LDPC Codes with Local and Global Decoding

Eshed Ram Yuval Cassuto
Andrew and Erna Viterbi Department of Electrical Engineering
Technion – Israel Institute of Technology, Haifa 32000, Israel
E-mails: {s6eshedr@campus, ycassuto@ee}.technion.ac.il

***Abstract*—This paper presents a theoretical study of a new type of LDPC codes that is highly motivated by practical storage applications. LDPCL codes (suffix L represents locality) are LDPC codes that can be decoded either as usual over the full code block, or locally when a smaller sub-block is accessed (to reduce latency). LDPCL codes are designed to maximize the error-correction performance vs. rate in the usual (global) mode, while at the same time providing a certain performance in the local mode. We develop a theoretical framework for the design of LDPCL codes over the binary erasure channel. Our results include generalizing the density-evolution analysis to two dimensions, proving the existence of a decoding threshold and showing how to compute it, and constructing capacity-achieving sequences for any pair of local and global thresholds.**

Proofs and more results are made available at the arXiv (http://arxiv.org/abs/1801.03951).

Keywords: Density evolution, iterative decoding, low-density parity-check codes, multi-block coding.

1. Introduction

Low-density parity-check (LDPC) codes and their low-complexity iterative decoding algorithm [3] are a powerful method to achieve reliable communication and storage with rates that approach Shannon's theoretical limit. When used in data-storage applications, unlike in communications, retransmissions are not possible, and any decoding failure implies data loss; hence strong LDPC codes need to be provisioned for extreme data reliability. Another key feature of modern storage devices is fast access, i.e., low-latency and high-throughput read operations. However, high data reliability forces very large block sizes and high complexity, and thus degrades the device's latency and throughput. This inherent conflict motivates a coding scheme that enables fast read access to small (sub) blocks with modest data protection and low complexity, while in case of failure providing a high data-protection "safety net" in the form of decoding a stronger code over a larger block. Our objective in this paper is to design LDPC codes to operate in such a *multi-block coding scheme*, where error-correction performance (vs. rate) is maximized in both the sub-block and full-block modes.

Formally, in a multi-block coding scheme a code block of length N is divided into M sub-blocks of length n (i.e., $N = Mn$). Each sub-block is a codeword of one code, and the concatenation of the M sub-blocks forms a codeword of another (stronger) code. This paper is the first to design LDPC codes for the multi-block scheme. Earlier work, such as [2] recently and [4], [5], [1] before, addressed the design of Reed-Solomon and related algebraic codes in multi-block schemes. While that prior work attests to the importance of the multi-block scheme, designing LDPC codes for it requires all-new tools and methods. Toward that we define a new type of LDPC codes we call *LDPCL codes*, where the suffix 'L' points to the code's *local* access to its sub-blocks. The LDPCL code is designed in such a way that each of the sub-blocks (of length n) can be decoded independently of the other sub-blocks (local decoding), and in addition the full block of length Mn can be decoded (global decoding) when local decoding fails.

Our theoretical results on analysis and construction of LDPCL codes lie upon the definition of the code through two distinct degree-distribution pairs. The *local degree distribution* specifies the connections between sub-block variable nodes and their local check nodes, while the *joint degree distribution* governs the connection of the global check nodes to variable nodes in the full block. The key challenge is to design the local and joint distributions such that both the local and the global (=local+joint composition) codes perform well. In particular, this requires the generalization of the binary erasure channel density-evolution analysis [12] to two dimensions (2D), where the local and joint dimensions are shown to have inherent asymmetries that need to be addressed to make the analysis work. Prior work related to this generalization are multi-edge type LDPC [12, Chapter 7] and IRA codes [6], in which more than one density are tracked. However, in contrast to IRA codes, the connectivity structure of LDPCL codes does not allow to collapse the analysis to a single density, and a two-dimensional reasoning is required to reach capacity. Moreover, LDPCL codes are application motivated, and thus have a specific structure, in contrast to the full generality of multi-edge type LDPC codes. This enables explicit threshold calculations and constructions of capacity-approaching LDPCL code ensembles. The 2D density-evolution analysis is derived in Section 4, and then used to prove the existence of a decoding threshold for LDPCL code ensembles, which can be calculated numerically. Then Section 5 presents the main result: a construction of local and joint degree-distribution sequences that are capacity achieving for any pair of local and global thresholds. We note two prior works related to our results: [8] proposed global coupling as a similar structure for LDPC codes, but were not interested in local decoding performance and focused on algebraic structured codes over non-binary alphabets; [10] studied a two-layer LDPC framework for a different application, but defined the codes through one big product degree distribution that is not amenable to asymptotic analysis and capacity-achieving constructions.

978-1-5386-4780-6/18/$31.00©2018 IEEE

2. Preliminaries

A. LDPC Codes

A linear block code is an LDPC code if it has at least one parity-check matrix that is sparse. This sparsity enables a low-complexity decoding algorithm. Every parity-check matrix H can be represented by a bipartite graph, called a Tanner graph, with nodes partitioned to variable nodes and check nodes; there exists an edge between check node i and variable node j, if and only if $H_{ij} = 1$. The fraction of variable (resp. check) nodes in a Tanner graph with degree i is denoted by Λ_i (resp. Ω_i), and the fraction of edges connected to variable (resp. check) nodes of degree i is denoted by λ_i (resp. ρ_i); Λ_i and Ω_i are called node-perspective degree distributions, and λ_i and ρ_i are called edge-perspective degree distributions. The degree-distribution polynomials associated with variable nodes in a Tanner graph are given by $\Lambda(x) = \sum_i \Lambda_i x^i$, $\lambda(x) = \sum_i \lambda_i x^{i-1}$, and the check-node polynomials $\Omega(x), \rho(x)$ are defined similarly. The rate of an LDPC code is lower bounded by the design rate, which is given by $R = 1 - \frac{\int_0^1 \rho(t)\mathrm{d}t}{\int_0^1 \lambda(t)\mathrm{d}t}$.

B. The Two-Sided Tanner Graph

We define an LDPCL code of length $N = Mn$ through a two-sided Tanner graph. In this graph, the variable nodes are divided to M disjoint sets of size n each, and the check nodes are divided into two disjoint sets: *local check nodes* and *joint check nodes*. To distinguish between local and joint check nodes, the former are drawn to the right of the variable nodes and the latter are drawn to the left (hence its name: two-sided Tanner graph). The graph construction is constrained such that each local check node is connected only to variable nodes that are in the same sub-block of length n; the joint check-node connections have no connection constraints. The set of edges in the graph is partitioned into two sets as well: edges connecting variable nodes to local check nodes (i.e., the right side of the graph) are called *local edges* and edges connecting variable nodes to joint check nodes are called *joint edges*. Finally, the local (resp. joint) degree of a variable node is the number of local (resp. joint) edges emanating from it.

We denote by $\Lambda_{L,i}$ the fraction of variable nodes with local degree i, and by $\Omega_{L,i}$ the fraction of local check nodes with degree i. Similarly, $\lambda_{L,i}$ designates the fraction of local edges connected to a variable node with local degree i, and $\rho_{L,i}$ designates the fraction of local edges connected to a local check node of degree i. We call $(\Lambda_{L,i}, \Omega_{L,i}, \lambda_{L,i}, \rho_{L,i})$ local degree distributions. Note that we do not distinguish between local degree distributions of different sub-blocks, and we assume that they are the same in all sub-blocks (but the instances drawn from the distributions are in general different between the sub-blocks). The joint degree distributions $(\Lambda_{J,i}, \Omega_{J,i}, \lambda_{J,i}, \rho_{J,i})$ are defined similarly with an important difference. In contrast to Tanner graphs associated to ordinary LDPC codes where the minimal variable-node degree is usually 2 (i.e., $\Lambda_{L,0} = \Lambda_{L,1} = 0$), in the two-sided Tanner graph we allow some variable nodes to have joint degree 0 or 1. The reason for removing this restriction is related to the iterative decoding algorithm, and will become clearer in Sections 4 and 5. In the rest of the paper we will use P_0 to denote the coefficient $\Lambda_{J,0}$, due to its importance.

3. LDPCL Ensembles

The LDPCL ensembles have six parameters: $M, n, \Lambda_L(\cdot), \Lambda_J(\cdot), \Omega_L(\cdot)$, and $\Omega_J(\cdot)$. M is the locality parameter that sets the number of sub-blocks in a code block, n is the sub-block length, and $\Lambda_L(\cdot), \Lambda_J(\cdot), \Omega_L(\cdot), \Omega_J(\cdot)$ are the node-perspective degree-distribution polynomials; this ensemble is denoted by $LDPCL(M, n, \Lambda_L, \Omega_L, \Lambda_J, \Omega_J)$. We can refer to an LDPCL ensemble through its edge-perspective degree-distribution polynomials, and then it is denoted by $LDPCL(M, n, \lambda_L, \rho_L, \lambda_J, \rho_J, P_0)$ (when using the edge-perspective notation, one must specify P_0 as well).

The sampling process from the $LDPCL(M, n, \Lambda_L, \Lambda_J, \Omega_L, \Omega_J)$ ensemble is a as follows. First, M Tanner graphs are sampled independently from the (ordinary) $LDPC(n, \Lambda_L, \Omega_L)$ ensemble. These local graphs are concatenated vertically without inter-connections. Another Tanner graph is then sampled from the $LDPC(Mn, \Lambda_J, \Omega_J)$ ensemble. The latter joint graph is flipped, its Mn variable nodes are randomly permuted (forcing statistical independence between the local and joint degrees of the variable nodes), and merged with the Mn variable nodes of the M local graphs to create a two-sided Tanner graph. The design rate of an $LDPCL(M, n, \Lambda_L, \Lambda_J, \Omega_L, \Omega_J)$ ensemble is given by

$$R(\lambda_L, \rho_L, \lambda_J, \rho_J, P_0) = 1 - \frac{\int_0^1 \rho_L(x)\mathrm{d}x}{\int_0^1 \lambda_L(x)\mathrm{d}x} - \frac{\int_0^1 \rho_J(x)\mathrm{d}x}{\int_0^1 \lambda_J(x)\mathrm{d}x}(1 - P_0). \quad (1)$$

4. Iterative Decoding Asymptotic Analysis

A standard decoding algorithm for ordinary LDPC codes over the BEC is a message-passing algorithm known as the belief-propagation (BP) algorithm; its decoding complexity per iteration is linear in the block length and despite being sub-optimal, there are LDPC ensembles that approach the BEC capacity with BP. To take advantage of the locality structure of the Tanner graphs described above, the suggested decoding algorithm will operate in two modes: local mode and global mode. In the local mode, the decoder tries to decode a sub-block of length n using BP on the local Tanner graph corresponding to the desired sub-block. If the decoder meets a failure criterion (e.g., maximum number of iterations or stable non-zero fraction of erased variable nodes), then it enters the global mode where it tries to decode the entire code block (of length $N = Mn$) using BP on the complete two-sided Tanner graph. If the decoder succeeds, it extracts the wanted sub-block, and if it fails, it declares a decoding failure.

The message scheduling in the global mode is a flooding schedule: in the first step of a global decoding iteration, the variable nodes send messages to the local and joint check nodes in parallel, and in the second step the local and global check nodes send their messages back to the variable nodes. As in the case of ordinary LDPC codes, the messages passed are either erasures or decoded 0/1 bits.

Example 1. *Figure 1 exemplifies the decoding algorithm for an LDPCL code with $M = 2$ and $n = 12$; the desired sub-block is the upper one.*

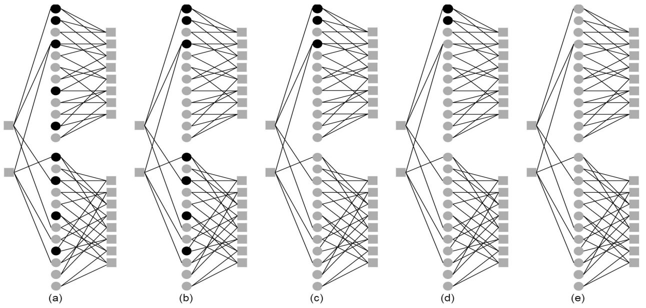

Fig. 1. (a) A two-sided Tanner graph with erased (black) variable nodes; (b) the decoder fails to locally decode the upper sub-block; (c) after one iteration in global mode; (d) after two iterations in global mode; (e) the decoder resolves the desired sub-block.

In the local mode, the analysis of the decoding algorithm is identical to the analysis of ordinary LDPC codes. Specifically, in the limit where $n \to \infty$, there exists a local decoding threshold ϵ_L^* such that if the fraction of erasures ϵ is less than ϵ_L^*, the decoder will resolve the desired sub-block in the local mode with probability converging to 1, and if $\epsilon > \epsilon_L^*$, the decoder will fail in the local mode with probability converging to 1. In contrast to the local mode, in the global mode there is a major difference in the asymptotic analysis of the BP algorithm between LDPC and LDPCL codes. Due to the built-in structure of the LDPCL ensemble, one cannot calculate the global degree-distribution polynomials (i.e., considering an "effective" one-sided Tanner graph ensemble), and use them to find the global BP decoding threshold with known methods of $LDPC$ codes.

A. Decoding in Global Mode

In this sub-section, we analyze the asymptotic performance (as $n \to \infty$) of the BP decoding algorithm over the BEC in the global mode, and derive density-evolution equations for the LDPCL codes.

Theorem 1. *Consider a random element from the $LDPCL(M, n, \Lambda_L, \Lambda_J, \Omega_L, \Omega_J)$ ensemble. Let $x_l(\epsilon)$ and $y_l(\epsilon)$ denote the probability that a local and joint edge, respectively, carries a variable-to-check erasure message after l BP iterations over the $BEC(\epsilon)$ as $n \to \infty$. Then, for every $l \geq 0$,*

$$x_l(\epsilon) = \epsilon\lambda_L\left(1 - \rho_L\left(1 - x_{l-1}(\epsilon)\right)\right) \cdot \Lambda_J\left(1 - \rho_J\left(1 - y_{l-1}(\epsilon)\right)\right), \tag{2a}$$

$$y_l(\epsilon) = \epsilon\Lambda_L\left(1 - \rho_L\left(1 - x_{l-1}(\epsilon)\right)\right) \cdot \lambda_J\left(1 - \rho_J\left(1 - y_{l-1}(\epsilon)\right)\right), \tag{2b}$$

with $x_{-1}(\epsilon) = y_{-1}(\epsilon) = 1$.

We call (2a)-(2b) the 2D density evolution (2D-DE) equations of LDPCL codes over the $BEC(\epsilon)$. To simplify notations, ϵ will be omitted from now on from $x_l(\epsilon)$ and $y_l(\epsilon)$ if it is clear from the context.

Remark 1. *Although x_l and y_l in (2a)-(2b) seem symmetric to each other, it is not necessarily true since we allow variable nodes to have joint degrees 0 or 1, while their local degrees are forced to be greater then 1. This asymmetry has a crucial effect on the global decoding process which is explained and detailed in the following sub-section.*

B. Threshold

In this sub-section we prove that for any given $LDPCL(M, n, \lambda_L, \rho_L, \lambda_J, \rho_J, P_0)$ ensemble, there exists a threshold denoted by ϵ_G^* such that when $n \to \infty$, the BP algorithm will successfully globally decode (for sufficiently large number of iterations) a code block transmitted on the $BEC(\epsilon)$ if and only if $\epsilon < \epsilon_G^*$. A method to calculate this threshold is provided as well.

For $x, y, \epsilon \in [0, 1]$, define

$$f(\epsilon, x, y) = \epsilon\,\lambda_L(1 - \rho_L\,(1 - x))\,\Lambda_J(1 - \rho_J\,(1 - y))\,, \tag{3a}$$

$$g(\epsilon, x, y) = \epsilon\,\Lambda_L(1 - \rho_L\,(1 - x))\,\lambda_J(1 - \rho_J\,(1 - y))\,, \tag{3b}$$

such that (2a)-(2b) can be re-written as

$$x_l = f\left(\epsilon, x_{l-1}, y_{l-1}\right), \; y_l = g\left(\epsilon, x_{l-1}, y_{l-1}\right), \quad l \geq 0. \tag{4}$$

Note that the functions f and g are monotonically non-decreasing in all of their variables, so by mathematical induction, it follows that $x_l(\epsilon)$ and $y_l(\epsilon)$ are monotonically non-increasing in l and monotonically non-decreasing in ϵ. In addition, for every $l \geq 0$, $x_l(0) = y_l(0) = 0$ and $x_l(1) = y_l(1) = 1$. Therefore, the limits $\lim_{l\to\infty} x_l(\epsilon)$ and $\lim_{l\to\infty} y_l(\epsilon)$ exist, and we can define a BP global decoding threshold by

$$\epsilon_G^* = \sup\left\{\epsilon \in [0,1] \colon \lim_{l\to\infty} y_l(\epsilon) = \lim_{l\to\infty} x_l(\epsilon) = 0\right\}. \tag{5}$$

Definition 1. *Let $\epsilon \in (0, 1)$. We say that $(x, y) \in [0, 1]^2$ is an (f, g)-fixed point if*

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f(\epsilon, x, y) \\ g(\epsilon, x, y) \end{pmatrix}. \tag{6}$$

Clearly, for every $\epsilon \in (0, 1)$, $(x, y) = (0, 0)$ is a trivial (f, g)-fixed point. However, it is not clear yet if there exists a non-trivial (f, g)-fixed point. In particular, we ask: for which choices of $\epsilon \in (0, 1)$, $\lambda_L, \rho_L, \lambda_J, \rho_J$ and P_0 does there exist a non-trivial (f, g)-fixed point? The following lemmas will help answering this question.

Lemma 1. *Let $\epsilon \in (0, 1)$, and let $(x, y) \in [0, 1]^2$ be an (f, g)-fixed point. Then,*

1) *$x = 0$ implies $y = 0$, and if $P_0 = 0$ or $\lambda_J(0) > 0$, then $y = 0$ implies $x = 0$.*
2) *$(x, y) \in [0, \epsilon)^2$.*
3) *If $\{x_l\}_{l=0}^{\infty}$ and $\{y_l\}_{l=0}^{\infty}$ are defined by (4), then for every $l \geq 0$, $x_l \geq x$, $y_l \geq y$.*

Remark 2. *Item 1) in Lemma 1 expresses the asymmetry between the local and joint sides during the decoding algorithm discussed in Remark 1.*

Note that from the continuity of g in (3b), Item 1) in Lemma 1 implies that if $\lim_{l\to\infty} x_l(\epsilon) = 0$, then $\lim_{l\to\infty} y_l(\epsilon) = 0$. Thus, (5) can be re-written as

$$\epsilon_G^* = \sup\left\{\epsilon \in [0,1]\colon \lim_{l\to\infty} x_l(\epsilon) = 0\right\}. \tag{7}$$

Theorem 2. *Let*

$$\hat{\epsilon} = \sup\{\ \epsilon \in [0,1]\colon \text{(6) has no solution with } (x,y) \in (0,1]\times[0,1]\ \}. \tag{8}$$

Then, $\epsilon_G^* = \hat{\epsilon}$.

We proceed by providing a numerical way to calculate the threshold of a given choice of $\Lambda_L, \Lambda_J, \Omega_L$ and Ω_J. For every $x \in (0,1]$, define $q_L(x) \triangleq x \cdot \frac{\Lambda_L(1-\rho_L(1-x))}{\lambda_L(1-\rho_L(1-x))}$ and $q_J(x) \triangleq x \cdot \frac{\Lambda_J(1-\rho_J(1-x))}{\lambda_J(1-\rho_J(1-x))}$. It can be verified (see [11]) that $\lim_{x\to 0} q_L(x) = 0$, thus the intermediate-value theorem implies that for every $w \in (0,1]$ there exists $x \in (0,1]$ such that $q_L(x) = w$. However, it is not true in general that $\lim_{x\to 0} q_J(x) = 0$, and this limit may be infinite (check for example the case $P_0 > 0$, $\rho_J(x) = x^3$ and $\lambda_J(x) = x^2$). In view of the above, for every $y > 0$ such that $q_J(y) \le 1$, define $q(y) \triangleq \max\{x\colon q_L(x) = q_J(y)\}$.

Theorem 3. *Let $\lambda_L, \rho_L, \lambda_J, \rho_J$ be degree-distribution polynomials, let $P_0 \in [0,1]$, and let ϵ_G^* be the BP global decoding threshold of the $LDPCL(M, n, \Lambda_L, \Lambda_J, \Omega_L, \Omega_J)$ ensemble on the BEC when $n \to \infty$. If $P_0 = 0$ or $\lambda_J(0) > 0$, then*

$$\epsilon_G^* = \inf_{\substack{y\in(0,1]\\ q_J(y)\le 1}} \frac{y}{g(1, q(y), y)}. \tag{9}$$

Else,

$$\epsilon_G^* = \min\left\{\inf_{\substack{y\in(0,1]\\ q_J(y)\le 1}} \frac{y}{g(1, q(y), y)},\ \frac{1}{P_0}\cdot\epsilon_L^*\right\}. \tag{10}$$

Example 2. *Consider an LDPCL ensemble characterized by $\lambda_L(x) = x$, $\rho_L(x) = x^9$, $\lambda_J(x) = 0.3396x + 0.6604x^4$, $\rho_J(x) = x^9$, and $P_0 = 0.2667$. Using* (1) *and* (10)*, the design rate is $R = 0.5571$ and the global decoding threshold is $\epsilon_G^* = 0.35$ (the local code is $(2,10)$-regular, thus $\epsilon_L^* = 0.111$). Figure 2 illustrates the 2D-DE equations in* (2a)-(2b) *for two bit-erasure probabilities:* 0.33 *and* 0.37 *from left to right.*

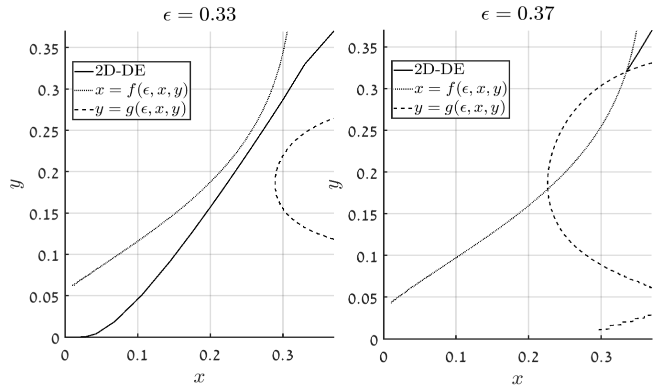

Fig. 2. Illustration of (2a)-(2b) for Example 2. When $\epsilon = 0.33 < \epsilon_G^*$, there are no (f,g)-fixed points – the decoding process ends successfully, and when $\epsilon = 0.37 > \epsilon_G^*$, there are two (f,g)-fixed points – the decoding process gets stuck.

5. An LDPCL Construction and Achieving Capacity

In this section, we propose an LDPCL code construction, and then we use capacity-achieving sequences of LDPC ensembles to construct an LDPCL capacity-achieving sequence that in addition can be sub-block decoded up to the local threshold permitted by the local degree distributions. In general, the construction's inputs are target local and global decoding thresholds, ϵ_L and ϵ_G, respectively; the outputs are degree-distributions $(\lambda_L, \rho_L, \lambda_J, \rho_J, P_0)$ such that $\epsilon_L^*(\lambda_L, \rho_L) = \epsilon_L$ and $\epsilon_G^*(\lambda_L, \rho_L, \lambda_J, \rho_J, P_0) = \epsilon_G$. Note that setting $P_0 = 0$, and picking any two LDPC ensembles (λ_L, ρ_L) and (λ_J, ρ_J) that induce thresholds $\epsilon^*(\lambda_L, \rho_L) = \epsilon_L$ and $\epsilon^*(\lambda_J, \rho_J) = \epsilon_G$ would suffice, but this choice yields poor rates (intuitively, with that choice the local and joint codes do not "cooperate"). Another solution is not using a joint ensemble at all, i.e., choosing (λ_L, ρ_L) such that $\epsilon^*(\lambda_L, \rho_L) = \epsilon_G > \epsilon_L$, and setting $P_0 = 1$. However, this solution is an undesired overkill since it would miss the opportunity to have a low-complexity local decoder for the majority of decoding instances where the erasure probabilities are below ϵ_L.

A. The Construction

Definition 2. *Let (λ_L, ρ_L) be degree-distribution polynomials, and let ϵ_L be their BP decoding threshold. For $\epsilon \in (\epsilon_L, 1)$, let*

1) $h_\epsilon(x) \triangleq \epsilon\lambda_L(1-\rho_L(1-x)) - x, \quad x \in [0,1]$
2) $x_s(\epsilon) \triangleq \max\{x \in [0,1]\colon h_\epsilon(x) \ge 0\}$
3) $a_s(\epsilon) \triangleq \Lambda_L(1-\rho_L(1-x_s(\epsilon)))$

For every $x \in [0,1]$, $h_\epsilon(x)$ is the erasure-probability change in one iteration of the BP algorithm on the local graph, if the current erasure probability is x. By definition, since $\epsilon > \epsilon_L$, $h_\epsilon(x) > 0$ for some $x \in [0,1]$. In addition, for every $x > \epsilon$, $h_\epsilon(x) < 0$, so $x_s(\epsilon)$ is well defined. Operationally, $x_s(\epsilon)$ is the local-edge erasure probabilty when the local decoder gets stuck. Definitions similar to items 1) and 2) have appeared in [7]; we add $a_s(\epsilon)$ as the variable-node erasure probability corresponding to $x_s(\epsilon)$.

Theorem 4. *let (λ_L, ρ_L) be a local ensemble inducing a local threshold ϵ_L, and let $\epsilon_G \in (\epsilon_L, 1)$. If $P_0 = \frac{\epsilon_L}{\epsilon_G}$, and (λ_J, ρ_J) is an ensemble having a decoding threshold $\epsilon_J = \epsilon_G \cdot a_s(\epsilon_G)$, then $\epsilon_G^*(\lambda_L, \rho_L, \lambda_J, \rho_J, P_0) = \epsilon_G$.*

B. Achieving Capacity

Let $\delta(\lambda,\rho)$ be the additive gap to capacity of the $LDPC(\lambda,\rho)$ ensemble, i.e., $\delta(\lambda,\rho) = 1-\epsilon^*(\lambda,\rho) - R(\lambda,\rho)$. Similarly, let $\delta(\lambda_L,\rho_L,\lambda_J,\rho_J,P_0) = 1 - \epsilon^*_G(\lambda_L,\rho_L,\lambda_J,\rho_J,P_0) - R(\lambda_L,\rho_L,\lambda_J,\rho_J,P_0)$ be the global additive gap to capacity. We say that a sequence of degree-distribution polynomials $\left\{\lambda_L^{(k)},\rho_L^{(k)},\lambda_J^{(k)},\rho_J^{(k)},P_0^{(k)}\right\}_{k\geq 1}$ achieves capacity on a $BEC(\epsilon_G)$, with local decoding capability ϵ_L if in the limit where $k\to\infty$ we have $\epsilon_L^*\left(\lambda_L^{(k)},\rho_L^{(k)}\right)\to\epsilon_L$, $\epsilon_G^*\left(\lambda_L^{(k)},\rho_L^{(k)},\lambda_J^{(k)},\rho_J^{(k)},P_0^{(k)}\right)\to\epsilon_G$, and $R\left(\lambda_L^{(k)},\rho_L^{(k)},\lambda_J^{(k)},\rho_J^{(k)},P_0^{(k)}\right)\to 1-\epsilon_G$. Note that the above implies that for capacity-achieving LDPCL sequences, $\lim_{k\to\infty}\delta\left(\lambda_L^{(k)},\rho_L^{(k)},\lambda_J^{(k)},\rho_J^{(k)},P_0^{(k)}\right)=0$.

Lemma 2. *Let $(\lambda_L,\rho_L,\lambda_J,\rho_J,P_0)$ be degree-distribution polynomials constructed according to Theorem 4, and let $\delta_L \triangleq \delta(\lambda_L,\rho_L)$ and $\delta_J \triangleq \delta(\lambda_J,\rho_J)$. Then,*

$$\delta(\lambda_L,\rho_L,\lambda_J,\rho_J,P_0) \leq \delta_L + \delta_J\cdot(1-P_0). \tag{11}$$

At this point, we can construct a capacity-achieving sequence of LDPCL ensembles on a $BEC(\epsilon_G)$, with a local decoding capability ϵ_L. Choose any two sequences of (ordinary) LDPC ensembles $\left\{\lambda_L^{(k)},\rho_L^{(k)},\right\}_{k\geq1}$ and $\left\{\lambda_J^{(k)},\rho_J^{(k)},\right\}_{k\geq1}$ that achieve capacity on the $BEC(\epsilon_L)$ and $BEC(\epsilon_G)$, respectively, and set $P_0^{(k)} = \left(1-\frac{\epsilon_L}{\epsilon_G}\right)$, for all $k\geq 1$. Clearly, the local threshold converges to ϵ_L, and in view of Theorem 4, the global threshold is lower bounded by ϵ_G ($a_s(\cdot)\leq 1$). Finally, Lemma 2 implies that

$$\begin{aligned}\lim_{k\to\infty}\delta\left(\lambda_L^{(k)},\rho_L^{(k)},\lambda_J^{(k)},\rho_J^{(k)},P_0\right)\leq\\ \lim_{k\to\infty}\delta\left(\lambda_L^{(k)},\rho_L^{(k)}\right)+\lim_{k\to\infty}\delta\left(\lambda_J^{(k)},\rho_J^{(k)}\right)\left(1-\frac{\epsilon_L}{\epsilon}\right)=0.\end{aligned}$$

Example 3. *We construct an LDPCL capacity-achieving sequence with local and global threshold $\epsilon_L = 0.05$ and $\epsilon_G = 0.2$, respectively. We set $P_0 = \frac{\epsilon_L}{\epsilon_G} = 0.25$, and we use the Tornado capacity-achieving sequence [9],*

$$\begin{aligned}\lambda_L^{(D_L)}(x) &= \frac{1}{H(D_L)}\sum_{i=1}^{D_L}\frac{x^i}{i},\\ \rho_L^{(D_L)}(x) &= e^{-\alpha_L}\sum_{i=0}^{\infty}\frac{(\alpha_L x)^i}{i!},\end{aligned} \tag{12}$$

where $H(\cdot)$ is the harmonic sum, $\alpha_L = \frac{H(D_L)}{\epsilon_L}$, and $\lambda_J^{(D_J)}(x),\rho_J^{(D_J)}(x)$ are defined similarly (the check degree-distribution series are truncated to get degree-distribution polynomials with finite degrees). D_L (resp. D_J) controls the local (resp. joint) gap to capacity δ_L (resp. δ_J); the bigger it is, the smaller the gap is. Table I exemplifies how the LDPCL sequence $\left\{\lambda_L^{(D_L)},\rho_L^{(D_L)},\lambda_J^{(D_J)},\rho_J^{(D_J)},P_0\right\}$ approaches capacity as $D_L\to\infty$, $D_J\to\infty$: Theorem 4 implies that for every value of D_L and D_J, the global decoding threshold is $\epsilon_G^*\geq 0.2$; the local additive gap to capacity δ_L and joint additive gap to capacity δ_J both vanish as $D_L\to\infty$ and $D_J\to\infty$, which in view of (11)*, implies that the global additive gap to capacity δ vanishes as well.*

TABLE I

D_L	δ_L	D_J	δ_J	Rate	δ
1	0.05	1	0.2	0.6	0.2
1	0.05	2	0.1	0.67	0.13
1	0.05	10	0.02	0.735	0.065
1	0.05	100	0.002	0.745	0.055
2	0.025	100	0.002	0.775	0.025
5	0.01	100	0.002	0.79	0.01
∞	0	∞	0	0.8	0

Table I shows the advantage of the multi-block scheme: one can achieve the local threshold with significantly simpler ensembles (use only few terms in (12)*). Such simpler codes allow to approach the theoretical threshold with shorter blocks and with lower complexity.*

6. Conclusion

This paper lays out the theoretical foundation for multi-block LDPC codes. Many directions for future work are opened: particularly interesting are finite block length design and analysis, and decoding algorithms with efficient local vs. global scheduling.

References

[1] M. Blaum and S. Hetzler, "Integrated interleaved codes as locally recoverable codes: properties and performance," Int. *J. Inf. Coding Theory*, vol. 3, no. 4, pp. 324344, Jan. 2016.

[2] Y. Cassuto, E. Hemo, S. Puchinger and M. Bossert, "Multi-block interleaved codes for local and global read access", *Proc. 2017 IEEE International Symposium on Information Theory*, pp. 1758–1762, Aachen, Germany, June 2017.

[3] R. Gallager, "Low-density parity-check codes," *IRE Trans. on Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[4] M. Hassner, K. Abdel-Ghaffar, A. Patel, R. Koetter, and B. Trager, "Integrated interleaving – a novel ECC architecture," *IEEE Trans. Magn.*, vol. 37, no. 3, pp. 773-775, Feb. 2001.

[5] J. Han and L. A. Lastras-Montano, "Reliable memories with subline accesses," *Proc. 2007 IEEE International Symposium on Information Theory*, pp. 2531–2535, Nice, France, June 2007.

[6] H. Jin, A. Khandekar and R. J. McEliece, "Irregular repeat-accumulate codes," *Proc. 2nd International Symposium on Turbo Codes and Related Topic*, pp. 1–8, Brest, France, Sep 2000.

[7] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold saturation via coupling: why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803-834, Feb. 2011.

[8] J. Li, K. Liu, S. Lin, K. Abdel-Ghaffar, "Reed-solomon based non-binary globally coupled LDPC codes: Correction of random errors and bursts of erasures", *Proc. 2017 IEEE International Symposium on Information Theory*, pp. 381–385, Aachen, Germany, June 2017.

[9] M.G. Luby, M. Mitzenmacher, M.A. Shokrollahi, and D.A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. on Inf. Theory*, vol. 47, no. 2, pp. 569-584, Feb. 2001.

[10] P. Razaghi and W. Yu, "Bilayer low-density parity-check codes for decode-and-forward in relay channels", *IEEE Trans. on Inf. Theory*, vol. 53, no. 10, pp. 3723-3739, Sep. 2007.

[11] E. Ram and Y. Cassuto, "LDPC codes with local and global decoding," preprint, January 2018. [Online]. Available at http://arxiv.org/abs/1801.03951.

[12] T. Richardson and R. Urbanke, *Modern Coding Theory*, New York, NY, USA: Cambridge Univ. Press, 2008.