Nonbinary decoding of structured LDPC codes

Daniele Capirone, Giacomo Como, Fabio Fagnani and Federica Garin
Politecnico di Torino, C.so Duca degli Abruzzi 24, 10129 Torino, Italy
Email: daniele.capirone@polito.it, giacomo.como@polito.it, fabio.fagnani@polito.it, federica.garin@polito.it

***Abstract*—A class of serial turbo codes admitting low-density parity-check (LDPC) representation is considered. Their parity matrix has a random and a structured part. Thanks to their turbo structure, these codes are linear-time encodable, while they can be decoded as LDPC codes. An ensemble analysis for the error floor region, on the line of classical results for serial turbo codes, suggests a design parameter. Simulation results using usual LDPC message-passing show poor performance and no dependence on such parameter. A different block-wise decoding algorithm is proposed, which considerably improves performance. With this new decoding scheme, simulations confirm the theoretical design parameters in the medium-high SNR region.**

I. Introduction

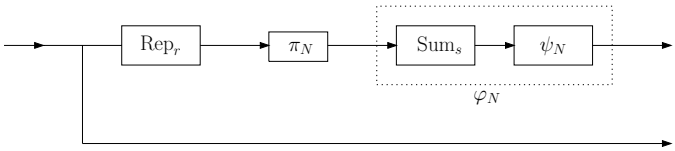
The encoding complexity of LDPC codes is generally quadratic in the block length, as the generating matrix is not low density. This, despite their linear decoding complexity using iterative belief propagation (BP), due to the sparsity of their Tanner graph. This issue has been addressed in the literature following two different approaches. On the one hand, there are the results in [8], which allow to construct, for given generic LDPC matrices, equivalent generating matrices with lower encoding complexity. On the other hand, constraining the parity check matrix to have a particular structure can a priori guarantee easy encoding. A successful construction uses matrices with a staircase part (i.e. a sub-matrix with ones on the diagonal and on the lower diagonal, and zeros everywhere else), so that the encoder can be seen as the serial concatenation of a repetition code, an interleaver and an accumulator. They are called Repeat-Accumulate (RA) [5] codes or, if repetition is not uniform, Irregular Repeat-Accumulate (IRA) codes [4].

We follow this second approach, studying LDPC codes encodable with a serial turbo structure. There is a wide literature on the analysis and design of IRA codes (we refer to [9] and references therein). Previous works focused on the design of the degree distribution of variable nodes (the time-varying number of repetitions) and of check nodes (grouping factor). On the contrary, in [2] the possibility to vary the structured part of the matrix has been investigated, focusing on the simpler case when the degrees are constant. This is equivalent to choosing an inner encoder different from the accumulator.

Theoretical results were presented in [2] enlightening how the inner encoder affects performance under maximum-likelihood (ML) decoding. Motivated by the poor behaviour of standard BP decoding observed in simulations, in this paper we propose a message-passing algorithm for these codes based on non-binary BP. This algorithm works on a factor graph without structured short cycles. On the contrary the standard Tanner graph contains plenty of small cycles. MonteCarlo simulations show good performance of this algorithm and hierarchies indicated by the ML analysis are respected.

II. Encoding schemes and parity check matrices

Consider the family of serially concatenated turbo encoders with the following structure:



By $\mathrm{Rep}_r : \mathbb{Z}_2^N \to \mathbb{Z}_2^{rN}$ we denote the repetition code with rate $1/r$; $\mathrm{Sum}_s : \mathbb{Z}_2^{rN} \to \mathbb{Z}_2^{rN/s}$ is defined by

$$\mathrm{Sum}_s(\boldsymbol{x}) = (x_1 + \ldots + x_s,\, x_{s+1} + \ldots + x_{2s},\, \ldots)$$

i.e. it gives the modulo-2 sum of every block of s bits. Finally, let $\psi(D) : \mathbb{Z}_2^k((D)) \to \mathbb{Z}_2^k((D))$ be a rate-1 non-catastrophic and recursive convolutional encoder, and $\psi_N : \mathbb{Z}_2^{rN/s} \to \mathbb{Z}_2^{rN/s}$ be the truncated encoder obtained by using the trellis of $\psi(D)$ for $rN/(sk)$ time steps. Define the rate $R = (1 + \frac{r}{s})^{-1}$ systematic encoder

$$\Phi_N : \mathbb{Z}_2^N \to \mathbb{Z}_2^{(1+\frac{r}{s})N}, \qquad \Phi_N \boldsymbol{u} = (\boldsymbol{u}, \mathrm{Sum}_s \circ \pi_N \circ \mathrm{Rep}_r\, \boldsymbol{u})\,.$$

We will always assume that rN is a multiple of sk, so that the above construction can be properly made. Notice that $\psi(D)$ can be seen as a $k \times k$ matrix whose entries are fractions of polynomials: $\psi(D)$ is non-catastrophic if and only if this matrix has an inverse whose entries are Laurent polynomials. Recursiveness of $\psi(D)$ is equivalent to the recursiveness of at least one entry in each column of the matrix. In particular, if $k = 1$, our assumptions imply that $\psi(D) = 1/p(D)$ for some polynomial $p(D)$.

Φ_N is a particular kind of systematic serial turbo encoder where the outer encoder is Rep_r and the inner encoder is $\varphi_N = \psi_N \circ \mathrm{Sum}_s$. φ_N can be considered as the truncation of a proper convolutional encoder, which is not injective, but the transmission of the systematic bits ensures injectivity and non-catastrophicity of Φ_N. Also notice that φ_N is recursive, in the sense that inputs of weight one produce outputs with weight growing to infinity when $N \to \infty$; this is essential for results about the interleaver gain.

The representation as serial turbo codes allows linear-time encoding, proportional to kN. The decoding can be performed exploiting the natural LDPC representation of these codes.

978-1-4244-1682-0/08/$25.00 ©2008 IEEE

Authorized licensed use limited to: MIT Libraries. Downloaded on January 5, 2009 at 17:08 from IEEE Xplore. Restrictions apply.

Indeed, notice that a pair $(\boldsymbol{u}, \boldsymbol{c})$ in $\mathbb{Z}_2^N \times \mathbb{Z}_2^{rN/s}$ is in the image of Φ_N if and only if $\boldsymbol{c} = \psi_N \circ \mathrm{Sum}_s \circ \pi_N \circ \mathrm{Rep}_r(\boldsymbol{u})$. This is equivalent to $\mathrm{Sum}_s \circ \pi_N \circ \mathrm{Rep}_r(\boldsymbol{u}) + \psi_N^{-1}(\boldsymbol{c}) = \boldsymbol{0}$ and can be represented in the matrix form $[H_N\, K_N]\left[\begin{smallmatrix}\boldsymbol{u}\\ \boldsymbol{c}\end{smallmatrix}\right] = \boldsymbol{0}$. Here H_N is a $\frac{r}{s}N \times N$ matrix depending on the permutation π_N only. It is sparse, having at most s ones per row and r ones per column. K_N is a $\frac{r}{s}N \times \frac{r}{s}N$ matrix depending on the choice of ψ only. It is also low density, having a number of ones per row and per column bounded by $k(\deg \psi^{-1}(D) + 1)$.

Example 1: The RA code fills in this framework by considering $k = 1$, $s = 1$ and $\psi(D) = 1/(1 + D)$. □

In this paper we will focus on inner convolutional encoders of type $\psi(D) = (A + BD)^{-1}$ with $A, B \in \mathbb{Z}_2^{k\times k}$ invertible matrices. This leads to a structured part of the parity matrix having the staircase form:

$$K_N = \begin{bmatrix} A & 0 & 0 & \dots & 0 \\ B & A & 0 & \dots & 0 \\ 0 & B & A & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & B & A \end{bmatrix} \tag{1}$$

Such matrices are good candidates for nonbinary BP decoding, because of their block-wise staircase structure. At the same time, thanks also to invertibility of A and B, they all have a minimal realization (trellis) with 2^k states, which is very easy to compute: given the current state $x_t \in \mathbb{Z}_2^k$ and the input $u \in \mathbb{Z}_2^k$, the new state x_{t+1} and the output $y_t \in \mathbb{Z}_2^k$ are the same, obtained as $x_{t+1} = y_t = A^{-1}u_t + A^{-1}Bx_t$ (invertibility of B ensures that this realization is minimal). From this realization, it is also clear that the corresponding encoder $\psi(D) = (A + BD)^{-1}$ is causal and recursive; non-catastrophicity immediately follows from the fact that $\psi^{-1}(D)$ is polynomial. Notice that every scalar convolutional encoder of the form $\psi(D) = 1/p(D)$ can be represented in this form with $k = \deg(p)$.

Example 2: The scalar encoder $1/(1 + D + D^3)$ can be represented by $\psi(D) = (A + BD)^{-1}$ with

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

□

III. Analytical results for ML decoding

Following a classical analysis of serial turbo codes [1], [5], it is possible to obtain analytical estimations of the ML error probability. In particular, it is possible to show that, like in classical serial interconnections, there is a natural distance parameter, that is responsible of the performance behavior in the range of high signal-to-noise ratio and sufficiently large blocklength. This analysis can be found in [2]. Differently from the classical case, it can not simply be obtained from the analysis of the 'uniform interleaver' ensemble, in this case it is necessary to work on a suitable expurgated ensemble. In the sequel of this section we present the precise results, whose proofs can be found in [2], [3].

We consider an ensemble where the convolutional encoder ψ is fixed, as well as the repetition coefficient r and the grouping factor s. For every N the interleaver Π_N is a random variable uniformly distributed over a subset $R_{r,s}^N$ of the set S_{rN} of all permutations of rN elements. In particular $R_{r,s}^N$ is chosen as the family of interleavers guaranteeing that ones coming from the same error event of Rep_r cannot end up in positions where they would be summed up by Sum_s. More precisely, define the set

$$R_{r,s}^N := \left\{\pi \in S_{rN} \colon \lfloor i/r \rfloor = \lfloor j/r \rfloor \;\Rightarrow\; \lfloor \pi(i)/s \rfloor \neq \lfloor \pi(j)/s \rfloor\right\}$$

Notice that restricting the permutation to $R_{r,s}^N$ is equivalent to enforcing the associated Tanner graph not to have 2-cycles. Also it may be observed that sampling from this ensemble is equivalent to picking H_N uniformly at random from the set of $N \times N$ binary matrices with exactly s ones per row and r ones per column. Finally notice that, asymptotically in N, $R_{r,s}^N$ is neither typical nor vanishing in S_{rN}. Indeed, the ratio $\left|R_{r,s}^N\right| / |S_{rN}|$ converges to $e^{-(r-1)(s-1)/2}$ as N grows.

Our results concern the asymptotic behaviour of the average word error probability of Φ_N, denoted by $\overline{P_w(e)}$, over a binary-input output-symmetric channel having Batthacharyya parameter γ and equivocation probability p. They have been obtained using expurgation techniques. Let $s \geq 2$, $r \geq 2$. Define $\mu := \lfloor \frac{r+1}{2} \rfloor$ and

$$d^* := \begin{cases} 2 & r = 2,\, 3 \\ 1 + \frac{r}{2} d_2^\psi & \text{even } r \geq 4 \\ 1 + \frac{r-3}{2} d_2^\psi + \min\left\{ d_2^\psi + d_{1,\mathrm{tr}}^\psi,\, d_3^\psi \right\} & \text{odd } r \geq 5\,, \end{cases}$$

where $d_{1,\mathrm{tr}}^\psi$ is the smallest weight of a truncated error event of ψ having an input weight 1 (if $k = 1$, then $d_{1,\mathrm{tr}} = 1$), while d_2^ψ and d_3^ψ are the smallest weight of an error event of $\psi(D)$ having input weight 2 and 3 respectively.

Theorem 1: There exist positive constants γ_0, c_1, c_2 (depending on r, s, k only) and c_3 (depending also on γ) such that, for all $\gamma \leq \gamma_0$:

$$c_1 p^{d^*} N^{-\mu+1} \leq \overline{P_w(e)} \leq c_2 \gamma^{d^*} N^{-\mu+1} + c_3 N^{-\mu}$$

□

Observe that the exponent μ depends on the repetition factor r only. On the contrary, the parameter d^* depends on the inner convolutional encoder ψ as well, in particular through d_2^ψ. As shown in [2], averaging over the set S_{rN} of all possible interleavers instead of R_N would have hidden such a dependence. Notice that $d_2^\psi = 1$ for the accumulator $\psi(D) = (1 + D)^{-1}$, while $\psi(D) = (1 + D + D^3)^{-1}$ has $d_2^\psi = 4$.

IV. Standard BP decoder and structured cycles

We first simulate the coding schemes based on the convolutional encoders of Examples 1 and 2 using the standard belief propagation algorithm over the Tanner graph associated to the low density matrix $[H_N K_N]$. While this approach is satisfactory for $\psi(D) = 1/(1 + D)$, this is not the case for the encoder $1/(1 + D + D^3)$ of Ex. 2. In fact, Monte-Carlo simulations reported in Fig. 1 are in contrast with the results of Theorem 1. The coding scheme based on the simple

Authorized licensed use limited to: MIT Libraries. Downloaded on January 5, 2009 at 17:08 from IEEE Xplore. Restrictions apply.

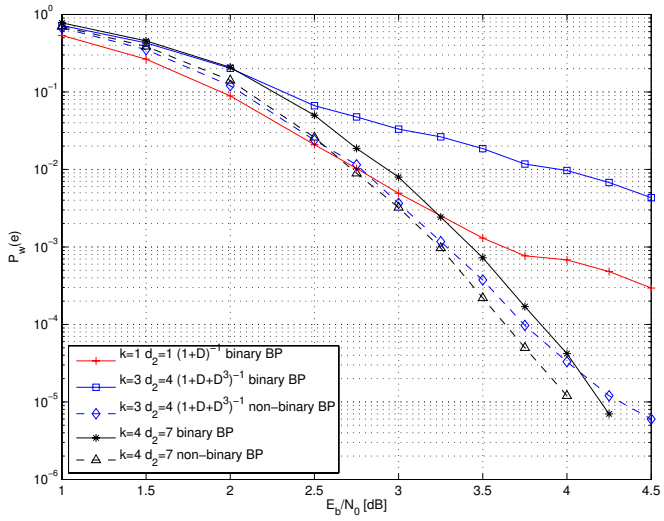


Fig. 1. Comparison between binary and nonbinary BP decoding. ($\frac{N}{R} = 300$)

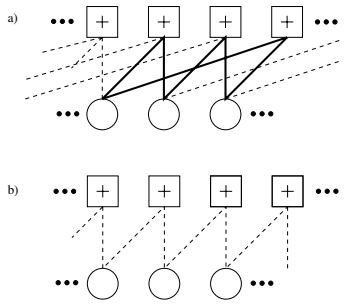


Fig. 2. Structured part of Tanner graph: a) $(1+D+D^3)^{-1}$, b) $(1+D)^{-1}$

accumulator $\psi(D) = 1/(1+D)$, having $d_2^\psi = 1$, performs much better than the one using $\psi(D) = 1/(1+D+D^3)$ as inner encoder, even if the latter has $d_2^\psi = 4$.

A close look to the structure of the Tanner graphs suggests a possible explanation for such a disappointing behaviour. Indeed, (see Fig. 2) the structured part of the graph contains a large number of 6-cycles. More precisely there are $N-2$ of such cycles and they are concatenated in a very particular way. The belief-propagation algorithm is known to be exact on cycle-free graphs [10] and has been shown to be highly performing on random graphs which with high probability do not contain small cycles [7]. Thus, the presence of such a big and structured collection of 6-cycles seems to be a possible explanation why the algorithm fails to converge. Notice that the Tanner graph of the Repeat-Accumulate does not contain any cycle in its structured part.

A possible way to avoid cycles in the structured part of the Tanner graph is constraining both A and B to be permutation matrices. However it is easy to show that this implies that $\psi(D) = (A+BD)^{-1}$ has $d_2^\psi = 1$. Also limiting only A to be a permutation matrix (and allowing B to be any invertible element of $\mathbb{Z}_2^{k\times k}$) can prevent from obtaining the best d_2^ψ

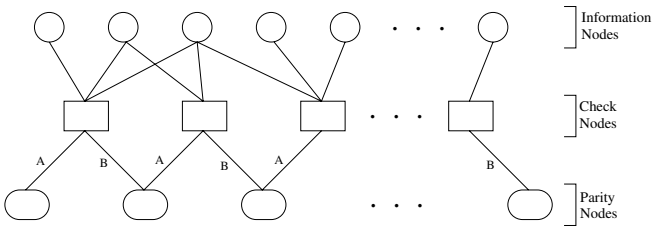


Fig. 3. Tanner Graph of the hybrid nonbinary algorithm

achievable with no restrictions on the choice of the invertible matrices A and B. For instance for $k=3$ the highest value of d_2^ψ which can be obtained with a permutation matrix A is 3, strictly smaller than the d_2^ψ of the encoder of Ex. 2. Similarly for $k=4$ and A a permutation we have $d_2^\psi \leq 7$, while for general invertible A also $d_2^\psi = 8$ exists.

V. NONBINARY BP DECODER

Motivated by the considerations of the previous section, we propose the following modified version of the BP algorithm for the case when $k > 1$. Associate to the parity matrix $[H_N K_N]$ a labeled factor graph with vertex set given by $\mathcal{V}_i \cup \mathcal{V}_p \cup \mathcal{V}_c$ (see Fig.3), where:

- $\mathcal{V}_i = \{i_1, \ldots, i_N\}$ is a set of N information nodes, each corresponding to an information bit (recall the codes are systematic);
- $\mathcal{V}_p = \{p_1, \ldots, p_{\frac{r}{ks}N}\}$ is a set of $\frac{r}{ks}N$ parity nodes, each corresponding to a group of k consecutive parity bits;
- $\mathcal{V}_c = \{c_1, \ldots, c_{\frac{r}{ks}N}\}$ is a set of $\frac{r}{ks}N$ check nodes each corresponding to a group of k consecutive rows of the matrix.

For every $1 \leq j \leq \frac{r}{ks}N$, the parity node p_j is connected only to the check node c_j with an edge labeled by $\lambda_{i_j,c_j} = A$, and to the check node c_{j+1} with an edge labeled by $\lambda_{i_j,c_{j+1}} = B$. There is an edge between a check node c_l in $\mathcal{V}_c$ and an information node i_j in $\mathcal{V}_i$ whenever the $k \times 1$ block $(H_N)_{[k(l-1)+1,kl],j}$ is nonzero; such an edge is labeled by the $k \times 1$ block $\lambda_{c_l,p_j} = (H_N)_{[k(l-1)+1,kl],j}$ itself.

We use a sum-product belief propagation algorithm over this graph. Messages exchanged between information nodes and check nodes consist in probability distributions over $\mathbb{Z}_2$, while messages exchanged between parity nodes and check nodes consist in probability distributions over $\mathbb{Z}_2^k$. For every parity or information node v denote the a posteriori probability distributions given by the channel output by $\boldsymbol{z}_v$. Denote the message sent from node v to node v' at the t-th iteration by $\boldsymbol{m}^t_{v\to v'}$. For every adjacent parity node v and check node c initialize $\boldsymbol{m}^0_{c\to v}$ as the uniform distribution over $\mathbb{Z}_2^k$ and similarly for every adjacent information node v and check node c let $\boldsymbol{m}^0_{c\to v}$ be the uniform distribution over $\mathbb{Z}_2$. Then for every time step $t \geq 1$

- the message sent from a node v in $\mathcal{V}_i \cup \mathcal{V}_p$ to an adjacent check node c, $\boldsymbol{m}^t_{v\to c}$ is the normalized pointwise product of $\boldsymbol{z}_v$ and of messages $\boldsymbol{m}^{t-1}_{c'\to v}$ received by the node v from all its neighbors c' but c;

Authorized licensed use limited to: MIT Libraries. Downloaded on January 5, 2009 at 17:08 from IEEE Xplore. Restrictions apply.

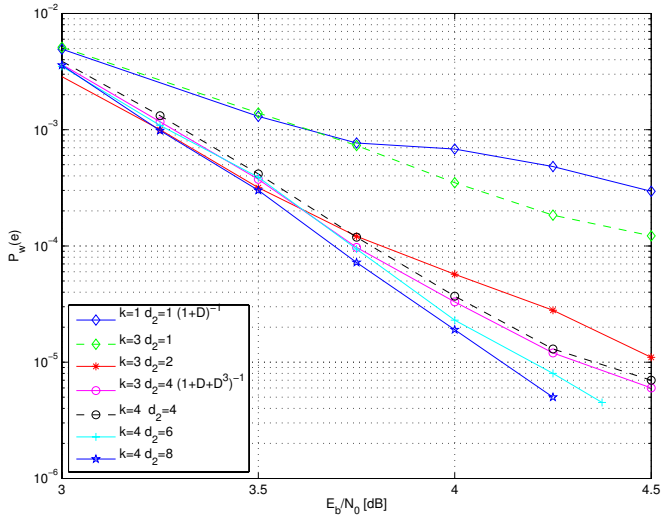


Fig. 4. Dependency on d_2^ψ for different values of k, block length $= 300$

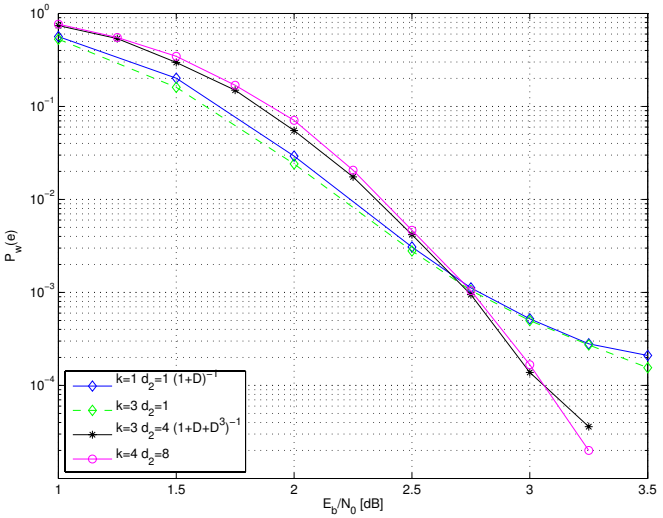


Fig. 5. Dependency on d_2^ψ for different values of k, block length $= 600$

- the message sent from a check node c to an adjacent information or parity node v is given by
$$\boldsymbol{m}_{c\to v}^t(x) = \mathbb{P}_{c\to v}^t\big(\textstyle\sum_{v'\sim c,\, v'\neq v} \lambda_{c,v'} X_{v'} = \lambda_{cv} x\big)$$
where the probability $\mathbb{P}_{c\to v}^t$ is evaluated by considering the random variables $X_{v'}$ mutually independent, each distributed accordingly to $\boldsymbol{m}_{v'\to c}$.

Notice that the complexity of this nonbinary BP algorithm (with an efficient implementation of the check-nodes updates), scales with k and N as $2^{2k+1}\frac{r}{ks}N$ operations per iteration, compared to $\frac{4r(k+1)}{s}N$ for the standard BP algorithm.

VI. Simulation results

All the examples we simulated have $r = 4$ and $s = 4$, consequently the overall rate R is $1/2$. A maximum of 50 iterations has been considered.

Fig. 1 shows how the use of the non-binary algorithm leads to an improvement with respect to the standard BP algorithm. This improvement is dramatic in some cases, such as Ex. 2, while it is still significant in most codes we have simulated, such as the curve we plot, corresponding to an encoder with $k = 4$ and $d_2^\psi = 7$.

Figures 4 and 5 show the role of d_2^ψ, comparing different encoders all decoded with the non-binary algorithm. The hierarchy given by this parameter is clearly respected in the error-floor region, as predicted by the theoretical results. At low SNR, we see that the hierarchy is inverted (see Fig. 5), so that we have cross points among curves. The codes in Figures 4 and 5 have blocklength 300 and 600 respectively; at higher lengths it is more difficult to get simulation results in the error floor region, which has very low $P_w(e)$.

VII. Conclusion

We analyzed a class of linear-time encodable LDPC codes with serial turbo structure, focusing on the optimization of the inner convolutional encoder. We propose a nonbinary BP decoding algorithm working on a factor graph which does not contain cycles in its structured part. Monte Carlo simulations show that this algorithm can significantly outperform the standard BP. Moreover, simulations with this new algorithm perfectly match the theoretical analysis under ML decoding [2], and confirm, for the error floor region, a hierarchy based on the effective free distance of the inner encoder (d_2^ψ).

Because of the absence of structured cycles in the factor graph, the nonbinary BP algorithm we presented is particularly appealing for a density evolution analysis combining the ideas in [9] and [6]. In this context, a natural further generalization is to study also irregular codes, where the repetition r and the grouping factor s are time-varying. We leave these as topics for future research.

References

[1] S. Benedetto, D. Divsalar, G. Montorsi and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design and iterative decoding", *IEEE Trans. on Inf. Th.*, vol. 44, pp. 909–926, May 1998.

[2] F. Garin, G. Como and F. Fagnani, "Staircase and other structured linear-time encodable LDPC codes: analysis and design", *Proc. ISIT 2007*.

[3] F. Garin and F. Fagnani, "Analysis of serial turbo codes over Abelian groups for Geometrically Uniform constellations", submitted (2007), `http://calvino.polito.it/rapporti/2007/pdf/20_2007/art_20_2007.pdf`

[4] H. Jin, A. Khandekar and R. J. McEliece, "Irregular Repeat-Accumulate Codes", *Proc. of the 2nd International Symposium on Turbo Codes and Related Topics*, Brest, France, Sept. 2000.

[5] H. Jin and R. J. McEliece, "Coding theorems for turbo code ensembles", *IEEE Trans. on Inform. Theory*, vol. 48 (6), pp. 1451–1461, June 2002.

[6] V. Rathi and R. Urbanke, " Density evolution, thresholds and the stability condition for non-binary LDPC codes", IEE Proc. on Communications, vol. 152 (6) , pp. 1069-1074, Dec. 2005.

[7] T.J. Richardson, R. Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding", *IEEE Trans. Inform. Theory*, vol. 47 (2), pp. 599-618, Feb. 2001.

[8] T. Richardson and R. Urbanke, "Efficient Encoding of Low-Density Parity-Check Codes", *IEEE Trans. on Inform. Theory*, vol. 47 (2), pp. 638–656, Feb. 2001.

[9] A. Roumy, S. Guemghar, G. Caire and S. Verdú, "Design Methods for Irregular Repeat-Accumulate Codes", *IEEE Trans. on Inform. Theory*, vol. 50 (8), Aug. 2005.

[10] N. Wiberg, Codes and Decoding on General Graphs, PhD thesis, Linkoping University, 1995. `http://citeseer.ist.psu.edu/wiberg96codes.html`

Authorized licensed use limited to: MIT Libraries. Downloaded on January 5, 2009 at 17:08 from IEEE Xplore. Restrictions apply.