{b}_{k-i} \tag{11}$$

where μ_k is the noiseless signal and $\hat{b}_k$ represents past *hard* bit decisions. A serious drawback of this approach is that feeding back past decisions induces error propagation, if the energy is significant in ISI. The algorithm of [20] and [21] is essentially based on this idea, but uses a fixed-delay tree construction, wherein the path/branch arbitration process simplifies in the presence of certain code constraints, and which leads to certain low-complexity implementation options when the equalizer's goal is to generate hard decisions.

In an effort to reduce unspecified ISI terms in the calculation of $l(\eta_k)$, we modify the algorithm of (8) and (9) by tracing the trellis one step backward, i.e.,

$$\begin{aligned} f\left(\eta_k, r_1^k\right) &= \sum_{b_{k-\tau-1}} f\left(b_{k-\tau-1}, v_k, b_k, r_1^{k-1}, r_k\right) \\ &= \sum_{b_{k-\tau-1}} f\left(r_k, b_k | b_{k-\tau-1}, v_k, r_1^{k-1}\right) \\ &\quad \times f\left(b_{k-\tau-1}, v_k, r_1^{k-1}\right) \\ &= \sum_{b_{k-\tau-1}} f\left(r_k | b_k, \eta_{k-1}, r_1^{k-1}\right) \\ &\quad \times P\left(b_k | \eta_{k-1}, r_1^{k-1}\right) f\left(\eta_{k-1}, r_1^{k-1}\right) \\ &= P(b_k) \sum_{\eta_{k-1} \in \mathcal{D}(v_k)} l(\eta_k, \eta_{k-1}) \\ &\quad \times f\left(\eta_{k-1}, r_1^{k-1}\right) \end{aligned} \tag{12}$$

where $l(\eta_k, \eta_{k-1})$ is a modified likelihood function defined as

$$l(\eta_k, \eta_{k-1}) \triangleq f\left(r_k | \eta_{k-1}, \eta_k, r_1^{k-1}\right). \tag{13}$$

Again assuming correct past hard decisions, $l(\eta_k, \eta_{k-1})$ can be calculated similarly to (10), but with μ_k computed as

$$\mu_k = \sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{I} h_i \hat{b}_{k-i}. \tag{14}$$

Compared with (9), the condition in the right-hand side (RHS) of (13) specifies one more ISI term, reducing the number of unknown ISI terms by one in determining the signal portion of r_k. Tracing back further in the trellis results in a recursion equation based on the computation of a likelihood function involving multiple preceding observation samples in every cycle; but this does not help in reducing the number of unspecified ISI terms in the likelihood-function calculation. The level of complexity required to implement (12) and (13) is about double that required for (8) and (9), as there are two likelihood functions (for binary inputs) to be computed for each transition in (12); the algorithm of (12) and (13) with delay τ is expected to perform as well as the algorithm of (8) and (9) with delay $\tau + 1$, but would not require the corresponding increase in the number of joint probabilities that need be carried to the next cycle.

Whether (12) and (13) or (8) and (9) are implemented, the use of hard decisions to cancel past ISI terms may cause substantial error propagation under severe ISI and/or small τ. To alleviate this issue, we now discuss ways to incorporate SDF in CD APP detection. In the following, we shall focus on the algorithm of (12) and (13) as a base algorithm in constructing SDF schemes.

B. Soft DF

One heuristic way of generating soft decisions is to simply use the hard decisions weighted by their APP of occurrence as

$$\hat{b}_k^S = \hat{b}_k \frac{\exp\left(\frac{\hat{b}_k+1}{2} L_{b_k}^{\text{pos}}\right)}{1 + \exp\left(L_{b_k}^{\text{pos}}\right)} \tag{15}$$

where the superscript S in $\hat{b}_k^S$ emphasizes that the decisions are *soft*. The likelihood function can be calculated by (10) and (14), but with $\hat{b}_k$ replaced by $\hat{b}_k^S$. This SDF scheme feeds back the mean sequence of the past decisions according to their APPs; we shall refer to this SDF scheme as mean SDF (MSDF).

While MSDF is easy to implement, as we shall show later, it may still exhibit error propagation in certain ISI situations, especially in conjunction with reduced-complexity detector processing. Let us discuss a more elaborate SDF scheme that makes the system more or less immune to error propagation, even in severe ISI situations.

The likelihood function given in (13) can be extended using the following sum:

$$\begin{aligned} l(\eta_k,\eta_{k-1}) &= \sum_{b_{k-M}^{k-\tau-2}} f\left(b_{k-M}^{k-\tau-2}, r_k|\eta_k,\eta_{k-1},r_1^{k-1}\right) \\ &= \sum_{b_{k-M}^{k-\tau-2}} P\left(b_{k-M}^{k-\tau-2}|\eta_k,\eta_{k-1},r_1^{k-1}\right) \\ &\quad \times f\left(r_k|b_{k-M}^{k-\tau-2},\eta_k,\eta_{k-1},r_1^{k-1}\right) \\ &= \sum_{b_{k-M}^{k-\tau-2}} P\left(b_{k-M}^{k-\tau-2}|\eta_{k-1},r_1^{k-1}\right) f\left(r_k|b_{k-M}^{k}\right) \\ &= \sum_{b_{k-M}^{k-\tau-2}} \left(\prod_{l=\tau+2}^{M} P\left(b_{k-l}|r_1^{k-l+\tau}\right)\right) \\ &\quad \times f\left(r_k|b_{k-M}^{k}\right) \end{aligned} \tag{16}$$

where the third equality arises because given b_{k-I}^k and uncorrelated noise, r_k and r_1^{k-1} are conditionally independent, and the conditional probability of $b_{k-I}^{k-\tau-2}$ does not depend on η_k in the absence of the observation sample r_k. In obtaining the last line, the probability $P(b_{k-I}^{k-\tau-2}|\eta_{k-1},r_1^{k-1})$ conditioned on a particular state transition η_{k-1} is approximated based on the set of available past APPs $\{P(b_{k-\tau-2}|r_1^{k-2}), P(b_{k-\tau-3}|r_1^{k-3}), \ldots, P(b_{k-I}|r_1^{k-I+\tau})\}$ that are not dependent upon η_{k-1}. Note that computation of the likelihood function given in (9) would result in $l(\eta_k)=\sum_{b_{k-I}^{k-\tau-1}} P(b_{k-I}^{k-\tau-1}|r_1^{k-1}) f(r_k|b_{k-I}^k)$, which can then be more straightforwardly approximated as $\sum_{b_{k-I}^{k-\tau-1}}(\prod_{l=\tau+1}^{I} P(b_{k-l}|r_1^{k-l+\tau}))f(r_k|b_{k-I}^k)$.

The conditional probability density function (pdf) $f(r_k|b_{k-I}^k)$ is Gaussian with mean

$$\mu_j = \sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{I} h_i b_{k-i}^{(j)} \tag{17}$$

and variance $\sigma^2 = N_0/2$. Note that $j \in \{0, 2^{I-\tau-1}-1\}$ represents the decimal representation of different binary patterns corresponding to $b_{k-I}^{k-\tau-2}$, and $b_k^{(j)}$ refers to the kth BPSK-modulated symbol of the jth sequence. The likelihood function given in (16) is, therefore, a weighted sum of $2^{I-\tau-1}$ Gaussian pdfs, expressed as

$$l(\eta_k,\eta_{k-1}) = \sum_{j=0}^{2^{I-\tau-1}-1} P_j \mathcal{N}\left(\mu_j, \frac{N_0}{2}\right) \tag{18}$$

where $P_j \triangleq \prod_{l=\tau+2}^{I} P(b_{k-l}^{(j)}|r_1^{k-l+\tau})$. We call this exhaustive SDF (ESDF), as it relies on exhaustive enumeration of all unspecified ISI patterns and incorporation of their probability measures. While this approach provides essentially the optimal way of feeding back past soft decisions, the complexity associated with evaluating (18) will be high for the cases where τ is significantly smaller than I, which is exactly the situation that we have set out to address.

Various reduced-complexity algorithms simplifying (18) are possible. For example, we can use

$$l(\eta_k,\eta_{k-1}) = \sum_{j\in\mathcal{J}} P_j \mathcal{N}\left(\mu_j, \frac{N_0}{2}\right) \tag{19}$$

where $\mathcal{J}$ is a small subset of $\{0,\ldots,2^{I-\tau-1}-1\}$ for which P_j is the largest.

Another possibility is to simply focus on getting the overall mean μ_T and overall variance σ_T^2 of $l(\eta_k,\eta_{k-1})$. This amounts to forcing $l(\eta_k,\eta_{k-1})$ to be approximated as a single Gaussian pdf, i.e., $N(\mu_T,\sigma_T^2)$. Note that there is no evidence of this approximation being reasonable at this point; we are simply forcing it to simplify SDF implementation. Later, however, performance analysis will indicate that this approximation is indeed effective in providing good performance/complexity tradeoffs. The overall mean and variance can be written as

$$\mu_T = \sum_j P_j \mu_j \tag{20}$$

$$\sigma_T^2 = \sum_j P_j \left(\int (r-\mu_T)^2 \mathcal{N}(r,\mu_T,\sigma^2)dr\right). \tag{21}$$

It is shown in the Appendix that (20) and (21) can be expressed as

$$\mu_T = \sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{I} h_i \left(2P_{k-i}^{(+)} - 1\right) \tag{22}$$

$$\sigma_T^2 = \sigma^2 + 4\sum_{i=\tau+2}^{I} h_i^2 \left(P_{k-i}^{(+)} \cdot P_{k-i}^{(-)}\right) \tag{23}$$

where $P_{k-i}^{(+)} \triangleq P(b_{k-i}=+1|r_1^{k-i+\tau})$ and $P_{k-i}^{(-)} \triangleq P(b_{k-i}=-1|r_1^{k-i+\tau})$. Once the overall mean and variance are found, the likelihood function is calculated as

$$l(\eta_k,\eta_{k-1}) = \frac{1}{\sqrt{2\pi\sigma_T^2}} \exp\left(-\frac{(r_k-\mu_T)^2}{2\sigma_T^2}\right). \tag{24}$$

This method of SDF, which is based on finding the overall mean and variance of the likelihood function as given in (22) and (23), ends up equivalent to the SDF schemes of [22] and [23], which is based on a direct modeling of unspecified interference terms as a Gaussian random variable (RV), and calculation of the mean and variance of the overall noise plus interference using available probability measures of the symbols, giving rise to the unspecified ISI terms. The derivation presented here, however, clearly shows where the approximation is made (and thus, the source of potential optimality loss) in eventually incorporating the APPs of past decisions in the computation of the likelihood

function. Our derivation also provides a link between the ESDF of (18), which represents essentially the optimal way of DF, and the SDF scheme of (24). Finally, the formulation of (18) easily points to other possible SDF variations, as suggested by (19). As this SDF scheme results from a Gaussian pdf approximation of the likelihood function, we shall call it Gaussian SDF (GSDF).

C. CD Detection in Log Domain

At the core of the log-domain algorithm is the following approximation to the log of the sum of two exponentials [27], [28]:

$$a_M \approx \log(e^{a_1} + e^{a_2}) \tag{25}$$

where $a_M \triangleq \max(a_1, a_2)$. This approximation is accurate if $|a_2 - a_1| \gg 0$, otherwise, exact computation is possible if we pay the complexity price of calculating a correction term and adding it to (25) as

$$\log(e^{a_1} + e^{a_2}) = a_M + \delta(a_1, a_2) \tag{26}$$

where $\delta(a_1, a_2) = \log(1 + e^{-|a_1 - a_2|})$ is the correction term and usually computed using a small lookup table. The term originally given to the left-hand side of (26) is $\max^*(a_1, a_2)$ [27]. More generally, we can implement $\max^*(a_1, \ldots, a_L) = \log(\sum_{i=1}^{L} e^{a_i})$, $L > 2$ by a structure similar to a binary tree whose leaves are the exponents a_i, $1 \le i \le L$, and whose nodes are either max or $\max^*$ operations, depending on the level of accuracy we seek. The final result will be available at the root of the tree.

The goal is to calculate $L^{\text{pos}}_{b_{k-\tau}}$ as given in (2). Replacing the conditional pdfs of (2) with the joint pdfs and using (7) in the resulting formula, we have

$$\begin{aligned} L^{\text{pos}}_{b_{k-\tau}} &= \log \frac{\sum_{\eta_k \in \mathcal{B}^+} f\left(\eta_k, z_1^k\right)}{\sum_{\eta_k \in \mathcal{B}^-} f\left(\eta_k, z_1^k\right)} \\ &= \log \sum_{\eta_k \in \mathcal{B}^+} e^{\log f(\eta_k, z_1^k)} - \log \sum_{\eta_k \in \mathcal{B}^-} e^{\log f(\eta_k, z_1^k)} \end{aligned} \tag{27}$$

Defining $\Lambda(\eta_k) \triangleq \log f(\eta_k, z_1^k)$ and applying the $\max^*$ operator to the two sums of exponentials [27], [28], we can write (27) as

$$L^{\text{pos}}_{b_{k-\tau}} = \max^*_{\eta_k \in \mathcal{B}^+} \Lambda(\eta_k) - \max^*_{\eta_k \in \mathcal{B}^-} \Lambda(\eta_k). \tag{28}$$

The $\max^*$ operation can be replaced by the max operation to reduce hardware complexity with some loss of performance. Taking the log of both sides of (12) and defining $\lambda(\eta_k, \eta_{k-1}) \triangleq \log l(\eta_k, \eta_{k-1})$, $\Lambda(\eta_k)$ is simply written as

$$\Lambda(\eta_k) = \log P(b_k) + \max^*_{\eta_{k-1} \in \mathcal{D}(v_k)} \{\Lambda(\eta_{k-1}) + \lambda(\eta_k, \eta_{k-1})\} \tag{29}$$

where by using $L^{\text{pri}}_{b_k}$, we can write

$$\log P(b_k) = \frac{b_k + 1}{2} L^{\text{pri}}_{b_k} - \max^*\left(0, L^{\text{pri}}_{b_k}\right) \tag{30}$$

where the second term on the RHS is independent of b_k, and so will be cancelled out in (28). It now remains to calculate $\lambda(\eta_k, \eta_{k-1})$ in (29). Inspecting (10) and (24), we see that the log of the likelihood function for HDF and SDF takes the simple quadratic form $-(r_k - \mu)^2/2\sigma^2$ but with different means and variances. The term $-(1/2)\log(\pi\sigma^2)$ is constant, and so again is cancelled out in (28).

D. Pseudocode Description of CD-HDF/GSDF

Let s be the decimal representation of the binary pattern $b_{k-\tau}^k$ specifying the state transition as in (6). We use the notation $\eta_{k,s}$ to refer to the state transition s at time k. The total number of state transitions in the CD trellis with decision delay τ is $S = 2^{\tau+1}$. Let r_1^N and $L^{\text{pri}}_{b_1^N}$ be the received samples and the *a priori* probabilities on b from time 1 to time N, respectively. Assuming the channel state is initially the all-zero state, the log-domain CD detection algorithm is summarized as follows.

1) For $k = -M + 1$ to 0,
   $\hat{b}_k = -1$, $L^{\text{pos}}_{b_k} = -\infty$,
   $P_k^{(-)} = 1$, $P_k^{(+)} = 0$.
2) $k = 0$, $\Lambda(\eta_{k,0}) = 0$, $\Lambda(\eta_{k,i}) = -\infty$, $0 < i < S$
3) For $k = 1$ to N
   a) For $s = 0$ to $S - 1$
      i) v_k = Starting state of $\eta_{k,s}$.
      ii) $\mathcal{D}(v_k)$ = All state transitions, $\eta_{k-1,t}$, leading to v_k.
      iii) For all $\eta_{k-1,t}$ in $\mathcal{D}(v_k)$
         A) Obtain $b_{k-\tau-1}^k$ from $\eta_{k,s}$ and $\eta_{k-1,t}$.
         B) If HDF: Calculate mean as in (11) and call it μ and set $\sigma^2 = N_0/2$.
         C) If SDF: Calculate mean and variance as in (22) and (23), respectively, and call them μ and σ^2.
         D) $\lambda(\eta_{k,s}, \eta_{k-1,t}) = -(r_k - \mu)^2/2\sigma^2$.
      iv) Calculate $\Lambda(\eta_{k,s})$ using (29) and (30).
   b) If $k > \tau$,
      i) Calculate $L^{\text{pos}}_{b_{k-\tau}}$ as in (28).
      ii) If HDF:
         A) $\hat{b}_{k-\tau} = \text{sgn}(L^{\text{pos}}_{b_{k-\tau}})$.
      iii) If SDF:
         A) $P_{k-\tau}^{(-)} = 1/(1 + \exp(L^{\text{pos}}_{b_{k-\tau}}))$.
         B) $P_{k-\tau}^{(+)} = 1 - P_{k-\tau}^{(-)}$.
4) Done. Go back to step 1) to process the next block.

Note that the past soft/hard decisions can feed a shift register of length $I - \tau - 1$. In the pseudocode, $\text{sgn}(\cdot)$ denotes the sign function, and the function $1/(1+\exp(\cdot))$ is simply obtained via a table lookup.

IV. Performance Evaluation Using EXIT Chart

A. Background

A sizable amount of research has been devoted to the convergence analysis of iterative decoders. This convergence analysis requires, conceptually, a measure that can quantify the quality of soft information as it iterates between the constituent decoders. Measures that are intuitively sound include: the SNR of the LLRs [29], [30], the mutual information between the LLRs and their corresponding transmitted bits [14], and the empirical

bit-error rate (BER) of the LLRs [31]. These measures will then be monitored as the soft information travels back and forth between the constituent decoders. If the chosen measure keeps improving from iteration to iteration, we are converging. Eventually, as the *a priori* information becomes more and more correlated, the iterative decoding ceases to improve.

We use the mutual information measure in this paper, as it is shown to be more numerically stable and less sensitive to the distribution of the LLRs [31]. The mutual information between a discrete-time RV X and a continuous-time RV Y is defined as [32]

$$I(X;Y) \triangleq \sum_{x} p_X(x) \int_{-\infty}^{+\infty} f_{Y|X}(y|x) \log_2 \frac{f_{Y|X}(y|x)}{f_Y(y)} dy \tag{31}$$

where $f_{Y|X}(\cdot|\cdot)$ is the conditional pdf of Y given X. It is shown in [33] that the LLR values are reasonably distributed according to the Gaussian pdf $\mathcal{N}(\sigma^2/2, \sigma^2)$, i.e., a normal pdf with its mean equal to half its variance. Let X be the transmitted sequence of the BPSK symbols, and Y the corresponding LLR sequence with distribution $\mathcal{N}(\sigma^2/2, \sigma^2)$. One can show that (31) simplifies to [14]

$$I(X;Y) = 1 - \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{+\infty} e^{-\frac{\left(y-\frac{\sigma^2}{2}\right)^2}{2\sigma^2}} \log_2(1+e^{-y}) dy. \tag{32}$$

B. Obtaining EXIT Chart

For ease of presentation, we use the following shorthand notations: $I_b^E = I(b; L_b^{\text{ext}})$, $I_b^P = I(b; L_b^{\text{pri}})$, $I_c^E = I(c; L_c^{\text{ext}})$, $I_c^P = I(c; L_c^{\text{pri}})$. The EXIT chart of [14] is basically the diagram containing the transfer characteristics (TC) of the two constituent decoders, i.e., that of the inner ISI detector, I_b^E versus I_b^P, and that of the outer RSC decoder, I_c^P versus I_c^E, with the abscissa and ordinate of the TC of the outer decoder reversed for better visualization of the iterative convergence.

Assuming the input *a priori* LLRs to individual constituent decoders are Gaussian, the following procedure is proposed in [14] to obtain the TC of the ISI channel detector.

1) For a given value of I_b^P, find (numerically) the corresponding σ^2 using (32) and inverse mapping. Note that $I(X;Y)$ of (32) is strictly increasing in its single parameter σ, and so inverse mapping is well defined.
2) Simulate L_b^{pri} according to $\mathcal{N}(\sigma^2/2, \sigma^2)$.
3) Given a fixed E_b/N_0, find the empirical distribution of L_b^{ext} through Monte Carlo simulation of the channel detector.
4) Calculate I_b^E using (31) and numerical integration.

A similar procedure is used to obtain the TC of the RSC decoder where the parameter E_b/N_0 is not present in step 3).

To visualize the iterative convergence, a simulated trajectory is obtained by letting the iterative decoder run freely, as we briefly explain. Initially, $I_b^P = 0$ at the input of the ISI detector. For a given E_b/N_0, the empirical distribution of the extrinsic LLRs at the output of the ISI detector is used to calculate I_b^E using (31). Likewise, the empirical distribution of the LLRs at the output of the RSC decoder is used to calculate I_c^E. If the extrinsic LLRs assume Gaussian pdf (the assumption underlying the TCs of the EXIT chart), the trajectory should closely follow the path suggested by the EXIT chart. We will show that the extrinsic LLRs generated by the HDF-based CD detector lacks this Gaussianity assumption when the energy of the tail ISI is appreciable and SNR is not high enough, i.e., when severe error propagation is present. This is not the case when SDF is used.

To measure the Gaussianity of the extrinsic LLRs in a free run of the compound iterative decoder, we use the normalized empirical kurtosis [34] defined as

$$\text{kurtosis} = \frac{\sum_{i=0}^{M-1} \sum_{n=0}^{N-1} \frac{|y_n(i)|^4}{M}}{\left(\sum_{i=0}^{M-1} \sum_{n=0}^{N-1} \frac{|y_n(i)|^2}{M}\right)^2} \tag{33}$$

where M denotes the number of blocks used in estimation, and $\boldsymbol{y}$ represents a length-N sequence of extrinsic LLRs. For a zero-mean Gaussian pdf, kurtosis is equal to three.

V. Numerical Results

The impulse responses of the unit-energy memory-4 and memory-6 ISI channels that are considered in this paper are denoted by the vectors $\boldsymbol{h}_1 = (1/\sqrt{19}) \cdot [1,2,3,2,1]^T$ and $\boldsymbol{h}_2 = (1/\sqrt{44}) \cdot [1,2,3,4,3,2,1]^T$, and are referred to as ISI-1 and ISI-2. These channels exhibit second-order spectral nulls at normalized frequencies $\omega = \pm 2.094$ (ISI-1) and $\omega = \pm 1.5708, 3.1416$ (ISI-2). Therefore, traditional equalization schemes, such as MMSE LE or DFE, would incur large performance losses compared with the "one-shot" matched-filter bound. For example, at BER of 10^{-3}, these losses are 30 dB (LE) and 9 dB (DFE) for ISI-1 [3] and 36 and 13 dB for ISI-2. Some researchers have considered ISI-1 for turbo equalization [13], [14]. We will show that our approach to turbo equalization fills the gap to no-ISI lower bound[1] much sooner (in the sense that in the BER versus SNR curve, the threshold SNR before we see the waterfall phenomenon is smaller), than other turbo-equalization techniques, e.g., [13], both for ISI-1 and ISI-2, at quite a moderate complexity, assuming the input alphabet size is small.

In obtaining the results of this section, we make the following assumptions.

- The interleaver size is 3E4.
- For all log-domain algorithms (BCJR and CD APP), the $\max^*$ operation is employed.
- The outer code is an RSC code (7,5) where "(7,5)" represents the feedback and feedforward polynomials of the RSC code in octal format.
- The lengths of the causal and anticausal parts of the equalizer in the case of the hybrid LE are 9 and 5, respectively, for ISI-1 as suggested in [13]. To account for the more severe ISI in ISI-2, these lengths are set to 15 and 10, respectively.
- The length of the anticausal forward equalizer used to phase-equalize the channel is 5 for ISI-1 and 7 for ISI-2.

Note that the forward equalizer taps are optimized for each SNR value according to the MMSE criterion [26]. As an example, the effective causal ISI channel seen at

[1]No-ISI lower bound refers to the best-case scenario where no ISI is present, but there exists an outer RSC code.

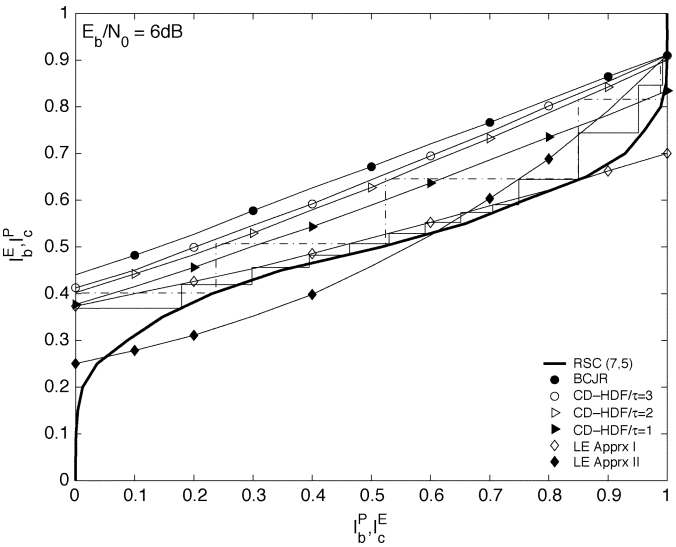

Fig. 3. EXIT charts corresponding to the CD-HDF detector for different values of τ for ISI-1. Solid and dash-dot trajectories correspond to the hybrid LE and CD-HDF/$\tau = 2$, respectively.

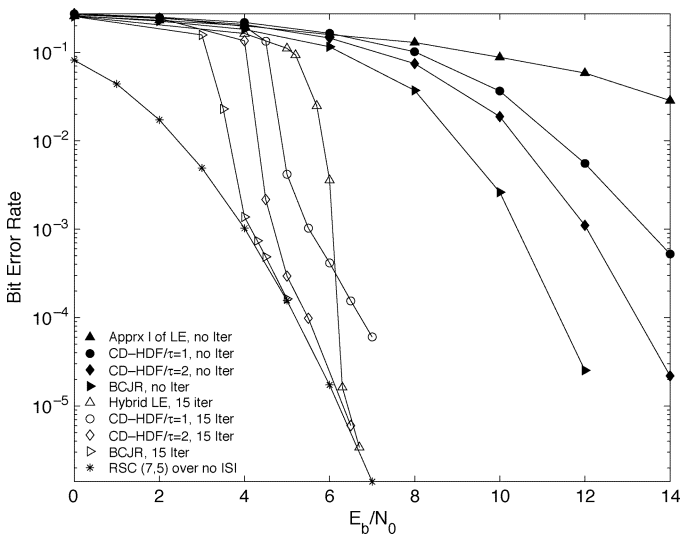

Fig. 4. BER versus E_b/N_0 of different soft-output channel-detection schemes for ISI-1.

the output of the forward equalizer is characterized by $1.0+0.71z^{-1}+0.575z^{-2}+0.267z^{-3}+0.081z^{-4}$ for ISI-1 at 6 dB, and by $1.0+0.886z^{-1}+0.929z^{-2}+0.783z^{-3}+0.461z^{-4}+0.227z^{-5}+0.073z^{-6}$ for ISI-2 at 10 dB. As opposed to the case in ISI-1 and ISI-2, the ISI energy in the effective channels is concentrated up front. The noise at the input of the CD detector is nearly white. We assume that the variance of this noise (to be used in the CD detection) is empirically obtained by means of training.

A. Effect of Decision Delay

The effect of decision delay for ISI-1 is demonstrated in Fig. 3, which shows the TC of the CD-HDF detector with varying values of τ at $E_b/N_0 = 6$ dB. Increasing τ moves the TC up in the EXIT chart. A big improvement is made as we increase τ from 1 to 2. The improvement is marginal if we further increase τ. Also shown in Fig. 3 is the TC of the outer RSC decoder and the simulated *average* trajectory (over 50 transmitted blocks) for $\tau = 2$ (indicated by the dash-dot staircase), visualizing the iterative convergence. The vertical lines of the trajectory represent the information flow from the RSC decoder to the ISI detector, whereas the horizontal ones specify the reverse direction.

An interesting result shown in Fig. 3 is that in a turbo-equalization setting, a CD detector with a decision delay covering most of the ISI energy performs almost as well as full MAP detection implemented by the BCJR algorithm. The proposed CD-APP scheme with $\tau = 2$ operates on a 4-state trellis, whereas the BCJR algorithm for $I = 4$ operates on a 16-state trellis. The computational burden of BCJR, measured in number of multiplications required per symbol period, is twice that of CD-APP, as will be shown shortly. Also, the CD detector enjoys an advantage over the BCJR algorithm in that the latency due to forward and backward recursion of the BCJR algorithm is no longer an issue, and that the storage requirements are vastly reduced.

B. Comparison With Hybrid LE

In Fig. 3, the TCs of the two time-invariant approximations of the LE, as discussed in [13], are also reported. Approximation I of the LE assumes no *a priori* information, whereas approximation II assumes perfect *a priori* information. This is confirmed by the TC of approximation I being higher for small values of *a priori* information, I_b^P, and the TC of approximation II being higher for large values of I_b^P. The hybrid LE scheme of [13] starts with the first approximation, when little *a priori* information is available, and later it switches to the second approximation as the *a priori* information improves.

We can conceptualize the TC of the hybrid scheme as consisting of the initial part of the TC of approximation I and the final part of the TC of approximation II. A switching criterion was developed in [13], which is based on the average variance of the extrinsic output LLRs of the equalizer. The hybrid scheme is argued to have an effective structure, because the two approximations are essentially two finite-impulse response (FIR) filters with the same length, and switching between the two FIR filters amounts to changing filter coefficients. Looking at the trajectory of hybrid LE in Fig. 3 (solid staircase), we see that the switching occurs one iteration after the crossing point between the TCs of the two approximations. Note that both hybrid LE and CD-HDF with $\tau = 2$ converge to the same point, but the trajectory of CD-HDF consists of a much lower number of iterations than that of hybrid LE (5 versus 12). This is an advantage in delay-sensitive applications. Although not shown here due to limited space, unlike the TC of hybrid LE, the TC of CD-HDF at 4 dB still stays above the TC of the RSC decoder, making it possible for the simulated trajectory to manage to "exit" the narrow tunnel between the TC of the CD detector and that of the RSC decoder, and finally converge. The fact that the trajectory associated with CD-HDF follows the EXIT chart closely indicates that for the given delay setting of $\tau = 2$ and ISI-1, HDF is effective, and error propagation is not an issue.

The EXIT chart analysis result of Fig. 3 is also verified in the BER curves of Fig. 4. For a large enough number of iterations, e.g., 15, all soft-output schemes approach the no-ISI lower

TABLE I
PER ITERATION PER SYMBOL COMPLEXITY OF VARIOUS SOFT-OUTPUT DETECTORS WITH BINARY INPUTS. (I: LENGTH OF THE ISI, τ: DECISION DELAY)

Soft output Detector	Multiplications	Additions	$\max^{(*)}$ operations
BCJR	2^{I+2}	$3 \cdot 2^I$	2^I
CD-HDF	$2^{\tau+3}$	$5 \cdot 2^{\tau+1} + 1$	$2 \cdot (2^{\tau+1} - 1)$
CD-MSDF	$2^{\tau+3} + (I - \tau - 1)$	$6 \cdot 2^{\tau+1} + (I - \tau + 2)$	$2 \cdot (2^{\tau+1} - 1)$
CD-ESDF	2^{I+2}	$2^I \cdot (4 + 2^{I-\tau}) + 2^{\tau+1} + 1$	$2^I + 2 \cdot (2^\tau - 1)$
CD-GSDF	$2^{\tau+3} + 3 \cdot (I - \tau - 1)$	$6 \cdot 2^{\tau+1} + 2 \cdot (I - \tau - 1)$	$2 \cdot (2^{\tau+1} - 1)$

bound. Note that in the case of BCJR and CD-HDF/$\tau = 2$, although we show the iterative BER after 15 iterations, no more gain was achieved after 5 iterations. The SNR threshold at which we see the waterfall phenomenon (referred to as pinch-off limit in [14]) is about 1.5 dB better for CD-HDF/$\tau = 2$ than for hybrid LE. This was also anticipated by the EXIT chart analysis result presented earlier. By comparing the no-iteration BER curves with their corresponding 15-iteration curves, we see that all soft-output schemes gain significantly through iteration. Interestingly, the gap between the BCJR and CD detectors decreases dramatically as the number of iterations increases. For example, at 10^{-5}, the gap between the BCJR and CD-HDF/$\tau = 2$ schemes is around 2 dB for no-iteration, but it essentially disappears after five iterations. The gap between the BCJR and hybrid LE algorithms also disappears when the BER falls below 10^{-5}, but with a far higher number of iterations.

Table I summarizes the complexity analysis of the CD detector for different DF schemes. In the same table, we see that the complexity of the BCJR algorithm is exponential in ISI length. We remind the reader that the overall complexity of our approach should include the front-end equalizer if phase equalization of the channel is necessary. However, compared with the hybrid LE, the phase equalizer is much shorter. Given the short-phase equalizer (or no such equalizer at all), the small value of decision delay, and the far lower number of iterations required to converge, it can be shown that the complexity of CD-based schemes is lower or comparable to that of the hybrid LE for binary input systems. The complexity associated with different versions of soft-output LE is provided in [13].

C. Hard versus Soft DF

Fig. 5 compares the EXIT charts corresponding to HDF and GSDF for ISI-2 at 10 dB. First note that the difference between HDF and GSDF shrinks as we increase τ, i.e., as we cover more ISI terms. Eventually, when $\tau = 5$, feedback of past decisions no longer exists, and so GSDF and HDF coincide. When $\tau = 1$, the TC of GSDF starts at a higher I_b^E, but it eventually meets the TC of HDF as the *a priori* information increases. Thus, it appears that given enough *a priori* information, it does not matter whether we use HDF or SDF. However, it turns out in reality that HDF suffers from a serious drawback when ISI is severe, as we now explain.

In Fig. 5, although the EXIT chart of HDF suggests a relatively easy convergence, we see that for $\tau = 1$, the respective trajectory does not follow the EXIT chart. In fact, it moves up for the first couple of iterations, but later, it surprisingly moves down (as indicated by the solid staircase). Although not shown to avoid cluttering Fig. 5, we observed a similar behavior when $\tau = 2$. For larger values of τ, HDF trajectory smoothly moves

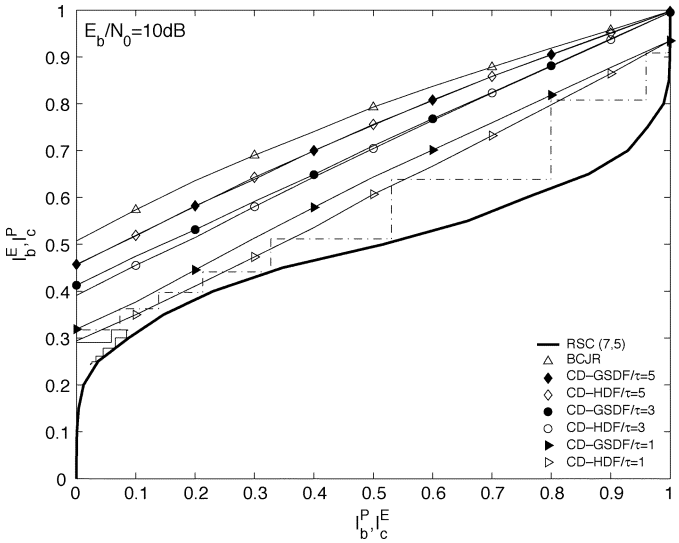

Fig. 5. EXIT charts corresponding to HDF and GSDF for different values of τ for ISI-2. For $\tau = 1$, the trajectory of HDF (solid) fails to follow the path suggested by the EXIT chart. This is not the case for GSDF (dash-dot).

up. The rather peculiar up–down movement of the HDF trajectory in high tail-ISI energy cases can be best explained by the detrimental effect of error propagation induced by HDF. This, in turn, implies that the LLRs generated by CD-HDF fail to satisfy the Gaussianity assumption underlying the TC's in the EXIT chart, as will be shown shortly. The trajectory corresponding to GSDF, on the other hand, follows more or less closely the path suggested by the EXIT chart (as indicated by the dash-dot staircase). The fact that the trajectory corresponding to GSDF, as opposed to HDF, moves up to a much higher point, translates into much better BER performance of GSDF. Similarly, Fig. 6 compares GSDF with the simpler SDF scheme, MSDF, at 8 dB and for $\tau = 2$. Again, we see that MSDF trajectory (solid staircase) fails to completely move up. Although not shown, we also observed that GSDF is generally more robust than MSDF when the suboptimal max operation is used in lieu of the $\max^*$ operation. Fig. 6 also shows the TCs corresponding to the two approximations of hybrid LE. Note that at $E_b/N_0 = 8$ dB, the performance of hybrid LE is not at all satisfactory, because the iterative process stops very early due to the crossing between the TCs of the RSC decoder and approximation I of LE. We observed that the convergence of hybrid LE is not possible until E_b/N_0 reaches 11 dB, and when that happens, we require significantly more iterations to converge, compared with the CD-based schemes at the same SNR.

Using normalized kurtosis, we illustrate in Fig. 7 that the pdf of the extrinsic LLR output of HDF and MSDF fail to eventually become Gaussian. The reverse is true for GSDF for similar

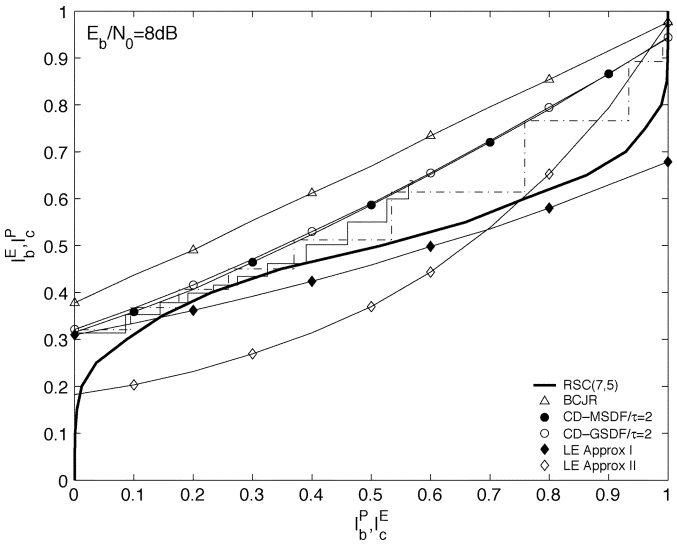

Fig. 6. EXIT charts corresponding to MSDF, GSDF, and hybrid LE for ISI-2. Unlike the trajectory of GSDF (dash-dot), that of the MSDF (solid) fails to follow the path suggested by the EXIT chart. Furthermore, the convergence speed is much faster for GSDF than for hybrid LE.

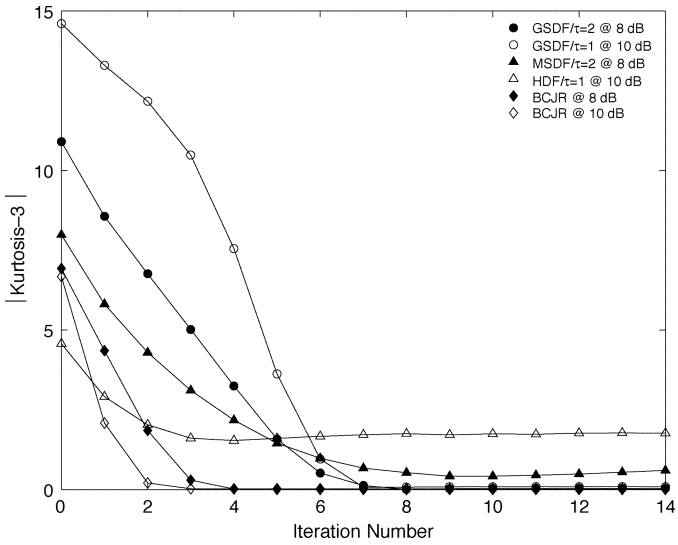

Fig. 7. Gaussianity measure of the extrinsic LLR outputs of different soft-output channel-detection schemes for ISI-2.

parameter settings. As an example, after seven iterations, Gaussianity is achieved in GSDF case. This is even quicker in the BCJR case.

Finally, in Fig. 8, we provide the BER performance of different soft-output channel detection schemes for ISI-2 after 15 iterations. At BER of 10^{-5}, we make a number of observations: 1) CD-GSDF$/\tau = 1$ performs almost 2.5 dB better than hybrid LE. In the former case, the complexity is due to a length-7 forward equalizer and a CD-GSDF detector with $\tau = 1$ as given in Table I, whereas in the latter case, we are using a length-25 LE. We emphasize again that the number of iterations needed for convergence is typically much more for hybrid LE than for CD-GSDF; 2) the performance difference between CD-GSDF and CD-HDF is 2.5 dB for $\tau = 1$ and 1.1 dB for $\tau = 2$; 3)

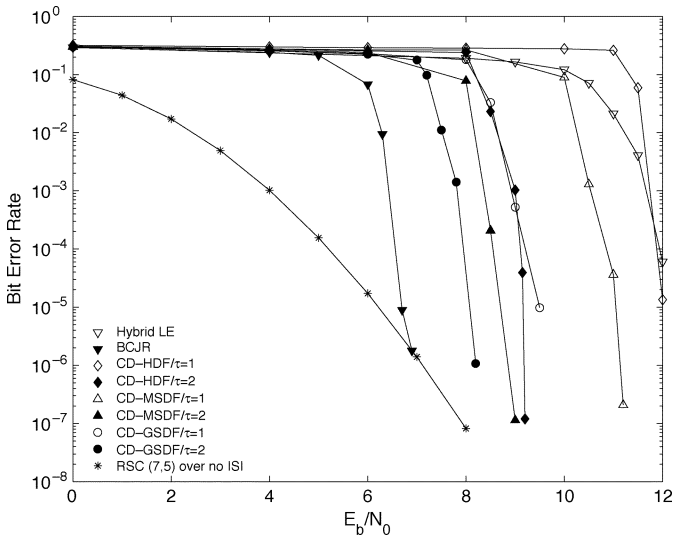

Fig. 8. BER versus E_b/N_0 of different soft-output channel-detection schemes for ISI-2.

CD-GSDF$/\tau = 2$ performs only 1.3 dB worse than the optimal BCJR algorithm, while being far less complex and not suffering from a significant detection delay inherent in the BCJR variants (due to the forward/backward recursion); 4) the performance of CD-MSDF lies between those of GSDF and HDF for both $\tau = 1$ and $\tau = 2$; and 5) as opposed to CD-GSDF$/\tau = 1$, CD-HDF$/\tau = 1$ performs very poorly at 10 dB. This was also corroborated in the EXIT chart of Fig. 5, as the HDF trajectory did not move up because of severe error propagation.

VI. Conclusion

We derived a CD APP algorithm based on a simple trellis representation of partial ISI components to be used in turbo-equalization settings. We also investigated trajectories of mutual information through the iterative process to evaluate various DF schemes that can be used in conjunction with the CD-APP algorithm. We demonstrated that different ISI structures called for different DF schemes. For the two examples of relatively severe ISI channels considered here, we showed that CD-APP combined with appropriate SDF schemes approached the optimal BCJR algorithm in performance, but with much reduced complexity. Compared with the exiting soft-output LE, the CD-APP equalizer yields much faster convergence as well as allowing convergence at SNRs at which the LE simply could not converge. At SNR and ISI conditions where the LE was able to converge, BER performance also indicated favorable performance/complexity tradeoffs for CD-APP/DF compared with the LE algorithm, especially for small input alphabets.

Appendix I
Derivation of (22)

Substituting (17) in (20), we can write

$$\mu_T = \sum_j P_j \left(\sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{M} h_i b_{k-i}^{(j)} \right) \qquad (34)$$

which can be further simplified as

$$\sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{M} h_i \bar{b}_{k-i} = \sum_{i=0}^{\tau+1} h_i b_{k-i} + \sum_{i=\tau+2}^{M} h_i \bar{b}_{k-i} \quad (35)$$

where we have used the fact that $\sum_j P_j = 1$ and where $\bar{b}_k \triangleq \sum_j P_j b_k^{(j)}$, and is simply obtained through

$$\begin{aligned}\bar{b}_{k-i} &= \sum_{j:b_{k-i}^{(j)}=+1} P_j(+1) + \sum_{j:b_{k-i}^{(j)}=-1} P_j(-1) \\ &= P\left(b_{k-i}=+1|r_1^{k-i+\tau}\right) - P\left(b_{k-i}=-1|r_1^{k-i+\tau}\right) \\ &= 2P_{k-i}^{(+)} - 1 \end{aligned} \quad (36)$$

where $P_{k-i}^{(+)} \triangleq P(b_{k-i} = +1|r_1^{k-i+\tau})$. Substituting (36) in (35), we arrive at (22).

Appendix II
Derivation of (23)

We express $(r-\mu_T)^2$ as $(r-\mu_j+\mu_j-\mu_T)^2$ and define $\Delta_j \triangleq \mu_j - \mu_T$. Extending the square term, it is easy to show that

$$\begin{aligned}\sigma_T^2 &= \frac{N_0}{2}\sum_j P_j + \sum_j P_j \Delta_j^2 \\ &= \frac{N_0}{2} + \overline{\Delta_j^2} \end{aligned} \quad (37)$$

where $\overline{\{\cdot\}}$ denotes the mean-value operator. Using (17) and (20) in the definition of Δ_j, we can write

$$\Delta_j = \sum_{i=\tau+2}^{M} h_i \left(b_{k-i}^{(j)} - \bar{b}_{k-i}\right). \quad (38)$$

Now, we use (38) to compute $\overline{\Delta_j^2}$ in (37)

$$\overline{\Delta_j^2} = \sum_l \sum_m h_j h_m \overline{\left(b_{k-l}^{(j)} - \bar{b}_{k-l}\right)\left(b_{k-m}^{(j)} - \bar{b}_{k-m}\right)}. \quad (39)$$

Since b_i's are independent, we can further simplify

$$\overline{\Delta_j^2} = \sum_{m=\tau+2}^{M} h_m^2 \sigma_m^2 \quad (40)$$

where $\sigma_m^2 \triangleq \overline{(b_{k-m}^{(j)} - \bar{b}_{k-m})^2}$. Using (36) and the fact that $P_m^{(+)} + P_m^{(-)} = 1$, we will arrive at

$$\begin{aligned}\sigma_m^2 &= P_m^{(+)}\left(2-2P_m^{(+)}\right)^2 + P_m^{(-)}\left(-2P_m^{(-)}\right)^2 \\ &= 4P_m^{(+)}P_m^{(-)}. \end{aligned} \quad (41)$$

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments.

References

[1] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 5, pp. 363–378, May 1972.

[2] E. A. Lee and D. G. Messerschmitt, *Digital Communication*. Norwell, MA: Kluwer, 1988.

[3] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.

[4] J. W. M. Bergmans, S. A. Rajput, and F. A. M. van der Laar, "On the use of decision feedback for simplifying the Viterbi detector," *Philips J. Res.*, vol. 42, pp. 399–428, 1987.

[5] A. Duel-Hallen and C. Heegard, "Delayed decision-feedback sequence estimation," *IEEE Trans. Commun.*, vol. 37, no. 5, pp. 428–436, May 1989.

[6] M. V. Eyuboglu and S. U. H. Qureshi, "Reduced-state sequence estimation for coded modulation on intersymbol interference channels," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 8, pp. 989–995, Aug. 1989.

[7] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 3, pp. 284–287, Mar. 1974.

[8] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE Globecom*, vol. 3, Nov. 1989, pp. 1680–1686.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near-Shannon-limit error-correcting coding and decoding: Turbo-codes," in *Proc. Int. Conf. Commun.*, Geneva, Switzerland, May 1993, pp. 1740–1745.

[10] P. H. Siegel, D. Divsalar, E. Eleftheriou, and J. Hagenauer, "The turbo principle, from theory to practice I," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 5, pp. 793–799, May 2001.

[11] ——, "The turbo principle, from theory to practice II," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 9, pp. 1657–1661, Sep. 2001.

[12] 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, Sep. 1995.

[13] M. Tuchler, R. Koetter, and A. C. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[14] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," *Int. J. Electron. Commun.*, vol. 54, pp. 389–398, Nov. 2000.

[15] Z. Wu and J. Cioffi, "Low-complexity iterative decoding with decision-aided equalization for magnetic recording channels," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 4, pp. 699–708, Apr. 2001.

[16] X. Wang and V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.

[17] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Algorithm for continuous decoding of turbo codes," in *Proc. IEEE Int. Conf. Commun.*, vol. 32, Feb. 1996, pp. 314–315.

[18] 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.

[19] K. Abend and B. D. Fritchman, "Statistical detection for communication channels with intersymbol interference," *Proc. IEEE*, vol. 58, no. 5, pp. 779–785, May 1970.

[20] J. Moon and L. R. Carley, "Performance comparison of detection methods in magnetic recording," *IEEE Trans. Magn.*, vol. 26, no. 11, pp. 3155–3172, Nov. 1990.

[21] ——, "Efficient sequence detection for intersymbol interference channels with run-length constraints," *IEEE Trans. Commun.*, vol. 42, no. 9, pp. 2654–2660, Sep. 1994.

[22] E. Baccarelli, A. Fasano, and A. Zucchi, "A reduced-state soft-statistics-based MAP/DF equalizer for data transmission over long ISI channels," *IEEE. Trans. Commun.*, vol. 48, no. 9, pp. 1441–1446, Sep. 2000.

[23] W. Choi, K. Cheong, and J. M. Cioffi, "Iterative soft interference cancellation for multiple antenna systems," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 1, Sep. 2000, pp. 304–309.

[24] X. Ma and A. Kavcic, "Path partitions and forward-only trellis algorithms," *IEEE Trans. Inf. Theory*, vol. 49, no. 1, pp. 38–52, Jan. 2003.

[25] Y. Li, B. Vucetic, and Y. Sato, "Optimum soft-output detection for channels with intersymbol interference," *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 704–713, May 1995.

[26] J. Moon and W. Zeng, "Equalization for maximum-likelihood detectors," *IEEE Trans. Magn.*, vol. 31, no. 3, pp. 1083–1088, Mar. 1995.

[27] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and suboptimal MAP decoding algorithms operating in the log domain," in *Proc. Int. Conf. Commun.*, Seattle, WA, Jun. 1995, pp. 1009–1013.
[28] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "A soft-input soft-output maximum *a posteriori* (MAP) module to decode parallel and serial concatenated codes,", TDA Prog. Rep. 42-127, Nov. 1996.
[29] 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.
[30] H. E. Gamal and R. Hammons, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE J. Sel. Areas Commun.*, vol. 47, no. 2, pp. 671–686, Feb. 2001.
[31] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.
[32] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.
[33] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 20, no. 3, pp. 429–445, Mar. 1996.
[34] J. K. Tugnait, "Identification and deconvolution of multichannel linear non-Gaussian process using higher order statistics and inverse filter criteria," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 658–672, Mar. 1997.

Jaekyun Moon (S'89–M'90–SM'97–F'05) received the B.S.E.E. degree from the State University of New York at Stony Brook, and the M.S. and Ph.D. degrees from the Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA, in 1987 and 1990, respectively.

He held a short-term position with IBM Research in 1987. Since 1990, he has been with the faculty of the Department of Electrical and Computer Engineering, University of Minnesota, Twin Cities. His research interests are in the area of channel characterization, signal processing and coding for data storage, and digital communication.

Dr. Moon was selected to receive the IEEE-Engineering Foundation Research Initiation Award in 1991. He received the 1994–1996 McKnight Land-Grant Professorship from the University of Minnesota, and the IBM Faculty Development Awards as well as the IBM Partnership Awards. He is a recipient of the National Storage Industry Consortium (NSIC) Technical Achievement Award. He served as Program Chair for the 1997 IEEE Magnetic Recording Conference. He is also Past Chair of the Signal Processing for Storage Technical Committee of the IEEE Communications Society. He served as a Guest Editor for the 2001 IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS issue on Signal Processing for High Density Recording. In 2001, he cofounded Bermai Inc., a fabless semiconductor startup, and served as founding President and CTO. He is currently an Editor for the IEEE TRANSACTIONS ON MAGNETICS in the area of signal processing and coding.

Farshid Rafiee Rad (S'01) was born in Shiraz, Iran in 1975. He received the B.S. degree from Shiraz University, Shiraz, Iran, in 1996, and the M.Sc. degree from Tehran Polytechnique University, Tehran, Iran, in 1999, both in computer engineering. He is currently working toward the Ph.D. degree in the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis.

Since 2000, he has been a member of the communications and data storage (CDS) lab in the same department. From 2001 to 2002, he was a Design Engineer with Bermai Inc., Minnetonka, MN. His research interests are in the area of communications/DSP algorithm design, with emphasis on low-complexity iterative coding and equalization for high-density magnetic recording.