Rate-Compatible LDPC Codes using Optimized Dummy Bit Insertion

Moritz Beermann, Tobias Breddermann, Peter Vary
Inst. of Comm. Systems and Data Processing (ind)
RWTH Aachen University, Germany
{beermann|breddermann|vary}@ind.rwth-aachen.de

***Abstract*—Much attention has been paid to Low-Density Parity-Check (LDPC) codes since their rediscovery by MacKay. They belong to the most powerful channel coding techniques known today and have a broad range of applications. In wireless communication systems it is desirable to be able to adjust the code rate of the employed channel coding scheme (rate-matching) to allow for a flexible strength of error protection for different services and to be able to adapt to the varying quality of the wireless transmission channel. Many of the current systems that employ LDPC codes like, e.g., WiMAX or WLAN specify separate codes for each supported code rate. This paper, in contrast, addresses the problem of using only one mother code and matching (almost) arbitrary code rates that are lower than the mother code rate by inserting known (dummy) bits into the information bit sequence before encoding (also known as pruning or code shortening). We present a novel rule of determining (heuristically) optimized positions of dummy bits within the information bit sequence suitable for LDPC codes. Simulation results show that the frame error rate performance can be improved by the novel approach of dummy bit insertion especially in the error floor region.**

I. Introduction

Modern wireless communication systems support a variety of applications and are designed to operate over a wide range of transmission channel qualities. The demanded data rate and strength of error protection strongly depend on the services, like, e.g., speech, video, or data transmission. At the same time the data rate that can be delivered to a user can be limited by many factors, e.g., the number of active users and the user's current channel quality. For such a system a so called rate-matching procedure is indispensable in order to support all these requirements and be able to adapt to varying conditions.

Since most channel codes have a fixed code rate, the purpose of rate-matching is to allow for flexible adjustment of this rate which can be realized in different ways. The UMTS LTE system [1], e.g., uses a rate-$1/3$ turbo code [2] and applies bit puncturing to achieve code rates higher than $1/3$ and bit repetition to achieve code rates lower than $1/3$. In other standards like IEEE 802.16e (WiMAX) [3] or IEEE 802.11n (WLAN) [4], separate Low-Density Parity-Check (LDPC) codes [5], [6] are specified for each supported code rate.

Many techniques have been proposed to construct codes of higher or lower code rate from a fixed-rate mother LDPC code. In [7] Li et al. studied parity bit puncturing to construct LDPC codes of higher code rates and proposed a special code extension to achieve lower code rates. Another approach of constructing codes of lower code rate from a high rate mother code was presented in [8] using information shortening (which has been termed dummy bit insertion in [9]). In this paper we concentrate on rate-matching by dummy bit insertion and propose a novel rule to select heuristically optimized positions to insert known dummy bits into the information bit sequence.

Note that in the literature many terms have been used to describe the concept of inserting known bits before encoding, including pruning (e.g., [10]) and code or information shortening (e.g., [8]). Throughout this work, however, we use the term dummy bit insertion as introduced in [9].

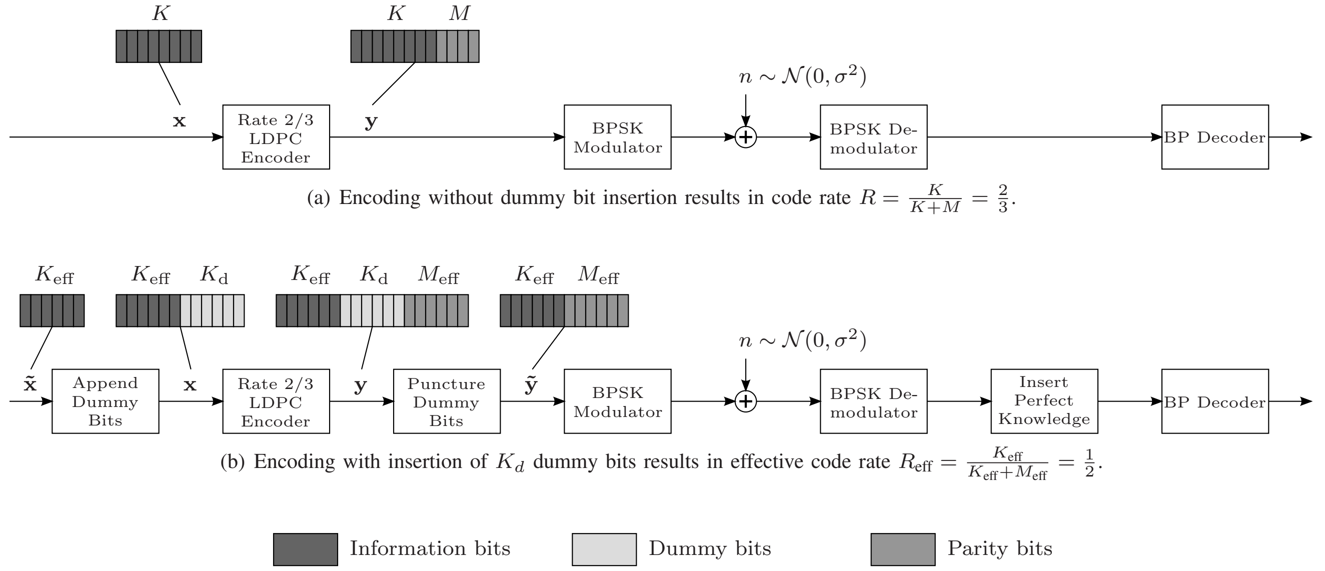
II. Low-Density Parity-Check Codes

A binary (N, K) LDPC code is a linear block code defined by a sparse parity check matrix $\mathbf{H}$ of dimension $M \times N$, with K denoting the number of information bits, N the total number of coded bits and $M = N - K$ the number of parity bits. In the following, we assume $\mathbf{H}$ to have full rank $\text{rank}(\mathbf{H}) = M$. The code rate is then given by $R = \frac{K}{N} = \frac{N-M}{N}$. H_{mn} denotes the entry of $\mathbf{H}$ at row m and column n. The set $\mathfrak{N}(m) = \{n : H_{mn} \neq 0\}$ contains all bits that participate in parity check equation m, $m = 1, \ldots, M$. Similarly, $\mathfrak{M}(n) = \{m : H_{mn} \neq 0\}$ denotes the set of all check equations in which bit n participates.

An LDPC code is commonly described by a bipartite graph consisting of N variable nodes and M check nodes (known as factor or Tanner graph [11]). Each variable node corresponds to a code bit and each check node corresponds to a parity check equation as defined by a row of the parity check matrix. Those variable nodes corresponding to information bits are denoted information nodes, while variable nodes corresponding to parity bits are called parity nodes. Variable node n is connected to all check nodes in the set $\mathfrak{M}(n)$ and check node m is connected to all variable nodes in the set $\mathfrak{N}(m)$.

For the decoding of LDPC codes, we apply the iterative Belief Propagation algorithm as described in [12]. After initialization of each variable-to-check message v_{mn} (i.e., the message sent from variable node n to check node m) with the received channel-related L-value Z_n [13] for the nth bit, the so called horizontal and vertical step are computed alternately. The horizontal step (or check node update) computes the check-to-variable messages c_{mn} (i.e., the message sent from

This work was supported by the research cluster UMIC located at RWTH Aachen University, Germany, funded by the German research council DFG.



(a) Encoding without dummy bit insertion results in code rate $R = \frac{K}{K+M} = \frac{2}{3}$.

(b) Encoding with insertion of K_d dummy bits results in effective code rate $R_{\text{eff}} = \frac{K_{\text{eff}}}{K_{\text{eff}}+M_{\text{eff}}} = \frac{1}{2}$.

Fig. 1. Exemplary transmission system (a) with and (b) without dummy bit insertion using a rate $2/3$ LDPC Encoder, BPSK modulation, an AWGN channel and Belief Propagation Decoding.

check node m to variable node n) as

$$c_{mn} = 2 \tanh^{-1} \left(\prod_{n' \in \mathfrak{N}(m) \setminus n} \tanh \left(\frac{v_{mn'}}{2} \right) \right), \tag{1}$$

where the operator "\" denotes exclusion of an element from a set. Likewise, the vertical step updates all variable-to-check messages according to

$$v_{mn} = Z_n + \sum_{m' \in \mathfrak{M}(n) \setminus m} c_{m'n} \,. \tag{2}$$

Additionally, after each vertical step, the hard decision

$$\hat{y}_n = \operatorname{sign} \left\{ Z_n + \sum_{m' \in \mathfrak{M}(n)} c_{m'n} \right\} \tag{3}$$

is computed for each variable node n to evaluate the parity check equations. Decoding is stopped if either all parity check equations are fulfilled or a maximum number of iterations is reached.

III. Rate-Matching by Dummy Bit Insertion

The concept of achieving lower code rates using a mother code of fixed rate and dummy bit insertion (i.e., code shortening) is depicted in Fig. 1 for an exemplary system using a rate $R = 2/3$ LDPC code, Binary Phase Shift Keying (BPSK) modulation, an Additive White Gaussian Noise (AWGN) channel, and Belief Propagation (BP) decoding.

In case (a) without dummy bit insertion the rate $R = 2/3$ code takes K information bits as input and generates $M = \frac{1-R}{R} K$ additional parity bits at the output. In case (b) K_d known dummy bits are appended to the K_{eff} information bits before encoding. The encoder, thus, generates $M_{\text{eff}} = \frac{1-R}{R}(K_{\text{eff}} + K_d)$ parity bits. Before passing the resulting bit stream to the modulator, the dummy bits are removed (punctured) since they are known at the receiver and therefore do not need to be transmitted. The resulting effective code rate R_{eff} amounts to

$$\begin{aligned} R_{\text{eff}} &= \frac{K_{\text{eff}}}{K_{\text{eff}} + M_{\text{eff}}} \\ &= \frac{K_{\text{eff}}}{K_{\text{eff}} + \frac{1-R}{R}(K_{\text{eff}} + K_d)} \\ &= \frac{R}{R + (1-R)(1 + \frac{K_d}{K_{\text{eff}}})} \\ &= \frac{R}{1 + (1-R)\frac{K_d}{K_{\text{eff}}}} \,. \end{aligned}$$

It can easily be seen that any code rate $0 < R_{\text{eff}} \leq R$ can be achieved by an accordant choice of the fraction $\frac{K_d}{K_{\text{eff}}}$ (of course only in quantized steps depending on the absolute values of K_d and K_{eff}). If no dummy bits are used $R_{\text{eff}} = R$ holds and if the number of information bits approaches zero the effective code rate also approaches zero.

At the receiver the dummy bits are known and, thus, provide perfect *a priori* information that is inserted (after demodulation) into the received sequence of L-values in form of values $\pm\infty$ (depending on whether the dummy bits have been set to 0 or 1). Subsequently, the decoding algorithm can be run as in the case without dummy bit insertion.

A. Optimized Dummy Bit Insertion for Convolutional Codes

Rate-matching by dummy bit insertion for convolutional codes has been introduced by Xu and Romme in [9] as a technique for achieving lower code rates by inserting known bits into the information bit sequence before convolutional encoding. The perfect a priori information provided by the dummy bits highly supports the decoding of the adjacent information bits resulting in significantly improved decoding quality. However, only slight decoding gains were realized in conjunction with non-iterative convolutional decoding compared to standardized rate-matching schemes based on bit repetition [9]. The effect of dummy bit insertion on transmission systems employing iterative decoding has firstly been analyzed in [14] for a system based on *iterative source-channel decoding* (ISCD) [15], [16]. It has further been shown that best performance can be expected if dummy bits are not simply appended to the information bits but instead inserted equidistantly into the information bit sequence. This guarantees that the number of code bits comprising dummy bits as well as information bits is maximized, which significantly increases the decoding reliability of the corresponding information bits. The adaptation to a system based on convolutional turbo decoding [2] has been performed in [17] for the standardized UMTS LTE system [1]. The convergence speed as well as the decoding performance were improved significantly compared to the bit repetition scheme employed in UMTS LTE.

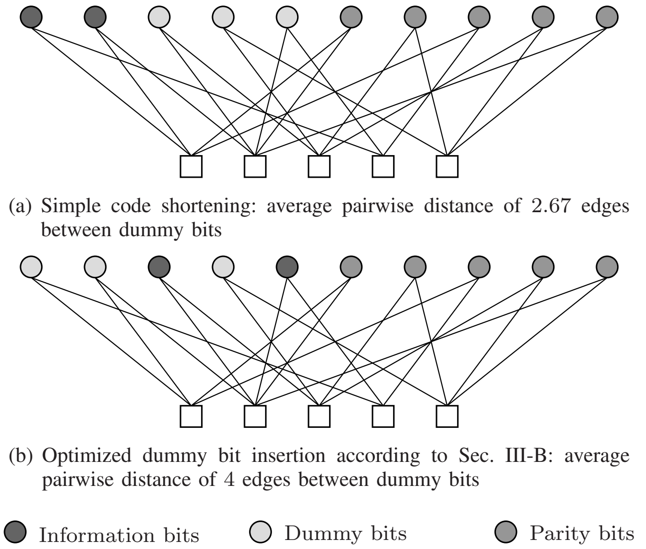
B. Optimized Dummy Bit Insertion for LDPC codes

We propose a novel rate-matching approach for LDPC codes that determines optimized dummy bit positions within the information bit sequence instead of simply appending the dummy bits to the information bits. Using dummy bit positions according to optimized degree distributions has already been proposed, e.g., in [8], [18]. In [19] Liu et al. presented an algorithm that selects dummy bit positions according to the variable node degrees and the so called *extrinsic-sum*. Our approach follows the argumentation of [17], where dummy bits are inserted equidistantly into the information bit vector before convolutional encoding. While convolutional codes introduce strong dependencies between neighboring bits, the relation between code bits of an LDPC code is not determined by the positions within a frame but solely by the connections of nodes in the Tanner graph. Thus, distributing variable nodes with perfect knowledge (corresponding to the known dummy bits at the receiver) throughout the Tanner graph as uniformly as possible is not as trivial as for convolutional codes.

We propose the following greedy heuristic to optimize the average minimum pairwise distance (i.e., the length of the shortest path) between a given number K_d of variable nodes in a Tanner graph. First, a simple breadth first search is applied to find the minimum pairwise distances between all K information nodes. The result is given by the $K \times K$ distance matrix $\mathbf{D}$, where the matrix entry D_{ij} holds the minimum distance between information node i and information node j. Note that the distance matrix $\mathbf{D}$ is symmetric and has zeros on the main diagonal, since the Tanner graph is undirected and thus, only $\frac{K(K-1)}{2}$ elements (either the ones above or the ones below the main diagonal) have to be stored in memory.

After the distance matrix has been determined, K_d information nodes are selected and added to the set of dummy nodes $\mathcal{V}$ in a node-by-node manner. The first two nodes that are added to $\mathcal{V}$ are chosen as $i_{\max}$ and $j_{\max}$ with

$$(i_{\max}, j_{\max}) = \underset{\forall i,j}{\arg\max}\ D_{ij}, \tag{4}$$


(a) Simple code shortening: average pairwise distance of 2.67 edges between dummy bits

(b) Optimized dummy bit insertion according to Sec. III-B: average pairwise distance of 4 edges between dummy bits

Fig. 2. Exemplary Tanner graph with five parity nodes and five information nodes, three of which are used for dummy bits. In case (a) simple code shortening is applied while case (b) uses the algorithm proposed in Sec. III-B to select dummy bit positions.

i.e., the two most distant information nodes. From the third node on, the node to be selected next is the information node i^* that maximizes the average distance to all information nodes j that have already been selected:

$$i^* = \underset{\forall i \notin \mathcal{V}}{\arg\max}\ \frac{1}{|\mathcal{V}|} \sum_{\forall j \in \mathcal{V}} D_{ij}, \tag{5}$$

where $|\mathcal{V}|$ denotes the cardinality of $\mathcal{V}$. The algorithm stops when K_d nodes have been added to $\mathcal{V}$ (i.e., $|\mathcal{V}| = K_d$). A summary of the algorithm is given in Alg. 1.

For a rate-matching scenario, the number of dummy bits K'_d is typically not fixed but possibly varies from one frame to the next. However, in this case it is still sufficient to run the algorithm once using $K_d = K$ and keep track of the order in which the information nodes are added to $\mathcal{V}$, resulting in an ordered list of potential dummy bit positions (and of length K). For each requested code rate smaller then the original code rate, the corresponding number of dummy bit positions can then simply be obtained as the first K'_d entries of this list.

Algorithm 1 Find K_d Optimized Dummy Bit Positions

```
Compute distance matrix $\mathbf{D}$ holding the minimum pairwise distances between all $K$ information nodes
Find two most distant nodes $(i_{\max}, j_{\max}) = \underset{\forall i,j}{\arg\max}\ D_{ij}$
Initialize $\mathcal{V} = \{i_{\max}, j_{\max}\}$
while $|\mathcal{V}| < K_d$ do
    Find $i^* = \underset{\forall i \notin \mathcal{V}}{\arg\max}\ \frac{1}{|\mathcal{V}|} \sum_{\forall j \in \mathcal{V}} D_{ij}$
    $\mathcal{V} = \mathcal{V} \cup \{i^*\}$
end while
```

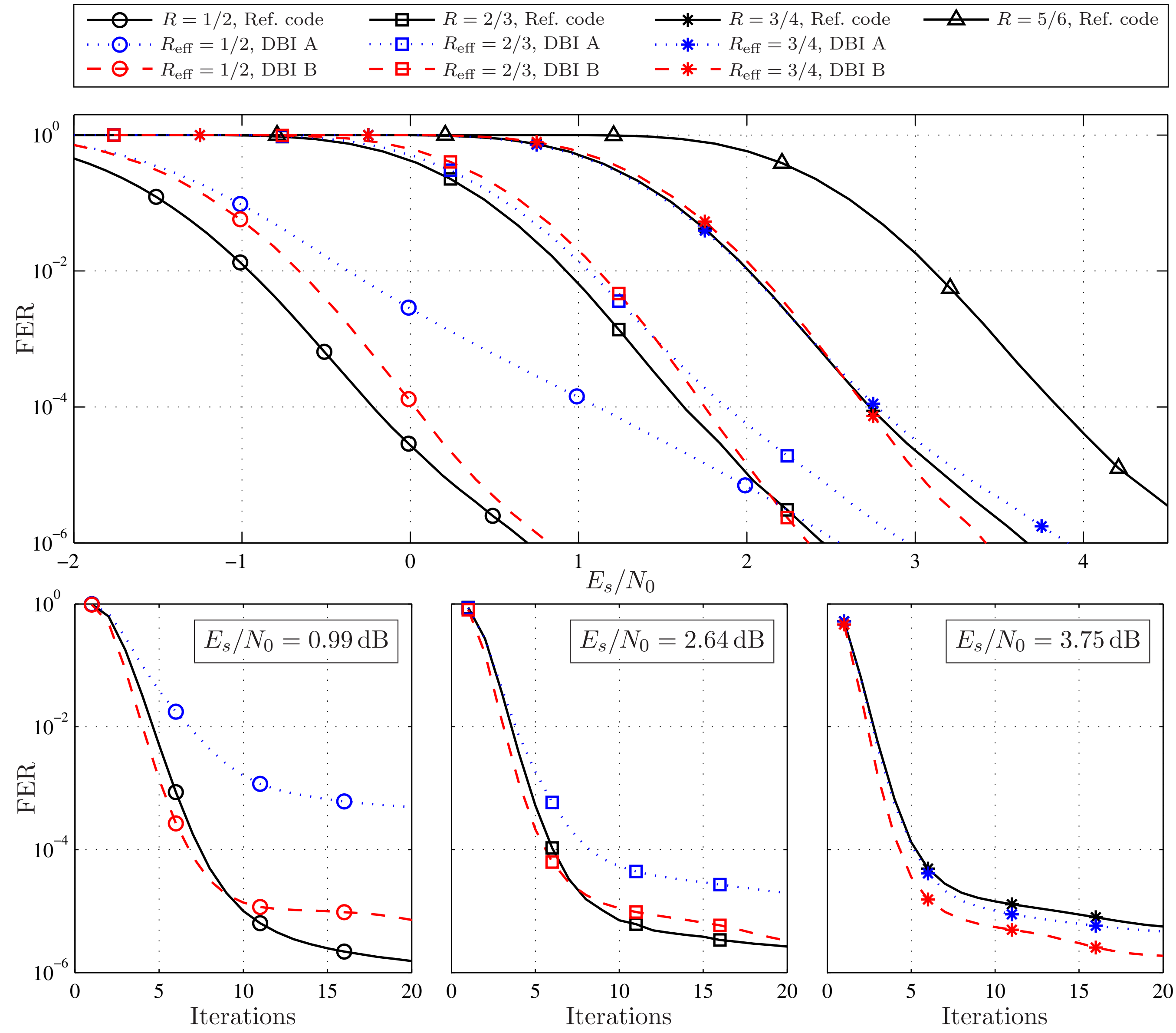


Fig. 3. Simulation results for the IEEE 802.16e LDPC codes of effective block length $N_{\text{eff}} = 576$. The upper plot shows the frame error rate (FER) over the channel quality in E_s/N_0 after 200 decoding iterations. The lower plot shows the FER over the number of decoding iterations at fixed E_s/N_0 values. Results are shown for the following three setups:
Reference codes: codes as specified in WiMAX standard.
DBI A: code rate $5/6$ WiMAX mother code, dummy bits are appended at the end of payload to match code rates $3/4$, $2/3$ and $1/2$.
DBI B: code rate $5/6$ WiMAX mother code, dummy bits are appended at optimized positions according to Sec. III-B to match code rates $3/4$, $2/3$ and $1/2$.

Figure 2 shows an exemplary Tanner graph with five parity nodes and five information nodes three of which are used for dummy bits. In case (a), the dummy bit positions are simply chosen as the last three information bit positions resulting in an average pairwise distance of 2.67 edges between nodes associated with dummy bits. In case (b), on the other hand, the first, second, and fourth node are selected as dummy bit positions according to the proposed algorithm resulting in an increased average pairwise distance of 4 edges.

IV. Simulation Results

To evaluate the performance of the proposed approach for LDPC code rate-matching by optimized dummy bit insertion, we consider the LDPC codes specified in the IEEE 802.16e WiMAX standard [3]. After LDPC encoding of $K = R \cdot N$ equiprobable information bits, BPSK modulation is applied and the resulting symbols are transmitted with energy $E_s = 1$. The channel is modeled as additive white Gaussian noise (AWGN) channel with (known) power spectral density $\sigma_n^2 = N_0/2$. The receiver uses a soft demapper followed by Belief Propagation decoding as described in Sec. II with a maximum of 200 iterations. The following three setups are considered:

- **Reference codes** (solid lines)
  The codes of coded block length $N = 576$ and code rates $R \in \{\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}\}$ as specified in the WiMAX standard.

- **Dummy bit insertion (DBI) A** (dotted lines)
  Dummy bits are simply appended at the end of the payload. The rate $R = \frac{5}{6}$ code is used as mother code and $40\,\%$, $60\,\%$, and $80\,\%$ of the available information bit positions are used for dummy bits to match the effective code rates $R_{\text{eff}} = \frac{3}{4}$, $R_{\text{eff}} = \frac{2}{3}$, and $R_{\text{eff}} = \frac{1}{2}$, respectively.
- **Dummy bit insertion (DBI) B** (dashed lines)
  Same as for the setup DBI A, but dummy bits are now inserted at optimized positions according to Sec. III-B.

The block lengths N of the codes for the DBI A and DBI B setups were chosen to result in the same effective block length $N_{\text{eff}} = K_{\text{eff}} + M_{\text{eff}} = \frac{1-R}{1-R_{\text{eff}}} N = 576$ of the transmitted frame (payload and parity bits without dummy bits) as for the reference codes (i.e., $N = 864$ for $R_{\text{eff}} = \frac{3}{4}$, $N = 1152$ for $R_{\text{eff}} = \frac{2}{3}$ and $N = 1728$ for $R_{\text{eff}} = \frac{1}{2}$). Note that despite the increased block lengths of the codes using dummy bit insertion, decoding complexity is not increased since all additional nodes and edges of the Tanner graph are associated with dummy bits. Thus, these nodes and edges do not have to be considered during decoding, since they hold messages with values $+\infty$, which is the identity element of the boxplus operator [13].

The upper plot in Fig. 3 shows the frame error rate performance over the channel quality E_s/N_0 in dB. For an effective code rate of $R_{\text{eff}} = \frac{3}{4}$ (star markers) the waterfall behavior of all three setups is nearly identical, yet, the error floor performance is slightly degraded for the non-optimized setup DBI A and slightly enhanced for the optimized case DBI B. Except for a small loss in the waterfall region, a similar behavior is observed for an effective code rate of $R_{\text{eff}} = \frac{2}{3}$ (square markers). The most evident difference between setups DBI A and DBI B can be seen for the code rate $R_{\text{eff}} = \frac{1}{2}$ (circle markers). For the non-optimized case DBI A, the error floor already starts at a frame error rate of about 10^{-2}, in contrast to the optimized case DBI B which comes very close to the original rate $1/2$ code with a loss of less than $0.2\,\text{dB}$ near the waterfall and even less in the error floor region.

In the three lower plots of Fig. 3, the convergence behavior is depicted for the three different setups. For each effective code rate R_{eff} the frame error rate is plotted over the number of decoding iterations for a fixed E_s/N_0 value. For $R_{\text{eff}} = \frac{1}{2}$ and $E_s/N_0 = 0.99\,\text{dB}$ (left plot) it can be seen that convergence is much slower for setup DBI A than for the reference codes and setup DBI B. The setup DBI B using optimized dummy bit insertion, however, converges even faster than the original rate $1/2$ code but achieves a slightly higher frame error rate if the number of decoding iterations is high. For $R_{\text{eff}} = \frac{2}{3}$ and $E_s/N_0 = 2.64\,\text{dB}$ (center plot), a very similar behavior is observed. For $R_{\text{eff}} = \frac{3}{4}$ and $E_s/N_0 = 3.75\,\text{dB}$ (right plot), the original rate $3/4$ code exhibits the slowest convergence of all setups but still converges to a lower frame error rate than for setup DBI A. The optimized dummy bit insertion (DBI B), however, has the fastest convergence and also achieves the best frame error rate for small as well as for high numbers of iterations.

V. Conclusion

In this paper we have presented a novel algorithm for LDPC code rate-matching by dummy bit insertion. Known dummy bits are inserted into the information bit sequence before encoding, which results in a reduction of the effective code rate. The positions of dummy bits are determined by a heuristic optimization rule which results in a maximized average pairwise distance between the associated variable nodes. This optimization ensures that the perfect knowledge provided by the dummy bits to the decoder, is evenly distributed throughout the code's Tanner graph. It was shown by simulation results that significant gains in terms of frame error rate can be achieved by this approach compared to non-optimized insertion of dummy bits at the end of the payload.

References

[1] 3GPP TS 36.212, "Evolved Terrestrial Radio Access (E-UTRA); Multiplexing and Channel Coding," Version 8.2.0, 3GPP Technical Specification, 2008.

[2] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes," *IEEE Trans. Comm.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[3] "IEEE 802.16e: Air Interface for Fixed and Mobile Broadband Wireless Access Systems," IEEE Standard 802.16e, 2004.

[4] "IEEE 802.11n, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 5: Enhancements for Higher Throughput," IEEE Standard 802.11n-2009, 2009.

[5] R. G. Gallager, "Low-Density Parity-Check Codes," *IRE Trans. Inform. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[6] D. J. C. MacKay, "Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, Mar. 1999.

[7] J. Li and K. Narayanan, "Rate-Compatible Low Density Parity Check Codes for Capacity-Approaching ARQ Schemes in Packet Data Communications," *Proceedings Int. Conf. on Communications, Internet, and Information Technology*, Virgin Islands, USA, Nov. 2002, pp. 201–206.

[8] T. Tian and C. R. Jones, "Construction of Rate-Compatible LDPC Codes Utilizing Information Shortening and Parity Puncturing," *EURASIP J. Wirel. Commun. Netw.*, vol. 5, pp. 789–795, Oct. 2005.

[9] W. Xu and J. Romme, "A Class of Multirate Convolutional Codes by Dummy Bit Insertion," *IEEE Global Telecommunications Conference (GLOBECOM)*, San Francisco, CA, USA, November 2000.

[10] O. Collins and M. Hizlan, "Determinate state convolutional codes," *IEEE Trans. Comm.*, vol. 41, no. 12, pp. 1785–1794, Dec. 1993.

[11] R. M. Tanner, "A Recursive Approach to Low Complexity Codes," *IEEE Trans. Inform. Theory*, vol. 27, no. 5, pp. 533–547, Sept. 1981.

[12] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[13] J. Hagenauer, E. Offer, and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Comm.*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[14] T. Breddermann, H. Lüders, P. Vary, I. Aktas, and F. Schmidt, "Iterative Source-Channel Decoding with Cross-Layer Support for Wireless VoIP," *Proc. International ITG Conference on Source and Channel Coding*, Siegen, Germany, January 2010.

[15] M. Adrat, J. Spittka, S. Heinen, and P. Vary, "Error Concealment by Near Optimum MMSE-Estimation of Source Codec Parameters," *Proc. IEEE Workshop on Speech Coding*, 2000, pp. 84–86.

[16] N. Görtz, "Iterative Source-Channel Decoding Using Soft-In/Soft-Out Decoders," *Proc. IEEE International Symposium on Information Theory (ISIT)*, Sorrento, Italy, June 2000, p. 173.

[17] T. Breddermann, B. Eschbach, and P. Vary, "Dummy Bit Rate Matching for UMTS LTE," *ITG Fachtagung Mobilkommunikation*, Osnabrück, Germany, May 2011.

[18] T. Okamura, "A Hybrid ARQ Scheme Based on Shortened Low-Density Parity-Check Codes," *IEEE Wireless Communications and Networking Conference*, Apr. 2008, pp. 82–87.

[19] X. Liu, X. Wu, and C. Zhao, "Shortening for Irregular QC-LDPC Codes," *IEEE Communications Letters*, vol. 13, no. 8, pp. 612–614, Aug. 2009.