# LDPC Ensembles that Universally Achieve Capacity under BP Decoding: A Simple Derivation

Anatoly Khina, Yair Yona, and Uri Erez
EE—Systems Department, Tel Aviv University
Tel Aviv, Israel
Email: {anatolyk,yairyo,uri}@eng.tau.ac.il

***Abstract*— A long-standing question in coding theory is whether code ensembles having a low-density parity check (LDPC) matrix can attain capacity under belief propagation (BP) decoding. An affirmative answer to this problem was recently given by the special class of spatially-coupled LDPC ensemble. In this work, we provide a simple derivation of a different LDPC ensemble that approaches capacity under BP decoding, following the classical approach of serial concatenation. This LDPC ensemble is constructed by concatenating a high-rate outer LDPC code with an inner random convolutional one. The analysis of the concatenated-coding framework takes a particularly simple — "black box" — form. Specifically, the joint effect of the particular inner code and the binary-input memoryless symmetric-output (BMS) channel is encapsulated in a single parameter — the Bhattacharyya parameter, which is maximal for the binary symmetric channel (BSC). This implies that an inner convolutional code designed for the BSC achieves good performance over all BMS channels with a given capacity. Moreover, the performance guarantee of the outer LDPC code under BP decoding is shown to be dictated solely by this parameter. This, in turn, implies that the overall concatenated code approaches capacity under BP decoding for all BMS channels with a given capacity, simultaneously.**

***Index Terms*—LDPC codes, convolutional codes, concatenated codes, belief propagation, Bhattacharyya parameter, error exponent, compound channel, BMS channels.**

## I. Introduction

Since the early days of information theory, a great deal of the effort has been dedicated to finding low-complexity schemes that are able to approach capacity. A major step towards this goal was made by Forney [1], who proposed using concatenated codes, taking the *inner code* to be a random convolutional code and the *outer code* — a Reed–Solomon (RS) one. Due to the polynomial decoding complexity of RS codes, the resulting code has complexity that grows polynomially with the code blocklength, while achieving an exponentially decaying error probability. In order to achieve a similar result but with linear complexity, Guruswami and Indyk (GI) proposed to replace the outer RS code, which is a *maximum distance separable* (MDS) code, with a near-MDS code [2] which they devised, that has linear decoding complexity and is based upon expander codes. Similar results were also obtained by Barg and Zémor [3], [4].

While the latter works have established that approaching capacity with low complexity is in a theoretical sense possible, these constructions are generally not considered practical and hence the search for practical codes (and decoderes) remains an active area.

The goal of achieving capacity over the binary erasure channel (BEC) with practical coding and decoding has been met by irregular low-density parity check (LDPC) codes under belief propagation (BP) decoding (which has linear decoding complexity), originally in the works of Luby *et al.* [5] and Shokrollahi [6].

Recent approaches for constructing capacity-achieving low-complexity codes over general binary-input memoryless output symmetric (BMS) channels include polar codes, conceived by Arıkan [7], and spatially-coupled low-density parity check (LDPC) codes, that were introduced by Felström and Zigangirov [8]. Polar codes have encoding and decoding complexities of the order of $\mathcal{O}(N \log N)$ and sub-exponential decay of the error probability in $N$, where $N$ is the blocklength. Spatially-coupled LDPC codes have been shown, by Kudekar *et al.* [9], to approach capacity under BP decoding (which has linear complexity).

The latter answers the question of whether codes having an LDPC matrix representation can achieve capacity under BP decoding for general BMS channels.

Another attractive property that spatially-coupled codes possess is *universality*. That is, they have been shown to achieve capacity simultaneously (*compound channel setting* [10]–[12]) for the class of BMS channels with a given capacity. We note, however, that even though the concatenated codes of Forney and GI were designed for the binary-symmetric channel (BSC), they attain capacity over the whole class of BMS channels simultaneously. This easily follows by noting that the inner codes used in the analysis of the concatenated schemes are random, and hence the recent results in [13] that show that the error exponent of the BSC is the lower envelope (pointwise infimum) of the error exponents of all possible BMS channels with a given capacity,[1] readily apply. Indeed, the BEC and the BSC serve as extremes for the class of BMS channels in terms of many properties (see [13], [14] and references therein), and their extremal properties shall be used in the proposed design and its analysis.

In this work, we provide a simple derivation of an LDPC ensemble, which is different from the spatially-coupled en-

[1] As will be shown in Section III, a straightforward consequence of the results of [13] is that the BSC has the worst (pointwise) error exponent also for random *convolutional* codes.

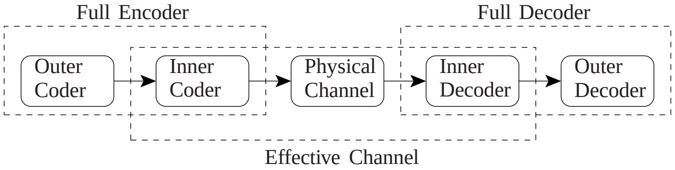

Fig. 1. Code concatenation.

semble, that universally approaches capacity for all BMS channels with a given capacity under BP decoding.[2] This LDPC ensemble is constructed by concatenating a high-rate outer LDPC code with an inner random convolutional one. The analysis of the concatenated-coding framework takes a particularly simple — "black box" — form. Specifically, the joint effect of the particular inner code and the BMS channel is encapsulated in a single parameter — the Bhattacharyya parameter. Coupled with the elegant result by Khandekar and McEliece [15], [16], the performance guarantee of an LDPC code under BP decoding is dictated solely by this parameter.[3] This, in turn, allows to translate the performance under BP decoding of LDPC codes over the BEC, to any BMS channel.We note that, as for convolutional codes belief propagation (BCJR algorithm [18]) amounts to (bitwise) *maximum a-posteriori* decoding [19], the proposed overall scheme achieves capacity under BP decoding. Furthermore, by invoking the extremal properties of the binary-symmetric channel (BSC) of [13], designing the inner convolutional code for the BSC guarantees the universality of the scheme over the whole class of BMS channels with a given capacity. Finally, as we shall show in the sequel, since the outer LDPC ensemble is of high-rate, it can be chosen to be regular, or alternatively having linear *encoding* complexity (in addition to the linear *decoding* complexity implied by BP decoding) and systematic representation, by incorporating irregular repeat-accumulate (IRA) codes [16], [20].

The rest of the paper is organized as follows. We start by presenting an overview of concatenated codes in Section II. We then introduce the necessary tools for the construction of the proposed scheme in Section III. We construct a sequence of LDPC codes that universally approach capacity under a two-stage message-passing decoding algorithm having linear decoding complexity, as well as under BP decoding, in Section IV. We conclude the paper in Section V.

## II. Background: Concatenated Schemes

Concatenated codes were conceived by Forney in his Ph.D. thesis [1]. Such codes are constructed from two base codes, operating in tandem, as depicted in Figure 1: The information bits are first encoded using the outer code, resulting in codewords,[4] whose entries are then used as the inputs to the second inner code, possibly interleaved prior to their encoding. Finally, the outputs of the inner code are transmitted over the physical channel.

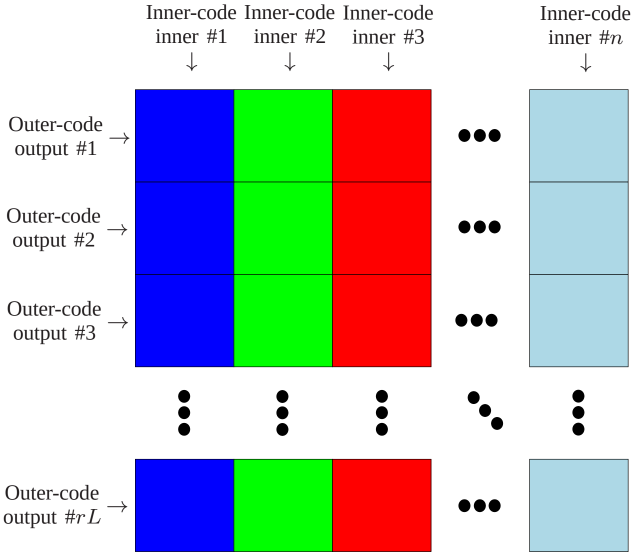

Fig. 2. Interleaving process used.

Decoding is carried out by reversing the process: The physical-channel outputs are fed to the decoder of the inner code; whose outputs (either bits/discrete symbols or "soft-decoded" real numbers) serve as the inputs to the decoder of the outer code. The decoder of the outer code recovers the information bits.

Forney, in his original scheme [1], proposed to use convolutional codes as the inner code and Reed–Solomon (RS) [21] as the outer one. By properly choosing the growth rates of the two codes, he showed that this ensemble can attain capacity with polynomial complexity and with an error probability that decays exponentially.

Guruswami and Indyk (GI) [2] leveraged this result by replacing the RS codes in Forney's scheme — which are *maximum distance separable* (MDS) codes (namely codes that achieve the Singleton bound [22]) having polynomial decoding complexity — with near-MDS codes having linear decoding complexity. Thus, for any gap to capacity, however small, using an inner code with long enough (yet fixed!) blocklength,[5] such that its error probability is small enough, and a sequence of near-MDS codes, the error probability can be made to decay exponentially with linear complexity, with an exponent arbitrarily close to Forney's error exponent (for a sufficiently large blocklength).

A frequently-used variant of the concatenation scheme incorporates systematic interleaving between the code layers, as depicted in Figure 1 and described next. Denote by $n$ and $L$ the

[2] With a slight abuse of notation, we shall refer to a sequence of LDPC code ensembles as an LDPC code ensemble of growing length.

[3] A similar and earlier universality result was obtained using the *expected soft-bit* parameter by Burshtein and Miller [17].

[4] When viewed as symbols, these codewords are referred to as *supersymbols* [1].

[5] GI employ an inner block code in contrast to Forney. However, Forney uses Gallager's error exponent [23] for *block codes* for the analysis of convolutional codes. Thus, the results of both Forney and GI are valid for inner block and convolutional codes alike.

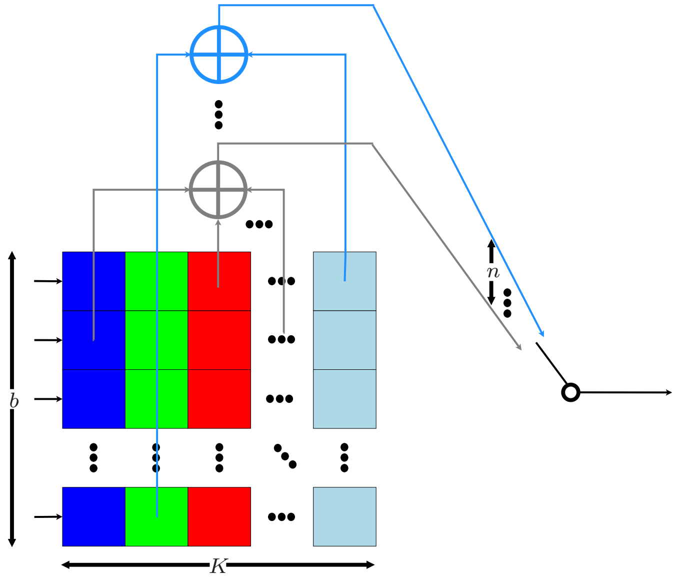

Fig. 3. Linear convolutional encoder. $\oplus$ denotes the exclusive-OR operation.

outer- and inner-code blocklengths, respectively, and by $r$ — the rate of the inner code. Then, the interleaver accumulates $rL$ consecutive outer-code codewords of length $n$, such that they comprise the rows of an $rL \times n$ matrix. The columns of the matrix are then fed to the inner coder, one by one, and the output of the coder is sent over the channel. At the decoder, the received outputs are recovered using the inner decoder. Every $n$ such recovered outputs of length $rL$ each are then accumulated at the de-interleaver, forming an $n \times rL$ matrix (which corresponds to the transpose of the matrix formed at the encoder). The columns of this matrix are then fed one by one to the outer decoder who recovers the information bits. This interleaving spreads adjacent errors within the inner-code block between different outer LDPC codewords, such that bursts of errors (stochastic process with memory) of the inner code are translated into independent error events at the decoding of the outer code.

In the remainder of the paper, we provide a simple derivation of an LDPC ensemble that universally approaches capacity under BP decoding. This LDPC ensemble can be regarded as a variant of the concatenated codes of Forney and GI, with the inner code being a random time-varying and zero-terminated convolutional code, and the outer code — an LDPC one.

## III. Building Blocks

In this section we introduce the tools that will serve in Section IV for obtaining the desired result.

### A. *Universality of Convolutional Codes*

We now derive an achievable bit error rate (BER) over the set of all BMS channels with a given capacity, using time-varying convolutional codes. For this, we review the results of [24, Chapter 5] on the error exponent and BER of convolutional codes, and combine them with the recent results of [13] on the universality of error exponents of block codes.

We use the notation and definitions of [24, Part 2] for convolutional codes. A compact representation (and implementation) of a convolutional code is via a shift register, as depicted in Figure 3. The length $K$ of the shift register is referred to as the constraint length of the resulting convolutional code, whereas its width $b$ is the number of information bits entering the shift register at each time instance. Thus, the total memory size is equal to $Kb$ bits. At each time instance, $n$ code bits are generated by evaluating $n$ functionals over the $Kb$ memory bits and the new $b$ information bits. Therefore, the rate of the code is equal to

$$r = \frac{b}{n}$$

bits per channel use. In general, these functionals may change at each time instance, resulting in a *time-varying* convolutional code. If these functionals are constant in time, we shall refer to the resulting code as *time invariant*.

*Remark 1:* The analysis in [24] considers an infinite stream of information and resulting code bits. Nevertheless, the derived bounds on the BER remain the same when terminating the convolutional code to a finite length. That is, using an information stream of finite length followed by $K$ zero inputs of width $b$, results in an effective finite block code. As long as the resulting block length is larger than $Kb$, the bound on the BER for infinite-length stream remains valid. This, however, comes at the price of reduced rate due to the zero termination, which can be made as small as desired by taking a long enough input stream-length (and blocklength).

Denote the channel capacity by $C$. The following proposition is due to Viterbi and Yudkin (VY) [24, Chapter 5].

*Proposition 1:* The BER of a random time-varying convolutional code with constraint length $K$, width $b$ and rate $r < C$ over a BMS channel is upper bounded by

$$P_b \leq \left(2^b - 1\right)^\rho \frac{2^{-K\frac{b}{r}E_0(\rho)}}{\left[1 - 2^{-b\left(\frac{E_0(\rho)}{r} - \rho\right)}\right]^2} \tag{1}$$

for any $0 \leq \rho < \min\{1, E_0(\rho)/r\}$, where

$$E_0(\rho) \triangleq -\log\left(\sum_y \left[\sum_x \frac{1}{2} p(y|x)^{\frac{1}{1+\rho}}\right]^{1+\rho}\right)$$

is the well-known $E_0$ of Gallager [23] and the logarithm is with respect to base 2.[6]

Note that $R_0 \triangleq E_0(\rho = 1)$ is the cutoff rate of Gallager's error exponent for block codes [23]. Beneath this rate, due to the expression in the denominator in (1), the optimization over $\rho$ should be carried over $[0, 1]$, whereas above the cutoff rate,

[6] The circular definition of $0 \leq \rho < E_0(\rho)/r$ means that we do not take into consideration the cases in which $\rho \geq E_0(\rho)/r$.

the optimization is restricted to the interval $[0, E_0(\rho)/r)$. Note that $E_0(\rho)$ encapsulates the channel law in the upper bound expression for the BER in (1). This will allow us, in turn, to establish universal upper bounds for $E_0(\rho)$, and hence also for the BER.

As the complexity per bit for convolutional codes is proportional to $2^{Kb} = 2^{Knr}$, it is meaningful to normalize $-\log\left(P_b(\rho, K, b)\right)$ by $Kn$ when considering the error exponent of convolutional codes. The interplay between the asymptotics of $K$ and $b$ may lead to different optimization problems for the error exponent of the convolutional code ensemble. For instance, at a fixed rate $r$ and fixed $K$, for asymptotically large $b = nr$, we get the following optimization problem

$$\max_{0\le\rho<\min\left(\frac{E_0(\rho)}{r},1\right)} \lim_{n\to\infty} -\frac{\log\left(\bar{P}_b(\rho, K, nr)\right)}{Kn} = \max_{0\le\rho<\min\left(\frac{E_0(\rho)}{r},1\right)} E_0(\rho) - \rho\frac{r}{K}\,.$$

Note that for $K = 1$ this optimization reduces to Gallager's random block-coding error exponent. On the other hand, for fixed $r$ and $b$, the optimization problem translates to

$$\max_{0\le\rho<\min\left(\frac{E_0(\rho)}{r},1\right)} \lim_{K\to\infty} -\frac{\log\left(\bar{P}_b(\rho, K, b)\right)}{Kn} = \max_{0\le\rho<\min\left(\frac{E_0(\rho)}{r},1\right)} E_0(\rho)\,. \tag{2}$$

The solution of (2) is larger or equal to Gallager's random block-coding error exponent.[7]

In the expressions above, taking $\rho = E_0(\rho)/r$ nullifies the denominator in (1) which corresponds to a trivial upper bound on the BER. The following resolves this issue by introducing an additional constant $\epsilon$ which can take any value within $(0, 1)$:

$$\begin{aligned} E_{\text{VY}}(r,\epsilon) &= \max_{0\le\rho\le\min\left((1-\epsilon)\frac{E_0(\rho)}{r},1\right)} E_0(\rho) \\ &= \begin{cases} R_0 & 0 \le r \le R_0(1-\epsilon) \\ E_0(\rho_0) & R_0(1-\epsilon) < r \le C(1-\epsilon) \end{cases}, \end{aligned} \tag{3}$$

where $\rho_0$ is the largest solution of $\rho r = (1-\epsilon)\,E_0(\rho)$.

By assigning $E_{\text{VY}}(r,\epsilon)$ of (3) in Proposition 1, the following is attained.

*Proposition 2:* The BER of a random time-varying convolutional code with constraint length $K$, width $b$ and rate $r < C$ over a BMS channel with capacity $C$ is upper bounded by

$$P_b \le \left(2^b - 1\right) \frac{2^{-K\frac{b}{r}E_{\text{VY}}(r,\epsilon)}}{\left[1 - 2^{-\epsilon\frac{b}{r}E_{\text{VY}}(r,\epsilon)}\right]^2}\ , \tag{4}$$

for any $\epsilon \in (0, 1)$.

We next show that the infimum of the error exponents of convolutional codes of the class of BMS channels with a given capacity is bounded away from zero, and reduces to that of the BSC. This result is based upon its parallel for block codes of [13].

To that end, denote the compound channel,whose possible transition distributions comprise all BMS channels with capacity $C$, by $\mathsf{BMS}(C)$; denote further the BSC with capacity $C$ by $\mathsf{BSC}(C)$. Block-code error exponents will be denoted by a subscript $G$, and the convolutional-code error exponents of (3) — by VY; superscripts indicate to which channel these error exponents refer.

*Theorem 1 ( [13]):* The error exponent of a random block code over any channel $c \in \mathsf{BMS}(C)$ is lower bounded by the random block-code error exponent of $\mathsf{BSC}(C)$:

$$E_G^{(c)}(r) \ge E_G^{\mathsf{BSC}}(r)\,.$$

*Corollary 1:* The (universal) error exponent of a random block code over the compound channel $\mathsf{BMS}(C)$ is equal to the random block-code error exponent over the $\mathsf{BSC}(C)$:

$$E_G^{\mathsf{BMS}}(r) = E_G^{\mathsf{BSC}}(r)\,.$$

We next connect the universal error exponent of block codes to that of convolutional codes.

*Lemma 1:* For any BMS channel $c \in \mathsf{BMS}(C)$, the BER can be upper bounded as in (4), with $E_{\text{VY}}^{(c)}(r,\epsilon)$ replaced with $E_G^{\mathsf{BSC}}\left(\frac{r}{1-\epsilon}\right) > 0$, for $0 \le r < (1-\epsilon)\,C$ and for any $0 < \epsilon < 1$.

*Proof:* The optimization problem in (3) and Theorem 1 give rise to the following chain of inequalities

$$E_{\text{VY}}^{(c)}(r,\epsilon) \ge E_G^{(c)}\left(\frac{r}{1-\epsilon}\right) \ge E_G^{\mathsf{BSC}}\left(\frac{r}{1-\epsilon}\right) > 0\,,$$

if $r < C(1-\epsilon)$, for any $\epsilon \in (0, 1)$ and $c \in \mathsf{BMS}(c)$. ■

Similarly to Corollary 1, the following is an immediate consequence of the randomness of the ensemble.

*Corollary 2:* The BER of a random time-varying convolutional code over the compound channel $\mathsf{BMS}(C)$ is upper bounded (universally) as in (4), with $E_{\text{VY}}^{(c)}(r,\epsilon)$ replaced with $E_G^{\mathsf{BSC}}\left(\frac{r}{1-\epsilon}\right) > 0$, for $0 \le r < (1-\epsilon)\,C$ and any $0 < \epsilon < 1$:

$$\begin{aligned} P_b &\le \left(2^b - 1\right) \frac{2^{-K\frac{b}{r}E_G^{\mathsf{BSC}}\left(\frac{r}{1-\epsilon}\right)}}{\left[1 - 2^{-\epsilon\frac{b}{r}E_G^{\mathsf{BSC}}\left(\frac{r}{1-\epsilon}\right)}\right]^2} \\ &\triangleq P_b'(K, r, b, \epsilon)\,. \end{aligned}$$

Denote further

$$\tilde{P}_b(K, r, b) \triangleq \min_{\epsilon\in(0,1)} P_b'(K, r, b, \epsilon)\,. \tag{6}$$

Consequently, for fixed $r < C$, $b$ and $n$, an arbitrarily small BER can be achieved universally over $\mathsf{BMS}(C)$, for a sufficiently large constraint length $K$.

[7] Note that while $E_{\text{VY}}(r,\epsilon)$ may be larger than Gallager's random coding error exponent, it scales with $n$, whereas Gallager's error exponent scales with the blocklength.

### *B. Bounds on the Performance of Ensembles of LDPC Codes via the Bhattacharyya Parameter*

We now obtain performance guarantees for LDPC codes under BP decoding over a *general BMS channel* via its performance over the BEC. These guarantees are formulated in Lemma 2 in terms of an upper bound on the achievable BER after $\ell$ iterations; this bound is based, in turn, on the density-evolution (DE) equations for the BEC, with the erasure probability replaced by the Bhattacharyya parameter of the BMS channel, and is given as part of the proof of Theorem 4.2 in [16]. We base our notation, as well as the conditions needed for the DE analysis to hold (the tree assumption, the concentration property *etc.*), on [25].

We state this result explicitly in the next proposition and provide its proof for completeness.

*Proposition 3:* Consider a BMS channel $P(y|x)$, where $x \in \{0,1\}$. Generate an LDPC ensemble w.r.t. variable- and check-node edge distributions $\lambda(x)$, $\rho(x)$ respectively, and denote by $x^{(\ell)}$ the log-likelihood ratio (LLR) of a variable-node message in iteration $\ell \geq 0$. Without loss of generality, assume that the zero codeword is transmitted. Then, in iteration $\ell+1$ we have

$$E\left[e^{-\frac{x^{(\ell+1)}}{2}}\middle|\boldsymbol{x}=0\right] \leq Z^{(\ell+1)} \triangleq B \cdot \lambda\left(1-\rho\left(1-Z^{(\ell)}\right)\right) , \tag{7}$$

where $Z^{(0)} = B$, $\boldsymbol{x}$ denotes the transmitted codeword,

$$x^{(0)} \triangleq \log\left(\frac{P(y|0)}{P(y|1)}\right) ,$$

is the initial LLR variable node message, and

$$B \triangleq \sum_y \sqrt{P(y|0)\,P(y|1)}$$

is the Bhattacharyya parameter.[8]

*Proof:* Combining equations $(4.6)$, $(4.7)$ of [16], we obtain

$$E\left[e^{-\frac{x^{(\ell+1)}}{2}}\middle|\boldsymbol{x}=0\right] \leq \tag{8}$$
$$B \cdot \lambda\left\{1-\rho\left(1-E\left[e^{-\frac{x^{(\ell)}}{2}}\middle|\boldsymbol{x}=0\right]\right)\right\} .$$

Now, by noting that

$$0 < E\left[e^{-\frac{x^{(\ell)}}{2}}\middle|\boldsymbol{x}=0\right] \leq Z^{(\ell)} \leq 1$$

it can be shown by induction, with

$$Z^{(0)} = E\left[e^{-\frac{x^{(0)}}{2}}\middle|\boldsymbol{x}=0\right] = B$$

serving as the base case, that

$$\lambda\left(1-\rho\left(1-Z^{(\ell)}\right)\right) \geq \tag{9}$$
$$\lambda\left(1-\rho\left\{1-E\left[e^{-\frac{x^{(\ell)}}{2}}\middle|\boldsymbol{x}=0\right]\right)\right\} .$$

Substituting (9) in (8) completes the proof. ■

Using Proposition 3 we obtain the following guarantee for the BER after $\ell$ iterations.

*Lemma 2:* The BER in iteration $\ell$ is upper bounded as

$$\Pr\left(x^{(\ell)} > 0\right) \leq Z^{(\ell)} .$$

Thus, if $\lim\limits_{\ell\to\infty} Z^{(\ell)} = 0$, then $\lim\limits_{\ell\to\infty} \Pr\left(x^{(\ell)} > 0\right) = 0$ as well.

*Proof:* Since $x^{(\ell)}$ is equal to the LLR in the $\ell$-th iteration, an error occurs if $e^{-\frac{x^{(\ell)}}{2}} > 1$. Thus, using Markov's inequality and (7), we attain

$$\Pr\left(x^{(l)} > 0\right) = \Pr\left(e^{-\frac{x^{(l)}}{2}} > 1\right)$$
$$\leq E\left[e^{-\frac{x^{(l)}}{2}}\middle|\boldsymbol{x}=0\right] \leq Z^{(\ell)} ,$$

as desired. ■

A threshold [25, Ch. 4] for the upper bound on the DE equations is defined as the largest Bhattacharyya parameter, $B_0$, such that for any $0 < Z^{(0)} = B < B_0$ we get $\lim_{\ell\to\infty} Z^{(\ell)} = 0$. In the next subsection we derive a performance guarantee based on the existence of such a threshold, as well as provide a proof for the existence of a threshold for regular LDPC codes.

### *C. LDPC Codes Achieve Capacity Under BP Decoding over Almost-Clean Channels*

In [2] it has been shown that by considering for the outer code an algebraic "near-MDS" code of rate approaching 1, Forney's error exponent [1], and as a consequence also the BSC capacity are achieved with linear complexity. The mechanism that enables this concatenated coding scheme to attain arbitrarily small error probability relies on the fact that the minimal distance of the outer code grows linearly with the block length. Therefore, if the inner code induces a transition probability for the outer code, that is smaller than the relative minimum distance, then the outer code error probability will decrease as desired for increasing block length. Further, since the rate of the outer code is nearly 1, the rate penalty is negligible.

Similarly, in our scheme we also consider an outer LDPC/IRA code with a rate very close to 1. However, the mechanism that enables the outer code in our scheme to reduce the error probability as desired is the existence of a threshold for the bound on the DE equation (7). The inner code induces a transition probability for the outer code such that the Bhattacharyya parameter $B$ is small enough to guarantee that $E\left[\exp\left\{-x^{(\ell+1)}/2\right\}\middle|\boldsymbol{x}=0\right]$ approaches zero as $\ell$ increases.

The next lemma shows that an LDPC ensemble at rate $1-\delta$, for which the upper bound on the DE equation (7) has a threshold, attains a fraction $1-\delta$ of capacity for any BMS channel with $B$ smaller than the threshold.

[8] The summation over $y$ is replaced by an integral for continuous alphabet channels.

*Lemma 3:* Consider an ensemble of LDPC codes at rate $1-\delta$, where $0<\delta<1$. Assume that the upper bound on the DE equation in (7) converges to zero as $\ell$ increases, for any $0<B<B_0$. In this case the LDPC ensemble achieves a fraction $1-\delta$ of capacity for any BMS channel with Bhattacharyya parameter $B$ smaller than $B_0$.

*Proof:* The channel capacity of any BMS channel is upper bounded by $1$. Based on Proposition 3 and Lemma 2, in case the upper bound for the DE equation has a threshold at $B_0$, then arbitrary small error probability will be achieved for any BMS channel for which $B<B_0$. Therefore, the LDPC codes ensemble achieves at least a fraction $\frac{1-\delta}{1}$ of the channel capacity for these BMS channels. ■

This lemma will serve as a building block for showing that the considered concatenated coding scheme universally achieves capacity.

Moreover, among other desired properties, three appealing choices are those of linear *encoding* complexity, systematic codes, and regularity (of LDPC codes). The first two properties are offered by the special class of IRA codes, introduced in [16]. In fact, the results of Section III-B, were also introduced in the Ph.D. thesis of Khandekar [16], and were shown to be valid both for general LDPC codes and for IRA codes. Achieving the desired result with regular LDPC codes is possible for a channel with a sufficiently small Bhattacharyya parameter, as follows.

*Lemma 4:* Consider any ensemble of regular LDPC codes with variable nodes of degree $d_v$ and check nodes of degree $d_c$. Then, there exists a threshold for the upper bound on the DE equation (7) for this ensemble.

*Proof:* For a regular ensemble of LDPC codes the upper bound for the DE equation in iteration $\ell$ takes the following form (see, *e.g.*, [25, Ch. 4]):

$$Z^{(\ell+1)} = B \cdot \left(1 - \left(1 - Z^{(\ell)}\right)^{d_c-1}\right)^{d_v-1} . \qquad (10)$$

In order to show the existence of a threshold we wish to find a certain value $B_0$ for which when assigning $B<B_0$ in (10) and also considering $Z^{(\ell)} \ll 1$ we get that $\lim_{\ell\to\infty} Z^{(\ell)} = 0$. Assuming $Z^{(\ell)} \ll 1$, (10) can be approximated via its first-order Taylor expansion as

$$Z^{(\ell+1)} = B \cdot (d_c-1)^{d_v-1} \left(Z^{(\ell)}\right)^{d_v-1} .$$

Therefore, taking $B < 1/\left(d_c-1\right)^{d_v-1}$ leads to $\lim_{\ell\to\infty} Z^{(\ell)} = 0$. Since $Z^{(0)} = B$, considering $B$ that satisfies both $B \ll 1$ and $B < 1/\left(d_c-1\right)^{d_v-1}$, leads to $\lim\limits_{\ell\to\infty} Z^{(\ell)} = 0$, which proves the existence of a threshold. ■

## IV. Putting It All Together

We build on the results of the previous section for the construction of a special ensemble with an LDPC matrix that approaches capacity under BP decoding over a factor graph, universally for the whole class of BMS channels with a given capacity. The construction is a concatenated one, as depicted in Figure 1, where the inner code is a convolutional code whose constraint length is chosen according to the desired gap to capacity, and the outer code is chosen to be an LDPC code whose length should be taken long enough to achieve any desired BER.

For the sake of simplicity of analysis, we consider a suboptimal message-passing decoding algorithm in Section IV-A, and show that it achieves the desired result. We then argue, in Section IV-B, that full BP decoding achieves performance at least as good as this crude message-passing algorithm.

### *A. Achieving Capacity under Suboptimal Message-Passing*

The concatenated code used throughout this section is generated using the following encoder.

*Algorithm 1 (Concatenated encoder):*

1) Encodes the information bits using an outer LDPC coder of length $n$.
2) Interleaves systematically the output of the LDPC coder, by accumulating $rL$ outer-code words of length $n$, as described in Section II and Figure 2.
3) Encodes the output of the interleaver using an inner zero-terminated convolutional coder of length $L$ and rate $r$.

*Remark 2:* As the outer LDPC code blocklength is much larger than that of the inner zero-terminated convolutional code, the resulting overall code has an LDPC structure.

In this subsection we make use of the following two-stage message-passing decoding algorithm.

*Algorithm 2 (Two-stage decoder):*

**Inner code decoding:** Calculates the LLRs of each input bit of the inner code using the BCJR algorithm; these bits constitute the outer LDPC coded bits.

**De-interleaving:** Reverses the interleaving used at the encoder, as described in Section II.

**Outer code decoding:** Applies BP decoding for the outer LDPC code of length $n$, over the effective BMS channel induced by the LLRs of the inner code.

*Remark 3:* This message-passing algorithm is not equivalent to full BP decoding over the entire scheme, as will be discussed in Section IV-B.

The following lemma states that the two-stage decoding of Algorithm 2 universally achieves capacity with linear complexity over all BMS channels with a given capacity.

*Lemma 5:* For any gap to capacity $\Delta>0$, however small, a code ensemble of rate $R=C-\Delta$ can be constructed using Algorithm 1, that universally achieves an arbitrarily small BER over BMS$(C)$ under the two-stage message-passing decoding of Algorithm 2 with linear complexity.

Specifically, this is achieved by a convolutional code of rate $r\in(R,C)$ and a long enough constraint length $K$, such that

$\tilde{P}(K, r, b)$ of Corollary 2 satisfies[9]

$$0 < 2\sqrt{\tilde{P}(K,r,b)\left[1 - \tilde{P}(K,r,b)\right]} \triangleq B_0 < 1 - \frac{R}{r};$$

and an LDPC ensemble of rate $R/r$ whose threshold over a BEC is above $B_0$. By taking the length $n$ of this ensemble to be large enough, an arbitrarily small BER can be achieved.

*Proof:* We start by showing that a random convolutional code and an LDPC code can be generated with the desired parameters.

As shown in [13], the Bhattacharyya parameter of any BMS channel with a given capacity, is upper bounded by that of the BSC of the same capacity. Moreover, the Bhattacharyya parameter of a BSC monotonically decreases with capacity. Therefore, the Bhattacharyya parameter $B$ of the effective BMS channel induced by the LLRs of the inner code, is upper bounded by the Bhattacharyya parameter of this channel after applying hard decoding ("slicing") to the channel outputs. The latter results in an effective BSC with a transition probability that is upper bounded by $\tilde{P}(K, r, b)$. This leads, in turn, to the upper bound

$$B \le B_0 \,.$$

By choosing $K$ large enough, $\tilde{P}_B(K, r, b)$, and hence also $B_0$, can be made as small as desired, according to Corollary 2.

LDPC ensembles of rate $R/r$ that have a threshold that is larger than $B_0$ over the BEC are well known to exist [5], [6] (see also [25], [16]). Proposition 3 and Lemma 2 guarantee that these ensembles achieve a BER as small as desired over all BMS channels with the same Bhattacharyya parameter, simultaneously.

By concatenating such codes, as in Algorithm 1, we achieve a code of total rate $R$. The decoder of Algorithm 2 first recovers the LLRs of each input bit of the inner code, using the BCJR algorithm. This induces an effective BMS channel with Bhattacharyya parameter $B$ that is upper bounded by $B_0$. The de-interleaving guarantees that this channel is memoryless [1]. Lastly, decoding the LDPC code over this induced BMS channel with $B < B_0$, achieves the desired result. ■

*Remark 4:* $L$ should be taken large enough such that the loss in rate due to the zero-padding is negligible. This loss can be absorbed in $\Delta$ and can be made arbitrarily small by choosing a large enough, but yet finite, $L$.

*Remark 5:* As is evident from the bounds in Lemma 5, considering an inner convolutional code that is designed for a BSC$(C)$ and an outer code that is designed for a BEC, suffices to prove the universality of the scheme over the whole class of BMS$(C)$.

*Remark 6:* In the proposed scheme, the rate of the convolutional code is chosen to be close to capacity, whereas the rate of the outer LDPC code is close to 1.

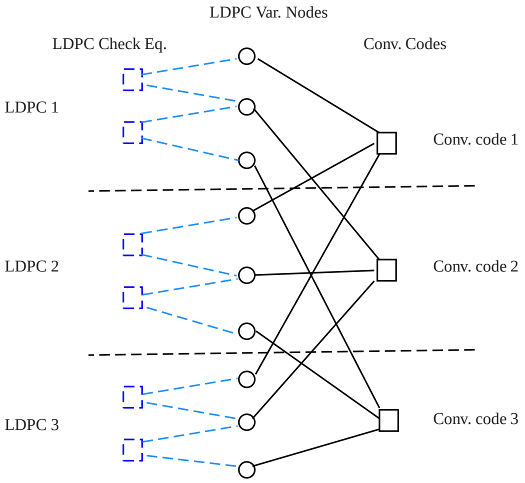

Fig. 4. Bipartite graph representation of the concatenated coding scheme for $n = L = 3$. The dashed squares represent the LDPC code parity check equations. The convolutional codes are represented by the solid squares. Finally, the circles represent the LDPC code variable nodes.

The following is a simple corollary of Lemmata 4 and 5.

*Corollary 3:* The result of Lemma 5 remains valid when using a *regular* LDPC ensemble as the outer code in Algorithm 2.

*Proof:* Lemma 4 shows that regular LDPC ensembles have a threshold that is bounded away from zero. Thus, retracing the proof of Lemma 5 and choosing $B_0$ to be beneath this threshold, proves the desired result with regular LDPC code ensembles. ■

Some desired properties for practical implementation are those of linear encoding complexity and systematic representation. Both can be easily achieved by replacing the outer LDPC code in Lemma 4 with an IRA code, as the lemma remains valid for such codes, as well.

### B. Achieving Capacity under Belief Propagation Decoding

In this subsection we consider a slightly generalized variant of the encoder of Algorithm 1: We use $rL$ independent LDPC ensembles of the same parameters. That is, the columns in the interleaver of Algorithm 1 are drawn from independently generated LDPC codebooks. We note that all the results of Section IV-A remain unchanged for this variant. This variant allows to guarantee an extended tree assumption (formally defined in the sequel), which is subsequently used to show that BP decoding of the overall resulting code is at least as good as that of the two-stage message-passing decoding of Algorithm 2. In particular, it achieves universally the channel capacity of BMS$(C)$ under BP decoding over the *factor graph* of the overall code, which results from the factor graph of the convolutional codes and the factor graphs of the LDPC codes.

Before considering the extended tree assumption, let us present the bipartite graph representation for our proposed coding scheme. We use $rL$ LDPC codes, each of length

[9] $B_0$ is the resulting Bhattacharyya parameter of an effective BSC with transition probability $\tilde{P}(K, r, b)$.

$n$ for the outer layer, and $n$ time-varying zero-terminated convolutional codes each of length $L$ for the inner code.[10] Denote the $j$-th symbol of the $i$-th LDPC codeword by $x_{i,j}$, where $1 \leq i \leq rL$ and $1 \leq j \leq n$. The mapping of the outer LDPC code variable nodes to the inner zero-terminated convolutional codes is as follows. Symbol $x_{i,j}$, $1 \leq i \leq rL$, is mapped to convolutional code $j$, *i.e.*, the first symbol in each LDPC code, $x_{i,1}$, is mapped to the first block of the zero-terminated convolutional codes, *etc*. Figure 4 presents an example for the (bipartite) factor graph for $n = rL = 3$.

The following assumption will be used in the BP analysis to follow.

*Assumption 1 (Extended tree assumption):* The $\ell$-depth extended tree assumption states that variable node $x_{i,j}$ shares no loops with the subtrees of depth $\ell$ spanned by each other variable node $x_{k,t}$ (with at least one of $i \neq k$ or $j \neq t$ holding).

In the proposed construction, this assumption amounts to the "regular" tree assumption (*cf.* [25, Ch. 3]) along with an "extension". The regular tree assumption states that $x_{i,j}$ shares no loop with the subtrees of depth $\ell$ stemming from variable nodes $\{x_{i,t} | t \neq j\}$, which comprise with it the same LDPC codeword. The extension to the regular tree assumption assumes also that $x_{i,j}$ shares no loop with the subtrees of depth $(\ell - 1)$ stemming from variable nodes $\{x_{k,j} | k \neq i\}$, which correspond to the same zero-terminated convolutional codeword.

The following lemma states that the extended tree assumption is satisfied for sufficiently long outer LDPC codes with high probability, and is a simple extension of the regular tree assumption [26].

*Lemma 6:* Let $L$ be the length of the zero-terminated convolutional code. Then, for any $\epsilon^{'} > 0$ and $\ell > 0$, we can choose the length $n$ of the LDPC ensembles to be sufficiently large, such that the $\ell$-depth extended tree assumption is satisfied with probability greater than $1 - \epsilon^{'}$ over the factor graph induced by the overall code.

*Proof:* The factor graph of the overall code is induced by the factor graph of the outer LDPC codes and the factor graph of the inner zero-terminated convolutional codes. As the zero-terminated convolutional code length $L$ and the number of iterations $\ell$ are finite, the resulting tree has a finite number of variable nodes, whereas the length of each of the outer codes $n$ can be chosen to be arbitrarily large. The proof that the extended tree assumption holds with arbitrarily high probability for a sufficiently large $n$, follows by retracing the proof for the regular tree assumption for LDPC ensembles (see, *e.g.*, [25]). ■

*Remark 7:* The length of the LDPC ensembles $n$ needed to satisfy the *extended* tree assumption with a given probability is greater than that needed for the regular tree assumption to hold with the same probability. Thus, the value of $n$ required for the analysis of full BP to hold is greater than that needed for the analysis of Algorithm 2 of Section IV-A.

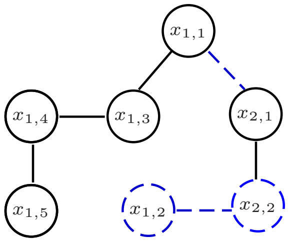

Fig. 5. The extended tree assumption. Solid lines connect symbols that take place in the same LDPC parity check equation. Dashed lines connect symbols that take place in the same convolutional code. The two-stage message-passing algorithm decodes over the subtree consisting of solid circles, whereas the BP decodes over the entire tree.

We now describe the BP decoding algorithm over the overall (bipartite) factor graph.

*Algorithm 3 (Belief-propagation decoder):*

**Variable node:** Sums all received LLR messages from the check nodes it is connected to: LDPC check nodes, convolutional code nodes and channel observation nodes. This sum is then sent back to these nodes.

**LDPC node:** Operates as in "regular" BP decoding of an LDPC code.

**Convolutional code node:** Operates as in regular BCJR decoding with non-uniform prior, where the latter is dictated by the messages coming from the LDPC codes.

The following lemma and theorem show that the concatenated LDPC ensemble achieves universally capacity under BP decoding.

*Lemma 7:* Under the extended tree assumption (Assumption 1), the BER achievable by Algorithm 3 is upper bounded by the BER achievable by Algorithm 2.

*Proof:* Under the extended tree assumption (Assumption 1), the two-stage message-passing decoding of Algorithm 2 is carried over a subtree of the BP decoder (see also the illustration in Figure 5). Since BP decoding is optimal under the tree assumption (see, *e.g.*, [25]), it follows that the BER achievable by Algorithm 3 is upper bounded by the BER achievable by Algorithm 2. ■

*Theorem 2:* For any gap from capacity $\Delta > 0$, however small, a code ensemble of rate $R = C - \Delta$ can be constructed using Algorithm 1 with $rL$ (independent) LDPC codes, that (universally) achieves an arbitrarily small BER under the BP decoding of Algorithm 3, over BMS$(C)$.

Specifically, this is achieved by a convolutional code of rate $r \in (R, C)$ and a long enough constraint length $K$, such that $\tilde{P}(K, r, b)$ of Corollary 2 satisfies

$$0 < 2\sqrt{\tilde{P}(K,r,b)\left[1 - \tilde{P}(K,r,b)\right]} \triangleq B_0 < 1 - \frac{R}{r}; \quad (11)$$

and LDPC ensemble of rate $R/r$ whose threshold over a BEC is above $B_0$. By taking the length $n$ of this ensemble to be

[10] Drawing $n$ independent codewords from the same zero-terminated convolutional code, in the analysis to follow, yields the same results.

large enough, an arbitrarily small BER can be achieved.

*Proof:* Use Lemma 5 to establish the desired parameters of the convolutional code for the two-stage message-passing decoding of Algorithm 2. Now take the length of the LDPC ensemble to be long enough such that the sum of the probability that the extended tree assumption fails, and the BER of the LDPC code, is smaller than the desired BER. Lemmata 5 and 7 guarantee that the BER of the overall code is lower than this desired BER, as it can be made arbitrarily small, by choosing large enough $n$ and $\ell$. Finally note that, as in Lemma 5, the rate of the overall code is $R$, as desired. ■

*Corollary 4:* A code as in Lemma 5 and Theorem 2 can be devised that achieves capacity simultaneously for any (finite) subset $\mathcal{S}$ of $\mathsf{BMS}(C)$, for a sufficiently large constraint length $K$, under the two-stage message-passing decoding of Algorithm 2 or the BP decoding of Algorithm 3.

*Proof:* We start by generating an appropriate inner convolutional code. For any $\varepsilon > 0$, however small, define $A_c$ as the event that the BER of a randomly generated convolutional code over channel $c \in \mathcal{S}$ is upper bounded by

$$P_b \leq \tilde{P}_b(K, r, b) \mathrm{e}^{K \frac{b}{r} \varepsilon} \triangleq \tilde{P}_b(K, r, b, \varepsilon) \,,$$

where $\tilde{P}_b$ was defined in (6).

Then for any $\varepsilon > 0$, there exists a convolutional code that attains a BER of at most $\tilde{P}_b(K, r, b, \varepsilon)$, for all $c \in \mathcal{S}$ simultaneously, for $K$ sufficiently large. We prove this by showing that the probability that a randomly generated convolutional code satisfies this with positive probability:

$$\begin{aligned} \Pr\left(\bigcap_{c \in \mathcal{S}} A_c\right) &= 1 - \Pr\left(\bigcup_{c \in \mathcal{S}} \overline{A_c}\right) \\ &\geq 1 - \sum_{c \in \mathcal{S}} \Pr\left(\overline{A_c}\right) && \text{(12a)} \\ &\geq 1 - |\mathcal{S}| \, e^{-K \frac{b}{r} \varepsilon} && \text{(12b)} \\ &> 0 \,, && \text{(12c)} \end{aligned}$$

where $\overline{A_c}$ denotes the complement of $A_c$, (12a) follows from the union bound, (12b) follows from the Markov inequality

$$\Pr\left(\overline{A_c}\right) \leq \frac{\tilde{P}_b(K, r, b)}{\tilde{P}_b(K, r, b, \varepsilon)} = e^{-K \frac{b}{r} \varepsilon} \qquad (13)$$

with the expected BER $P_b$ over the random-code ensemble being upper bounded by the numerator in (13) according to (6), and (12c) holds true for a sufficiently large $K$. By taking $\varepsilon$ to be small enough we achieve $B_0$ of (11).

Next, generate an LDPC code from the outer-code ensemble proposed in the proof of Theorem 2, *viz.*. a long enough ensemble, s.t. the sum of the probability that the extended tree assumption fails and of the BER of the LDPC code is below the desired BER of the overall code. Using the *concentration of the BER* phenomenon for LDPC codes (see, *e.g.*, [25]), an LDPC code with the desired BER can be generated with high probability for a sufficiently large $n$. This establishes the desired result for the two-stage message-passing decoding of Algorithm 2.

Finally, as in the ensemble analysis, under the extended tree assumption (which we already took into consideration when bounding the BER), BP decoding of the overall concatenated code as in Algorithm 3 is optimal and hence achieves a BER which is upper bounded by that of the two-stage message-passing decoder. ■

## V. Conclusion and Extensions

In this work, we have shown how the classical concatenated approach allows constructing an LDPC ensemble that universally achieves capacity under BP decoding. The key elements that were utilized are the BER extremal property of the BSC of a random (convolutional) code and the performance guarantees provided by the Bhattacharyya parameter of BP decoding of LDPC codes.

Several interesting directions for further research are as follows.

**Time variance of convolutional codes.** Throughout the paper, we have made use of *time-variant* convolutional codes. Indeed, similar meaningful results for time-invariant and periodically time-variant convolutional codes similar to the upper bounds on the BER of Section III-C are not available [27], which calls for further research.

**Avoiding interleaving.** For the proof of Lemma 5 and Theorem 2 we assumed the incorporation of an interleaver. This interleaver simplified analysis by providing effective memoryless channels for the outer LDPC codes. However, as in the construction the inner code is of fixed length and the length of the outer LDPC codes goes to infinity (to achieve an arbitrarily small BER), this interleaver is not material and can be dropped. This can be seen by noting that, in this limit, according to the law of large numbers, the empiric distribution of the LLRs within the blocklength of a single LDPC code is close, with high probability, to their statistical distribution,[11] and since the LDPC ensembles are robust to place permutations.

**LDPC performance guarantees.** In this paper, we have used the performance guarantees offered by the Bhattacharyya parameter, established by Khandekar and McEliece [15], [16], coupled with the extremal properties of the BEC and the BSC. Nevertheless, similar results were obtained by Burshtein and Miller [17] via the expected soft-bit parameter, whereas other parameters and analysis techniques were proposed in [28], [29] and in references therein. These results can be applied to derive similar universality results as well as improve the scaling laws of the proposed ensemble.

**Replacing high-rate outer codes with fixed-rate codes.** The outer LDPC ensembles used were of a rate that approaches 1 with the decrease in the gap-to-capacity $\Delta$. This is in contrast to spatially coupled codes, where the underlying

[11] This is, in fact, a finite-memory stochastic process, due to the finiteness of the constraint length of the inner convolutional code.

regular LDPC code is of close-to-capacity rate. A simple means allowing the outer LDPC ensemble, in our scheme, to operate at any fixed rate is by splitting the coded bits of the LDPC coder into two subsets, such that a subset of rate close to 1 is fed to the convolutional coder, whereas the other bypasses the convolutional coder and is declared erased. This induces an effective "nearly BEC". Unfortunately, this construction prohibits the usage of *regular* LDPC ensembles. Note however, that the locations of the injected erasures need not be random and can be chosen to enhance the performance of the BP decoder. Specifically, it would be interesting to investigate whether a judicious design of the injection locations could prevent the failure of BP decoding of regular LDPC codes due to stopping sets; thus, attaining ML performance of the underlying regular LDPC codes.

## References

[1] G. D. Forney Jr., "Concatenated codes," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 1965.

[2] V. Guruswami and P. Indyk, "Linear-time encodebale/decodable codes with near-optimal rate," *IEEE Trans. Info. Theory*, vol. 51, pp. 3393–3400, 2005.

[3] A. Barg and G. Zémor, "Error exponent of expander codes," *IEEE Trans. Info. Theory*, vol. 48, pp. 1725–1729, 2002.

[4] ——, "Concatenated codes: Serial and parallel," *IEEE Trans. Info. Theory*, vol. 51, pp. 1625–1634, 2005.

[5] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proc. 29th annual ACM Symp. Theory of Comp.*, 150–159, 1997.

[6] M. A. Shokrollahi, "Capacity-achieving sequences," *Codes, Systems, and Graphical Models (IMA Vol. in Math. and Its App.)*, vol. 47, pp. 153–166, Feb. 2001.

[7] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Info. Theory*, vol. 55, pp. 3051–3073, July 2009.

[8] A. J. Felström and K. S. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Trans. Info. Theory*, vol. 45, pp. 2181–2190, 1999.

[9] S. Kudekar, T. J. Richardson, and R. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Trans. Info. Theory*, submitted, June 2013. [Online]. Available: http://arxiv.org/abs/1201.2999

[10] D. Blackwell, L. Breiman, and A. J. Thomasian, "The capacity of a class of channels," *The Annals of Math. Stat.*, vol. 30, pp. 1229–1241, Dec. 1959.

[11] R. L. Dobrushin, "Optimal information transmission over a channel with unknown parameters," (in Russian) *Radiotekh. i Elektron.*, vol. 4, no. 12, pp. 1951–1956, Dec. 1959.

[12] J. Wolfowitz, "Simultaneous channels," *Arch. Rational Mech. Anal.*, vol. 4, pp. 371–386, 1960.

[13] A. Guillen i Fabregas, I. Land, and A. Martinez, "Extremes of error exponents," *IEEE Trans. Info. Theory*, vol. 59, pp. 2201–2207, 2013.

[14] M. Alsan, "Extremality properties for Gallager's random coding exponent," in *Proc. IEEE Int. Symp. on Info. Theory (ISIT)*, Cambridge, MA, USA, July 2012, pp. 2944–2948.

[15] A. Khandekar and R. McEliece, "A lower bound on the iterative decoding threshold of irregular LDPC code ensembles," in *Proc. 36th Conf. Information Sciences and Systems*, Princeton, NJ, USA, Mar. 2002.

[16] A. Khandekar, "Graph-based codes and iterative decoding," Ph.D. dissertation, California Institute of Technology, 2002.

[17] D. Burshtein and G. Miller, "Bounds on the performance of belief propagation decoding," *IEEE Trans. Info. Theory*, vol. 48, pp. 112–122, 2002.

[18] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Info. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[19] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Info. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[20] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat–accumulate codes," in *Proc. 2nd Int. Symp. Turbo Codes and Related Topics*, Sept. 2000, pp. 1–8.

[21] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Society for Industrial and App. Math. (SIAM)*, vol. 8, no. 2, pp. 300–304, 1960.

[22] R. C. Singleton, "Maximum distance q-nary codes," *IEEE Trans. Info. Theory*, vol. 10, pp. 116–118, Apr. 1964.

[23] R. G. Gallager, *Information Theory and Reliable Communication*. John Wiley & Sons, 1968.

[24] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*. New York: McGraw-Hill, 1979.

[25] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge: Cambridge University Press, 2008.

[26] A. Khina, Y. Yona, and U. Erez, "LDPC ensembles that universally achieve capacity under BP decoding: A simple derivation," Tech. Rep., Nov. 2014. [Online]. Available: www.eng.tau.ac.il/~anatolyk/concat_ldpc.pdf

[27] N. Shulman and M. Feder, "Improved error exponent for time-invariant and periodically time-variant convolutional codes," *IEEE Trans. Info. Theory*, vol. 46, pp. 97–103, 2000.

[28] I. Land, S. Huettinger, P. A. Hoeher, and J. B. Huber, "Bounds on information combining," *IEEE Trans. Info. Theory*, vol. 51, pp. 612–619, 2005.

[29] I. Sutskover, S. Shamai, and J. Ziv, "Constrained information combining: Theory and applications for LDPC coded systems," *IEEE Trans. Info. Theory*, vol. 51, pp. 612–619, 2005.