Bayesian equalization for LDPC channel decoding

Luis Salamanca, *Student Member, IEEE*, Juan José
Murillo-Fuentes, *Senior Member, IEEE* and Fernando
Pérez-Cruz, *Senior Member, IEEE*

Abstract—We describe the channel equalization problem and its prior estimate of the channel estate information (CSI), as a joint Bayesian estimation problem to improve each symbol posterior estimates at the input of the channel decoder. Our solution takes into consideration not only the uncertainty due to the noise in the channel, but also the uncertainty in the CSI estimate. The marginalization to compute each posterior from the full posterior cannot be computed in linear time, because it depends on all the transmitted symbols. Hence, we also put forward an approximate posterior, inspired by the BCJR algorithm, which is optimal from the Kullback-Leibler divergence viewpoint and presents a complexity identical to the BCJR algorithm. We also use a graphical model representation of the full posterior, in which the proposed approximation can be readily understood. The proposed posterior estimates are more accurate than those computed using the ML estimate for the CSI. To illustrate this point, we measure the bit error rate at the output of a Low-Density Parity-Check decoder, which needs the exact posterior for each symbol to detect the incoming word and it is sensitive to a mismatch in those posterior estimates.

Index Terms—channel equalization, Bayesian inference, fading channels, inter-symbol interference, LDPC coding, BCJR algorithm.

I. INTRODUCTION

Channel coding typically assumes the symbols passes through an additive white Gaussian noise (AWGN) channels [1]. For dispersive channels, e.g. in wireless multipath channels, to avoid inter-symbol interference (ISI), an optimal channel equalizer recovers the AWGN-channel memoryless sequence prior to the channel decoder [2], [3]. Hence, the channel state information (CSI) is assumed known and the maximum likelihood sequence detector (MLSD) [4] (i.e. the Viterbi algorithm) provides at the receiver end the most probable transmitted sequence. Alternatively, the BCJR [5] algorithm computes the a posteriori probabilities (APP) for each transmitted symbol providing bitwise optimal decisions. These APP are suitable inputs to reliable error correcting codes, such as low-density parity-checks (LDPC), allowing maximal achievable rates for the communication system [6]–[8].

Maximum likelihood (ML) is the standard approach to estimate the CSI. The ML criterion uses a known training sequence in a preamble or pilots of the transmitted data. These preambles are typically short to improve the efficiency of the transmission in time varying channels. However, short sequences yield inaccurate CSI estimates. Therefore, the BCJR with the CSI estimated by ML, denoted hereafter as ML-BCJR, only delivers an approximation to the APP for the transmitted symbols. These inaccurate APP estimates degrade the performance of the LDPC decoder, which may fail to deliver the correct transmitted codeword or may even fail to converge at all.

Several works focus on the effects of this imperfect estimation of the CSI upon the performance of the communication system [9]–[11]. Hence, different techniques have been proposed to deal with

L. Salamanca and J.J. Murillo-Fuentes are with the Dept. Teoría de la Señal y Comunicaciones, Escuela Superior de Ingenieros, Universidad de Sevilla, Camino de los Descubrimientos s/n, 41092 Sevilla, Spain. E-mail: {salamanca,murillo}@us.es

F. Pérez-Cruz is with the Dept. Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid, Avda. de la Universidad 30, 28911, Leganés (Madrid), Spain. E-mail: fernando@tsc.uc3m.es

the problem of an inaccurate knowledge of the CSI. In [12], per-survivor processing is proposed for maximum likelihood sequence estimation, whenever the uncertainties in the channel estimation restrict the standard Viterbi algorithm. Since the advent of turbo processing, some Bayesian approaches have been proposed to embed and consider the uncertainties in the whole iterative process of equalization and decoding. Otnes and Tuchler propose an approximate solution to include the uncertainties in the computation of the APP [13]. Wang and Chen put forward a blind algorithm, based on an iterative marginalization of the channel posterior through Gibbs sampling, to compute an approximation to the APP [14]. Similarly, Lu and Wang consider the uncertainties of other critical parameters in an OFDM system [15]. Anastasopoulos and Chugg propose an adaptive soft-input soft-output algorithm that computes the APP exchanging soft metrics to combine and marginalize information of the channel estimation [16]. Alternatively, Skoglund *et al.* show that it is possible to face the multipath fading by jointly considering channel estimation and error correction [17], through a practical construction of error-correcting codes and its metrics [18]–[20]. Recently, Chi *et al.* put forward a Bayesian blind detector that, considering only the channel distribution information, jointly faces the tasks of data detection and channel estimation in a MIMO system [21]. Finally, in [22] we introduce a nonlinear nonparametric equalizer to provide accurate symbol-by-symbol estimates with good performance at the output of a LDPC decoder.

In this paper to improve the performance of the standard ML-BCJR equalizer, we propose a Bayesian equalizer (BE), which integrates the uncertainty in the CSI to produce more accurate APP estimates. The bit error rate (BER) at the equalizer output only differ slightly in favor of the BE, because at this point we only consider hard decisions. By further employing an LDPC code [8], [23], we experimentally show that these new estimates of the APP significantly improve the performance of the channel decoder, which needs accurate APP to provide a correct estimation of the transmitted word. These gains are more pronounced for high signal-to-noise ratios (SNRs), channels with long impulsive responses and/or short training sequences. Nevertheless, the proposed approach does not have an analytical description and cannot be computed in linear-time in the number of symbols as the BCJR algorithm, thus we also propose an approximation to the Bayesian solution, hereafter referred to as the approximate Bayesian equalizer (ABE), that presents the same complexity as the ML-BCJR solution, but it is able to retain most of the gain of the full Bayesian approach. Compared to the solution in [22], the BE is parametric, it provides the posterior probability for linear channels and, most important, its computational complexity is identical to the BCJR equalizer. Some preliminary results were presented in [24], [25]. In this paper we rewrite and extend these works to better show the connection with the BCJR and justify the ABE as the optimal Kullback-Leibler divergence approximation to the BE. Also, we analyze the proposed methods from a graphical model point of view and we address the memoryless channel problem. Finally, we extend the approaches for any linear modulation.

The following notation is used throughout the paper. If $\mathbf{u}$ is a vector, $\mathbf{u}/u_i$ denotes the whole vector $\mathbf{u}$ except for its ith entry and $\mathbf{u}_i^j$ is a vector with the entries of $\mathbf{u}$ in the range i to j. For the sake of simplicity, $\mathbf{u}_i^{i+L-1}$ will be referred to as $\mathbf{u}_i$. If $\mathbf{A}$ is a matrix, $\mathbf{A}^\top$ represents the transpose of $\mathbf{A}$ and $\mathbf{A}^{\mathrm{H}}$ its Hermitian.

II. ML-BCJR EQUALIZATION

We consider the discrete-time dispersive communication system depicted in Fig. 1. A block of K message bits, $\mathbf{m} = [m_1, m_2, \ldots, m_K]^\top$, is encoded with a rate $R = K/N$ code into $\mathbf{b} = [b_1, b_2, \ldots, b_N]^\top$. An M-ary modulation is considered to obtain
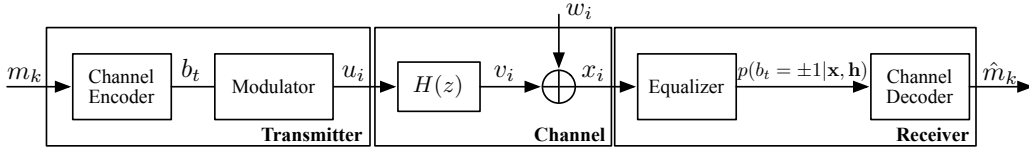
Fig. 1. System model.

$N_S = \lceil N / \log_2 M \rceil$ symbols. The block frame transmitted[1] over the channel is $\mathbf{u} = [u_1, u_2, \ldots, u_{N_S}]^\top$. The channel $H(z)$ is completely specified by the CSI, i.e., $\mathbf{h} = [h_1, h_2, \ldots, h_L]^\top$, where L is the length of the channel impulsive response. In this paper, we model $\mathbf{h}$ as independent zero-mean complex Gaussians (Rayleigh fading). The received signal yields:

$$x_i = \mathbf{u}_i^\top \mathbf{h} + w_i \tag{1}$$

where w_i represents an AWGN channel with variance σ_w^2. Thus, we denote the received sequence by $\mathbf{x} = [x_1, x_2, \ldots, x_{N_S}]^\top$.

The maximum likelihood criterion is the standard tool for channel estimation. Thus, prior to every coded block of data, we transmit a preamble with n known symbols $(u_1^\circ, \ldots, u_n^\circ)$. The receiver uses $\mathcal{D} = \{x_i^\circ, u_i^\circ\}_{i=1}^n$, i.e. the training sequence, to estimate the channel by maximizing:

$$\hat{\mathbf{h}}_{ML} = \arg \max_{\mathbf{h}} p(\mathbf{x}^\circ | \mathbf{u}^\circ, \mathbf{h}). \tag{2}$$

In the ML-BCJR equalizer, this estimation is provided to the BCJR algorithm to obtain the APP for each transmitted symbol,

$$p(u_i | \mathbf{x}, \hat{\mathbf{h}}_{ML}) \qquad i = 1, \ldots, N_S, \tag{3}$$

which are the inputs to the LDPC decoder [26], that provides the maximum a posteriori estimate for m_k. This receiver is limited by the fact that it only takes into consideration the uncertainty in the channel noise and not the uncertainty in the estimation process, i.e. it assumes the ML estimate is the true CSI. If the training sequence is long enough this might be the case, but it does not need to be in most cases of interest, where we need to keep this training sequence as short as possible.

III. BAYESIAN EQUALIZATION

In this paper, we propose to use Bayesian statistics to take into consideration the uncertainties in the CSI estimate and provide more accurate estimates of the APP. The Bayesian equalizer (BE) estimates the APP as follows:

$$p(u_i | \mathbf{x}, \mathcal{D}) = \int p(u_i | \mathbf{x}, \mathbf{h}) p(\mathbf{h} | \mathcal{D}) d\mathbf{h}, \tag{4}$$

where $p(u_i | \mathbf{x}, \mathbf{h})$ is given by the BCJR algorithm for each particular $\mathbf{h}$ and $p(\mathbf{h} | \mathcal{D})$ is the posterior of the CSI:

$$p(\mathbf{h} | \mathcal{D}) = \frac{p(\mathbf{h}) p(\mathbf{x}^\circ, \mathbf{u}^\circ | \mathbf{h})}{p(\mathbf{x}^\circ, \mathbf{u}^\circ)} = \frac{p(\mathbf{h}) \prod_{i=1}^n p(x_i^\circ | \mathbf{u}_i^\circ, \mathbf{h})}{p(x_1^\circ, \ldots, x_n^\circ | u_1^\circ, \ldots, u_n^\circ)}. \tag{5}$$

In our model $p(\mathbf{h})$ is Gaussian, and according to (1) the likelihood is Gaussian too:

$$p(\mathbf{x}^\circ | \mathbf{u}^\circ, \mathbf{h}) \sim \mathcal{N}((\mathbf{U}^\circ)^\top \mathbf{h}, \sigma_w^2 \mathbf{I}), \tag{6}$$

$$p(\mathbf{h}) \sim \mathcal{N}(0, \mathbf{C}_h), \tag{7}$$

---

[1] We transmit an additional sequence of $L - 1$ known symbols to end in a known state of the Trellis, but for the sake of simplicity they are not considered in the notation.

hence, the numerator in (5) is the product of complex valued Gaussians that leads to a Gaussian posterior:

$$p(\mathbf{h} | \mathcal{D}) \sim \mathcal{N}(\overline{\mathbf{h}}_{\mathbf{h} | \mathcal{D}}, \mathbf{C}_{\mathbf{h} | \mathcal{D}}), \tag{8}$$

whose mean and covariance matrix are:

$$\overline{\mathbf{h}}_{\mathbf{h} | \mathcal{D}} = (\mathbf{C}_h^{-1} + \mathbf{U}^\circ (\mathbf{U}^\circ)^{\mathrm{H}} \sigma_w^{-2})^{-1} \mathbf{U}^\circ \mathbf{x}^\circ \sigma_w^{-2}, \tag{9}$$

$$\mathbf{C}_{\mathbf{h} | \mathcal{D}} = (\mathbf{C}_h^{-1} + \mathbf{U}^\circ (\mathbf{U}^\circ)^{\mathrm{H}} \sigma_w^{-2})^{-1}. \tag{10}$$

Equation (4) can only be computed numerically, as we later show, because $p(u_i | \mathbf{x}, \mathbf{h})$ is a discrete random variable. But we can interchange the integral by the sums in the forward-backward BCJR algorithm as follows:

$$p(u_i | \mathbf{x}, \mathcal{D}) = \frac{1}{Z} \sum_{\mathbf{u} / u_i} \int p(\mathbf{x} | \mathbf{u}, \mathbf{h}) p(\mathbf{u}) p(\mathbf{h} | \mathcal{D}) d\mathbf{h}. \tag{11}$$

In the light of (4)-(10), both the CSI posterior and the likelihood of the data are Gaussians and the marginalization over $\mathbf{h}$ in (11) can be analytically computed as:

$$p(u_i | \mathbf{x}, \mathcal{D}) = \frac{1}{Z} \sum_{\mathbf{u} / u_i} p(\mathbf{x} | \mathbf{u}, \mathcal{D}), \tag{12}$$

where this Gaussian likelihood yields:

$$p(\mathbf{x} | \mathbf{u}, \mathcal{D}) \sim \mathcal{N}(\mathbf{U}^\top \overline{\mathbf{h}}_{\mathbf{h} | \mathcal{D}}, \mathbf{U}^{\mathrm{H}} \mathbf{C}_{\mathbf{h} | \mathcal{D}} \mathbf{U} + \sigma_w^2 \mathbf{I}). \tag{13}$$

We can solve the Bayesian equalizer by using either (4) or (12), since they are equivalent. But both approaches have their own limitations: the BE in (4) is not analytically tractable; and the BE in (12) needs to sum over a Gaussian with a full covariance matrix and we cannot run the forward backward recursions, because the Markov property is lost.

We can use Monte Carlo sampling to solve (4). We consider the following steps:

1) Obtain G random samples from the posterior of the CSI in (5).
2) Solve the BCJR algorithm for each sample from $p(\mathbf{h} | \mathcal{D})$.
3) Average the results in Steps 1-2,

$$p(u_i | \mathbf{x}, \mathcal{D}) \approx \frac{1}{G} \sum_{j=1}^G p(u_i | \mathbf{x}, \mathbf{h}_j). \tag{14}$$

This solution is time demanding, because we have to calculate G times the BCJR algorithm.

IV. APPROXIMATE BAYESIAN EQUALIZER

Since the covariance matrix of the likelihood in (13) is non-diagonal, the forward and backward recursions approach cannot be used by the Bayesian equalizer in (12). This poses a major problem when computing the solution, because the APP cannot be computed in linear-time in the number of symbols and the full integration is NP hard. In this section, we approximate the solution in (12). The resulting algorithm, the ABE, presents the same complexity of the ML-BCJR and it is able to incorporate the uncertainties in the CSI estimation.

In Fig. **??** we have plotted the inverse covariance matrix in (13) for 1000 BPSK symbols transmitted trough a channel with $L = 6$, $n = 15$ training symbols and an $E_b/N_0 = 6$ dB (this result corresponds to a particularization of Section VI-B). We plot the inverse covariance, because the zero covariates represent conditional independent components in a Markov random field [**?**]. In this figure, we notice that the main diagonal dominates the inverse cross-covariance terms and those do not decay as the symbols are further away. The inverse cross-covariance terms only decay towards zero as the training sequence increases and they all decay equally fast. We can picture the ML-BCJR algorithm as an approximation to the BE, in which the Gaussian density in (13) is replaced by:

$$\widehat{p}(\mathbf{x}|\mathbf{u}, \mathcal{D}) \sim \mathcal{N}(\mathbf{U}^\top \mathbf{h}_{ML}, \sigma_w^2 \mathbf{I}), \tag{15}$$

and being the covariance matrix diagonal allows using the forward-backward recursion to compute $p(u_i|\mathbf{x}, \mathcal{D})$ in linear time. We propose to use the Expectation Propagation (EP) algorithm [27] to approximate (13) with independent Gaussians, so we are able to use forward-backward recursions to estimate the APP, as the ML-BCJR does. EP minimizes the inclusive Kullback-Leibler divergence between the actual density and the proposed approximation by matching the moments between the joint distribution and its approximation, therefore the approximation to (13) can be readily computed as:

$$p(\mathbf{x}|\mathbf{u}, \mathcal{D}) \approx p'(\mathbf{x}|\mathbf{u}, \mathcal{D}) \sim \mathcal{N}(\mathbf{U}^\top \overline{\mathbf{h}}_{\mathbf{h}|\mathcal{D}}, \Sigma), \tag{16}$$

where

$$\Sigma_{ij} = \begin{cases} (\mathbf{U}^H \mathbf{C}_{\mathbf{h}|\mathcal{D}} \mathbf{U})_{ii} + \sigma_w^2, & \text{if } i = j \\ 0, & \text{if } i \neq j. \end{cases} \tag{17}$$

Thus, (12) yields:

$$p(u_i|\mathbf{x}, \mathcal{D}) \approx \frac{1}{Z} \sum_{\mathbf{u}/u_i} p'(\mathbf{x}|\mathbf{u}, \mathcal{D}), \tag{18}$$

which can be computing through the BCJR algorithm.

The proposed ABE is a better approximation to the BE than the ML-BCJR equalizer in two ways. First, it uses the posterior mean instead of the ML estimate. Second, the variance for each sample has two components: one due to the noise (the only one considered by the ML-BCJR equalizer) and the other due to the CSI estimation error. In Fig. **??** the values of LSM: **CAMBIAR VALORES** $\sigma_w = 0.251$ and the variances in (17) vary between LSM: **CAMBIAR VALORES** 0.414 and 0.308, which explains the best performance of the ABE equalizer with respect to ML-BCJR. The approximation loses the correlation between the symbols, but these correlations are not so significant in the SNR ranges of interest and they disappear as the training sequence increases.

Finally, we can point towards two possible directions of improvement for the ABE. We can either use a low rank approximation for the covariance matrix or a tree-structure for the EP approximation [28]. But in both cases the number of estates in the Trellis grows as the number of estates in the BCJR to the power of the order of the approximation. Furthermore, given the structure of the inverse covariance matrix (See Fig. **??**), the order of the approximation has to be increased significantly to make a difference, because most inverse covariates are close to zero and there are no significant correlation between adjacent symbols. Therefore the complexity of improving the ABE is higher than solving the Monte Carlo numerical approximation proposed in the previous Section.

LSM: **Fuera, no?** Based on these results, we provide some details on the practical implementation of the forward-backward algorithm to solve (18). Instead of computing γ in [5], [7] assuming $\mathbf{h}$ is obtained through some estimation criterion such as ML, we marginalize it over

the CSI posterior:

$$\gamma_i^{\mathcal{D}}(p, q) = p'(x_i|\mathbf{u}_i, \mathcal{D})p(u_i = u) \tag{19}$$

where $p'(x_i|\mathbf{u}_i, \mathcal{D})$ yields:

$$p'(x_i|\mathbf{u}_i, \mathcal{D}) = \int p(x_i|\mathbf{u}_i, \mathbf{h})p(\mathbf{h}|\mathcal{D})d\mathbf{h}. \tag{20}$$

The result of this integral can be analytically calculated since both terms are Gaussians:

$$p'(x_i|\mathbf{u}_i, \mathcal{D}) \sim \mathcal{N}(\mathbf{u}_i^\top \overline{\mathbf{h}}_{\mathbf{h}|\mathcal{D}}, \mathbf{u}_i^H \mathbf{C}_{\mathbf{h}|\mathcal{D}} \mathbf{u}_i + \sigma_w^2). \tag{21}$$

The α and β terms are computed by means of the forward and backward recursions, as in [5], [7]. Finally, the APP is calculated from the transition probabilities in a similar way as in [5], [7].

Although the ABE provides a suboptimal computation of the APP compared to (12), since the diagonal of the covariance matrix in (13) is dominant, discarding the off-diagonal elements of $p(\mathbf{x}|\mathbf{u}, \mathcal{D})$ does not significantly change the APP estimates. LSM: **Hasta aqui.**

There is a graphical model interpretation of the BE that illustrates why the forward-backward recursions cannot be applied over it and why the approximation in ABE allows it. In the computation of the APP for the Bayesian equalizer in (11), the integration over $p(\mathbf{h}|\mathcal{D})$ introduces a dependence of all the received symbols with this posterior, as depicted in Fig. 2. Therefore, Markovity cannot be assumed to obtain the APP through the BCJR algorithm.
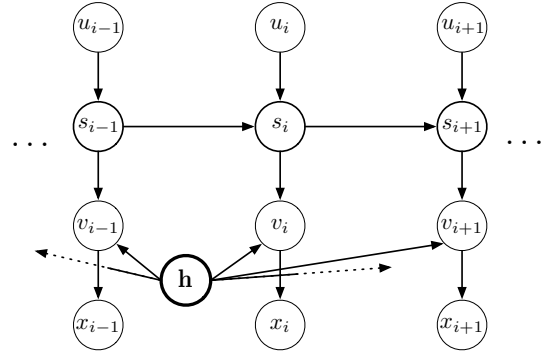

Fig. 2. Bayesian network for the equivalent Trellis assuming the marginalization of $p(\mathbf{h}|\mathcal{D})$ in (11).

To recover the Markov property, the ABE does not consider this statistical dependence and assumes for each local computation of the forward-backward algorithm that $p(\mathbf{h}|\mathcal{D})$ is independent for each received symbol. The corresponding graphical model is included in Fig. 3. It exhibits a hidden Markov model structure equivalent to the BCJR equalizer, since x_i only depends on the state s_i, obtained with the previous state s_{i-1} and the transmitted symbol u_i. In this graphical model, we find an interpretation for (16), in which we assume that the channel changes in each transmission according to the CSI posterior, instead of using the same realization for all the symbols. This model in Fig. 3 is also the one used for the ML-BCJR algorithm, but in this case $\mathbf{h}$ is not a random variable, but a fixed value equal to the ML estimate that misses the uncertainty in the CSI.

V. BE FOR MEMORYLESS CHANNEL

The memoryless channel it is an important special case. First, it can be analytically solved without needing to relay on the ABE. Second, in OFDM the ISI channel is divided into non-interfering memoryless channels and, consequently, the one-tap channel model is of particular interest. If this is the case, we do not have to run the BCJR algorithm
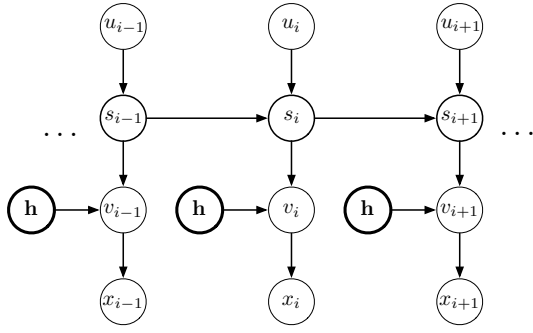
Fig. 3. Bayesian network for the equivalent Trellis assuming statistical independence over $p(\mathbf{h}|\mathcal{D})$.

to compute the APP since x_i only depends on u_i. The distribution of the APP in (4) for an AWGN channel can be obtained by first computing

$$p(u_i|x_i, h) = \frac{p(x_i|u_i, h)p(u_i)}{p(x_i|h)} = \frac{1}{Z}\, p(x_i|u_i, h), \qquad (22)$$

and

$$p(x_i|u_i, h) \sim \mathcal{N}(u_i\, h, \sigma_w^2). \qquad (23)$$

We can analytically integrate (4) with (23), since the posterior of the CSI is Gaussian, ant it follows that

$$p(u_i = u|x_i, \mathcal{D}) = \frac{1}{Z} \int p(x_i|u_i = u, h)p(h|\mathcal{D})dh \qquad (24)$$

$$= \frac{1}{Z} \exp\left(-\frac{|x_i - u_i h_{h|\mathcal{D}}|^2}{2(\sigma_{h|\mathcal{D}}^2 + \sigma_w^2)}\right) \qquad (25)$$

is the APP estimate for the Bayesian equalizer in memoryless channels. In this case, it can be seen that the ABE would provide the same result as the BE an it has two independent error components, one due to the noise and the other due to the estimation error, while the ML-BCJR would only count the noise term and would provide overconfident estimates to the LDPC decoder.

VI. SIMULATION RESULTS

To illustrate the performance of the proposed BE and ABE, we compare them versus the standard ML-BCJR solution. The BE is computed for $G = 100$ independent random samples. In all the experiments presented on this section we consider:

- Block frames of 500 random bits encoded with a regular LDPC code (3,6) of rate $1/2$.
- Between frames, a training sequence of n uncoded symbols is transmitted to estimate the channel.
- Every frame, and its associated training sequence, is sent over the same Rayleigh fading channel. We assume that the channel coherence time is greater than the duration of the frame, i.e., the channel does not change along the word. All the frames are transmitted over the same random channel.
- A zero-mean unit-variance Gaussian prior.

In the following simulations we assume different modulations and lengths of the channel to provide the curves of BER and word error rate (WER) before and after the LDPC decoder. We use the Belief Propagation (BP) as the LDPC decoder [26].

A. Memoryless channel

As proposed in Section V, we can exactly compute the APP estimates for a memoryless channel. In Fig. 4 we compare the BER

curves after the LDPC decoder for the ML-BCJR and the Bayesian equalizers. We depict the curves for QPSK, 16-QAM and 256-QAM modulations, assuming a training sequence of length $n = 4$. We can observe in Fig. 4 that the gain of the Bayesian equalizer compared to the ML-BCJR becomes greater with the modulation order, since the Bayesian approach has more room for improvement. Therefore, without and increasing in the computational complexity, we obtain a gain about 0.25 dB for the 256-QAM case and an $E_b/N_0 = 11$ dB.
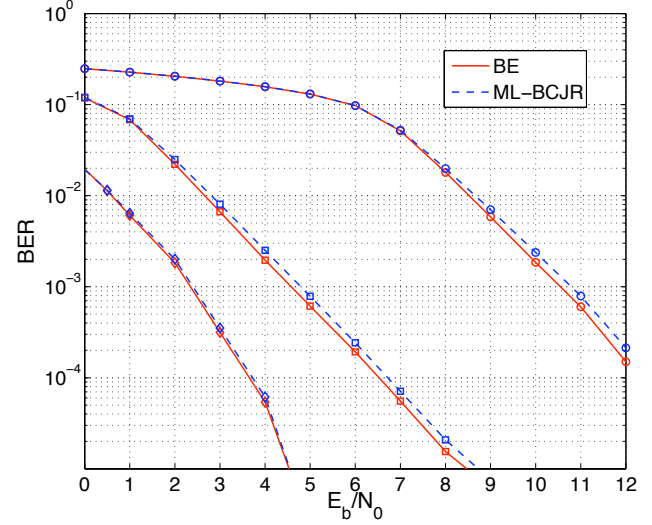

Fig. 4. BER performance for the BE (solid lines) and ML-BCJR (dashed lines) after the LDPC decoder, for a memoryless channel, and different modulation orders: QPSK ($\diamond$), 16-QAM ($\square$) and 256-QAM ($\circ$).

B. Performance after equalization and decoding

In Fig. 5 we depict the BER curves before and after the LDPC decoder for the BE, the ABE and ML-BCJR equalizer. The codeword is BPSK modulated and the symbols are transmitted through a 6 taps channel. The length of the training sequence is $n = 15$ and $n = 40$ symbols.

In Fig. 5 we first observe that the difference between BER curves before the LDPC decoder is negligible, since at this point we only measure how good the APP estimate of the 50% percentile is. Nevertheless, when we measure the BER after the LDPC decoder, we obtain a significant gain, because the LDPC decoder benefits from accurate APP estimates to decode correctly the transmitted codeword, i.e., the BP uses the APP for each individual bit. These results sustain our claim that the Bayesian equalizer provides more accurate predictions of the APP than the ML-BCJR equalizer, as the LDPC decoding is improved with them.

We propose in Section II Monte Carlo sampling to obtain the APP estimates of the Bayesian equalizer in (4). Then, to face the complexity problem, in Section IV we introduce the approximate Bayesian equalizer (ABE) which, although is an approximation to the exact result of (4), has the same complexity of the standard ML-BCJR solution. As illustrated in Fig. 5 and explain in Section IV, there is a loss of performance for the ABE with respect to the optimal BE, but the ABE outperforms the ML-BCJR equalizer at the same computational complexity, while to solve the BE through Monte Carlo we have to compute G times the BCJR algorithm.

C. Results for different modulations and lengths of the channel

We compare the performance in terms of WER for the ABE versus the ML-BCJR equalizer, for different modulations and lengths of the
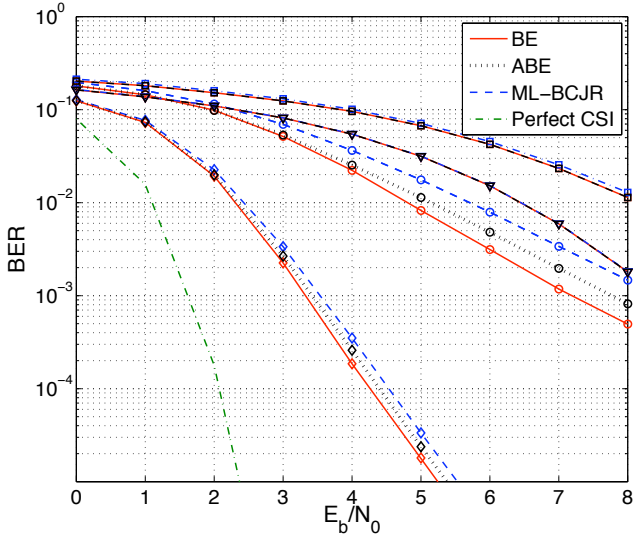
Fig. 5. BER performance for BE (solid lines), the ABE (dotted lines) and ML-BCJR equalizer (dashed lines), for a channel with 6 taps before the decoder with $n = 15$ (□), before the decoder with $n = 40$ (∇), after the decoder with $n = 15$ (○), and after the decoder with $n = 40$ (◇).

channel. In Fig. 6 we compare both methods for different lengths of the training sequence, a QPSK modulation and a channel with $L = 3$. The WER curve with a perfect knowledge of the CSI is also included to set a lower bound of performance for the system.
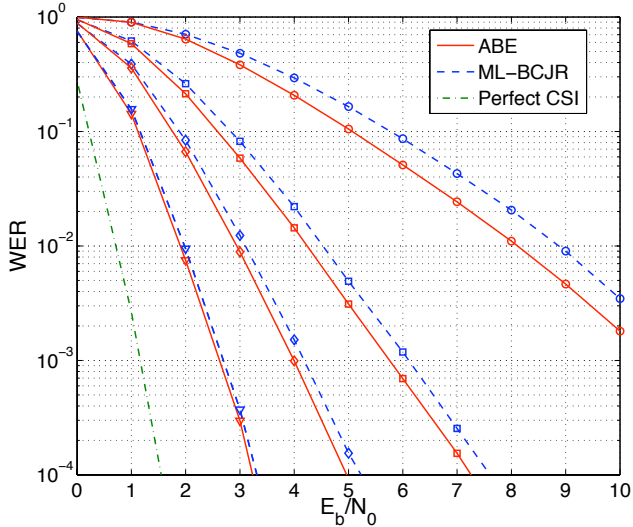

Fig. 6. WER performance for the ABE (solid lines) and ML-BCJR equalizer (dashed lines) after the LDPC decoder, for a channel with 3 taps, QPSK modulation and different lengths of the training sequence, $n = 5$ (○), $n = 10$ (□), $n = 15$ (◇) and $n = 25$ ($\triangledown$). In dash-dotted line, the BCJR with perfect CSI.

At higher values of SNR and worse estimations of the channel, due to shorter training sequences, the difference between E_b/N_0 values of both methods for a certain word error rate increases. This yields a gain close to 0.3 dB at an $E_b/N_0 = 7$ dB for $n = 10$ training samples. Although the performance of the ML-BCJR equalizer tends to the Bayesian equalizer for higher lengths of the training sequence, this involves a loss of efficiency (number of information bits over all transmitted bits) since it requires longer training sequences to obtain an accurate estimation.

A higher modulation order translates into more states in the Trellis

considered for the forward and backward recursions. Therefore, in case of uncertainties in the estimated CSI, the number of inaccurate operations grows and we can expect a higher degradation of the equalizer performance, which finally yields into more inaccurate APP estimations. Thus, if we increase the order of the modulation, we can expect a greater gain for the proposed Bayesian equalizer. To illustrate this point, in Fig. 7 we include the WER curves for the ABE and the ML-BCJR equalizer after the LDPC decoder, assuming a 16-QAM modulation, $L = 3$ and different lengths of the training sequence. We can observe in Fig. 7 a gain over 0.5 dB for an $E_b/N_0 = 9$ dB and $n = 10$. In all the curves the gain of the Bayesian equalizer increases compared to the previous results for a QPSK modulation.
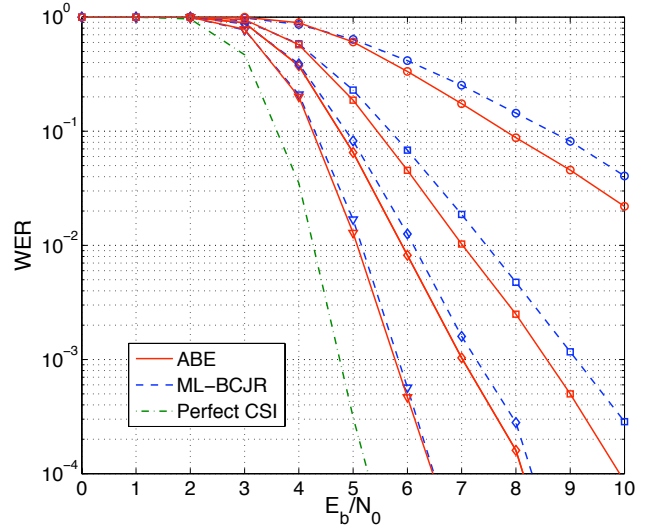

Fig. 7. WER performance for the ABE (solid lines) and ML-BCJR equalizer (dashed lines) after the LDPC decoder, for a channel with 3 taps, 16-QAM modulation and different lengths of the training sequence, $n = 5$ (○), $n = 10$ (□), $n = 15$ (◇) and $n = 25$ ($\triangledown$). In dash-dotted line, the BCJR with perfect CSI.

The longer the impulsive response of the channel is the more uncertainty in the CSI estimate and the higher the room for improvement for the Bayesian equalizer. To illustrate this, we depict in Fig. 8 the WER curves for the ABE and the ML-BCJR equalizer after the LDPC decoder, using a QPSK modulation and $L = 5$. In Fig. 8 we can observe a gain close to 1 dB for an $E_b/N_0 = 9$ dB and $n = 10$, which shows an increase in its performance compared to the 3 taps scenario.

VII. CONCLUSIONS

Channel equalization has been traditionally solved using a frequentist approach, i.e. maximum likelihood. The Bayesian model introduced in this paper, in which the posterior probability of the estimated CSI is taken into consideration, is a more principled solution, because it takes into account not only the uncertainty due to the noise, but also the uncertainty about the CSI estimation. The maximum likelihood solution and the Bayesian equalizer perform similarly when we predict the transmitted symbol. However, the Bayesian equalizer presents more accurate APP estimates than the standard maximum likelihood solution. We measure the quality of the APP estimates using an LDPC decoder, the standard channel codes in today's communications systems, since the LDPC decoder needs the exact APP to perform optimally. We also propose an approximate Bayesian equalizer that can keep most of the gain of the Bayesian equalizer at the same computational cost as the ML-BCJR equalizer. This gain is remarkable in scenarios with short training sequences,
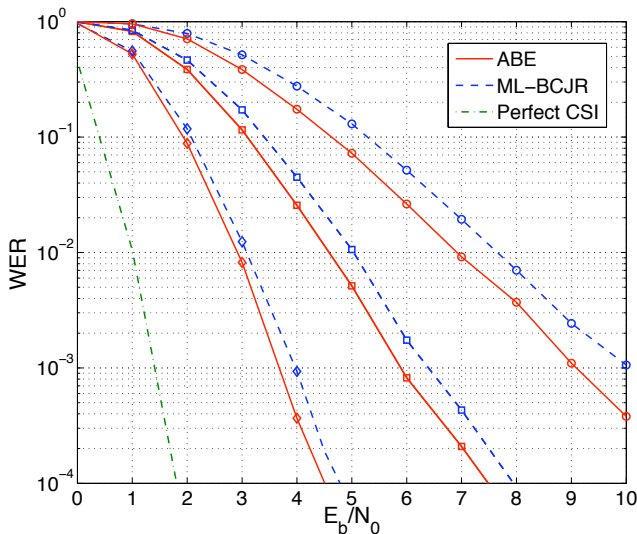
Fig. 8. WER performance for the ABE (solid lines) and ML-BCJR equalizer (dashed lines) after the LDPC decoder, for a channel with 5 taps, QPSK modulation and different lengths of the training sequence, $n = 10$ ($\circ$), $n = 15$ ($\square$) and $n = 25$ ($\diamond$). In dash-dotted line, the BCJR with perfect CSI.

long channels and multilevel constellations. We have illustrated these results for LDPC codes and they can be carried over to other coding schemes that need accurate APP, such as turbo codes.

REFERENCES

[1] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, USA: Wilson and Sons, 1991.
[2] J. G. Proakis, *Digital Communications*, 5th ed. New York, NY: McGraw-Hill, 2008.
[3] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital Communication Receivers: Synchronization, Channel Estimation and Signal Processing*. New York: Wiley, 1997.
[4] D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 2, pp. 268–278, Mar. 1973.
[5] 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. 20, no. 2, pp. 284–287, Mar. 1974.
[6] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New Jersey, USA: John Wiley & Sons, 2006.
[7] T. K. Moon, *Error Correction Coding: mathematical methods and algorithms*, ser. Information Theory. John Wiley & Sons, 2005.
[8] S. Chung, D. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.
[9] N. Merhav, G. Kaplan, A. Lapidoth, and S. Shamai Shitz, "On information rates for mismatched decoders," *IEEE Trans. Inf. Theory*, vol. 40, no. 6, pp. 1953 –1967, Nov. 1994.
[10] A. Lapidoth and P. Narayan, "Reliable communication under channel uncertainty," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2148 –2177, Oct. 1998.
[11] A. Lapidoth and S. Shamai, "Fading channels: how perfect need "perfect side information" be?" *IEEE Trans. Inf. Theory*, vol. 48, no. 5, pp. 1118 –1134, May. 2002.
[12] R. Raheli, A. Polydoros, and C.-K. Tzou, "Per-survivor processing: a general approach to MLSE in uncertain environments," *IEEE Trans. Commun.*, vol. 43, no. 234, pp. 354 –364, Feb. 1995.
[13] R. Otnes and M. Tuchler, "On iterative equalization, estimation, and decoding," in *Proc. IEEE Int. Conf. on Commun.*, vol. 4, May 2003, pp. 2958–2962.
[14] X. Wang and R. Chen, "Blind Turbo equalization in Gaussian and impulsive noise," *IEEE Trans. Veh. Technology*, vol. 50, no. 4, pp. 1092–1105, Jul. 2001.
[15] B. Lu and X. Wang, "Bayesian blind Turbo receiver for coded OFDM systems with frequency offset and frequency-selective fading," in *Proc. IEEE Int. Conf. on Commun.*, vol. 1, Apr.-May 2002, pp. 44–48.
[16] A. Anastasopoulos and K. Chugg, "Adaptive soft-input soft-output algorithms for iterative detection with parametric uncertainty," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1638–1649, Oct. 2000.
[17] M. Skoglund, J. Giese, and S. Parkvall, "Code design for combined channel estimation and error protection," *IEEE Trans. Inf. Theory*, vol. 48, no. 5, pp. 1162 –1171, May. 2002.
[18] O. Coskun and K. Chugg, "Combined coding and training for unknown ISI channels," *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1310 – 1322, Aug. 2005.
[19] C.-L. Wu, P.-N. Chen, Y. Han, and M.-H. Kuo, "Maximum-likelihood priority-first search decodable codes for combined channel estimation and error correction," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4191 –4203, Sep. 2009.
[20] P. Piantanida, S. Sadough, and P. Duhamel, "On the outage capacity of a practical decoder accounting for channel estimation inaccuracies," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1341–1350, May 2009.
[21] Y. Chi, Y. Wu, and R. Calderbank, "Regularized blind detection for MIMO communications," in *Proc. IEEE Int. Symp. on Inf. Theory (ISIT)*, Jun. 2010, pp. 2108 –2112.
[22] P. M. Olmos, J. Murillo-Fuentes, and F. Perez-Cruz, "Joint nonlinear channel equalization and soft LDPC decoding with Gaussian processes," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1183–1192, Mar 2010.
[23] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, March 2008.
[24] L. Salamanca, J. Murillo-Fuentes, and F. Pérez-Cruz, "Channel decoding with a Bayesian equalizer," in *Proc. IEEE Int. Symp. on Inf. Theory (ISIT)*, Jun. 2010, pp. 1998 –2002.
[25] ——, "Bayesian BCJR for channel equalization and decoding," in *Proc. IEEE Int. Workshop on Mach. Learning for Signal Process.*, Aug.-Sept 2010, pp. 53–58.
[26] F. R. Kschischang, B. I. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
[27] T. Minka, "Expectation propagation for approximate Bayesian inference," in *UAI*, 2001, pp. 362–369.
[28] T. Minka and Y. Qi, "Tree-structured approximations by expectation propagation," in *Proc. of the Neural Inf. Process. Systems Conference, (NIPS)*, 2003.