Bounds on the Number of Iterations for Turbo-Like Ensembles over the Binary Erasure Channel

Igal Sason, *Member, IEEE* and Gil Wiechman, *Student Member, IEEE*

Abstract

This paper provides simple lower bounds on the number of iterations which is required for successful message-passing decoding of some important families of graph-based code ensembles (including low-density parity-check codes and variations of repeat-accumulate codes). The transmission of the code ensembles is assumed to take place over a binary erasure channel, and the bounds refer to the asymptotic case where we let the block length tend to infinity. The simplicity of the bounds derived in this paper stems from the fact that they are easily evaluated and are expressed in terms of some basic parameters of the ensemble which include the fraction of degree-2 variable nodes, the target bit erasure probability and the gap between the channel capacity and the design rate of the ensemble. This paper demonstrates that the number of iterations which is required for successful message-passing decoding scales at least like the inverse of the gap (in rate) to capacity, provided that the fraction of degree-2 variable nodes of these turbo-like ensembles does not vanish (hence, the number of iterations becomes unbounded as the gap to capacity vanishes).

Index terms – Accumulate-repeat-accumulate (ARA) codes, area theorem, binary erasure channel (BEC), density evolution (DE), extrinsic information transfer (EXIT) charts, iterative message-passing decoding, low-density parity-check (LDPC) codes, stability condition.

I. Introduction

During the last decade, there have been many developments in the construction and analysis of low-complexity error-correcting codes which closely approach the Shannon capacity limit of many standard communication channels with feasible complexity. These codes are understood to be codes defined on graphs, together with the associated iterative decoding algorithms. Graphs serve not only to describe the codes themselves, but more importantly, they structure the operation of their efficient sub-optimal iterative decoding algorithms.

Proper design of codes defined on graphs enables to asymptotically achieve the capacity of the binary erasure channel (BEC) under iterative message-passing decoding. Capacity-achieving sequences of ensembles of low-density parity-check (LDPC) codes were originally introduced by Shokrollahi [29] and by Luby et al. [13], and a systematic study of capacity-achieving sequences of LDPC ensembles was presented by Oswald and Shokrollahi [19] for the BEC. Analytical bounds on the maximal achievable rates of LDPC ensembles were derived by Barak et al. [6] for the asymptotic case where the block length tends to infinity; this analysis provides a lower bound on the gap between the channel capacity and the achievable rates of LDPC ensembles under iterative decoding. The decoding complexity of LDPC codes under iterative message-passing decoding scales linearly with the block length, though their encoding complexity may be super-linear with the block length. However, the class of repeat-accumulate codes and their more recent variants (see, e.g., [1], [10] and [21]) exhibit the 'interleaver gain' phenomenon, and their encoding and decoding complexities scale both linearly with the block length. Due to the simplicity of the density evolution analysis for the BEC, suitable constructions of capacity-achieving ensembles of variants of repeat-accumulate codes were devised in [10], [20], [21] and [26]. All these works rely on the density evolution analysis for the BEC, and provide an asymptotic analysis which refers to the case where we let the block length of these code ensembles tend to infinity.

Rateless capacity-achieving codes for the BEC were introduced by Luby [14], and later improved by Shokrollahi [30]. The innovation of this approach enables to achieve the capacity of the BEC without the knowledge of the channel parameter.

This paper was submitted to the *IEEE Transactions on Information Theory* in November 2007, and Revised in February 2009. The research work of was supported by the Israel Science Foundation (grant no. 1070/07).

The authors are with the Department of Electrical Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel (email: {sason@ee, igillw@tx}.technion.ac.il).

The performance analysis of finite-length LDPC code ensembles whose transmission takes place over the BEC was introduced by Di et al. [8]. This analysis considers sub-optimal iterative message-passing decoding as well as optimal maximum-likelihood decoding. In [2], an efficient approach to the design of LDPC codes of finite length was introduced by Amraoui et al.; this approach is specialized for the BEC, and it enables to design such code ensembles which perform well under iterative decoding with a practical constraint on the block length. In [23], Richardson and Urbanke initiated the analysis of the distribution of the number of iterations needed for the decoding of LDPC ensembles of finite block length which are communicated over the BEC.

For general channels, the number of iterations is an important factor in assessing the decoding complexity of graph-based codes under iterative message-passing decoding. The second factor determining the decoding complexity of such codes is the complexity of the Tanner graph which is used to represent the code; this latter quantity, defined as the number of edges in the graph per information bit, serves as a measure for the decoding complexity per iteration.

The extrinsic information transfer (EXIT) charts, pioneered by ten Brink ([31], [32]), form a powerful tool for an efficient design of codes defined on graphs by tracing the convergence behavior of their iterative decoders. EXIT charts provide a good approximative engineering tool for tracing the convergence behavior of soft-input soft-output iterative decoders; they suggest a simplified visualization of the convergence of these decoding algorithms, based on a single parameter which represents the exchange of extrinsic information between the constituent decoders. For the BEC, the EXIT charts coincide with the density evolution analysis (see [22]) which is simplified in this case to a one-dimensional analysis.

A numerical approach for the joint optimization of the design rate and decoding complexity of LDPC ensembles was provided in [4]; it is assumed there that the transmission of these code ensembles takes place over a memoryless binary-input output-symmetric (MBIOS) channel, and the analysis refers to the asymptotic case where we let the block length tend to infinity. For the simplification of the numerical optimization, a suitable approximation of the number of iterations was used in [4] to formulate this joint optimization as a convex optimization problem. Due to the efficient tools which currently exist for a numerical solution of convex optimization problems, this approach suggests an engineering tool for the design of good LDPC ensembles which possess an attractive tradeoff between the decoding complexity and the asymptotic gap to capacity (where the block length of these code ensembles is large enough). This numerical approach however is not amenable for drawing rigorous theoretical conclusions on the tradeoff between the number of iterations and the performance of the code ensembles. A different numerical approach for approximating the number of iterations for LDPC ensembles operating over the BEC is addressed in [15].

A different approach for characterizing the complexity of iterative decoders was suggested by Khandekar and McEliece (see [11], [12], [16]). Their questions and conjectures were related to the tradeoff between the asymptotic achievable rates and the complexity under iterative message-passing decoding; they initiated a study of the encoding and decoding complexity of graph-based codes in terms of the achievable gap (in rate) to capacity. It was conjectured there that for a large class of channels, if the design rate of a suitably designed ensemble forms a fraction $1-\varepsilon$ of the channel capacity, then the decoding complexity scales like $\frac{1}{\varepsilon}\ln\frac{1}{\varepsilon}$. The logarithmic term in this expression was attributed to the graphical complexity (i.e., the decoding complexity per iteration), and the number of iterations was conjectured to scale like $\frac{1}{\varepsilon}$. There is one exception: For the BEC, the complexity under the iterative message-passing decoding algorithm behaves like $\ln\frac{1}{\varepsilon}$ (see [13], [25], [26] and [29]). This is true since the absolute reliability provided by the BEC allows every edge in the graph to be used only once during the iterative decoding. Hence, for the BEC, the number of iterations performed by the decoder serves mainly to measure the delay in the decoding process, while the decoding complexity is closely related to the complexity of the Tanner graph which is chosen to represent the code. The graphical complexity required for LDPC and systematic irregular repeat-accumulate (IRA) code ensembles to achieve a fraction $1-\varepsilon$ of the capacity of a BEC under iterative decoding was studied in [25] and [26]. It was shown in these papers that the graphical complexity of these ensembles must scale at least like $\ln\frac{1}{\varepsilon}$; moreover, some explicit constructions were shown to approach the channel capacity with such a scaling of the graphical complexity. An additional degree of freedom which is obtained by introducing state nodes in the graph (e.g., punctured bits) was exploited in [20] and [21] to construct capacity-achieving ensembles of graph-based codes which achieve an improved tradeoff between complexity and achievable rates. Surprisingly, these capacity-achieving ensembles under iterative decoding were demonstrated to maintain a *bounded graphical complexity* regardless of the erasure probability of the BEC. A similar result of a bounded graphical complexity for capacity-achieving ensembles over the BEC was also obtained in [9].

This paper provides simple lower bounds on the number of iterations which is required for successful message-passing decoding of graph-based code ensembles. The transmission of these ensembles is assumed to take place over the BEC, and the bounds refer to the asymptotic case where the block length tends to infinity. The simplicity of the bounds derived in this paper stems from the fact that they are easily evaluated and are expressed in terms of some basic parameters of the considered ensemble; these include the fraction of degree-2 variable nodes, the target bit erasure probability and the gap between the channel capacity and the design rate of the ensemble. The bounds derived in this paper demonstrate that the number of iterations which is required for successful message-passing decoding scales at least like the inverse of the gap (in rate) to capacity, provided that the fraction of degree-2 variable nodes of these turbo-like ensembles does not vanish (hence, the number of iterations becomes unbounded as the gap to capacity vanishes). The behavior of these lower bounds matches well with the experimental results and the conjectures on the number of iterations and complexity, as provided by Khandekar and McEliece (see [11], [12] and [16]). Note that lower bounds on the number of iterations in terms of the target bit erasure probability can be alternatively viewed as lower bounds on the achievable bit erasure probability as a function of the number of iterations performed by the decoder. As a result of this, the simple bounds derived in this paper provide some insight on the design of stopping criteria for iteratively decoded ensembles over the BEC (for other stopping criteria see, e.g., [3], [27]).

This paper is structured as follows: Section II presents some preliminary background, definitions and notation, Section III introduces the main results of this paper and discusses some of their implications, the proofs of these statements and some further discussions are provided in Section IV. Finally, Section V summarizes this paper. Proofs of some technical statements are relegated to the appendices.

II. Preliminaries

This section provides preliminary background and introduces notation for the rest of this paper.

A. Graphical Complexity of Codes Defined on Graphs

As noted in Section I, the decoding complexity of a graph-based code under iterative message-passing decoding is closely related to its graphical complexity, which we now define formally.

Definition 2.1 (Graphical Complexity): Let $\mathcal{C}$ be a binary linear block code of length n and rate R, and let $\mathcal{G}$ be an arbitrary representation of $\mathcal{C}$ by a Tanner graph. Denote the number of edges in $\mathcal{G}$ by E. The graphical complexity of $\mathcal{G}$ is defined as the number of edges in $\mathcal{G}$ per information bit of the code $\mathcal{C}$, i.e., $\Delta(\mathcal{G}) \triangleq \frac{E}{nR}$.
Note that the graphical complexity depends on the specific Tanner graph which is used to represent the code. An analysis of the graphical complexity for some families of graph-based codes is provided in [9], [20], [21], [25], [26].

B. Accumulate-Repeat-Accumulate Codes

Accumulate-repeat-accumulate (ARA) codes form an attractive coding scheme of turbo-like codes due to the simplicity of their encoding and decoding (where both scale linearly with the block length), and due to their remarkable performance under iterative decoding [1]. By some suitable constructions of puncturing patterns, ARA codes with small maximal node degree are presented in [1]; these codes perform very well even for short to moderate block lengths, and they suggest flexibility in the design of efficient rate-compatible codes operating on the same ARA decoder.

Ensembles of irregular and systematic ARA codes, which asymptotically achieve the capacity of the BEC with bounded graphical complexity, are presented in [21]. This bounded complexity result stays in contrast to LDPC ensembles, which have been shown to require unbounded graphical complexity in order to approach channel capacity, even under maximum-likelihood decoding (see [25]). In this section, we present ensembles of irregular and systematic ARA codes, and give a short overview of their encoding and decoding algorithms; this overview is required for the later discussion. The material contained in this section is taken from [21, Section II], and is introduced here briefly in order to make the paper self-contained.

From an encoding point of view, ARA codes are viewed as interleaved and serially concatenated codes. The encoding of ARA codes is done as follows: first, the information bits are accumulated (i.e., differentially encoded), and then the bits are repeated a varying number of times (by an irregular repetition code) and interleaved. The interleaved bits are partitioned into disjoint sets (whose size is not fixed in general), and the parity of each set of bits is computed (i.e., the bits are passed through an irregular single parity-check (SPC) code). Finally, the bits are accumulated a second time. A codeword of systematic ARA codes is composed of the information bits and the parity bits at the output of the second accumulator.

Since the iterative decoding algorithm of ARA codes is performed on the appropriate Tanner graph (see Fig. 1), this leads one to view them as sparse-graph codes from a decoding point of view.

Following the notation in [21], we refer to the three layers of bit nodes in the Tanner graphs as 'systematic bits' which form the systematic part of the codeword, 'punctured bits' which correspond to the output of the first accumulator and are not a part of the transmitted codeword, and 'code bits' which correspond to the output of the second accumulator and form the parity-bits of the codeword (see Fig. 1). Denoting the block length of the code by n and its dimension by k, each codeword is composed of k systematic bits and $n-k$ code bits. The two layers of check nodes are referred to as 'parity-check 1' nodes and 'parity-check 2' nodes, which correspond to the first and the second accumulators of the encoder, respectively. An ensemble of irregular ARA codes is defined by the block length n and the degree distributions of the 'punctured bit' and 'parity-check 2' nodes. Following the notation in [21], the degree distribution of the 'punctured bit' nodes is given by the power series

$$L(x) \triangleq \sum_{i=1}^{\infty} L_i x^i \tag{1}$$

where L_i designates the fraction of 'punctured bit' nodes whose degree is i. Similarly, the degree distribution of the 'parity-check 2' nodes is given by

$$R(x) \triangleq \sum_{i=1}^{\infty} R_i x^i \tag{2}$$

where R_i designates the fraction of these nodes whose degree is i. In both cases, degree of a node only refers to edges connecting the 'punctured bit' and the 'parity-check 2' layers, without the extra two edges which are connected to each of the 'punctured bit' nodes and 'parity-check 2' nodes from the accumulators (see Fig. 1). Considering the distributions from the edge perspective, we let

$$\lambda(x) \triangleq \sum_{i=1}^{\infty} \lambda_i x^{i-1}, \quad \rho(x) \triangleq \sum_{i=1}^{\infty} \rho_i x^{i-1} \tag{3}$$

designate the degree distributions from the edge perspective; here, λ_i (ρ_i) designates the fraction of edges connecting 'punctured bit' nodes to 'parity-check 2' nodes which are adjacent to 'punctured bit' ('parity-check 2') nodes of degree i. The design rate of a systematic ARA ensemble is given by $R = \frac{a_{\mathrm{R}}}{a_{\mathrm{L}}+a_{\mathrm{R}}}$ where

$$\begin{aligned} a_{\mathrm{L}} &\triangleq \sum_i iL_i = L'(1) = \frac{1}{\int_0^1 \lambda(t)\mathrm{d}t} \\ a_{\mathrm{R}} &\triangleq \sum_i iR_i = R'(1) = \frac{1}{\int_0^1 \rho(t)\mathrm{d}t} \end{aligned} \tag{4}$$

designate the average degrees of the 'punctured bit' and 'parity-check 2' nodes, respectively.

Iterative decoding of ARA codes is performed by passing messages on the edges of the Tanner graph in a layer-by-layer approach. Each decoding iteration starts with messages for the 'systematic bit' nodes to the 'parity-check 1' nodes, the latter nodes then use this information to calculate new messages to the 'punctured bit' nodes and so the information passes through layers down the graph and back up until the iteration ends with messages from the 'punctured bit' nodes to the 'parity-check 1' nodes. The final phase of messages from the 'parity-check 1' nodes to the 'systematic bit' nodes is omitted since the latter nodes are of degree one and so the outgoing message is not changed by incoming information. Assume that the code is transmitted over a BEC with erasure probability p. Since the systematic bits receive input from the channel, the probability of erasure in messages from the 'systematic bit' nodes to the 'parity-check 1' nodes is equal to p throughout the decoding process. For other messages, we denote by $x_i^{(l)}$ where $i = 0, 1, \ldots, 5$ the probability of erasure of the different message types at decoding iteration

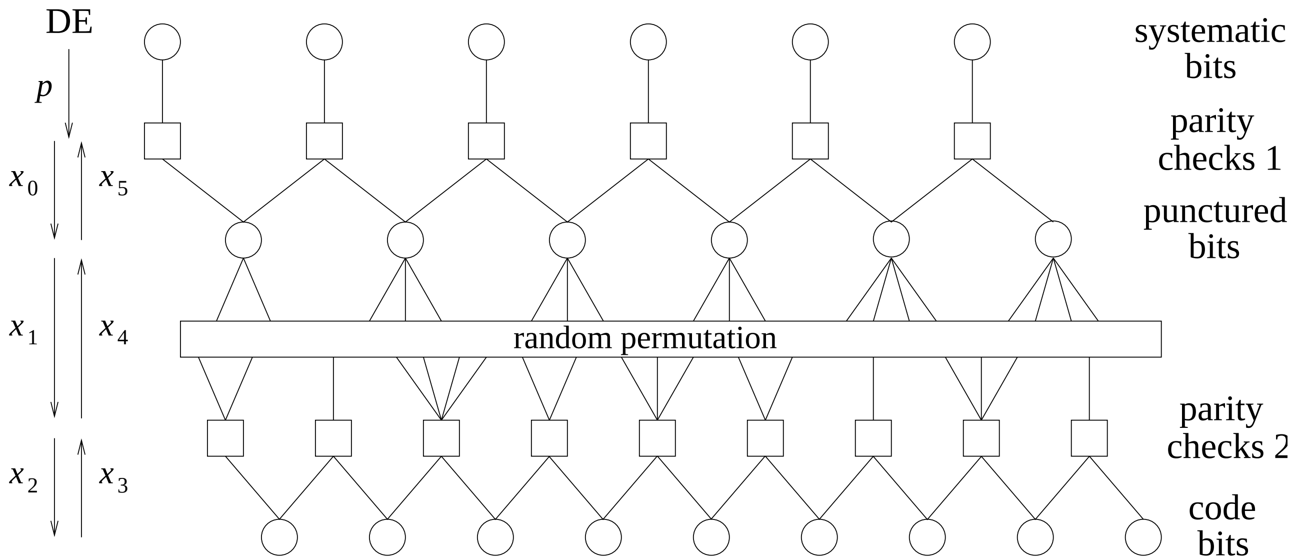

Fig. 1. Tanner graph of an irregular and systematic accumulate-repeat-accumulate code. This figure is reproduced from [21].

number l (where we start counting at zero). The variable $x_0^{(l)}$ corresponds to the probability of erasure in message from the 'parity-check 1' nodes to the 'punctured bit' nodes, $x_1^{(l)}$ tracks the erasure probability of messages from the 'punctured bit' nodes to the 'parity-check 2' nodes and so on. The density evolution (DE) equations for the decoder based on the Tanner graph in Figure 1 are given in [21], and we repeat them here:

$$\begin{aligned}
x_0^{(l)} &= 1 - \left(1 - x_5^{(l-1)}\right)(1-p) \\
x_1^{(l)} &= \left(x_0^{(l)}\right)^2 \lambda\left(x_4^{(l-1)}\right) \\
x_2^{(l)} &= 1 - R\left(1 - x_1^{(l)}\right)\left(1 - x_3^{(l-1)}\right) \qquad l = 1, 2, \ldots \\
x_3^{(l)} &= p\, x_2^{(l)} \\
x_4^{(l)} &= 1 - \left(1 - x_3^{(l)}\right)^2 \rho\left(1 - x_1^{(l)}\right) \\
x_5^{(l)} &= x_0^{(l)}\, L\left(x_4^{(l)}\right).
\end{aligned} \tag{5}$$

The stability condition for systematic ARA ensembles is derived in [21, Section II.D] and states that the fixed point $x_i^{(l)} = 0$ of the iterative decoding algorithm is stable if and only if

$$p^2 \lambda_2 \left(\rho'(1) + \frac{2pR'(1)}{1-p}\right) \leq 1\,. \tag{6}$$

C. Big-O notation

The terms O, Ω and Θ are widely used in computer science to describe asymptotic relationships between functions (for formal definitions see e.g., [34]). In our context, we refer to the gap (in rate) to capacity, denoted by ε, and discuss in particular the case where $0 \leq \varepsilon \ll 1$ (i.e., sequences of capacity-approaching ensembles). Accordingly, we define

- $f(\varepsilon) = O(g(\varepsilon))$ means that there are positive constants c and δ, such that $0 \leq f(\varepsilon) \leq c\, g(\varepsilon)$ for all $0 \leq \varepsilon \leq \delta$.
- $f(\varepsilon) = \Omega(g(\varepsilon))$ means that there are positive constants c and δ, such that $0 \leq c\, g(\varepsilon) \leq f(\varepsilon)$ for all $0 \leq \varepsilon \leq \delta$.
- $f(\varepsilon) = \Theta(g(\varepsilon))$ means that there are positive constants c_1, c_2 and δ, such that $0 \leq c_1\, g(\varepsilon) \leq f(\varepsilon) \leq c_2\, g(\varepsilon)$ for all $0 \leq \varepsilon \leq \delta$.

Note that for all the above definitions, the values of c, c_1, c_2 and δ must be fixed for the function f and should not depend on ε.

III. Main Results

In this section, we present lower bounds on the required number of iterations used by a message-passing decoder for code ensembles defined on graphs. The communication is assumed to take place over a BEC, and we consider the asymptotic case where the block length of these code ensembles tends to infinity.

Definition 3.1: Let $\{\mathcal{C}_m\}_{m\in\mathbb{N}}$ be a sequence of code ensembles. Assume a common block length (n_m) of the codes in $\mathcal{C}_m$ which tends to infinity as m grows. Let the transmission of this sequence take place over a BEC with capacity C. The sequence $\{\mathcal{C}_m\}$ is said to *achieve a fraction* $1-\varepsilon$ *of the channel capacity under some given decoding algorithm* if the asymptotic rate of the codes in $\mathcal{C}_m$ satisfies $R \geq (1-\varepsilon)C$ and the achievable bit erasure probability under the considered algorithm vanishes as m becomes large.
In the continuation, we consider a standard iterative message-passing decoder for the BEC, and address the number of iterations which is required in terms of the achievable fraction of the channel capacity under this decoding algorithm.

Theorem 3.1: **[Lower bound on the number of iterations for LDPC ensembles transmitted over the BEC]**. Let $\{(n_m, \lambda, \rho)\}_{m\in\mathbb{N}}$ be a sequence of LDPC ensembles whose transmission takes place over a BEC with erasure probability p. Assume that this sequence achieves a fraction $1-\varepsilon$ of the channel capacity under message-passing decoding. Let $L_2 = L_2(\varepsilon)$ be the fraction of variable nodes of degree 2 for this sequence. In the asymptotic case where the block length tends to infinity, let $l = l(\varepsilon, p, P_{\mathrm{b}})$ denote the number of iterations which is required to achieve an average bit erasure probability P_{b} over the ensemble. Under the mild condition that $P_{\mathrm{b}} < p\,L_2(\varepsilon)$, the required number of iterations satisfies the lower bound

$$l(\varepsilon, p, P_{\mathrm{b}}) \geq \frac{2}{1-p}\left(\sqrt{p\,L_2(\varepsilon)} - \sqrt{P_{\mathrm{b}}}\right)^2 \frac{1}{\varepsilon}\,. \tag{7}$$

Corollary 3.1: Under the assumptions of Theorem 3.1, if the fraction of degree-2 variable nodes stays strictly positive as the gap (in rate) to capacity vanishes, i.e., if

$$\lim_{\varepsilon\to 0} L_2(\varepsilon) > 0$$

then the number of iterations which is required in order to achieve an average bit erasure probability $P_{\mathrm{b}} < p\,L_2(\varepsilon)$ under iterative message-passing decoding scales at least like the inverse of this gap to capacity, i.e.,

$$l(\varepsilon, p, P_{\mathrm{b}}) = \Omega\left(\frac{1}{\varepsilon}\right).$$

Discussion 3.1: **[Effect of messages' scheduling on the number of iterations]** The lower bound on the number of iterations as provided in Theorem 3.1 refers to the *flooding schedule* where in each iteration, all the variable nodes and subsequently all the parity-check nodes send messages to their neighbors. Though it is the commonly used scheduling used by iterative message-passing decoding algorithms, an alternative scheduling of the messages may provide a faster convergence rate for the iterative decoder. As an example, [28] considers the convergence rate of a *serial scheduling* where instead of sending all the messages from the variable nodes to parity-check nodes and then all the messages from check nodes to variable nodes, as done in the flooding schedule, these two phases are interleaved. Based on the density evolution analysis which applies to the asymptotic case of an infinite block length, it is demonstrated in [28] that under some assumptions, the required number of iterations for LDPC decoding over the BEC with serial scheduling is reduced by a factor of two (as compared to the flooding scheduling). It is noted that the main result of Theorem 3.1 is the introduction of a rigorous and simple lower bound on the number of iterations for LDPC ensembles which scales like the reciprocal of the gap between the channel capacity and the design rate of the ensemble. Though such a scaling of this bound is proved for the commonly used approach of flooding scheduling, it is likely to hold also for other efficient approaches of scheduling. It is also noted that this asymptotic scaling of the lower bound on the number of iterations supports the conjecture of Khandekar and McEliece [11].

Discussion 3.2: **[On the dependence of the bounds on the fraction of degree-2 variable nodes]** The lower bound on the number of iterations in Theorem 3.1 becomes trivial when the fraction of variable nodes of degree 2 vanishes. Let us focus our attention on sequences of ensembles which approach the channel capacity under iterative message-passing decoding (i.e., $\varepsilon \to 0$). For the BEC, several such sequences have been constructed (see e.g. [13], [29]). Asymptotically, as the gap to capacity vanishes, all of these sequences known to date satisfy the stability condition with equality; this property is known as the flatness condition [29]. In [24, Lemma 7], the asymptotic fraction of degree 2 variable nodes for capacity-approaching sequences of LDPC ensembles over the BEC is calculated. This lemma states that for such sequences which satisfy the following two conditions as the gap to capacity vanishes:

- The stability condition is satisfied with equality (i.e., the flatness condition holds)
- The limit of the ratio between the standard deviation and the expectation of the right degree exists and is finite

the asymptotic fraction of degree–2 variable nodes does not vanish. In fact, for various sequences of capacity approaching LDPC ensembles known to date (see [13], [19], [29]), the ratio between the standard deviation and the expectation of the right degree-distribution tends to zero; in this case, [24, Lemma 7] implies that the fraction of degree-2 variable nodes tends to $\frac{1}{2}$ irrespectively of the erasure probability of the BEC, as can be verified directly for these code ensembles.

Discussion 3.3: **[Concentration of the lower bound]** Theorem 3.1 applies to the required number of iterations for achieving an average bit erasure probability P_{b} where this average is taken over the LDPC ensemble whose block length tends to infinity. Although we consider an expectation over the LDPC ensemble, note that l is deterministic as it is the smallest integer for which the average bit erasure probability does not exceed a fixed value. As shown in the proof (see Section IV), the derivation of this lower bound relies on the density evolution technique which addresses the average performance of the ensemble. Based on concentration inequalities, it is proved that the performance of individual codes from the ensemble concentrates around the average performance over the ensemble as we let the block length tend to infinity [22, Appendix C]. In light of this concentration result and the use of density evolution in Section IV (which applies to the case of an infinite block length), it follows that the lower bound on the number of iterations in Theorem 3.1 is valid with probability 1 for individual codes from the ensemble. This also holds for the ensembles of codes defined on graphs considered in Theorems 3.2 and 3.3.

Discussion 3.4: **[On the number of required iterations for showing a mild improvement in the erasure probability during the iterative process]** Note that for capacity-approaching LDPC ensembles, the lower bound on the number of iterations tells us that even for successfully starting the iteration process and reducing the bit erasure probability by a factor which is below the fraction of degree-2 variable nodes, the required number of iterations already scales like $\frac{1}{\varepsilon}$. This is also the behavior of the lower bound on the number of iterations even when the bit erasure probability should be made arbitrarily small; this lower bound therefore indicates that for capacity-approaching LDPC ensembles, a significant number of the iterations is performed for the starting process of the iterative decoding where the bit erasure probability is merely reduced by a factor of $\frac{1}{2}$ as compared to the erasure probability of the channel (see Discussion 3.2 as a justification for the one-half factor). This conclusion is also well interpreted by the area theorem and the asymptotic behavior of the two EXIT curves (for the variable nodes and the parity-check nodes) in the limit where $\varepsilon \to 0$; as the gap to capacity vanishes, both curves tend to be a step function jumping from 0 to 1 at the origin, so the iterations progress very slowly at the initial stages of the decoding process.

In the asymptotic case where we let the block length tend to infinity and the transmission takes place over the BEC, suitable constructions of capacity-achieving systematic ARA ensembles enable a fundamentally improved tradeoff between their graphical complexity and their achievable gap (in rate) to capacity under iterative decoding (see [21]). The graphical complexity of these systematic ARA ensembles remains bounded (and quite small) as the gap to capacity for these ensembles vanishes under iterative decoding; this stays in contrast to un-punctured LDPC code ensembles [25] and *systematic* irregular repeat-accumulate (IRA) ensembles [26] whose graphical complexity necessarily becomes unbounded as the gap to capacity vanishes (see [21, Table I]). This observation raises the question whether the number of iterations which is required to achieve a desired bit erasure probability under iterative decoding, can be reduced by using systematic ARA ensembles. The following theorem provides a lower bound on the number of iterations required to achieve a desired bit erasure probability under iterative message-passing decoding; it shows that similarly to the parallel result for LDPC ensembles (see Theorem 3.1), the required number of iterations for systematic ARA codes scales at least like the inverse of the gap to capacity.

Theorem 3.2: **[Lower bound on the number of iterations for systematic ARA ensembles transmitted over the BEC]**. Let $\{(n_m, \lambda, \rho)\}_{m\in\mathbb{N}}$ be a sequence of systematic ARA ensembles whose transmission takes place over a BEC with erasure probability p. Assume that this sequence achieves a fraction $1-\varepsilon$ of the channel capacity under message-passing decoding. Let $L_2 = L_2(\varepsilon)$ be the fraction of 'punctured bit' nodes of degree 2 for this sequence (where the two edges related to the accumulator are not taken into account). In the asymptotic case where the block length tends to infinity, let $l = l(\varepsilon, p, P_{\text{b}})$ designate the required number of iterations to achieve an average bit erasure probability P_{b} of the systematic bits. Under the mild condition that $1 - \sqrt{1 - \frac{P_{\text{b}}}{p}} < p\, L_2(\varepsilon)$, the number of iterations satisfies the lower bound

$$l(\varepsilon, p, P_{\text{b}}) \geq 2p(1-\varepsilon) \left(\sqrt{p\, L_2(\varepsilon)} - \sqrt{1 - \sqrt{1 - \frac{P_{\text{b}}}{p}}} \right)^2 \frac{1}{\varepsilon}\,. \tag{8}$$

As noted in Section II-B, systematic ARA codes can be viewed as serially concatenated codes where the systematic bits are associated with the outer code. These codes can be therefore decoded iteratively by using a turbo-like decoder for interleaved and serially concatenated codes. The following proposition states that the lower bound on the number of iterations in Theorem 3.2 is also valid for such an iterative decoder.

Proposition 3.1: **[Lower bound on the number of iterations for systematic ARA codes under turbo-like decoding]**. Under the assumptions and notation of Theorem 3.2, the lower bound on the number of iterations in (8) is valid also when the decoding is performed by a turbo-like decoder for uniformly interleaved and serially concatenated codes.

The reader is referred to Appendix I for a detailed proof. The following theorem which refers to irregular repeat-accumulate (IRA) ensembles is proved in a conceptually similar way to the proof of Theorem 3.2.

Theorem 3.3: **[Lower bound on the number of iterations for IRA ensembles transmitted over the BEC]**. Let $\{(n_m, \lambda, \rho)\}_{m \in \mathbb{N}}$ be a sequence of (systematic or non-systematic) IRA ensembles whose transmission takes place over a BEC with erasure probability p. Assume that this sequence achieves a fraction $1 - \varepsilon$ of the channel capacity under message-passing decoding. Let $L_2 = L_2(\varepsilon)$ be the fraction of 'information bit' nodes of degree 2 for this sequence. In the asymptotic case where the block length tends to infinity, let $l = l(\varepsilon, p, P_{\text{b}})$ designate the required number of iterations to achieve an average bit erasure probability P_{b} of the information bits. For systematic codes, if $P_{\text{b}} < p\, L_2(\varepsilon)$, then the number of iterations satisfies the lower bound

$$l(\varepsilon, p, P_{\text{b}}) \geq 2(1-\varepsilon) \left(\sqrt{p\, L_2(\varepsilon)} - \sqrt{P_{\text{b}}} \right)^2 \frac{1}{\varepsilon}\,. \tag{9}$$

For non-systematic codes, if $P_{\text{b}} < L_2(\varepsilon)$, then

$$l(\varepsilon, p, P_{\text{b}}) \geq 2(1-\varepsilon) \left(\sqrt{L_2(\varepsilon)} - \sqrt{P_{\text{b}}} \right)^2 \frac{1}{\varepsilon}. \tag{10}$$

IV. Derivation of the Bounds on the Number of Iterations

A. Proof of Theorem 3.1

Let $\{x^{(l)}\}_{l \in \mathbb{N}}$ designate the expected fraction of erasures in messages from the variable nodes to the check nodes at the l'th iteration of the message-passing decoding algorithm (where we start counting at $l = 0$). From density evolution, in the asymptotic case where the block length tends to infinity, $x^{(l)}$ is given by the recursive equation

$$x^{(l+1)} = p\, \lambda \left(1 - \rho(1 - x^{(l)}) \right)\,, \quad l \in \mathbb{N} \tag{11}$$

with the initial condition

$$x^{(0)} = p \tag{12}$$

where p designates the erasure probability of the BEC. Considering a sequence of $\{(n_m, \lambda, \rho)\}$ LDPC ensembles where we let the block length n_m tend to infinity, the average bit erasure probability after the l'th iteration is given by

$$P_{\text{b}}^{(l)} = p\, L(1 - \rho(1 - x^{(l)})) \tag{13}$$

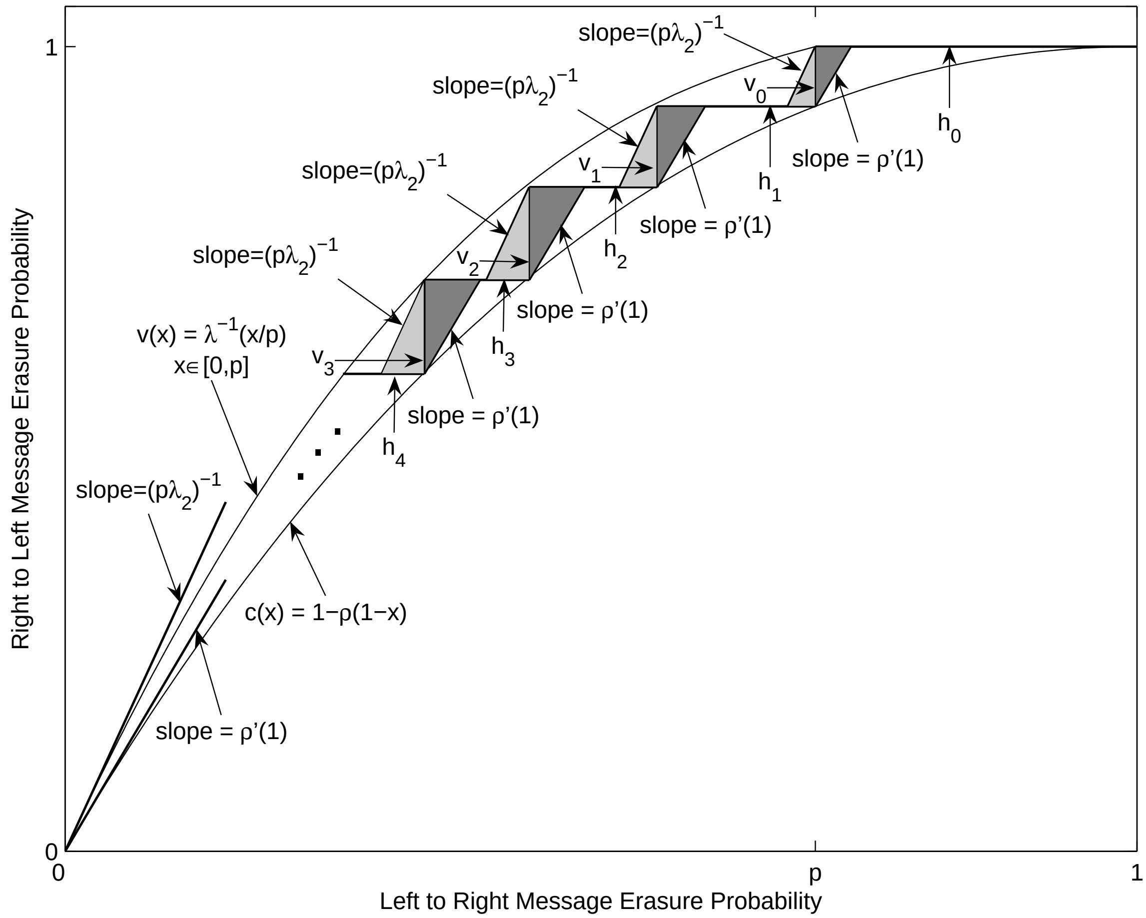

Fig. 2. Plot of the functions $c(x)$ and $v(x)$ for an ensemble of LDPC codes which achieves vanishing bit erasure probability under iterative message-passing decoding when communicated over a BEC whose erasure probability is equal to p. The horizontal and vertical lines track the evolution of the expected fraction of erasure messages from the variable nodes to the check nodes at each iteration of the message-passing decoding algorithm.

where L designates the common left degree distribution of the ensembles from the node perspective. Since the function $f(x) = p\,\lambda\big(1-\rho(1-x)\big)$ is monotonically increasing, Eqs. (11)–(13) imply that an average bit erasure probability of $P_{\rm b}$ is attainable under iterative message-passing decoding if and only if

$$p\;\lambda\big(1-\rho(1-x)\big) < x\,, \quad \forall x \in (x^*, p] \tag{14}$$

where x^* is the unique solution of

$$P_{\rm b} = p\;L\big(1-\rho(1-x^*)\big)\,.$$

Let us define the functions

$$c(x) \triangleq 1-\rho(1-x), \qquad v(x) = \begin{cases} \lambda^{-1}\left(\frac{x}{p}\right) & 0 \le x \le p \\ 1 & p < x \le 1 \end{cases}\,. \tag{15}$$

From the condition in (14), an average bit erasure probability of $P_{\rm b}$ is attained if and only if $c(x) < v(x)$ for all $x \in (x^*, p]$. Since we assume that vanishing bit erasure probability is achievable under message-passing decoding, it follows that $c(x) < v(x)$ for all $x \in (0, p]$. Figure 2 shows a plot of the functions $c(x)$ and $v(x)$ for an ensemble of LDPC codes which achieves vanishing bit erasure probability under iterative decoding as the block length tends to infinity. The horizontal and vertical lines, labeled $\{h_l\}_{l\in\mathbb{N}}$ and $\{v_l\}_{l\in\mathbb{N}}$, respectively, are used to track the expected fraction of erased messages from the variable nodes to the parity-check nodes at each iteration of the message-passing decoding algorithm. From (11) and (12), the expected fraction of erased left to right messages in the l'th decoding iteration (where we start counting at zero) is equal to the x value at the left tip of the horizontal line h_l. The right-angled triangles shaded in gray will be used later in the proof.

The first step in the proof of Theorem 3.1 is calculating the area bounded by the curves $c(x)$ and $v(x)$. This is done in the following lemma which is based on the area theorem for the BEC [5].

Lemma 4.1:

$$\int_0^1 \big(v(x) - c(x)\big)\mathrm{d}x = \frac{C-R}{a_{\mathrm{L}}} \tag{16}$$

where $C = 1-p$ is the capacity of the BEC, R is the design rate of the ensemble, and a_{L} is the average left degree of the ensemble.

Proof: The proof of this equality is straightforward. Alternatively, the reader is referred to the matching condition in [22, Section 3.14.4] which is justified via the area theorem in [5]. ∎

Let us consider the two sets of right-angled triangles shown in two shades of gray in Figure 2. The set of triangles which are shaded in dark gray are defined so that one of the legs of triangle number i (counting from right to left and starting at zero) is the vertical line v_i, and the slope of the hypotenuse is equal to $c'(0) = \rho'(1)$. Since $c(x)$ is concave for all $x \in [0,1]$, these triangles are guaranteed to be above the curve of the function c. Since the slope of the hypotenuse is $\rho'(1)$, the area of the i'th triangle in this set is

$$A_i = \frac{1}{2}\,|v_i|\,\left(\frac{|v_i|}{\rho'(1)}\right) = \frac{|v_i|^2}{2\rho'(1)} \tag{17}$$

where $|v_i|$ is the length of v_i. We now turn to consider the second set of triangles, which are shaded in light gray. Note that the function $\lambda(x)$ is monotonically increasing and convex in $[0,1]$ and also that $\lambda(0) = 0$ and $\lambda(1) = 1$. This implies that λ^{-1} is concave in $[0,1]$ and therefore $v(x)$ is concave in $[0,p]$. The triangles shaded in light gray are defined so that one of the legs of triangle number i (again, counting from the right and starting at zero) is the vertical line v_i and the slope of the hypotenuse is given by

$$v'(0) = \frac{1}{p}\left(\lambda^{-1}\right)'(0) = \frac{1}{p\lambda'(0)} = \frac{1}{p\lambda_2}$$

where the second equality follows since $\lambda(0) = 0$. The concavity of $v(x)$ in $[0,p]$ guarantees that these triangles are below the curve of the of function v. The area of the i'th triangle in this second set of triangles is given by

$$B_i = \frac{1}{2}\,|v_i|\,\left(|v_i|\,p\lambda_2\right) = \frac{p\lambda_2\,|v_i|^2}{2}\,. \tag{18}$$

Since $v(x)$ is monotonically increasing with x, the dark-shaded triangles lie below the curve of the function v. Similarly, the monotonicity of $c(x)$ implies that the light-shaded triangles are above the curve of the function c. Hence, both sets of triangles form a subset of the domain bounded by the curves of $c(x)$ and $v(x)$. By their definitions, the i'th dark triangle is on the right of v_i, and the i'th light triangle lies to the left of v_i; therefore, the triangles do not overlap. Combining (17), (18) and the fact that the triangles do not overlap, and applying Lemma 4.1, we get

$$\begin{aligned}\frac{C-R}{a_{\mathrm{L}}} &= \int_0^1 \big(v(x)-c(x)\big)\mathrm{d}x \\ &\geq \sum_{i=0}^{\infty}(A_i + B_i) \\ &\geq \frac{1}{2}\left(\frac{1}{\rho'(1)} + p\lambda_2\right)\sum_{i=0}^{l-1}|v_i|^2\end{aligned} \tag{19}$$

where l is an arbitrary natural number. Since we assume that the bit erasure probability vanishes under iterative message-passing decoding, the stability condition implies that

$$\frac{1}{\rho'(1)} \geq p\,\lambda_2\,. \tag{20}$$

Substituting (20) and $R = (1-\varepsilon)C$ in (19) gives

$$C\varepsilon \geq a_{\rm L}\, p\lambda_2 \sum_{i=0}^{l-1} |v_i|^2. \tag{21}$$

The definition of h_l and v_l in Figure 2 implies that for an arbitrary iteration l

$$1 - \rho(1 - x^{(l)}) = c(x^{(l)}) = 1 - \sum_{i=0}^{l} |v_i| \, .$$

Substituting the last equality in (13) yields that the average bit erasure probability after iteration number $l-1$ can be expressed as

$$P_{\rm b}^{(l-1)} = p\, L\left(1 - \sum_{i=0}^{l-1} |v_i|\right) \, . \tag{22}$$

Let l designate the number of iterations required to achieve an average bit erasure probability $P_{\rm b}$ over the ensemble (where we let the block length tend to infinity), i.e., l is the smallest integer which satisfies $P_{\rm b}^{(l-1)} \leq P_{\rm b}$ since we start counting at $l = 0$. Although we consider an expectation over the LDPC ensemble, note that l is deterministic as it is the smallest integer for which the average bit erasure probability does not exceed $P_{\rm b}$. Since L is monotonically increasing, (22) provides a lower bound on $\sum_{i=0}^{l-1} |v_i|$ of the form

$$\sum_{i=0}^{l-1} |v_i| \geq 1 - L^{-1}\left(\frac{P_{\rm b}}{p}\right) \, . \tag{23}$$

From the Cauchy-Schwartz inequality, we get

$$\left(\sum_{i=0}^{l-1} |v_i|\right)^2 \leq \sum_{i=0}^{l-1} 1 \sum_{i=0}^{l-1} |v_i|^2 = l \sum_{i=0}^{l-1} |v_i|^2. \tag{24}$$

Combining the above inequality with (21) and (23) gives the inequality

$$C\varepsilon \geq \frac{a_{\rm L}\, p\lambda_2 \left(1 - L^{-1}\left(\frac{P_{\rm b}}{p}\right)\right)^2}{l}$$

which provides the following lower bound on the number of iterations l:

$$l \geq \frac{a_{\rm L}\, p\lambda_2 \left(1 - L^{-1}\left(\frac{P_{\rm b}}{p}\right)\right)^2}{(1-p)\varepsilon}. \tag{25}$$

To continue the proof, we derive a lower bound on $1 - L^{-1}(x)$ for $x \in (0, 1)$. Since the fraction of variable nodes of degree i is non-negative for all $i = 2, 3, \ldots$, we have

$$L(x) = \sum_i L_i x^i \geq L_2 x^2, \quad x \geq 0.$$

Substituting $t = L(x)$ gives

$$t \geq L_2 \cdot \left(L^{-1}(t)\right)^2, \quad \forall t \in (0, 1)$$

which is transformed into the following lower bound on $1 - L^{-1}(x)$:

$$1 - L^{-1}(x) \geq 1 - \sqrt{\frac{x}{L_2}}\,, \quad \forall x \in (0, 1) \, . \tag{26}$$

Under the assumption $\frac{P_{\rm b}}{p} < L_2$, substituting (26) in (25) gives

$$\begin{aligned} l \quad &\geq \quad \frac{a_{\rm L}\, p\lambda_2 \left(\sqrt{L_2} - \sqrt{\frac{P_{\rm b}}{p}}\right)^2}{L_2\,(1-p)\varepsilon} \\ &= \quad \frac{a_{\rm L}\, \lambda_2 \left(\sqrt{p\, L_2} - \sqrt{P_{\rm b}}\right)^2}{L_2\,(1-p)\varepsilon} \, . \end{aligned} \tag{27}$$

The lower bound in (7) is obtained by substituting the equality $L_2 = \frac{\lambda_2 a_{\mathrm{L}}}{2}$ into (27).

Taking the limit where the average bit erasure probability tends to zero on both sides of (7) gives the following lower bound on the number of iterations:

$$l(\varepsilon, p, P_{\mathrm{b}} \to 0) \geq \frac{2p}{1-p} \frac{L_2(\varepsilon)}{\varepsilon} \,.$$

B. Proof of Theorem 3.2

We begin the proof by considering the expected fraction of erasure messages from the 'punctured bit' nodes to the 'parity-check 2' nodes (see Fig. 1). The following lemma provides a lower bound on the expected fraction of erasures in the l'th decoding iteration in terms of this expected fraction at the preceding iteration.

Lemma 4.2: Let (n, λ, ρ) be an ensemble of systematic ARA codes whose transmission takes place over a BEC with erasure probability p. Then, in the limit where the block length tends to infinity, the expected fraction of erasure messages from the 'punctured bit' nodes to the 'parity-check 2' nodes at the l'th iteration satisfies

$$x_1^{(l)} \geq \widetilde{\lambda}\left(1 - \widetilde{\rho}(1 - x_1^{(l-1)})\right), \quad l = 1, 2, \dots \tag{28}$$

where the tilted degree distributions $\widetilde{\lambda}$ and $\widetilde{\rho}$ are given as follows (see [21]):

$$\widetilde{\lambda}(x) \triangleq \left(\frac{p}{1-(1-p)L(x)}\right)^2 \lambda(x) \tag{29}$$

$$\widetilde{\rho}(x) \triangleq \left(\frac{1-p}{1-pR(x)}\right)^2 \rho(x) \tag{30}$$

and L and R designate the degree distributions of the ARA ensemble from the node perspective.

Proof: See Appendix II.A. ■

From Fig. 1, it can be readily verified that the probabilities x_0 and x_1 for erasure messages at iteration number zero are equal to 1, i.e.,

$$x_0^{(0)} = x_1^{(0)} = 1. \tag{31}$$

Let us look at the RHS of (28) as a function of x, and observe that it is monotonically increasing over the interval $[0, 1]$. Let us compare the performance of a systematic ARA ensemble whose degree distributions are (λ, ρ) with an LDPC ensemble whose degree distributions are given by $(\widetilde{\lambda}, \widetilde{\rho})$ (see (29) and (30)) under iterative message-passing decoding. Given the initial condition $x_1^{(0)} = 1$, the following conclusion is obtained by recursively applying Lemma 4.2: For any iteration, the erasure probability for messages delivered from 'punctured bit' nodes to 'parity-check 2' nodes of the ARA ensemble (see Fig. 1) is lower bounded by the erasure probability of the left-to-right messages of the LDPC ensemble; this holds even if the a-priori information from the BEC is not used by the iterative decoder of the LDPC ensemble (note that the coefficient of $\widetilde{\lambda}$ in the RHS of (28) is equal to one). Note that unless the fraction of 'parity-check 2' nodes of degree 1 is strictly positive (i.e., $R_1 > 0$), the iterative decoding cannot be initiated for both ensembles (unless some the values of some 'punctured bits' of the systematic ARA ensemble are known, as in [21]). Hence, the comparison above between the ARA and LDPC ensembles is of interest under the assumption that $R_1 > 0$; this property is implied by the assumption of vanishing bit erasure probability for the systematic ARA ensemble under iterative message-passing decoding.

In [21, Section II.C.2], a technique called 'graph reduction' is introduced. This technique transforms the Tanner graph of a systematic ARA ensemble, transmitted over a BEC whose erasure probability is p, into a Tanner graph of an equivalent LDPC ensemble (where this equivalence holds in the asymptotic case where the block length tends to infinity). The variable and parity-check nodes of the equivalent LDPC code evolve from the 'punctured bit' and 'parity-check 2' nodes of the ARA ensemble, respectively, and their degree distributions (from the edge perspective) are given by $\widetilde{\lambda}$ and $\widetilde{\rho}$, respectively. It is also shown in [21] that $\widetilde{\lambda}$ and $\widetilde{\rho}$ are legitimate degree distribution functions, i.e., all the derivatives at zero are non-negative and $\widetilde{\lambda}(1) = \widetilde{\rho}(1) = 1$. As shown in [21, Eqs. (9)–(12)], the left and right degree distributions of the equivalent LDPC ensemble from the node perspective are given, respectively, by

$$\widetilde{L}(x) = \frac{\int_0^x \widetilde{\lambda}(t)\mathrm{d}t}{\int_0^1 \widetilde{\lambda}(t)\mathrm{d}t} = \frac{p\,L(x)}{1-(1-p)L(x)} \tag{32}$$

and

$$\widetilde{R}(x) = \frac{\int_0^x \widetilde{\rho}(t)\mathrm{d}t}{\int_0^1 \widetilde{\rho}(t)\mathrm{d}t} = \frac{(1-p)\,R(x)}{1-pR(x)}\ . \tag{33}$$

Let $P_{\mathrm{b}}^{(l)}$ designate the average erasure probability of the systematic bits after the l'th decoding iteration (where we start counting at $l = 0$). For LDPC ensembles, a simple relationship between the erasure probability of the code bits and the erasure probability of the left-to-right messages at the l'th decoding iteration is given in (13). For systematic ARA ensembles, a similar, though less direct, relationship exists between the erasure probability of the systematic bits after the l'th decoding iteration and $x_1^{(l)}$; this relationship is presented in the following lemma.

Lemma 4.3: Let (n, λ, ρ) be an ensemble of systematic ARA codes whose transmission takes place over a BEC with erasure probability p. Then, in the asymptotic case where the block length tends to infinity, the average erasure probability of the systematic bits after the l'th decoding iteration, $P_{\mathrm{b}}^{(l)}$, satisfies the inequality

$$1 - \sqrt{1 - \frac{P_{\mathrm{b}}^{(l)}}{p}} \geq \widetilde{L}\left(1 - \widetilde{\rho}\left(1 - x_1^{(l)}\right)\right) \tag{34}$$

where $\widetilde{\rho}$ and $\widetilde{L}$ are defined in (30) and (32), respectively (similarly to their definitions in [21]).

Proof: See Appendix II.B. ■

Remark 4.1: We note that when $P_{\mathrm{b}}^{(l)}$ is very small, the LHS of (34) satisfies

$$1 - \sqrt{1 - \frac{P_{\mathrm{b}}^{(l)}}{p}} \approx \frac{P_{\mathrm{b}}^{(l)}}{2p}\ ,$$

so (34) takes a similar form to (13) which refers to the erasure probability of LDPC ensembles.

Consider the number of iterations required for the message-passing decoder, operating on the Tanner graphs of the systematic ARA ensemble, to achieve a desired bit erasure probability P_{b}. Combining Lemmas 4.2 and 4.3, and the initial condition in (31), a lower bound on this number of iterations can be deduced. More explicitly, it is lower bounded by the number of iterations which is required to achieve a bit erasure probability of $1 - \sqrt{1 - \frac{P_{\mathrm{b}}}{p}}$ for the LDPC ensemble whose degree distributions are $(\widetilde{\lambda}, \widetilde{\rho})$ and where the erasure probability of the BEC is equal to 1. It is therefore tempting to apply the lower bound on the number of iterations in Theorem 3.1, which refers to LDPC ensembles, as a lower bound on the number of iterations for the ARA ensemble. Unfortunately, the LDPC ensemble with the tilted pair of degree distributions $(\widetilde{\lambda}, \widetilde{\rho})$ is transmitted over a BEC whose erasure probability is 1, so the channel capacity is equal to zero and the multiplicative gap to capacity is meaningless. This prevents a direct use of Theorem 3.1; however, the continuation of the proof follows similar lines in the proof of Theorem 3.1.

Let x^* denote the unique solution in $[0, 1]$ of the equation

$$1 - \sqrt{1 - \frac{P_{\mathrm{b}}}{p}} = \widetilde{L}(1 - \widetilde{\rho}\,(1 - x^*)). \tag{35}$$

From (28), (31) and (34), a necessary condition for achieving a bit erasure probability P_{b} of the systematic bits is that

$$\widetilde{\lambda}\big(1 - \widetilde{\rho}(1 - x)\big) < x\ , \quad \forall x \in (x^*, 1]\ . \tag{36}$$

In the limit where the fixed point of the iterative decoding process is attained, the inequalities in (28), (31) and (34) are replaced by equalities; hence, (36) also forms a sufficient condition. Analogously to the case of LDPC ensembles, as in the proof of Theorem 3.1, we define the functions

$$\widetilde{c}(x) = 1 - \widetilde{\rho}(1 - x) \quad \text{and} \quad v(x) = \widetilde{\lambda}^{-1}(x)\ . \tag{37}$$

Due to the monotonicity of $\widetilde{\lambda}$ in $[0, 1]$, the necessary and sufficient condition for attaining an erasure probability P_{b} of the systematic bits in (36) can be rewritten as

$$\widetilde{c}(x) < \widetilde{v}(x)\ , \quad \forall x \in (x^*, 1]\ .$$

Since we assume that the sequence of ensembles asymptotically achieves vanishing bit erasure probability under message-passing decoding, it follows that

$$\widetilde{c}(x) < \widetilde{v}(x)\,, \quad \forall x \in (0,1]\,.$$

The next step in the proof is calculating the area of the domain bounded by the curves $\widetilde{c}(x)$ and $\widetilde{v}(x)$. This is done in the following lemma which is analogous to Lemma 4.1.

Lemma 4.4:

$$\int_0^1 (\widetilde{v}(x) - \widetilde{c}(x))\mathrm{d}x = \frac{C-R}{(1-R)\,a_{\mathrm{R}}} \tag{38}$$

where $\widetilde{v}$ and $\widetilde{c}$ are introduced in (37), $C = 1-p$ is the capacity of the BEC, R is the design rate of the systematic ARA ensemble, and a_{R} is defined in (4) and it designates the average degree of the 'parity-check 2' nodes when the edges that are connected to the 'code bit' nodes are ignored.

Proof: From (37)

$$\begin{aligned}
&\int_0^1 (\widetilde{v}(x) - \widetilde{c}(x))\mathrm{d}x \\
&= \int_0^1 \widetilde{\lambda}^{-1}(x)\mathrm{d}x - 1 + \int_0^1 \widetilde{\rho}(1-x)\mathrm{d}x \\
&= \left(1 - \int_0^1 \widetilde{\lambda}(x)\mathrm{d}x\right) - 1 + \int_0^1 \widetilde{\rho}(x)\mathrm{d}x \\
&= \int_0^1 \widetilde{\rho}(x)\mathrm{d}x - \int_0^1 \widetilde{\lambda}(x)\mathrm{d}x
\end{aligned} \tag{39}$$

where the second equality is obtained via integration by parts (note that $\widetilde{\lambda}(0) = 0$ and $\widetilde{\lambda}(1) = 1$). From (32), we get

$$\int_0^1 \widetilde{\lambda}(x)\mathrm{d}x = \frac{1}{\widetilde{L}'(1)} = \frac{p}{L'(1)} = \frac{p}{a_{\mathrm{L}}} \tag{40}$$

(see also [21, Eq. (23)]) where a_{L} is defined in (4), and it designates the average degree of the 'punctured bit' nodes in Fig. 1 when the edges that are connected to the 'parity-check 1' nodes are ignored. Similarly, (33) gives

$$\int_0^1 \widetilde{\rho}(x)\mathrm{d}x = \frac{1}{\widetilde{R}'(1)} = \frac{1-p}{R'(1)} = \frac{1-p}{a_{\mathrm{R}}} \tag{41}$$

(see also [21, Eq. (24)]). Substituting (40) and (41) into (39) gives

$$\begin{aligned}
&\int_0^1 (\widetilde{v}(x) - \widetilde{c}(x))\mathrm{d}x \\
&= \frac{1-p}{a_{\mathrm{R}}} - \frac{p}{a_{\mathrm{L}}} \\
&\stackrel{(a)}{=} \frac{1}{a_{\mathrm{R}}}\left[1 - p\underbrace{\left(\frac{a_{\mathrm{L}} + a_{\mathrm{R}}}{a_{\mathrm{L}}}\right)}_{\frac{1}{1-R}}\right] \\
&= \frac{1}{a_{\mathrm{R}}}\,\frac{1-R-p}{1-R} \\
&= \frac{C-R}{(1-R)\,a_{\mathrm{R}}}
\end{aligned} \tag{42}$$

where (a) follows since the design rate of the systematic ARA ensemble is given by $R = \frac{a_{\mathrm{R}}}{a_{\mathrm{L}}+a_{\mathrm{R}}}$ (see Fig. 1). ∎

To continue the proof, we consider a plot similar to the one in Figure 2 with the exception that $c(x)$ and $v(x)$ are replaced by $\widetilde{c}(x)$ and $\widetilde{v}(x)$, respectively. Note that in this case the horizontal line h_0 is reduced to the point $(1,1)$. Consider the two sets of gray-shaded right-angled triangles. The triangles shaded in dark gray are defined so that the height of triangle number i (counting from right to left and starting at zero) is the vertical line v_i and the slope of their hypotenuse is equal to $\widetilde{c}'(0) = \widetilde{\rho}'(1)$. Since $\widetilde{c}(x)$ is concave, these triangles form a subset of the domain bounded by the curves $\widetilde{c}(x)$ and $\widetilde{v}(x)$. The area of the i'th triangle in this set is given by

$$A_i = \frac{1}{2}\,|v_i|\,\left(\frac{|v_i|}{\widetilde{\rho}'(1)}\right) = \frac{|v_i|^2}{2\,\widetilde{\rho}'(1)}$$

where $|v_i|$ is the length of v_i. The second set of right-angled triangles, which are shaded in light gray, are also defined so that the height of the i'th triangle (counting from right to left and starting at zero) is the vertical line v_i, but the triangle lies to the left of v_i and the slope of its hypotenuse is equal to

$$\widetilde{v}'(0) = \left(\widetilde{\lambda}^{-1}\right)'(0) = \frac{1}{\widetilde{\lambda}'(0)} = \frac{1}{p^2\lambda'(0)} = \frac{1}{p^2\lambda_2}$$

where the second equality follows since $\widetilde{\lambda}(0) = 0$ and the third equality follows from the definition of $\widetilde{\lambda}$ in (29). Since $\widetilde{\lambda}$ is monotonically increasing and convex over the interval $[0,1]$ and it satisfies $\widetilde{\lambda}(0) = 0$ and $\widetilde{\lambda}(1) = 1$, then it follows that $v(x) = \widetilde{\lambda}^{-1}(x)$ is concave over this interval. Hence, the triangles shaded in light gray also form a subset of the domain bounded by the curves $c(x)$ and $v(x)$. The area of the i'th light-gray triangle is given by

$$B_i = \frac{1}{2}\,|v_i|\,\left(|v_i|\,p^2\lambda_2\right) = \frac{p^2\lambda_2\,|v_i|^2}{2}$$

Applying Lemma 4.4 and the fact that the triangles in both sets do not overlap, we get

$$\frac{C-R}{(1-R)\,a_{\rm R}} \geq \frac{1}{2}\left(\frac{1}{\widetilde{\rho}'(1)} + p^2\lambda_2\right)\sum_{i=0}^{l-1}|v_i|^2 \tag{43}$$

where l is an arbitrary natural number. Since the sequence of ensembles asymptotically achieves vanishing bit erasure probability under iterative message-passing decoding, the stability condition for systematic ARA codes (see (6) or equivalently [21, Eq. (14)]) implies that

$$p^2\lambda_2 \leq \frac{1}{\rho'(1) + \frac{2pR'(1)}{1-p}} = \frac{1}{\widetilde{\rho}'(1)} \tag{44}$$

where the last equality follows from (30). Substituting (44) in (43) gives

$$\frac{C-R}{(1-R)\,a_{\rm R}} \geq p^2\lambda_2 \sum_{i=0}^{l-1}|v_i|^2\,. \tag{45}$$

Let $x^{(l)}$ denote the x value of the left tip of the horizontal line h_l. The value of $x^{(l)}$ satisfies the recursive equation

$$x^{(l+1)} = \widetilde{\lambda}\Big(1 - \widetilde{\rho}(1 - x^{(l)})\Big), \quad \forall\, l \in \mathbb{N} \tag{46}$$

with $x^{(0)} = 1$. As was explained above (immediately following Lemma 4.2), from (28), (31), and the monotonicity of the function $f(x) = \widetilde{\lambda}(1 - \widetilde{\rho}(1-x))$ over the interval $[0,1]$, we get that $x^{(l)} \leq x_1^{(l)}$ for $l \in \mathbb{N}$. The definition of h_l and v_l in Figure 2 implies that

$$1 - \widetilde{\rho}(1 - x^{(l)}) = \widetilde{c}(x^{(l)}) = 1 - \sum_{i=0}^{l}|v_i|\,. \tag{47}$$

Starting from (34) and applying the monotonicity of $\widetilde{L}$ and $\widetilde{\rho}$ gives

$$\begin{aligned}
1 - \sqrt{1 - \frac{P_{\rm b}^{(l-1)}}{p}} &\geq \widetilde{L}\Big(1 - \widetilde{\rho}(1 - x_1^{(l-1)})\Big) \\
&\geq \widetilde{L}\left(1 - \widetilde{\rho}(1 - x^{(l-1)})\right) \\
&= \widetilde{L}\left(1 - \sum_{i=0}^{l-1}|v_i|\right)
\end{aligned}$$

where the last equality follows from (47). Since $\widetilde{L}$ is strictly monotonically increasing in $[0, 1]$, then

$$\sum_{i=0}^{l-1} |v_i| \geq 1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}^{(l-1)}}{p}}\right). \tag{48}$$

Applying the Cauchy-Schwartz inequality (as in (24)) to the RHS of (45), we get

$$\begin{aligned} \frac{C-R}{(1-R)\,a_{\text{R}}} &\geq p^2\lambda_2 \sum_{i=0}^{l-1} |v_i|^2 \\ &\geq \frac{p^2\lambda_2}{l}\left(\sum_{i=0}^{l-1}|v_i|\right)^2 \\ &\geq \frac{p^2\lambda_2}{l}\left(1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}^{(l-1)}}{p}}\right)\right)^2 \end{aligned}$$

where the last inequality follows from (48). Since the design rate R is assumed to be a fraction $1-\varepsilon$ of the capacity of the BEC, the above inequality gives

$$C\varepsilon \geq \frac{p^2\lambda_2\,(1-R)\,a_{\text{R}}\,\left(1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}^{(l-1)}}{p}}\right)\right)^2}{l}$$

where l is an arbitrary natural number. Let l designate the number of iterations required to achieve an average bit erasure probability P_{b} of the systematic bits, i.e., l is the smallest integer which satisfies $P_{\text{b}}^{(l-1)} \leq P_{\text{b}}$ (since we start counting the iterations at $l = 0$). Note that l is deterministic since it refers to the smallest number of iterations required to achieve a desired average bit erasure probability over the ensemble. From the inequality above and the monotonicity of $\widetilde{L}$, we obtain that

$$C\varepsilon \geq \frac{p^2\lambda_2\,(1-R)\,a_{\text{R}}\,\left(1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}}{p}}\right)\right)^2}{l}$$

which provides a lower bound on the number of iterations of the form

$$\begin{aligned} l &\geq \frac{p^2\lambda_2(1-R)\,a_{\text{R}}\,\left(1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}}{p}}\right)\right)^2}{C\varepsilon} \\ &= \frac{p^2\lambda_2(1-\varepsilon)\,a_{\text{L}}\,\left(1 - \widetilde{L}^{-1}\left(1 - \sqrt{1 - \frac{P_{\text{b}}}{p}}\right)\right)^2}{\varepsilon} \end{aligned} \tag{49}$$

where the last equality follows since $\frac{a_{\text{R}}}{a_{\text{L}}} = \frac{R}{1-R}$ (see Fig. 1) and $R = (1-\varepsilon)C$. To continue the proof, we derive a lower bound on $1 - \widetilde{L}^{-1}(x)$. Following the same steps which lead to (26) gives the inequality

$$1 - \widetilde{L}^{-1}(x) \geq 1 - \sqrt{\frac{x}{\widetilde{L}_2}}\,, \quad \forall\, x \geq 0 \tag{50}$$

where (32) implies that

$$\widetilde{L}_2 = \frac{\widetilde{L}''(0)}{2} = \frac{p\,L''(0)}{2} = p\,L_2\,. \tag{51}$$

Under the assumption that $1 - \sqrt{1 - \frac{P_{\text{b}}}{p}} < p\,L_2$, substituting (50) and (51) in (49) gives

$$l \geq \frac{p\lambda_2(1-\varepsilon)\,a_{\text{L}}\,\left(\sqrt{p\,L_2} - \sqrt{1 - \sqrt{1 - \frac{P_{\text{b}}}{p}}}\right)^2}{L_2\,\varepsilon}\,. \tag{52}$$

Finally, the lower bound on the number of iterations in (8) follows from (52) by substituting $L_2 = \frac{\lambda_2\,a_{\text{L}}}{2}$.

Considering the case where $P_{\rm b} \to 0$ on both sides of (8) gives

$$l(\varepsilon, p, P_{\rm b} \to 0) \geq 2p^2\,(1-\varepsilon)\,\frac{L_2(\varepsilon)}{\varepsilon}\,.$$

V. Summary and Conclusions

In this paper, we consider the number of iterations which is required for successful message-passing decoding of code ensembles defined on graphs. In the considered setting, we let the block length of these ensembles tend to infinity, and the transmission takes place over a binary erasure channel (BEC).

In order to study the decoding complexity of these code ensembles under iterative decoding, one needs also to take into account the graphical complexity of the Tanner graphs of these code ensembles. For the BEC, this graphical complexity is closely related to the total number of operations performed by the iterative decoder. For various families of code ensembles, Table I compares the number of iterations and the graphical complexity which are required to achieve a given fraction $1-\varepsilon$ (where ε can be made arbitrarily small) of the capacity of a BEC with vanishing bit erasure probability. The results in Table I are based on lower bounds and some achievability results which are related to the graphical complexity of various families of code ensembles defined on graphs (see [20], [21], [25], [26]); the results related to the number of iterations are based on the lower bounds derived in this paper.

Code family	Number of decoding iterations as function of ε	Graphical complexity as function of ε
LDPC	$\Omega\left(\frac{1}{\varepsilon}\right)$ (Theorem 3.1)	$\Theta\left(\ln\frac{1}{\varepsilon}\right)$ [25, Theorems 2.1 and 2.3]
Systematic IRA	$\Omega\left(\frac{1}{\varepsilon}\right)$ (Theorem 3.3)	$\Theta\left(\ln\frac{1}{\varepsilon}\right)$ [26, Theorems 1 and 2]
Non-systematic IRA	$\Omega\left(\frac{1}{\varepsilon}\right)$ (Theorem 3.3)	$\Theta(1)$ [20]
Systematic ARA	$\Omega\left(\frac{1}{\varepsilon}\right)$ (Theorem 3.2)	$\Theta(1)$ [21]

TABLE I
NUMBER OF ITERATIONS AND GRAPHICAL COMPLEXITY REQUIRED TO ACHIEVE A FRACTION $1-\varepsilon$ OF THE CAPACITY OF A BEC WITH VANISHING BIT ERASURE PROBABILITY UNDER ITERATIVE MESSAGE-PASSING DECODING.

Theorems 3.1–3.3 demonstrate that for various attractive families of code ensembles (including low-density parity-check (LDPC) codes, systematic and non-systematic irregular repeat-accumulate (IRA) codes, and accumulate-repeat-accumulate (ARA) codes), the number of iterations which is required to achieve a desired bit erasure probability scales at least like the inverse of the gap between the channel capacity and the design rate of the ensemble. This conclusion holds provided that the fraction of degree-2 variable nodes in the Tanner graph does not tend to zero as the gap to capacity vanishes.

When the graphical complexity of these families of ensembles is considered, the results are less homogenous. More explicitly, assume a sequence of LDPC codes (or ensembles) whose block length tends to infinity, and consider the case where their transmission takes place over a memoryless binary-input output-symmetric channel. It follows from [25, Theorem 2.1] that if a fraction $1-\varepsilon$ of the capacity of this channel is achieved with vanishing bit error (erasure) probability under ML decoding (or any sub-optimal decoding algorithm), then the graphical complexity of an arbitrary representation of the codes using bipartite graphs scales at least like $\ln\frac{1}{\varepsilon}$. For systematic IRA codes which are transmitted over the BEC and decoded by a standard iterative message-passing decoder, a similar result on their graphical complexity is obtained in [26, Theorem 1]. In [25, Theorem 2.3], the lower bound on the graphical complexity of LDPC ensembles is achieved for the BEC (up to a small additive constant), even under iterative message-passing decoding, by the right-regular LDPC ensembles of Shokrollahi [29]. Similarly, [26, Theorem 2] presents an achievability result of this form for ensembles of systematic IRA codes transmitted over the BEC; the graphical complexity of these ensembles scales logarithmically with $\frac{1}{\varepsilon}$. For ensembles of non-systematic IRA and systematic ARA codes, however, the addition of state nodes in their standard representation by Tanner graphs allows to achieve an improved tradeoff between the gap to capacity and the graphical complexity; suitable constructions of such ensembles enable to approach the capacity of the BEC with vanishing bit erasure probability under iterative decoding while maintaining *a bounded graphical complexity* (see [20] and [21]). We note that the ensembles in [21] have the additional advantage of being systematic, which allows a simple decoding of the information bits.

The lower bounds on the number of iterations in Theorems 3.1–3.3 become trivial when the fraction of degree-2 variable nodes vanishes. As noted in Discussion 3.2, for all known capacity-approaching sequences of LDPC ensembles, this fraction tends to $\frac{1}{2}$ as the gap to capacity vanishes. For some ensembles of capacity approaching systematic ARA codes presented in [21], the fraction of degree-2 'punctured bit' nodes (as introduced in Fig. 1) is defined to be zero (see [21, Table I]). For these ensembles, the lower bound on the required number of iterations in Theorem 3.2 is ineffective. However, this is mainly a result of our focus on the derivation of simple lower bounds on the number of iterations which do not depend on the full characterization of the degree distributions of the code ensembles. Following the proofs of Theorems 3.1 and 3.2, and focusing on the case where the fraction of degree-2 variable nodes vanishes, it is possible to derive lower bounds on the number iterations which are not trivial even in this case; these bounds, however, require the knowledge of the entire degree distribution of the examined ensembles.

The simple lower bounds on the number of iterations of graph-based ensembles, as derived in this paper, scale like the inverse of the gap in rate to capacity and also depend on the target bit erasure probability. The behavior of these lower bounds matches well with the experimental results and the conjectures on the number of iterations and complexity, as provided by Khandekar and McEliece (see [11], [12] and [16]). In [12, Theorem 3.5], it was stated that for LDPC and IRA ensembles which achieve a fraction $1-\varepsilon$ of the channel capacity of a BEC with a target bit erasure probability of $P_{\rm b}$ under iterative message-passing decoding, the number of iterations grows like $O\left(\frac{1}{\varepsilon}\right)$. In light of the outline of the proof of this statement, as suggested in [12, p. 71], it implicitly assumes that the flatness condition is satisfied for these code ensembles and also that the target bit erasure probability vanishes; under these assumptions, the reasoning suggested by Khandekar in [12, Section 3.6] serves to support the behavior of the simple and rigorous lower bounds which are derived in this paper.

The matching condition for generalized extrinsic information transfer (GEXIT) curves serves to conjecture in [17, Section XI] that the number of iterations scales like the inverse of the achievable gap in rate to capacity (see also [18, p. 92]); this conjecture refers to LDPC ensembles whose transmission takes place over a general memoryless binary-input output-symmetric (MBIOS) channel. Focusing on the BEC, the derivation of the lower bounds on the number of iterations (see Section IV) makes the heuristic reasoning of this scaling rigorous. It also extends the bounds to various graph-based code ensembles (e.g., IRA and ARA ensembles) under iterative message-passing decoding, and makes them universal for the BEC in the sense that they are expressed in terms of some basic parameters of the ensembles which include the fraction of degree-2 variable nodes, the target bit erasure probability and the asymptotic gap between the channel capacity and the design rate of the ensemble (but the bounds here do not depend explicitly on the degree distributions of the code ensembles). An interesting and challenging direction which calls for further research is to extend these lower bounds on the number of iterations for general MBIOS channels; as suggested in [17, Section XI], a consequence of the matching condition for GEXIT curves has the potential to lead to such lower bounds on the number of iterations which also scale like the inverse of the gap to capacity for general MBIOS channels.

Acknowledgment

This work was initiated during a visit of the first author at EPFL in Lausanne, Switzerland, and it benefited from a short unpublished write-up which was jointly written by S. Dusad, C. Measson, A. Montanari and R. Urbanke. This included preliminary steps towards the derivation of a lower bound on the number of iterations for LDPC ensembles. The first author also wishes to acknowledge H. D. Pfister for various discussions prior to this work on accumulate-repeat-accumulate codes; the 'graph reduction' principle presented in [21] for the binary erasure channel was helpful in the derivation of Theorems 3.2 and 3.3.

APPENDICES

APPENDIX I. PROOF OF PROPOSITION 3.1

We begin the proof by considering an iterative decoder of systematic ARA codes by viewing them as interleaved and serially concatenated codes. The outer code of the systematic ARA code consists of the first accumulator which operates on the systematic bits (see the upper zigzag in Fig. 1), followed by the irregular repetition code. The inner code consists of the irregular SPC code, followed by the second accumulator (see the lower zigzag in Fig. 1). These two constituent codes are joined by an interleaver which permutes the repeated bits at the output of the outer code before they are used as input to the inner encoder; for the considered ARA ensemble, we assume that the interleaver is chosen uniformly at random over all interleavers of the appropriate length. The turbo-like decoding algorithm is based on iterating extrinsic information between bitwise MAP decoders of the two constituent codes (see e.g., [7]). Each decoding iteration begins with an extrinsic bitwise MAP decoding for each non-systematic output bit of the outer code (these are the bits which serve as input to the inner code) based on the information regarding these bits received from the extrinsic bitwise MAP decoder of the inner code in the previous iteration and the information on the systematic bits received from the communication channel. In the second stage of the iteration, this information is passed from the outer decoder to an extrinsic bitwise MAP decoder of the inner code and is used as a-priori knowledge for decoding the input bits of the inner code. A Tanner graph for turbo-like decoding of systematic ARA codes is presented in Figure 3. Considering the asymptotic case where the block length tends to infinity, we denote the probability of erasure messages from the outer decoder to the inner decoder and vice versa at the l'th decoding iteration by $x_0^{(l)}$ and $x_1^{(l)}$, respectively. Keeping in line with the notation in the proofs of Theorems 3.1 and 3.2, we begin counting the iterations at $l = 0$. Since there is no a-priori information regarding the non-systematic output bits of the outer decoder (which are permuted to form the input bits of the inner decoder, as shown in Fig. 3) we have

$$x_0^{(-1)} = x_1^{(-1)} = 1. \tag{I.1}$$

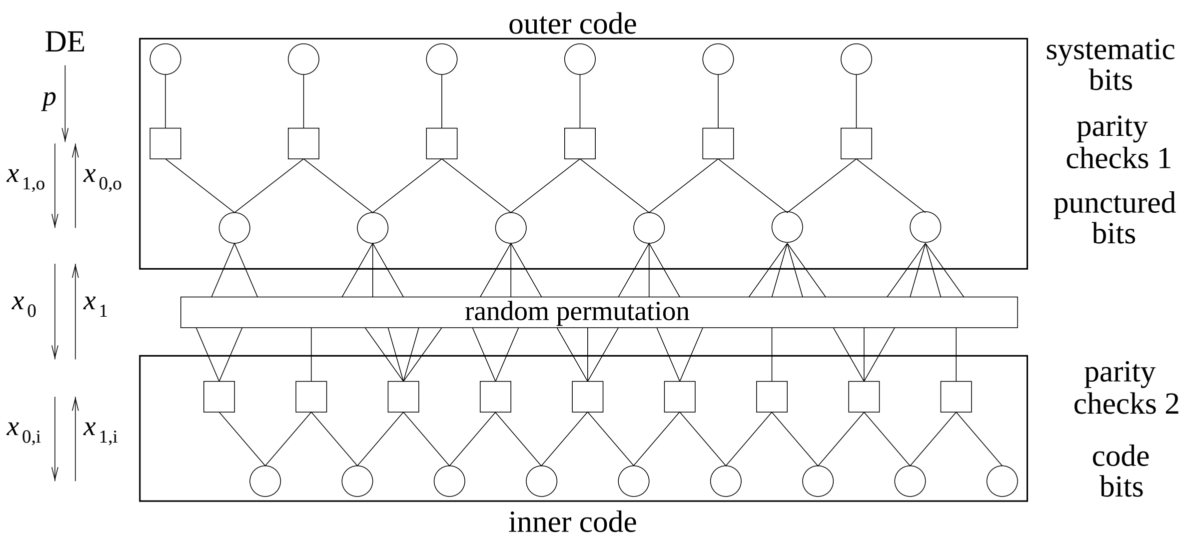

Fig. 3. Tanner graph of a systematic accumulate-repeat-accumulate (ARA) code for turbo-like decoding as an interleaved and serially concatenated code.

We now turn to calculate the erasure probability $x_0^{(l)}$ in an extrinsic bitwise MAP decoding of non-systematic output bits of the outer code, given that the a-priori erasure probability of these bits is $x_1^{(l-1)}$. To this end, we consider the Tanner graph of the outer code, shown in the top box of Figure 3. We note that this Tanner graph contains no cycles, and therefore bitwise MAP decoding of this code can be performed by using the standard iterative message-passing decoding algorithm until a fixed-point is reached. In such a decoder which operates on the Tanner graph of the outer code, messages are transferred between the 'punctured bit' and the 'parity-check 1' nodes of the graph. Let us denote by $x_{0,\text{o}}(x)$ the probability of erasure in messages from the 'punctured bit' nodes to the 'parity-check 1' nodes at the fixed point of the iterative decoding algorithm, when the a-priori erasure probability of the output bits is x. Similarly, we denote by $x_{1,\text{o}}(x)$ the erasure probability in messages from the

’parity-check 1’ nodes to the ’punctured bit’ nodes at the fixed point, where x is the a-priori erasure probability of the non-systematic output bits. Based on the structure of the Tanner graph, we have

$$x_{0,\mathrm{o}}(x) = x_{1,\mathrm{o}}(x) \cdot L(x) \tag{I.2}$$

and

$$x_{1,\mathrm{o}}(x) = 1 - (1-p)\big(1 - x_{0,\mathrm{o}}(x)\big) \tag{I.3}$$

where L is defined in (1) and it forms the degree distribution of the ‘punctured bit’ nodes from the node perspective, and p denotes the erasure probability of the BEC. Substituting (I.2) into (I.3) gives

$$x_{1,\mathrm{o}}(x) = \frac{p}{1-(1-p)L(x)}\,. \tag{I.4}$$

Therefore, the structure of the Tanner graph of the outer code implies that the erasure probability $x_0^{(l)}$ in messages from the outer decoder to the inner decoder at iteration number l of the turbo-like decoding algorithm is given by

$$\begin{aligned} x_0^{(l)} &= \Big(x_{1,\mathrm{o}}\big(x_1^{(l-1)}\big)\Big)^2 \lambda\big(x_1^{(l-1)}\big) \\ &= \left(\frac{p}{1-(1-p)L\big(x_1^{(l-1)}\big)}\right)^2 \lambda\big(x_1^{(l-1)}\big) \\ &= \widetilde{\lambda}\big(x_1^{(l-1)}\big) \end{aligned} \tag{I.5}$$

where the second equality relies on (I.4), and $\widetilde{\lambda}$ is introduced in (29)). We now employ a similar technique to calculate the erasure probability $x_1^{(l)}$ in an extrinsic bitwise MAP decoding of input bits of the inner code, given that the a-priori erasure probability of these bits is $x_0^{(l)}$. Since the Tanner of the inner code is also cycle-free (see the lower box in Figure 3), extrinsic bitwise MAP decoding can be done by using the iterative decoder operating on the Tanner graph of the inner code. We denote by $x_{0,\mathrm{i}}(x)$ the erasure probability of messages from the ‘parity check 2’ nodes to the ‘code bit’ nodes at the fixed point of the iterative decoding algorithm when x is the a-priori erasure probability of the input bits. Similarly, $x_{1,\mathrm{i}}(x)$ designates the erasure probability of messages from the ‘code bit’ nodes to the ‘parity check 2’ nodes at the fixed point of the decoding algorithm, when x is the a-priori erasure probability of the input bits. The structure of the Tanner graph implies that

$$x_{0,\mathrm{i}}(x) = 1 - \big(1 - x_{1,\mathrm{i}}(x)\big)R(1-x) \tag{I.6}$$

and

$$x_{1,\mathrm{i}}(x) = p\, x_{0,\mathrm{i}}(x) \tag{I.7}$$

where R is defined in (2). Substituting (I.6) into (I.7) gives

$$x_{1,\mathrm{i}}(x) = \frac{p\big(1 - R(1-x)\big)}{1 - p\,R(1-x)}\,. \tag{I.8}$$

Therefore, the erasure probability $x_1^{(l)}$ in messages from the inner decoder to the outer decoder at iteration number l of the turbo-like decoding algorithm is given by

$$\begin{aligned} x_1^{(l)} &= 1 - \Big(1 - x_{1,\mathrm{i}}\big(x_0^{(l)}\big)\Big)^2 \rho\big(1 - x_0^{(l)}\big) \\ &= 1 - \left(\frac{1-p}{1 - p\,R\big(1-x_0^{(l)}\big)}\right)^2 \rho\big(1 - x_0^{(l)}\big) \\ &= 1 - \widetilde{\rho}\big(1 - x_0^{(l)}\big) \end{aligned} \tag{I.9}$$

where the second equality relies on (I.8), and $\widetilde{\rho}$ is the tilted degree distribution resulting from graph reduction (see (30)). Combining (I.1), (I.5) and (I.9) gives

$$\begin{aligned} x_0^{(0)} &= \widetilde{\lambda}\big(x_1^{(-1)}\big) = \widetilde{\lambda}(1) = 1\,, \\ x_0^{(l)} &= \widetilde{\lambda}\Big(1 - \widetilde{\rho}\big(1 - x_0^{(l-1)}\big)\Big)\,, \qquad l \in \mathbb{N}\,. \end{aligned} \tag{I.10}$$

Observing the proof of Theorem 3.2, we note that $x_0^{(l)} = x^{(l)}$ for all $l = 0, 1, \ldots$, where is the $x^{(l)}$ value at the left tip of the horizontal line h_l in Figure 2 (see Eq. (46) on page 15).

Let $P_\text{b}^{(l)}$ designate the average erasure probability of the systematic bits at the end of the l'th iteration of the turbo-like decoder. From the definition of the turbo-like decoding algorithm, $P_\text{b}^{(l)}$ is the erasure probability of bitwise MAP decoding for the input bits to the outer code, given that the a-priori erasure probability of the output bits of this code is given by $x_1^{(l)}$. Based of the structure of the Tanner graph of the outer code in Figure 3, we get

$$P_\text{b}^{(l)} = p\left[1 - \left(1 - x_{0,\text{o}}(x_1^{(l)})\right)^2\right] \tag{I.11}$$

where $x_{0,\text{o}}(x)$ in the fixed point erasure probability of messages from the 'punctured bit' nodes to the 'parity-check 1' nodes in the case that the a-priori erasure probability of the non-systematic output bits of the code is x. Substituting (I.3) in (I.2) gives

$$x_{0,\text{o}}(x) = \frac{p\,L(x)}{1-(1-p)L(x)}\,.$$

Substituting the above equality into (I.11), we have

$$\begin{aligned} P_\text{b}^{(l)} &= p\left[1 - \left(1 - \widetilde{L}(x_1^{(l)})\right)^2\right] \\ &= p\left[1 - \left(1 - \widetilde{L}\Big(1 - \widetilde{\rho}(1 - x_0^{(l)})\Big)\right)^2\right] \end{aligned}$$

where the first equality follows from the definition of $\widetilde{L}$ in (32) and the second equality relies on (I.9). Using simple algebra

$$1 - \sqrt{1 - \frac{P_\text{b}^{(l)}}{p}} = \widetilde{L}\Big(1 - \widetilde{\rho}(1 - x_0^{(l)})\Big)\,. \tag{I.12}$$

Hence, the lower bound on the average erasure probability of the systematic bits at the end of the l'th iteration of the standard iterative decoder for ARA codes in Lemma 4.3 is satisfied (with equality) also for the turbo-like decoder.

Let l designate the required number of iterations for the turbo-like decoder to achieve an average erasure probability P_b of the systematic bits. Eq. (I.12) implies that l is the smallest natural number which satisfies

$$1 - \sqrt{1 - \frac{P_\text{b}}{p}} \geq \widetilde{L}\Big(1 - \widetilde{\rho}(1 - x_0^{(l-1)})\Big)\,.$$

However, this is exactly the quantity for which we calculated the lower bound in the proof of Theorem 3.2 (see Lemmas 4.2 and 4.3, and Eq. (31)). Therefore, the lower bound in Theorem 3.2 also holds when the considered turbo-like algorithm is used to decode the systematic ARA codes as interleaved and serially concatenated codes.

APPENDIX II.

A. *Proof of Lemma 4.2*

The proof of Lemma 4.2 is based on the DE equations in (5) for systematic ARA ensembles. From the DE equations for $x_2^{(l)}$ and $x_3^{(l)}$, we have

$$\begin{aligned} x_3^{(l)} &= p\left[1 - R\left(1 - x_1^{(l)}\right)\left(1 - x_3^{(l-1)}\right)\right] \\ &\geq p\left[1 - R\left(1 - x_1^{(l)}\right)\left(1 - x_3^{(l)}\right)\right] \end{aligned}$$

where the inequality follows since the decoding process does not add erasures, so $x_i^{(l)}$ is monotonically decreasing with l (for $i = 0, 1, \ldots, 5$). This gives

$$1 - x_3^{(l)} \leq \frac{1-p}{1 - pR\left(1 - x_1^{(l)}\right)}. \tag{II.1}$$

Substituting (II.1) into the DE equation for $x_4^{(l)}$ (see (5)) gives

$$\begin{aligned} x_4^{(l)} &\geq 1 - \left(\frac{1-p}{1-pR\left(1-x_1^{(l)}\right)}\right)^2 \rho\left(1-x_1^{(l)}\right) \\ &= 1-\widetilde{\rho}\left(1-x_1^{(l)}\right) \end{aligned} \tag{II.2}$$

where $\widetilde{\rho}$ is defined in (30). From (5), we get

$$\begin{aligned} x_5^{(l)} &= \left[1-\left(1-x_5^{(l-1)}\right)(1-p)\right] L\left(x_4^{(l)}\right) \\ &\geq \left[1-\left(1-x_5^{(l)}\right)(1-p)\right] L\left(x_4^{(l)}\right) \end{aligned}$$

and solving for $1-x_5^{(l)}$ gives

$$1-x_5^{(l)} \leq \frac{1-L\left(x_4^{(l)}\right)}{1-(1-p)L\left(x_4^{(l)}\right)}\,. \tag{II.3}$$

Substituting (II.3) into the DE equation for $x_0^{(l)}$ in (5) gives

$$x_0^{(l)} \geq \frac{p}{1-(1-p)L\left(x_4^{(l-1)}\right)}\,.$$

Substituting this inequality into the DE equation for $x_1^{(l)}$ gives

$$x_1^{(l)} \geq \widetilde{\lambda}\left(x_4^{(l-1)}\right) \tag{II.4}$$

where $\widetilde{\lambda}$ is defined in (29). Finally (28) follows from (II.2), (II.4), and the monotonicity of $\widetilde{\lambda}$ over the interval $[0,1]$.

B. Proof of Lemma 4.3

From Fig. 1) and the DE equation for $x_5^{(l)}$ in (5)

$$\begin{aligned} P_{\mathrm{b}}^{(l)} &= p\left[1-\left(1-x_5^{(l)}\right)^2\right] \\ &= p\left[1-\left(1-x_0^{(l)}L\left(x_4^{(l)}\right)\right)^2\right]\,. \end{aligned} \tag{II.5}$$

The DE equation (5) for $x_1^{(l)}$ and (29) imply that

$$\begin{aligned} \left(x_0^{(l)}\right)^2 &= \frac{x_1^{(l)}\,p^2}{\widetilde{\lambda}\left(x_4^{(l-1)}\right)\,\left[1-(1-p)\,L\left(x_4^{(l-1)}\right)\right]^2} \\ &\geq \left(\frac{p}{1-(1-p)\,L\left(x_4^{(l-1)}\right)}\right)^2 \end{aligned}$$

where the last inequality follows from (II.4), and then

$$x_0^{(l)} \geq \frac{p}{1-(1-p)\,L\left(x_4^{(l-1)}\right)}\,. \tag{II.6}$$

Substituting (II.6) in (II.5), we get

$$\begin{aligned} P_{\mathrm{b}}^{(l)} &\geq p\left[1-\left(1-\frac{p\,L\left(x_4^{(l)}\right)}{1-(1-p)\,L\left(x_4^{(l-1)}\right)}\right)^2\right] \\ &\geq p\left[1-\left(1-\frac{p\,L\left(x_4^{(l)}\right)}{1-(1-p)\,L\left(x_4^{(l)}\right)}\right)^2\right] \end{aligned} \tag{II.7}$$

which follows since $x_4^{(l)} \leq x_4^{(l-1)}$, and from (32)

$$\begin{aligned} P_{\mathrm{b}}^{(l)} &\geq p\left[1-\left(1-\widetilde{L}\left(x_4^{(l)}\right)\right)^2\right] \\ &\geq p\left\{1-\left[1-\widetilde{L}\left(1-\rho\left(x_1^{(l)}\right)\right)\right]^2\right\} \end{aligned} \tag{II.8}$$

where the last inequality follows from (II.2). Finally, (34) follows directly from (II.8).

REFERENCES

[1] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. on Communications*, vol. 55, no. 4, pp. 692–702, April 2007.

[2] A. Amraoui, A. Montanari and R. Urbanke, "How to find good finite-length codes: from art towards science," *European Trans. on Telecommunications*, vol. 18, no. 5, pp. 491-508, August 2007.

[3] K. Andrews, V. Stanton, S. Dolinar, V. Chen, J. Berner and F. Pollara, "Turbo-decoder implementation for the deep space network," *The Interplanetary Network Progress Report*, IPN Progress Report 42-148, pp. 1–20, October-December 2001.

[4] M. Ardakani, B. Smith, W. Yu and F. R. Kschischang, "Complexity-optimized low-density parity-check codes," *Proceedings of the Forty-Third Annual Allerton Conference on Communication, Control and Computing*, pp. 45–54, Urbana-Champaign, Illinois, USA, September 28–30, 2005.

[5] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Trans. on Information Theory*, vol. 50, no. 11, pp. 2657–2673, November 2004.

[6] O. Barak, D. Burshtein and M. Feder, "Bounds on achievable rates of LDPC codes used over the binary erasure channel," *IEEE Trans. on Information Theory*, vol. 50, no. 10, pp. 2483–2489, October 2004.

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

[8] C. Di, D. Proietti, I. E. Telatar and R. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. on Information Theory*, vol. 48, no. 6, pp. 1570-1579, June 2002.

[9] C. H. Hsu and A. Anastasopoulos, "Capacity-achieving codes for noisy channels with bounded graphical complexity and maximum-likelihood decoding," submitted to *IEEE Trans. on Information Theory*. [Online]. Available: http://www.eecs.umich.edu/~anastas/docs/tit06_ha.pdf.

[10] H. Jin, A. Khandekar and R. J. McEliece, "Irregular repeat-accumulate codes," *Proceedings of the Second International Symposium on Turbo Codes & Related Topics*, pp. 1–8, Brest, France, September 2000.

[11] A. Khandekar and R. J. McEliece, "On the complexity of reliable communication on the erasure channel," *Proceedings 2001 IEEE International Symposium on Information Theory*, p. 1, Washington, D.C., USA, June 2001.

[12] A. Khandekar, *Graph-based Codes and Iterative Decoding*, PhD dissertation, Caltech University, California, USA, June 2002. [Online]. Available: http://etd.caltech.edu/etd/available/etd-06202002-170522.

[13] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi and D. A. Spielman, "Efficient erasure-correcting codes," *IEEE Trans. on Information Theory*, vol. 47, pp. 569–584, February 2001.

[14] M. G. Luby, "LT codes," *Proceedings of the 43rd Symposium on Foundations of Computer Science (FOCS 2002)*, pp. 271–280, Vancouver, BC, Canda, November 2002.

[15] X. Ma and E. Yang, "Low-density parity-check codes with fast decoding convergence speed," *Proceedings 2004 International Symposium on Information Theory*, p. 277, Chicago, IL, USA, June 2004.

[16] R. J. McEliece, "Are turbo-like codes effective on nonstandard channels ?," *IEEE Information Theory Society Newsletter*, vol. 51, no. 4, pp. 1–8, December 2001.

[17] C. Measson, A. Montanari, T. Richardson and R. Urbanke, "The generalized area theorem and some of its consequences," submitted to *IEEE Trans. on Information Theory*, November 2005. [Online]. Available: http://arxiv.org/abs/cs/0511039.

[18] C. Measson, *Conservation Laws for Coding*, PhD thesis, EPFL, Lausanne, Switzerland, March 2006. [Online]. Available: http://lthcwww.epfl.ch/~cyril/research/These_3485_Measson_Book.pdf.

[19] P. Oswald and A. Shokrollahi, "Capacity-achieving sequences for the erasure channel," *IEEE Trans. on Information Theory*, vol. 48, pp. 3017–3028, December 2002.

[20] H. D. Pfister, I. Sason and R. Urbanke, "Capacity-achieving ensembles for the binary erasure channel with bounded complexity," *IEEE Trans. on Information Theory*, vol. 51, pp. 2362–2379, July 2005.

[21] H. D. Pfister and I. Sason, "Accumulate-repeat-accumulate codes: Capacity-achieving ensembles of systematic codes for the erasure channel with bounded complexity," *IEEE Trans. on Information Theory*, vol. 53, no. 6, pp. 2088–2115, June 2007.

[22] T. Richardson and R. Urbanke, *Modern Coding Theory*, Cambridge University Press, 2008. [Online]. Available: http://lthcwww.epfl.ch/mct/index.php.

[23] T. Richardson and R. Urbanke, "Finite-length density evolution and the distribution of the number of iterations for the binary erasure channel," unpublished. [Online]. Available: http://lthcwww.epfl.ch/RiU02.ps.

[24] I. Sason, "On universal properties of capacity-approaching LDPC code ensembles," accepted to *IEEE Trans. on Information Theory*, 2009.

[25] I. Sason and R. Urbanke, "Parity-check density versus performance of binary linear block codes over memoryless symmetric channels," *IEEE Trans. on Information Theory*, vol. 49, pp. 1611–1635, July 2003.

[26] I. Sason and R. Urbanke, "Complexity versus performance of capacity-achieving irregular repeat-accumulate codes on the binary erasure channel," *IEEE Trans. on Information Theory*, vol. 50, no. 6, pp. 1247–1256, June 2004.

[27] R. Y. Shao, S. Lin and M. P. C. Fossorier, “Two simple stopping criteria for turbo decoding,” *IEEE Trans. on Communications*, vol. 47, no. 8, pp. 1117–1120, August 1999.

[28] E. Sharon, S. Litsyn and J. Goldenberg, “Efficient serial message-passing schedules for LDPC decoding,” *IEEE Trans. on Information Theory*, vol. 53, no. 11, pp. 4076–4091, November 2007.

[29] A. Shokrollahi, “Capacity-achieving sequences,” *IMA Volume in Mathematics and its Applications*, vol. 123, pp. 153–166, 2000.

[30] A. Shokrollahi, “Raptor codes,” *IEEE Trans. on Information Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[31] S. ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes,” *IEEE Trans. on Communications*, vol. 49, no. 10, pp. 1727–1737, October 2001.

[32] S. ten Brink, “Code characteristic matching for iterative decoding of serially concatenated codes,” *Annals of Telecommunications*, vol. 56, no. 7–8, pp. 394–408, July-August 2001.

[33] W. Yu, M. Ardakani, B. Smith, and F. R. Kschischang, “Complexity-optimized LDPC codes for Gallager decoding algorithm B,” *Proceedings 2005 IEEE International Symposium on Information Theory*, pp. 1488–1492, Adelaide, Australia, September 2005.

[34] *Big-O notation*. [Online]. Available: http://www.nist.gov/dads/HTML/bigOnotation.html.