Maxwell Construction: The Hidden Bridge between Iterative and Maximum a Posteriori Decoding

Cyril Méasson† , Andrea Montanari* and Rüdiger Urbanke ‡

***Abstract*— There is a fundamental relationship between belief propagation and maximum a posteriori decoding. A decoding algorithm, which we call the Maxwell decoder, is introduced and provides a constructive description of this relationship. Both, the algorithm itself and the analysis of the new decoder are reminiscent of the Maxwell construction in thermodynamics. This paper investigates in detail the case of transmission over the binary erasure channel, while the extension to general binary memoryless channels is discussed in a companion paper.**

***Index Terms*— belief propagation, maximum a posteriori, maximum likelihood, Maxwell construction, threshold, phase transition, Area Theorem, EXIT curve, entropy**

I. Introduction

It is a key result, and the starting point of iterative coding, that belief propagation (BP) is optimal on trees. See, e.g., [5]–[8]. However, trees with bounded state size appear not to be powerful enough models to allow transmission arbitrarily close to capacity. For instance, it is known that in the setting of standard binary Tanner graphs the error probability of codes defined on trees is lower bounded by a constant which only depends on the channel and the rate of the code [9], [10]. The general wisdom is therefore to apply BP decoding to graphs with loops and to consider this type of decoding as a (typically) strictly suboptimal attempt to perform maximum a posteriori (MAP) bit decoding. One would therefore not expect any link between the BP and the MAP decoder except for the obvious suboptimality of the BP decoder.

This contribution demonstrates that there is a fundamental relationship between BP and MAP decoding which appears in the limit of large blocklengths. This relationship is furnished by the so-called Maxwell (M) decoder. The M decoder combines the BP decoder with a "guessing" device to perform MAP decoding. It is possible to analyze the performance of the M decoder in terms of the EXIT curve introduced in [11]. This analysis leads to a precise characterization of how difficult it is to convert the BP decoder into a MAP decoder and this "gap" between the MAP and BP decoder has a pleasing graphical interpretation in terms of an area under the EXIT curve.[1] Further, the MAP threshold is determined by a balance between two areas representing the number of guesses and the reduction in uncertainty, respectively. The analysis gives also rise to a generalized Area Theorem, see also [12], and it provides an alternative tool for proving area-like results.

The concept of a "BP decoder with guesses" itself is not new. In [13] the authors introduced such a decoder in order to improve the performance of the BP decoder. Our motivation though is quite different. Whereas, from a practical point of view, such enhancements work best for relatively small code lengths, or to clean up error floors, we are interested in the asymptotic setting in which the unexpected relationship between the MAP decoder and the BP decoder emerges.

A. Preliminaries

Assume that transmission takes place over a binary erasure channel with parameter ϵ, call it BEC(ϵ). More precisely, the transmitted bit x_i at time i, $x_i \in \mathcal{X} \triangleq \{0,1\}$, is erased with probability ϵ. The channel output is the random variable Y_i which takes values in $\mathcal{Y} \triangleq \{0,*,1\}$. To be concrete, we will exemplify all statements using Low-Density Parity-Check (LDPC) code ensembles [14]. However, the results extend to other ensembles like, e.g., Generalized LDPC or turbo codes, and we will state the results in a general form. For an in-depth introduction to the analysis of LDPC ensembles see, e.g., [15]–[18]. For convenience of the reader, and to settle notation, let us briefly review some key statements. The degree distribution (dd) pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\sum_j \lambda_j \mathtt{x}^{j-1}, \sum_j \rho_j \mathtt{x}^{j-1})$ represents the degree distribution of the graph from the *edge* perspective. We consider the ensemble LDPC(λ,ρ,n) of such graphs of length n and we are interested in its asymptotic average performance (when the blocklength $n \to \infty$). This ensemble can equivalently be described by $\Xi \triangleq (\Lambda(\mathtt{x}), \Gamma(\mathtt{x})) = (\sum_j \Lambda_j \mathtt{x}^j, \sum_j \Gamma_j \mathtt{x}^j)$, which is the dd pair from the *node* perspective[2]. An important characteristic of the ensemble LDPC(λ,ρ,n) is the *design rate* $r \triangleq 1 - \int\rho / \int\lambda = 1 - \Lambda'(1)/\Gamma'(1)$. We will write $r = r(\lambda,\rho)$ or $r = r(\Lambda,\Gamma)$ whenever we regard the design rate as a function of the degree distribution pair.

The BP threshold, call it $\epsilon^{\text{BP}} = \epsilon^{\text{BP}}(\lambda,\rho)$, is defined in [15]–[18] as $\epsilon^{\text{BP}} \triangleq \sup\{\epsilon \in [0,1] : \epsilon\lambda(1-\rho(1-\mathtt{x})) < \mathtt{x}, \forall \mathtt{x} \in (0,1]\}$. Operationally, if we transmit at $\epsilon < \epsilon^{\text{BP}}$ and use a BP decoder, then all bits except possibly a sub-linear fraction can be recovered when $n \to \infty$. On the other hand, if $\epsilon \geq \epsilon^{\text{BP}}$, then a fixed fraction of bits remains erased after BP

† EPFL, School for Computer and Communication Sciences, CH-1015 Lausanne, Switzerland. E-mail: cyril.measson@epfl.ch

∗ ENS, Laboratoire de Physique Théorique, F-75231 Paris, France. E-mail: montanar@lpt.ens.fr

‡ EPFL, School for Computer and Communication Sciences, CH-1015 Lausanne, Switzerland. E-mail: ruediger.urbanke@epfl.ch

Parts of the material were presented in [1]–[4].

[1]The EXIT curve is here the EXIT curve associated to the iterative coding system and not to its individual component codes. This differs from the original EXIT chart context presented in [11].

[2]The changes of representation are obtained via $\Lambda(\mathtt{x}) = (1/\int\lambda)\int_0^{\mathtt{x}} \lambda(u)\mathrm{d}u$, $\Gamma(\mathtt{x}) = (1/\int\rho)\int_0^{\mathtt{x}} \rho(u)\mathrm{d}u$, $\lambda(\mathtt{x}) = \Lambda'(\mathtt{x})/\Lambda'(1)$ and $\rho(\mathtt{x}) = \Gamma'(\mathtt{x})/\Gamma'(1)$.

decoding when $n \to \infty$. In a similar manner we can define the MAP threshold. This threshold was first found via the replica method in [19]. Further, in [2] a simple counting argument leading to an upper bound for this threshold was given. The argument is explained and sharpened in Sec. V. In this paper we develop the point of view taken in [1]. The reference quantity is then the extrinsic[3] entropy, in short EXIT.[4] The EXIT curve associated to the i^{th} variable is a function of the channel entropy and it is defined as $H(X_i \,|\, Y_{[n]\setminus\{i\}})$. Hereby, X_i represents the i^{th} input bit and, for $S \subseteq [n] \triangleq \{1, \dots, n\}$, X_S represents the $|S|$-tuple of all bits indexed by S. For notational simplicity, let us write $X_{\sim i} = X_{[n]\setminus\{i\}}$ when a single bit is omitted and $X = X_{[n]}$ for the entire vector. The uniformly averaged quantity $\frac{1}{n}\sum_{i=1}^{n} H(X_i \,|\, Y_{\sim i})$ is called the EXIT function. Recall that if there is a uniform prior on the set of hypotheses then the maximum a posteriori and the maximum likelihood decoding rule are identical. Let $\Phi_i^{\text{MAP}} = \phi_i^{\text{MAP}}(Y_{\sim i})$ denote the extrinsic MAP bit estimate (sometimes called extrinsic information) associated to the i^{th} bit. This can be any sufficient statistics for X_i given $Y_{\sim i}$. Since we deal with binary variables, we can always think of it as the conditional expectation $\phi_i^{\text{MAP}}(Y_{\sim i}) \triangleq \mathbb{E}[X_i | Y_{\sim i}]$. Observe that $H(X_i \,|\, Y_{\sim i}) = H(X_i \,|\, \Phi_i^{\text{MAP}})$.

B. Overview of Results

Consider a dd pair (λ, ρ) and the corresponding sequence of ensembles LDPC(n, λ, ρ) of increasing length n. Fig. 1 shows the asymptotic EXIT curve for the regular dd pair $(\lambda(x) = x^2, \rho(x) = x^5)$.

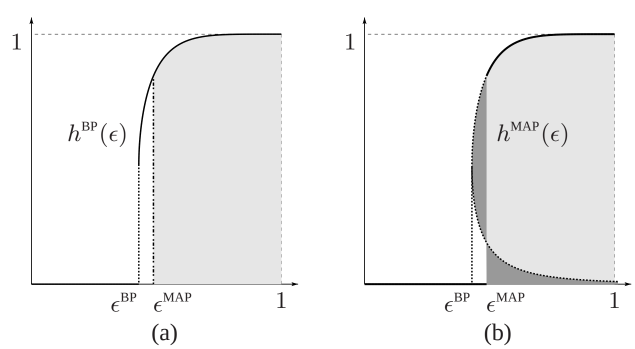


Fig. 1. BP and MAP EXIT curves for the dd pair $(\lambda(x) = x^2, \rho(x) = x^5)$. (a) BP EXIT curve $h^{\text{BP}}(\epsilon)$: its parametric equation is stated in (1). It is zero until ϵ^{BP} at which point it jumps. It further continues smoothly until it reaches one at $\epsilon = 1$. (b) MAP EXIT curve $h^{\text{MAP}}(\epsilon)$. Note that the figure (b) includes also the "spurious" branch of Eq. (1). The spurious branch corresponds to unstable fixed points. The MAP threshold is determined by the balance of the two dark gray areas.

Formally, this EXIT curve is $h^{\text{MAP}}(\epsilon) \triangleq \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} H(X_i \,|\, Y_{\sim i}(\epsilon)) = \lim_{n\to\infty} \frac{1}{n} H(X_i \,|\, \Phi_i^{\text{MAP}})$. Its main characteristics are as follows: the function is zero below the MAP threshold ϵ^{MAP}, it jumps at ϵ^{MAP} to a non-zero value and continues then smoothly until it reaches one for $\epsilon = 1$. The area under the EXIT curve equals the rate of the code, see [12]. Compare this to the equivalent function of the BP decoder which is also shown in Fig. 1. The BP EXIT curve $h^{\text{BP}}(\epsilon) \triangleq \lim_{n\to\infty} \frac{1}{n} H(X_i \,|\, \Phi_i^{\text{BP}})$ corresponds to running a BP decoder on a very large graph until the decoder has reached a fixed point. The extrinsic entropy of the bits at this fixed point gives the BP EXIT curve. This curve is given in parametric form by

$$\left(\frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}, \Lambda(1-\rho(1-\mathtt{x})) \right), \tag{1}$$

where $\mathtt{x}$ indicates the erasure probability of the variable-to-check messages. To see this, note that when transmission takes place over BEC(ϵ), then the BP decoder reaches a fixed point $\mathtt{x}$ which is given by the solution of the density evolution (DE) equation $\epsilon\lambda(1-\rho(1-\mathtt{x}))$. We can therefore express ϵ as $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}$. Now the average extrinsic probability that a bit is still erased at the fixed point is equal to $\Lambda(1-\rho(1-\mathtt{x}))$. Note that the BP EXIT curve is the trace of this parametric equation for $\mathtt{x}$ starting at $\mathtt{x} = 1$ until $\mathtt{x} = \mathtt{x}^{\text{BP}}$. This is the critical point and $\epsilon(\mathtt{x}^{\text{BP}}) = \epsilon^{\text{BP}}$. Summarizing, the BP EXIT curve is zero up to the BP threshold ϵ^{BP} where it jumps to a non-zero value and then continues smoothly until it reaches one at $\epsilon = 1$.

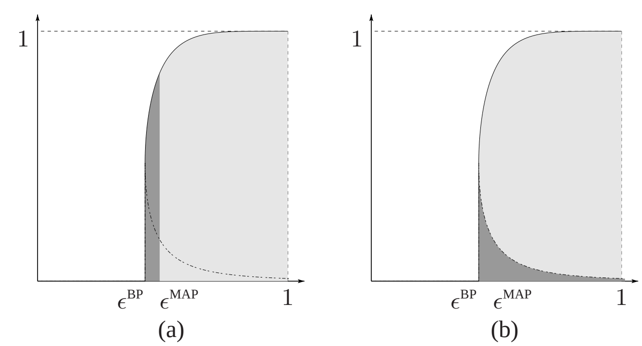


Fig. 2. Balance of areas for the Maxwell decoder between the number of guesses in (a) and the number of contradictions in (b). The two dark gray areas are equal at the MAP threshold. These two areas differ from the areas indicated in Fig. 1 only by a common part.

In [1] it was pointed out that for the investigated cases the following two curious relationships between these two curves hold: First, the BP and the MAP curve coincide above ϵ^{MAP}. Second, the MAP curve can be constructed from the BP curve in the following way. If we draw the BP curve as parameterized in (1) not only for $\mathtt{x} \in [\mathtt{x}^{\text{BP}}, 1]$ but also for $\mathtt{x} \in (0, \mathtt{x}^{\text{BP}})$ we get the curve shown in the right picture of Fig. 1. Notice that the branch for $\mathtt{x} \in (0, \mathtt{x}^{\text{BP}})$ corresponds to unstable fixed points under BP decoding. Moreover, the fraction of erased messages $\mathtt{x}$ decreases along this branch when the erasure probability is increased and it satisfies $\epsilon(\mathtt{x}) > \epsilon$. Because of these peculiar features, it is usually considered as "spurious". To determine the MAP threshold take a vertical line at $\epsilon = \epsilon^{\text{BP}}$ and shift it to the right until the area which lies to the left of this line and is enclosed by the line and the BP EXIT curve is equal to the area which lies to the right of the line and is enclosed by the line and the BP EXIT curve (these areas are indicated in dark gray in the picture). This unique point determines the MAP

[3] The term extrinsic is used when the observation of the bit itself is ignored, see [20], [21].

[4] The term EXIT , introduced in [11], stands for extrinsic (mutual) information transfer. Rather than using mutual information we opted to use entropies which in our setting simply means one minus mutual information. It is natural to use entropy in the setting of the binary erasure channel since the parameter ϵ itself represents the channel entropy.

threshold. The MAP EXIT curve is now the curve which is zero to the left of the threshold and equals the iterative curve to the right of this threshold. In other words, the MAP threshold is determined by a balance between two areas. It turns out that there is an operational meaning to this balance condition. We define the so-called Maxwell (M) decoder which performs MAP decoding by combining BP decoding with guessing. The dark gray areas in in the right picture of Fig. 2 differ from the ones in Fig. 1 only by a common part. We can show that the gray area on the left is connected to the number of "guesses" the M decoder has to venture, while the gray area on the right represents the number of "confirmations" regarding these guesses. The MAP threshold is determined by the condition that the number of confirmations balances the number of guesses (i.e., that each guess is confirmed), and therefore the two areas are equal: in other words, at the MAP threshold (and below) there is just a single codeword compatible with the channel received bits.

The EXIT curves depicted in Fig. 1 are representative for a large family of degree distributions, e.g., those of regular LDPC ensembles. But more complicated scenarios are possible. Fig. 3 depicts a slightly more general case in which the BP EXIT curve and the MAP EXIT curve have two jumps. As can be seen from this figure, the same kind of balance condition holds in this case *locally* and it determines the position of each jump.

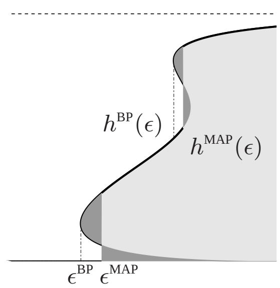


Fig. 3. BP (dashed and solid line) and MAP (thick solid line) EXIT curves for the ensemble discussed in Examples 7 and 10. Both curves have two jumps. The two jumps of the MAP EXIT curve are both determined by a local balance of areas.

C. Paper Outline

We start by considering the conditional entropy $H(X \mid Y)$, where X is the transmitted codeword and Y the received sequence, and we derive the so-called Area Theorem for finite-length codes. When applying the Area Theorem to the binary erasure channel, the notion of EXIT curve enters explicitly. Next, we show that when the codes are chosen randomly from a suitable defined ensemble then the individual conditional entropies and EXIT curves concentrate around their ensemble averages. This is the first step towards the asymptotic analysis.

We continue by defining the three asymptotic EXIT curves of interest. These are the (MAP) EXIT curve, the BP EXIT curve, and the EBP EXIT curve (which holds extended BP EXIT and includes the spurious branch). We show that the Area Theorem remains valid in the asymptotic setting. As an immediate consequence we will see that for some classes of ensembles (roughly those for which the stability condition determines the threshold) BP decoding coincides with MAP decoding.

We then present a key point of the paper, which is the derivation of an upper-bound for the MAP threshold. Several examples illustrate this technique and lead to suggests the tightness of the bound.

The same result is recovered through a counting argument that, supplemented by a combinatorial calculation, implies the tightness of the bound.

Finally, we introduce the so-called M decoder which provides a unified framework for understanding the connection between the BP and the MAP decoder. A closer analysis of the performance of the M decoder will allow us to prove a refined upper bound on the MAP threshold and it will give rise to a pleasing interpretation of the MAP threshold as that parameter in which two areas under the EBP EXIT curve are in balance.

We conclude the paper by discussing some applications of our method.

II. Finite-Length Codes: Area Theorem and Concentration

Let X be the transmitted codeword and let Y be the received word. The conditional entropy $H(X \mid Y)$ is of fundamental importance if we consider the question whether reliable communication is possible. Let us see how this quantity appears naturally in the context of decoding. To this end, we first recall the original Area Theorem as introduced in [12].

Theorem 1 (Area Theorem): Let X be a binary vector of length n chosen with probability $p_X(x)$ from a finite set. Let Y be the result of passing X through BEC(ϵ). Let Ω be a further observation of X so that $p_{\Omega \mid X,Y}(\omega \mid x, y) = p_{\Omega \mid X}(\omega \mid x)$. To emphasize that Y depends on the channel parameter ϵ we write $Y(\epsilon)$. Then

$$\frac{H(X \mid \Omega)}{n} = \int_0^1 \frac{1}{n} \sum_{i \in [n]} H(X_i \mid Y_{\sim i}(\epsilon), \Omega) \mathrm{d}\epsilon. \tag{2}$$

The reader familiar with the original statement in [12] will have noticed that we have rephrased the theorem. First, we expressed the result in terms of entropy instead of mutual information. Second, the observations Y and Ω represent what in the original theorem were called the "extrinsic" information and the "channel," respectively.

In (2) the integration ranges from zero (perfect channel) to one (no information conveyed). The following is a trivial extension.

Theorem 2 (Area Theorem): Let X be a binary vector of length n chosen with probability $p_X(x)$ from a finite set. Let Y be the result of passing X through BEC(ϵ). Let Ω be a further observation of X so that $p_{\Omega \mid X,Y}(\omega \mid x, y) = p_{\Omega \mid X}(\omega \mid x)$. Then

$$\frac{H(X \mid Y(\epsilon^*), \Omega)}{n} = \int_0^{\epsilon^*} \frac{1}{n} \sum_{i \in [n]} H(X_i \mid Y_{\sim i}(\epsilon), \Omega) \mathrm{d}\epsilon.$$

Proof of Theorem 2: Let $Y^{(1)}$ be the result of passing X through BEC(ϵ) and $Y^{(2)}$ be the result of passing X through BEC(ϵ^*). Let Ω be the additional observation of X. Applying Theorem 1, with $Y = Y^{(1)}$ and with additional observation $(Y^{(2)}, \Omega)$, we have $p_{\Omega, Y^{(2)} \mid X, Y^{(1)}}(\omega, y^{(2)} \mid x, y^{(1)}) =$

$p_{\Omega, Y^{(2)} \,|\, X}(\omega, y^{(2)} \,|\, x)$, as required, so that we get

$$H(X \,|\, Y^{(2)}(\epsilon^*), \Omega) = \int_0^1 \sum_{i \in [n]} H(X_i \,|\, Y_{\sim i}^{(1)}(\epsilon), Y^{(2)}(\epsilon^*), \Omega) \mathrm{d}\epsilon.$$

Now note that

$$H(X_i \,|\, Y_{\sim i}^{(1)}(\epsilon), Y^{(2)}(\epsilon^*), \Omega) = \epsilon^* H(X_i \,|\, Y_{\sim i}(\epsilon\epsilon^*), \Omega).$$

This is true since the bits of $Y_{\sim i}^{(1)}(\epsilon)$ and $Y^{(2)}(\epsilon^*)$ are erased independently (so that the respective erasure probabilities multiply) and since $Y^{(2)}(\epsilon^*)$ contains the intrinsic observation of bit X_i, which is erased with probability ϵ^*. If we now substitute the right hand side of the last expression in our previous integral and make the change of variables $\epsilon' = \epsilon \cdot \epsilon^*$, Theorem 2 follows. ■

Assume that we allow each X_i to be passed through a different channel BEC(ϵ_i). Rather than phrasing our result specifically for the case of the BEC(ϵ_i), let us state the area theorem right away in its general form as introduced in [4]. In this paper we will only be interesting in the consequences as they pertain to transmission over the BEC(ϵ). The investigation of the general case is relegated to the companion paper [22].

In order to state this and subsequent results in a more compact form we introduce the following definition.

Definition 1 (Channel Smoothness): Consider a family of memoryless channels with input and output alphabets $\mathcal{X}$ and $\mathcal{Y}$, respectively, and characterized by their transition probability distribution functions (pdf's) $p_{Y \,|\, X}(y \,|\, x)$. If $\mathcal{Y}$ is discrete, we interpret $p_{Y \,|\, X}(\cdot \,|\, x)$ as a pdf with respect to the counting measure. If $\mathcal{Y}$ is continuous, $p_{Y \,|\, X}(y \,|\, x)$ is a density with respect to Lebesgue measure. Assume that the family is parameterized by ϵ, where ϵ takes values in some interval $I \subseteq \mathbb{R}$. The channel is said to be *smooth* with respect to the parameter ϵ if the pdf's $\{p_{Y \,|\, X}(y \,|\, x) : x \in \mathcal{X},\, y \in \mathcal{Y}\}$ are differentiable functions of $\epsilon \in I$.

Notice that, if a channel family is smooth, then several basic properties of the channel are likely to be differentiable with respect to the channel parameter. A basic (but important) example is the channel conditional entropy $H(Y|X) = \mathbb{E}[-\log\{p_{Y \,|\, X}(Y \,|\, X)\}]$ given a reference measure $p_X(x)$ on $\mathcal{X}$. Suppose that $\mathcal{Y}$ is finite, and that, for any $\epsilon \in I$, $p_{Y|X}(y|x) > 0$ for any $x \in \mathcal{X}$, $y \in \mathcal{Y}$. Then

$$\frac{\mathrm{d}H(X|Y)}{\mathrm{d}\epsilon} = \sum_{x,y} p_X(x) \log\left(\frac{1}{p_{Y|X}(y|x)}\right) \frac{\mathrm{d}p_{Y|X}}{\mathrm{d}\epsilon}(y|x)\,.$$

In other words, differentiability of $H(Y|X)$ follows from differentiability of $p_{Y|X}(y|x)$ and of $-x\log x$. In this paper we consider families of binary erasure channels which are trivially smooth with respect to the parameter ϵ.

Theorem 3 (General Area Theorem-[4]): Let X be a binary vector of length n chosen with probability $p_X(x)$ from a finite set. Let the channel from X to Y be memoryless, where Y_i is the result of passing X_i through a smooth channel with parameter ϵ_i, $\epsilon_i \in I_i$. Let Ω be a further observation of X so that $p_{\Omega \,|\, X,Y}(\omega \,|\, x, y) = p_{\Omega \,|\, X}(\omega \,|\, x)$. Then

$$\mathrm{d}H(X \,|\, Y, \Omega) = \sum_{i=1}^{n} \frac{\partial H(X_i \,|\, Y, \Omega)}{\partial \epsilon_i} \mathrm{d}\epsilon_i. \tag{3}$$

Proof: For $i \in [n]$, the entropy rule gives $H(X \,|\, Y, \Omega) = H(X_i \,|\, Y, \Omega) + H(X_{\sim i} \,|\, X_i, Y, \Omega)$. We have $p_{X_{\sim i} \,|\, X_i, Y, \Omega} = p_{X_{\sim i} \,|\, X_i, Y_{\sim i}, \Omega}$ since the channel is memoryless and $p_{\Omega \,|\, X, Y} = p_{\Omega \,|\, X}$. Therefore, $H(X_{\sim i} \,|\, X_i, Y, \Omega) = H(X_{\sim i} \,|\, X_i, Y_{\sim i}, \Omega)$ and $\frac{\partial H(X \,|\, Y, \Omega)}{\partial \epsilon_i} = \frac{\partial H(X_i \,|\, Y, \Omega)}{\partial \epsilon_i}$. From this the total derivate as stated in (3) follows immediately. ■

Alternative proof of Theorem 2: Keeping in mind that transmission takes place over a binary erasure channel, we write

$$H(X_i \,|\, Y, \Omega) = \sum_{y_i \in \{0,*,1\}} p_{Y_i}(y_i) H(X_i \,|\, Y_i = y_i, Y_{\sim i}, \Omega)\,.$$

The terms corresponding to $y_i \in \{0, 1\}$ vanish because X_i is then completely determined by the channel output. The remaining term yields $H(X_i \,|\, Y, \Omega) = \epsilon_i H(X_i \,|\, Y_{\sim i}, \Omega)$, because $p_{Y_i}(*) = \epsilon_i$, and the occurrence at the channel output of an erasure at position i is independent from X, $Y_{\sim i}$ and Ω. We can then write

$$\begin{aligned} \mathrm{d}H(X \,|\, Y(\epsilon), \Omega) &= \sum_{i \in [n]} \frac{\partial H(X_i \,|\, Y, \Omega)}{\partial \epsilon_i} \mathrm{d}\epsilon_i \\ &= \sum_{i \in [n]} H(X_i \,|\, Y_{\sim i}, \Omega) \mathrm{d}\epsilon_i, \end{aligned}$$

which, when we assume that $\epsilon_i = \epsilon$ for all $i \in [n]$, gives Theorem 2. ■

A few remarks are in order. First, the additional degree of freedom afforded by allowing an extra observation Ω is useful when studying the dynamical behavior of certain iterative coding schemes via EXIT chart arguments. (For example, in a parallel concatenation, Y typically represents the observation of the systematic bits and Ω represents the fixed channel observation of the parity bits.) For the purpose of this paper however, the additional observation Ω is not needed since we are not concerned by componentwise EXIT charts. We will therefore skip Ω in the sequel. Second, as emphasized in the last step in the previous proof, we can assume at this point, more generally, that the individual channel parameters ϵ_i are not the same but that the individual channels are all parametrized by a common parameter ϵ. For instance one may think of a families $\{\text{BEC}(\epsilon_i)\}$ where $\epsilon_i(\epsilon)$ are smooth functions of $\epsilon \in [0, 1]$. In the simplest case some parameter might be chosen to be constant. This degree of freedom allows for an elegant proof of Theorem 8.

One of the main aims of this paper is to investigate the MAP performance of sparse graph codes in the limit of large blocklengths. Our task is made much easier by realizing that we can restrict our study to the *average* such performance. More precisely, let $\mathtt{G}$ be chosen uniformly at random from LDPC(λ, ρ, n) and let $H_{\mathtt{G}}(X \,|\, Y)$ denote the conditional entropy for the code $\mathtt{G}$. We state the following theorems right away for general binary memoryless symmetric (BMS) channels.

Theorem 4 (Concentration of Conditional Entropy): Let $\mathtt{G}$ be chosen uniformly at random from LDPC(n, λ, ρ). Assume that $\mathtt{G}$ is used to transmit over a BMS channel. By some abuse of notation, let $H_{\mathtt{G}(n)} = H_{\mathtt{G}}(X \,|\, Y)$ be the associated conditional entropy. Then for any $\xi > 0$

$$\Pr\left\{|H_{\mathtt{G}(n)} - \mathbb{E}\left[H_{\mathtt{G}(n)}\right]| > n\xi\right\} \leq 2\,\mathrm{e}^{-nB\xi^2},$$

where $B = 1/(2(\mathtt{r}_{\max}+1)^2(1-r))$ and where $\mathtt{r}_{\max}$ is the maximal check-node degree.

Proof: The proof uses the standard technique of first constructing a Doob's martingale with bounded differences and then applying the Hoeffding-Azuma inequality. The complete proof can be found in [23] and it is reported in an adapted and streamlined form in Appendix I. ■

Let us now consider the concentration of the MAP EXIT curve. For the BEC this curve is given equivalently by $\frac{1}{n}\sum_{i=1}^{n} H_{\mathtt{G}(n)}(X_i \,|\, Y_{\sim i}(\epsilon))$ or by $\frac{1}{n}H'_{\mathtt{G}(n)}(X \,|\, Y(\epsilon))$. We choose the second representation and phrase the statement in terms of the derivative of the conditional entropy with respect to the channel parameter ϵ.

Theorem 5 (Concentration of MAP EXIT Curve): Let $\mathtt{G}$ be chosen uniformly at random from $\text{LDPC}(n,\lambda,\rho)$ and let $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ denote a family of BMS channels ordered by physical degradation (with $\text{BMS}(\epsilon')$ physically degraded with respect to $\text{BMS}(\epsilon)$ whenever $\epsilon' > \epsilon$) and smooth with respect to ϵ. Assume that $\mathtt{G}$ is used to transmit over the $\text{BMS}(\epsilon)$ channel. Let $H_{\mathtt{G}(n)} = H_{\mathtt{G}}(X \,|\, Y)$ be the associated conditional entropy. Denote by $H'_{\mathtt{G}(n)}$ the derivative of $H_{\mathtt{G}(n)}$ with respect to ϵ (such a derivative exists because of the explicit calculation presented in Theorem 3) and let $J \subseteq I$ be an interval on which $\lim_{n\to\infty}\frac{1}{n}\mathbb{E}\left[H_{\mathtt{G}(n)}\right]$ exists and is differentiable with respect to ϵ. Then, for any $\epsilon \in J$ and $\xi > 0$ there exist an $\alpha_\xi > 0$ such that, for n large enough

$$\Pr\left\{|H'_{\mathtt{G}(n)} - \mathbb{E}[H'_{\mathtt{G}(n)}]| > n\xi\right\} \leq \mathrm{e}^{-n\alpha_\xi}.$$

Furthermore, if $\lim_{n\to\infty}\frac{1}{n}\mathbb{E}\left[H_{\mathtt{G}(n)}\right]$ is twice differentiable with respect to $\epsilon \in J$, there exists a strictly positive constant A such that $\alpha_\xi > A\xi^4$.

The proof is deferred once more to Appendix I.

Notice the two extra hypothesis with respect to Theorem 4. First, we assumed that the channel family $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ is ordered by physical degradation. This ensures that H'_n is non-negative. This condition is trivially satisfied for the family $\{\text{BEC}(\epsilon)\}_{\epsilon\in[0,1]}$. More generally, we can let ϵ be any function of the erasure probability differentiable and increasing from zero to one. The second condition, namely the existence and differentiability of the expected entropy per bit in the limit, is instead crucial. As discussed in the previous section (see, e.g., Fig. 1), the asymptotic EXIT curve may have jumps. By Theorem 2 these jumps correspond to discontinuities in the derivative of the conditional entropy. At a jump ϵ_*, the value of the EXIT curve may vary dramatically when passing from one element of the ensemble to the other. Some (a finite fraction) of the codes will perform well, and have an EXIT curve close to the asymptotic value at $\epsilon_* - \delta$, while others (a finite fraction) may have an EXIT function close to the asymptotic value at $\epsilon_* + \delta$ (δ is here a generic small positive number).

Theorem 6 (Concentration of BP EXIT Curve): Let $\mathtt{G}$ be chosen uniformly at random from $\text{LDPC}(n,\lambda,\rho)$. Assume that $\mathtt{G}$ is used to transmit over a BMS channel and let $\Phi_i^{\text{BP},t} = \phi_i^{\text{BP},t}(Y_{\sim i})$ denote the extrinsic estimate (conditional mean) of X_i produced by the BP decoder after t iterations. Denote by $H^{\text{BP},t}_{\mathtt{G},i} = H_{\mathtt{G}}\big(X_i \,|\, \Phi_i^{\text{BP},t}\big)$ the resulting (extrinsic) entropy of the binary variable X_i. Then, for all $\xi > 0$, there exists $\alpha_\xi > 0$, such that

$$\Pr\Big\{\Big|\sum_{i=1}^{n}\Big(H^{\text{BP},t}_{\mathtt{G},i} - \mathbb{E}_{\mathtt{G}}\Big[H^{\text{BP},t}_{\mathtt{G},i}\Big]\Big)\Big| > n\xi\Big\} \leq \mathrm{e}^{-\alpha_\xi n}. \quad (4)$$

Proof: The proof is virtually identical to the ones given in [15], [17] where the probability of decoding error is considered. ■

III. Asymptotic Setting

A. (MAP) EXIT

The next definition and theorem define our main object of study.

Definition 2: Let $\mathcal{C}(n)$ be a sequence of code ensembles of diverging blocklength n and let $\mathtt{G}(n)$ be chosen uniformly at random from $\mathcal{C}(n)$. Assume that $\lim_{n\to\infty}\mathbb{E}_{\mathtt{G}}\left[\frac{1}{n}\sum_{i=1}^{n} H'_{\mathtt{G}(n)}(X \,|\, Y(\epsilon))\right]$ exists. Then this limit is called the asymptotic EXIT function of the family of ensembles and we denote it by $h^{\text{MAP}}(\epsilon)$. We define the MAP threshold ϵ^{MAP} to be the supremum of all values ϵ such that $h^{\text{MAP}}(\epsilon) = 0$.

Given a dd pair (λ,ρ), consider the sequence of ensembles $\{\text{LDPC}(\lambda,\rho,n)\}_n$. It is natural to conjecture that the associated asymptotic EXIT function exists. Note that from Theorem 5 we know that if this limit exists, then individual code instances are closely concentrated around the ensemble average. It is therefore meaningful to define in such a setting the MAP threshold in terms of the ensemble average.

Unfortunately, no *general* proof of the existence of the MAP EXIT curve is known. But we will show how one can in *most cases* compute the asymptotic EXIT function explicitly for a given ensemble, thus proving existence of the limit in such cases. See also [24] for a discussion on asymptotic thresholds.

It is worth pointing out that we defined the MAP threshold to be the channel parameter at which the conditional entropy becomes sublinear. At this point the average conditional bit entropy converges to zero, so that this point is the bit MAP threshold. We note that for some ensembles the block MAP threshold is strictly smaller than the bit MAP threshold.

Theorem 7 (Asymptotic Area Theorem): Consider a dd pair (λ,ρ). Assume that the associated asymptotic EXIT function as defined in Definition 2 exists for all $\epsilon \in [0,1]$. Assume further that the limit $r_{\text{as}} = \lim_{n\to\infty}\mathbb{E}_{\mathtt{G}}\left[\frac{H(X)}{n}\right]$ exists. Then

$$r_{\text{as}} = \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon.$$

Proof: Let $h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)$ denote the EXIT function associated to a particular $\mathtt{G} \in \text{LDPC}(\lambda,\rho,n)$ with rate $r_{\mathtt{G}(n)}$. We have

$$\int_0^1 \mathbb{E}_{\mathtt{G}}\left[h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)\right]\mathrm{d}\epsilon = \mathbb{E}_{\mathtt{G}}\left[\int_0^1 h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)\mathrm{d}\epsilon\right]$$
$$= \mathbb{E}_{\mathtt{G}}\left[\frac{H(X)}{n}\right] \underset{n\to\infty}{\longrightarrow} r_{\text{as}} \quad (5)$$

The first equality is obtained by noticing that the function $h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)$ is non-negative. We are therefore justified by Fubini theorem to switch the order of integration. The second step follows from the Area Theorem (the rate being equal to $\frac{H(X)}{n}$).

On the other hand, the Dominated Convergence Theorem can be applied to the sequence $\left\{\mathbb{E}_{\mathtt{G}}\left[h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)\right]\right\}_n$ since it converges (as assumed in the hypothesis) to $h^{\text{MAP}}(\epsilon)$ and is trivially upper-bounded by 1. We therefore get

$$\lim_{n\to\infty}\int_0^1 \mathbb{E}_{\mathtt{G}}\left[h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)\right]\mathrm{d}\epsilon = \int_0^1 \lim_{n\to\infty}\mathbb{E}_{\mathtt{G}}\left[h^{\text{MAP}}_{\mathtt{G}(n)}(\epsilon)\right]\mathrm{d}\epsilon = \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon.$$

which, combined with (5), concludes the proof. ■

Lemma 7 gives a sufficient condition for the limit r_{as} to exists. Note that under this condition the asymptotic rate r_{as} is equal to to the design rate $r(\lambda,\rho)$. Most dd pairs (λ,ρ) encountered in practice fulfill this condition. This condition is therefore not very restrictive.

B. BP EXIT

Recall that the MAP EXIT curve can be expressed as $H(X_i \,|\, \Phi_i^{\text{MAP}})$ where $\Phi_i^{\text{MAP}} = \phi_i^{\text{MAP}}(Y_{\sim i})$ is the posterior estimate (conditional mean) of X_i given $Y_{\sim i}$. Unfortunately this quantity is not easy to evaluate. In fact, the main aim of this paper is to accomplish this task.

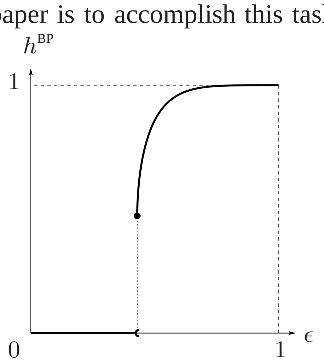


Fig. 4. BP EXIT function $\epsilon \mapsto h^{\text{BP}}(\epsilon)$.

A related quantity which is much easier to compute is the BP EXIT curve shown in Fig. 4 for the dd pair $(\mathtt{x}^2,\mathtt{x}^5)$. The BP EXIT corresponds to $H(X_i \,|\, \Phi_i^{\text{BP}})$, where $\Phi_i^{\text{BP}} = \phi_i^{\text{BP}}(Y_{\sim i})$ is the extrinsic estimate of X_i delivered by the BP decoder Here a fixed number of iterations, let us say t, is understood. Asymptotically, we consider $t \to \infty$ *after* $n \to \infty$. An exact expression for the average asymptotic BP EXIT curve for LDPC ensembles is easily computed via the DE method [15]–[18].

Consider the fixed-point condition for the density evolution equations,

$$\epsilon\lambda(1-\rho(1-\mathtt{x})) = \mathtt{x}.$$

Solving for ϵ, we get $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}$, $\mathtt{x} \in (0,1]$. In words, for each non-zero fixed-point $\mathtt{x}$ of density evolution, there is a unique channel parameter ϵ. At this fixed-point the asymptotic average BP EXIT function equals $\Lambda(1-\rho(1-\mathtt{x}))$. If $\epsilon(\mathtt{x})$ is monotonically increasing in $\mathtt{x}$ over the whole range $[0,1]$, then the BP EXIT curve is given in parametric form by

$$\big(\epsilon(\mathtt{x}), \Lambda(1-\rho(1-\mathtt{x}))\big). \tag{6}$$

For some ensembles (e.g., regular cycle-code ensembles) $\epsilon(\mathtt{x})$ is indeed monotone increasing over the whole range $[0,1]$, but for most ensembles this is not true. In this case we have to restrict the above parameterization to the unique union of intervals

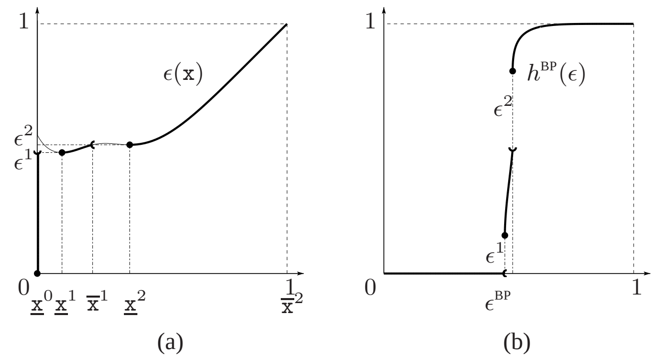


Fig. 5. BP EXIT curve with two discontinuities (J=2): (a) Channel entropy function $\mathtt{x} \mapsto \epsilon(\mathtt{x})$ (b) BP EXIT function $\epsilon \mapsto h^{\text{BP}}(\epsilon)$. This example corresponds to the dd pair $(\lambda,\rho) = (0.3x + 0.3x^2 + 0.4x^{13}, x^6)$, which has design rate $r \approx 0.48718$. The BP threshold is $\epsilon^{\text{BP}} \approx 0.48437$ at $\mathtt{x}^{\text{BP}} \approx 0.09904$. This is also the first discontinuity, i.e., $\epsilon^1 = \epsilon^{\text{BP}}$, $\underline{\mathtt{x}}^1 = \mathtt{x}^{\text{BP}}$ and $\overline{\mathtt{x}}^1 \approx 0.22156$. The second discontinuity occurs for $\epsilon = \epsilon^2 \approx 0.51553$ at $\underline{\mathtt{x}}^2 \approx 0.37016$ ($\overline{\mathtt{x}}^1 = 1$).

$$\mathcal{I} \triangleq \bigcup_{i\in[J]} [\underline{\mathtt{x}}^i, \overline{\mathtt{x}}^i) \cup \{1\},$$

which has the property that $\epsilon(\mathtt{x})$ is continuously and monotonically increasing from ϵ^{BP} to one as $\mathtt{x}$ takes on increasing values in $\mathcal{I}$ and for all $i \in [J]$, $\underline{\mathtt{x}}^i = 0$ or $\epsilon'(\underline{\mathtt{x}}^i) = 0$. An example of such a partition is shown in Fig. 5. That such a partition exists and is unique follows from the fact that $\epsilon(\mathtt{x})$ is a differentiable function for $\mathtt{x} \in [0,1]$ as can be verified by direct computation. Set $\overline{\mathtt{x}}^J = 1$ and note that $\epsilon(1) = 1 \geq 0$. Define $\underline{\mathtt{x}}^J$ as the largest nonnegative value of $\mathtt{x} \leq \overline{\mathtt{x}}^J$ for which $\epsilon'(\mathtt{x}) = 0$. If no such value exists then $\epsilon(\mathtt{x})$ is monotonically increasing over the whole range $[0,1]$. In this case $J = 1$ and we set $\underline{\mathtt{x}}^J = 0$. Now proceed recursively. Assume that the intervals $[\underline{\mathtt{x}}^{i+1}, \overline{\mathtt{x}}^{i+1})$ have been defined and that $\underline{\mathtt{x}}^{i+1} > 0$. Define $\overline{\mathtt{x}}^i$ as the largest nonnegative value of $\mathtt{x} < \underline{\mathtt{x}}^{i+1}$ such that $\epsilon(\mathtt{x}) = \epsilon(\underline{\mathtt{x}}^{i+1})$. Note that if such a value exists then we must have $\epsilon'(\mathtt{x}) \geq 0$. If no such value exists then we have already found the sought after partition and we stop. Otherwise define $\underline{\mathtt{x}}^i$ as the largest nonnegative value of $\mathtt{x} \leq \overline{\mathtt{x}}^i$ for which $\epsilon'(\mathtt{x}) = 0$. As before, if no such value exists then set $\underline{\mathtt{x}}^i = 0$ and stop. Without loss we can eliminate from the resulting partition any interval of zero length. Let J denote the number of remaining intervals of nonzero length. Note, if the BP threshold happens at a discontinuous phase transition (jump), then $\mathtt{x}^{\text{BP}} = \underline{\mathtt{x}}^1$ and $\epsilon^{\text{BP}} = \epsilon(\underline{\mathtt{x}}^1)$, otherwise, if the BP threshold is given by the stability condition, then $\mathtt{x}^{\text{BP}} = \underline{\mathtt{x}}^0 = 0$ and $\epsilon^{\text{BP}} = \epsilon(\underline{\mathtt{x}}^0)$. See also Fig. 8.

Corollary 1: Assume we are given a dd pair (λ,ρ) and that transmission takes place over the BEC. Let $\mathcal{I} \triangleq \bigcup_{i\in[J]}[\underline{\mathtt{x}}^i, \overline{\mathtt{x}}^i) \cup \{1\}$ be the partition associated to (λ,ρ). Define $\epsilon^{\text{BP}} = \epsilon(\underline{\mathtt{x}}^1)$. Then the BP EXIT function $h^{\text{BP}}(\epsilon)$ is equal to zero for $0 \leq \epsilon < \epsilon^{\text{BP}}$ and for $\epsilon > \epsilon^{\text{BP}}$ it has the parametric characterization

$$(\epsilon(\mathtt{x}), \Lambda(1-\rho(1-\mathtt{x}))),$$

where $\mathtt{x}$ takes on all values in $\mathcal{I}$.

Fact 1 (Regular LDPC Ensembles "Jump" at Most Once): Consider the regular dd pair $(\lambda(x), \rho(x)) = (x^{\mathtt{l}-1}, x^{\mathtt{r}-1})$. Then the function $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}$ has a unique minimum in the range $[0,1]$. Let $\mathtt{x}^{\text{BP}}$ denote the location of this minimum. Then $\epsilon(\mathtt{x})$ is strictly decreasing on $(0, \mathtt{x}^{\text{BP}})$ and strictly increasing on $(\mathtt{x}^{\text{BP}}, 1)$. Moreover, $\mathtt{x}^{\text{BP}} = 0$ if and only if $\mathtt{l} = 2$.

Proof: Note that $\epsilon(1) = 1$ and by direct calculation we see that $\epsilon'(1) = 1$. Therefore, either $\epsilon(\mathtt{x})$ takes on its minimum value within the interval $[0,1]$ for $\mathtt{x} = 0$ or its minimum value is in the interior of the region $[0,1]$. Computing explicitly the derivative of $\epsilon(\mathtt{x})$, we see that the location of the minima of $\epsilon(\mathtt{x})$ must be a root of $W(\mathtt{x}) \triangleq 1 - (1-\mathtt{x})^{\mathtt{r}-1} - (\mathtt{l}-1)(\mathtt{r}-1)(1-\mathtt{x})^{\mathtt{r}-2}\mathtt{x}$. Furthermore $W'(\mathtt{x}) = -(\mathtt{r}-1)(1-\mathtt{x})^{\mathtt{r}-3}\{(\mathtt{l}-2) - [(\mathtt{l}-1)(\mathtt{r}-1)-1]\mathtt{x}\}$. Notice that $W(0) = 0$, $W'(0) = -(\mathtt{r}-1)(\mathtt{l}-2) < 0$ and $W(1) = 1$. By the Intermediate Value Theorem, $W(\mathtt{x})$ vanishes at least once in $(0,1)$. Suppose now that $W(\mathtt{x})$ vanishes more than once in $(0,1)$, and consider the first two such zeros $\mathtt{x}_1$, $\mathtt{x}_2$. It follows that $W'(\mathtt{x})$ must vanish at least twice: once in $(0, \mathtt{x}_1)$ and once in $(\mathtt{x}_1, \mathtt{x}_2)$. On the other end, the above explicit expression implies that $W'(\mathtt{x})$ vanishes just once in $(0,1)$, at $\mathtt{x} = (\mathtt{l}-2)/[(\mathtt{l}-1)(\mathtt{r}-1)-1]$. Therefore $W(\mathtt{x})$ has exactly one root in $(0,1)$. See also [25]. ∎

A dynamic interpretation of the convergence of the BP decoding when the number of iterations $t \to \infty$ is shown in Appendix IV using component EXIT curves. It is further shown in Appendix III and Theorem 11 how to compute the area under the BP EXIT curve. The calculations show that this area is always larger or equal the design rate. Moreover, some calculus reveals that, whenever the BP EXIT function has discontinuities, then the area is strictly larger than the design rate r.

C. Extended BP EXIT Curve

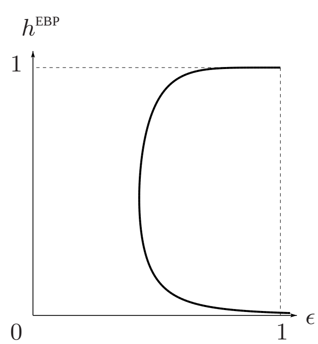


Fig. 6. EBP EXIT function $\{(\epsilon(\mathtt{x}), \Lambda(\mathtt{y}(\mathtt{x})))\}_{\mathtt{x}}$.

Surprisingly, we can apply the Generalized Area Theorem also to BP decoding if we consider the *Extended* BP EXIT (EBP) curve. Fig. 6 shows this EBP EXIT curve for the running example, i.e., for the dd pair $(\mathtt{x}^2, \mathtt{y}^5)$. We will see shortly that this EBP EXIT curve plays a central role in our investigation. First, let us give its formal definition.

Definition 3: Assume we are given a dd pair (λ, ρ). The EBP EXIT curve, denote it by h^{EBP}, is given in parametric form by

$$(\epsilon, h^{\text{EBP}}) = (\epsilon(\mathtt{x}), \Lambda(1-\rho(1-\mathtt{x}))),$$

where $\epsilon(\mathtt{x}) = \frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}$ and $\mathtt{x} \in [0,1]$.

Theorem 8 (Area Theorem for EBP Decoding): Assume we are given a dd pair (λ, ρ) of design rate r. Then the EBP EXIT curve satisfies

$$\int_0^1 h^{\text{EBP}}(\mathtt{x}) \mathrm{d}\epsilon(\mathtt{x}) = r.$$

Proof: We will give two proofs of this fact.

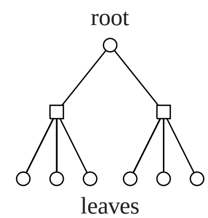


Fig. 7. Graph of a small tree code: computation tree of depth one for the regular (2,4) LDPC ensemble.

(i) The first proof applies only if $\epsilon(\mathtt{x}) \leq 1$ for $\mathtt{x} \in (0,1]$. This in turn happens only if $\lambda'(0) > 0$, i.e., if the ensemble has a non-trivial stability condition. We use the (General) Area Theorem for transmission over binary erasure channels where we allow the parameter of the channel to vary as a function of the bit position. First, let us assume that the ensemble is $(\mathtt{l}, \mathtt{r})$-regular. Consider a variable node and the corresponding computation tree of depth one as shown in Fig. 7. Let us further define two channel families. The first is the family $\{\text{BEC}(\mathtt{x})\}_{\mathtt{x}=0}^1$. The second one is the family[5] $\{\text{BEC}(\epsilon(\mathtt{x}))\}_{\mathtt{x}=0}^1$ where $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}$). The two families are parametrized by a common parameter $\mathtt{x}$ which is the fixed-point of density evolution: they are smooth since $\epsilon(\mathtt{x})$ is differentiable with respect to $\mathtt{x}$. Let us now assume that the bit associated to the root node is passed through a channel BEC($\epsilon(\mathtt{x})$), while the ones associated to the leaf nodes are passed through a channel BEC($\mathtt{x}$). We can apply the General Area Theorem: let $X = (X_1, \dots, X_{1+\mathtt{l}\times(\mathtt{r}-1)})$ be the transmitted codeword chosen uniformly at random from the tree code and $Y(\mathtt{x})$ be the result of passing X through the respective erasure channels parameterized by the common parameter $\mathtt{x}$. The General Area Theorem states that $H(X \,|\, Y(\mathtt{x}=1)) - H(X \,|\, Y(\mathtt{x}=0)) = H(X)$ is equal to the sum of the integrals of the individual EXIT curves, where the integral extends from $\mathtt{x} = 0$ to $\mathtt{x} = 1$. There are two types of individual EXIT curves, namely the one associated to the root node, call it $h_{\text{root}}(\mathtt{x})$ and the $\mathtt{l}(\mathtt{r}-1)$ ones associated to the leaf nodes, call them $h_{\text{leaf}}(\mathtt{x})$. To summarize, the General Area Theorem states

$$H(X) = \int_0^1 h_{\text{root}}(\mathtt{x})\, \mathrm{d}\epsilon(\mathtt{x}) + \mathtt{l}(\mathtt{r}-1) \int_0^1 h_{\text{leaf}}(\mathtt{x}) \mathrm{d}\mathtt{x}.$$

Note that $H(X) = 1 + \mathtt{l}(\mathtt{r}-1) - \mathtt{l} = 1 - \mathtt{l}(\mathtt{r}-2)$ since the computation tree contains $1 + \mathtt{l}(\mathtt{r}-1)$ variable nodes and $\mathtt{l}$ check nodes. Moreover, $\int_0^1 h_{\text{leaf}}(\mathtt{x}) \mathrm{d}\mathtt{x} = \int_0^1 1 - \rho(1-\mathtt{x}) \mathrm{d}\mathtt{x} =$

[5] Recall that $0 \leq \epsilon(\mathtt{x}) \leq 1$ for all $\mathtt{x} \in [0,1]$ by assumption.

$(\mathtt{r}-1)/\mathtt{r}$ since the message flowing from the root node to the check nodes is erased with probability $\mathtt{x}$ (Recall that $\mathtt{x} = \epsilon(\mathtt{x})\lambda(1-\rho(1-\mathtt{x}))$, where $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^{1-1}, \mathtt{x}^{\mathtt{r}-1})$. Moreover, observe that the result could also be obtained by applying the Area Theorem locally to the Single-Parity-Check code). Collecting these observations and solving for $\int_0^1 h_{\text{root}}(\mathtt{x}) d\epsilon(\mathtt{x})$, we get

$$\int_0^1 h_{\text{root}}(\mathtt{x}) d\epsilon(\mathtt{x}) = 1 - \mathtt{l}/\mathtt{r} = r,$$

as claimed since $h_{\text{root}} = h^{\text{EBP}}$. The irregular case follows in the same manner: we consider the ensemble of computation trees of depth one where the degree of the root note is chosen according to the node degree distribution $\Lambda(\mathtt{x})$ and each edge emanating from this root node is connected to a check node whose degree is chosen according to the edge degree distribution $\rho(x)$. As before, leaf nodes experience the channel BEC($\mathtt{x}$), whereas the root node experiences the channel BEC($\epsilon(\mathtt{x})$). We apply the General Area Theorem to each such choice and average with the respective probabilities.

(ii) The second proof applies in all cases. Applying integration by parts twice we can write

$$\begin{aligned}
\int_0^1 h^{\text{EBP}}(\mathtt{x}) \mathrm{d}\epsilon(\mathtt{x}) &= h^{\text{EBP}}(\mathtt{x})\epsilon(\mathtt{x})\Big|_{\mathtt{x}=0}^{1} - \int_0^1 \frac{\mathrm{d}h^{\text{EBP}}(\mathtt{x})}{\mathrm{d}\mathtt{x}} \epsilon(\mathtt{x}) \mathrm{d}\mathtt{x} \\
&\stackrel{(a)}{=} 1 - \Lambda'(1) \int_0^1 x\rho'(1-x) \mathrm{d}\mathtt{x} \\
&= 1 - \frac{\left(x\rho(1-x)\big|_{\mathtt{x}=0}^{1} + \int_0^1 \rho(1-x)\mathrm{d}\mathtt{x}\right)}{\int_0^1 \lambda(\mathtt{x})\mathrm{d}\mathtt{x}} \\
&= 1 - \Lambda'(1)/\Gamma'(1) = r,
\end{aligned}$$

where (a) follows since $h^{\text{EBP}}(\mathtt{x}) = \Lambda'(1) \int_0^{1-\rho(1-\mathtt{x})} \lambda(\mathtt{x}) \mathrm{d}\mathtt{x}$ and $\Lambda'(1) = 1/\int_0^1 \lambda$. Similar computations will be performed several times throughout this paper. In this respect it is handy to be able to refer to two basic facts related to this integration which are summarized as Lemma 14 and Lemma 15 in Appendix III-A. ■

IV. An Upper-Bound for the Maximum A Posteriori Threshold

Assume that transmission takes places over BEC(ϵ). Given a dd pair (λ, ρ), we trivially have the relations

$$\epsilon^{\text{BP}} \le \epsilon^{\text{MAP}} \le \min\{\epsilon^{\text{Sh}}, \epsilon^{\text{Stab}}\}, \tag{7}$$

where ϵ^{Sh} and ϵ^{Stab} denote, respectively, the Shannon and stability threshold. As we have discussed, it is straightforward to compute ϵ^{BP} by means of DE and $\epsilon^{\text{BP}} \le \epsilon^{\text{MAP}}$ follows from the sub-optimality of BP decoding. The inequality $\epsilon^{\text{MAP}} \le \epsilon^{\text{Sh}} = 1-r$ is a rephrasing of the Channel Coding Theorem. Finally $\epsilon^{\text{MAP}} \le \epsilon^{\text{Stab}} = 1/(\lambda'(0)\rho'(1))$ can be proved through the following graph-theoretic argument. Assume, by contradiction that $\epsilon^{\text{MAP}} > \epsilon^{\text{Stab}}$ and let ϵ be such that $\epsilon^{\text{Stab}} < \epsilon < \epsilon^{\text{MAP}}$. Notice that $\epsilon^{\text{Stab}} < \epsilon$ is equivalent to $\epsilon\lambda'(0)\rho'(1) > 1$. Consider now the residual Tanner graph once the received variable nodes have been pruned, and focus on the subgraph of degree 2 variable nodes. Such a Tanner graph can be identified with an ordinary graph by mapping the check nodes to vertices and the variable nodes to edges. The average degree of such a graph is $\epsilon\lambda'(0)\rho'(1) > 1$ and therefore a finite fraction of its vertices belong to loops [26]. If a bit belongs to such a loop, it is not determined by the received message: in particular $\mathbb{E}[X_i|Y] = 1/2$. In fact, there exist a codeword such that $x_i = 1$: just set $x_j = 1$ if j belongs to some fixed loop through i and 0 otherwise. Since there is a finite fraction of such vertices $h(\epsilon) > 0$ (if the limit exist) and therefore $\epsilon > \epsilon^{\text{MAP}}$. We reached a contradiction, therefore $\epsilon^{\text{MAP}} \le \epsilon^{\text{Stab}}$ as claimed.

While ϵ^{Stab} and ϵ^{Sh} are simple quantities, the threshold ϵ^{MAP} is not as easy to compute. In this section we will prove an *upper-bound* on ϵ^{MAP} in terms of the (extended) BP EXIT curve. In the next sections, we will see that in fact this bound is tight for a large class of ensembles. The key to this bound is to associate the Area Theorem with the following intuitive inequality.

Lemma 1: Consider a dd pair (λ, ρ) and the associated EXIT functions h^{BP} and h^{MAP}. Then $h^{\text{MAP}} \le h^{\text{BP}}$.

Proof: Note that Lemma 1 expresses the natural statement that BP processing is in general suboptimal. For a given length n, pick a code at random from LDPC(λ, ρ, n). Call Φ_i^{BP} the extrinsic BP estimate of bit i and note that $\Phi_i^{\text{BP}} = \Phi_i^{\text{BP}}(Y_{\sim i})$, i.e., the extrinsic BP estimate is a well defined function of $Y_{\sim i}$. The Data Processing Theorem asserts that $H(X_i|Y_{\sim i}) \le H(X_i|\Phi_i^{\text{BP}}(Y_{\sim i}))$. This is true for all codes in LDPC(λ, ρ, n). Therefore taking first the average over the ensemble and second the limit when the blocklength $n \to \infty$ (assuming the limit of the MAP EXIT function exists), we get $h^{\text{MAP}}(\epsilon) \le h^{\text{BP}}(\epsilon)$. ■

Because of Lemma 1, it is of course not surprising that the integral under h^{BP} is larger or equal than the asymptotic rate of the code r_{as} as pointed out in Section III-B. In most of the cases encountered in practice, $r = r_{\text{as}}$, (see Section V), the area under the MAP EXIT curve is therefore r and the area under the BP EXIT curve is strictly larger than r if and only if the curve exhibits discontinuities (in the absence of discontinuities, the two curves coincide and the MAP/BP threshold is given by the stability condition).

Example 1 refines and illustrates this observation by showing that the BP and MAP threshold might be equal even if their respective EXIT functions are not pointwise equal.

Example 1: Consider the dd pair $(\lambda, \rho) = (0.4x + 0.6x^6, x^6)$ and the corresponding LDPC ensemble with design rate $r = 0.5$. Using a weight enumerator function, see, e.g., Section V, one can show that $r = r_{\text{as}} = \int h^{\text{MAP}}$. A quick look shows that the BP threshold is given by the stability condition, i.e., it is $\epsilon^{\text{BP}} \approx 0.4167$ obtained for $\mathtt{x} \approx \underline{\mathtt{x}}^0 = 0$. When the parameter is $\overline{\mathtt{x}}^0 \approx 0.04828$, i.e., at $\epsilon^1 \approx 0.4691$, a discontinuity of the BP EXIT curve appears and the edge erasure probability $\mathtt{x}$ "jumps" to $\underline{\mathtt{x}}^1 \approx 0.3309$. This situation is shown in Fig. 8. Since the BP threshold is determined by the stability condition, as explained previously we have $\epsilon^{\text{BP}} = \epsilon^{\text{MAP}}$. This is true despite the fact that the integral under the BP EXIT is larger than $r = r_{\text{as}}$!

Recall that the Area Theorem asserts that $\int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon = r_{\text{as}}$, where r_{as} is the asymptotic rate of the ensemble defined

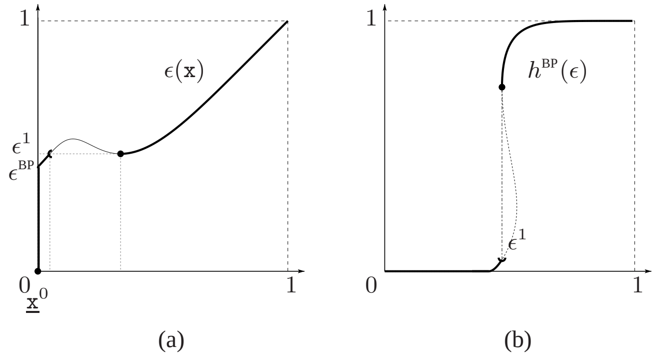


Fig. 8. BP EXIT entropy curve with 1 discontinuity (J=1) for which the BP threshold $\epsilon^{\text{BP}} = \epsilon^{\text{MAP}}$ is given by the stability condition: (a) Channel entropy function $\mathtt{x} \mapsto \epsilon(\mathtt{x})$ (b) BP EXIT function $\epsilon \mapsto h^{\text{BP}}(\epsilon)$.

in Theorem 7. By definition $h^{\text{MAP}}(\epsilon) = 0$ for $\epsilon \leq \epsilon^{\text{MAP}}$. Therefore we have in fact $\int_{\epsilon^{\text{MAP}}}^1 h^{\text{MAP}}(\epsilon)\text{d}\epsilon = r_{\text{as}}$. Now note that the BP decoder is in general suboptimal so that $h^{\text{MAP}}(\epsilon) \leq h^{\text{BP}}(\epsilon)$. Further, in general $r_{\text{as}} \geq r(\lambda,\rho)$. Combining these statements we see that if $\overline{\epsilon}^{\text{MAP}}$ is a real number in $[\epsilon^{\text{BP}}, 1]$ such that $\int_{\overline{\epsilon}^{\text{MAP}}}^1 h^{\text{BP}}(\epsilon)\text{d}\epsilon = r(\lambda,\rho)$ then $\int_{\overline{\epsilon}^{\text{MAP}}}^1 h^{\text{MAP}}(\epsilon)\text{d}\epsilon \leq r_{\text{as}}$. We conclude that for such a $\overline{\epsilon}^{\text{MAP}}$, $\epsilon^{\text{MAP}} \leq \overline{\epsilon}^{\text{MAP}}$. Let us summarize a slightly strengthened version of this observation as a lemma.

Lemma 2 (First Upper Bound on ϵ^{MAP}): Assume we are given a dd pair (λ,ρ). Let $h^{\text{BP}}(\epsilon)$ denote the associated BP EXIT function and let $\overline{\epsilon}^{\text{MAP}}$ be the unique real number in $[\epsilon^{\text{BP}}, 1]$ such that $\int_{\overline{\epsilon}^{\text{MAP}}}^1 h^{\text{BP}}(\epsilon)\text{d}\epsilon = r(\lambda,\rho)$. Then $\epsilon^{\text{MAP}} \leq \overline{\epsilon}^{\text{MAP}}$. If in addition $\overline{\epsilon}^{\text{MAP}} = \epsilon^{\text{BP}}$ then $\epsilon^{\text{MAP}} = \epsilon^{\text{BP}}$, and in fact $h^{\text{MAP}}(\epsilon) = h^{\text{BP}}(\epsilon)$ for all $\epsilon \in [0,1]$.

Proof: We have already discussed the first part of the lemma. To see the second part, if $\overline{\epsilon}^{\text{MAP}} = \epsilon^{\text{BP}}$ then by (7) we have a lower and an upper bound that match and therefore we have equality. This can only happen if the two EXIT functions are in fact identical (and if $r_{\text{as}} = r(\lambda,\rho)$). ■

Example 2: For the dd pair $(\lambda(x),\rho(x)) = (x, x^3)$, we obtain $\overline{\epsilon}^{\text{MAP}} = 1/3 = \epsilon^{\text{BP}}$. Therefore, for this case the MAP EXIT function is equal to the BP EXIT function and in particular both decoders have equal thresholds.

Example 3: For the dd pair $(\lambda(x),\rho(x)) = (x^2, x^3)$, we obtain $\overline{\epsilon}^{\text{MAP}} = \frac{102-7\sqrt{21}}{108} \approx 0.647426$. Note that this dd pair has rate $1/4$ so that this upper bound on the threshold should be compared to the Shannon limit $3/4 = 0.75$.

Example 4: For the dd pair $(\lambda(x),\rho(x)) = (x^2, x^5)$ of our running example, we get

$$\overline{\epsilon}^{\text{MAP}} = \frac{7-\sqrt{-1-a+b}-\sqrt{-2+a-b+\frac{4}{\sqrt{-1-a+b}}}}{6\left(-1+\left(-\frac{1}{6}+\frac{\sqrt{-1-a+b}}{6}+\frac{\sqrt{-2+a-b+\frac{4}{\sqrt{-1-a+b}}}}{6}\right)^5\right)^2},$$

with $a \triangleq \frac{7\cdot 5^{\frac{2}{3}}}{\left(11+6\sqrt{51}\right)^{\frac{1}{3}}}$ and $b \triangleq \left(55+30\sqrt{51}\right)^{\frac{1}{3}}$. Numerically, $\overline{\epsilon}^{\text{MAP}} \approx 0.4881508841915644$. The Shannon threshold for this ensemble is 0.5.

For a dd pair which exhibits a single jump the computation of this upper bound is made somewhat easier by the following lemma. Note that by Fact 1 this lemma is applicable to regular ensembles.

Lemma 3: Assume we are given a dd pair (λ,ρ). Define the polynomial $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$ and, for $\mathtt{x} \in (0,1]$ the function $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$. Assume that $\epsilon(\mathtt{x})$ is increasing over $[\mathtt{x}^{\text{BP}}, 1]$. Let $\mathtt{x}^*$ be the unique root of the polynomial

$$P(\mathtt{x}) \triangleq \Lambda'(1)\mathtt{x}(1-\mathtt{y}(\mathtt{x})) - \frac{\Lambda'(1)}{\Gamma'(1)}[1-\Gamma(1-\mathtt{x})] + \epsilon(\mathtt{x})\Lambda(\mathtt{y}(\mathtt{x})),$$

in the interval $[\mathtt{x}^{\text{BP}}, 1]$. Then $\overline{\epsilon}^{\text{MAP}} = \epsilon(\mathtt{x}^*)$.

Proof: Recall that if $\epsilon(\mathtt{x})$ is increasing over $[\mathtt{x}^{\text{BP}}, 1]$ then we have the parametric representation of $h^{\text{BP}}(\epsilon)$ as given in (6). Using Lemmas 14 and 15 we can express the integral $\int_{\overline{\epsilon}^{\text{MAP}}}^1 h^{\text{BP}}(\epsilon)\text{d}\epsilon$ as a function of $\overline{\epsilon}^{\text{MAP}}$. More precisely, we parametrize $\overline{\epsilon}^{\text{MAP}}$ by $\mathtt{x}$ and express the integral as a function of $\mathtt{x}$. Equating the result to $r(\lambda,\rho) = 1-\Lambda'(1)/\Gamma'(1)$ and solving for $\mathtt{x}$ leads to the polynomial condition $P(\mathtt{x}) = 0$ stated above. ■

Example 5: The following table compares the thresholds and bounds for various ensembles. Hereby $\lambda^{(1)}(x) = x$, $\lambda^{(2)}(x) = \frac{7x^2+2x^3+1x^4}{10}$, $\lambda^{(3)}(x) = \frac{2857x+3061.47x^2+4081.53x^9}{10000}$, $\lambda^{(4)}(x) = \frac{7.71429x^2+2.28571x^7}{10}$, and $\lambda^{(5)}(x) = \frac{9x^2+1x^7}{10}$. The threshold of the first ensemble is given by the stability condition. Its exact value is $7/28 \approx 0.1786$.

$\lambda(x)$	$\rho(x)$	ϵ^{BP}	$\overline{\epsilon}^{\text{MAP}}$	ϵ^{MAP}	ϵ^{Sh}
$\lambda^{(1)}(x)$	$\frac{2x^5+3x^6}{5}$	0.1786	0.1786	0.1786	0.3048
$\lambda^{(2)}(x)$	$\frac{2x^5+3x^6}{5}$	0.4236	0.4948	0.4948	0.5024
$\lambda^{(3)}(x)$	x^6	0.4804	0.4935	0.4935	0.5000
$\lambda^{(4)}(x)$	x^4	0.5955	0.6979	0.6979	0.7000
$\lambda^{(5)}(x)$	x^7	0.3440	0.3899	0.3899	0.4000

The polynomial $P(\mathtt{x})$ provides in fact a fundamental characterization of the MAP threshold and has some important properties. These are more conveniently stated in terms of a slightly more general concept.

Definition 4: The *trial entropy* for the channel BEC(ϵ) associated to the dd pair (λ,ρ) is the bi-variate polynomial

$$P_\epsilon(\mathtt{x},\mathtt{y}) \triangleq \Lambda'(1)\mathtt{x}(1-\mathtt{y}) - \frac{\Lambda'(1)}{\Gamma'(1)}[1-\Gamma(1-\mathtt{x})] + \epsilon\Lambda(\mathtt{y}).$$

A few properties of the trial entropy are listed in the following.

Lemma 4: Let (λ,ρ) be a dd pair and $P_\epsilon(\mathtt{x},\mathtt{y})$ the corresponding trial entropy. Consider furthermore the DE equations for the ensemble $\mathtt{x}_{t+1} = \epsilon\lambda(\mathtt{y}_t)$, $\mathtt{y}_{t+1} = 1-\rho(1-\mathtt{x}_t)$, t being the iteration number. Then (in what follows we always consider $\mathtt{x},\mathtt{y} \in [0,1]$)

1) The fixed points of density evolution are stationary points of the trial entropy. Vice versa, any stationary point of the trial entropy is a fixed point of density evolution.
2) $P(\mathtt{x}) = P_{\epsilon(\mathtt{x})}(\mathtt{x},\mathtt{y}(\mathtt{x}))$.
3) $P(\mathtt{x}=1) = P_{\epsilon=1}(\mathtt{x}=1,\mathtt{y}=1) = r(\lambda,\rho)$.
4) Let $a \triangleq (\epsilon_a = \epsilon(\mathtt{x}_a), h^{\text{EBP}}(\mathtt{x}_a))$ and $b \equiv (\epsilon_b = \epsilon(\mathtt{x}_b), h^{\text{EBP}}(\mathtt{x}_b))$ be two points on the EBP EXIT curve (with $\mathtt{x}_{a/b} \in (0,1]$) and define $\mathtt{y}_{a/b} = 1-\rho(1-\mathtt{x}_{a/b})$. Then

$$\int_a^b h^{\text{EBP}}(\epsilon(\mathtt{x}))\, d\epsilon(\mathtt{x}) = P_{\epsilon_b}(\mathtt{x}_b,\mathtt{y}_b) - P_{\epsilon_a}(\mathtt{x}_a,\mathtt{y}_a).$$

Proof: (1) is proved by explicitly computing the partial derivatives of $P_\epsilon(\mathtt{x},\mathtt{y})$ with respect to $\mathtt{x}$ and $\mathtt{y}$: $\partial_{\mathtt{x}} P_\epsilon(\mathtt{x},\mathtt{y}) =$

$\Lambda'(1)[1-\mathtt{y}-\rho(1-\mathtt{x})]$, $\partial_{\mathtt{y}}P_\epsilon(\mathtt{x},\mathtt{y}) = \Lambda'(1)[-\mathtt{x}+\epsilon\lambda(\mathtt{y})]$. Since $\Lambda'(1)>0$, the stationarity conditions $\partial_{\mathtt{x}}P_\epsilon(\mathtt{x},\mathtt{y})=0$ and $\partial_{\mathtt{y}}P_\epsilon(\mathtt{x},\mathtt{y})=0$ are equivalent to the fixed point conditions for DE. (2) and (3) are elementary algebra. In order to prove (4), notice that we have $\partial_{\mathtt{x}}P_\epsilon(\mathtt{x},\mathtt{y}) = \partial_{\mathtt{y}}P_\epsilon(\mathtt{x},\mathtt{y}) = 0$ at any point $(\mathtt{x},\mathtt{y}(\mathtt{x}),\epsilon(\mathtt{x}))$ along the EBP EXIT curve. This follows from the fact that points on the EBP EXIT curve are fixed points of density evolution. Therefore

$$\frac{\mathrm{d}}{\mathrm{d}\mathtt{x}}P_{\epsilon(\mathtt{x})}(\mathtt{x},\mathtt{y}(\mathtt{x})) = \Lambda(\mathtt{y}(\mathtt{x}))\frac{\mathrm{d}\epsilon}{\mathrm{d}\mathtt{x}}(\mathtt{x}) = h^{\text{EBP}}(\epsilon(\mathtt{x}))\frac{\mathrm{d}\epsilon}{\mathrm{d}\mathtt{x}}(\mathtt{x})\,.$$

The thesis follows by integrating over $\mathtt{x}$. Equivalently, we could have used again Lemmas 14 and 15. ■

Unfortunately, the upper-bound stated in Lemma 2 is not always tight. In particular, this can happen if the EBP EXIT curve exhibits multiple jumps (i.e., if $\epsilon(\mathtt{x})$ has more than one local maximum in the interval $(0,1]$). We will state a precise sufficient condition for tightness in the next section. An improved upper bound is obtained as follows.

Theorem 9 (Improved Upper-Bound on ϵ^{MAP}): Assume we are given a dd pair (λ,ρ). Let $h^{\text{EBP}}(\epsilon)$ denote the associated EBP EXIT function and let $(\overline{\epsilon}^{\text{MAP}} = \epsilon(\mathtt{x}^*), h^{\text{EBP}}(\mathtt{x}^*))$ be a point on this curve. Assume that $\int_{\mathtt{x}^*}^1 h^{\text{EBP}}(\mathtt{x})\mathrm{d}\epsilon(\mathtt{x}) = r(\lambda,\rho)$ and that there exist no $\mathtt{x}'\in(\mathtt{x}^*,1]$ such that $\epsilon(\mathtt{x}')=\epsilon(\mathtt{x}^*)$. Then $\epsilon^{\text{MAP}}\le\overline{\epsilon}^{\text{MAP}}$.

The proof of this theorem will be given in Section VI using the so-called Maxwell construction. Notice that in general there can be more than one value of ϵ satisfying the theorem hypotheses. We shall always use the symbol $\overline{\epsilon}^{\text{MAP}}$ to refer to the smallest such value. On the other hand, it is a consequence of the proof of theorem that there always exists at least one such value.

As before, the following lemma simplifies the computation of the upper bound by stating the following more explicit characterization.

Lemma 5: Consider a dd pair (λ,ρ). Let $\mathtt{x}^*\in(0,1]$ be a root of the polynomial $P(\mathtt{x})$ defined in (3), such that there exist no $\mathtt{x}'\in(\mathtt{x}^*,1]$ with $\epsilon(\mathtt{x}')=\epsilon(\mathtt{x}^*)$. Then $\epsilon^{\text{MAP}}\le\epsilon(\mathtt{x}^*)$, and $\overline{\epsilon}^{\text{MAP}}$ is the smallest among such upper bounds.

Proof: Let $\mathtt{x}^*$ be defined as in the statement. Then, by Lemma 4, points (2), (3) and (4):

$$\int_{\mathtt{x}^*}^1 h^{\text{EBP}}(\mathtt{x})\,\mathrm{d}\epsilon(\mathtt{x}) = P(1)-P(\mathtt{x}^*) = r(\lambda,\rho)-P(\mathtt{x}^*)\,.$$

Therefore, $\int_{\mathtt{x}^*}^1 h^{\text{EBP}}(\mathtt{x})\,\mathrm{d}\epsilon(\mathtt{x}) = r(\lambda,\rho)$ if and only if $P(\mathtt{x}^*)=0$. ■

For a large family of dd pairs the upper bound stated in Theorem 9 is indeed tight. Nevertheless, it is possible to construct examples where we can not evaluate the bound at all roots $\mathtt{x}^*$ of $P(\mathtt{x})$ since for some of those roots there exists a point $\mathtt{x}'\in(\mathtt{x}^*,1]$ with $\epsilon(\mathtt{x}')=\epsilon(\mathtt{x}^*)$. In these cases we expect the bound not to be tight. Indeed, we conjecture that the extra condition on the roots of $P(\mathtt{x})$ are not necessary and that the MAP threshold is in general given by the following statement.

Conjecture 1: Consider a degree distribution pair (λ,ρ) and the associated polynomial $P(\mathtt{x})$ defined as in (3). Let $\mathcal{X}\subset(0,1]$ be the set of positive roots of $P(\mathtt{x})$ in the interval $(0,1]$ (since $P(\mathtt{x})$ is a polynomial, $\mathcal{X}$ is finite). Equivalently, $\mathcal{X}$ is the set of $\mathtt{x}_*\in(0,1]$ such that $\int_{\mathtt{x}^*}^1 h^{\text{EBP}}(\mathtt{x})\,\mathrm{d}\epsilon(\mathtt{x}) = r(\lambda,\rho)$. Then $\epsilon^{\text{MAP}} = \min\{\epsilon(\mathtt{x}^*);\,\mathtt{x}\in\mathcal{X}\}$.

V. Counting Argument

We will now describe a counting argument which yields an alternative proof of Lemma 2. More interestingly, the argument can be strengthened to obtain an easy-to-evaluate sufficient condition for tightness of the upper-bound.

The basic idea is quite simple. Recall that we define the MAP threshold as the maximum of all channel parameters for which the normalized conditional entropy converges to zero as the block length tends to infinity. For the binary erasure channel, the conditional entropy is equal to the logarithm of the number of codewords which are compatible with the received word. Therefore, a first naive way of upper bounding the MAP threshold consists in lower bounding the expected number of codewords in the residual graph, after eliminating the received variables. If, for a given channel parameter, this lower bound is exponential with a strictly positive exponent, then the corresponding conditional entropy is strictly positive and we are operating above the threshold. It turns out that a much better result is obtained by considering the residual graph after iterative decoding has been applied. In fact, this simple modification allows one to obtain matching upper and lower bounds in a large number of cases.

Let $\mathtt{G}$ be chosen uniformly at random from the ensemble characterized by $\Xi\triangleq(\Lambda,\Gamma)$. Assume further that transmission takes place over BEC(ϵ) and that a BP decoder is applied to the received sequence. Denote by $\mathtt{G}(\epsilon)$ the residual graph after decoding has halted, and by $\Xi_{\mathtt{G}(\epsilon)} = (\Lambda_{\mathtt{G}(\epsilon)},\Gamma_{\mathtt{G}(\epsilon)})$ its degree profile (i.e., the fraction of nodes of any given degree). We adopt here the convention of normalizing the dd pair of $\mathtt{G}(\epsilon)$ with respect to the number of variable nodes and check nodes in the *original* graph. Therefore, $\Lambda_{\mathtt{G}(\epsilon)}(1)\le 1$ is the number of variable nodes in $\mathtt{G}(\epsilon)$ divided by n. Analogously, $\Gamma_{\mathtt{G}(\epsilon)}(1)\le 1$ is the number of check nodes in $\mathtt{G}(\epsilon)$ divided by $n\Lambda'(1)/\Gamma'(1)$.

It is shown in [16] that, conditioned on the degree profile of the residual graph, $\mathtt{G}(\epsilon)$ is uniformly distributed. The dd pair $\Xi_{\mathtt{G}(\epsilon)}$ itself is of course a random quantity because of the channel randomness. However, it is is sharply concentrated around its expected value. For increasing blocklengths this expected value converges to $\Xi_\epsilon = (\Lambda_\epsilon,\Gamma_\epsilon)$, which is given by[6]

$$\Lambda_\epsilon(\mathtt{z}) \triangleq \epsilon\Lambda(\mathtt{z}\mathtt{y})\,, \tag{8}$$

$$\Gamma_\epsilon(\mathtt{z}) \triangleq \Gamma(1-\mathtt{x}+\mathtt{z}\mathtt{x})-\Gamma(1-\mathtt{x})-\mathtt{z}\mathtt{x}\Gamma'(1-\mathtt{x})\,. \tag{9}$$

Here, $\mathtt{x}$ and $\mathtt{y}$ denote the fraction of erased messages at the fixed point of the BP decoder. More precisely, $\mathtt{x}\in[0,1]$ is the largest solution of $\mathtt{x}=\epsilon\lambda(1-\rho(1-\mathtt{x}))$ and $\mathtt{y}=1-\rho(1-\mathtt{x})$. The precise concentration statement follows.

[6]The standard dd pair from the node perspective of the residual graph when transmission takes place over BEC(ϵ) is then simply given by $\left(\frac{\Lambda_\epsilon(\mathtt{x})}{\Lambda_\epsilon(1)},\frac{\Gamma_\epsilon(\mathtt{x})}{\Gamma_\epsilon(1)}\right)$.

Lemma 6: Let $\epsilon \in (0,1]$ be a continuity point of $\mathtt{x}(\epsilon)$ (we shall call such an ϵ *non-exceptional*). Then, for any $\xi > 0$,

$$\lim_{n\to\infty} \Pr\{d(\Xi_{\mathtt{G}(\epsilon)}, \Xi_\epsilon) \geq \xi\} = 0\,. \tag{10}$$

Here, $d(\cdot,\cdot)$ denotes the L_1 distance

$$d(\Xi, \tilde{\Xi}) \triangleq \sum_{\mathtt{l}} |\Lambda_{\mathtt{l}} - \tilde{\Lambda}_{\mathtt{l}}| + \sum_{\mathtt{r}} |\Gamma_{\mathtt{r}} - \tilde{\Gamma}_{\mathtt{r}}|\,. \tag{11}$$

The proof is deferred to Appendix II.

Under the zero-codeword assumption, the set of codewords compatible with the received bits coincides with the set of codewords of the residual graph. Their expected number can be computed through standard combinatorial tools. The key idea here is that, under suitable conditions on the dd pair (of the residual graph), the actual rate of codes from the (residual) ensemble is close to the design rate. We state here a slightly strengthened version of this result from [27].

Lemma 7: Let $\mathtt{G}$ be chosen uniformly at random from the ensemble LDPC(n,Ξ) =LDPC(n,Λ,Γ), let $r_{\mathtt{G}}$ be its rate and $r \triangleq 1 - \Lambda'(1)/\Gamma'(1)$ be the design rate. Consider the function $\Psi_\Xi(u)$,

$$\Psi_\Xi(u) = -\,\Lambda'(1)\log_2\left[\frac{(1+uv)}{(1+u)(1+v)}\right] + \sum_{\mathtt{l}} \Lambda_{\mathtt{l}} \log_2\left[\frac{1+u^{\mathtt{l}}}{2(1+u)^{\mathtt{l}}}\right] + \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{\mathtt{r}}\Gamma_{\mathtt{r}}\log_2\left[1+\left(\frac{1-v}{1+v}\right)^{\mathtt{r}}\right], \tag{12}$$

$$v = \left(\sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}}}{1+u^{\mathtt{l}}}\right)^{-1}\left(\sum_{\mathtt{l}}\frac{\lambda_{\mathtt{l}}u^{\mathtt{l}-1}}{1+u^{\mathtt{l}}}\right). \tag{13}$$

Assume that $\Psi_\Xi(u)$ takes on its global maximum in the range $u \in [0,\infty)$ at $u = 1$. Then there exists $B > 0$ such that, for any $\xi > 0$, and $n > n_0(\xi,\Xi)$,

$$\Pr\{|r_{\mathtt{G}} - r(\Lambda,\Gamma)| > \xi\} \leq e^{-Bn\xi}\,.$$

Moreover, there exist $C > 0$ such that, for $n > n_0(\xi,\Xi)$,

$$\mathbb{E}[|r_{\mathtt{G}} - r(\Lambda,\Gamma)|] \leq C\frac{\log n}{n}\,.$$

Proof: The idea of the proof is the following. For any parity-check ensemble we have $r_{\mathtt{G}} \geq r(\Lambda,\Gamma)$. If it is true that the expected value of the rate (more precisely, the logarithm of the expected number of codewords divided by the length) is close to the design rate, then we can use the Markov inequality to show that most codes have rate close to the design rate.

Let us start by computing the exponent of the expected number of codewords. We know from [27]–[36] that the expected number of codewords involving E edges is given by

$$\mathbb{E}[N_{\mathtt{G}}(E)] = \frac{\text{coef}\left\{\prod_{\mathtt{l}}(1+u^{\mathtt{l}})^{n\Lambda_{\mathtt{l}}}\prod_{\mathtt{r}} q_{\mathtt{r}}(v)^{n\frac{\Lambda'(1)}{\Gamma'(1)}\Gamma_{\mathtt{r}}}, u^E v^E\right\}}{\binom{n\Lambda'(1)}{E}},$$

where $q_{\mathtt{r}}(v) = ((1+v)^{\mathtt{r}} + (1-v)^{\mathtt{r}})/2$. Let n tend to infinity and define $e = E/(n\Lambda'(1)$. From standard arguments presented in the cited papers it is known that, for a fixed e, the exponent $\lim_{n\to\infty}\frac{1}{n}\log_2\big(\mathbb{E}[N_{\mathtt{G}}(en\Lambda'(1))]\big)$ is given by the infimum with respect to $u, v > 0$ of

$$\sum_{\mathtt{l}}\Lambda_{\mathtt{l}}\log_2(1+u^{\mathtt{l}}) - \Lambda'(1)e\log_2 u + \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{\mathtt{r}}\Gamma_{\mathtt{r}}\log_2 q_{\mathtt{r}}(v) - \Lambda'(1)e\log_2 v - \Lambda'(1)h(e). \tag{14}$$

We want to determine the exponent corresponding to the expected number of codewords, i.e., $\lim_{n\to\infty}\frac{1}{n}\log_2\big(\mathbb{E}[N_{\mathtt{G}}]\big)$, where $N_{\mathtt{G}} = \sum_E N_{\mathtt{G}}(E)$. Since there is only a polynomial number of "types" (numbers E) this exponent is equal to the supremum of (14) over all $0 \leq e \leq 1$. In summary, the sought after exponent is given by a stationary point of the function stated in (14) with respect to u, v and e.

Take the derivative with respect to e. This gives $e = uv/(1+uv)$. If we substitute this expression for e into (14), subtract the design rate $r(\Lambda,\Gamma)$, and rearrange the terms somewhat we get (12). Next, if we take the derivative with respect to u and solve for v we get get (13). In summary, $\Psi_\Xi(u)$ is a function so that

$$\log_2\mathbb{E}[N_{\mathtt{G}}] = n\{r(\Lambda,\Gamma) + \sup_{u\in[0,\infty)}\Psi_\Xi(u) + \omega_n\}\,,$$

where $\omega_n = o(1)$. In particular, by explicit computation we see that $\Psi_\Xi(u = 1) = 0$. A closer look shows that $u = 1$ corresponds to the exponent of codewords of weight $n/2$. Therefore, the condition that the global maximum of $\Psi_\Xi(u)$ is achieved at $u = 1$ is equivalent to the condition that the expected weight enumerator is dominated by codewords of weight (close to) $n/2$. Therefore,

$$\Pr\{r_{\mathtt{G}} \geq r(\Lambda,\Gamma) + \xi\} = \Pr\left\{N_{\mathtt{G}} \geq 2^{n(\xi-\omega_n)}\mathbb{E}[N_{\mathtt{G}}]\right\} \leq e^{-Bn\xi}\,,$$

where the step follows from the Markov inequality if $B = (\log 2)/2$ and $\omega_n \leq \xi/2$ for any $n \geq n_0$.

Finally, we observe that, since $r_{\mathtt{G}} \leq 1$

$$\mathbb{E}[|r_{\mathtt{G}} - r(\Lambda,\Gamma)|] \leq \xi + e^{-Bn\xi}\,,$$

and the second claim follows by choosing $\xi = \log n/Bn$. ∎

We would like to apply this result to the residual graph $\mathtt{G}(\epsilon)$. Since the degree profile of $\mathtt{G}(\epsilon)$ is a random variable, we need a preliminary observation on the "robustness" of the hypotheses in the Lemma 7.

Lemma 8: Let $\Psi_\Xi(\cdot)$ be defined as in Lemma 7. Then $\Psi_\Xi(u)$ achieves its maximum over $u \in [0,+\infty)$ in $[0,1]$.

Moreover, there exists a constant $A > 0$ such that, for any two degree distribution pairs $\Xi = (\Lambda,\Gamma)$ and $\tilde{\Xi} = (\tilde{\Lambda},\tilde{\Gamma})$, and any $u \in [0,1]$,

$$|\Psi_\Xi(u) - \Psi_{\tilde{\Xi}}(u)| \leq A\,d(\Xi,\tilde{\Xi})\,(1-u)^2\,. \tag{15}$$

For the proof we refer to Appendix II.

We turn now to the main result of this section.

Theorem 10: Let $\mathtt{G}$ be a code picked uniformly at random from the ensemble LDPC(n,Λ,Γ) and let $H_{\mathtt{G}}(X|Y)$ be the conditional entropy of the transmitted message when the code is used for communicating over BEC(ϵ). Denote by $P_\epsilon(\mathtt{x},\mathtt{y})$ the corresponding trial entropy. Let $\Xi_\epsilon = (\Lambda_\epsilon,\Gamma_\epsilon)$ be the typical degree distribution pair of the residual graph, see

Eqs. (8), (9), and $\Psi_{\Xi_\epsilon}(x)$ be defined as in Lemma 7, Eq. (12).

Assume that $\Psi_{\Xi_\epsilon}(u)$ achieves its global maximum as a function of $u \in [0, \infty)$ at $u = 1$, with $\Psi''_{\Xi_\epsilon}(1) < 0$, and that ϵ is non-exceptional. Then

$$\lim_{n\to\infty} \frac{1}{n}\mathbb{E}[H_{\mathtt{G}}(X|Y)] = P_\epsilon(\mathtt{x}, \mathtt{y})\,, \tag{16}$$

where $\mathtt{x} \in [0, 1]$ is the largest solution of $\mathtt{x} = \epsilon\lambda(1-\rho(1-\mathtt{x}))$ and $\mathtt{y} = 1 - \rho(1 - \mathtt{x})$.

Proof: As above, we denote by $\mathtt{G}(\epsilon)$ the residual graph after BP decoding and by $r_{\mathtt{G}(\epsilon)}$ its rate normalized to the original blocklength n. Notice that $H_{\mathtt{G}}(X|Y) = nr_{\mathtt{G}(\epsilon)}$: iterative decoding does not exclude any codeword compatible with the received bits. Furthermore, the design rate (always normalized to n) for the dd pair of the residual graph is

$$r(\Xi_{\mathtt{G}(\epsilon)}) = \Lambda_{\mathtt{G}(\epsilon)}(1) - \frac{\Lambda'(1)}{\Gamma'(1)}\,\Gamma_{\mathtt{G}(\epsilon)}(1)\,.$$

We further introduce the notation r_ϵ for the design rate of the typical dd pair of the residual graph. Using Eqs. (8) and (9), we can find

$$\begin{aligned} r_\epsilon &= \Lambda'(1)\rho(1-\mathtt{x})\mathtt{x} - \frac{\Lambda'(1)}{\Gamma'(1)}[1-\Gamma(1-\mathtt{x})] + \epsilon\Lambda(\mathtt{y}) \\ &= P_\epsilon(\mathtt{x}, \mathtt{y}), \end{aligned}$$

where the last step follows from the fixed-point condition $\mathtt{y} = 1 - \rho(1 - \mathtt{x})$.

Since by assumption $\Psi_{\Xi_\epsilon}(u)$ achieves its global maximum at $u = 1$, with $\Psi''_{\Xi_\epsilon}(1) < 0$, and $\Psi_{\Xi_\epsilon}(1) = 0$, there exists a positive constant δ such that $\Psi_{\Xi_\epsilon}(u) \le -\delta(1 - u)^2$ for any $u \in [0, 1]$. As a consequence of Lemma 8, there exist a $\xi > 0$ such that, for any dd pair Ξ, with $d(\Xi, \Xi_\epsilon) \le \xi$, $\Psi_\Xi(u) \le -\delta(1-u)^2/2$ for $u \in [0, 1]$.

Let $\mathrm{Pr}_\epsilon(\tilde\Xi)$ be the probability that the degree distribution pair of the residual graph $\mathtt{G}(\epsilon)$ is $\tilde\Xi = (\tilde\Lambda, \tilde\Gamma)$. Denote by $\tilde{\mathbb{E}}$ expectation with respect to a uniformly random code in the $(\tilde n, \tilde\Lambda, \tilde\Gamma)$ ensemble (here $\tilde n \triangleq n\tilde\Lambda(1)$). Denote by $\mathcal{N}(\xi)$ the set of dd pairs $\tilde\Xi$, such that $d(\tilde\Xi, \Xi_\epsilon) \le \xi$. The above remarks imply that we can apply Lemma 7 to any ensemble in $\mathcal{N}(\xi)$. Then

$$\begin{aligned} \frac{1}{n}\mathbb{E}[H_{\mathtt{G}}(X|Y)] &= \sum_{\tilde\Xi} \mathrm{Pr}_\epsilon(\tilde\Xi)\,\tilde{\mathbb{E}}[r_{\mathtt{G}(\epsilon)}] \\ &= \sum_{\tilde\Xi\in\mathcal{N}(\xi)} \mathrm{Pr}_\epsilon(\tilde\Xi)\,\tilde{\mathbb{E}}[r_{\mathtt{G}(\epsilon)}] + \omega(n, \xi). \end{aligned}$$

The remainder can be estimated by noticing that $r_{\mathtt{G}(\epsilon)} \le 1$ while the probability of $\tilde\Xi \notin \mathcal{N}(\epsilon)$ is bounded by Lemma 6. Therefore

$$\lim_{n\to\infty} \omega(n, \xi) = 0\,.$$

Now we can apply Lemma 7 to get

$$\begin{aligned} \left|\frac{1}{n}\mathbb{E}[H_{\mathtt{G}}(X|Y)] - r_\epsilon\right| &\le \sum_{\tilde\Xi\in\mathcal{N}(\xi)} \mathrm{Pr}_\epsilon(\tilde\Xi)\,|\tilde{\mathbb{E}}[r_{\mathtt{G}(\epsilon)}] - r(\tilde\Xi)| \\ &\quad + \sum_{\tilde\Xi\in\mathcal{N}(\xi)} \mathrm{Pr}_\epsilon(\tilde\Xi)\,|r(\tilde\Xi) - r_\epsilon| + \omega(n, \xi) \\ &\le \sum_{\tilde\Xi\in\mathcal{N}(\xi)} \mathrm{Pr}_\epsilon(\tilde\Xi)\,|r(\tilde\Xi) - r_\epsilon| + \omega'(n, \xi)\,, \end{aligned}$$

where $\omega'(n, \xi) = \omega(n, \xi) + C \log n/n$. Notice that there exist $B > 0$ such that for any pair Ξ_1, Ξ_2

$$|r(\Xi_1) - r(\Xi_2)| \le B\,d(\Xi_1; \Xi_2)\,.$$

Therefore,

$$\lim_{n\to\infty} \left|\frac{1}{n}\mathbb{E}[H_{\mathtt{G}}(X|Y)] - r_\epsilon\right| \le B\xi\,.$$

The claim follows by noticing that ξ can be chosen arbitrarily small. ■

Theorem 10 allows to compute the exact MAP threshold whenever the required conditions are verified. An explicit characterization is given below.

Corollary 2: Consider transmission over BEC(ϵ) using elements picked uniformly at random from the ensemble (Λ, Γ). Let $\mathtt{x}^*, \mathtt{y}^* > 0$ be the DE fixed-point achieved by the BP decoder at a non-exceptional erasure probability ϵ^* (i.e., $\mathtt{x}^* \in (0, 1]$ is the largest solution of $\mathtt{x}^* = \epsilon^*\lambda(1 - \rho(1 - \mathtt{x}^*))$). Assume that $P_{\epsilon^*}(\mathtt{x}^*, \mathtt{y}^*) = 0$ and that $\Psi_{\Xi_{\epsilon^*}}(u) \le 0$ for $u \in [0, +\infty)$ together with $\Psi''_{\Xi_{\epsilon^*}}(1) < 0$. Let $\mathcal{W} \subseteq [0, +\infty)$ be the set of points $u \ne 1$ such that $\Psi_{\Xi_{\epsilon^*}}(u) = 0$. If, for any $u \in \mathcal{W}$, $\partial_\epsilon\Psi_{\Xi_{\epsilon^*}}(u) < \partial_\epsilon\Psi_{\Xi_{\epsilon^*}}(1)$, then $\epsilon^{\text{MAP}} = \epsilon^*$.

Proof: We claim that there exist a $\delta > 0$ such that the hypothesis of Theorem 10 are verified for any $\epsilon \in (\epsilon^*, \epsilon^* + \delta)$. Before proving this claim, let us show that it implies the thesis. Consider any $\epsilon \in (\epsilon^*, \epsilon^* + \delta)$ and let $\mathtt{x}, \mathtt{y}$ be the corresponding density evolution fixed point. Then

$$\lim_{n\to\infty} \frac{1}{n}\mathbb{E}[H(X|Y)] = P_\epsilon(\mathtt{x}(\epsilon), \mathtt{y}(\epsilon)) \qquad \forall\epsilon \in (\epsilon^*, \epsilon^* + \delta)\,.$$

Moreover $P_{\epsilon^*}(\mathtt{x}(\epsilon^*), \mathtt{y}(\epsilon^*)) = 0$ by hypothesis and

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon}P_\epsilon(\mathtt{x}(\epsilon), \mathtt{y}(\epsilon)) = \Lambda(\mathtt{y}(\epsilon)) > 0\,.$$

Therefore $P_\epsilon(\mathtt{x}(\epsilon), \mathtt{y}(\epsilon)) > 0$ for any $\epsilon > \epsilon^*$. This implies $\epsilon^{\text{MAP}} \le \epsilon^*$. On the other hand $\mathbb{E}[H(X|Y)]$ is strictly increasing with ϵ. This implies

$$\lim_{n\to\infty} \frac{1}{n}\mathbb{E}[H(X|Y)] = 0, \qquad \forall\epsilon \in [0, \epsilon^*],$$

which in turn implies $\epsilon^{\text{MAP}} \ge \epsilon^*$ and, therefore, $\epsilon^{\text{MAP}} = \epsilon^*$.

Let us now prove the claim. By assumption ϵ^* is non-exceptional and therefore the residual dd pair Ξ_ϵ is continuous at ϵ^*. This implies, via Lemma 8 that, for any $\xi > 0$, there exist δ such that for $\epsilon \in [\epsilon^*, \epsilon^* + \delta)$ and any $u \in [0, 1]$,

$$|\Psi_{\Xi_\epsilon}(u) - \Psi_{\Xi_{\epsilon^*}}(u)| \le \xi(1 - u)^2\,.$$

Together with $\Psi''_{\Xi_\epsilon}(1) < 0$, this implies that, if δ is small enough, $u = 1$ is a local maximum of $\Psi_{\Xi_\epsilon}(u)$. It follows

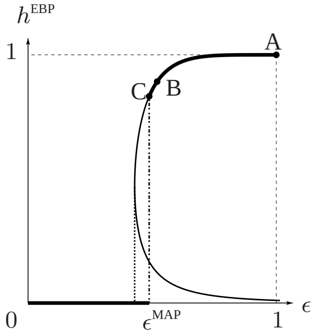


Fig. 9. (E)BP EXIT function $h^{\text{EBP}}(\epsilon)$.

from the hypotheses on $\partial_\epsilon \Psi_{\Xi_{\epsilon^*}}(u)$, $u \in \mathcal{W}$, that it is also a global maximum. ■

The conditions in the above corollary are relatively easy to verify. Let us demonstrate this by means of two examples.

Example 6 (Ensemble LDPC(x^2, x^5)): Consider the $(3,6)$-regular LDPC ensemble. For convenience of the reader its EBP EXIT curve is repeated in Fig. 9.

Let us apply Theorem 10. We start with $\epsilon_A = 1$ (point A). The residual degree distribution at this point corresponds of course to the $(3,6)$-ensemble itself. As shown in the left-most picture in Fig. 10, the corresponding function $\Psi_\Xi(u)$ has only a single maximum at $u = 1$ and one can verify that $\Psi''_\Xi(1) < 0$. Therefore, by Lemma 7 we know that with high probability the rate of a randomly chosen element from this ensemble is close to the design rate. Next, consider the

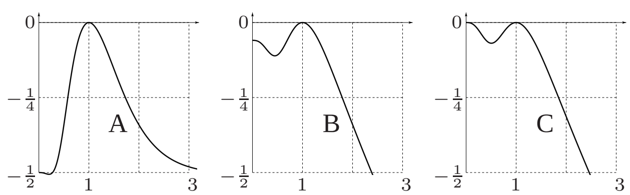


Fig. 10. Function $\Psi_\Xi(u)$ for the dd pair formed by the residual ensemble in A, B and C.

point $\epsilon_B = 0.52$ (point B). Again, the conditions are verified, and therefore the conditional entropy at this point is given by equation (16). We get $H(X \mid Y(\epsilon_B)) \approx 0.02755$. Finally, consider the "critical" point $\epsilon_C \approx 0.48815$. As one can see from the right-most picture in Fig. 10, this is the point at which a second global maximum appears. Just to the right of the point the conditions of Theorem 10 are still fulfilled, whereas to the left of it they are violated. Further, at this point Eq. (16) states that $H(X \mid Y(\epsilon_C)) = 0$. We conclude that $\epsilon^{\text{MAP}} = \epsilon_C \approx 0.48815$, confirming our result from Example 4. Since the bound is tight at the MAP threshold it follows that $h^{\text{MAP}} = h^{\text{BP}}$ for all points "to the right" of the MAP threshold (this is true since $h^{\text{MAP}} \leq h^{\text{BP}}$ always, and the tightness of the bound at the MAP threshold shows that the area under h^{BP} is exactly equal to the rate). We see that in this simple case Theorem 10 allows us to construct the complete MAP EXIT curve.

Example 7 (Ensemble LDPC($\frac{3x+3x^2+4x^{13}}{10}, x^6$)): Consider the ensemble described in Fig. 3. Its EPB EXIT curve is repeated for the convenience of the reader in Fig. 11. The

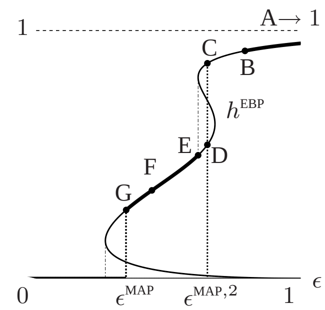


Fig. 11. (E)BP EXIT function $h^{\text{EBP}}(\epsilon)$.

corresponding BP EXIT curve is shown in detail in Fig. 5. A further discussion of this ensemble can be found in Example 10. Let us again apply Theorem 10. We start with $\epsilon_A = 1$

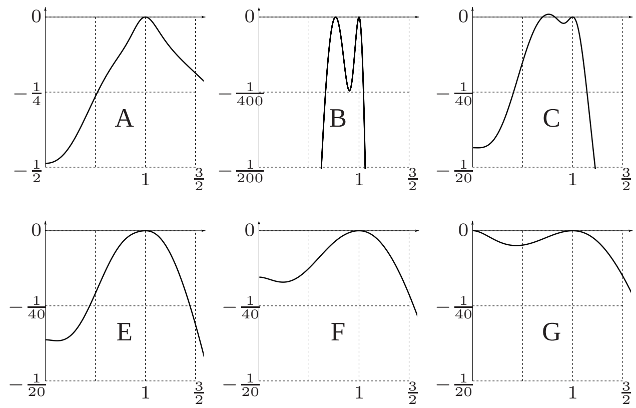


Fig. 12. Function $\Psi_\Xi(u)$ for the dd pair formed by the residual ensemble in A, B, C, E, F and G.

(point A). The residual degree distribution corresponds of course to the ensemble itself. As the top left-most picture in Fig. 11 shows, the hypotheses are fulfilled and we conclude again that with high probability the rate of a randomly chosen element from this ensemble is close to the design rate which is equal to $r \approx 0.4872$. Now decrease ϵ smoothly. The conditions of Theorem 10 stay fulfilled until we get to $\epsilon_B \approx 0.5313$ (point B). At this point a second global maximum of the function $\Psi_\Xi(u)$ occurs. As the pictures in the bottom row of Fig. 11 show, the hypotheses of Theorem 10 are again fulfilled over the whole segment from E (the first threshold of the BP decoder corresponding to $\epsilon_E \approx 0.5156$) till G. In particular, at the point G, which corresponds to $\epsilon_G = \epsilon^{\text{MAP}} \approx 0.4913$, the trial entropy reaches zero, which shows that this is the MAP threshold.

We see that for this example Theorem 10 allows us to construct the MAP EXIT curve for the segment from A to B and the segment from E to G. Over both these segments we have $h^{\text{MAP}} = h^{\text{BP}}$. In summary, we can determine the MAP threshold and we see that the balance condition applies "at the jump G" (the MAP threshold). But the straightforward application of Theorem 10 does not provide us with a means of determining h^{MAP} between the points B and D. Intuitively, h^{MAP} should go from B to C (which corresponds to $\epsilon^C \approx 0.5156$). At this point one would hope that a local balance condition again applies and that the MAP EXIT curve jumps to the "lower branch" to point D. It should then continue smoothly until the point G (the MAP threshold) at which it finally jumps to zero. As we will discuss in more detail in Example 10, after our analysis of the M decoder, this is indeed true, and h^{MAP} is as shown in Fig. 3.

Assuming Theorem 10 applies, we know that at the MAP threshold the matrix corresponding to the residual graph becomes a full rank square matrix. What happens at the jump at point C? At this point the matrix corresponding to the residual graph takes, after some suitable swapping of columns and rows, the form

$$\begin{pmatrix} U & V \\ 0 & W \end{pmatrix},$$

where W is a full rank square matrix of dimension $\epsilon_C(\Lambda(\mathrm{y}_C) - \Lambda(\mathrm{y}_D))$. The MAP decoder can therefore solve the part of the equation corresponding to the submatrix W.

VI. Maxwell Construction

The balance condition described in Section I-B and Section IV is strongly reminiscent of the well-known "Maxwell construction" in the theory of phase transitions. This is described briefly in Fig. 13.

A. Maxwell Decoder

Inspired by the statistical mechanics analogy, we will explain the balance condition (shown on the right in Fig. 1) which determines the MAP threshold by analyzing a "BP decoder with guessing". The state of the algorithm can be associated to a point moving along the EBP EXIT curve. The evolution starts at the point of full entropy and ends at zero entropy. The analysis of this algorithm is also most conveniently phrased in terms of the EBP EXIT curve and implies a proof of Theorem 9. Because of this balance condition we term this decoding algorithm the Maxwell (M) decoder. Note that a similar algorithm is discussed in [13] although it is motivated by some more practical concerns.

Analogously to the usual BP decoder for the erasure channel, the M decoder admits two equivalent descriptions: either as a *sequential* (i.e., bit-by-bit in the spirit of [16]) or as a *message-passing* algorithm. While the former approach is more intuitive, the latter allows for a simpler analysis. We shall first describe the M decoder as a sequential procedure and sketch the main features of its behavior. In the next section we will turn to a message-passing setting and complete its analysis.

Given the received word which was transmitted over BEC(ϵ), the decoder proceeds iteratively as does the standard BP decoder. At each time step a parity-check equation involving a single undetermined variable is chosen and used to determine the value of the variable. This value is substituted in any parity-check equation involving the same variable. If at any time the iterative decoding process gets stuck in a non-empty stopping set, a position $i \in [n]$ is chosen uniformly at random. The decoder is said to *guess* a bit. If the bit associated to this position is not known yet, the decoder replicates[7] any running copy of the decoding process, and it proceeds by running one copy of each process under the assumption that $x_i = \mathtt{0}$ and the other one under the assumption that $x_i = \mathtt{1}$.

It can happen that during the decoding process a variable receives non-erased messages from several check nodes. In such a case, these messages can be distinct and, therefore, inconsistent. Such an event is termed a *contradiction*. Any running copy of the decoding process which encounters a contradiction terminates. The decoding process finishes once all bits have been determined. At this point, each surviving copy outputs the determined word. Each such word is by construction a codeword which is compatible with the received information. Vice versa, for each codeword which is compatible with the received information, there will be a surviving copy. In other words, the M decoder performs a complete *list decoding* of the received message. Fig. 14 shows the workings of the M decoder by means of a specific example.

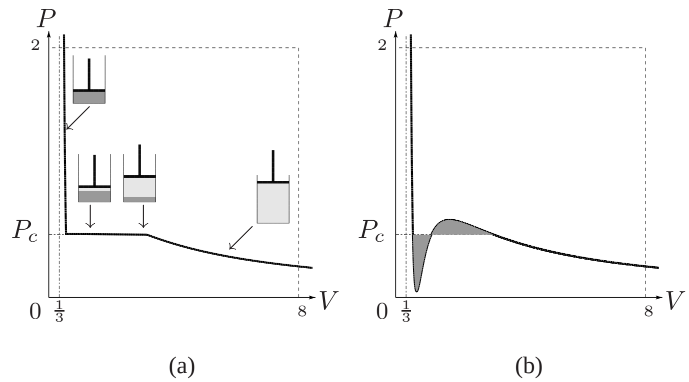
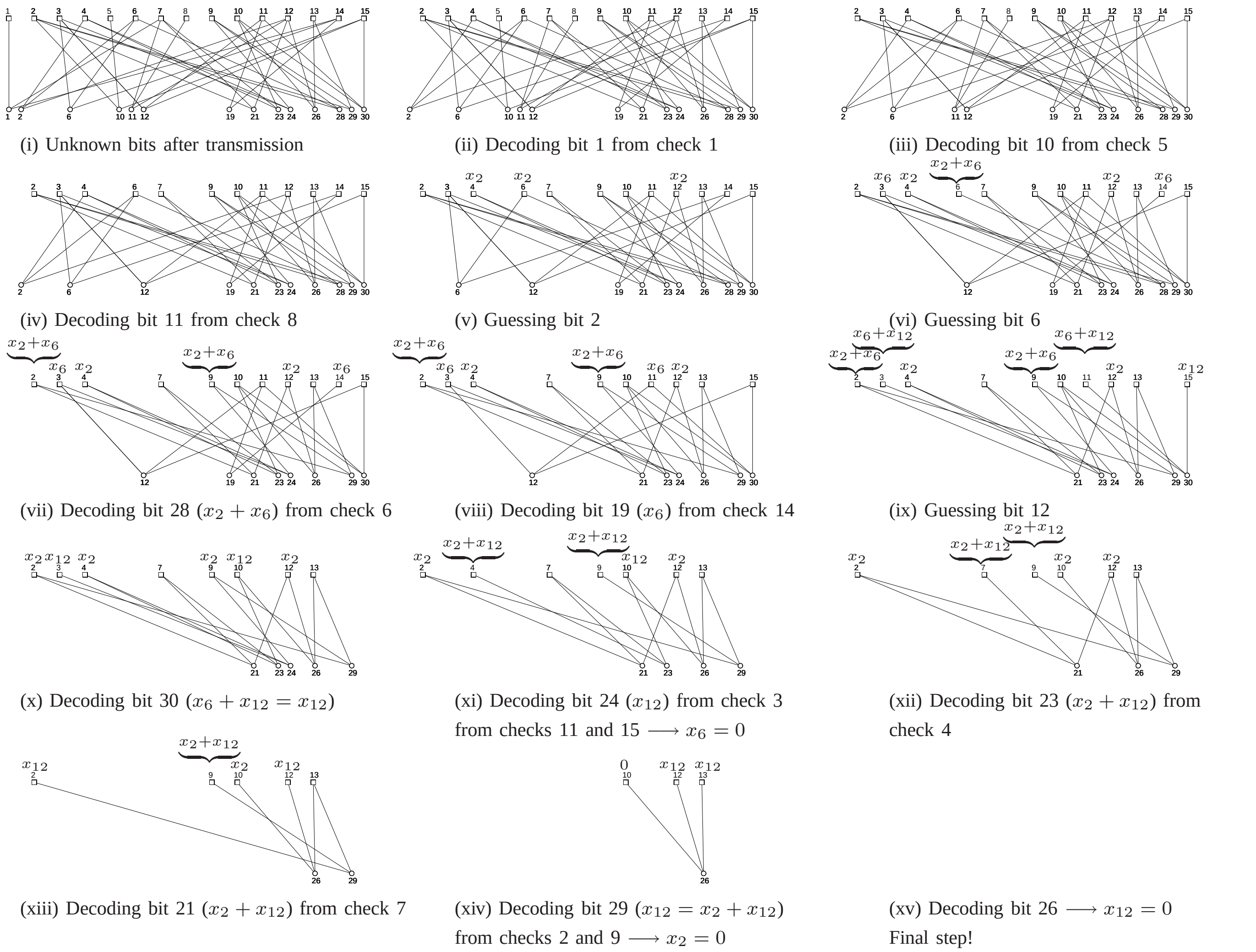


Fig. 13. Maxwell construction in thermodynamics. (a) Pressure-volume diagram for the liquid-vapor phase transition (b) Van der Waals curve (using reduced variables, given by $(p + \frac{2}{V^2})(3V - 1) = 8T$ at the reduced temperature $T = 0.85$) and the Maxwell construction. Consider the case of a liquid-gas phase transition of water. If a small amount of liquid is placed in a completely empty (and hermetically closed) large container at room temperature, the water evaporates. The vapor exerts pressure on the walls of the container. By gradually reducing the volume of the container, we increase the vapor pressure P until it reaches a *critical* value P_c (which depends on the temperature). At this point the vapor condensates into liquid water. The pressure stays constant throughout this transformation. When there is no space left for the vapor, the pressure starts to rise again, and as shown in (a) it does so very quickly (since it is difficult to compress water). In many theoretical descriptions of this phenomenon, a non-monotonic pressure-volume curve is obtained like in (b) with the Van Der Waals model. The Maxwell construction allows to modify the "unphysical" part of this curve and to obtain a consistent result. We want to join the two decreasing branches of the theoretical curve with a constant-pressure line, as observed in experiments. At which height should we placed the horizontal line? The basic idea of the Maxwell construction is that, at the critical pressure P_c, the vapor and the liquid are in "equilibrium". This means that we can transform an infinitesimal quantity of vapor into liquid (or vice versa) without doing any "work" on the system. Because of this reason, the vapor begins its transformation into liquid at P_c. The work done on the system in an infinitesimal transformation is $P\,\mathrm{d}V$, where $\mathrm{d}V$ represents the variation of the volume. Using this fact, it can be shown that the above equilibrium condition implies the equality of the areas of the two regions between the horizontal line and the original non-monotonous pressure-volume curve. See, e.g., [37].

[7] Here we describe the decoder as a 'breadth-first' search procedure: at each bifurcation we explore in parallel all the available options. One can easily construct an equivalent 'depth-first' search: first take a complete sequence of choices and, if no codeword is found, backtrack.



(i) Unknown bits after transmission (ii) Decoding bit 1 from check 1 (iii) Decoding bit 10 from check 5

(iv) Decoding bit 11 from check 8 (v) Guessing bit 2 (vi) Guessing bit 6

(vii) Decoding bit 28 ($x_2 + x_6$) from check 6 (viii) Decoding bit 19 (x_6) from check 14 (ix) Guessing bit 12

(x) Decoding bit 30 ($x_6 + x_{12} = x_{12}$) (xi) Decoding bit 24 (x_{12}) from check 3 from checks 11 and 15 $\longrightarrow x_6 = 0$ (xii) Decoding bit 23 ($x_2 + x_{12}$) from check 4

(xiii) Decoding bit 21 ($x_2 + x_{12}$) from check 7 (xiv) Decoding bit 29 ($x_{12} = x_2 + x_{12}$) from checks 2 and 9 $\longrightarrow x_2 = 0$ (xv) Decoding bit 26 $\longrightarrow x_{12} = 0$ Final step!

Fig. 14. M decoder applied to a simple example: a $(3, 6)$ LDPC code of length $n = 30$. Assume that the all-zero codeword has been transmitted. At the decoder, the received (i.e., known and equal to 0) bits are removed from the bipartite graph. The remaining graph is shown in (i). The first phase is the standard BP algorithm: in the first three steps, the decoder proceeds as the standard BP decoder and determines the bits 1, 10 and 11, until it gets stuck in a stopping set shown in (iv). The second phase is distinct to the M decoder: it is the guessing/contradiction phase. The decoder guesses the (randomly chosen) bit 2: this means that it creates two simultaneously running copies, one which proceeds under the assumption that bit 2 takes the value 0, the other which assumes that this bit takes the value 1. The decoder then proceeds as the standard BP algorithm. Any time it gets stuck, it guesses a new bit and duplicates the number of simultaneously running copies. This process continues until a contradiction occurs, e.g., at the 9$^{\text{th}}$ step (ix): the variable node x_{30} (either $x_{30} = 0$ or $x_{30} = 1$ depending of which copy we are considering) is connected to two check nodes of degree one. The incoming messages from those nodes are $x_6 + x_{12}$ and x_{12}, respectively. Consistency now requires that $x_6 + x_{12} = x_{12}$, i.e., that $x_6 = 0$, such that only the decoding copies corresponding to $x_6 = 0$ survive. Phases of guessing and phases of standard BP decoding might alternate. Decoding is successful (in the sense that a MAP decoder would have succeeded) if only a single copy survives at the very end of the decoding process. "Contradictions" can be seen as "confirmations" or "conditions" in this message-passing setting.

The corresponding instance of the decoding process is depicted in Fig. 15 from the perspective of the various simultaneous copies.

Let us briefly describe how the analysis of the above algorithm is related to the balance condition and the proof of Theorem 9. Instead of explaining the balance between the areas as shown in Fig. 1, we consider the balance of the two areas shown in Fig. 2. Note that these two areas differ from the previous ones only by a common term, so that the condition for balance stays unchanged. From the above description it follows that at any given time t there are $2^{\hat{H}(t)}$ copies running, where $\hat{H}(t)$ is a natural number which evolves with time. In fact, each time a bit is guessed, the number of copies is doubled, while it is halved each time a contradiction occurs. Call t_{out} the time at which all transmitted bits have been determined and the list of decoded words is output (t_{out} does not depend upon the particular copy of the process in consideration). Since the M decoder is a complete list decoder and since all output codewords have equal posterior probability, $H(X|Y) = \hat{H}(t_{\text{out}})$. On the other hand, $\hat{H}(t_{\text{out}})$ is equal to the total number of guesses minus the total number of contradictions which occurred during the evolution of the algorithm. As we will see in greater detail in the next section, the total number of guesses divided by n converges to the area of the dark gray region in Fig. 2 (a), while the total number of contradictions divided by n is asymptotically not larger than the dark gray area in Fig. 2 (b). Therefore, as long as ϵ is strictly larger than the value at which we have balance, call this value $\overline{\epsilon}^{\text{MAP}}$, $\lim_{n\to\infty} \frac{\mathbb{E}[H(X|Y(\epsilon))]}{n} > 0$. This implies that $\overline{\epsilon}^{\text{MAP}} \geq \epsilon^{\text{MAP}}$.

We expect that the number of contradictions divided by n is indeed asymptotically equal to the dark gray area in Fig. 2 (b). Although we are not able to prove this statement in full generality, it follows from Theorem 10, whenever the hypotheses hold.

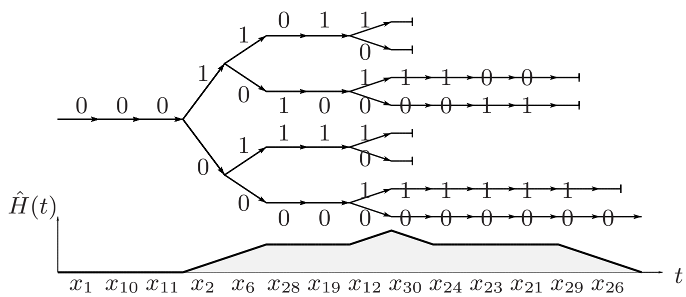


Fig. 15. M decoder applied to the simple example shown in Fig. 14. The all-zero codeword is decoded. The initial phase coincides with standard message-passing BP algorithm: a single copy of the process decodes a bit at a time. After three steps, the BP decoder gets stuck in a stopping set and several steps of guessing follow. During this phase the associated entropy $\hat{H}(t)$ increases. After this guessing phase, the standard message passing phase resumes. More and more copies terminate due to inconsistent messages arriving at variable nodes. At the end only one copy survives. This shows that this example has a unique MAP solution.

B. Message-Passing Setting

We describe now a message-passing algorithm that is equivalent to the above sequential formulation. First note that because of the code linearity, the symmetries of the channel and the decoding algorithm, we can simplify our analysis by making the all-zero codeword assumption, see [17].

We assign a label μ_i^ϵ to the variable node of index i. The label can take three possible values $\mu_i^\epsilon \in \{\mathtt{0}, *, \mathtt{g}\}$. It can be viewed as the output of some fictitious channel, and indicates how the algorithm is going to treat that variable node. The fictitious channel is memoryless: each variable node is assigned a $\mathtt{0}$ with probability $1-\epsilon$, a $*$ with probability $\epsilon(1-\gamma)$ and a $\mathtt{g}$ with probability $\epsilon\gamma$. The parameter γ represents the fraction of *guesses* ventured so far.

The new message-passing algorithm employs left-to-right messages μ^{x} and right-to-left messages μ^{y}, all of which take values in $\{\mathtt{0}, *, \mathtt{g}\}$. The meaning of the 0 message and the $*$ message is the same as for the BP algorithm. A $\mathtt{g}$ message indicates that either the bit from which this message emanates has been guessed or that the value of this bit can be expressed as a linear combination of other bit values which have been guessed. Operationally, we can think of the message $\mu_i = \mathtt{g}$ as being a shorthand for a non-empty list of indices $\Theta_i = \{j_1, \dots, j_k\}$. This list indicates that x_i is expressible as $x_i = x_{j_1} + \dots + x_{j_k}$, where $\{x_{j_1}, \cdots, x_{j_k}\}$ is a set of guessed bits.

This motivates the following update rules for the parity-check and variable nodes shown in Fig. 16.

(i) Update rule for a parity-check node of degree $\mathtt{r}$: Assume that the index set for the $(\mathtt{r}-1)$ messages which enter the check node is $\mathcal{R} = [\mathtt{r}-1]$. Then

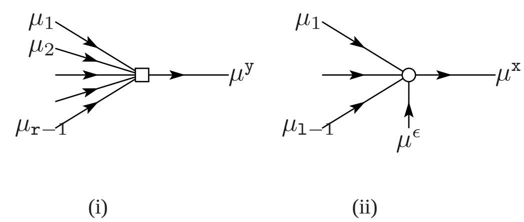


Fig. 16. Update rule for parity-check nodes (i) and variable nodes (ii).

$$\mu^{\text{y}} = \begin{cases} \mathtt{0}, & \text{if } \forall i \in \mathcal{R},\ \mu_i = \mathtt{0}, \\ *, & \text{if } \exists i \in \mathcal{R},\ \mu_i = *, \\ \mathtt{g}, & \text{if } \forall j \in \mathcal{R},\ \mu_j \neq *, \text{ and } \exists i \in \mathcal{R},\ \mu_i = \mathtt{g}. \end{cases}$$

With respect to the BP decoder, the only new rule is the one which leads to $\mu^{\text{y}} = \mathtt{g}$. It is motivated as follows. Assume that for all $i \in \mathcal{R}$ we have $\mu_i^{\text{x}} = 0\backslash\mathtt{g}$ and that at least one such message is $\mathtt{g}$. This means that the connected variables x_i, $i \in \mathcal{R}$, are either known, have been guessed themselves, or can be expressed as a linear combination of guessed bits (and at least one such value is indeed either a guess itself or expressible as a linear combination of guesses). Since the variable connected to the outgoing edge is the sum of the variables connected to the incoming edges, it follows that this variable is also expressible as a linear combination of guesses. Therefore, $\mu^{\text{y}} = \mathtt{g}$ in this case. Operationally, we have $\mathtt{r}-1$ lists $\Theta_1, \dots, \Theta_{\mathtt{r}-1}$ (at least one of which is non-empty) entering the check node. The outgoing list Θ^{y} is obtained as the *union* of the incoming lists, where indices which occur an even number of times in the incoming lists are eliminated. The list Θ^{y} provides a resolution rule for $x_1 + \dots + x_{\mathtt{r}-1}$, and therefore for the variable connected to the outgoing edge.

In the above description and the definition of the message-passing rules we have ignored the possibility that the union of the incoming lists (at least one of which is non-empty) is empty. This can happen if a complete cancellation occurs (every index appears an even number of times in the incoming lists). Fortunately, as we shall see, this assumption has no influence on the proof of Theorem 9.

(ii) Update rule for a variable node of degree $\mathtt{l}$: Assume that the index set for the $\mathtt{l}-1$ messages which enter the variable node is $\mathcal{L} = [\mathtt{l}-1] \cup \{\epsilon\}$. Then

$$\mu^{\text{x}} = \begin{cases} \mathtt{0}, & \text{if } \exists i \in \mathcal{L},\ \mu_i = \mathtt{0}, \\ *, & \text{if } \forall i \in \mathcal{L},\ \mu_i = *, \\ \mathtt{g}, & \text{if } \forall i \in \mathcal{L},\ \mu_i \neq \mathtt{0} \text{ and } \exists j \in \mathcal{L},\ \mu_j = \mathtt{g}. \end{cases}$$

Once again, it should be enough to motivate the rule which leads to $\mu^{\text{x}} = \mathtt{g}$. Recall that $\mathtt{g}$ indicates that the bit is not known but that it has either been guessed or that the bit is expressible as a linear combination of guessed bits. Therefore, if none of the incoming messages is a 0, and at least one is a $\mathtt{g}$, then the outgoing message is a $\mathtt{g}$. Operationally, this means that the outgoing list is equal to *one* of the incoming non-empty lists. E.g., if the bit itself has been guessed (i.e., $\mu_i^\epsilon =$

g) and all other incoming messages are $*$ then the outgoing message is $\{i\}$.

From the messages we can obtain estimates $\nu_i, \in [n]$, of the transmitted bits (the ν_i's are node- rather than edge-quantities). In order to obtain these estimates we apply the same rule as for the variable node update, see (ii) above, with incoming messages corresponding to *all* of the neighboring check nodes. In other words, for a degree $\mathtt{l}$ variable node, we have $\mathcal{L} = [\mathtt{l}] \cup \{\epsilon\}$ instead of $\mathcal{L} = [\mathtt{l}-1] \cup \{\epsilon\}$.

The consistency of the estimates implies a set of linear *conditions* on the guessed variables. Consider all the messages μ_i entering a fixed variable node and the associated (possibly empty) lists $\Theta_i = \{j_1^i, \cdots, j_k^i\}$. Let $\mathcal{L}_\mu$, $\mu \in \{\mathtt{0}, \mathtt{g}, *\}$ denote the subsets of indices i with $\mu_i = \mu$.

1) If $\mathcal{L}_\mathtt{0} \neq \emptyset$ and $\mathcal{L}_\mathtt{g} \neq \emptyset$, then, for any $i \in \mathcal{L}_\mathtt{g}$, we have the condition

$$x_{j_1^i} + \cdots + x_{j_k^i} = 0\,, \qquad \mod 2\,. \tag{17}$$

The total number of resulting conditions is $|\mathcal{L}_\mathtt{g}|$.

2) If $\mathcal{L}_\mathtt{0} = \emptyset$ and $|\mathcal{L}_\mathtt{g}| \geq 2$, then fix $i \in \mathcal{L}_\mathtt{g}$. For any $l \in \mathcal{L}_\mathtt{g} \backslash \{i\}$, we have the condition

$$x_{j_1^i} + \cdots + x_{j_k^i} = x_{j_1^l} + \cdots + x_{j_k^l}\,, \qquad \mod 2\,. \tag{18}$$

The total number of resulting conditions is $|\mathcal{L}_\mathtt{g}| - 1$.

The algorithm stores in memory each new condition produced during its execution. Notice that each conditions involves uniquely bits x_i for which $\mu_i^\epsilon = \mathtt{g}$. It can happen that a particular condition is either linearly dependent upon previous ones or empty. The last case occurs if the corresponding lists are empty, which in turn may be the consequence of a previous parity-check node update (see the description of the check-node update rule above). Given a set of guesses, any subset of them whose values can be chosen freely without violating any of the conditions produced by the M decoder, is said to be *independent*. Of course, the maximal number of independent guesses is equal to the number of guesses minus the number of linearly independent conditions.

Conditions are equivalent, in the present setting to what have been called contradictions in the description of of Sec. VI-A. In fact, if one thinks of guessed bits as i.i.d. uniformly random in $\{\mathtt{0}, \mathtt{1}\}$ then each new, independent condition, cf. Eqs. (17), (18) is satisfied with probability $1/2$.

It is useful to estabilish the following convention for denoting the successive message passing iterations. At the t^{th} iteration (with $t = 0, 1, \dots$) we first update all the left-to-right messages and then all the right-to-left messages. We have therefore $\cdots \to \mu^{\mathtt{y}}(t-1) \to \mu^{\mathtt{x}}(t) \to \mu^{\mathtt{y}}(t) \to \mu^{\mathtt{x}}(t+1) \to \dots$. Notice that, as the number of iterations increases, a given message can change its status according to one of the transitions $* \to \mathtt{g}$, $\mathtt{g} \to \mathtt{0}$ or $* \to \mathtt{0}$. Therefore the algorithms surely stops after a finite number of iterations (at most twice the number of edges in the graph). We shall denote the fixed point as $\mu^{\mathtt{x}}(\infty)$, $\mu^{\mathtt{y}}(\infty)$. At the t^{th} iteration the algorithm deliver an estimate $\nu_i(t)$, $i \in [n]$ of the i^{th} transmitted bit.

C. The Case of Tree Graphs and Some Simple Consequences

As for other message-passing algorithms, it is instructive to study the behavior of the M decoder on trees. In particular, we will show that: (a) On a tree the *sequential* M decoder guesses exactly as many variables as there are degrees of freedom in the system (implying that all these guesses are independent); (b) on a tree the number of *independent* guesses ventured by the (not necessarily sequential) M decoder by end of the decoding process is equal to the number of degrees of freedom of the system and it can be computed in a *local* way; (c) the same local counting formula gives in general (for Tanner graphs that are not necessarily trees) an *upper bound* on the number of independent guesses which remain at the end of the decoding process.

We have already explained that, for the purpose of analysis, we can make the all-zero codeword assumption. Therefore, in the sequel we only have to consider linear systems of equations with a zero right side. We say that the M decoder is *bit-by-bit* (or *sequential*) if any time the BP phase comes to a halt, the decoder guesses a *single* unknown bit and then proceeds by processing all consequences until no further progress is achieved.

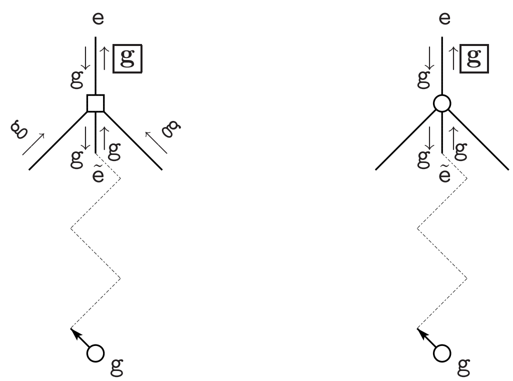
Lemma 9 (Number Of Guesses of Sequential M Decoder): Consider a binary linear system of equations with right side equal to zero and k degrees of freedom (i.e., k is equal to the number of variables minus the rank of the system). Assume that the Tanner graph associated to this system is a tree. Then the sequential M decoder ventures exactly k guesses during the decoding process and all these guesses are independent.

Proof: Without loss of generality we can assume that there are no check leaf nodes. In fact, whenever degree-one check nodes are present, the standard BP decoder can be run until all such nodes have been removed. For each variable node which is removed in this fashion, the rank of the system is decreased by exactly one as well.

We claim that the resulting system of equations has full rank. To see this, assume to the contrary that there is a non-zero linear combinations of equations that yields zero. Look at the Tanner graph corresponding to this subset of equations: all variable nodes have (even) degree at least two and all check nodes have degree at least two (as argued above). It is well known that a graph with minimum degree at least two contains at least one cycle, contradicting the hypothesis that the initial graph was a tree.

Consider therefore a Tanner graph which is a tree and all of its leaf nodes are variables. Let $\mathtt{l}_i$, $i \in [n]$, ($\mathtt{r}_i$, $i \in [m]$) denote the degree of variable (check) node i. By our remarks above, the corresponding system of equations has $n - m$ degrees of freedom. Therefore, it is clear that the M decoder has to guess *at least* $n - m$ bits before it stops. We claim that it ventures exactly $n-m$ guesses, i.e., that on a tree the sequential guesses are *independent*.

At the start of the decoding process all messages are erasures. We will show that at the end of the decoding process each edge carries exactly one $\mathtt{g}$ message in one direction and a $*$ message in the other direction. This proves our claim: it implies that a variable node which has been guessed, and hence all of its outgoing messages carry a $\mathtt{g}$ message, has no incoming g message. It is therefore not constrained by any of the other guesses, i.e., it is independent. Clearly, at the end of the decoding process each edge has to carry a g message in *at least one* direction; otherwise the connected bit has not been determined yet, contradiction the assumption that the M decoder has halted.



(a) (b)

Fig. 17. In (a) consider the messages flowing along edge e. Assume that the *outgoing* message (shown in a frame) switches as a consequence of a newly guessed bit from $*$ to g. Assume further that the *incoming* message flowing in the opposite direction is g as well. This provides the induction step from odd levels to even levels. As indicated in the figure, it then follows that both messages along edge $\tilde{\mathtt{e}}$ are g as well. The case of an edge exiting a variable node is shown in (b) and follows by essentially the same argument.

Let us show that it can not carry a g message in both directions. Initially all messages are $*$. The sequential M decoder proceeds in phases, guessing a bit and then determining all consequences of this guess during the BP phase until it gets stuck again. Let us call one such guess followed by the BP phase one *iteration*. Let us agree that during the BP phase the consequence of a newly guessed bit are computed *in order of increasing distance* from the guessed bit. This means, that we first process all edges directly connected to this bit (call this *level zero*), then all edges at distance one (call this *level one*) and so on. Assume that when we process level t, $t \geq 1$, we encounter an edge whose outgoing (away from the newly guessed bit) message switches from $*$ to g and whose incoming message already is g. We claim that then the same must have occurred at level $t-1$. This is quickly verified by checking explicitly both cases: an edge which goes from a check node to a variable node (odd levels t; left picture in Fig. 17) and the case of an edge which goes from a variable node to a check node (even levels t; left picture in Fig. 17). If we apply this argument inductively, we see that the guessed variable node must have had an incoming message which was g, contradicting the fact that the M decoder decided to guess this bit. ∎

What happens if we run the M decoder in a non-sequential way, i.e., if we guess many/several bits each time we get stuck? In this case it can happen that some of the guesses are dependent. Nevertheless, the number of independent guesses remaining at the end of the process is still equal to the degrees of freedom of the system of equations. More importantly, on a tree this number of independent guesses can be computed in a *local* way.

Lemma 10 (Number of Independent Guesses): Consider a binary linear system of equations with right side equal to zero and k degrees of freedom (i.e., k is equal to the number of variables minus the rank of the system). Assume that the Tanner graph associated to this system is a tree and that it contains no check nodes of degree one. Then the number of *independent* guesses ventured by the M decoder at the end of the decoding process is equal to k. Further, let $\mathbb{G}$ denote the total number of guesses of the M decoder, denote by $\mathtt{l}_i^{\mathtt{g}}$ the number of incoming g messages at variable node i (including, if applicable, the guess of the bit itself), and by $\mathcal{C}_{\mathtt{g}}$ the subset of all check nodes all of its incoming messages are g. Then

$$k = \mathbb{G} - \sum_{i \in \mathcal{V}} (\mathtt{l}_i^{\mathtt{g}} - 1) + \sum_{i \in \mathcal{C}_{\mathtt{g}}} (\mathtt{r}_i - 1). \tag{19}$$

Proof: By definition of the algorithm, at the end of the decoding process all bits have been determined (i.e., guessed or expressed in terms of guessed bits). This means that among the guesses ventured by the M decoder there must be k independent such guesses. Now note that the final state of the messages is independent of the order in which the guesses are taken. It is convenient to imagine that we first venture the k independent guesses and then apply the BP decoder. At the end of this phase all bits are known. Further, from Lemma 9 we know that $\mathtt{l}_i^{\mathtt{g}} = 1$ for all $i \in [n]$ and $\mathcal{C}_g$ is the empty set. Therefore, the stated counting formula is correct at this stage. Assume now we proceed in iterations, adding one guess at a time and propagating all its consequences. We will verify that the counting formula stays valid. Assume therefore that the counting formula is correct at the start of an iteration and add a further guess, lets say of variable i. This extra guess increases $\mathtt{l}_i^{\mathtt{g}}$ by one and increases the number of guesses by one, keeping the counting formula intact. Consider now the ensuing BP phase. Consider an edge e emanating from a variable node i, the check node connected to it, call it j and all the edges and variable nodes connected to this check node. Assume that the message from i to j is $*$ (in the case that this message is already g, the message does not change and there is nothing to prove). As a consequence the message from j to i must be a g because of the argument above. Also, all the incoming messages into j but the one form i must be g as well (otherwise the update rule would have been violated at node j). Update all the corresponding edge messages. If the message from i to j does not change, then neither does any of the messages outgoing at the check node and the counting formula stays valid. If, on the other hand, the outgoing message along edge e flips to g then so do all the messages outgoing from the check node j. Assume that the check node has degree $\mathtt{r}_j$. Then, $\mathcal{C}_g$ now contains j. This increases the right hand side of the counting formula by $\mathtt{r}_j - 1$. On the other hand it also increases $\mathtt{l}_l^{\mathtt{g}}$ by one for all $l \in \mathcal{V}$ which are connected to check node j, but for node i (the corresponding message was already a g). In total this decreases the right hand side of the counting formula by $\mathtt{r}_j - 1$. ∎

Each part of the counting equation (19) has a pleasing interpretation. As stated, $\mathbb{G}$ is the total number of ventured guesses. If a variable node has $\mathtt{l}^{\mathtt{g}}$ incoming g messages then these correspond to $\mathtt{l}^{\mathtt{g}}$ linear equations, each of which determines the same bit. This gives rise to $(\mathtt{l}^{\mathtt{g}} - 1)$ linear conditions which the $\mathbb{G}$ guesses have to fulfill. But not all these conditions are linearly independent. Consider Fig. 18. If a check node of degree $\mathtt{r}$ has all of its incoming messages equal to g then the $\mathtt{r}$ equations which correspond to the $\mathtt{r}$ outgoing messages are identical, i.e., $\mathtt{r}-1$ of them are linearly dependent. The last term in the counting formula (19) therefore corrects the over-counting of dependent conditions.

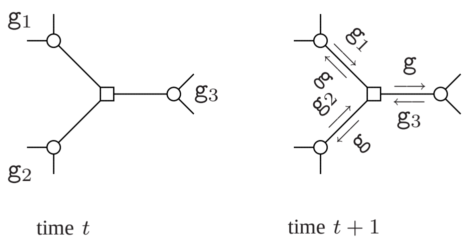


Fig. 18. Computation of the number of linearly independent conditions. To each of the incoming edges corresponds a list. To keep things simple and without essential loss of generality, assume that $\Theta_i = \{i\}$. The three outgoing lists are then $\Theta_1 = \{2,3\}$, $\Theta_2 = \{1,3\}$, and $\Theta_3 = \{1,2\}$. Compare the incoming and outgoing list at node 1: we get the condition $x_1 = x_2 + x_3$. But exactly the same condition appears at node 2 and node 3. In general, a check node of degree $\mathtt{r}$, all of its incoming messages are g, generates $\mathtt{r}-1$ linearly dependent conditions.

Example 8: Consider a code whose Tanner graph is a tree and all leaves are variable nodes. Let the set of variables (checks) be indexed by $[n]$ ($[m]$, and let $\mathtt{l}_i$, $i \in [n]$, ($\mathtt{r}_i$, $i \in [m]$) be the degree of variable (check) node i. Assume that the M decoder guesses all leaf (variable) nodes and then proceeds by message passing. It is not very hard to see that in this setting the decoder proceeds with the message-passing phase (starting from the leaf nodes) until all variables have been determined and that no further guesses have to be made. Further, at the end of the decoding process *all* messages are g.

Let us determine the number of independent guesses at the end of the decoding process using the counting formula (20). Note that for each leaf node we have $\mathtt{l}^{\mathtt{g}} = 2$ (one guess and one additional incoming g message. For all internal variable nodes we have $\mathtt{l}^{\mathtt{g}} = \mathtt{l}$. Finally, $\mathcal{C}_{\mathtt{g}} = \mathcal{C}$. If we let n_l denote the number leaf nodes, so that $\mathbb{G} = n_l$, we get that the number of independent guesses is equal to

$$n_l - \sum_{i \in \text{leaves}} (2-1) - \sum_{i \in [n] \setminus \text{leaves}} (\mathtt{l}_i - 1) + \sum_{i \in [m]} (\mathtt{r}_i - 1)$$
$$= - \sum_{i \in [n]} (\mathtt{l}_i - 1) + \sum_{i \in [m]} (\mathtt{r}_i - 1) = n - m.$$

This is of course the expected result since the system has exactly $n-m$ degrees of freedom.

So far we have only considered sets of equations whose Tanner graph is a tree. What happens if we run the M decoder on a general system of equations. For a general Tanner graph, the above counting of the total number of independent guesses is not necessarily tight. The counting of the total number of conditions generated by the M decoder is always correct. But it can happen that besides the obvious over-counting at check nodes, there are other dependencies generated by loops in the graph which are not considered in the counting formula. Therefore, in general we only get a lower bound. Let us state this explicitly.

Lemma 11 (Lower Bound on Independent Guesses): Consider a binary linear system of equations with right side equal to zero and k degrees of freedom (i.e., k is equal to the number of variables minus the rank of the system). Assume that the Tanner graph associated to this system contains no check nodes of degree one. Let $\mathbb{G}$ denote the number of all guesses of the M decoder, denote by $\mathtt{l}_i^{\mathtt{g}}$ the number of incoming g messages at variable node i (including the guess if this node has been guessed), and by $\mathcal{C}_{\mathtt{g}}$ the subset of all check nodes all of whose incoming messages are g. Then

$$k \geq \mathbb{G} - \sum_{i \in \mathcal{V}} (\mathtt{l}_i^{\mathtt{g}} - 1) + \sum_{i \in \mathcal{C}_{\mathtt{g}}} (\mathtt{r}_i - 1). \tag{20}$$

D. Density Evolution Analysis

Let us now perform the usual DE analysis. Let $\mathtt{x}^t_{\mu^{\mathtt{x}}}$ denote the probability that a left-to-right message at time t is equal to $\mu^{\mathtt{x}} \in \{0, *, \mathtt{g}\}$, and let $\mathtt{y}^t_{\mu^{\mathtt{y}}}$ denote the corresponding probability for a right-to-left message.

(i) At the check node side the DE relations read

$$\mathtt{y}_0^t = \rho(\mathtt{x}_0^t),$$
$$\mathtt{y}_*^t = 1 - \rho(\mathtt{x}_0^t + \mathtt{x}_{\mathtt{g}}^t) = 1 - \rho(1 - \mathtt{x}_*^t),$$
$$\mathtt{y}_{\mathtt{g}}^t = 1 - \mathtt{y}_0^t - \mathtt{y}_*^t = \rho(\mathtt{x}_0^t + \mathtt{x}_{\mathtt{g}}^t) - \rho(\mathtt{x}_0^t).$$

(ii) At the variable node side the DE relations are

$$\mathtt{x}_0^{t+1} = 1 - \epsilon\lambda(\mathtt{y}_{\mathtt{g}}^t + \mathtt{y}_*^t),$$
$$\mathtt{x}_*^{t+1} = (1-\gamma)\epsilon\lambda(\mathtt{y}_*^t),$$
$$\mathtt{x}_{\mathtt{g}}^{t+1} = \epsilon\lambda(\mathtt{y}_{\mathtt{g}}^t + \mathtt{y}_*^t) - (1-\gamma)\epsilon\lambda(\mathtt{y}_*^t).$$

According to our convention, the iteration counter is increased only in the variable node operation. Moreover, the variables $\mathtt{x}_*^t$ ($\mathtt{y}_*^t$) and $\mathtt{x}_*^t + \mathtt{x}_{\mathtt{g}}^t$ ($\mathtt{y}_*^t + \mathtt{y}_{\mathtt{g}}^t$) satisfy the same equations as the fractions of erased messages in the standard BP decoder with erasure probabilities $\epsilon(1-\gamma)$ and γ, respectively. This is an immediate consequence of the update rules defined in section VI-B.

When the time t tends to ∞, DE converges to the fixed-point probability distribution. To settle our notation, we write $(\mathtt{x}_0^t, \mathtt{x}_*^t, \mathtt{x}_{\mathtt{g}}^t) \underset{t\to\infty}{\longrightarrow} \big(\mathtt{x}_0^\infty(\epsilon,\gamma), \mathtt{x}_*^\infty(\epsilon,\gamma), \mathtt{x}_{\mathtt{g}}^\infty(\epsilon,\gamma)\big)$ and equivalently $(\mathtt{y}_0^t, \mathtt{y}_*^t, \mathtt{y}_{\mathtt{g}}^t) \underset{t\to\infty}{\longrightarrow} \big(\mathtt{y}_0^\infty(\epsilon,\gamma), \mathtt{y}_*^\infty(\epsilon,\gamma), \mathtt{y}_{\mathtt{g}}^\infty(\epsilon,\gamma)\big)$. Observe that $\mathtt{x}_*^\infty(\epsilon,\gamma)$ satisfies the equation $\mathtt{x} = \epsilon(1-\gamma)\lambda(1 - \rho(1-\mathtt{x}))$, while $\mathtt{x}_0^\infty(\epsilon,\gamma) = \mathtt{x}_0^\infty(\epsilon)$ satisfies the equation $(1-\mathtt{x}) = \epsilon\lambda(1-\rho(1-(1-\mathtt{x})))$.

Notice that the asymptotic state of the algorithm has the following structure. The variable nodes such that $\nu_i(\infty) = *$ *or* $\nu_i(\infty) = \mathtt{g}$, form a stopping set: in fact this is the largest stopping set contained in the set of variable nodes for which $\mu_i^\epsilon = *$ *or* $\mu_i^\epsilon = \mathtt{g}$. Further, the set of variable nodes such that $\nu_i(\infty) = *$ form a stopping set contained in the previous one: this is the largest stopping set contained in the set $\mu_i^\epsilon = *$.

In the analysis below we shall repeatedly use the following trick. We shall compute expectations with respect to asymptotic ($t = \infty$) incoming messages in a given node. In such computations, we shall treat such messages as i.i.d.

with distribution $\left(\mathtt{x}_0^\infty, \mathtt{x}_*^\infty, \mathtt{x}_\mathtt{g}^\infty\right)$, (for left-to-right messages) or $\left(\mathtt{y}_0^\infty, \mathtt{y}_*^\infty, \mathtt{y}_\mathtt{g}^\infty\right)$, (for right-to-left messages). As long as (ϵ, γ) take non-exceptional values, i.e., at continuity points of $\left(\mathtt{x}_0^\infty(\epsilon,\gamma), \mathtt{x}_*^\infty(\epsilon,\gamma), \mathtt{x}_\mathtt{g}^\infty(\epsilon,\gamma)\right)$, cf. Section V, this is justified as follows. First consider messages after a finite number of iterations t. For n large enough these are independent because the Tanner graph is locally a tree. But, if (ϵ, γ) is non-exceptional the number of message which change between the t^{th} iteration and the asymptotic state is bounded by $n\delta(t)$ with $\delta(t) \to 0$ as $t \to \infty$. This argument is essentially the same as the one of App. II-A.

E. Guessing Strategy

In the analysis of the M decoder, we can chose the order of guesses at our convenience. As long as the message is completely decoded and the final estimates are $\nu_i(\infty) \in \{\mathtt{0}, \mathtt{g}\}$ for any bit i, the algorithm realizes a complete list decoding.

We shall adopt the following strategy: we perform n_{rounds} "decoding rounds". Our progress will be measured by the parameter γ, which is initially set to zero and which advances by $\Delta\gamma = 1/n_{\text{rounds}}$ in each round.

Set $\gamma = 0$. Start with the messages received via BEC(ϵ) and apply BP decoding until the algorithm gets stuck. Then consider each of the bits not yet determined and set $\mu_i^\epsilon = \mathtt{g}$ independently for each of them with probability $\Delta\gamma/(1-\gamma)$. (In the first round this probability is equal to $\Delta\gamma$.) Set $\gamma \stackrel{\Delta}{=} \gamma + \Delta\gamma$.[8] Apply the M decoder until it gets stuck. This is repeated n_{rounds} times until $\gamma = 1$. If at any earlier phase complete decoding is achieved, the algorithm is halted and the current set of decoded codewords output.

The analysis becomes simpler (and the algorithm more efficient) if we take $\Delta\gamma \to 0$. We shall always think of this limit being taken after $n \to \infty$. We will see that in this limit the appearance of contradictions is sharply concentrated to those rounds which include a discontinuity of the EXIT curve. In other words, we will see that the algorithm alternates between the following two phases which are well separated: in the "guessing phase" the algorithm guesses a small fraction of bits and the processes the consequences but theses consequences do not propagate too far and essentially stay local; in the "contradiction phase" on the other hand the algorithm suddenly discovers many relationships (finds many contradictions) and the size of the residual graph changes by a constant fraction which is independent of the step size $\Delta\gamma$.

F. Analysis: Guess Work

Consider a non-exceptional point (ϵ, γ) and let $n\Delta\mathbb{G}$ be the number of newly guessed variables when γ is changed by an amount $\Delta\gamma > 0$.

The process can be described as follows. For each $i \in [n]$, i is selected independently with probability $\Delta\gamma/(1-\gamma)$. For each selected bit, we consider the present estimate provided by the M decoder: $\nu_i(\infty) \in \{\mathtt{0}, \mathtt{g}, *\}$. If $\nu_i(\infty) = *$, the observation on i is changed from $\mu_i^\epsilon = *$ to $\mu_i^\epsilon = \mathtt{g}$: the counter of newly guessed variables is increased by one. By linearity of expectation, we get

$$\begin{aligned}\mathbb{E}[\Delta\mathbb{G}] &= \frac{1}{n}\sum_{i\in[n]} \Pr(i \text{ is selected}) \Pr(\nu_i(\infty) = *) \\ &= \frac{\Delta\gamma}{1-\gamma}\,\epsilon(1-\gamma)\Lambda(\mathtt{y}_*^\infty) = \epsilon\Lambda(\mathtt{y}_*^\infty)\Delta\gamma\,.\end{aligned}$$

Notice that, in this computation we assumed $n \to \infty$ and $t \to \infty$ afterwards.

Recall that, after γ is changed to $\gamma + \Delta\gamma$ and the $n\Delta\mathbb{G}$ new guesses are introduced, the message passing M decoder is started again until a new fixed point is reached.

G. Analysis: Confirmation Work

At each step of the above algorithm, it may happen that several $\mathtt{g}$ messages are transmitted to the same variable node x_i. Each of these lists corresponds to a distinct resolution rule for x_i. Their convergence on the same node imposes some non-trivial condition on the variables which appear in the resolution rules. Here we estimate the number of independent such conditions by exploiting Lemma 11 above. Notice that in Lemma 11 we assume $\mu_i^\epsilon \in \{\mathtt{g}, *\}$. In order to make contact with this assumption we could first run the classical BP decoder until no further progress can be made. We could now directly apply Lemma 11 to the *residual* graph. The disadvantage of this strategy is that in this scheme it is not so straightforward to relate the progress of the M decoder on the residual graph to the original DE equations.

Alternatively we can apply Lemma 11 directly to the original graph if (i) we do not count contradictions generated at variable nodes which receive at least one 0 message (either from the channel or from the graph) and (ii) we count towards the degree of a check node only those edges whose incoming messages are not 0. With these two conventions one can check that Lemma 11 holds for a general graph including degree-one check nodes as well as variable nodes which are known.

Let (ϵ, γ) be a non-exceptional point and denote by $n\mathbb{C}$ the number of contradictions as estimated by the right-hand side of (20). The first term counts the number of conditions arising at that node. We get

$$\begin{aligned}&\mathbb{E}\left\{\frac{1}{n}\sum_{i\in\mathcal{V}} \max(|\mathcal{L}_{i,\mathtt{g}}|-1, 0)\right\} = \\ &\qquad \epsilon(1-\gamma)\sum_{\mathtt{l}} \Lambda_{\mathtt{l}}\mathbb{E}_{\mathtt{l}}\left\{\max(n_{\mathtt{g}}-1,0)\,\mathbb{I}_{n_0=0}\right\} \\ &\qquad\qquad + \epsilon\gamma\sum_{\mathtt{l}} \Lambda_{\mathtt{l}}\mathbb{E}_{\mathtt{l}}\left\{\max(n_{\mathtt{g}},0)\,\mathbb{I}_{n_0=0}\right\}\,,\end{aligned}$$

where $\mathbb{I}_A$ is the indicator function for the event A and where $n_\mathtt{g}$, $n_\mathtt{0}$, and n_* count the number of incoming $\mathtt{g}$, $\mathtt{0}$, and ? messages. Here the limits $n \to \infty$ and $t \to \infty$ are understood and $\mathbb{E}_{\mathtt{l}}$ denotes expectation with respect the multinomial variables $n_\mathtt{0}, n_\mathtt{g}, n_*$ with sum $\mathtt{l}$ and parameters $\mathtt{y}_0^\infty, \mathtt{y}^\infty, \mathtt{y}_*^\infty$. Note that we have the indicator function $\mathbb{I}_{n_0=0}$ since by our remarks above we should only consider nodes "in the residual graph", i.e., nodes which were not already

[8] Note that if a bit is first selected with probability γ and then independently selected with probability $\Delta\gamma/(1-\gamma)$, then the probability that it was selected at least once is equal to $\gamma + \Delta\gamma$. This is the rational for our choice of parameters.

determined in the BP phase as a consequence of the received bits. Throughout this section we shall adopt the shorthands $\mathtt{y}_0, \mathtt{y}, \mathtt{y}_*$ for $\mathtt{y}_0^\infty, \mathtt{y}_\mathtt{g}^\infty, \mathtt{y}_*^\infty$ (and analogous ones for left-to-right messages). By computing these expectations we get

$$\mathbb{E}\left\{\frac{1}{n}\sum_{i\in\mathcal{V}}\max(|\mathcal{L}_{i,\mathtt{g}}|-1,0)\right\} = \epsilon(1-\gamma)\left\{\Lambda'(\mathtt{y}_*+\mathtt{y}_\mathtt{g})\mathtt{y}_\mathtt{g} - \Lambda(\mathtt{y}_*+\mathtt{y}_\mathtt{g}) + \Lambda(\mathtt{y}_*)\right\} + \epsilon\gamma\Lambda'(\mathtt{y}_*+\mathtt{y}_\mathtt{g})\mathtt{y}_\mathtt{g}\,. \quad (21)$$

We must now evaluate the correction term in (20). Consider a check node a. Assume that its "residual" degree is $\mathtt{r}'_a$. I.e., $\mathtt{r}'_a$ counts the number of edges whose incoming messages are not zero. If the corresponding $\mathtt{r}'_a$ outgoing messages are all $\mathtt{g}$ (equivalently, the $\mathtt{r}'_a$ ingoing messages are all $\mathtt{g}$), then the same condition has been overcounted $\mathtt{r}'_a - 1$ times. We denote the set of such check nodes as $\mathcal{C}$ and obtain

$$\mathbb{E}\left\{\frac{1}{n}\sum_{a\in\mathcal{C}}(\mathtt{r}'_a-1)\right\} = \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{\mathtt{r}}\Gamma_\mathtt{r}\,\mathbb{E}_\mathtt{r}\left\{\max(n_\mathtt{g}-1,0)\,\mathbb{I}_{n_*=0}\right\}\,,$$

where $\mathbb{E}_\mathtt{r}$ denotes expectation with respect the multinomial variables $n_0, n_\mathtt{g}, n_*$ with sum $\mathtt{r}$ and parameters $\mathtt{x}_0^\infty, \mathtt{x}^\infty, \mathtt{x}_*^\infty$. Once again, it is quite easy to compute the above expectations. One obtains

$$\mathbb{E}\left\{\frac{1}{n}\sum_{a\in\mathcal{C}}(\mathtt{r}'_a-1)\right\} = \frac{\Lambda'(1)}{\Gamma'(1)}\{\Gamma'(1-\mathtt{x}_*)\mathtt{x}_\mathtt{g} - \Gamma(1-\mathtt{x}_*) + \Gamma(1-\mathtt{x}_*-\mathtt{x}_\mathtt{g})\}\,. \quad (22)$$

By taking the difference of Eqs. (21) and (22), and after a few algebraic manipulations, we finally get the desired result

$$\mathbb{E}[\mathbb{C}] = F(\mathtt{x},\epsilon,\gamma)\,,$$

where

$$F(\mathtt{x},\epsilon,\gamma) \stackrel{\triangle}{=} \Lambda'(1)[\mathtt{x}_*(1-\mathtt{y}_*) - (\mathtt{x}_*+\mathtt{x}_\mathtt{g})(1-\mathtt{y}_*-\mathtt{y}_\mathtt{g})] - \epsilon(1-\gamma)[\Lambda(\mathtt{y}_*+\mathtt{y}_\mathtt{g}) - \Lambda(\mathtt{y}_*)] + \frac{\Lambda'(1)}{\Gamma'(1)}[\Gamma(1-\mathtt{x}_*) - \Gamma(1-\mathtt{x}_*-\mathtt{x}_\mathtt{g})]\,.$$

Here we used the shorthand $\mathtt{x}$ for the vector $(\mathtt{x}_*, \mathtt{x}_\mathtt{g}, \mathtt{x}_0, \mathtt{y}_*, \mathtt{y}_\mathtt{g}, \mathtt{y}_0)$.

Imagine now changing $\gamma \to \gamma + \Delta\gamma$ and computing the number of new conditions on the newly guessed variables (whose expected number was computed in the previous section). Call $\Delta\mathbb{C}$ the upper bound on their number provided by Lemma 11. It is clear that, repeating the above derivation, we get

$$\mathbb{E}[\Delta\mathbb{C}] = F(\mathtt{x}^\infty(\epsilon,\gamma+\Delta\gamma),\epsilon,\gamma+\Delta\gamma) - F(\mathtt{x}^\infty(\epsilon,\gamma),\epsilon,\gamma+\Delta\gamma)\,,$$

Consider now two separate possibilities. In the first case $\mathtt{x}^\infty(\epsilon,\gamma')$ is continuous (and therefore analytic) in the interval $\gamma' \in [\gamma, \gamma+\Delta\gamma]$. By Taylor expansion we get

$$\mathbb{E}[\Delta\mathbb{C}] = -\frac{\partial F}{\partial \mathtt{x}}(\mathtt{x}',\epsilon,\gamma+\Delta\gamma)\cdot\frac{\partial \mathtt{x}^\infty(\epsilon,\gamma)}{\partial\gamma}\,\Delta\gamma + O((\Delta\gamma)^2)\,.$$

with the gradient of F being evaluated at $\mathtt{x}' = \mathtt{x}^\infty(\epsilon,\gamma+\Delta\gamma)$. A direct calculation shows that the gradient vanishes at this point leading to $\mathbb{E}[\Delta\mathbb{C}] = O((\Delta\gamma)^2)$.

In the second case, the interval $[\gamma, \gamma+\Delta\gamma]$ includes a discontinuity point (a jump) γ_j. Let $\mathtt{x}_{\mathrm{j}+} \stackrel{\triangle}{=} \underline{\mathtt{x}}^{j+1} = \lim_{\gamma\downarrow\gamma_\mathrm{j}} \mathtt{x}^\infty(\epsilon,\gamma)$ and $\mathtt{x}_{\mathrm{j}-} \stackrel{\triangle}{=} \overline{\mathtt{x}}^j = \lim_{\gamma\uparrow\gamma_\mathrm{j}} \mathtt{x}^\infty(\epsilon,\gamma)$. We have

$$\mathbb{E}[\Delta\mathbb{C}] = F(\mathtt{x}_{\mathrm{j}+},\epsilon,\gamma_\mathrm{j}) - F(\mathtt{x}_{\mathrm{j}-},\epsilon,\gamma_\mathrm{j}) + O(\Delta\gamma)\,.$$

H. Finishing the proof

Consider now the guessing strategy explained in Section VI-E. First the received message is decoded with the usual iterative decoder. At this point $\gamma = 0$. Then each bit is selected independently with $\Delta\gamma/(1-\gamma)$ and guessed if its valued was not determined (eventually in terms of former guesses) at previous stages. The M decoder is then run until a fixed point is reached. The number of new guesses at this stage is $\Delta\mathbb{G}_\gamma$ and the number of new conditions is upper bounded by $\Delta\mathbb{C}_\gamma$. This operation is repeated until $\nu_i(\infty) \in \{0, \mathtt{g}\}$ for each i. Without loss of generality, we may imagine this to happen at $\gamma = 1$.

At this point each realization of the guesses compatible with the conditions yields a codeword compatible with the received message. We have

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}_\mathtt{G}[H_\mathtt{G}(X|Y)] \ge \sum_\gamma \mathbb{E}[\Delta\mathbb{G}_\gamma] - \sum_\gamma \mathbb{E}[\Delta\mathbb{C}_\gamma] = \int_0^1 \epsilon\Lambda(\mathtt{y}_*(\gamma,\epsilon))\,\mathrm{d}\gamma - \sum_{\gamma_\mathrm{j}}\Delta F_\mathrm{j} + O(\Delta\gamma)\,,$$

where the last sums runs over the jump positions γ_j and $\Delta F_\mathrm{j} \le F(\mathtt{x}_{\mathrm{j}+},\epsilon,\gamma_\mathrm{j}) - F(\mathtt{x}_{\mathrm{j}-},\epsilon,\gamma_\mathrm{j})$ is the discontinuity of F at those positions. In order to finish the proof of Lemma 9, notice that $H(X|Y)$ does not depend upon $\Delta\gamma$ and we can therefore take the limit $\Delta\gamma \to 0$ discarding $O(\Delta\gamma)$ terms. Moreover $\mathtt{y}_*(\gamma,\epsilon) = \mathtt{y}(\epsilon(1-\gamma))$ (the last quantity being the fixed point of DE for the usual BP decoder at erasure probability ϵ), and therefore

$$\int_0^1 \epsilon\Lambda(\mathtt{y}_*(\gamma,\epsilon))\,\mathrm{d}\gamma = \int_0^\epsilon \Lambda(\mathtt{y}(\epsilon'))\,\mathrm{d}\epsilon'$$

is just the area under the BP EXIT curve (dark gray in Fig. 1, (a)). Finally, let $\epsilon_\mathrm{j} = (1-\gamma_\mathrm{j})\epsilon$ and $(\mathtt{x}(\epsilon_\mathrm{j}+), \mathtt{y}(\epsilon_\mathrm{j}+))$ and $(\mathtt{x}(\epsilon_\mathrm{j}-), \mathtt{y}(\epsilon_\mathrm{j}-))$ be the fixed point of DE for the usual iterative decoder just above and below the jump. Then

$$\Delta F_\mathrm{j} = P_{\epsilon_\mathrm{j}}(\mathtt{x}(\epsilon_\mathrm{j}-), \mathtt{y}(\epsilon_\mathrm{j}-) - P_{\epsilon_\mathrm{j}}(\mathtt{x}(\epsilon_\mathrm{j}+), \mathtt{y}(\epsilon_\mathrm{j}+))\,,$$

where $P_\epsilon(\mathtt{x},\mathtt{y})$ is the trial entropy, cf. Def. 4. Because of Lemma 4, ΔF_j is just the area delimited by the EBP EXIT curve and a vertical line through the jump, (dark gray in Fig. 1, (b)).

I. Maxwell Decoder: Illustration and Implementation

The Maxwell decoder provides an *interpretation* for the balance of areas which we described in Sections IV and V. For many ensembles, e.g., the $(3,6)$-regular ensemble, Theorem 10 gives a complete characterization of the MAP

EXIT function and therefore a complete justification of the Maxwell construction. In some other cases we are not quite as lucky, see e.g. the ensemble discussed in Example 7, and we can only conjecture that the parts of the MAP EXIT function which are not covered by Theorem 10 also follow the Maxwell construction. Let us now review some typical case.

Example 9 ($(3,6)$ LDPC ensemble): Consider the dd pair $(\lambda,\rho) = (x^2, x^5)$ and the corresponding LDPC ensemble with design rate one-half. Its BP and MAP EXIT functions are depicted in Fig. 1 together with the balance conditions. Fig. 19 shows the evolution of the entropy $\hat{H}(t)$, i.e., the logarithm of the number of running copies as discussed in Fig. 15, as a function of the fraction of bits determined by the decoding process for the $(3,6)$-regular LDPC ensemble. Transmission takes place over BEC($\epsilon = 0.46$), i.e., we fix the channel parameter ϵ so that $\epsilon^{\text{BP}} \approx 0.4294 < \epsilon < \epsilon^{\text{MAP}} \approx 0.4882$. After transmission, a fraction $1-\epsilon = 0.54$ of bits is known. The classical BP algorithm proceeds until it gets stuck at the fixed point $(\mathtt{x}^{\epsilon} \approx 0.3789, \mathtt{y}^{\epsilon} \approx 0.9076)$ of DE. At this point (point A in the figure), a fraction $1-\epsilon\Lambda(\mathtt{y}^{\epsilon}) \approx 0.6561$ of bits has been determined. Now the guessing phase of the M decoder starts. It ends at point B, which corresponds to the BP threshold $(\mathtt{x}^{\text{BP}} \approx 0.2606, \mathtt{y}^{\text{BP}} \approx 0.7790)$. The total fraction of guesses that the M decoder has to venture is $\int_{\mathtt{x}^{\text{BP}}}^{\mathtt{x}^{\epsilon}} h(\epsilon(\mathtt{x}))\mathrm{d}\epsilon(\mathtt{x}) = P(\mathtt{x}^{\epsilon},\mathtt{y}^{\epsilon}) - P(\mathtt{x}^{\text{BP}},\mathtt{y}^{\text{BP}})$. For our specific example we have $P(\mathtt{x},\mathtt{y}(\mathtt{x})) = -\frac{5\mathtt{x}^2}{2} + 10\mathtt{x}^3 - \frac{25\mathtt{x}^4}{2} + 7\mathtt{x}^5 - \frac{3\mathtt{x}^6}{2}$, so that the total fraction of guesses is equal to 0.0201509. For a blocklength of $n = 34000$ this corresponds to roughly 685 guesses. At this point the BP decoding phase resumes. More and more guesses are confirmed. Since we are operating below the MAP threshold, (essentially) all guesses are eventually confirmed and the M decoder comes to a halt.

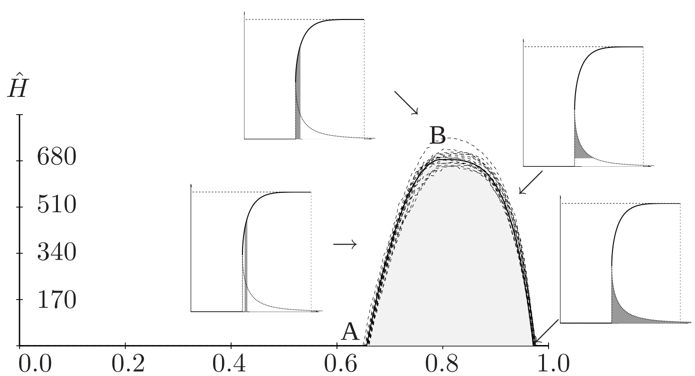


Fig. 19. M decoder applied to the $(3,6)$-regular LDPC ensemble. Asymptotic entropy of the M decoder $\hat{H}$ (logarithm of the number of running copies) as a function of the fraction of determined bits. 15 channel and code realizations with $\epsilon = 0.46$ and blocklength $n = 34\cdot 10^3$ are shown (dashed curves) together with the analytic asymptotic curve (solid curve). The inserts show how the entropy curve can be constructed from the EXIT curve. The fraction of guesses is shown in the 2 left-most inserts while the fraction of contradictions is shown in the 2 right inserts.

Example 10 (Typical Double "Jump"): Consider the dd pair $(\lambda,\rho) = (\frac{3x+3x^2+4x^{13}}{10}, x^6)$ and the corresponding LDPC ensemble with design rate $r = \frac{19}{39} \approx 0.4872$. Its BP EXIT function is depicted in Fig. 5, its EBP EXIT curve together with the balance conditions is shown in Fig. 3.

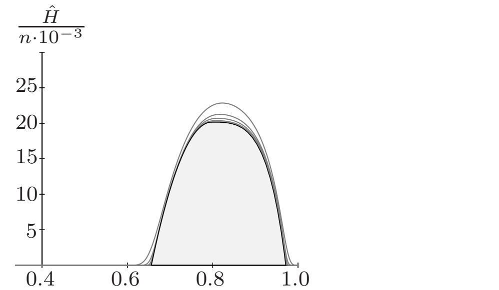


Fig. 20. M decoder applied to the $(3,6)$ LDPC ensemble: Expected symptotic entropy as a function of the fraction of determined bits at $\epsilon = 0.46$ (solid curve) and empirical average entropy curves (gray curves). Simulations are shown for $n = 780$ (average over $6\cdot 10^4$ realizations), $n = 3125$ (average over $16\cdot 10^3$ realizations), $n = 12500$ (average over $4\cdot 10^3$ realizations), $n = 50000$ (average over 10^3 realizations), $n = 200000$ (average over 150 realizations).

Finally, in Example 7 we have discussed how large parts of the MAP EXIT curve can be constructed based on Theorem 10. The MAP threshold is $\epsilon^{\text{MAP}} \approx 0.4913$ (at $\mathtt{x}^{\text{MAP}} \approx 0.1434$). According to the Maxwell costruction, the second MAP discontinuities occurs at $\epsilon^{\text{MAP},2} \approx 0.5186$ (at $\underline{\mathtt{x}}^{\text{MAP},2} \approx 0.2378$, $\overline{\mathtt{x}}^{\text{MAP},2} \approx 0.4121$) .

Fig. 21 shows the evolution of the entropy $\hat{H}(t)$ for $\epsilon = 0.5313$. This corresponds to the point C in Fig. 7, the first point at which the counting argument no longer applies. By comparing the result of the simulations to the analytic curve, corresponding to the Maxwell construction we can see that at least emperically the Maxwell construction seems to be valid over the whole range.

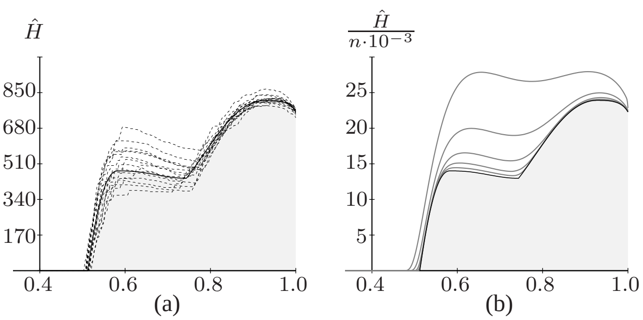


Fig. 21. M decoder applied to the irregular "double-jump" LDPC ensemble shown in Fig. 5: Asymptotic entropy as a function of the fraction of determined bits at $\epsilon = 0.5313$ (point B). (a) 15 channel and code realizations of blocklength $n = 34000$ are shown (dashed curves) together with the analytic asymptotic curve (solid curve). (b) Convergence of the average entropy curves (gray curves) to the analytic expected curve (solid curve). Simulations are shown for $n = 780$ (average over $6\cdot 10^4$ realizations), $n = 3120$ (average over $16\cdot 10^3$ realizations), $n = 12480$ (average over $4\cdot 10^3$ realizations), $n = 50017$ (average over 10^3 realizations), $n = 200500$ (average over 250 realizations).

VII. Some Further Examples

A. Special Cases

Although (for sake of simplicity) we did not discuss this case in the previous sections, other curious (but frequent) examples are those when the number of discontinuities J^{BP} of the BP EXIT curves is not equal to the number of discontinuities J^{MAP} of the MAP EXIT curve. Examples 11 and 12 show two such cases.

Example 11 ($J^{MAP} < J^{BP}$): Consider the dd pair $(\lambda, \rho) = (\frac{x^{10}+x^{60}}{2}, \frac{3x^{10}+17x^{80}}{20})$ and the corresponding LDPC ensemble with rate $r = \frac{3209}{5832} \approx 0.5502$.

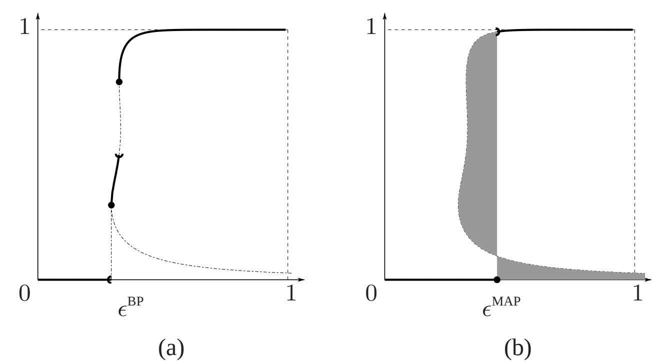


Fig. 22. When the numbers of BP and MAP "jumps" (respectively, J^{BP} and J^{MAP}) are different: (a) BP EXIT function with $J^{BP} = 2$ (b) MAP EXIT function with $J^{MAP} = 1$ and Maxwell constriction.

The MAP EXIT curve has a single "jump" at $\epsilon^{MAP} \approx 0.4493$ ($\mathtt{x}^{MAP} \approx 0.4425$) whereas the BP EXIT curve has two such singularities at $\epsilon^{BP} \approx 0.2941$ ($\mathtt{x}^{BP} \approx 0.05738$) and $\epsilon^{(BP,2)} \approx 0.3254$ ($\overline{\mathtt{x}}^{(BP,2)} \approx 0.2117$) as shown in Fig. 22. As shown in Fig. 23, Theorem 10 applies at the MAP threshold and so the whole MAP EXIT curve is determined by the counting argument in this case. The Maxwell construction is therefore confirmed in this case.

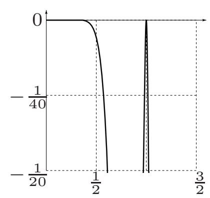


Fig. 23. Function $\Psi_{\Xi}(u)$ for the dd pair formed by the residual ensemble at $\epsilon^{MAP} = 0.4493$.

Example 12 ($J^{BP} < J^{MAP}$): Consider the dd pair $(\lambda, \rho) = (\frac{3x+3x^2+14x^{50}}{20}, x^{15})$ and the corresponding LDPC ensemble with design rate $r = \frac{311}{566} \approx 0.5495$. The BP EXIT curve has a single "jump" at $\epsilon^{BP} \approx 0.3531$ ($\mathtt{x}^{BP} \approx 0.3008$).

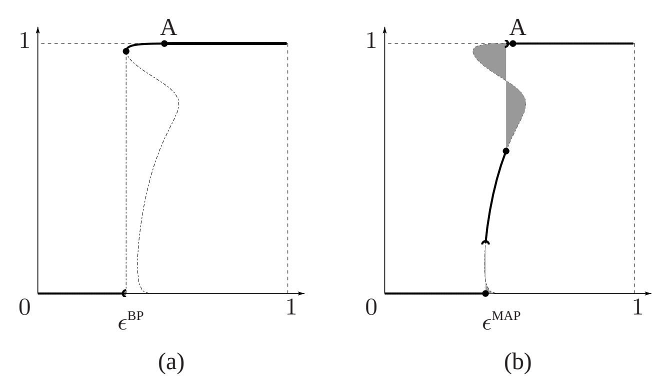


Fig. 24. When the numbers of BP and MAP "jumps" (respectively, J^{BP} and J^{MAP}) are different: (a) BP EXIT function with $J^{BP} = 1$ (b) MAP EXIT function with $J^{MAP} = 2$ and Maxwell construction.

Unfortunately, Theorem 10 shows the tightness of the M construction only up to point A (at $\epsilon \approx 0.5063$, see Fig. 24) . But it is quite natural to conjecture that the MAP EXIT curve has two singularities, namely at $\epsilon^{MAP} \approx 0.3986$ ($\mathtt{x}^{MAP} \approx 0.0340$) and at $\epsilon^{(MAP,2)} \approx 0.4855$ ($\overline{\mathtt{x}}^{(MAP,2)} \approx 0.1096$) as shown in Fig. 24. This is validated by the M decoder. Namely the M decoder gives a residual entropy (as a fraction of the blocklength) of $\frac{\hat{H}}{n} \approx 0.0121$ at $\epsilon = 0.44$. This value is exactly the value of the area (between $\epsilon = 0$ and $\epsilon = 0.44$) under the conjectured MAP EXIT curve. This shows that, between the two conjectured MAP phase transitions, the M decoder follows the part of the EBP EXIT function which is "hidden" from the BP decoder. The Maxwell construction is conjectured to hold in this case.

B. Difference Between MAP and BP Threshold

Let $r < 1$ be the design rate. Consider a sequence of degree distribution pairs $\{(\lambda(x), \rho(x)) = (x^{\mathtt{l}-1}, x^{\frac{\mathtt{l}}{1-r}-1})\}_{\mathtt{l} \geq 2}$ with fixed design rate r. Ensembles associated to this sequence are regular LDPC code ensembles. We have seen in Fact 1 that such ensembles have at most one jump and therefore we expect our bound on the MAP threshold to be tight. It was shown already in [38], that if $\mathtt{l}$ is increased then the weight distribution of such ensembles converges to the one of Shannon's random ensemble and, hence, the MAP threshold of such ensembles converges to the Shannon limit. Using the replica method, an explicit asymptotic expansion of the MAP threshold was given in [39].

Let us give here an alternative proof of this fact using our machinery. That the MAP threshold $\epsilon^{MAP}(\mathtt{l})$ converges to the Shannon threshold is shown in Fact 3. On the other hand, as stated in Fact 2, the BP threshold $\epsilon^{MAP}(\mathtt{l})$ goes to 0 when $\mathtt{l} \to \infty$. This shows that the two thresholds can be arbitrarily far apart, and nevertheless the MAP EXIT curve can be constructed from the corresponding (E)BP EXIT curve!

This is illustrated in Fig. 25 and the proofs are given in the sequel.

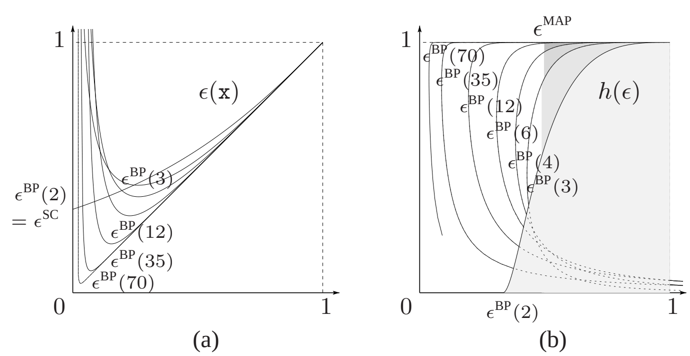


Fig. 25. Regular BP EXIT entropy curves with design rate $r = \frac{1}{2}$. (a) Channel entropy function $\mathtt{x} \mapsto \epsilon^{(1)}(\mathtt{x})$ (b) EXIT curve $h^{(1)}(\epsilon) \longleftrightarrow \epsilon^{(1)}(h)$. The depicted ensembles are, in decreasing order, the $(100, 200)$, the $(35, 70)$, the $(12, 24)$, the $(6, 12)$, the $(4, 8)$, the $(3, 6)$ and the $(2, 4)$ regular ensemble. While the BP threshold goes to 0, the bit MAP threshold goes to the Shannon limit 0.5.

Lemma 12: For a fixed non-negative $\mathtt{x} \in (0,1]$, denoting $\epsilon^{(\mathtt{l})}(\mathtt{x}) \triangleq \frac{x}{(1-(1-\mathtt{x})^{\frac{1}{1-r}-1})^{\mathtt{l}-1}}$, we get $\epsilon^{(\mathtt{l})}(\mathtt{x}) \underset{\mathtt{l}\to\infty}{\longrightarrow} \mathtt{x}$.

Proof: This limit is classically obtained with $(\mathtt{l}-1)\log[1-(1-\mathtt{x})^{\frac{1}{1-r}-1}] \sim -(\mathtt{l}-1)(1-\mathtt{x})^{\frac{1}{1-r}-1}$ which gives $(1-(1-\mathtt{x})^{\frac{1}{1-r}-1})^{\mathtt{l}-1} \underset{\mathtt{l}\to\infty}{\longrightarrow} 1^-$. ∎

Fact 2: Consider the sequence $(x^{\mathtt{l}-1}, x^{\frac{1}{1-r}-1})\}_{\mathtt{l}\geq 2}$ with fixed rate $r<1$, then the BP threshold $\epsilon^{\text{BP}}(\mathtt{l}) \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$.

Proof: Consider first the BP threshold $\epsilon^{\text{BP}}(\mathtt{l}) \triangleq \min_{\mathtt{x}}\{\epsilon^{(\mathtt{l})}(\mathtt{x})\}$. Fix $\xi > 0$ (very small). Clearly $0 \leq \epsilon^{\text{BP}}(\mathtt{l}) \leq \epsilon^{(\mathtt{l})}(\frac{\xi}{2})$, and, since $\epsilon^{(\mathtt{l})}(\frac{\xi}{2}) \underset{\mathtt{l}\to\infty}{\longrightarrow} \frac{\xi}{2}$ with Lemma 12, we can state

$$\exists \mathtt{l}_0 \in \mathbb{N}, \quad \forall \mathtt{l} \geq \mathtt{l}_0 \quad \epsilon^{(\mathtt{l})}(\frac{\xi}{2}) \leq \frac{\xi}{2} + \frac{\xi}{2}.$$

This gives that, for all $\mathtt{l} \geq \mathtt{l}_0$, the statement $0 \leq \epsilon^{\text{BP}}(\mathtt{l}) \leq \xi$ holds. This is true for any fixed ξ meaning $\epsilon^{\text{BP}}(\mathtt{l}) \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$. ∎

Instead of studying the parameterized EXIT quantity $h(\mathtt{x}) \triangleq (1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}}$, it is often more convenient to work directly with the inverse mapping $h \mapsto \mathtt{x}(h) \triangleq 1 - [1-h^{\frac{1}{\mathtt{l}}}]^{\frac{1}{\mathtt{r}-1}}$ such that we can eventually use $\epsilon(h) = \frac{1-[1-h^{\frac{1}{\mathtt{l}}}]^{\frac{1}{\mathtt{r}-1}}}{h^{\frac{\mathtt{l}-1}{\mathtt{l}}}}$ for $h \in (0,1]$.

Lemma 13: For a fixed $h \in (0,1)$, we have $\epsilon(h) = \frac{1-(1-h^{\frac{1}{\mathtt{l}}})^{\frac{r-1}{1-r+1}}}{h^{\frac{\mathtt{l}-1}{\mathtt{l}}}} \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$.

Proof: The second term of the numerator goes to 1 since, $\log(1-h^{\frac{1}{\mathtt{l}}}) = \frac{\log h}{\mathtt{l}} + \log(\frac{1}{h^{\frac{1}{\mathtt{l}}}}-1) = \frac{\log h}{\mathtt{l}} + \log(\frac{-\log h}{\mathtt{l}} + o(\frac{1}{\mathtt{l}}))$ such that $\frac{r-1}{1-1+r}[\frac{\log h}{\mathtt{l}} + \log(\frac{-\log h}{\mathtt{l}} + o(\frac{1}{\mathtt{l}}))] \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$. The lemma follows from $h^{\frac{\mathtt{l}-1}{\mathtt{l}}} \sim h > 0$. ∎

Fact 3: Consider again the sequence $(x^{\mathtt{l}-1}, x^{\frac{1}{1-r}-1})\}_{\mathtt{l}\geq 2}$ with fixed rate $r<1$, then $\epsilon^{\text{MAP}}(\mathtt{l}) \underset{\mathtt{l}\to\infty}{\longrightarrow} \epsilon^{\text{Sh}} = 1-r > 0$.

Proof: First, the inequality $0 \leq \epsilon^{\text{Sh}} - \epsilon^{\text{MAP}}(\mathtt{l})$ holds from the Area Theorem.[9] Second,

$$\epsilon^{\text{Sh}} - \epsilon^{\text{MAP}}(\mathtt{l}) = (1-r) - \epsilon^{\text{MAP}}(\mathtt{l}) = \mathcal{A}^{(\mathtt{l})} \leq \tilde{\mathcal{A}}^{(\mathtt{l})}$$

where, in short, $\mathcal{A}^{(\mathtt{l})}$ represents the closed area between $\{\epsilon(h)\}_{\epsilon^{\text{BP}}\leq\epsilon\leq 1}$, the horizontal axis $\{\epsilon = \epsilon^{\text{MAP}}\}$ and the vertical axis $\{h = 1\}$. The area $\tilde{\mathcal{A}}^{(\mathtt{l})}$ is the surface of the unit square which lies under $\{\epsilon(h)\}_{0\leq\epsilon\leq 1}$. Now, consider the function $\tilde{\epsilon}^{(\mathtt{l})}(h) = \min\{\epsilon(h)^{(\mathtt{l})}, 1\} \leq 1$. The Dominated Convergence Theorem[10] applied to the sequence $\tilde{\epsilon}^{(\mathtt{l})}$ gives that $\lim_{\mathtt{l}\to\infty} \tilde{\mathcal{A}}^{(\mathtt{l})} = 0$, which concludes the proof. ∎

C. Application to other Iterative Coding Schemes

Although LDPC ensembles have been used to present the discussed concepts, the picture is not limited to such ensembles. Equivalent statements are expected to hold in large generality.

To give just one example, consider generalized LDPC (GLDPC) ensembles: Part of our results can be directly applied like, e.g., Lemma 3. Consider a GLDPC ensemble: Equivalently to the dd pair (λ, ρ), the pair $(\lambda(\mathtt{x}), \mathtt{y}(\mathtt{x})) \triangleq (\lambda(\mathtt{x}), 1-\rho(1-\mathtt{x}))$ suffices to describe the BP decoding of the ensemble in the asymptotic limit. The left (right) component of the pair $(\lambda(\mathtt{x}), \mathtt{y}(\mathtt{x}))$ gives the EXIT entropy outgoing from the left (right) nodes during the BP decoding. To be more precise, at a fixed channel parameter ϵ, the function $\mathtt{x}(\mathtt{y}) \triangleq \epsilon\lambda(\mathtt{x})$ is the EXIT entropy outgoing from the left and $\mathtt{y}(\mathtt{x}) \triangleq \mathtt{y}(\mathtt{x}, \epsilon)$ is the EXIT entropy outgoing from the right.[11] A few calculus or computations lead, in general, to an expression for the right component EXIT entropy (see, e.g, [40], [41]).

Example 13 (GLDPC Codes): Generalized LDPC codes (see, e.g., [42]–[44]) are LDPC codes whose check nodes are replaced by some more complex linear constraints. Such constraints are viewed as component codes which typically have minimum distance $d_{\text{min}} \geq 3$: they are bit MAP decoded and the component EXIT entropy $\mathtt{y}(\mathtt{x})$ has smallest degree $d_{\text{min}} - 1$ (see, e.g., [41]). The EXIT entropy $\mathtt{y}(\mathtt{x})$ is the function $\mathtt{y}(\mathtt{x}) \triangleq \mathbb{E}\frac{1}{\mathtt{r}}\sum_{i=1}^{\mathtt{r}} \mathtt{y}_i(\mathtt{x})$, where $\mathtt{r}$ is the length of a particular component code and where the expectation is taken with respect to all such component codes. The distribution λ can be freely chosen but must satisfy the design rate constraint $r = 1 - \frac{1-\int \mathtt{y}}{\int \lambda}$ where $\int \mathtt{y}$ is the rate of the average component code (Area Theorem). For example, consider GLDPC ensembles using $[2^p-1, 2^p-p-1, 3]$ binary Hamming codes as component codes. Then, when $\mathbb{E}d_{\text{min}} \geq 3$, the BP EXIT entropy has at least one discontinuity at the BP threshold. It is given as,

$$(\epsilon, h) = \Big(\frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}, \Lambda(\mathtt{y}(\mathtt{x}))\Big).$$

Theorem 3 shows that, in general, $\epsilon^{\text{BP}} \neq \epsilon^{\text{MAP}}$ (The BP threshold being not given by the stability condition whenever the right component code has $d_{\text{min}} \geq 3$). In the next table, the first example uses $[7,4,3]$ Hamming codes such that its design rate is $r = \frac{1}{7}$ with the pair $(\lambda, \mathtt{y}) = (\mathtt{x}, 3\mathtt{x}^2 + 4\mathtt{x}^3 - 15\mathtt{x}^4 + 12\mathtt{x}^5 - 3\mathtt{x}^6)$ whereas the second example uses the $[15,11,3]$ Hamming code. It can be observed that this classical GLDPC have relatively bad BP threshold compared to its MAP upper-bound. In the third example, d_{min} is no longer > 2 since we choose, in the node perspective, a mixture of 40 percent of $[7,6,2]$ Single Parity-Check codes, 40 percent of $[7,4,3]$ Hamming codes and 20 percent of $[15,11,3]$ Hamming codes. The BP EXIT function has however still a discontinuity at the BP threshold.

$\lambda(\mathtt{x})$	$\mathtt{y}(\mathtt{x})$	ϵ^{BP}	$\overline{\epsilon}^{\text{MAP}}$	ϵ^{Sh}
$\mathtt{x}$	$[7,4,3]$	0.75645	0.85616	0.85714
$\mathtt{x}$	$[15,11,3]$	0.46785	0.52780	0.53333
$\frac{3\mathtt{x}+7\mathtt{x}^8}{10}$	mixture	0.70483	0.71301	0.72801

VIII. Conclusion

We have shown that there is a close connection between the BP and the MAP decoder. While this connection is quite general, we focused in this paper on communication over the binary erasure channel. In this case, the relation is furnished

[9] An alternative way is to show it via the Shannon Coding Theorem!

[10] Observe that $\epsilon(h)$ does not uniformly converge to 0 on $(0,1)$ since $\int_0^1 \epsilon(h)\mathrm{d}h = 1 - r \neq 0$.

[11] Contrary to the left nodes which stay simple repetition codes, the right nodes can be more complex linear codes. Therefore, $\mathtt{y}(\mathtt{x})$ often depends on the edge type. For GLDPC ensembles, we consider the average over all types of node. For Turbo codes, one usually distinguish between systematic versus parity bits.

by the so-called Maxwell decoder which gives an operational meaning to the various areas under the EBP EXIT curve as number of guesses and number of confirmations. Unfortunately, this paper falls slightly short on several accounts of proving this relationship in the most general case. Let us summarize what seem to be the most important issues that still need to be addressed.

First, there is currently no direct proof which establishes the existence of the asymptotic MAP EXIT curve. Rather, the existence follows from the explicit characterization of this limit. This occurs via Theorem 10 in all those cases where the conditions of the theorem are fulfilled. Although theses conditions apply to a large class of ensembles, it would be pleasing to show the existence of the limit in the general case.

A further point that needs some clarification is the restriction we had to impose in the second proof of Theorem 8. Recall that the argument on the computation tree via the Area Theorem required that the underlying ensemble has a non-trivial stability condition, since otherwise part of the EBP EXIT curve lies "outside the unit box," i.e., part of the curve corresponds to "erasure probabilities above one." While an analytical prove of Theorem 8 is possible, it would be interesting (especially in view of generalizations) to have a conceptual proof valid for unconditionally stable ensembles.

Without doubt the most important challenge is to assert the correctness of Conjecture 1. This would yield an easy and geometrically pleasing way of constructing the MAP EXIT curve from the EBP EXIT curve in the general case.

Finally, an interesting research direction consists in the analysis of more general combinatorial search problems through a suitable 'Maxwell construction'. An example (extremely close to the topic of this paper) consists in the problem of satisfiability of random sparse linear systems ('XORSAT') considered in [45], [46]. The counting argument presented in Section V is indeed closely related to the approach of these papers. The ideas presented here can probably be used to analyze the behavior of simple resolution algorithms for this problem (see [47] for a numerical exploration).

ACKNOWLEDGMENT

The authors would like to thank Nicolas Macris, Changyan Di, Gerhard Kramer, and Tom Richardson for useful discussions.

A.M. has been partially supported by EVERGROW (i.p. 1935 in the complex systems initiative of the Future and Emerging Technologies directorate of the IST Priority, EU Sixth Framework).

APPENDIX I
PROOFS FOR CONCENTRATION THEOREMS

Throughout this section, we use the shorthand $H_n = H_{\mathtt{G}}(X|Y)$ to denote the conditional entropy under transmission over the BMS channel $p_{Y_{[n]}\,|\,X_{[n]}}(\cdot\,|\,\cdot)$ using a code $\mathtt{G}$ chosen uniformly at random from LDPC(n, λ, ρ).

A. Concentration of the Conditional Entropy

Fix an arbitrary order for the $m = (1-r)n$ parity-check nodes, and let $\mathtt{G}_t$, $t \in [m]$, be a random variable describing the first t parity-check equations. Furthermore, let $\mathtt{G}_0$ be a trivial (empty) random variable. Define the Doob martingale $Z_t \equiv \mathbb{E}[H_n\,|\,\mathtt{G}_t]$. The martingale property $\mathbb{E}[Z_{t+1}\,|\,Z_0,\dots,Z_t] = Z_t$ follows by construction. In order to stress that Z_t is a (deterministic) function of the random variable $\mathtt{G}_t$, we will write $Z_t = Z(\mathtt{G}_t)$. Obviously, $Z_0 = \mathbb{E}[H_n]$ is the expected conditional entropy over the code ensemble, and $Z_m = H_n \equiv H_{\mathtt{G}}(X\,|\,Y)$ is the conditional entropy for a random code $\mathtt{G}$. Theorem 4 follows therefore from the Hoeffding-Azuma inequality, once we bound the differences $|Z_{t+1} - Z_t|$. This is our aim in the remaining of this subsection.

Assume, for the sake of definiteness, that parity-checks have been ordered by increasing degree. The first m_1 of them have degree $\mathtt{r}_1$, the successive m_2 have degree $\mathtt{r}_2$, and so on, with $\mathtt{r}_1 < \mathtt{r}_2 < \dots$. The $(t+1)^{\text{th}}$ parity-check will therefore have a well defined degree, to be denoted by $\mathtt{r}$. Consider two realizations $\mathtt{G}_{t+1}$ and $\mathtt{G}'_{t+1}$ of the first $(t+1)$ parity-checks which differ uniquely in the $(t+1)^{\text{th}}$ check. Let $\mathtt{G}$ be a code uniformly distributed over LDPC(λ, ρ, n) whose restriction to the first $(t+1)$ parity-checks coincides with $\mathtt{G}_{t+1}$. Construct a new code $\mathtt{G}'$ whose restriction to the first $(t+1)$ parity-checks is $\mathtt{G}'_{t+1}$, and which differs from $\mathtt{G}$ in at most $(\mathtt{r}+1)$ parity-checks. This can be done by the 'switching' procedure of [17]. This switching procedure results in a "pairing up" of graphs. In order to obtain the desired result, it is now enough to show that $|H_{\mathtt{G}}(X\,|\,Y) - H_{\mathtt{G}'}(X\,|\,Y)| \le \alpha$, for some n-independent constant α.

Let us focus on the variation in conditional entropy under the addition of a single parity-check. Let $\mathtt{G}$ be a generic linear code and let $\mathtt{G}+1$, be the same code with the added constraint that $x_{i_1} \oplus \cdots \oplus x_{i_{\mathtt{r}}} = \mathtt{0}$. Define the corresponding parity bit $\tilde{x} = x_{i_1} \oplus \cdots \oplus x_{i_{\mathtt{r}}}$, Then

$$\begin{aligned} H_{\mathtt{G}}(X\,|\,Y) &= H_{\mathtt{G}}(X\,|\,\tilde{X}, Y) + H_{\mathtt{G}}(\tilde{X}\,|\,Y) - H_{\mathtt{G}}(\tilde{X}\,|\,X, Y) \\ &= H_{\mathtt{G}}(X\,|\,\tilde{X}=0, Y) + H_{\mathtt{G}}(\tilde{X}\,|\,Y) \\ &= H_{\mathtt{G}+1}(X\,|\,Y) + H_{\mathtt{G}}(\tilde{X}\,|\,Y)\,. \end{aligned}$$

The second equality follows since $H_{\mathtt{G}}(\tilde{X}\,|\,X, Y) = 0$ and by using the channel symmetry. The third step is a consequence of the definition of $\mathtt{G}+1$. Since $\tilde{X}$ is a bit, its entropy is between 0 and 1 and therefore

$$|H_{\mathtt{G}}(X\,|\,Y) - H_{\mathtt{G}+1}(X\,|\,Y)| \le 1\,. \tag{23}$$

Recall that $\mathtt{G}$ and $\mathtt{G}'$ differ in at most $(\mathtt{r}+1)$ parity-checks, where $\mathtt{r}$ is upper bounded by $\mathtt{r}_{\text{max}}$, the maximal check-node degree. Equation (23) implies $|H_{\mathtt{G}}(X\,|\,Y) - H_{\mathtt{G}'}(X\,|\,Y)| \le (\mathtt{r}+1)$ and, therefore, Theorem 4.

B. Concentration of the Derivative of the Conditional Entropy

It is convenient to introduce the per-bit conditional entropy $h_n(\epsilon) \triangleq \frac{1}{n} H_{\mathtt{G}}(X|Y)$ and its expected value $\overline{h}_n(\epsilon) \triangleq \frac{1}{n}\mathbb{E} H_{\mathtt{G}}(X|Y)$ when $\mathtt{G}$ is a random code drawn uniformly from the LDPC(λ, ρ, n) ensemble.

Since the channel family $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ is smooth and ordered by physical degradation, $h_n(\epsilon)$ is differentiable convex function of $\epsilon \in I$. Therefore

$$\frac{1}{\Delta}[h_n(\epsilon) - h_n(\epsilon-\Delta)] \le h'_n(\epsilon) \le \frac{1}{\Delta}[h_n(\epsilon+\Delta) - h_n(\epsilon)]\,, \tag{24}$$

for any $\Delta > 0$ such that $[\epsilon-\Delta, \epsilon+\Delta] \in I$. Because of Theorem 4, we also have

$$\frac{1}{\Delta}[\bar{h}_n(\epsilon) - \bar{h}_n(\epsilon-\Delta) - 2\tilde{\xi}] \le h'_n(\epsilon) \le \le \frac{1}{\Delta}[\bar{h}_n(\epsilon+\Delta) - \bar{h}_n(\epsilon) + 2\tilde{\xi}]\,,$$

with probability greater than $1 - Ae^{-nB\tilde{\xi}^2}$ (it follows from the proof in the previous subsection that A and B can be chosen uniformly in ϵ). By averaging (24) over the code $\mathtt{G}$, and subtracting it from the last equation, we get

$$|h'_n(\epsilon) - \bar{h}'_n(\epsilon)| \le \frac{1}{\Delta}[\bar{h}_n(\epsilon+\Delta) - 2\bar{h}_n(\epsilon) + \bar{h}_n(\epsilon-\Delta) + 2\tilde{\xi}]\,,$$

which, using the convexity of $\bar{h}_n(\epsilon)$, and fixing $\Delta = \tilde{\xi}^{1/2}$, implies

$$|h'_n(\epsilon) - \bar{h}'_n(\epsilon)| \le [\bar{h}'_n(\epsilon+\tilde{\xi}^{1/2}) - \bar{h}'_n(\epsilon-\tilde{\xi}^{1/2})] + 2\tilde{\xi}^{1/2}\,.$$

The functions $\bar{h}_n$ are differentiable and convex and (by hypothesis) they converge to $\bar{h}(\epsilon) = h^{\text{MAP}}(\epsilon) = \lim_{n\to\infty}\frac{1}{n}\mathbb{E}H_n$ which is differentiable in J. It is a standard result in convex analysis (see [48]) that the derivatives $\bar{h}'_n$ converge to $\bar{h}'$ uniformly in J. Therefore, there exists a sequence $\delta_n \to 0$, such that

$$|h'_n(\epsilon) - \bar{h}'_n(\epsilon)| \le [\bar{h}'(\epsilon+\tilde{\xi}^{1/2}) - \bar{h}'(\epsilon-\tilde{\xi}^{1/2})] + \delta_n + 2\tilde{\xi}^{1/2}\,.$$

with probability greater than $1 - Ae^{-nB\tilde{\xi}^2}$. In order to complete the proof, it is sufficient to let $\tilde{\xi}_*(\xi)$ be the largest value of $\tilde{\xi}$, such that $[\bar{h}'(\epsilon+\tilde{\xi}^{1/2}) - \bar{h}'(\epsilon-\tilde{\xi}^{1/2})] + 2\tilde{\xi}^{1/2} < \xi/2$. Then the thesis holds with $\alpha_\xi = B\tilde{\xi}_*^2(\xi)/2$. In particular, if $\bar{h}(\epsilon)$ is twice differentiable with respect to $\epsilon \in J$, then $[\bar{h}'(\epsilon+\tilde{\xi}^{1/2}) - \bar{h}'(\epsilon-\tilde{\xi}^{1/2})] \le \tilde{A}\tilde{\xi}^{1/2}$, and $\tilde{\xi}_*(\xi) \ge \tilde{A}'\xi^2$.

Appendix II
Proofs of Lemmas in the Counting Argument

A. Proof of Lemma 6

Let $\mathtt{G}(t)$ denote the residual graph after t iterations of the message passing decoder, and $\Xi_{\mathtt{G}(t)} = (\Lambda_{\mathtt{G}(t)}, \Gamma_{\mathtt{G}(t)})$ be the corresponding degree distribution pair. Moreover, denote by $\Xi_t = (\Lambda_t, \Gamma_t)$ the typical degree distribution pair of $\mathtt{G}(t)$. Explicitly

$$\Lambda_t(\mathtt{z}) \triangleq \Lambda(\mathtt{z}\mathtt{x}_t)\,,$$
$$\Gamma_t(\mathtt{z}) \triangleq \Gamma(1-\mathtt{y}_t+\mathtt{z}\mathtt{y}_t) - \Gamma(1-\mathtt{y}_t) - \mathtt{z}\mathtt{y}_t\Gamma'(1-\mathtt{y}_t)\,,$$

where $\mathtt{x}_t, \mathtt{y}_t$ denote the typical fractions of erased messages after t iterations of the decoder. These are obtained by solving the density evolution equations $\mathtt{x}_{t+1} = \epsilon\lambda(\mathtt{y}_t)$, $\mathtt{y}_{t+1} = 1 - \rho(1-\mathtt{x}_t)$ with initial condition $\mathtt{x}_0 = \mathtt{y}_0 = 1$.

Notice that

$$d(\Xi_\epsilon, \Xi_{\mathtt{G}\epsilon}) \le d(\Xi_\epsilon, \Xi_t) + d(\Xi_t, \Xi_{\mathtt{G}(t)}) + d(\Xi_{\mathtt{G}(t)}, \Xi_{\mathtt{G}(\epsilon)})\,.$$

We claim that

$$\lim_{t\to\infty} d(\Xi_{\mathtt{G}(t)}, \Xi_{\mathtt{G}(\epsilon)}) = 0\,, \tag{25}$$
$$\lim_{n\to\infty} \mathbb{E}[d(\Xi_t, \Xi_{\mathtt{G}(t)})] = 0\,, \tag{26}$$
$$\lim_{t\to\infty}\lim_{n\to\infty} \mathbb{E}[d(\Xi_\epsilon, \Xi_t)] = 0\,. \tag{27}$$

Before proving those claims, let us show that they imply the thesis. It follows from the triangular inequality above that $\lim_{t\to\infty}\lim_{n\to\infty}\mathbb{E}\,d(\Xi_\epsilon, \Xi_{\mathtt{G}(\epsilon)}) = 0$. But $d(\Xi_\epsilon, \Xi_{\mathtt{G}(\epsilon)})$ does not depend upon t, therefore

$$\lim_{n\to\infty} \mathbb{E}[d(\Xi_\epsilon, \Xi_{\mathtt{G}(\epsilon)})] = 0\,.$$

This in turns imply the thesis via Markov inequality.

We must now prove the inequalities (25) to (27). The first one is a trivial consequence of the convergence of DE to its fixed point: $\lim_{t\to\infty}\mathtt{x}_t = \mathtt{x}$, $\lim_{t\to\infty}\mathtt{y}_t = \mathtt{y}$, together with the continuity of the expressions (8), (9) with $\mathtt{x}, \mathtt{y}$. Eq. (26) follows from the general concentration analysis in [17].

In order to prove (27), consider a variable node i in the residual graph and imagine changing the received symbol at i, and update all the messages consequently. Consider the edges whose distance from i is larger than t, and denote by $W_i^{(t)}$ the number of messages on such edges that change of value after the received symbol at i has been changed. It is clear that

$$\mathbb{E}[d(\Xi_\epsilon, \Xi_t)] \le \mathbb{E}[W_i^{(t)}], \tag{28}$$

The limit $\lim_{n\to\infty}\mathbb{E}[W_i^{(t)}]$ can be computed through a branching process analysis. The calculation is very similar to the one in [49] and we do not reproduce it here. The final result is that, as long as $\epsilon\lambda'(\mathtt{y})\rho'(1-\mathtt{x}) < 1$, there exist two positive constants A, b with $b < 1$ such that $\mathbb{E}[W_i^{(t)}] \le A\,b^t$. The proof is finished by noticing that the condition $\epsilon\lambda'(\mathtt{y})\rho'(1-\mathtt{x}) < 1$ is satisfied whenever ϵ is a continuity point of $\mathtt{x}(\epsilon)$.

B. Proof of Lemma 8

Notice that the function $u \mapsto v(u)$ defined in (13) enjoys the property $v(1/u) = 1/v(u)$ for any $u > 0$. Assume *ab absurdum* that Ψ_Ξ does not achieves its maximum in the interval $[0,1]$. Therefore, there exist $u > 1$ such that $\Psi_\Xi(u') < \Psi_\Xi(u)$ for any $u' \in [0,1]$. We will show that $\Psi_\Xi(1/u) \ge \Psi_\Xi(u)$ thus reaching a contradiction. In fact, some algebra shows that

$$\Psi_\Xi(1/u) = -\Lambda'(1)\log_2\left[\frac{(1+uv)}{(1+u)(1+v)}\right] + \sum_{\mathtt{l}}\Lambda_{\mathtt{l}}\log_2\left[\frac{1+u^{\mathtt{l}}}{2(1+u)^{\mathtt{l}}}\right] + \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{\mathtt{r}}\Gamma_{\mathtt{r}}\log_2\left[1+\left(\frac{v-1}{v+1}\right)^{\mathtt{r}}\right].$$

The claim follows from $0 < \frac{v-1}{v+1} < 1$ together with the monotonicity of the logarithm.

In order to prove the second claim, i.e., the regularity of Ψ_Ξ with respect to the dd pair write $\Psi_\Xi^{(1)}(u) + \Psi_\Xi^{(2)}(u) + \Psi_\Xi^{(3)}(u)$ with $\Psi_\Xi^{(1,2,3)}$ the three summands in (12). The estimate (15) can be proved for each of the three terms separately. Here, we limit ourselves to consider $\Psi_{\Xi}^{(1)}(u)$, the derivation being nearly identical for the two other summands. Start by noticing that, for any $u \in [0,1]$ and any dd pair , we have

$$\frac{1}{2} \leq \sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}}}{1+u^{\mathtt{l}}} \leq 1\,, \qquad \sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}} u^{\mathtt{l}-1}}{1+u^{\mathtt{l}}} \leq 1\,.$$

Now fix two dd pair Ξ and $\tilde{\Xi}$. Let $v(u)$ and $\tilde{v}(u)$ the corresponding functions defined as in (13). Notice that

$$\begin{aligned}\left|\sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}}-\lambda_{\mathtt{l}}}{1+u^{\mathtt{l}}}\right| &= \left|\sum_{\mathtt{l}} \left(\frac{1}{1+u^{\mathtt{l}}}-\frac{1}{2}\right)(\lambda_{\mathtt{l}}-\lambda_{\mathtt{l}})\right| \\ &\leq \frac{\mathtt{l}_{\max}}{2}(1-u)\sum_{\mathtt{l}} |\lambda_{\mathtt{l}}-\lambda_{\mathtt{l}}| \\ &\leq \frac{1}{2}\mathtt{l}_{\max}^2(1-u)\, d(\Xi,\tilde{\Xi})\end{aligned}$$

Using these inequalities, some calculus shows that

$$\begin{aligned} 1 \geq v(u), \tilde{v}(u) \geq 1-2\,\mathtt{l}_{\max}(1-u)\,, \\ |v(u)-\tilde{v}(u)| \leq 3\,\mathtt{l}_{\max}^2\,(1-u)\, d(\Xi,\tilde{\Xi})\,.\end{aligned}$$

Next notice that, if we set $f(u,v) \triangleq \log_2\left[\frac{2(1+uv)}{(1+u)(1+v)}\right]$, then, for any $u,v,\tilde{v} \in [0,1]$, we have

$$\begin{aligned}|f(u,v)| &\leq \frac{(1-u)(1-v)}{\log 2}\,, \\ |f(u,v)-f(u,\tilde{v})| &\leq \frac{(1-u)}{\log 2}\,|v-\tilde{v}|\,.\end{aligned}$$

Using these observations we obtain

$$\begin{aligned}|\Psi_{\Xi}(u)-\Psi_{\tilde{\Xi}}(u)| &\leq \max[f(u,v),f(u,\tilde{v})]\,|\Lambda'(1)-\tilde{\Lambda}'(1)| \\ &\quad + \max[\Lambda'(1),\tilde{\Lambda}'(1)]\,|f(u,v)-f(u,\tilde{v})| \\ &\leq \frac{2\mathtt{l}_{\max}}{\log 2}(1-u)^2|\Lambda'(1)-\tilde{\Lambda}'(1)| \\ &\quad + \frac{\mathtt{l}_{\max}}{\log 2}(1-u)|v-\tilde{v}| \\ &\leq A_1\,(1-u)^2\, d(\Xi,\tilde{\Xi})\,,\end{aligned}$$

which confirms our thesis with constant $A_1 = (2\,\mathtt{l}_{\max}^2 + 3\mathtt{l}_{\max}^3)/\log 2$. The variations of $\Psi_{\Xi}^{(2)}$ and $\Psi_{\Xi}^{(3)}$ are bounded analogously.

APPENDIX III
AREA AND BP EXIT

A. Two Useful Tricks

We give here two lemmas which contain the two computational tricks which are used all along this paper. Lemma 14 and Lemma 15 will be again used in the next subsection of the appendix. Observe that the function $\mathtt{x} \mapsto h \triangleq \Lambda(\mathtt{y}(\mathtt{x}))$ is composed by two functions $\mathtt{y}$ and Λ which are strictly increasing over $[0,1]$. Therefore, the inverse function $\mathtt{x}(h)$ exists and $h \mapsto \mathtt{x}(h) \triangleq \mathtt{y}^{-1} \circ \Lambda^{-1}(h)$ is a continuous and strictly increasing bijection from $[0,1]$ to $[0,1]$. The values $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ can then equivalently be described by $\epsilon(h) \triangleq \frac{\mathtt{y}^{-1}\circ\Lambda^{-1}}{\lambda\circ\Lambda^{-1}}(h)$.

Lemma 14: Given a dd pair (λ,ρ) and any couple $(\mathtt{x}_a,\mathtt{x}_b) \in [0,1]^2$. With the notations $h_a = h(\mathtt{x}_a) \triangleq \Lambda \circ \mathtt{y}(\mathtt{x}_a)$ and $h_b = h(\mathtt{x}_b)$, we can then write

$$\int_{h_a}^{h_b} \epsilon(h)\mathrm{d}h = \frac{1}{\int\lambda}\left(\mathtt{x}_b\mathtt{y}(\mathtt{x}_b) - \mathtt{x}_a\mathtt{y}(\mathtt{x}_a) - \int_{\mathtt{x}_a}^{\mathtt{x}_b} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x}\right).$$

Proof: This is a simple integration by parts once it has been observed $\epsilon(\mathtt{x}) \cdot \frac{\mathrm{d}h(\mathtt{x})}{\mathrm{d}\mathtt{x}} = \frac{\mathtt{x}}{\lambda\circ\mathtt{y}(\mathtt{x})} \cdot \frac{(\lambda\circ\mathtt{y})(\mathtt{x})\cdot\mathtt{y}'(\mathtt{x})}{\int\lambda} = \frac{\mathtt{x}\mathtt{y}'(\mathtt{x})}{\int\lambda}$. ■

Lemma 15: Given a dd pair (λ,ρ) and any interval $(\mathtt{x}_a\mathtt{x}_b) \subseteq [0,1]$, $\mathtt{x}^{\mathrm{BP}} \leq \mathtt{x}_a$ over which $\epsilon(\mathtt{y}) \triangleq \frac{\mathtt{x}}{\lambda\circ\mathtt{y}(\mathtt{x})}$ is increasing. Then, the function $h^{\mathrm{BP}}(\epsilon)$ is continuous over (ϵ_a,ϵ_b), where $\epsilon_a \triangleq \epsilon(\mathtt{x}_a)$ and $\epsilon_b \triangleq \epsilon(\mathtt{x}_b)$, and

$$\begin{aligned}\int_{\epsilon_a}^{\epsilon_b} h^{\mathrm{BP}}(\epsilon)\mathrm{d}\epsilon = \frac{1}{\int\lambda}\Big(\epsilon_b \int_0^{\mathtt{y}(\mathtt{x}_b)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \epsilon_a \int_0^{\mathtt{y}(\mathtt{x}_a)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} \\ - \mathtt{x}_b\mathtt{y}(\mathtt{x}_b) + \mathtt{x}_a\mathtt{y}(\mathtt{x}_a) + \int_{\mathtt{x}_a}^{\mathtt{x}_b} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x}\Big).\end{aligned}$$

Proof: This is proved by, first, integrating by parts and, second, using Lemma 14. ■

B. Area under the BP EXIT Curve

Theorem 11 (Area Theorem for BP Decoding): Given a dd pair (λ,ρ) and the asymptotic BP EXIT entropy as defined in Corollary 1, then

$$r + \frac{1}{\int\lambda}\sum_{i=1}^{J} D_i = \int_0^1 h^{\mathrm{BP}}(\epsilon)\mathrm{d}\epsilon,$$

where $D_i = A_i - B_i - C_i$ with $A_i \triangleq \underline{\mathtt{x}}^i\mathtt{y}(\underline{\mathtt{x}}^i) - \overline{\mathtt{x}}^{i-1}\mathtt{y}(\overline{\mathtt{x}}^{i-1})$, $B_i \triangleq \epsilon^i \int_{\mathtt{y}(\overline{\mathtt{x}}^{i-1})}^{\mathtt{y}(\underline{\mathtt{x}}^i)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y}$, and $C_i = \int_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x}$.

Proof: Using Corollary 1, we can derive (29) as shown above where (a) comes from Lemma 15 and (b) uses the fact that $\epsilon^i = \epsilon(\overline{\mathtt{x}}^{i-1}) = \epsilon(\underline{\mathtt{x}}^i)$. ■

First, observe that Theorem 11 quantifies the average sub-optimality of BP decoding compared to MAP decoding. The area under the BP EXIT curve is trivially larger or equal than the design rate since the D_i's are non-negative. Moreover, it seems to indicate that there performance loss occurs at each phase transition.

Second, Theorem 11 has a pleasing geometric interpretation which goes back to the asymptotic analysis and which is explained in appendix IV.

APPENDIX IV
DYNAMIC INTERPRETATION OF THE AVERAGE GAP BETWEEN MAP AND BP DECODING

It is now well-known that the determination of capacity-achieving sequences on the erasure channel reduces to a curve-fitting problem, see, e.g., [50], [40]. This was the motivation for the Area Theorem and - so far - its unique application. Let us recall this view. For the purpose of illustration, and without essential loss of generality, we focus on the case of (G)LDPC ensembles.

$$
\begin{aligned}
&\int_0^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon \\
&= \int_0^{\epsilon^{\text{BP}}} h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon + \sum_{i=1}^{J} \int_{\epsilon^i}^{\epsilon^{i+1}} h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon \\
&\stackrel{(a)}{=} 0 + \frac{1}{\int \lambda} \sum_{i=1}^{J} \left(\left[\epsilon(\mathtt{x}) \int_0^{h(\mathtt{x})} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} \right]_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} - \left[\mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} + \int_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right) \\
&= \frac{\left(\int_0^1 \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \sum_{i=1}^{J} \left[\epsilon(\mathtt{x}) \int_0^{h(\mathtt{x})} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \right) - \left(1 - \sum_{i=1}^{J} \left[\mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \right) + \left(\int_0^1 \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} - \sum_{i=1}^{J} \int_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right)}{\int \lambda} \\
&\stackrel{(b)}{=} \frac{\int \lambda - 1 + \int \mathtt{y}}{\lambda} + \frac{1}{\int \lambda} \sum_{i=1}^{J} \left(\left[\mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} - \epsilon^i \int_{\mathtt{y}(\overline{\mathtt{x}}^{i-1})}^{\mathtt{y}(\underline{\mathtt{x}}^i)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \int_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right) \qquad (29)
\end{aligned}
$$

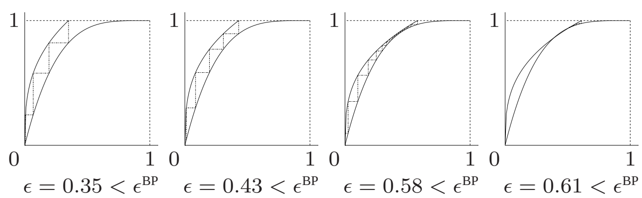


Fig. 26. Iterative decoding trajectory for the ensemble LDPC$(n, \mathtt{x}^3, \mathtt{x}^4)$ (in the limit when $n \to \infty$): increasing values of the channel parameter ϵ.

A. EXIT Chart

Fig. 26 summarizes the DE analysis of the BP decoding by showing the convergence of the recursive sequence formed the edge entropy $\{\mathtt{x}_t\}_t$ (i.e., the edge erasure probability). Such a representation (which emphasizes two component EXIT functions, one associated to the left nodes and one associated to the right nodes) is called EXIT chart in [11]. This representation is (asymptotically) exact for the binary erasure channel (since it is DE) whereas it is only approximate in the general case.

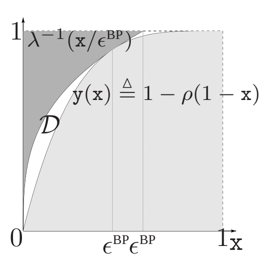


Fig. 27. Additive gap to capacity for the dd pair (x^3, x^4).

Fig. 27 represents the EXIT chart when transmission takes place at the BP threshold $\epsilon = \epsilon^{\text{BP}}$. The EXIT functions are here the ones associated to the component of the LDPC ensemble. The function on the left is associated to repetition codes on the left while the one on the right is associated to parity-check codes. At channel parameter $\epsilon = \epsilon^{\text{BP}}$, the two EXIT curves are tangent in $(\mathtt{x}^{\text{BP}}, \mathtt{y}^{\text{BP}})$ and the EXIT chart offers also a graphical representation of the limiting gap to capacity of the LDPC ensemble. The additive gap $C(\epsilon^{\text{BP}}) - r$ to the Shannon threshold is indeed represented by the entire white area $\mathcal{D}$ such that

$$C(\epsilon^{\text{BP}}) - r = \epsilon^{\text{Sh}} - \epsilon^{\text{BP}} = \frac{\mathcal{D}}{\int \lambda},$$

where $\frac{1}{\int \lambda} = \Lambda'(1)$ is the average left degree. In words, the area $\mathcal{D}$ is the area between the left EXIT curve $\mathtt{x} \mapsto \lambda^{-1}(\mathtt{x}/\epsilon^{\text{BP}})$ (at the BP threshold) and the right EXIT curve $\mathtt{x} \mapsto 1 - \rho(1 - \mathtt{x})$ which is bounded away by the unit square. This statement is presented, e.g., in [40]. We will now refine this statement by applying the Area Theorem to the EXIT curve of the LDPC ensemble previous statement (i.e., using the basic principle of our method). We will see that, in short, the area $\mathcal{D}$ can be itself divided into two parts where the subarea below $\mathtt{x}^{\text{BP}}$ represents the average gap between MAP and BP decoding. The determination of LDPC codes for which BP decoding is MAP reduces then again to a curve-fitting problem below $\mathtt{x}^{\text{BP}}$.

B. Geometric Interpretation at the Component Level

Fig. 28 shows a geometric representation of Theorem 11. In (a) one see that the additive gap between BP threshold and Shannon threshold is represented by the total area between the component EXIT functions. Further, the part of this area which corresponds to the average gap between MAP and BP decoding is D_1 as defined in Theorem 11.

References

[1] C. Méasson and R. Urbanke, “An upper-bound for the ML threshold of iterative coding systems over the BEC,” in *Proc. of the Forty-First Allerton Conference*, Allerton House, Monticello, USA, October 1–3 2003.

[2] A. Montanari, “Why “practical” decoding algorithms are not as good as “ideal” ones?” in *Proc. DIMACS Workshop on Codes and Complexity*, Rutgers University, Piscataway, USA, December 4–7 2001, pp. 63–66.

[3] C. Méasson, A. Montanari, and R. Urbanke, “Maxwell’s construction: The hidden bridge between maximum-likelihood and iterative decoding,” in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Chicago, 2004.

[4] C. Méasson, A. Montanari, T. Richardson, and R. Urbanke, “Life above threshold: From list decoding to area theorem and MSE,” in *Proc. of the IEEE Inform. Theory Workshop*, San Antonio, Texas, October 24–29 2004.

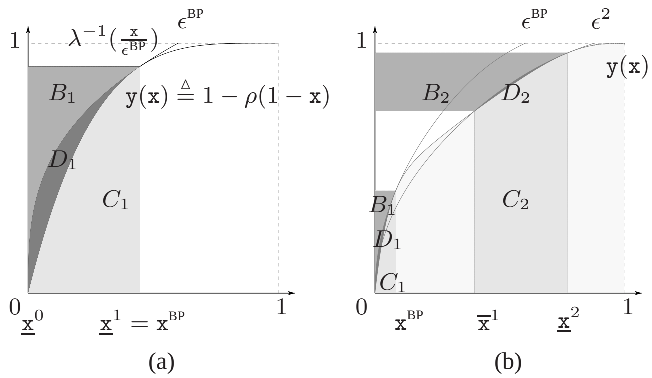


Fig. 28. Graphical interpretation of Theorem 11 at a microscopic (dynamic) level: (a) Standard one-jump case: Ensemble LDPC(x^3, x^4) and transmission at $\epsilon = \epsilon^{BP}$ (b) Double-jump case: The left distribution is $\lambda(x) = 0.78x^2 + 0.1x^3 + 0.12x^{14}$ and the y represents the EXIT function of a mixture of component codes composed by 50% of $[19, 18]$ single parity-check codes, 35% of $[7, 4]$ Hamming codes and 15% of $[15, 11]$ Hamming codes in the edge perspective. Transmission is represented for two channel parameters.

[5] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Mateo: Morgan Kaufmann Publishers, 1988.

[6] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping University, S-581 83, Linköping, Sweden, 1996.

[7] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.

[8] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[9] T. Etzion, A. Trachtenberg, and A. Vardy., "Which codes have cycle-free Tanner graphs?" *IEEE Trans. Inform. Theory*, vol. 45, no. 6, pp. 2173 – 2181, Sept. 1999.

[10] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2005, in preparation.

[11] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Inform. Theory*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[12] A. Ashikhmin, G. Kramer, and S. ten Brink, "Code rate and the area under extrinsic information transfer curves," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Lausanne, Switzerland, June 30–July 5 2002, p. 115.

[13] H. Pishro-Nik and F. Fekri, "On decoding of low-density parity-check codes over the binary erasure channel," *IEEE Trans. Inform. Theory*, vol. 50, no. 3, pp. 439–454, Mar. 2004.

[14] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, Massachusetts: M.I.T. Press, 1963.

[15] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[16] ——, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inform. Theory*, vol. 47, pp. 585–598, 2001.

[17] T. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.

[18] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–637, Feb. 2001.

[19] A. Montanari, "The glassy phase of Gallager codes," *European Physical Journal*, vol. 23, no. 121, 2001, arXiv:cond-math/cond-mat/0104079.

[20] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding," in *Proceedings of ICC'93*, Geneve, Switzerland, May 1993, pp. 1064–1070.

[21] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 429–445, Mar. 1996.

[22] C. Méasson, A. Montanari, T. Richardson, and R. Urbanke, "The generalized area theorem and some of its consequences," 2005, in preparation.

[23] A. Montanari, "Tight bounds for LDPC and LDGM codes under MAP decoding," *IEEE Trans. Inform. Theory*, 2004, submitted.

[24] G. Zémor and G. Cohen, "The threshold probability of a code," *IEEE Trans. Inform. Theory*, vol. 41, pp. 469–477, Mar. 1995.

[25] L. Bazzi, T. Richardson, and R. Urbanke, "Exact thresholds and optimal codes for the binary-symmetric channel and Gallager's decoding algorithm A," *IEEE Trans. Inform. Theory*, vol. 50, no. 9, Sept. 2004.

[26] B. Bollabás, *Random Graphs*. Cambridge University Press, 2001.

[27] C. Di, A. Montanari, and R. Urbanke, "Weight distributions of LDPC code ensembles: combinatorics meets statistical physics," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Chicago, USA, June 27–July 2 2004, p. 102.

[28] R. G. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, Jan. 1962.

[29] G. Miller and D. Burshtein, "Asymptotic enumeration method for analyzing LDPC codes," *IEEE Trans. Inform. Theory*, vol. 50, no. 6, pp. 1115–1131, June 2004.

[30] S. L. Litsyn and V. S. Shevelev, "On ensembles of low-density parity-check codes: asymptotic distance distributions," *IEEE Trans. Inform. Theory*, vol. IT–48, pp. 887 –908, Apr. 2002.

[31] ——, "Distance distribution in ensembles of irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. IT–49, pp. 3140 –3159, Dec. 2003.

[32] C. Di, T. Richardson, and R. Urbanke, "Weight distribution of iterative coding systems: How deviant can you be?" in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Washington, USA, June 24–29 2001.

[33] ——, "Weight distribution of low-density parity-check codes," *IEEE Trans. Inform. Theory*, 2004, submitted.

[34] C. Di, "Asymptotic and finite-length analysis of low-density parity-check codes," Ph.D. dissertation, EPFL, Lausanne, Switzerland, 2004, thèse no 3072.

[35] R. Sedgewick and P. Flajolet, *An Introduction to Analysis of Algorithms*. Addison-Wesley, 1996.

[36] P. Flajolet and R. Sedgewick, "The average case analysis of algorithms: Saddle point asymptotics," RR 2376, Tech. Rep., 1994.

[37] C. Kittel and H. Kroemer, *Thermal Physics*, 2nd ed. New York: W. H. Freeman and Co., Mar. 2002.

[38] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431, 1999.

[39] A. Montanari, "The glassy phase of gallager codes," *Eur. Phys. J. B*, vol. 23, pp. 121–136, 2001.

[40] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel property," *IEEE Trans. Inform. Theory*, vol. 50, no. 11, pp. 2657–2673, Nov. 2004.

[41] C. Méasson and R. Urbanke, "Further analytic properties of EXIT-like curves and applications," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Yokohama, Japan, June 29–July 4 2003, p. 266.

[42] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. 27, no. 5, pp. 533–547, Sept. 1981.

[43] J. Boutros, O. Pothier, and G. Zémor, "Generalized low-density (Tanner) codes," in *Proceedings of the ICC'99*, Vancouver, Canada, June 1999, pp. 441-445.

[44] M. Lentmaier and K. Zigangirov, "Iterative decoding of generalized low-density parity-check codes," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, Boston, USA, August 16–21 1998, pp. 441-445.

[45] M. Mézard, F. Ricci-Tersenghi, and R. Zecchina, "Alternative solutions to diluted p-spin models and XORSAT problems," *J. Stat. Phys.*, vol. 111, p. 505, 2003, arXiv:cond-math/cond-mat/0207140.

[46] S. Cocco, O. Dubois, J. Mandler, and R. Monasson, "Rigorous decimation-based construction of ground pure states for spin glass models on random lattices," *Phys. Rev. Lett.*, vol. 90, no. 047205, 2003, arXiv:cond-math/cond-mat/0206239.

[47] A. Braunstein, M. Leone, F. Ricci-Tersenghi, and R. Zecchina, "Complexity transitions in global algorithms for sparse linear systems over finite fields," *J. Stat. Phys.*, vol. 35, p. 7559, 2002, arXiv:cond-math/cond-mat/0203613.

[48] R. Rockafellar, *Convex Analysis*. Princeton: Princeton University Press, 1970.

[49] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 638–656, Feb. 2001.

[50] A. Shokrollahi, "Capacity-achieving sequences," in *Codes, Systems, and Graphical Models*, ser. IMA Volumes in Mathematics and its Applications, B. Marcus and J. Rosenthal, Eds., vol. 123. Springer-Verlag, 2000, pp. 153–166.