Reducing the Average Complexity of LDPC Decoding for Incremental Redundancy Scheme

S. Sesia, G. Vivier, G. Caire [1]
Institut Eurecom, Sophia Antipolis and Centre de Recherche Motorola, Gif sur-Yvette
Phone: (+33) 169352575 , Fax: (+33)169352501
E-mail: {sesia,caire}@eurecom.fr,vivier@crm.mot.com

Abstract: *We consider LDPC codes with Belief Propagation decoding algorithm for packetized data transmission over block-fading channels. We propose a simple method to reduce the average decoder complexity detecting if successful decoding can be achieved with high probability. We evaluate the average complexity of the proposed algorithm and compare it to that of a conventional decoder in the context of a type II Hybrid-ARQ protocol.*

Keywords: Complexity, Iterative Decoding, LDPC codes, Hybrid ARQ Protocols.

1. INTRODUCTION

In [1] the authors analyzed criteria to stop the iteration process in turbo decoding; here we propose a simple method to compute implicitly an *approximated* region of convergence of the Belief Propagation (BP) decoder for a given LDPC code. By using our method, we can check very efficiently if the vector of received SNRs is such that successful decoding is expected with high probability. Then, the iterative decoder is triggered only if the vector of received SNRs is in the region. This reduces dramatically the expected complexity of the decoder, with very important savings in both computation time and battery energy.

Our scheme is suited for the Incremental Redundancy (IR) HARQ protocol: the rationale for using LDPC codes for this IR-HARQ scheme, and the relative performance analysis, are provided in [4].

2. CONTEXT AND PROBLEM

We consider slotted transmission where each slot is affected by an independent fading coefficient, according to the block-fading channel model [2].

In the following, because of space limitations, we assume that the reader is familiar with message passing iterative decoding and Density Evolution (DE), for further details see [3] and references therein. An LDPC ensemble is defined by its left and right degree distributions $\lambda(x) = \sum_{k\geq 2}^{d_v} \lambda_k x^{k-1}$ and $\rho(x) = \sum_{j\geq 2}^{d_c} \rho_j x^{j-1}$, where λ_k (ρ_j) represents the fraction of edges emanating from a variable (check) node of degree k (j). The performance of BP iterative decoding for a randomly selected LDPC codes in the ensemble with a random realization of the channel noise can be predicted by DE with high probability, for large blocklength. For standard time-invariant channels (e.g., the binary-input AWGN channel), it can be shown that DE converges either to zero bit-error rate (BER) or to some BER bounded away from zero depending on the channel parameter measuring the noise level, known as the *iterative decoding threshold.* This idea does not carry over straightforwardly for a time-varying channel such as in the case of the IR-HARQ protocol considered here. Let

$$\mathbf{y}_s = \sqrt{\mathcal{E}}\, c_s\, \mathbf{x}_s + \boldsymbol{\nu}_s, \quad s = 1, \dots, m \qquad (1)$$

define the block fading channel in slot s, where $\boldsymbol{\nu}_s$ is an i.i.d. complex circularly-symmetric Gaussian noise vector with per-component variance N_0, c_s is the (complex scalar) fading coefficient during slot s with power gain $\alpha_s = |c_s|^2$, $\mathbf{x}_s$ is the s-th sub-block of the LDPC codeword[1] (for simplicity, we assume BPSK modulation) and $\mathcal{E}$ is the energy per symbol. The fading is normalized so that $E[\alpha_s] = 1$. Therefore, the instantaneous received SNR in slot s is given by $\beta_s \triangleq \alpha_s \gamma$ with $\gamma = \mathcal{E}/N_0$.

The BP decoder at slot m "sees" a time-varying channel defined by the instantaneous SNRs $\{\beta_1, \dots, \beta_m\}$ and by the fact that the symbols in slots $m+1, \dots, M$ are erased (i.e., the corresponding channel outputs are zero). Clearly, no simple threshold criterion for convergence of BP can be applied here. Indeed, we might define a region of convergence for the decoder in slot m as an m-dimensional region $\mathcal{R}_m \subset \mathbb{R}_+^m$, such that if $(\beta_1, \dots, \beta_m) \in \mathcal{R}_m$, then DE converges to vanishing BER and the BP decoder applied to the actual finite-length code with channel observations $\{\mathbf{y}_1, \dots, \mathbf{y}_m\}$ yields successful decoding with high probability. In order to overcome this problem, we propose to use an on-line low-complexity approximation of DE and run it in real-time at each newly received slot before activating the BP decoder.

[1] This research was founded by Motorola Labs, Paris.

[1] Note that under IR-HARQ scheme, the LDPC codeword is divided into M bursts, each burst being transmitted as long as the decoding is erroneous.

Hopefully, the approximate DE is able to approximate accurately the convergence region $\mathcal{R}_m$ for all $m = 1, \ldots, M$. Therefore, if the approximate DE converges to zero error probability, the BP decoder is triggered and actual decoding is performed, otherwise a NACK is sent without actually performing decoding. Here we use an approximation of DE based on Gaussian Approximation [3], expressed by [4]

$$I^l_{out,v} = \frac{1}{M}\sum_{s=1}^{M} F_\lambda\big(1 - F_\rho\big(1 - I^{l-1}_{out,v}, 0\big), \beta_s\big) \quad (2)$$

with initial condition $I^0_{out,v} = 0$; $I^l_{out,v}$ is the average mutual information at the output of a variable node, and for a general distribution $g(x) = \sum_{i\geq 2} g_i x^{i-1}$ and $b \geq 0$ we define the function

$$F_g(z, b) \triangleq \sum_{i\geq 2} g_i J\left((i-1)J^{-1}(z) + b\right) \quad (3)$$

where $J(\beta_s)$ is the instantaneous mutual information per input symbol on slot s. The recursion for a given number of received blocks m with fading gains $\alpha_1, \ldots, \alpha_m$ is obtained by letting $\beta_s = \gamma\alpha_s$ for $s = 1, \ldots, m$ and $\beta_s = 0$ for $s = m+1, \ldots, M$ in (2).

When we consider infinite blocklength codes, in principle vanishing BER implies vanishing FER. In [4] we show that the condition of vanishing BER for given instantaneous SNRs $(\beta_1, \ldots, \beta_m)$ can be approximated by the condition that the one-dimensional dynamical system in (2) has a unique fixed-point $I^\infty_{out,v} = 1$ otherwise the iterative decoding is said to be non convergent. In the case of finite blocklength codes, the presence of cycles in the graph makes DE only an approximation method to evaluate the BP performance.

3. AVERAGE THROUGHPUT AND COMPLEXITY

It is important to note that this method may, in general, decrease the average throughput since it declares a decoding failure whenever DE does not converge, while there is a chance that the actual BP decoder is successful even if DE does not converge. The average throughput is a function of the frame error probability, $p(m)$ for a given number of received blocks m; in [4] we show that

$$\eta = R\,M\,\frac{1 - p(M)}{1 + \sum_{m=1}^{M-1} p(m)} \quad (4)$$

where R is the code rate, M is the number of sub-blocks and where $p(m)$, for a given LDPC random ensemble with degree distributions (λ, ρ), is expressed by

$$p(m) = \mathop{\mathbb{E}}_{\mathcal{C}(n,\lambda,\rho),\boldsymbol{\alpha}}\left[Pr(\overline{\mathcal{A}}_1, \overline{\mathcal{A}}_2, \ldots, \overline{\mathcal{A}}_m \,|\boldsymbol{\alpha}, (\lambda, \rho))\right] \quad (5)$$

where $\mathcal{A}_m$ is the event of successful decoding at step m and where the code parity-check matrix is randomly generated with uniform probability over all bipartite graphs with degree distributions λ, ρ (see [3]). We define $p^{BP}(m)$ and $p^T(m)$ as the outage probability obtained when running always the BP algorithm and when using the DE-test, and $q^{BP}(m)$, $q^T(m)$ as the probability of successful decoding at step m for the two methods; thus, redefining $\mathcal{A}_m$ as $\mathcal{A}_m = \{$The BP algorithm converges at step $m\}$, and $\mathcal{B}_m = \{$The DE test converges at step $m\}$ it follows that $q^{BP}(m) = \Pr(\overline{\mathcal{A}}_1, .., \overline{\mathcal{A}}_{m-1}, \mathcal{A}_m) = p^{BP}(m-1) - p^{BP}(m)$ and

$$q^T(m) = \sum_{i=1}^{m} \Pr\big(\overline{\mathcal{B}}_1, \ldots \overline{\mathcal{B}}_{i-1}, \mathcal{B}_i, \overline{\mathcal{A}}_i, \ldots \overline{\mathcal{A}}_{m-1}, \mathcal{A}_m\big)$$
$$= \sum_{i=1}^{m} q^T_c(m|i) \cdot q^{DE}(i) = p^T(m-1) - p^T(m) \quad (6)$$

where we have defined $q^{DE}(i) = p^{DE}(m-1) - p^{DE}(m) = \Pr(\overline{\mathcal{B}}_1, \ldots \overline{\mathcal{B}}_{i-1}, \mathcal{B}_i)$ and

$$q^T_c(m|i) = \Pr\big(\overline{\mathcal{A}}_i, .., \overline{\mathcal{A}}_{m-1}, \mathcal{A}_m \,|\, \overline{\mathcal{B}}_1, .., \overline{\mathcal{B}}_{i-1}, \mathcal{B}_i\big)$$

We call η_{std} the average throughput obtained when using always BP algorithm and η_{test} when using the DE-test based decoder; they can be obtained substituting $p^{BP}(m)$ and $p^T(m)$ in (4) repectively. Let us consider the simple case when we let the BP decoder run for a maximum number of iterations without stopping criterion, and for DE we search for the solution of the fixed point equation

$$z = \frac{1}{M}\sum_{s=1}^{M} F_\lambda(1 - F_\rho(1 - z, 0), \alpha_s\gamma) \quad (7)$$

corresponding of the fixed points of the recursion (2), over a fine grid of points over the interval $[0, 1]$. Under these simplifying hypothesis, figure 1 shows the comparison between η_{std} and η_{test} vs rate for $\gamma = 3$dB, the length of the code $n = 10000$ and the number of bursts $M = 10$. As we expect, the DE-test method does not affect the average throughput, $q^{BP}(m) \simeq q^{DE}(m) \rightarrow \eta_{std} \simeq \eta_{test}$.

Average Complexity: Simple Case. In the following we demonstrate that using this method the average complexity is greatly reduced with respect to the case when we perform always the BP decoder. We consider the simple case described before when the complexity of both BP and DE are independent on the fading realization and on the index m. We compute the average complexity under the following hypothesis: 1) Look-up table cost zero (i.e., evaluation of $J(.)$ and $J^{-1}(.)$), 2) We consider only additions, comparisons (we suppose they cost the same

$\mathcal{O}_a$), multiplications ($\mathcal{O}_m$) and logarithm and exponential ($\mathcal{O}_L$). Let C_{DE} and C_{BP} be the complexity of the DE test and BP algorithm. Calling c and d the number of different degrees in the check node and variable node degree distribution, n_e the number of edges in the graph, and n the length of the code, it follows that

$$C_{DE} \simeq [(2c + d(1 + 2M))\mathcal{O}_a + (2c + d(1 + M))\mathcal{O}_m]\, I_d \quad (8)$$

$$C_{BP} \simeq [4n_e\mathcal{O}_L + 2n_e\mathcal{O}_m + (10n_e + 2n)\,\mathcal{O}_a]\, I_b \quad (9)$$

where I_d is the number of points used to evaluate equation (7) and I_b is the maximum number of iterations of BP algorithm. Let τ be the RV denoting the number of slots needed to stop the transmission of the current codeword, $h = 1, \ldots M$ the number of slots needed to have DE convergent, $\phi = 0, \ldots M - h + 1$ the number of slots after DE have converged (including the slot for which DE converges) until BP converges, thus $\tau = h + \phi - 1$. We call $\bar{C}_{std}$ and $\bar{C}_{test}$ the average complexity when using BP always and when using the proposed method. We can always write the expected complexity conditioning on the value of τ, thus $\mathbb{E}[C] = \mathbb{E}[\,\mathbb{E}[C|\tau]\,]$, yielding

$$\begin{aligned}\bar{C}_{std} &= \mathbb{E}[C_{std}] = \mathbb{E}[\,\mathbb{E}[C_{std}|\tau]\,] = \mathbb{E}[C_{BP}\,\tau] \\ &= C_{BP}\mathbb{E}[\tau] \stackrel{(a)}{=} C_{BP}\left[1 + \sum_{m=1}^{M-1} p^{BP}(m)\right] \quad (10)\end{aligned}$$

where (a) follows from equation (4) noticing that $\eta \propto 1/\mathbb{E}[\tau]$, with $p^{BP}(m)$ defined above. The average complexity of the DE-test method can be obtained averaging over the value of h and ϕ, yielding

$$\begin{aligned}\bar{C}_{test} &= \mathbb{E}[C_{test}] = \mathbb{E}[\,\mathbb{E}[\,\mathbb{E}[\,C_{test}|h, \phi\,]\,]\,] \\ &= C_{DE}\mathbb{E}[h] \,+\, C_{BP}\mathbb{E}[\phi] \quad (11)\end{aligned}$$

where $\mathbb{E}[h]$ is the average number of slots after which DE converges $E[h] = 1 + \sum_{m=1}^{M-1} p^{DE}(m)$ By definition $E[\phi] = \sum_{i=0}^{i=M} \phi \Pr(\phi = i)$ and

$$\Pr(\phi = i) = \sum_{k=1}^{M} q_c^T(k + i - 1|k) \cdot q^{DE}(k) \quad (12)$$

The quantities $q^{BP}(m)$, $q_c^T(m|i)$ and $q^{DE}(m)$ are computed by Monte Carlo simulations.

Figure 2 gives the average complexity $\bar{C}_{test}$ and $\bar{C}_{std}$ as a function of the rate when we consider $I_d = 100$ and $I_b = 200$ and for the same setting as figure 1. As we can see the average complexity is drastically decreased when using DE-test based method.

Average Complexity: BP Stopping Criteria. Consider now the case when C_{BP} and C_{DE} are not constant but depend on the fading realization and index m. This includes cases when we consider stopping criteria to reduce the number of iterations of the BP algorithm. [1] considers stopping criteria for HARQ and Turbo Codes based on Cross-Entropy (CE) computation, whose complexity costs roughly as an additional iteration, per iteration. Since we use LDPC codes, here we introduce another simple stopping criterion based on syndrome computation. As far as the evaluation of the fixed points of DE are concerned, instead of numerically solving equation (7), we can iterate recursion (2) and declare the fixed points when the difference between two iterations is less then a certain threshold: this is shown to provide some saving.

In this case the average complexity has been computed through Monte Carlo simulations. Figure 1 shows results in terms of average throughput for all the methods considered here, BP without any countermeasure (η_{std}), BP with stopping criteria based on Cross Entropy and Syndrome computation (η_{stdCE}, η_{stdSY}), the DE-test based method for the three cases above (η_{test}, η_{testCE}, η_{testSY}). As we can see all these methods do not significantly modify the throughput. However the average complexity shown in figure 2 shows that DE-test based system decreases considerably the average complexity. It is interesting to notice that even if CE based stopping criterion introduces more complexity per iteration, on the other hand it reduces considerably the average number of iterations when choosing the correct threshold value. On the contrary the syndrome computation, used with the standard algorithm, does not significantly reduce the complexity, while on the contrary it performs well when used in conjunction with the DE-test: this can be explained considering the fact that when the BP does not converge, the probability of finding a codeword (syndrome equals to zero) is small while this probability is much higher if conditioned on convergence of BP.

Modified DE-Test. In this section we modify the DE-test in order to find a trade off between throughput and complexity. Clearly if we use a test with a very high rejection rate the average complexity can be made as small as we want but the throughput will also go to zero. Therefore by penalizing the DE-test we can reduce the complexity at the price of accepting some throughput degradation. In order to control the trade off η vs C we introduce the penalized SNR $\gamma_\Delta|_{dB} = \gamma|_{dB} - \Delta|_{dB}$ where the parameter Δ can be interpreted as an SNR margin of the actual BP decoder over the ideal BP decoder applied to an infinite length LDPC code. In order to find the fixed points of DE, we now iterate the recursion (2) substituting γ_Δ to γ. The introduction of the parameter $\Delta \geq 0$, reduces the probability of having DE convergent for a certain vector $(\alpha_1, \ldots, \alpha_m)$ at step m, thus reducing the average number of bursts processed with BP algorithm. Figure 3 and 4 show the average throughput and complexity as a function of Δ when $R = 0.3$bit/symbol and $\gamma = 3$dB. It is interesting to notice that using this simple method there are values of $\Delta \simeq 2$dB for which the throughput loss is negligible (~ 0.55bit/sec/Hz vs ~ 0.53bit/sec/Hz) while reducing the complexity of a factor of about 50%.

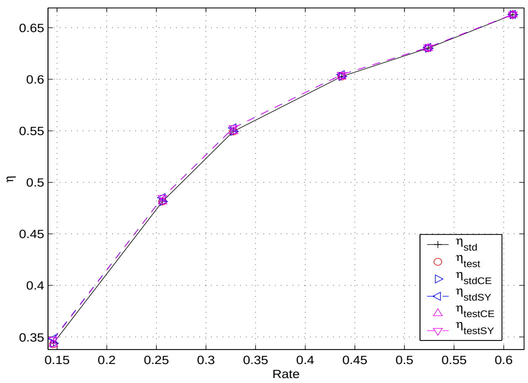

Figure 1: Throughput (η_{std}, η_{stdCE}, η_{stdSY}, η_{test}, η_{testCE} and η_{testSY}) for $\gamma = 3$dB.

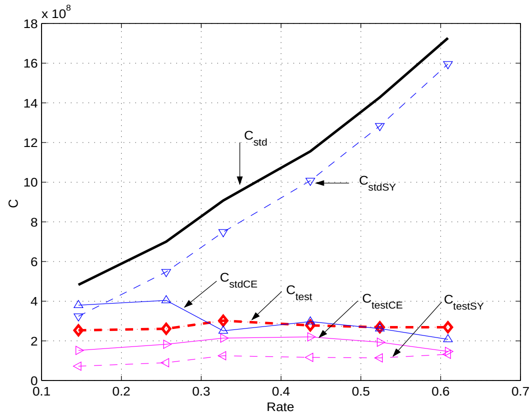

Figure 2: Complexity (C_{std}, C_{test}, C_{stdCE}, C_{stdSY}, C_{testCE} and C_{testSY}) for $\gamma = 3$dB.

4. CONCLUSIONS

In this paper we have shown easy to implement methods to speed up the iterative decoding technique. The principal method consists on introducing a test based on Density Evolution prior to decode that prevent using the iterative decoder if it is likely to be non convergent. We have shown that the proposed algorithm reduces considerably the average complexity without degrading the performances, and modification of the same algorithm allows to achieve all range of trade offs between throughput and complexity.

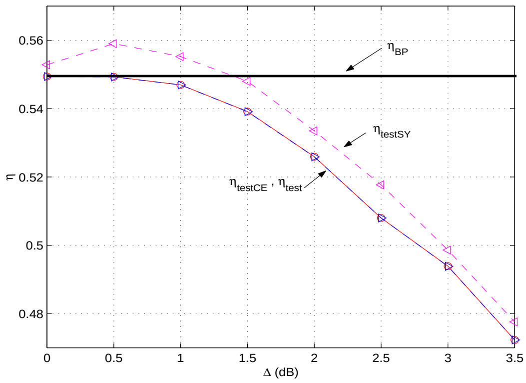

Figure 3: Throughput of the modified DE-test method vs Δ (dB) for $R = 0.3$bit/symbol and $\gamma = 3$dB.

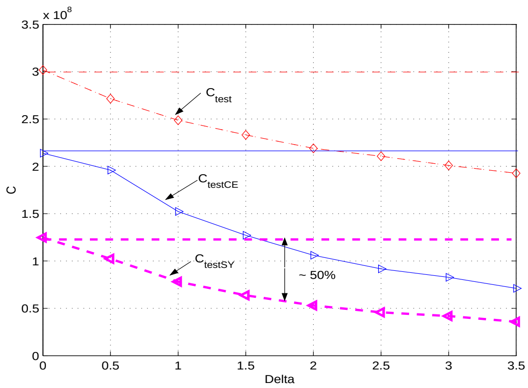

Figure 4: Complexity of the modified DE-test method vs Δ (dB) for $R = 0.3$bit/symbol and $\gamma = 3$dB.

REFERENCES

[1] A. Banerjee, D. J. Costello, and T. E. Fuja, "Comparison of Different Retransmission Strategies for Bandwidth Efficient Hybrid ARQ Scheme Using Turbo Codes". *IEEE Trans. on Information Theory*, pp. 1–10, July 2001.

[2] E. Biglieri, J. Proakis, and S. Shamai, " Fading channels: information-theoretic and communications aspects". *IEEE Trans. on Information Theory*, Vol. 44, pp. 2619–2692, October 2001.

[3] T. J. Richardson and R. L Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding". *IEEE Trans. on Information Theory*, Vol. 47, pp. 599–618, February 2001.

[4] S. Sesia, G. Caire, and G. Vivier, "Incremental Redundancy Hybrid ARQ Schemes Based on Low-Density Parity-Check Codes". *submitted to IEEE Trans. on Communication*, February 2003.