Expectation Propagation as Turbo Equalizer in ISI Channels

Irene Santos, Juan José Murillo-Fuentes, Rafael Boloix-Tortosa, Eva Arias-de-Reyna, and Pablo M. Olmos

***Abstract*—In probabilistic equalization of channels with inter-symbol interference, the BCJR algorithm and its approximations become intractable for high order modulations, even for moderate channel dispersions. In this paper we introduce a novel soft equalizer to approximate the symbol a posteriori probabilities (APP) where the expectation propagation (EP) algorithm is used to provide an accurate estimation. This new soft equalizer is presented as a block solution, denoted as block-EP (BEP), where the structure of the matrices involved is exploited to reduce the complexity order to $\mathcal{O}(LN^2)$, i.e., linear in the length of the channel, L, and quadratic in the frame length, N. The solution is presented in complex-valued formulation within a turbo equalization scheme. This algorithm can be cast as a linear minimum-mean-squared-error (LMMSE) turbo equalization with double feedback architecture where constellations being discrete is a restriction exploited by the EP that provides a first refinement of the APP. In the experiments included, the BEP exhibits a robust performance, regardless of the channel response, with gains in the range 1.5-5 dB compared to the LMMSE equalization.**

***Index Terms*—Expectation propagation (EP), BCJR, complex-valued, turbo equalization, ISI.**

I. Introduction

SOFT or probabilistic channel equalization [1] is a technique to mitigate the interference between symbols (ISI) provoked by the dispersive nature of the channel [2], [3]. It provides the posterior probabilities of the estimated transmitted symbols given the observation, from which nowadays decoders highly benefit [4]. These two tasks, equalization and decoding, were initially considered separately, but the performance was remarkably improved by joining them into a turbo equalization scheme [5]–[7]. In turbo equalization, an equalizer and a decoder exchange information in terms of log-likelihood ratios (LLRs). After one or more iterations, the channel decoder generates the LLRs which are delivered back to the equalizer as updated a priori information.

The optimal BCJR algorithm [8] computes the a posteriori probability (APP) for each transmitted symbol providing maximum a posteriori (MAP) probabilistic decisions. It works on a trellis representation, assuming perfect knowledge of the channel impulse response (CIR) and a channel with finite memory [9]. The BCJR complexity is proportional to the number of trellis branches, M^L, increasing with the number of taps of the channel, L, and the size of the constellation used, M. The BCJR memory requirements per step also grow with the number of states. Therefore, for a few taps and a multilevel constellation the complexity becomes intractable.

To reduce the complexity of the BCJR some suboptimal algorithms, based on performing a simplified trellis search with only M_e states, have been proposed in the literature. They can be divided into two different families. The first one consists in reducing the effective length of the CIR, such as the reduced-state BCJR (RS-BCJR) algorithm [10], which is based on the reduced-state sequence detection (RSSD) [11]–[13]. The key idea is to cancel the final channel taps, by truncating the memory of the channel, to reduce the number of states. On the other hand, other algorithms only keep the states with highest APP, i.e., unlike the previous algorithms, they perform a reduced search on the original full trellis, instead of a full search on a reduced-state trellis. This is the case of the M-BCJR algorithm [14]. Some approaches try to join both families to improve the results. We mention the M*-BCJR in [15], that outperforms both RS-BCJR and M-BCJR algorithms. These approaches have some important limitations. A first issue is that they are usually designed for some types of channels [10], [14], [15]. Secondly, they are unable to merge the paths determined by forward and backward trellises, since these paths generally do not match in both procedures. To overcome these problems, a variation of the M-BCJR algorithm is proposed in [16], where the authors use a different active state selection criterion. In [17] the channels are equalized differently according to their minimum, maximum or mixed phase nature [18]. Depending on the channel realization, they resort to the forward recursion or forward-trellis, backward-trellis or an optimized mixture of them. This approach is referred to as nonzero (NZ) completion. Thirdly, all approximated methods above ignore some paths in the trellis, which means that the explored paths tend to be considered much more reliable than they really are. To reduce this overestimation, the NZ with output saturation (NZ-OS) is proposed in [17]. The above issues can be mitigated with channel shortening approaches, such as [19], where the authors perform a full search over an optimized and reduced trellis. However, it is important to remark that the performance of this approach and approximated BCJR solutions degrades if the number of survivor paths, M_e, does not grow accordingly with the total number of states. Hence they are computationally

I. Santos, J.J. Murillo-Fuentes, R. Boloix-Tortosa and E. Arias-de-Reyna are with the Dept. Teoría de la Señal y Comunicaciones, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Camino de los Descubrimiento s/n, 41092 Sevilla, Spain. E-mail: {irenesantos,murillo,rboloix,earias}@us.es.

P. M. Olmos 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: olmos@tsc.uc3m.es.

This work was partially funded by Spanish government (Ministerio de Economía y Competitividad TEC2016-78434-C3-{2-3}-R and Juan de la Cierva Grant No. IJCI-2014-19150), by the European Union (FEDER and Marie Curie Initial Training Network "Machine Learning for Personalized Medicine" MLPM2012, Grant No. 316861), and by Comunidad de Madrid in Spain (project 'CASI-CAM-CM', id. S2013/ICE-2845).

unfeasible for large trellises and equalizers of the MMSE type are preferred [20].

Message passing approaches have been also investigated, see [21]–[25] and references therein. In [21], an equalizer based on the belief propagation (BP) algorithm is developed to reduce the inference complexity in sparse channels. However, the complexity of the method still grows exponentially with the size of the modulation and the number of nonzero channel interferers. In [24] the graphical model of the system is rewritten to end with a graph with loops equal or larger than 6, and the loopy BP is then applied. Its output is an approximation reported to provide good results in ISI channel equalization for BPSK modulations, although the method can be applied to other modulations. Its complexity scales linearly with the frame length and the channel memory, but quadratically with the constellation order. A different approach is proposed in [23], where a successive interference canceler applied to equalization is developed by considering the interference plus noise as Gaussian distributed. In [22] the authors develop an approach that introduces expectation propagation (EP) approximate inference [26]–[28] to incorporate into the BP algorithm the information from the BPSK symbol estimates coming from the channel decoder. In [25] the authors develop a different EP implementation for the equalizer in [22]. These two works develop particular instances of EP to project the BP messages into the right distribution (Gaussian/discrete), allowing feasible updates of the BP messages. The main difference between the EP method proposed by [22] and [25] is the procedure to tackle numerical instabilities during the EP updates related to variance parameters taking extremely small values or even negative ones. While in [22] the authors introduce a damping approach, in [25] negative variance parameters are replaced by their absolute values. At this point it is important to remark that [22], [23], [25] consider BPSK transmissions and the EP is just used to better approximate messages from the decoder to the equalizer.

Soft linear equalization, such as the linear minimum-mean-squared-error (LMMSE) [29], is a suboptimal but low-cost alternative. Its complexity is dominated by the inversion of a matrix of size N when all observations are processed in a block or batch approach. When no turbo scheme is used the complexity can be reduced to $\mathcal{O}(N \log N)$ by exploiting the circulant nature of the involved matrix and the fast Fourier transform (FFT). However, after the feedback in the turbo equalization the matrix is no longer circulant and we can not use the FFT, yielding a cubic complexity in N [30]. In [31], some approximations are introduced to lower the complexity. Some windowed versions have also been developed in order to reduce this complexity. Specifically, in [32] a sliding window LMMSE algorithm is proposed and in [33] these results are improved by replacing the sliding window with an extending window. In [5], [32], [34] the authors also propose some approximated windowed solutions to further reduce the complexity.

In this work we focus on a novel block solution to improve the equalization algorithms above when dealing with multidimensional constellations, with linear complexity in the constellation size and for any channel realization. We present an EP-based algorithm that approximates the joint posterior probability of the transmitted bits in a centralized manner, i.e., we do not use EP to project the BP messages into a different distribution, as [22], [25] would do. EP has been already successfully applied to multiple-input multiple-output (MIMO) detection [35]–[37], low-density parity-check (LDPC) channel decoding [38], [39], tracking of flat-fading channels [40] and equalization of BPSK transmissions with message passing BP approaches [22], [25]. We exploit the key idea in [36], where compared to [35] the EP is used to better approximate the full posterior rather than to improve message passing algorithms at some points. Preliminary results for equalization of real-valued systems were discussed in [41]. In this paper we propose a turbo equalization scheme where we implement EP to obtain a complex-valued Gaussian approximation to the probability of the transmitted symbols conditioned to the received signal. We denote this approach as turbo BEP equalizer (T-BEP). We discuss the interpretation of this T-BEP as a double turbo LMMSE equalizer, where the discrete nature of the transmitted symbols is used as a first feedback. To deal with negative variance we avoid updates whenever EP provides negative variance. The computational complexity per step of the T-BEP is dominated by the inversion of an N-dimensional banded covariance matrix, which has exactly the same structure as in the turbo LMMSE scheme [30]. We exploit the structure of the matrices to reduce it to $\mathcal{O}(LN^2)$, i.e., quadratic complexity in N and independent of M. In addition, we address the estimation of the mutual information between the detected and transmitted symbols to explain the obtained gain. Finally, we compare it to the most relevant approximated BCJR approaches discussed above. As a result, at low dimensional scenarios we achieve a performance close to the optimal BCJR solution, regardless of the channel realization. For multilevel constellations the BEP outperforms the turbo LMMSE equalizer (T-LMMSE). This performance is further improved with the T-BEP. Gains in the 2-5 dB range are reported for 16 and 64-QAM constellations.

The paper is organized as follows. We first describe in Section II the structure and model of the communication system at hand and review the formulation for the block LMMSE turbo equalizer. In Section III the EP algorithm is introduced. Section IV is devoted to describe the novel proposed T-BEP equalizer and develop its formulation for complex numbers. We also study the convergence of the algorithm to propose values for the parameters of the EP equalizer and exploit the structure of the matrices involved to propose efficient computations. In Section V, we include several experiments to show the good performance of the T-BEP. We end with some conclusions.

The following specific notation is used throughout the paper. If $\mathbf{u}$ is a vector, $\mathbf{u}_{i:j}$ refers to a column vector with the entries of vector $\mathbf{u}$ indexed by the set $\{i, i-1, i-2, ..., j\}$. We use $\mathbf{u}^*$ to denote the complex conjugate of $\mathbf{u}$. To denote a normal distribution of a random proper complex vector $\mathbf{u}$ with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ we use $\mathcal{CN}(\mathbf{u} : \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

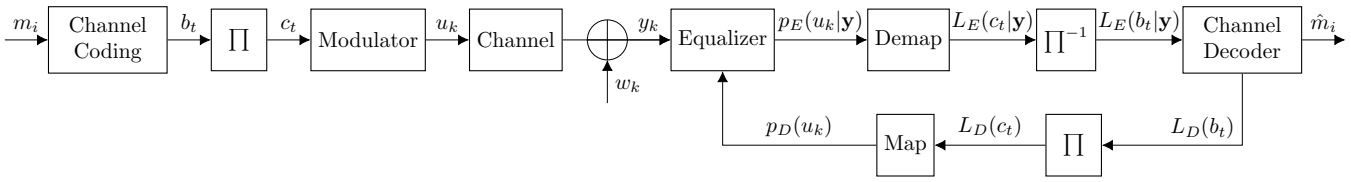


Fig. 1: System model and turbo equalization.

II. System model and LMMSE solution

A. System model

In Fig. 1 we represent the discrete-time communication system with turbo equalization. It can be divided into four parts: transmitter, channel model, equalizer and turbo equalization.

1) Transmitter: A block of K message bits, $\mathbf{m} = [m_1, ..., m_K]^\top$, is encoded with a rate $R = K/V$ into the codeword $\mathbf{b} = [b_1, ..., b_V]^\top$ and permuted with an interleaver to $\mathbf{c} = [c_1, ..., c_V]^\top$. An M-ary modulation is considered to obtain $N = \lceil V/\log_2 M \rceil$ symbols, in $\mathbf{u}$. Then, the block frames $\mathbf{u} = [u_1, ..., u_N]^\top = \mathcal{R}(\mathbf{u}) + \mathrm{j}\mathcal{I}(\mathbf{u})$ are transmitted over the channel, where each component $u_k = \mathcal{R}(u_k) + \mathrm{j}\mathcal{I}(u_k) \in \mathcal{A}$. Hereafter, $\mathcal{A}$ denotes the set of symbols of the constellation of order $|\mathcal{A}| = M$. The mean transmitted symbol energy and energy per bit are denoted by E_s and E_b, respectively.

2) Channel model: The channel is completely specified by the CIR, i.e., $\mathbf{h} = [h_1, ..., h_L]^\top$, where L is the length of the CIR, and the noise variance σ_w^2, that we assume known at the receiver. The received signal $\mathbf{y} = [y_1, ..., y_{N+L-1}]^\top \in \mathbb{C}^{N+L-1}$ is given by

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{N+L-1} \end{bmatrix} = \begin{bmatrix} h_1 & & \mathbf{0} \\ \vdots & \ddots & \\ h_L & \ddots & h_1 \\ & \ddots & \vdots \\ \mathbf{0} & & h_L \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix} + \begin{bmatrix} w_1 \\ \vdots \\ w_{N+L-1} \end{bmatrix} \tag{1}$$

or

$$\mathbf{y} = \mathbf{H}\mathbf{u} + \mathbf{w} \tag{2}$$

in matrix form, where

$$y_k = \sum_{l=1}^{L} h_l u_{k-l+1} + w_k = \mathbf{h}^\top \mathbf{u}_{k:k-L+1} + w_k, \tag{3}$$

$u_k = 0$ for $k < 1$ and $k > N$, and $\mathbf{w} \sim \mathcal{CN}(\mathbf{w} : \mathbf{0}, \sigma_w^2 \mathbf{I})$. In this case, it is circular complex AWGN due to its zero mean [42].

3) Equalization: Given the model above, the posterior probability of the transmitted symbol vector $\mathbf{u}$ yields

$$p(\mathbf{u}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{u})p(\mathbf{u})}{p(\mathbf{y})} \propto \mathcal{CN}(\mathbf{y} : \mathbf{H}\mathbf{u}, \sigma_w^2 \mathbf{I}) \prod_{k=1}^{N} \mathbb{I}_{u_k \in \mathcal{A}}, \tag{4}$$

where $\mathbb{I}_{u_k \in \mathcal{A}}$ is the indicator function that takes value one if $u_k \in \mathcal{A}$ and zero otherwise. The optimal equalizer to estimate $p(u_k|\mathbf{y})$, $k = 1, \ldots, N$, is the BCJR algorithm, unaffordable for multilevel constellations when the channel memory, L, grows.

4) Turbo Equalization: The key point of turbo equalization is the iterative exchange of information between the equalizer and the decoder for the same set of received symbols [5], [7], [34]. Given the extrinsic log-likelihood ratios (LLR) from the equalizer to the decoder, $L_E(b_t|\mathbf{y})$, the latter computes (after one or more iterations) an estimation of the information bits, $\hat{\mathbf{m}}$, and an extrinsic LLR on the coded bits

$$L_D(b_t) = \log \frac{p(b_t = 0|L_E(\mathbf{b}|\mathbf{y}))}{p(b_t = 1|L_E(\mathbf{b}|\mathbf{y}))} - L_E(b_t|\mathbf{y}). \tag{5}$$

These LLRs are mapped again and delivered back to the equalizer as updated a priori probability, with a slight abuse of notation we denote it by $p_D(\mathbf{u})$. This process is repeated iteratively for a given maximum number of iterations, T, or until convergence.

B. Turbo LMMSE Equalization Algorithm

We include next the description of the turbo equalization with LMMSE, that we denote by T-LMMSE, since we will use it as benchmark and its structure is a good starting point to develop our solution. Given the CIR, the posterior approximation provided by the LMMSE equalizer is obtained by replacing the discrete uniform prior $p(\mathbf{u})$ in (4) by a product of independent Gaussian distributions with mean $\mathbb{E}[u_k] \in \mathbb{C}$ and variance $\mathbb{V}[u_k] \in \mathbb{R}_+$,

$$q_{MMSE}(\mathbf{u}) = \mathcal{CN}(\mathbf{y} : \mathbf{H}\mathbf{u}, \sigma_w^2 \mathbf{I}) \prod_{k=1}^{N} \mathcal{CN}(u_k : \mathbb{E}[u_k], \mathbb{V}[u_k]). \tag{6}$$

This distribution is a proper complex Gaussian [42]

$$q_{MMSE}(\mathbf{u}) = \mathcal{CN}(\mathbf{u} : \boldsymbol{\mu}_{MMSE}, \boldsymbol{\Sigma}_{MMSE}) \tag{7}$$

where

$$\boldsymbol{\Sigma}_{MMSE} = \left(\sigma_w^{-2} \mathbf{H}^{\mathrm{H}} \mathbf{H} + \boldsymbol{\Sigma}_{\mathbf{u}}^{-1}\right)^{-1}, \tag{8}$$

$$\boldsymbol{\mu}_{MMSE} = \boldsymbol{\Sigma}_{MMSE} \left(\sigma_w^{-2} \mathbf{H}^{\mathrm{H}} \mathbf{y} + \boldsymbol{\Sigma}_{\mathbf{u}}^{-1} \boldsymbol{\mu}_{\mathbf{u}}\right) \tag{9}$$

and $\boldsymbol{\mu}_{\mathbf{u}} = [\mathbb{E}[u_1], ..., \mathbb{E}[u_N]]^\top$, $\boldsymbol{\Sigma}_{\mathbf{u}} = \mathrm{diag}\left(\mathbb{V}[u_1], ..., \mathbb{V}[u_N]\right)$. The complexity of this solution is dominated by the matrix inversion in (8). For the first iteration, we set $\mathbb{E}[u_k] = 0$ and $\mathbb{V}[u_k] = E_s$, and the matrix yields a circulant matrix whose inverse can be computed with complexity of order $\mathcal{O}(N \log N)$ by means of the FFT. Then, the symbol probability of each entry is computed by independently deciding on each component.

In turbo equalization, the extrinsic information is passed to the channel decoder. The extrinsic information for symbol k is computed from (7), assuming equally probable symbols

for the a priori information of the k-th one. By using this turbo scheme in the receiver, the equalizer is fed back with the statistics $\mathbb{E}[u_k]$ and $\mathbb{V}[u_k]$, which are obtained from the updated a priori probability $p_D(u_k)$ in Fig. 1 as

$$\mathbb{E}(u_k) = \sum_{u\in\mathcal{A}} u \cdot p_D(u_k = u), \tag{10}$$

$$\mathbb{V}(u_k) = \sum_{u\in\mathcal{A}} (u - \mathbb{E}[u_k])^* (u - \mathbb{E}[u_k]) \cdot p_D(u_k = u). \tag{11}$$

At this point, it is important to remark that since the variances are no longer equal, the matrix to invert in (8) is not circulant. Therefore, the complexity of its inversion is of order $\mathcal{O}(N^2L)$, as discussed later in this paper.

III. Expectation Propagation

Expectation propagation [26]–[28], [43] is a technique in Bayesian machine learning to approximate an intractable probability distribution, in which inference is unfeasible, by exponential family distributions. Suppose we are given some statistical distribution with hidden variables $\mathbf{x}$ and observables $\mathcal{D}$ that factors as

$$p(\mathbf{x}|\mathcal{D}) \propto f(\mathbf{x}) \prod_{i=1}^{\mathcal{I}} t_i(\mathbf{x}), \tag{12}$$

where $f(\mathbf{x})$ belongs to an exponential family $\mathcal{F}$ with sufficient statistics $\mathbf{\Phi}(\mathbf{x})$, and $t_i(\mathbf{x})$ are nonnegative factors that do not belong to $\mathcal{F}$, making direct inference over (12) not possible. EP provides a feasible approximation to $p(\mathbf{x}|\mathcal{D})$ by an exponential distribution $q(\mathbf{x})$ from $\mathcal{F}$ which factorizes as

$$q(\mathbf{x}) \propto f(\mathbf{x}) \prod_{i=1}^{\mathcal{I}} \tilde{t}_i(\mathbf{x}), \tag{13}$$

where factors $\tilde{t}_i(\mathbf{x}) \in \mathcal{F}$ are optimized to achieve an accurate global approximation $q(\mathbf{x}) \leftarrow p(\mathbf{x}|\mathcal{D})$, which optimally satisfies $\mathbb{E}_{q(\mathbf{x})}[\mathbf{\Phi}(\mathbf{x})] = \mathbb{E}_{p(\mathbf{x}|\mathcal{D})}[\mathbf{\Phi}(\mathbf{x})]$. This is known as the moment matching solution. A feasible algorithm to approximate this solution is the sequential EP algorithm [26], [27], which optimizes each factor $\tilde{t}_i(\mathbf{x})$ in turns independently in the context of all of the remaining factors. A sketch of the EP algorithm is given in Algorithm 1 where $q^{[\ell]}(\mathbf{x})$ is the approximation to $q(\mathbf{x})$ in (13) at iteration ℓ.

IV. Block-EP turbo equalizer

The EP is endowed with a great flexibility, given by the model in (13). In order to improve the accuracy of the EP solution, it is important to retain as much structure as possible from the true distribution and separate it from the latent (unknown) factors, $t_i(\mathbf{x})$ [26]. Bearing this in mind, we develop an EP approximation to (4), namely the posterior distribution of the transmitted symbols given the channel outcome $\mathbf{y}$.

A. The BEP equalizer

The following Gaussian exponential family will be considered to find a suitable approximation to (4):

$$q(\mathbf{u}) \propto \mathcal{CN}\big(\mathbf{y} : \mathbf{H}\mathbf{u}, \sigma_w^2\mathbf{I}\big) \prod_{k=1}^{N} \exp\left(u_k^*\gamma_k + \gamma_k^* u_k - \Lambda_k u_k^* u_k\right), \tag{15}$$

Algorithm 1 The EP algorithm

Initialize approximating factors $\tilde{t}_i(\mathbf{x})$ and then $q(\mathbf{x})$ in (13).
repeat
  for $i = 1, ..., \mathcal{I}$ **do**
    1) Compute the distribution

$$\tilde{p}_i(\mathbf{x}) \propto t_i(\mathbf{x}) q^{[\ell]\setminus i}(\mathbf{x}) = t_i(\mathbf{x}) q^{[\ell]}(\mathbf{x}) / \tilde{t}_i^{[\ell]}(\mathbf{x}) \tag{14}$$

    and its moments, where $q^{[\ell]\setminus i}(\mathbf{x})$ is the so called *cavity* function.
    2) Compute the refined factor $\tilde{t}_i^{[\ell+1]}(\mathbf{x})$ by setting the moments of the distribution $\tilde{t}_i^{[\ell+1]}(\mathbf{x}) q^{[\ell]\setminus i}(\mathbf{x})$ equal to the moments of $\tilde{p}_i(\mathbf{x})$.
  end for
until convergence (or stopped criterion)

where the product of indicator functions is replaced by a product of univariate proper complex Gaussians, each parameterized by a (γ_k, Λ_k) pair, $k = 1, \ldots, N$. For any value $\gamma_k \in \mathbb{C}$ and $\Lambda_k \in \mathbb{R}_+$, $q(\mathbf{u})$ is also proper complex Gaussian $\mathcal{CN}(\mathbf{u} : \mathbf{\mu}, \mathbf{\Sigma})$ with

$$\mathbf{\Sigma} = \mathbf{R}^{-1} = (\sigma_w^{-2}\mathbf{H}^{\mathrm{H}}\mathbf{H} + \mathrm{diag}(\mathbf{\Lambda}))^{-1}, \tag{16}$$

$$\mathbf{\mu} = \mathbf{\Sigma}(\sigma_w^{-2}\mathbf{H}^{\mathrm{H}}\mathbf{y} + \mathbf{\gamma}), \tag{17}$$

exhibiting a similar structure to the LMMSE in (7). Based on these definitions in mind, we adapt the EP algorithm in Algorithm 1 to our setting. We denote the resulting algorithm block-EP (BEP) soft equalizer, and it can be found in Algorithm 2. A few remarks:

- Step 1) is the "LMMSE" step of the algorithm, as it requires computing $\mathbf{\Sigma}$ and $\mathbf{\mu}$ in (16) and (17) for the current configuration of $(\mathbf{\Lambda}, \mathbf{\gamma})$. Note that this step is equivalent to the LMMSE method in (10) and (11).
- Steps 2.1)-2.3) can be done in parallel for $k = 1, \ldots, N$. They can be seen as a refinement to the estimate computed at Step 1) by enforcing a discrete distribution. Given the factorization in (15), marginal $q^{[\ell]}(u_k)$, $k = 1, \ldots, N$ is proportional to $\exp\left(u_k^*\gamma_k + \gamma_k^* u_k - \Lambda_k u_k^* u_k\right)$. This term is canceled out in $\hat{p}^{[\ell]}(u_k)$ and the"true" discrete factor is introduced. Parameters canceled are recomputed by moment matching in (22)-(23). Eqn. (22) and (23) are proposed following the guidelines in [36, Eq. 35-36]. The parameter update in (22) may return a negative value for some k's. For those k's, we keep the values from the previous iteration. We introduce a smoothing parameter $\beta \in [0, 1]$ and a small constant ϵ. To avoid numerical instabilities, constant ϵ is the minimum allowed variance at each iteration, i.e., $\sigma_{p_k}^{2[\ell]} = \max(\epsilon, \sigma_{p_k}^{2[\ell]})$.

A block diagram of the BEP equalizer is included in Fig. 2 (a). The LMMSE block corresponds to Step 1). The grey block represents the refinement of the current marginals $q^{[\ell]}(u_k)$ through projection over discrete alphabet, whose output is fed back to the LMMSE block.

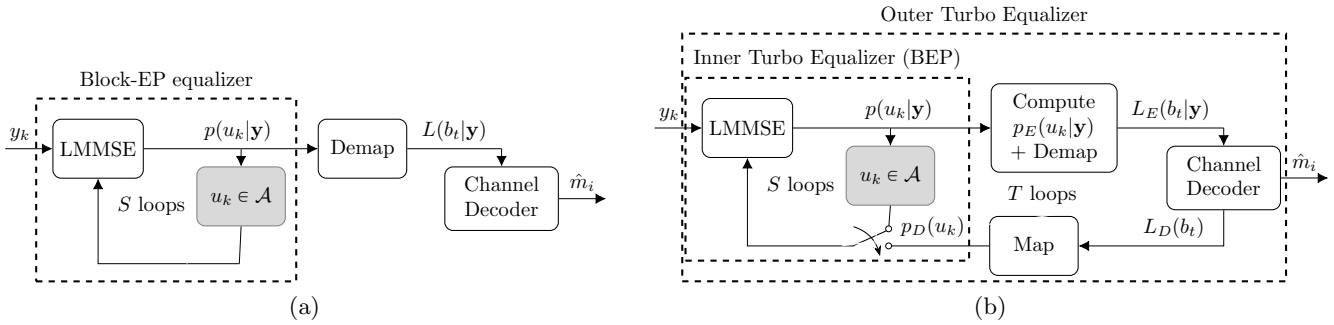


Fig. 2: In (a), BEP equalizer block diagram. In (b), Turbo BEP block diagram.

B. The Turbo BEP equalizer

The BEP equalizer can be further improved by using a turbo scheme. The resulting proposed equalizer, denoted as T-BEP, can be easily described as the T-LMMSE in Subsection II-B by just replacing (7) with the result of the BEP, Algorithm 2, in (15). A detailed implementation of the T-BEP is included in Algorithm 3. Also, a block diagram is included in Fig. 2(b). Note that the T-BEP can actually be seen as a turbo equalizer with two loops. First we run an *inner* turbo scheme, i.e. BEP. After S iterations of the BEP iterative procedure, the extrinsic LLR is given to the decoder in an *outer* loop, which is repeated T times. We propose to approximate the extrinsic probabilities by the cavity functions at the end of the EP algorithm, see (18). In this second stage the restrictions from the channel coding are exploited. The output of the channel decoder is used to initialize the BEP iterative procedure, whose outputs are then fed forward to the channel decoder. This is a major difference with respect to the previous scheme used in [5], [32], [34], where the estimates were refined only using the output of the decoder, as shown in Fig. 1, and the proposed inner loop was not present.

Algorithm 2 Block-EP equalizer (BEP)

Input: $(\gamma_k^{[1]}, \Lambda_k^{[1]})$ initialization.
for $\ell = 1, ..., S$ **do**
1) Calculate the moments of $q^{[\ell]}(\mathbf{u})$ in (15) for the current values of $\gamma_k \leftarrow \gamma_k^{[\ell]}$ and $\Lambda_k \leftarrow \Lambda_k^{[\ell]}$.
for $k = 1, ..., N$ **do**
2.1) Compute the k-th marginal of the distribution $q^{[\ell]}(\mathbf{u})$, denoted as $q^{[\ell]}(u_k) = \mathcal{CN}\Big(u_k : \mu_k^{[\ell]}, \sigma_k^{2[\ell]}\Big)$, and the distribution

$$q^{[\ell]\backslash k}(u_k) = \frac{q^{[\ell]}(u_k)}{\exp\left(u_k^* \gamma_k^{[\ell]} + \gamma_k^{[\ell]^*} u_k - \Lambda_k^{[\ell]} u_k^* u_k\right)} \sim$$
$$\sim \mathcal{CN}\Big(u_k : z_k^{[\ell]}, v_k^{2[\ell]}\Big), \tag{18}$$

namely *cavity* marginal function, where

$$v_k^{2[\ell]} = \frac{\sigma_k^{2[\ell]}}{1 - \sigma_k^{2[\ell]} \Lambda_k^{[\ell]}}, \quad z_k^{[\ell]} = v_k^{2[\ell]} \left(\frac{\mu_k^{[\ell]}}{\sigma_k^{2[\ell]}} - \gamma_k^{[\ell]}\right).$$

2.2) Obtain the distribution $\hat{p}^{[\ell]}(u_k) \propto q^{[\ell]\backslash k}(u_k) \mathbb{I}_{u_k \in \mathcal{A}}$ and estimate its mean $\mu_{p_k}^{[\ell]}$ and variance $\sigma_{p_k}^{2[\ell]}$.
2.3) Set the mean and variance of the unnormalized Gaussian distribution

$$q^{[\ell]\backslash k}(u_k) \exp\left(u_k^* \gamma_{k,new}^{[\ell+1]} + \gamma_{k,new}^{[\ell+1]^*} u_k - \Lambda_{k,new}^{[\ell+1]} u_k^* u_k\right) \tag{19}$$

equal to $\mu_{p_k}^{[\ell]}$ and $\sigma_{p_k}^{2[\ell]}$. To this end, compute:

$$\Lambda_{k,new}^{[\ell+1]} = \left(\sigma_{p_k}^{-2[\ell]} - v_k^{-2[\ell]}\right), \tag{20}$$
$$\gamma_{k,new}^{[\ell+1]} = \left(\mu_{p_k}^{[\ell]} \sigma_{p_k}^{-2[\ell]} - z_k^{[\ell]} v_k^{-2[\ell]}\right). \tag{21}$$

2.4) Update the values as

$$\Lambda_k^{[\ell+1]} = \beta \Lambda_{k,new}^{[\ell+1]} + (1-\beta)\Lambda_k^{[\ell]}, \tag{22}$$
$$\gamma_k^{[\ell+1]} = \beta \gamma_{k,new}^{[\ell+1]} + (1-\beta)\gamma_k^{[\ell]}. \tag{23}$$

end for
end for
With the values $\boldsymbol{\gamma}^{[S+1]}, \boldsymbol{\Lambda}^{[S+1]}$ obtained after EP algorithm, calculate the final distribution $q(\mathbf{u})$ in (15).

Algorithm 3 Block-EP turbo equalizer (T-BEP)

0) Initialize $(\gamma_k^{[1]}, \Lambda_k^{[1]}) = (0, E_s^{-1})$.
for $t = 1, \ldots, T$ **do**
1) With the current initialization to $(\gamma_k^{[1]}, \Lambda_k^{[1]})$, run BEP equalizer in Algorithm 2.
2) Compute an estimate to the extrinsic LLRs, $L_E(b_t|\boldsymbol{y})$, by feeding

$$p_E(u_k|\mathbf{y}) = q^{[S+1]\backslash k}(u_k) \tag{24}$$

to the demapper. Feed $L_E(b_t|\boldsymbol{y})$ to the channel decoder.
3) From the channel decoder per-bit soft output, recompute a probability distribution for each symbol $p_D(u_k)$ from the decoder and compute its mean $\mathbb{E}[u_k]$ and variance $\mathbb{V}[u_k]$ given by (10) and (11).
4) Re-initialize $(\gamma_k^{[1]}, \Lambda_k^{[1]})$ to $(\mathbb{E}[u_k]\mathbb{V}[u_k]^{-1}, \mathbb{V}[u_k]^{-1})$.
end for

C. Efficient implementation

All $(\gamma_k^{[\ell]}, \Lambda_k^{[\ell]})$ pairs for $k = 1, ..., N$ can be updated in parallel. The most involved step is the computation of an N-dimensional inverse matrix in (16) for each ℓ-iteration, whose complexity is dominated by its size, i.e., $\mathcal{O}(N^3)$. Once this inverse is computed, the parallel update of all pairs $(\gamma_k^{[\ell]}, \Lambda_k^{[\ell]}) \leftarrow (\gamma_k^{[\ell+1]}, \Lambda_k^{[\ell+1]})$ by means of step 2 and 3 in

Algorithm 3 has a smaller computational complexity, linear in NM. To reduce the complexity of the matrix inversion, we propose to exploit the banded structure of the channel matrix along with the short length of the channel compared to N. The matrix $\mathbf{R}$ in (16) is a symmetric, positive-definite and banded matrix with bandwidth $2L-1$. We can decompose $\mathbf{R}$ using the band Cholesky factorization [44] such that

$$\mathbf{R} = \mathbf{G}\mathbf{G}^{\top}, \tag{25}$$

where $\mathbf{G}$ is a lower triangular banded matrix with bandwidth L that can be computed with NL^2 operations. Then, the inverse of the covariance matrix can be rewritten as

$$\mathbf{\Sigma} = \mathbf{R}^{-1} = \mathbf{G}^{-1^{\top}}\mathbf{G}^{-1}. \tag{26}$$

We invert matrix $\mathbf{G}$ by Gauss-Jordan elimination. For every diagonal element, say $G_{k,k}$, we divide row k of less than N non-null elements by $G_{k,k}$ and cancel all the $L-1$ lower elements of its column. Repeated for the whole diagonal yields a complexity of $\mathcal{O}(N^2L)$.

D. Convergence

Although the convergence is not guaranteed, we concluded empirically that in about $S = 10$ iterations the distribution $q(\mathbf{u})$ constructed in (15) tipically reaches a stationary value. Experimental results show that controlling numerical instabilities in the parameter updates in the turbo case ($T > 0$) is simply done by setting $\epsilon = 10^{-9}$ and $\beta = 0.1$. This solution is robust regardless the constellation order, SNR or channel realization. In the equalization case ($T = 0$), we have found that results can be improved if ϵ is first kept constant to a relatively high constant (0.5) to then reduce it. More precisely, we have used $\epsilon = 2^{-\max(\ell-5,1)}$. For equalizing 64-QAM constellations, the best performance has been reported by setting fixed to $\epsilon = 0.9$. We have selected the previous values after extensive experimentation as a trade off between convergence speed and accuracy. We emphasize again that the latter heuristics to control stability in the equalization case are not needed if the turbo scheme is used, as the feedback loop naturally stabilizes the BEP output. In Fig. 3, we include a representative example of the convergence by depicting the evolution of some components of the mean vector $\boldsymbol{\mu}$ in (17) and covariance matrix $\mathbf{\Sigma}$ in (16) along different values of S in the low-E_b/N_0 regime (specifically, $E_b/N_0 = 3$ dB) for a given observation, $\mathbf{y}$, with a 16-PAM modulation, $L = 4$ and $T = 0$. As shown in Fig. 3, approximately after $S = 10$ iterations the EP equalizer reaches a stationary value for the mean and variance, equal to the value provided by the optimal BCJR approach.

E. Performance analysis

We compare probabilistic equalizers using the mutual information between the transmitted symbol u_k and detected symbol $\hat{u}_k$, distributed according to the estimation of the posterior distribution of u_k given $\mathbf{y}$:

$$I(u_k, \hat{u}_k) = \sum_{u_k}\sum_{\hat{u}_k} p(u_k, \hat{u}_k) \log_2 \frac{p(\hat{u}_k|u_k)}{p(\hat{u}_k)}, \tag{27}$$

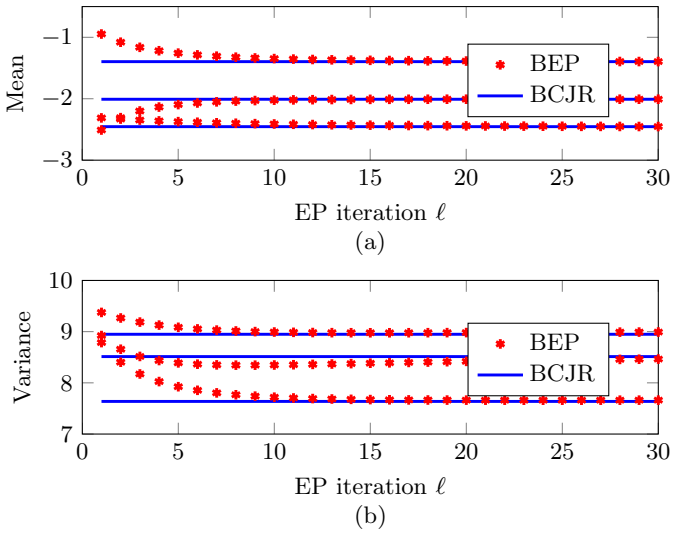


Fig. 3: Evolution of the mean (a) and variance (b) of 3 randomly chosen entries of the approximate posterior as EP iterates with 16-PAM, $L = 4$ and $E_b/N_0 = 3$ dB.

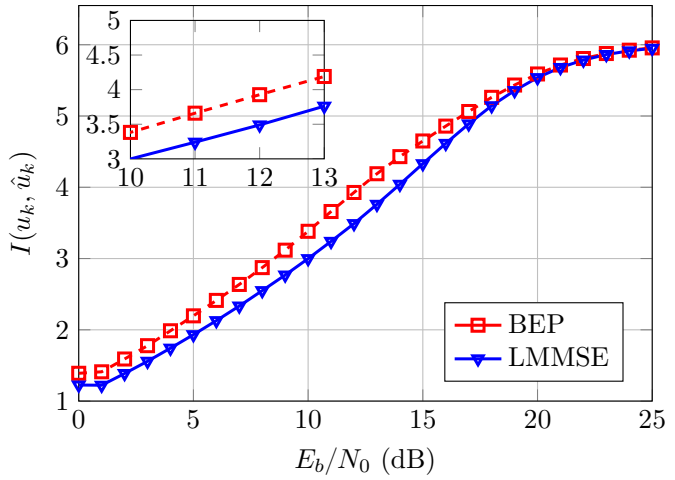


Fig. 4: Mutual information for BEP and LMMSE for 64-QAM and random channels with $L = 7$.

where note that $p(u_k, \hat{u}_k)$ is the joint probability distribution of u_k and $\hat{u}_k$ after marginalizing the channel output $\mathbf{y}$, the channel impulse response (in the case we consider it random) and the rest of symbols in the sequence $\mathbf{u}$. We resort to Monte Carlo estimates to generate samples from $p(u_k, \hat{u}_k)$ and then evaluate $I(u_k, \hat{u}_k)$ from these samples. We assume that the channel taps are Gaussian distributed, but are perfectly known at the receiver. First, we collect $N \in \mathbb{Z}_+$ samples from the joint distribution of $\mathbf{u}, \mathbf{h}, \mathbf{y}$ and $\hat{u}_k$ using standard sampling techniques in directed graphical models [43]. The key aspect is to note that, during the sampling procedure, $p(\hat{u}_k|\mathbf{y})$ is dependent on the considered probabilistic detection method. After N samples of possible $(u_k, \hat{u}_k)$ pairs have been collected, we use these samples to estimate $p(\hat{u}_k)$, $p(\hat{u}_k|u_k)$ and, finally, $I(u_k, \hat{u}_k)$ in (27). The higher $I(u_k, \hat{u}_k)$ is for each probabilistic equalizer, the closer we perform to channel capacity. In Fig. 4 we depict the mutual information in (27)

computed for $N = 10^6$ samples per E_b/N_0 point and averaged over all the symbols in the transmitted sequence, for the BEP and LMMSE algorithms, considering a 64-QAM constellation and channels of $L = 7$ complex-valued taps. Observe that, for very small E_b/N_0, the noise is so large that both methods achieve a small mutual information. On the other hand, for very large E_b/N_0 values, both methods converge to 6 bits, i.e., the number of bits transmitted per QAM symbol. Any probabilistic method will eventually saturate to this value. However, the key aspect is to be able to design a probabilistic equalizer able to improve the mutual information for intermediate E_b/N_0 values. Observe that before the saturation, the BEP achieves a gain w.r.t. LMMSE of around 1.5dB.

F. Computational Complexity

A detailed comparison of the complexity for the T-BEP and the T-LMMSE is included in Table I. We also include the computational complexity of the T-BCJR and its approximated approaches with turbo. From the computational point of view, as M and/or L grow, the BCJR and approximated approaches are unaffordable.

Algorithm	Complexity
T-BEP	$S'(LN^2 + NM) + \alpha + S'T(LN^2 + \alpha)$
T-LMMSE	$N \log N + \alpha + T(LN^2 + \alpha)$
T-BCJR	$(T + 1)(NM^L + \alpha)$
Approx. T-BCJR	$(T + 1)(NM_eM + \alpha)$

TABLE I: Complexity comparison between algorithms, where $S' = S + 1$ and α is the complexity of the LDPC decoder.

V. Experimental results

In this section, we illustrate the good performance of the BEP equalizer for different scenarios. Each channel tap is i.i.d. complex circular Gaussian distributed with zero mean and variance equal to $1/L$. The channel response is normalized. We average the BER over 1000 random frames per channel realization. We limit to 5 the absolute value of LLRs given to the decoder in order to avoid very confident probabilities which negatively affect its estimations. In the following, when mentioning approximated BCJR algorithms we refer to the M-BCJR [14], M*-BCJR [15], RS-BCJR [10], NZ and NZ-OS [17] solutions. We denote by NZ the approach consisting in running FT, BT or DT-NZ in [17] depending on the phase of the channel. If output saturation is used we refer to them as NZ-OS. We use a (3,6)-regular LDPC of rate 1/2, for a maximum of 200 iterations using the belief propagation as decoder [45], [46]. The codes were generated using the progressive edge-growth algorithm [47]. The maximum number of iterations of the LDPC decoder is set to 100 in every iteration of the turbo equalizer.

A. BEP soft equalizer

In Fig. 5 and Fig. 6 we show a comparison for some typical channels found in the literature using codewords of $V = 1024$ bits. In Fig. 5 we simulate the following scenario in [17]:

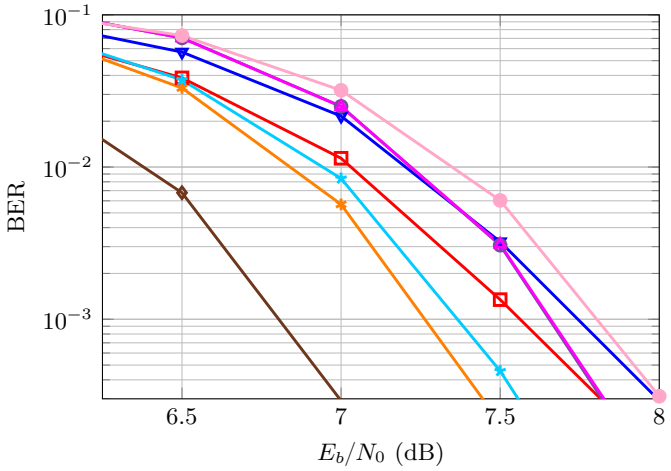


Fig. 5: BER for LMMSE (▽), BEP (□), BCJR (◇), M-BCJR (○), M*-BCJR (✳), RS-BCJR (⋆), NZ (△) and NZ-OS (●) equalizers for BPSK and the minimum phase channel $\mathbf{h} = \frac{1}{\sqrt{140}}[7\ 6\ 5\ 4\ 3\ 2\ 1]^\top$.

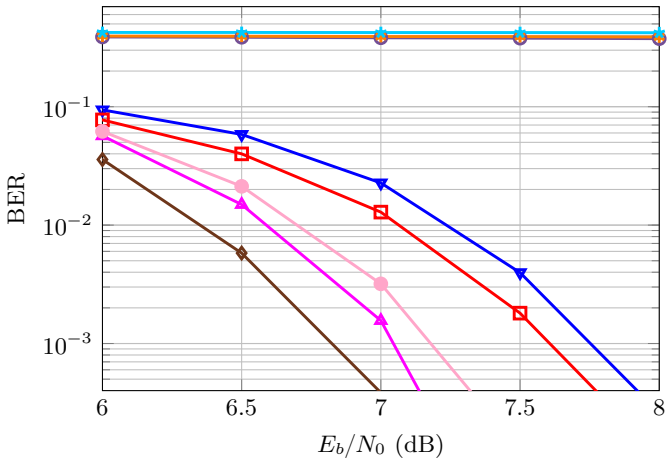


Fig. 6: BER for LMMSE (▽), BEP (□), BCJR (◇), M-BCJR (○), M*-BCJR (✳), RS-BCJR (⋆), NZ (△) and NZ-OS (●) equalizers for BPSK and the maximum phase channel $\mathbf{h} = \frac{1}{\sqrt{140}}[1\ 2\ 3\ 4\ 5\ 6\ 7]^\top$.

minimum phase channel $\mathbf{h} = [7\ 6\ 5\ 4\ 3\ 2\ 1]^\top/\sqrt{140}$, BPSK symbols and $M_e = 4$ states out of 64. In Fig. 6 we simulate the maximum phase channel $\mathbf{h} = [1\ 2\ 3\ 4\ 5\ 6\ 7]^\top/\sqrt{140}$ and BPSK symbols with $M_e = 8$ states for the approximated solutions as in [17]. To study the performance for other channels, in Fig. 7 and Fig. 8 we include the averaged BER over 100 random channels with $L = 5$ real-valued taps. We fix to $M_e = 8$ the number of states for the approximated solutions. In Fig. 7, we consider BPSK modulation, hence the BCJR has 16 states per step, while in Fig. 8 a 4-PAM is used, increasing the number of states to 256.

In Fig. 5 and Fig. 6 we can observe that approximated methods M-BCJR, M*-BCJR, RS-BCJR are quite sensitive to the channel realization. This is not only caused because they are based on just a forward or a backward strategy, but also because their parameters need to be tuned according to the particular channel. For this reason, the approximations M-BCJR, M*-BCJR and RS-BCJR fail with maximum phase

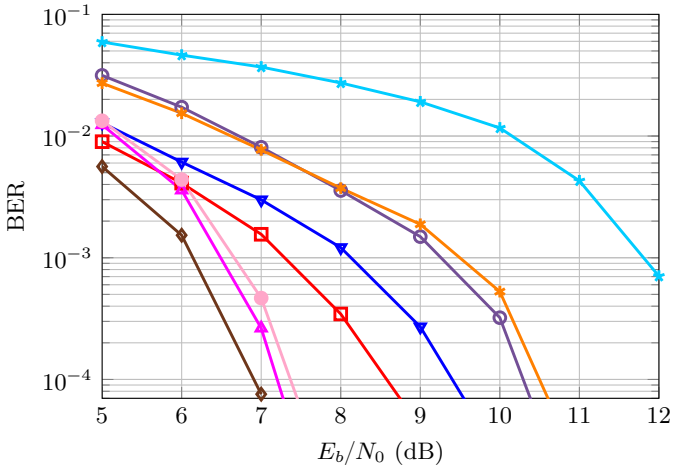


Fig. 7: BER for LMMSE (▽), BEP (▫), BCJR (◇), M-BCJR (○), M*-BCJR (∗), RS-BCJR (⋆), NZ (△) and NZ-OS (•) equalizers for BPSK and 100 random channels with $L = 5$.

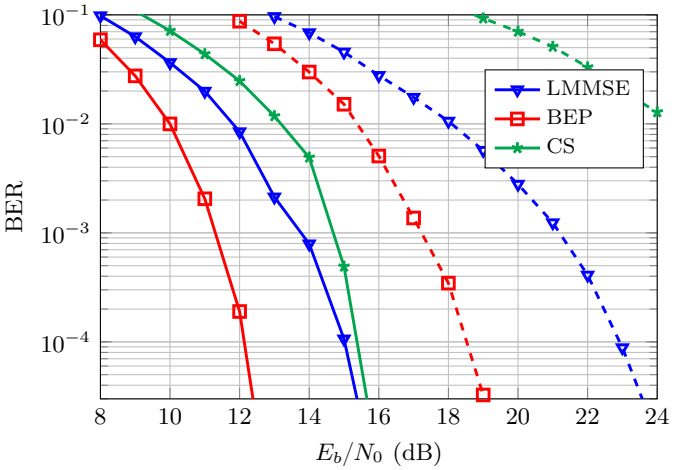


Fig. 9: BER for LMMSE (▽), BEP (▫) and CS (⋆) equalizers for 16-QAM (solid lines) and 64-QAM (dashed lines) and 100 random channels with $L = 7$.

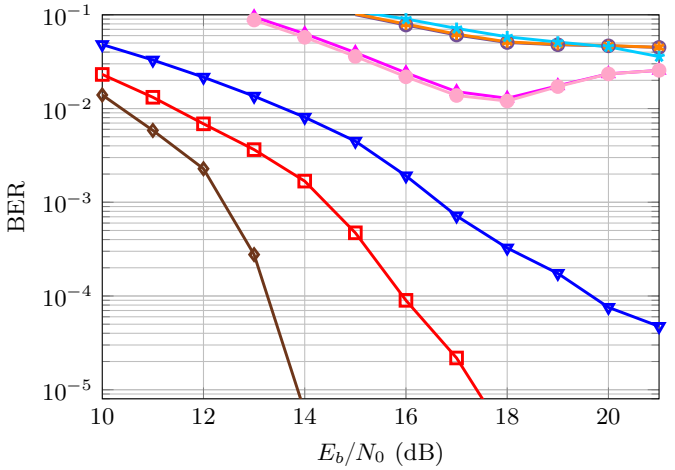


Fig. 8: BER for LMMSE (▽), BEP (▫), BCJR (◇), M-BCJR (○), M*-BCJR (∗), RS-BCJR (⋆), NZ (△) and NZ-OS (•) equalizers for 4-PAM and 100 random channels with $L = 5$.

channels, as observed in Fig. 6. For averaged channels, see Fig. 7, NZ and NZ-OS exhibit a good performance at a low computational complexity due to the low state reduction ratio (M^L/M_e). However, these solutions fail if M_e does not grow accordingly, as can be observed in Fig. 8. This means that if the number of states, M_e, is kept as a fraction of the total number, M^L, they may exhibit a good performance. But if a large number of states is unaffordable, a fraction of it becomes also intractable. The LMMSE in these experiments does not fail as the approximated approaches do, but its performance degrades significantly compared to the BER of the BCJR in Fig. 8. Finally, the BEP exhibits a quite robust behavior, closer to the BCJR performance. Note here that the number of states is low, and that the BCJR can be used with optimal results at a low complexity.

For multilevel constellations and a channel with a few taps, the BCJR or their approximations are no longer computationally affordable. In the case of large dimensions, filter-based equalizers of the MMSE type are preferred [20]. In Fig. 9, we depict the BER curves for BEP, LMMSE equalization and the algorithm in [19] that we denote as channel shortening (CS), considering codewords of $V = 4096$ bits, 100 random channels of $L = 7$ complex-valued taps and two different modulations: 16-QAM (solid lines) and 64-QAM (dashed lines). The CS algorithm has been simulated with a reduced memory $\nu = 2$ ($\nu = 1$) for the 16-QAM (64-QAM) case. We observe in this experiment that the BEP exhibits significant improvements with respect to the LMMSE and CS solution for both constellations.

B. Turbo BEP

To improve the estimates the turbo-equalization can be used. We simulate scenarios with large dimensions, where the only computationally feasible algorithms are the BEP and LMMSE solutions. In Fig. 10 we simulate a 16-QAM constellation, averaging over 100 random channels with a large memory, $L = 20$ complex-valued taps and codewords of $V = 1024$ bits. We represent the first 3 iterations of the turbo scheme, since we found no further improvement for $T \geqslant 3$. In Fig. 11 and Fig. 12, we depict the BER curves for turbo BEP and LMMSE equalization, averaged over 100 random channels of $L = 7$ complex-valued taps, 64-QAM modulation and codewords of $V = 1024$ and $V = 4096$ bits, respectively, for $T = 3$.

The results are remarkable. The BEP with no feedback from the decoder is 1.5-4.5 dB away from the LMMSE equalizer for BER=10^{-4}, depending on the channel length, L, and the constellation size, M. It even outperforms the turbo LMMSE equalizer. The BEP estimation can be further improved using the turbo scheme, in about 1 or 1.5 dB depending on the codeword length.

VI. Conclusions

When the number of states involved in the BCJR equalization or its approximations is high, their computational complexity is unaffordable. Approximated strategies are interesting whenever the number of states is not too high. Therefore,

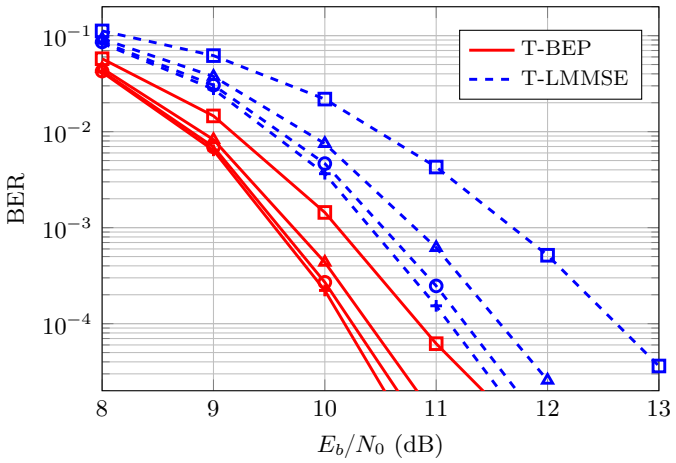


Fig. 10: BER for T-BEP (red lines, solid) and T-LMMSE (blue lines, dashed) equalizers using the outer loop for 16-QAM and 100 random channels with $L = 20$. No feedback (▫), one loop (▵), two loops (○) and three loops (+).

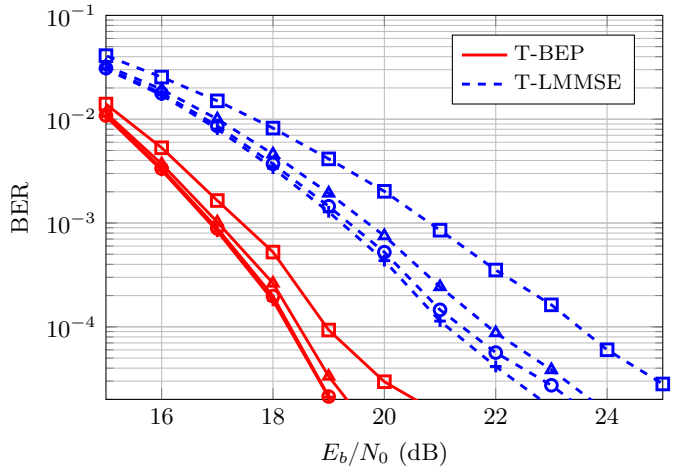


Fig. 11: BER for T-BEP (red lines, solid) and T-LMMSE (blue lines, dashed) equalizers using the outer loop for 64-QAM, 100 random channels with $L = 7$ and codewords of $V = 1024$ bits. No feedback (▫), one loop (▵), two loops (○) and three loops (+).

in the case of multilevel constellations with moderate or large channel lengths, it is preferable to resort to MMSE type equalizers. The turbo LMMSE equalization improves the estimations by feeding back the output of the channel decoder to the equalizer. In this paper we propose the EP as a novel alternative, where the whole posterior probabilities are approximated in a complex-valued formulation. This equalizer, denoted as BEP, can be cast as an LMMSE with an inner turbo scheme. The BEP exploits the fact that the transmitted symbols belong to a known constellation to improve the estimations. This soft equalizer outperforms the turbo LMMSE equalization. The BEP equalization can be further improved by exploiting the channel decoder output to propose the turbo BEP equalizer. This can be interpreted as an outer turbo equalization as compared to the proposed EP based inner turbo equalization. In the included experiments, we report gains in the range 1.5-5 dB with respect to the LMMSE. We have

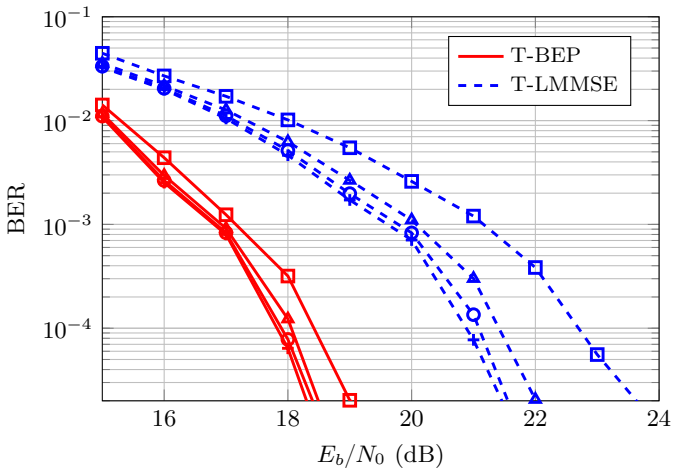


Fig. 12: BER for T-BEP (red lines, solid) and T-LMMSE (blue lines, dashed) equalizers using the outer loop for 64-QAM, 100 random channels with $L = 7$ and codewords of $V = 4096$ bits. No feedback (▫), one loop (▵), two loops (○) and three loops (+).

focused on block or batch solutions, to report computational complexities of order $\mathcal{O}(N^2L)$. The development of windowed or filtered solutions remains as future research, to reduce the computational complexity to be linear in the frame length.

REFERENCES

[1] R. Schober, "Detection and estimation of signals in noise," University of British Columbia, Vancouver, Tech. Rep., 2010.

[2] J. G. Proakis, *Digital Communications*, 5th ed. New York, NY: McGraw-Hill, 2008.

[3] S. Haykin, *Communication Systems*, 5th ed. Wiley Publishing, 2009.

[4] L. Salamanca, J. J. Murillo-Fuentes, and F. Pérez-Cruz, "Bayesian equalization for LDPC channel decoding," *IEEE Trans. on Signal Processing*, vol. 60, no. 5, pp. 2672–2676, May 2012.

[5] M. Tüchler and A. Singer, "Turbo equalization: An overview," *IEEE Trans. on Information Theory*, vol. 57, no. 2, pp. 920–952, Feb 2011.

[6] C. Douillard, M. Jezequel, C. Berrou, P. Didier, and A. Picart, "Iterative correction of intersymbol interference: turbo-equalization," *European Trans. on Telecommunications*, vol. 6, no. 5, pp. 507–512, Sep 1995.

[7] R. Koetter, A. Singer, and M. Tüchler, "Turbo equalization," *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 67–80, Jan 2004.

[8] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Trans. on Information Theory*, vol. 20, no. 2, pp. 284–287, 1974.

[9] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Information Theory*, vol. 47, no. 2, pp. 498 –519, Feb 2001.

[10] G. Colavolpe, G. Ferrari, and R. Raheli, "Reduced-state BCJR-type algorithms," *IEEE Journal on Sel. Areas in Communications*, vol. 19, no. 5, pp. 848–859, May 2001.

[11] M. Eyuboglu and S. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. on Communications*, vol. 36, no. 1, pp. 13–20, Jan 1988.

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

[13] P. Chevillat and E. Eleftheriou, "Decoding of trellis-encoded signals in the presence of intersymbol interference and noise," *IEEE Trans. on Communications*, vol. 37, no. 7, pp. 669–676, Jul 1989.

[14] V. Franz and J. Anderson, "Concatenated decoding with a reduced-search BCJR algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 186–195, Feb 1998.

[15] M. Sikora and D. Costello, "A new SISO algorithm with application to turbo equalization," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Sep 2005, pp. 2031–2035.

[16] C. Vithanage, C. Andrieu, and R. Piechocki, "Approximate inference in hidden Markov models using iterative active state selection," *IEEE Signal Processing Letters*, vol. 13, no. 2, pp. 65–68, Feb 2006.

[17] D. Fertonani, A. Barbieri, and G. Colavolpe, "Reduced-complexity BCJR algorithm for turbo equalization," *IEEE Trans. on Communications*, vol. 55, no. 12, pp. 2279–2287, Dec 2007.

[18] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 2nd ed. Prentice Hall, 1996.

[19] F. Rusek and A. Prlja, "Optimal channel shortening for MIMO and ISI channels," *IEEE Trans. on Wireless Communications*, vol. 11, no. 2, pp. 810–818, Feb 2012.

[20] C. Berrou, *Codes and turbo codes*, ser. Collection IRIS. Springer Paris, 2010.

[21] G. Colavolpe and G. Germi, "On the application of factor graphs and the sum-product algorithm to ISI channels," *IEEE Trans. on Communications*, vol. 53, no. 5, pp. 818–825, May 2005.

[22] J. Hu, H. Loeliger, J. Dauwels, and F. Kschischang, "A general computation rule for lossy summaries/messages with examples from equalization," in *Proc. 44th Allerton Conf. Communication, Control, and Computing*, Sep 2006, pp. 27–29.

[23] Q. Guo and L. Ping, "LMMSE turbo equalization based on factor graphs," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 2, pp. 311–319, Feb 2008.

[24] G. Colavolpe, D. Fertonani, and A. Piemontese, "SISO detection over linear channels with linear complexity in the number of interferers," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 8, pp. 1475–1485, Dec 2011.

[25] P. Sun, C. Zhang, Z. Wang, C. Manchon, and B. Fleury, "Iterative receiver design for ISI channels using combined belief- and expectation-propagation," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1733–1737, Oct 2015.

[26] T. P. Minka, "A family of algorithms for approximate Bayesian inference," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[27] T. Minka, "Expectation propagation for approximate Bayesian inference," in *Proc. 17th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2001, pp. 362–369.

[28] M. Seeger, "Expectation propagation for exponential families," *Univ. Calif., Berkeley, CA, USA, Tech. Rep.*, 2005.

[29] K. Muranov, "Survey of MMSE channel equalizers," University of Illinois, Chicago, Tech. Rep.

[30] J. Wu and Y. Zheng, "Low complexity soft-input soft-output block decision feedback equalization," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, Nov 2007, pp. 3379–3383.

[31] S. Kim, J. Lee, and Y. Kim, "Adaptive Cholesky based MMSE equalizer in GSM," in *Proc. IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, Nov 2008, pp. 886–889.

[32] M. Tüchler, A. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. on Signal Processing*, vol. 50, no. 3, pp. 673–683, Mar 2002.

[33] L. Liu and L. Ping, "An extending window MMSE turbo equalization algorithm," *IEEE Signal Processing Letters*, vol. 11, no. 11, pp. 891–894, Nov 2004.

[34] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Trans. on Communications*, vol. 50, no. 5, pp. 754–767, May 2002.

[35] M. Senst and G. Ascheid, "How the framework of expectation propagation yields an iterative IC-LMMSE MIMO receiver," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM)*, Dec 2011, pp. 1–6.

[36] J. Céspedes, P. M. Olmos, M. Sánchez-Fernández, and F. Pérez-Cruz, "Expectation propagation detection for high-order high-dimensional MIMO systems," *IEEE Trans. on Communications*, vol. 62, no. 8, pp. 2840–2849, Aug 2014.

[37] J. Céspedes, P. M. Olmos, M. Sánchez-Fernández, and F. Pérez-Cruz, "Improved performance of LDPC-coded MIMO systems with EP-based soft-decisions," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Jun 2014, pp. 1997–2001.

[38] P. M. Olmos, J. J. Murillo-Fuentes, and F. Pérez-Cruz, "Tree-structure expectation propagation for LDPC decoding over the BEC," *IEEE Trans. on Information Theory*, vol. 59, no. 6, pp. 3354–3377, 2013.

[39] L. Salamanca, P. M. Olmos, F. Pérez-Cruz, and J. J. Murillo-Fuentes, "Tree-structured expectation propagation for LDPC decoding over BMS channels," *IEEE Trans. on Communications*, vol. 61, no. 10, pp. 4086–4095, Oct 2013.

[40] Y. Qi and T. Minka, "Window-based expectation propagation for adaptive signal detection in flat-fading channels," *IEEE Trans. on Wireless Communications*, vol. 6, no. 1, pp. 348–355, Jan 2007.

[41] I. Santos, J. J. Murillo-Fuentes, and P. M. Olmos, "Block expectation propagation equalization for ISI channels," in *Proc. 23rd European Signal Processing Conference (EUSIPCO)*, Sep 2015, pp. 379–383.

[42] P. J. Schreier and L. L. Scharf, *Statistical Signal Processing of Complex-Valued Data. The Theory of Improper and Noncircular Signals*. Cambridge, UK: Cambridge University Press, 2010.

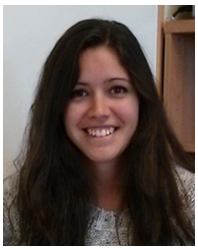
[43] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag, New York, 2006.

[44] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD, USA: Johns Hopkins University Press, 1996.

[45] T. J. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, Mar. 2008.

[46] L. Salamanca, J. J. Murillo-Fuentes, P. M. Olmos, F. Pérez-Cruz, and S. Verdú, "Approaching the DT bound using linear codes in the short blocklength regime," *IEEE Communications Letters*, vol. 19, no. 2, pp. 123–126, Feb 2015.

[47] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. on Information Theory*, vol. 51, no. 1, pp. 386–398, 2005.

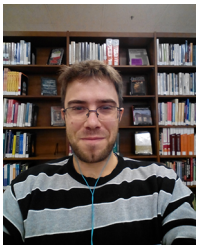


Irene Santos received the M.Sc. degree in telecommunication (2013) and electrical engineering (2014) from Universidad de Sevilla, where she is currently a doctoral candidate at the Department of Signal Theory and Communications. She has been a visiting researcher at Stony Brook University (USA). Her principal research interests are focused on Bayesian inference techniques and their application to digital communication systems.

Juan José Murillo-Fuentes received his Telecommunication Engineering degree in 1996 from Universidad de Sevilla, and his Ph.D. degree in Telecommunication Engineering in 2001 from Universidad Carlos III de Madrid, Spain. Since 2016 he is full professor at the Universidad de Sevilla. His research interests lie in algorithm development for signal processing and machine learning, and their applications to digital communication systems and learning from images.

Rafael Boloix-Tortosa received the M.Sc. degree in 2000 and the Ph.D. degree in 2005 from the Universidad de Sevilla, Spain, both in telecommunication engineering. He is currently an assistant professor at the Universidad de Sevilla, Dep. of Signal Processing and Communications. He has been a visiting researcher at the University of Edinburgh. His current research interests include algorithm development for complex-valued machine learning and applications to communications and signal processing.

Eva Arias-de-Reyna (M'07) received the degree in Telecommunication Engineering and the Ph.D. degree from the Universidad de Sevilla, Spain, in 2001 and 2007, respectively. She is currently assistant professor with the Dep. of Signal Theory and Communications at the Universidad de Sevilla, where she joined in 2002 after working in the industry for a year. Her current research interests include machine learning, communication receiver design, localization techniques and UWB technology.



Pablo M. Olmos received the M.Sc. degree in 2008 and the Ph.D. degree in 2011 from the Universidad de Sevilla, both in telecommunication engineering. He is currently an assistant professor at Universidad Carlos III de Madrid. Dr. Olmos has held appointments as visiting researcher at Princeton University, EPFL, Notre Dame University, ENSEA and Bell Labs. His research interests range from approximate inference methods for Bayesian machine learning to information theory and digital communications.