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

Anatoly Khina
EE—Systems Dept., TAU
Tel Aviv, Israel
Email: anatolyk@eng.tau.ac.il

Yair Yona
Dept. of EE, UCLA
Los Angeles, CA, USA
Email: yairyo99@ucla.edu

Uri Erez
EE—Systems Dept., TAU
Tel Aviv, Israel
Email: 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 code ensemble. In this work, we provide a simple derivation of a different LDPC code ensemble that approaches capacity under BP decoding, following the classical approach of serial concatenation. This LDPC code 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 output-symmetric (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 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.**

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* to be a Reed–Solomon one.[1]

The asymptotic encoding/decoding complexity of concatenated codes was subsequently reduced from polynomial to linear by replacing the outer Reed–Solomon codes with expander-based codes [2], [3]. While these works have established that approaching capacity with low complexity is in a theoretical sense possible, they are generally not considered practical and hence the search for practical codes (and decoders) remains an active area.

The goal of achieving capacity over the binary erasure channel (BEC) with practical 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 work of Luby *et al.* [4].

However, the question of whether codes having an LDPC matrix representation can achieve capacity under BP decoding for general binary-input memoryless output-symmetric (BMS) channels, remained open until an affirmative answer was provided by Kudekar *et al.* [5] for the special class of spatially-coupled LDPC codes, first introduced by Felström and Zigangirov [6]. Moreover, this class was shown to be *universal*: It achieves capacity simultaneously (compound channel setting) for the class of BMS channels with a given capacity.

In this work, we provide a simple derivation of an LDPC ensemble, which is different from the spatially-coupled ensemble, 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 code. 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 (B-parameter). Coupled with the elegant result by Khandekar [7], the performance guarantee of an LDPC code under BP decoding is dictated solely by this parameter. This, in turn, allows to translate the performance under BP decoding of LDPC codes over the BEC, to any BMS channel. We note that, since for convolutional codes belief propagation (BCJR algorithm) amounts to (bit-wise) *maximum a-posteriori* decoding [8], the proposed concatenated scheme achieves capacity under BP decoding. Furthermore, by invoking the extremal properties of the binary-symmetric channel (BSC) of [9], designing the inner convolutional code for the BSC guarantees the universality of the scheme over the class of BMS channels with a given capacity. Finally, since the outer LDPC ensemble is of high-rate, regular LDPC ensembles suffice to obtain the desired result over the resulting factor graph.

II. Building Blocks

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

[1]Forney proposed to use an inner convolutional code. However, since a bit-error rate analysis for convolutional codes was not available at the time, he used bounds on block codes in the analysis. As will become evident in the sequel, in Lemma 1, such an anaysis is possible also in our case, but results in bounds that are much less tight.

[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.

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, applicable to time-varying convolutional codes. For this, we review the results of [10, Chapter 5] on the error exponent and BER of convolutional codes, and combine them with the recent results of [9] on the universality of error exponents of block codes.

We use the notation and definitions of [10, Part 2] for convolutional codes. A compact representation (and implementation) of a convolutional code is via a shift register: The delay-line (shift register) length is denoted by K, whereas its width b is the number of information bits entering the shift register at each time instant. Thus, the total memory size is equal to Kb bits. At each time instant, 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 $r = b/n$ bits per channel use. In general, these functionals may change at each time instant, resulting in a *time-varying* convolutional code.

Remark 1: The analysis in [10] considers an infinite stream of information and resulting code bits. Nevertheless, the derived upper bounds on the BER remain valid when zero-terminating the convolutional code to a finite length, at the price of a reduced rate that may be made arbitrarily small.

Denote the channel capacity by C. The following proposition is due to Viterbi and Yudkin (VY) [10, Chapter 5].

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

$$P_{\mathrm{b}} \leq \left(2^b - 1\right) \frac{2^{-K\frac{b}{r}E_{\mathrm{VY}}(r,\epsilon)}}{\left[1 - 2^{-\epsilon\frac{b}{r}E_{\mathrm{VY}}(r,\epsilon)}\right]^2} \quad ,$$

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

$$E_{\mathrm{VY}}\left(r,\epsilon\right) = \max_{0\leq\rho\leq\min((1-\epsilon)E_0(\rho)/r,1)} E_0\left(\rho\right) \tag{1a}$$

$$= \begin{cases} R_0 & 0 \leq r \leq R_0(1-\epsilon) \\ E_0(\rho_0) & R_0(1-\epsilon) < r \leq C(1-\epsilon) \end{cases}, \tag{1b}$$

ρ_0 is the largest solution of $\rho r = (1-\epsilon)\, E_0\left(\rho\right)$,[3]

$$E_0\left(\rho\right) \triangleq -\log_2\left(\sum_y \left[\sum_x \frac{1}{2} p\left(y|x\right)^{\frac{1}{1+\rho}}\right]^{1+\rho}\right)$$

is Gallager's E_0 (see, *e.g.*, [10, Ch. 2]), and $R_0 \triangleq E_0\left(\rho = 1\right)$ is the corresponding cutoff rate.

We next show that the VY error exponent of convolutional codes over any BMS channel is lower bounded by that of the BSC with the same capacity. This result is based upon its parallel for block codes [9].

To that end, denote the compound channel [11], 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 (1b) — by VY; superscripts indicate to which channel these error exponents refer.

Theorem 1 ([9]): Let c be any BMS channel with capacity C, $c \in \mathsf{BMS}\left(C\right)$. Then, the following relations hold

$$E_0^{(c)}\left(\rho\right) \geq E_0^{\mathsf{BSC}}(\rho) \tag{2a}$$

$$R_0^{(c)} \geq R_0^{\mathsf{BSC}}(C) \tag{2b}$$

$$E_G^{(c)}\left(r\right) \geq E_G^{\mathsf{BSC}}(r)\,, \quad \forall r \in [0, C]\,. \tag{2c}$$

The optimization problems in (1a) gives rise to the following relation between VY and block-code error exponents, which implies that capacity can be achieved by convolutional codes.

Lemma 1: Let c be any BMS channel with capacity C, $c \in \mathsf{BMS}\left(C\right)$. Then, for any $0 \leq r < (1-\epsilon)\, C$ and $0 < \epsilon < 1$:

$$E_{\mathrm{VY}}^{(c)}(r,\epsilon) \geq E_G^{(c)}\left(\frac{r}{1-\epsilon}\right) > 0\,.$$

Similarly to the Gallager exponent for block codes, the next lemma states that the VY error exponent of a BMS channel is at least as good as that of a BSC with the same capacity.

Lemma 2: Let c be any BMS channel with capacity C, $c \in \mathsf{BMS}\left(C\right)$. Then, for any $0 \leq r < (1-\epsilon)\, C$ and $0 < \epsilon < 1$:

$$E_{\mathrm{VY}}^{(c)}(r,\epsilon) \geq E_{\mathrm{VY}}^{\mathsf{BSC}}(r,\epsilon)\,.$$

By combining Lemma 2 and Proposition 1 we have the following.

Corollary 1: The BER of a random time-varying convolutional code over the compound channel $\mathsf{BMS}(C)$ is upper bounded (universally) as follows:

$$P_{\mathrm{b}} \leq \min_{\epsilon\in(0,1)} \left(2^b - 1\right) \frac{2^{-K\frac{b}{r}E_{\mathrm{VY}}^{\mathsf{BSC}}(r,\epsilon)}}{\left[1 - 2^{-\epsilon\frac{b}{r}E_{\mathrm{VY}}^{\mathsf{BSC}}(r,\epsilon)}\right]^2} \triangleq \tilde{P}_{\mathrm{b}}(K,r,b)\,.$$

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

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 3 in terms of an upper bound on the achievable BER after ℓ iterations; this bound is based, in turn, on the density-evolution (DE) equations for the BEC, with the erasure probability replaced by the B-parameter of the BMS channel, and is given as part of the proof of Theorem 4.2 in [7] (see also [12]). We base our notation, as well as the conditions needed for the DE analysis to hold (the tree assumption, the concentration property *etc.*), on [13].

Proposition 2: Consider a BMS channel $P\left(y|x\right)$, where $x \in \{0,1\}$. Generate an LDPC code ensemble w.r.t. variable- and check-node degree distributions $\lambda\left(x\right)$, $\rho\left(x\right)$ respectively, and denote by $x^{(\ell)}$ the log-likelihood ratio (LLR) of a variable-node message in iteration $\ell \geq 0$. W.l.o.g., assume that the zero

[3]The constraint $0 \leq \rho < E_0(\rho)/r$ means that we do not take into consideration the cases in which $\rho \geq E_0\left(\rho\right)/r$.

codeword is transmitted. Then, in iteration $\ell+1$ we have

$$E\left[e^{-\frac{x^{(\ell+1)}}{2}}\right] \leq Z^{(\ell+1)} \triangleq B \cdot \lambda\left(1-\rho\left(1-Z^{(\ell)}\right)\right), \quad (3)$$

where $Z^{(0)} = B$, $x^{(0)} \triangleq \log_2\left(\frac{P(y|0)}{P(y|1)}\right)$ is the initial LLR variable node message, and $B \triangleq \sum_y \sqrt{P(y|0)\,P(y|1)}$ is the B-parameter.[4]

Proposition 2 gives rise to the following simple bound on the BER, formally proved in [12].

Lemma 3: The BER in iteration ℓ is upper bounded as

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

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

A threshold [13, Ch. 4] for the upper bound on the DE equations is defined as the largest B-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. Regular LDPC Codes Achieve Capacity Under BP Decoding over Almost-Clean Channels

In [2] it has been shown that by considering an algebraic "near-MDS" outer 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 blocklength. 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 code with a rate very close to 1. However, the mechanism that enables the outer code in our case to reduce the error probability as desired is the existence of a threshold for the bound on the DE equation (3). The inner code induces a transition probability for the outer code such that the B-parameter B is small enough to guarantee that $E\left[\exp\left\{-x^{(\ell+1)}/2\right\}\right]$ approaches zero as ℓ increases.

The next lemma shows that the desired result can be achieved by regular LDPC codes.

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 (3) for this ensemble.

Proof: For a regular ensemble of LDPC codes the upper bound for the DE equation in iteration ℓ takes the following form (see, *e.g.*, [13, Ch. 4]):

$$Z^{(\ell+1)} = B \cdot \left(1-\left(1-Z^{(\ell)}\right)^{d_c-1}\right)^{d_v-1}. \quad (4)$$

[4]The summation over y is replaced by an integral for continuous channels.

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 (4) and also considering $Z^{(\ell)} \ll 1$ we get that $\lim_{\ell\to\infty} Z^{(\ell)} = 0$. Assuming $Z^{(\ell)} \ll 1$, (4) can be approximated via the first non-zero term of its Taylor series expansion as

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

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

III. 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, where the inner code is a convolutional code whose delay-line 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 Sec. III-A, and show that it achieves the desired result. We then argue, in Sec. III-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 M.
2) Interleaves systematically the output of the LDPC coder: The interleaver accumulates rL consecutive outer-code codewords of length M, such that they comprise the rows of an $rL \times M$ matrix, where r and L are the rate and blocklength of the inner code, respectively. The columns of the matrix are then fed to the inner coder, one by one.
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 convolutional code (which is fixed for a given gap-to-capacity), 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: Undoes the encoder's interleaving.

Outer code decoding: Applies BP decoding for the outer LDPC code of length M, 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; see Sec. III-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 $\mathsf{BMS}(C)$ 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 (fixed) delay-line length K, such that $\tilde{P}_{\mathrm{b}}(K, r, b)$ of Corollary 1 satisfies[5]

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

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

Proof: We first show that random convolutional code and LDPC codes can be generated with the desired parameters.

As shown in [9], the B-parameter of any BMS channel with a given capacity is upper bounded by that of the BSC of the same capacity. Moreover, the B-parameter of a BSC monotonically decreases with capacity. Therefore, the B-parameter B of the effective BMS channel induced by the LLRs of the inner code is upper bounded by the B-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 $\tilde{P}_{\mathrm{b}}(K, r, b)$. This leads, in turn, to the upper bound $B \leq B_0$. By choosing K large enough, $\tilde{P}_{\mathrm{b}}(K, r, b)$, and hence also B_0, can be made as small as desired, according to Corollary 1.

LDPC code ensembles of rate R/r that have a threshold that is larger than B_0 over the BEC are well known to exist [4], [7], [13]. Proposition 2 and Lemma 3 guarantee that these ensembles achieve a BER as small as desired over all BMS channels with the same B-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 B-parameter B that is upper bounded by B_0. The deinterleaving guarantees that this channel is memoryless. 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 Δ 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 $\mathsf{BSC}(C)$ and an outer code that is designed for a BEC, suffices to prove the universality of the scheme over $\mathsf{BMS}(C)$.

[5] B_0 is the resulting B-parameter of an effective BSC with transition probability $\tilde{P}_{\mathrm{b}}(K, r, b)$.

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

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

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 below this threshold establishes the desired result with regular ensembles. ∎

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. A standard way for obtaining a high rate code from a lower rate "mother code" is via puncturing, if one is willing to sacrifice regularity. Since the outer code is designed for an erasure channel, (random) puncturing amounts to increasing the erasure probability. Moreover, judicious puncturing may further enhance performance; see [14], [15].

Lastly, 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 with a systematic irregular repeat–accumulate (IRA) code. In fact, the results of Sec. II-B were also introduced in the Ph.D. thesis of Khandekar [7], and were shown to be valid both for general LDPC codes and for IRA codes.

B. Achieving Capacity under Belief Propagation Decoding

Here we consider a slightly generalized variant of the encoder of Algorithm 1: We use rL independent LDPC code ensembles of the same parameters. That is, the rows in the interleaver of Algorithm 1 are drawn from independently generated LDPC codebooks. We note that all the results of Sec. III-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 universally achieves the channel capacity of $\mathsf{BMS}(C)$ under BP decoding over the *factor graph* of the overall code, which results from the factor graphs 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 M for the outer layer, and M time-varying zero-terminated convolutional codes each of length L for the inner code.[6] Denote the j-th symbol of the i-th LDPC codeword by $x_{i,j}$, where $1 \leq i \leq rL$ and $1 \leq j \leq M$. 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.*

The following assumption will be used in the BP analysis.

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

Assumption 1 (Extended tree assumption): The depth-ℓ extended tree assumption states that variable node $x_{i,j}$ shares no loops with the subtrees of depth ℓ 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.* [13, Ch. 3]) along with an "extension". The regular tree assumption states that $x_{i,j}$ shares no loop with the subtrees of depth ℓ 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 convolutional code codeword.

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 [12].

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 M of the LDPC code ensembles to be sufficiently large, such that the depth-ℓ extended tree assumption is satisfied with probability greater than $1 - \epsilon'$ over the factor graph induced by the overall code.

Remark 7: The length M of the LDPC code ensembles 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 M required for the analysis of full BP to hold is greater than that needed for the analysis of Algorithm 2 of Sec. III-A.

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

Algorithm 3 (Belief-propagation decoder): Variable and LDPC code nodes use the standard sum-product algorithm message-update rule (*cf.* [13]). The convolutional code node carries BCJR decoding with non-uniform prior that is dictated by the messages coming from the LDPC codes.[7]

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

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. Since BP decoding is optimal under the tree assumption (see, *e.g.*, [13]), it follows that the BER achievable by Algorithm 3 is upper bounded by the BER achievable by Algorithm 2. ■

Theorem 2: For any gap to 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 $\mathsf{BMS}(C)$.

Specifically, this is achieved by a convolutional code of rate $r \in (R, C)$ and a long enough (fixed) delay-line length K, such that $\tilde{P}(K, r, b)$ of Corollary 1 satisfies

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

and LDPC code ensemble of rate R/r whose threshold over a BEC is above B_0. By taking the length M of this ensemble to be 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 M of the LDPC code ensemble to be large 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. Lemmas 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 M, ℓ. Finally note that, as in Lemma 5, the rate of the overall code is R, as desired. ■

Corollary 3: 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 delay-line length K, under the two-stage message-passing decoding of Algorithm 2 or the BP decoding of Algorithm 3.

The proof follows by using the *concentration phenomenon* w.r.t. the inner convolutional and the outer LDPC codes.

[7] BP decoding of convolutional codes amounts to BCJR decoding [8].

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

[4] 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.

[5] S. Kudekar, T. J. Richardson, and R. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Trans. Info. Theory*, submitted, June 2013.

[6] 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.

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

[8] 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, 2001.

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

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

[11] A. Lapidoth and P. Narayan, "Reliable communication under channel uncertainty," *IEEE Trans. Info. Theory*, vol. 44, pp. 2148–2177, 1998.

[12] A. Khina, Y. Yona, and U. Erez, Tech. Rep., Nov. 2014. [Online]. Available: www.eng.tau.ac.il/~anatolyk/papers/journal/concat_ldpc.pdf

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

[14] H. D. Pfister, I. Sason, and R. Urbanke, "Capacity-achieving ensembles for the binary erasure channel with bounded complexity," *IEEE Trans. Info. Theory*, vol. 51, pp. 2352–2379, 2005.

[15] H. Pishro-Nik and F. Fekri, "Results on punctured low-density parity-check codes and improved iterative decoding techniques," *IEEE Trans. Info. Theory*, vol. 53, pp. 599–614, 2007.