# CONSERVATION LAWS FOR CODING

THÈSE N$^{O}$ 3485 (2006)

PRÉSENTÉE LE 31 MARS 2006

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS

Laboratoire de théorie des communications

SECTION DES SYSTÈMES DE COMMUNICATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Cyril MEASSON

Ingénieur ENSEEIHT, Toulouse, France
et de nationalité française

acceptée sur proposition du jury:

Prof. B. Rimoldi, président du jury
Prof. R. Urbanke, Dr A. Montanari, directeurs de thèse
Prof. D. Forney, rapporteur
Prof. J. Hagenauer, rapporteur
Prof. M. Shokrollahi, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Lausanne, EPFL
2006

# Abstract

This work deals with coding systems based on sparse graph codes. The key issue we address is the relationship between iterative (in particular belief propagation) and maximum a posteriori decoding. We show that between the two there is a fundamental connection, which is reminiscent of the *Maxwell construction* in thermodynamics.

The main objects we consider are EXIT-like functions. EXIT functions were originally introduced as handy tools for the design of iterative coding systems. It gradually became clear that EXIT functions possess several fundamental properties. Many of these properties, however, apply only to the erasure case. This motivates us to introduce *GEXIT functions* that coincide with EXIT functions over the erasure channel. In many aspects, GEXIT functions over general memoryless output-symmetric channels play the same role as EXIT functions do over the erasure channel. In particular, GEXIT functions are characterized by the *general area theorem*. As a first consequence, we demonstrate that in order for the rate of an ensemble of codes to approach the capacity under belief propagation decoding, the GEXIT functions of the component codes have to be matched perfectly. This statement was previously known as the *matching condition* for the erasure case.

We then use these GEXIT functions to show that in the limit of large blocklengths a fundamental connection appears between belief propagation and maximum a posteriori decoding. A decoding algorithm, which we call *Maxwell decoder*, provides an operational interpretation of this relationship for the erasure case. Both the algorithm and the analysis of the decoder are the translation of the Maxwell construction from statistical mechanics to the context of probabilistic decoding. We take the first steps to extend this construction to general memoryless output-symmetric channels. More exactly, a general upper bound on the maximum a posteriori threshold for sparse graph codes is given. It is conjectured that the fundamental connection between belief propagation and maximum a posteriori decoding carries over to the general case.

**Key words** probabilistic decoding, sparse graphs, threshold, belief propagation, maximum a posteriori, maximum likelihood, phase transition, EXIT chart, Maxwell construction, entropy, area theorem

# Zusammenfassung

Diese Arbeit behandelt Codierungssysteme, die auf graphischen Codes basieren. Der Schwerpunkt liegt auf der Beziehung zwischen der optimalen (Maximum a Posteriori) und der iterativen (Belief Propagation) Decodierung. Wir zeigen, daß die Verbindung dieser zwei Codierungsarten durch eine Konstruktion gegeben ist, die identisch mit der *Maxwell-Konstruktion* in der Thermodynamik ist.

Unser Hauptaugenmerk liegt auf einem Funktionstyp, der *EXIT-Funktionen* sehr ähnelt. EXIT-Funktionen wurden ursprünglich als handliches Mittel zum Design von iterativen Codierungssystemen eingeführt. Es stellte sich heraus, daß EXIT-Funktionen einige wichtige grundlegende Eigenschaften haben. Einige dieser Eigenschaften lassen sich aber einzig auf den Löschkanal anwenden. Dies führte zu der Idee, *GEXIT-Funktionen* einzuführen, die im Fall des Löschkanals den EXIT-Funktionen entsprechen. Die GEXIT-Funktionen haben in vielen Fällen die gleichen Eigenschaften in Bezug auf allgemeine symmetrische Kanäle ohne Gedächtnis wie EXIT-Funktionen für Löschkanäle. Im Besonderen erfüllen die GEXIT-Funktionen das *allgemeine Flächenerhaltungsgesetz*. Als eine erste Anwendung dieses Theorems zeigen wir, daß eine perfekte Übereinstimmung der GEXIT-Funktionen der Komponenten des Codes notwendig ist, um durch Belief Propagation die Rate der Kanalkapazität anzunähern. Diese Bedingung war bis jetzt nur für den Löschkanal bekannt.

Als weiteres Ergebnis zeigen wir, daß GEXIT-Funktionen für unendlich lange Blocklängen eine fundamentale Beziehung zwischen Belief Propagation und Maximum a Posteriori Decodierung aufzeigen. Ein Decodierungsalgorithmus, den wir *Maxwell-Decodierer* nennen, erlaubt eine operationelle Interpretierung für den Löschkanal. Sowohl der Algorithmus als auch die Analyse des Decodierers entstehen aus der Übertragung der Maxwell-Konstruktion von der statistischen Physik auf das Gebiet der wahrscheinlichkeitsbasierten Decodierung. Wir zeigen erste Schritte, um diese Konstruktion für allgemeine symmetrische Kanäle ohne Gedächtnis zu generalisieren. Genauer gesagt entwickeln wir eine allgemeine obere Schranke für den Maximum a Posteriori Schwellenwert von graphischen Codes. Des weiteren folgern wir, daß die grundlegende Beziehung zwischen Belief Propagation und Maximum a Posteriori Decodierung auf den allgemeinen Fall übertragen werden kann.

**Schlüsselwörter** wahrscheinlichkeitsbasierte Decodierung, Turbo-Codes, graphische Codes, Schwellenwert, Belief Propagation, Maximum a Posteriori, Phasenübergang, EXIT-Chart, Maxwell-Konstruktion, Entropie, Flächererhaltungsgesetz

# Version Abrégée

Ce travail se consacre aux systèmes de codage de type *Turbo codes*. L'accent est mis sur la relation entre le décodage à maximum de vraisemblance et le décodage itératif dit *à propagation de croyances*. On montre que les deux types de décodage sont liés par une construction identique à la construction de Maxwell en thermodynamique.

L'objet principal de notre étude est une fonction similaire à la fonction d'entropie de sortie qui est aussi appelée *fonction EXIT*. Il est vite apparu que les fonctions EXIT possèdent plusieurs propriétés extrêmement fortes. Malheureusement, la plupart d'entre elles ne s'appliquent qu'au canal à effacement. En introduisant les *fonctions GEXIT*, nous généralisons les propriétés fondamentales des fonctions EXIT. Plus exactement, les fonctions GEXIT et EXIT sont confondues sur le canal à effacement où elles partagent des propriétés communes. Elles diffèrent, en général, sur un canal symétrique et sans mémoire, où la fonction GEXIT conserve les mêmes propriétés. Les fonctions GEXIT satisfont notamment le *théorème général des aires*. Une première application de ce théorème est de généraliser la *condition d'ajustement* des courbes pour les codes constituants d'un code composite. Cette condition était, jusqu'à présent, connue uniquement dans le cadre du canal à effacement.

Une deuxième application, principale et fondamentale, est le fait que les courbes GEXIT contiennent, par essence, le lien entre décodage à maximum de vraisemblance et décodage itératif. Ce lien apparaît lorsque les longueurs de mots utilisées tendent vers l'infini. Dans le cadre du canal à effacement nous introduisons le *décodeur de Maxwell*. Cet algorithme et son analyse sont la traduction exacte de la construction de Maxwell, mais cette fois dans le domaine du décodage probabilistique. Nous formulons la conjecture que cette construction de Maxwell se généralise à tout canal symétrique sans mémoire. Nous apportons plusieurs éléments de réponse qui valident cette hypothèse. En particulier, nous donnons une borne supérieure fine sur le seuil de décodage à maximum de vraisemblance d'un ensemble de codes de type Turbo codes.

**Mots-clés** décodage probabilistique, Turbo codes, seuil, propagation de croyances, maximum a posteriori, maximum de vraisemblance, transition de phase, diagramme EXIT, construction de Maxwell, entropie, théorème des aires

# Acknowledgments

First I want to express my deepest gratitude to my thesis advisor, Rüdiger Urbanke, for many reasons, personal and professional. Thanks to him, I have deeply enjoyed my time in Lausanne, from running on the lake to the fun at work. I am also very grateful to my second thesis advisor Andrea Montanari who has the gift of making things easier. Working with Andrea and Rüdiger is a great pleasure as well as a continuous educational experience. Without their insights and contributions this work would not have been possible.

I am very indebted and grateful to my thesis committee. Joachim Hagenauer introduced me to the beauty of research by showing that surprising intuitions may translate into elegant results. David Forney has been a permanent source of motivation. His encouraging comments have always given a strong motivational boost. Amin Shokrollahi shows a great ability for explaining and clarifying things. I would like to thank him for varied interesting discussions (initially in German at Bell Labs, English later, and now French!).

I thank a great deal the committee president Bixio Rimoldi for his interest and enthusiasm in research, teaching and social events.

I would like to take the opportunity to thank Emre Telatar for precious advice as well as “ces petits riens qui signifient beaucoup” such as the art of PostScript programing. I would also like to thank Tom Richardson for helpful suggestions and discussions.

Finally, I would like to thank the students, the administrative staff and my colleagues from the I&C school at EPF Lausanne. Thanks for the friendly atmosphere and the “ready-to-help” attitude. I also would like to thank the colleagues I met during my time at TU Munich and at Bell Labs Murray Hill for the many and varied interesting discussions. For the sake of conciseness, I did not write an exhaustive list but everyone’s help was much appreciated. Thanks again to my colleagues, friends and relatives.

# Contents

# 1 Introduction

This thesis is entitled "Conservation Laws for Coding," in reference to general laws of physics. The title is deliberately ambitious. More modestly, the main "conservation law" we present is the so-called general area theorem, and due to technical challenges, we have to phrase some of the key observations as conjectures. The title, however, is supposed to reflect our underlying aim: to sketch fundamental principles that govern modern iterative coding, as well as many other physical phenomena.

Towards this goal we first investigate a one-dimensional measure of the decoding performance that is known as the *EXIT function*. Many other alternative measures of the decoder performance have been suggested in the literature. To name but a few, the expected value, the standard deviation of the densities, or the minimum-mean square error are useful alternatives. However – in spite of the pun – EXIT curves (and further GEXIT curves) are the true "entry point" to uncover the strong relationship between *belief propagation* and *maximum a posteriori* decoding.

*Belief propagation (BP)* is the "locally optimum" message-passing algorithm. Given a binary memoryless symmetric channel with Shannon capacity $C$, it is conjectured in [1] that there is a sequence of sparse graph codes such that, for any transmission rate $r = (1-\delta)C$ and any target bit error probability, the decoding complexity, in operations per bit, is of order $O(\frac{1}{\delta}\log\frac{1}{\delta})$ (the encoding complexity is of order $O(\log\frac{1}{\delta})$ as a result of the graph density). Because of this low complexity and its iterative nature, BP decoding on sparse graphs is considered *practical*.

*Maximum a posteriori (MAP)* decoding (which in the case of equal priors is equivalent to maximum likelihood decoding) is an optimal decoding rule in the sense that it minimizes the error probability (see, e.g., [2]). For general codes, however, the complexity is very high (more precisely, the decoding is NP complete). MAP decoding is therefore considered *ideal*.

The focus of this thesis is the relationship between MAP and BP decoding. The key insight is that the bridge that connects the two can be seen as the translation of the *Maxwell construction* into the field of probabilistic coding. This construction uses (G)EXIT curves.

The Maxwell construction plays a central role in the theory of phase transitions. Hence we review it in Section 1.1 of this introduction.

In Section 1.2, we translate the Maxwell construction to the setting of sparse graph codes and belief propagation. This is the main message of this work.

The thesis outline follows in Section 1.3.

We chose to present our work from the point of view of the relationship between BP and MAP decoding, which we hope will help to make it more accessible. As an alternative choice, we could have presented (G)EXIT functions on their own and listed potential applications. GEXIT functions are one-dimensional transfer functions. Their (perhaps) most remarkable application is the Maxwell construction. However, many other applications are possible. For example, we will see that GEXIT functions are the somewhat "true" measure for the decoding progress: In order for the rate of an ensemble of codes to approach the capacity under BP decoding, the GEXIT functions of the component codes have to be matched perfectly. Prior work related to this thesis is listed in Section 1.4.

## 1.1 Maxwell Construction in Thermodynamics

Thermodynamics and statistical physics study properties of physical systems. Thermodynamics is concerned with the macroscopic behavior of a system. It historically precedes statistical physics that is based on microscopic considerations. At a microscopic level, a system is described by a very large number of variables such as the position, the speed, or the magnetic moment (spin) of each particle. The evolution of the system is then explained by the laws of dynamics (Newton's law). At a macroscopic level, a system is characterized by a small set of variables that describe the *state* of the system: For example, in the case of a fluid, they are the pressure, the temperature, or the energy. Such macroscopic quantities provide a sufficiently precise description of the systems in many cases. They are particularly helpful because the complete microscopic dynamical description of the system turns out to be, in general, intractable.

Let us focus on the classical case of the compression of a fluid in a container. The pressure, the volume and the temperature are state variables that are linked to each other. Consider for example an *ideal gas* (more precisely, an ideal fluid) that satisfies the law $p \cdot V = NRT$, where $p$ is the pressure (in Pa), $V$ is the volume (in $m^3$), $T$ is the absolute temperature (in K), $N$ is the number of moles, and $R$ is the *gas constant* ($R \approx 8.314510 \mathrm{JK}^{-1}\mathrm{mol}^{-1}$). Recall that $p \cdot V$ represents *work* (in J) or *energy*. Assume that we have a fixed $T$ (see the corresponding *isotherm* in Figure 1.1) and a fixed number of particles (atoms or molecules). We aim at describing how the system evolves when the volume decreases. From the previous law we see that by reducing the volume $V$ of the container, we increase the pressure $p$. The ideal gas law is obtained by assuming that the particles have negligible sizes and that they do not interact with each other. Clearly, one can not compress a gas indefinitely, and the ideal gas approximation is only valid in the limit of a low-density gas.

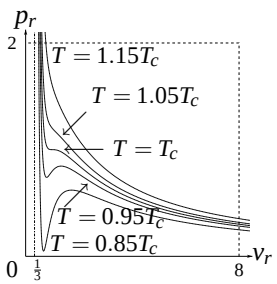



Figure 1.1: Van der Waals isotherms in reduced[1] coordinates.

The *Van der Waals equation of state* [3] provides an alternative model that is closer to the real behavior. A first refinement takes into account the limits of the compression imposed by the physical size of the particles. It assumes that molecules are not dimensionless points but have a total volume equal to $Nb$. Therefore, the free space that the system can offer to the molecules reduces to $V - Nb$. The second refinement captures the effects of the pairwise attractive force between particles. This causes the average free energy $Nf$ to be reduced by an amount proportional to the fluid density $\frac{N}{V}$. Since the pressure obeys the thermodynamic relationship $p = -\frac{\partial f}{\partial (V/N)}$, it is therefore reduced by an amount proportional to $\frac{N^2}{V^2}$. The equation of state therefore reads $p = \frac{NRT}{V-Nb} - a\frac{N^2}{V^2}$ (which can be viewed as a second order approximation), or $\left(p + a\frac{N^2}{V^2}\right)(V - Nb) = NRT$, where the non-negative constants $a$ and $b$ characterize the considered fluid. Figure 1.1 depicts typical isotherms.[1]

Let us describe what happens experimentally for the case of the liquid-gas transformation of water. If a small amount of liquid is placed in a completely empty (and hermetically closed) large container at room temperature, it evaporates. The vapor exerts pressure on the walls of the container. Figure 1.2 (left) depicts an *experimental* observation of the system behavior. By gradually reducing the volume of the container, we increase the vapor pressure until it reaches a *critical* value $p^c$. At this point the vapor condenses into water and the pressure stays constant throughout this transformation. When there is no vapor left, the pressure starts to rise again (very quickly since it is difficult to compress water).

[1] Let $v$ denote the volume divided by the numbers of particles, the Van der Waals isotherm is equivalently described by the equation $(p + \frac{a}{v^2})(v - b) = kT$ where $k = R/N_A \approx 1.380658 \cdot 10^{-23}\mathrm{JK}^{-1}$ is the Boltzmann constant ($N_A \approx 6.0221367 \cdot 10^{23}$ $\mathrm{mol}^{-1}$ being the *Avogadro number*). Although the constants $a$ and $b$ change from fluid to fluid, this equation can be recast into an invariant form (which applies to any fluid). Critical values of $p$, $v$ or $T$ are obtained at the *critical point* that separates domains where the system behavior is different. On a diagram representing $p$ versus $v$ as in Figure 1.1, this critical point is an inflexion point such that $\frac{\partial^j p}{\partial v^j}|_c = 0$ for $j = 1, 2$. This yields to $p^c = \frac{a}{27b^2}$, $v^c = 3b$ and $T_c = \frac{8a}{27bR}$. Define the *reduced* variables $p_r \triangleq \frac{p}{p^c}$, $v_r \triangleq \frac{v}{v^c}$ and $T_r \triangleq \frac{T}{T_c}$, then the Van der Waals equation is recast in the *reduced* (invariant) form $(p_r + \frac{3}{v_r^2})(v_r - \frac{1}{3}) = \frac{8}{3}T_r$.

In many *theoretical* descriptions of this phenomenon, such as the Van der Waals model for $T < T_c$, a non-monotonic function $p(V)$ is obtained. See Figure 1.2 (right). The *Maxwell construction* [4] allows us to modify the "unphysical" part of this theoretical function $p(V)$ in order to obtain a consistent behavior of the system: The two decreasing branches of $p(V)$ are joined by a constant-pressure line as observed in experiments. At which height should the horizontal line $p = p^c$ corresponding to the phase transition be placed? The basic idea of the Maxwell construction is that, at the critical pressure $p^c$, the vapor and the liquid are in *equilibrium*: the rates of the forward (vapor into liquid) and reverse transformations (liquid into vapor) are equal, therefore infinitesimal quantities of vapor can be transformed into liquid – and vice versa – without any *work* being performed on the system. This reversible transformation implies that when we compress the fluid in the container, the vapor begins its transformation into liquid at $p^c$.

Formally, the Gibbs free energy is a macroscopic quantity that indicates the total work performed on the system. The Gibbs free energy $G$ of the system is constant during the liquefaction process because of the phase equilibrium. It is known that the Gibbs free energy is equal to $G = \mathrm{d}(pV) - p\mathrm{d}V$ for a fixed amount of fluid (in which two pure phases of same chemical potential coexist).
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. Integrating between $A$ and $B$ (right picture in Figure 1.2), we get $0 = G_B - G_A = p^c(V_B - V_A) - \int_A^B p(V)\mathrm{d}V$. In words, this shows that the above equilibrium condition implies the equality of the areas enclosed between the horizontal line and the original non-monotonic Van der Waals curve $p(V)$. See, e.g., [5–8].

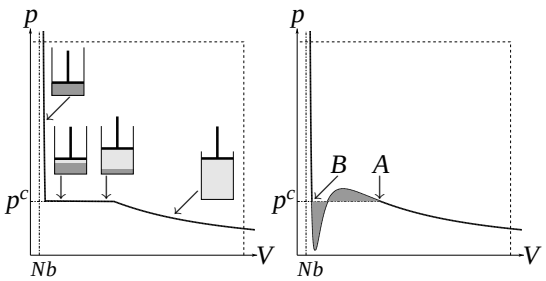



Figure 1.2: Maxwell construction in thermodynamics. Left: Pressure-volume diagram for the liquid-vapor phase transition and corresponding modeled containers. Right: Van der Waals isotherm and the Maxwell construction.

In the setting of iterative coding, the global variables that play the role of $V$ and $p$ are the intrinsic and extrinsic *symbol entropies* (or measures derived from the symbol information). This thesis will show that one can derive a global conservation law on the conditional *word entropy*. This law is similar in essence to the previous conservation of the Gibbs free energy. The Maxwell construction will eventually allow us to determine the performance curve under MAP decoding (which is equivalent to the physical system behavior) from the one under BP decoding (linked to the theoretical Van der Waals equation of state).

## 1.2 Maxwell Construction in Coding

In practice the performance of a communication scheme is often assessed by plotting the "bit (or word) error rate" versus a measure of the channel quality. For the binary erasure channel this means that we plot the bit erasure probability $\frac{1}{n}\sum_{i=1}^{n}\Pr\{\hat{x}_i^{\mathrm{DEC}}(\epsilon) = \text{❋}\}$ obtained at the output of a given decoder as a function of the channel erasure probability $\epsilon$ (here $\hat{x}_i^{\mathrm{DEC}}$ denotes the estimate of the $i^{\mathrm{th}}$ bit). As an illustration, the performance of LDPC codes under BP decoding [9–11] is depicted in Figure 1.3 using a non-logarithmic scale. The x-axis depicts the channel erasure probability, i.e., a measure of the channel noise and the y-axis depicts the erasure probability under BP decoding. The result for several blocklengths $n$, $n \in \{100, 250, 500, 1000, 2500, 5000, 10000, 50000, 100000\}$, is shown. Observe that, when the blocklength becomes very large, the bit erasure performance converges to an *asymptotic* curve. This curve is zero below a certain value of the noise (called *BP threshold* and denoted by $\epsilon^{\mathrm{BP}}$ in Figure 1.3), then "jumps" to some non-zero value and finally continues smoothly until it reaches one.

Let us describe more precisely the typical behavior under BP decoding. Figure 1.3 shows the *average* BP performance curves $h^{\mathrm{BP}}(\epsilon)$ obtained from Monte Carlo simulations. Formally, $h^{\mathrm{BP}}(\epsilon) \triangleq \frac{1}{\epsilon}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Pr\{\hat{x}_i(\epsilon) = \text{❋}\}\right]$, where the expectation is taken over elements chosen uniformly at random

from the ensemble characterized by a fixed degree distribution pair (dd pair) and a fixed blocklength $n$.

A few general comments are in order. First, the performance of particular instances of codes concentrates around the average performance, which makes it meaningful to analyze the average. See [12–15].

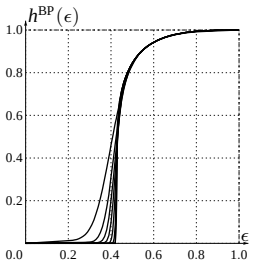



Figure 1.3: Probability of bit erasure for LDPC codes with dd pair $(\frac{x+4x^3}{5}, \frac{x^3+4x^7}{5})$ (edge perspective) and $n = 100, \cdots, 10^6$ on the binary erasure channel.

For a fixed length $n$, the average curve can be analytically predicted as shown, e.g., in [12, 16, 17], where the number of *stopping sets* (residual "cores" in which the iterative decoder gets stuck) is analyzed.

When the length $n$ becomes large, the average performance of an ensemble of sparse graph codes is given by the performance of the corresponding infinite tree or *computation tree*. Density evolution on the computation tree permits us to predict the complete asymptotic performance curve under BP decoding. In particular, density evolution determines the BP threshold associated with the considered ensemble of sparse graphs: The BP threshold is $\epsilon^{\mathrm{BP}} \approx 0.4273$ in the example of Figure 1.3. Operationally, this means that transmission at a vanishing erasure probability is asymptotically guaranteed to succeed with high probability if and only if it takes place over the binary erasure channel with parameter $\epsilon < \epsilon^{\mathrm{BP}}$. As has been observed for phase transitions in many other physical systems, this threshold also acts as a fundamental quantity to describe the finite-length performance of the ensemble. It is indeed possible (at least on the binary erasure channel) to think of the BP threshold as the zero order term in a Taylor series so that a scaling law represents the first order term. See [18–20]. More precisely, we write the bit erasure probability as $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\Pr\{\hat{x}_i(\epsilon) = \text{❋}\}\right] = \nu^{\mathrm{BP}} Q(\frac{\sqrt{n}(\epsilon^{\mathrm{BP}}-\epsilon)}{\alpha_{\lambda,\rho}})(1+o_n(1))$, where $Q(u) \triangleq \int_u^{+\infty} \mathrm{e}^{-\frac{u^2}{2}}\mathrm{d}u$. This means that, when the blocklength increases, the bit error/erasure performance in the so-called "waterfall" region is given by the previous first order scaling.

So far, we have only discussed the case of BP decoding. This is of course the most interesting decoding for practical implementations and the interest in the maximum a posteriori (MAP) decoding is mainly theoretical. In practice, MAP decoding requires an exponential (typically prohibitive) amount of computational resources. Nevertheless, the hope is that a better understanding of this type of decoding will give valuable hints for the design of sparse graph codes, e.g., on complexity or capacity-approaching issues. It is known, see, e.g., [21, 22], that the MAP performance of a sequence of codes with an increasing minimum distance is also characterized by a threshold phenomena. Similar properties such as the ones concerning the BP threshold are expected to hold for the MAP threshold. The MAP analysis for sparse graphs has been less investigated than its BP counterpart.

To date there are basically two types of analysis that are employed for the MAP decoding. On the one hand, a large body of literature concerns bounds on the MAP threshold via bounds on the weight distribution or on the parity-check matrix density, see [23, 24]. Although, in general, these bounds are not expected to be tight. On the other hand, MAP thresholds have been determined via the *replica method*, see [25–29], but the method itself is not completely rigorous. Lately some of these bounds were converted into rigorous bounds via an *interpolation* method, see [30, 31]. The fact that such results come from the field of statistical mechanics is not surprising since the MAP threshold corresponds to the *physical critical* point described in the previous section. The MAP threshold is therefore a more "natural" threshold than its BP counterpart (which is called the *dynamical* threshold) from a thermodynamical standpoint.

For sparse graph codes, we will demonstrate that a Maxwell-type construction holds: It connects the performance curve under BP decoding to the one under MAP decoding in the asymptotic setting of increasing blocklengths. The curve that plays the role of the Van der Waals curve is the *EBP GEXIT* curve. The EBP GEXIT curve is determined in a purely theoretical fashion: It is given by the set of all fixed points of density evolution. Note that some of these fixed points are *unstable* and some are "hidden" so that they cannot be reached in practice by the BP decoder. The part of the curve which corresponds to unstable and "hidden" fixed points extends the (operationally reachable) BP curve. The complete curve is therefore called extended BP or EBP GEXIT curve. This is a smooth curve that is depicted in Figure 1.4. In this example the "spurious" branch (dashed part of the curve) corresponds

to *unstable* fixed points of density evolution. Generally, the EBP GEXIT curve is a "non-physical" description of the system as is the case for the Van der Waals curve in the setting of thermodynamics.

The asymptotic BP performance curve (called BP GEXIT curve) is found to be the *envelope* of the EBP GEXIT curve. The transition given by the Maxwell construction on the EBP GEXIT curve is located exactly at the MAP threshold (for the considered example). Furthermore, below the BP threshold and above the MAP threshold, MAP and BP decoding coincide. To summarize, the MAP performance (or GEXIT) curve is zero below the MAP threshold, then jumps to some value, and from that point on it coincides with the BP performance curve. It then continues smoothly until it reaches one (as does the BP GEXIT curve).

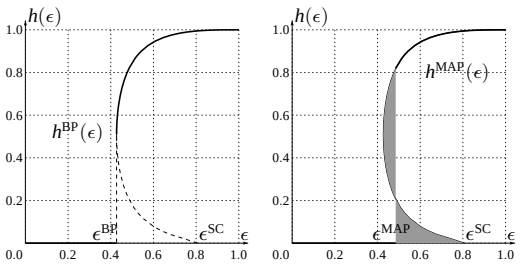



Figure 1.4: Asymptotic performance of the LDPC ensemble with dd pair $(\lambda(\mathtt{x}),\rho(\mathtt{x})) = (\frac{\mathtt{x}+4\mathtt{x}^3}{5}, \frac{\mathtt{x}^3+4\mathtt{x}^7}{5})$. It has design rate $r_{\lambda,\rho} \triangleq 1-(\int\rho)/(\int\lambda) = 1/2$, Shannon threshold $\epsilon^{\mathrm{SH}} \triangleq 1 - r_{\lambda,\rho} = 0.5$ and stability condition threshold $\epsilon^{\mathrm{SC}} \triangleq \frac{1}{\lambda'(0)\rho'(1)} \approx 0.8065$. Left: BP decoding with $\epsilon^{\mathrm{BP}} \approx 0.4273$. Right: MAP decoding with $\epsilon^{\mathrm{MAP}} \approx 0.4821$.

The MAP GEXIT curve corresponds to the true monotonic relationship between pressure and volume in thermodynamics whereas the theoretical EBP GEXIT curve corresponds to the Van der Waals model. This curiosity is explored in this thesis.

## 1.3 Thesis Outline

The chapters are relatively independent. The main material is contained in Chapter 3, Chapter 4, Chapter 5, and Chapter 6: Chapters 3 and 4 introduce our formalism and present the case of the binary erasure channel (BEC). Chapters 5 and 6 extend the concepts to general memoryless channels.

**Chapter 2** Following Shannon's communication paradigm, we settle notations, revisit and redefine some fundamental notions. First, standard channel models are described and the entropy operator is introduced. Some elements of statistics and some natural estimators are recalled. We then describe the BP and the MAP decoding algorithms. Notions of asymptotic rate, the order implied by physical degradation, thresholds, and families of channels are further discussed.

**Chapter 3** EXIT functions and the asymptotic analysis of iterative coding systems via density evolution are reviewed. We see that, when applied on the BEC, EXIT functions exhibit various interesting properties. In particular, a first generalization of the area theorem of [32] is stated.

Although these two chapters consist mainly of a review of known statements, they also introduce a slightly novel viewpoint: For example, this concerns the asymptotic rate of LDPC ensembles as discussed in Chapter 2, or the conservation law presented in Chapter 3, which generalizes the original area theorem.

**Chapter 4** We present the connection between MAP and BP decoding for the case of the erasure channel. The area theorem implies a simple upper bound on the MAP threshold based on the BP EXIT function. In many cases, we are able to prove the tightness of this bound. The Maxwell construction is interpreted as an exchange of information during the decoding process. The Maxwell construction has an operational meaning, which is given by the so-called Maxwell decoder. This decoder performs MAP decoding and shows how complex it is to transform a BP decoder into a MAP decoder.

For the BEC, EXIT functions suffice for a detailed analysis of the Maxwell construction. But, in general, one needs to *define* new functions that we call GEXIT functions and that extend our field of investigation.

**Chapter 5** We define GEXIT functions and investigate their properties. The presentation follows in lock-step with the presentation of EXIT functions on the BEC. GEXIT functions over general binary-input memoryless output-symmetric (BMS) channels and EXIT functions over the BEC share almost all their properties. Analog to EXIT charts for the BEC, GEXIT charts permit us to derive a matching condition for general BMS channels.

**Chapter 6** Using the general area theorem with GEXIT functions, we extend the upper bound on the MAP threshold to general BMS channels. As for the BEC, an area theorem associated to the EBP GEXIT curve is derived. We conjecture that the Maxwell construction carries over to general BMS channels. Partial results are provided and numerical evidence is shown.

Many properties known for EXIT functions on the BEC extend to general memoryless symmetric channels using GEXIT functions. For example, the matching condition is derived for general BMS channels and the Maxwell construction is expected to hold to a large extent.

**Chapter 7** Our concepts apply to graphs that have the required sparseness property and are expected to hold in a much wider setting. Further examples are discussed, and, in particular, the historical example of Turbo codes for which an exact derivation is presented in the BEC case.

Promising and challenging tasks for future research and applications in the context of coding include code optimization and complexity study. Other possible extensions of this work concern general (e.g., combinatorial) search problems.

## 1.4 Related Work

GEXIT functions are similar in many respects to EXIT functions introduced by ten Brink [33]. More specifically, GEXIT functions coincide with EXIT functions on the erasure channel. The area theorem we introduce is a generalization of the area theorem by Ashikhmin, Kramer, and ten Brink in [32] (in fact similar notions are found earlier in the work by Shokrollahi *et al.* [12, 34, 35]). This area theorem, when applied to the erasure channel, leads back to the notion of EXIT functions. The upper bound on the MAP threshold, which we originally presented in [36], has been extended to general channels with the help of GEXIT functions. For the erasure case, we then show that in many cases the upper bound on the MAP threshold is tight by strengthening the counting argument of [37]. Notice that a similar technique is used by Mézard *et al.* in [38] for the "XORSAT" problem. Over general channels, we define the GEXIT function as the derivative of the (normalized) conditional entropy with respect to some measure of the noise in the channel. In [39,40] Guo, Shamai and Verdú showed that for Gaussian channels, the derivative (with respect of the signal-to-noise ratio) of the mutual information is equal to the minimum mean square error (MMSE), and in [39] they showed that a similar relationship holds for Poisson channels. One can think of GEXIT functions as providing such a relationship in a more general setting (where the generalization is with respect to the admissible channel families). For some channel families, GEXIT functions have a particularly nice interpretation. For Gaussian channels, the interpretation in terms of the MMSE detector can be simplified even further: It can be seen as the "magnetization" of the system as shown by Macris in [41]. Gaussian channels are further investigated by Zakai in [42]. The results in [43], which have appeared since the introduction of GEXIT functions in [44], can be reformulated to give an interpretation of GEXIT functions for the class of additive channels. Finally note that, inspired by Tüchler, ten Brink and Hagenauer [45] and based on the result of Guo *et al.*, Bhattad and Narayanan introduce MMSE charts in [46] using a Gaussian approximation. This corresponds to GEXIT charts under the Gaussian hypothesis in our framework.[2]

[2]Partial results of our work have been communicated in [36, 44, 47–51] and parts have been submitted for publication in [52, 53].

**Overview:** Some key notions of communications and coding are revisited, in particular the notions of statistics, estimators, design rate, and threshold. For a more detailed introduction into these concepts we refer the reader to [2, 10–15, 54–66].

# 2 | Preliminaries

Let $\mathcal{X}$ denote the channel input alphabet and $\mathcal{Y}$ the channel output alphabet. Without loss of generality, we assume that the distributions encountered all along this thesis admit a probability density function.[1] We then write all general statements in terms of densities, the translation to the discrete case being immediate with the use of Dirac delta distributions. The conditional density $p_{Y|X}(y|x)$ denotes the channel model with random input $X$ and output $Y$; this includes discrete channel models. Let the lower case letter $x \in \mathcal{X}$ denote a deterministic value taken by a random $X$ with probability $p_X(x)$. A vector (or matrix), let us say $X$, will also be denoted by $X_{[n]}$, where $[n] \triangleq \{1, \cdots, n\}$ is the index set of its columns, $n$ being its length. In a similar way, if $\mathcal{S} \subseteq [n]$, then $X_{\mathcal{S}}$ is the sub-vector formed by the columns of $X$ indexed by $\mathcal{S}$, e.g., $X_{\{1,4\}} = (X_1, X_4)$. By a slight abuse of notation, the $i^{\text{th}}$ component of $X$ is simply denoted by $X_i \triangleq X_{\{i\}}$, and $X_{\sim i} \triangleq X_{[n]\setminus\{i\}}$ when a single bit is omitted, following the factor graph terminology, see [58–60, 67].

## 2.1 Channel Model

Recall that a channel model $p_{Y|X}$ is said to be binary if its input alphabet is binary, i.e., if $|\mathcal{X}| = 2$. For simplicity, this thesis deals mainly with binary channels. Without loss of generality, we choose the binary alphabet $\mathcal{X} = \{-1, +1\}$ (standard bipolar or Binary Phase-Shift Keying (BPSK) modulation: $\mathtt{0} \leftrightarrow +1$ and $\mathtt{1} \leftrightarrow +1$).

**Example 2.1** [BEC($\epsilon$)] Figure 2.1 (left) depicts the *Binary Erasure Channel* (BEC) model with parameter $\epsilon$, call it BEC($\epsilon$). The input $X$ takes value $x \in \mathcal{X} = \{-1, +1\}$ and the output $Y$ takes value $y \in \mathcal{Y} = \{-1, \text{❋}, +1\}$ where ❋ is the *erasure symbol*. The transition probabilities are discrete and given by $p_{Y|X}(y|x) = 1 - \epsilon$ if $y = x$, $\epsilon$ if $y =$ ❋, and 0 otherwise.

**Example 2.2** [BSC($\epsilon$)] Figure 2.1 (middle) depicts the *Binary Symmetric Channel* (BSC) model with parameter $\epsilon$, call it BSC($\epsilon$). The input value $x$, as well as the output value $y$, is an element of $\mathcal{X} =$

[1] We restrict ourselves to the case of channels without feedback. Results in this thesis are written in the language of densities for notational simplicity. However, they can be stated in the more general context of distributions as discussed in [14, 15, 65]. All our results translate in a straightforward manner to this context. It suffices to adopt the convention of formally denoting channels by their transition density even when such a density does not exist, and write $\int f(y) p_{Y|X}(y|x) \mathrm{d}y$ as a proxy for the corresponding expectation whenever $\mathbb{E}[f(Y)|X = x]$ exists (e.g., if $f(y)$ is bounded).

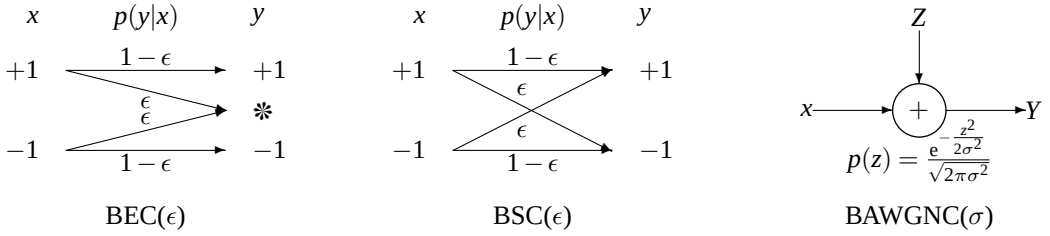



Figure 2.1: Standard binary memoryless channels.

$\mathcal{Y} = \{-1,+1\}$. The transition probabilities are discrete and given by $p_{Y|X}(y|x) = 1-\epsilon$ if $y = x$, and $\epsilon$ otherwise.

**Example 2.3** [BAWGNC($\sigma$)] Figure 2.1 (right) depicts the *Binary Additive White Gaussian Noise Channel* (BAWGNC) model with zero-mean noise of standard deviation $\sigma$, call it BAWGNC($\sigma$). The input value $x$ is an element of $\mathcal{X} = \{-1,+1\}$ and the output value $y \in \mathcal{Y} = \mathbb{R}$. The transition probability function is $p_{Y|X}(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{-\frac{(y-x)^2}{2\sigma^2}}$.

**Definition 2.1** [Memoryless] For $n \in \mathbb{N}\setminus\{0\}$, let $X_{[n]}$ be a random vector with components $X_i$ defined over $\mathcal{X}$, i.e., $X_{[n]}$ takes values in the product alphabet $\mathcal{X}^n$ and, in a similar way, let $Y_{[n]}$ be a random vector taking value in $\mathcal{Y}^n$. The channel family $\{p_{Y_{[n]}|X_{[n]}}\}_n$ is said to be *memoryless* if there is a family of individual channels $\{p_{Y_i|X_i}\}_i$ such that, for all $n$, $\forall (x,y) \in \mathcal{X}^n \times \mathcal{Y}^n$, $p_{Y_{[n]}|X_{[n]}}(y_{[n]}|x_{[n]}) = \prod_{i=1}^n p_{Y_i|X_i}(y_i|x_i)$.

By a slight abuse of notation, if a channel family $\{p_{Y_{[n]}|X_{[n]}}\}_n$ is memoryless, then the family of individual channels $\{p_{Y_i|X_i}\}_i$ will be said to be memoryless. Furthermore, if this family of individual channels is such that $p_{Y_i|X_i} = p_{Y_1|X_1}$ for all $i$, then the channel $p_{Y_1|X_1}$ itself will be said to be *memoryless*. For example, assume $\{p_{Y_i|X_i}\}_i$ describes a family of individual BECs, call it $\{\mathrm{BEC}_i(\epsilon_i)\}_i$. If the channel family is memoryless and such that $\exists\epsilon, \forall i, \epsilon_i = \epsilon$, then we say that BEC($\epsilon$) is memoryless. In a similar manner, BSC($\epsilon$) is said to be memoryless, as well as BAWGNC($\sigma$) (assuming that $\{\mathrm{BAWGNC}_i(\sigma)\}_i$ is such that the $Z_i$'s are independent random variables).
Another particularity of the three simple families of channels discussed above (shared by many other channel families) is that they are parameterized by a single scalar parameter $\mathfrak{p}$. For example $\mathfrak{p} = \sigma$ is the standard deviation for the BAWGNC.

**Definition 2.2** [Channel Symmetry] A binary memoryless channel with real-valued output is said to be *output-symmetric*, in short, *symmetric*, if and only if $p_{Y|X}(y|+1) = p_{Y|X}(-y|-1)$.

It is straightforward to verify that BEC($\epsilon$), BSC($\epsilon$) and BAWGNC($\sigma$) are examples of Binary Memoryless Symmetric Channels (BMSC). By a slight abuse of notation, a generic BMSC (family) with parameter $\mathfrak{p}$ will be called BMSC($\mathfrak{p}$). It is convenient and natural to choose the parametrization $\mathfrak{p}$ so that there is a one-to-one mapping between $\mathfrak{p}$ and the entropy of the channel (see also Section 2.8). In other words, BMSC($\mathfrak{p}$) is a shorthand to denote a given family $\{\mathrm{BMSC}(\mathfrak{p})\}_{\mathfrak{p}\in P}$ where $\mathfrak{p} \in P$ is in one-to-one correspondence with the channel entropy $H(X|Y)$. Channels of the type BMSC($\mathfrak{p}$) will be our main domain of study.

## 2.2 Channel Entropy

Let $X, Y$ be random variables. Assume $X$ is binary with alphabet $\mathcal{X} = \{-1,+1\}$.

**Definition 2.3** [LLR] Consider a binary channel $p_{Y|X}$. The associated *Log-Likelihood Ratio* (LLR) function is the function $y : y \mapsto y(y) \triangleq \log \frac{p_{Y|X}(y|+1)}{p_{Y|X}(y|-1)}$ taking values in $\overline{\mathbb{R}}$ with $y(y) \triangleq \pm\infty$ if $p_{Y|X}(y|\mp 1) = 0$. The random LLR associated with $Y$ is denoted by $Y \triangleq y(Y)$.

A value $y(y)$ (once the channel output $y$ has been post-processed through $y$) is called a channel output value in the $L$-domain. We will later state (in Section 2.4) that, for a binary memoryless channel, the post-processing on the $y$s does not cause information loss.

**Definition 2.4** [Symmetry of Density] Let $\mathsf{a}$ be a probability density function defined over $\mathbb{R}$. The density $\mathsf{a}$ is said to be *symmetric* if $\mathsf{a}(-y) = \mathrm{e}^{-y}\mathsf{a}(y)$ for $y \in \mathbb{R}$.

**Fact 2.1** [Symmetry of $L$-Density] Consider a binary symmetric channel $p_{Y|X}$ and define $Z \triangleq y(Y)$ (i.e, the channel input is transformed into a LLR). Define $\mathsf{a}(z)$ to be the density of $Z$ given $X = +1$. Then $\mathsf{a}(z)$ is symmetric.

*Proof.* For a given LLR value $z \in \overline{\mathbb{R}}$, consider the set $S_z \triangleq \{y : y(y) = z\}$. For notational simplicity, assume that $S_z$ is discrete and that $\alpha(y) \triangleq 1/\left|\frac{p'_{Y|X}(y|+1)}{p_{Y|X}(y|+1)} - \frac{p'_{Y|X}(y|-1)}{p_{Y|X}(y|-1)}\right|$ is well-defined so that, in the language of densities (see [68]), we can write $\mathsf{a}(z) = \sum_{y\in S_z}\alpha(y)p_{Y|X}(y|+1) \overset{(a)}{=} \sum_{y\in S_z}\alpha(y)\mathrm{e}^{y(y)}p_{Y|X}(y|-1) \overset{(b)}{=} \sum_{y\in S_z}\alpha(y)\mathrm{e}^{z}p_{Y|X}(y|-1) \overset{(c)}{=} \mathrm{e}^{z}\sum_{y\in S_z}\alpha(y)p_{Y|X}(-y|+1)$, where $(a)$ uses the definition of $y$, $(b)$ uses the definition of $S_z$ and $(c)$ uses the channel symmetry. Now, observe that $y(-y) = \log\frac{p_{Y|X}(y|-1)}{p_{Y|X}(y|+1)} = -y(y)$ by channel symmetry, therefore the change of variable $y \to -y$ implies $S_z \to S_{-z}$. Moreover, the channel symmetry shows that $\alpha(y) = \alpha(-y)$. It follows that $\sum_{y\in S_z}\alpha(y)p_{Y|X}(-y|+1) = \sum_{y\in S_{-z}}\alpha(y)p_{Y|X}(y|+1) = \mathsf{a}(-z)$. □

**Lemma 2.1** Consider a binary symmetric channel channel $p_{Y|X}$. If $X$ has uniform priors $p_X(x) = 1/2$ ($x = \pm 1$), then $H(X|Y) = \mathbb{E}_{Y|X=+1}[\log_2(1+\mathrm{e}^{-y(Y)})]$.

*Proof.* Since $p_X(x) = \frac{1}{2}$, we use the channel symmetry and the Bayes rule to write
$H(X|Y) = -\int p_{Y|X}(y|+1)\log_2\frac{p_{Y|X}(y|+1)}{p_{Y|X}(y|+1)+p_{Y|X}(y|-1)}\mathrm{d}y = \int p_{Y|X}(y|+1)\log_2(1+\mathrm{e}^{-y(y)})\mathrm{d}y.$ □

**Definition 2.5** [Entropy Operator] Consider a symmetric density $\mathsf{a}$ defined over $\overline{\mathbb{R}}$. The operator $\mathsf{a} \mapsto \mathtt{H}(\mathsf{a}) \triangleq \int_{-\infty}^{+\infty}\mathsf{a}(y)\log_2(1+\mathrm{e}^{-y})\mathrm{d}y$ is called *entropy operator* in the $L-$domain.

Assume that the binary random variable $X$ with $p_X(\pm 1) = 1/2$ is passed through a BMSC and then through the LLR function. The following examples compute the conditional entropy of the resulting channels.

**Example 2.4** [Entropy – BEC($\epsilon$)] With $\mathsf{a}(y) \triangleq p_{Y|X}(y|+1) = \epsilon\cdot\delta_0(y) + (1-\epsilon)\cdot\delta_{+\infty}(y)$, we have $H(X|Y) = \epsilon$.

**Example 2.5** [Entropy – BSC($\epsilon$)] With $\mathsf{a}(y) \triangleq p_{Y|X}(y|+1) = \epsilon\cdot\delta_{-\log\frac{1-\epsilon}{\epsilon}}(y) + (1-\epsilon)\cdot\delta_{+\log\frac{\epsilon}{1-\epsilon}}(y)$, we have $H(X|Y) = h_2(\epsilon)$.

**Example 2.6** [Entropy – BAWGN($\sigma$)] The LLRs $y(y) = \frac{2}{\sigma^2}y$ have density $\mathsf{a}(y) \triangleq \frac{\sigma}{\sqrt{8\pi}}\mathrm{e}^{-\frac{(y\sigma^2-2)^2}{8\sigma^2}}$. Unfortunately $H(X|Y) = \mathtt{H}(\mathsf{a})$ can only be expressed in terms of an integral that one has to compute numerically.

Of course, the previous values coincide with the well-known corresponding entropies without post-processing. In the following sections, we will indeed see that the channel post-processing $y$ does not affect the channel entropy: More formally, $y(Y)$ constitutes a *sufficient statistic* for *estimating* $X$ (see Fact 2.6).

## 2.3 Statistics and Estimators

Let $X$ be a random vector. Consider the family $\{p_{Y|X=x}(y)\}_x$ where the random vector $Y$ represents the *observed sample*. For any function $\phi(y)$, the random vector or variable $\phi(Y)$ is called *statistic*. When the statistic $\phi(Y)$ is used to estimate some unobservable quantity (for example $X$ by choosing $\phi$ to be the minimum mean-square estimator[2]), then the statistic $\phi(Y)$ is called an *estimator*.
Estimators or statistics are fairly general notions. More subsequent definitions show some estimators that are common and useful in coding. Definition 2.6 and Definition 2.7 assume that transmission takes place over a channel with input vector $X$ and output vector $Y$.

**Definition 2.6** [Maximum-Likelihood Decision Rule] For a fixed vector $y$, the quantity[3] $\hat{x}_i^{\mathrm{ML}}(y) \triangleq \operatorname{argmax}_\xi(p_{Y|X_i}(y|\xi))$ is called Maximum Likelihood (ML) *decision* (or *hard estimate*) for the $i^{\text{th}}$ symbol.

**Definition 2.7** [Maximum A Posteriori Decision Rule] For a fixed vector $y$, the quantity[3] $\hat{x}_i^{\mathrm{MAP}}(y) \triangleq \operatorname{argmax}_\xi(p_{X_i|Y}(\xi|y))$ is called (bit) Maximum A Posteriori (MAP) *decision* (or *hard estimate*) for the $i^{\text{th}}$ symbol.

The MAP decoding rule (as well as the ML decoding rule in case of equal priors) is known to be an *optimal* decoding rule in the sense that it minimizes the probability of error, see [2]. The following fact is a straightforward implication of the Bayes rule.

**Fact 2.2** [Equivalence between MAP and ML Estimator] If $X_i$ is uniformly distributed over $\mathcal{X}$, then $\hat{x}_i^{\mathrm{ML}}(y) = \hat{x}_i^{\mathrm{MAP}}(y)$ for all $y$.

In this thesis, we deal mainly with binary alphabets $\mathcal{X} = \{-1,+1\}$. The ML and MAP decisions are therefore simply given by the sign of the associated $L$-values (logarithms of ratios) $\hat{y}_i^{\mathrm{ML}}(y) \triangleq \log\frac{p_{Y|X_i}(y|+1)}{p_{Y|X_i}(y|-1)}$ and $\hat{y}_i^{\mathrm{MAP}}(y) \triangleq \log\frac{p_{X_i|Y}(+1|y)}{p_{X_i|Y}(-1|y)}$. Following [63, 71, 72], the $L$-values can be viewed as the $i^{\text{th}}$ ML and MAP *soft estimates* in $\overline{\mathbb{R}}$. MAP and ML estimates are linked by the relationship $\hat{y}_i^{\mathrm{MAP}}(y) = a_i + \hat{y}_i^{\mathrm{ML}}(y)$, where $a_i \triangleq \log\frac{p_{X_i}(+1)}{p_{X_i}(-1)}$ is the $i^{\text{th}}$ *a priori* estimate. More specifically, assume that $Y_i$ and $Y_{\sim i}$ are independent given $X_i$. For example, this hypothesis, written $Y_i \to X_i \to Y_{\sim i}$ in (next) Section 2.4, is satisfied by a memoryless channel.[4] When written in terms of the LLRs of Definition 2.3, the property $p_{Y|X_i}(y|\xi) = p_{Y_i|X_i}(y_i|\xi)p_{Y_{\sim i}|X_i}(y_i|\xi)$ becomes $\hat{y}_i^{\mathrm{ML}}(y) = y_i(y_i) + \phi_i^{\mathrm{ML}}(y_{\sim i})$, where

$$y_i(y_i) \triangleq \log\frac{p_{Y_i|X_i}(y_i|+1)}{p_{Y_i|X_i}(y_i|-1)} \text{ is the } i^{\text{th}} \textit{ intrinsic } \text{estimate},$$
$$\phi_i^{\mathrm{ML}}(y_{\sim i}) \triangleq \log\frac{p_{Y_{\sim i}|X_i}(y_{\sim i}|+1)}{p_{Y_{\sim i}|X_i}(y_{\sim i}|-1)} \text{ is the } i^{\text{th}} \text{ (ML) } \textit{extrinsic } \text{estimate},$$

as introduced in [62, 73, 74]. One could alternatively define the extrinsic MAP estimate $\phi_i^{\mathrm{MAP}}(y_{\sim i}) \triangleq \log\frac{p_{X_i|Y_{\sim i}}(+1|y_{\sim i})}{p_{X_i|Y_{\sim i}}(-1|y_{\sim i})} = a_i + \phi_i^{\mathrm{ML}}(y_{\sim i})$. In case of equal priors, i.e., $a_i = 0$, then $\phi_i^{\mathrm{MAP}}(y_{\sim i}) = \phi_i^{\mathrm{ML}}(y_{\sim i})$ and $\hat{y}_i^{\mathrm{MAP}}(y) = \hat{y}_i^{\mathrm{ML}}$.

## 2.4 Markov Chains and Sufficient Statistics

Let $X, W, V$ be random vectors.

[2] In Chapter 5, we will encounter a quantity called *minimum mean-square error*. Let us review this notion briefly. Further details are available in standard literature, or in [69, 70]. Define $\hat{x}^{\mathrm{MMS}}(Y) \triangleq \mathbb{E}[X|Y]$; it is called minimum mean-square estimator because it is shown to minimize the estimation error in the mean-square sense. The *minimum mean-square error* (MMSE) is defined as $\mathbb{E}[(X - \hat{x}^{\mathrm{MMS}}(Y))^2]$. By definition of the conditional expectation, $\mathbb{E}[(X - \hat{x}^{\mathrm{MMS}}(Y))^2] = \mathbb{E}[\mathbb{E}[X^2|Y] - \mathbb{E}[X|Y]^2]$.

[3] By convention, if this maximum is not unique, we define the hard estimate to be equal to the erasure symbol ❋.

[4] Indeed, if the channel is memoryless (and discrete for simplicity), then $p(y|x_i) = \sum_{\sim x_i} p(y, x_{\sim i}|x_i) = \sum_{x_{\sim i}} p(x_{\sim i}|x_i)p(y|x) = p(y_i|x_i)\sum_{x_{\sim i}} p(x_{\sim i}|x_i)p(y_{\sim i}|x_{\sim i}) = p(y_i|x_i)\sum_{x_{\sim i}} p(x_{\sim i}|x_i)p(y_{\sim i}|x_{\sim i}, x_i) = p(y_i|x_i)\sum_{x_{\sim i}} p(y_{\sim i}, x_{\sim i}|x_i) = p(y_i|x_i)p(y_{\sim i}|x_i)$.

**Definition 2.8** [Markov Chain] $X, W, V$ are said to form a *Markov chain* if $X$ and $V$ are conditionally independent given $W$. This relationship is denoted by $X \to W \to V$.

The next fact gives some alternate characterizations of a Markov chain.

**Fact 2.3** [Various Characterizations] Assume that the joint probability density function $p_{X,W,V}(x,w,v)$ exists. $X \to W \to V$ is equivalent to the following:

(i) $V \to W \to X$ (ii) $p_{X,V|W}(x,v|w) = p_{X|W}(x,w)p_{V|W}(v|w)$

(iii) $p_{X,W,V}(x,w,v) = p_X(x)p_{W|X}(w,x)p_{V|W}(v,w)$ (iv) $p_{V|W,X}(v|w,x) = p_{V|W}(v|w)$.

Clearly, for any function $\phi$, if $V = \phi(W)$, then $X \to W \to V$. The next example illustrates an important special instance of this fact.

**Example 2.7** With the conventions of Section 2.3, $X_i \to Y_{\sim i} \to \Phi_i^{\mathrm{ML}}$ since $\Phi_i^{\mathrm{ML}} = \phi_i^{\mathrm{ML}}(Y_{\sim i})$. Therefore $\Phi_i^{\mathrm{ML}} \to Y_{\sim i} \to X_i$. Of course, the same is true for $\Phi_i^{\mathrm{MAP}}$.

**Definition 2.9** [Sufficient Statistic] Let $X$ and $Y$ be two random vectors. A function $\phi(Y)$ is said to be a *sufficient statistic* relative to $\{p_{Y|X=x}(y)\}_x$ (or, short, a *sufficient statistic* for estimating $X$) if and only if $X \to \phi(Y) \to Y$.

The following examples play a central role in the remainder of the thesis.

**Fact 2.4** [Extrinsic MAP Estimate as Sufficient Statistic] Assume $Y_i \to X_i \to Y_{\sim i}$, using the conventions of Section 2.3, and let $\Phi_i^{\mathrm{MAP}} \triangleq \phi_i^{\mathrm{MAP}}(Y_{\sim i})$ be the (extrinsic) MAP estimator. Then $\Phi_i^{\mathrm{MAP}}$ is a sufficient statistic for estimating $X_i$ (given $Y_{\sim i}$), i.e., $X_i \to \Phi_i^{\mathrm{MAP}} \to Y_{\sim i}$.

*Proof.* Assume that we are given $z \in \overline{\mathbb{R}}$. Consider a vector $y_{\sim i}$ such that $z = \phi_i^{\mathrm{MAP}}(y_{\sim i})$ and let $S_z \triangleq \{y'_{\sim i} : \phi_i^{\mathrm{MAP}}(y'_{\sim i}) = z\}$ denote the set of all such vectors.

First, note that $\Phi_i^{\mathrm{MAP}} = \phi_i^{\mathrm{MAP}}(Y_{\sim i})$ so that $p_{X_i|\Phi_i^{\mathrm{MAP}},Y_{\sim i}}(x_i|z,y_{\sim i}) = p_{X_i|Y_{\sim i}}(x_i|y_{\sim i}) \overset{(a)}{=} \frac{(1-x_i)+(1+x_i)\mathrm{e}^z}{2(1+\mathrm{e}^z)}$.

Second, for notational simplicity, assume that $S_z$ is finite and that there exists a well-defined family $\{\alpha(y'_{\sim i})\}$ with $\mathcal{V}'_z \triangleq \sum_{y'_{\sim i} \in S_z} p_{Y_{\sim i}}(y'_{\sim i})\alpha(y'_{\sim i})$ so that, in the language of densities, we can write $p_{\Phi_i^{\mathrm{MAP}}|X_i}(z|x_i) = \sum_{y'_{\sim i} \in S_z} p_{Y_{\sim i}|X_i}(y'_{\sim i}|x_i)\alpha(y'_{\sim i}) = \sum_{y'_{\sim i} \in S_z} \frac{(1-x_i)+(1+x_i)\mathrm{e}^z}{2(1+\mathrm{e}^z)} \frac{\alpha(y'_{\sim i})p_{Y_{\sim i}}(y'_{\sim i})}{p_{X_i}(x_i)} = \frac{(1-x_i)+(1+x_i)\mathrm{e}^z}{2(1+\mathrm{e}^z)} \frac{\mathcal{V}'_z}{p_{X_i}(x_i)}$.

This shows that $\log \frac{p_{X_i|\Phi_i^{\mathrm{MAP}}}(+1|z)}{p_{X_i|\Phi_i^{\mathrm{MAP}}}(-1|z)} = z$ with the Bayes rule.

Finally, substitute $z$ in the equation obtained from $(a)$ to get $p_{X_i|\Phi_i^{\mathrm{MAP}},Y_{\sim i}}(x_i|z,y_{\sim i}) = p_{X_i|\phi_i^{\mathrm{MAP}}}(x_i|z)$, i.e., $Y_{\sim i} \to \Phi_i^{\mathrm{MAP}} \to X_i$. □

Observe that $\Phi_i^{\mathrm{MAP}}$ and $\Phi_i^{\mathrm{ML}}$ differ only by the constant term $a_i$ of the priors. Therefore $\Phi_i^{\mathrm{ML}}$ is also a sufficient statistic for estimating $X_i$.

**Fact 2.5** [MAP Estimate as Sufficient Statistic] Assume $Y_i \to X_i \to Y_{\sim i}$ using the conventions of Section 2.3, and let $\Phi_i^{\mathrm{MAP}} \triangleq \phi_i^{\mathrm{MAP}}(Y_{\sim i})$ be the (extrinsic) MAP estimator. Then $(Y_i, \Phi_i^{\mathrm{MAP}})$ is a sufficient statistic for estimating $X_i$ (given $Y$), i.e., $X_i \to (Y_i, \Phi_i^{\mathrm{MAP}}) \to Y$.

*Proof.* Assume that we are given $z \in \overline{\mathbb{R}}$ and let $y_{\sim i}$ be an element of $S_z \triangleq \{y'_{\sim i} : \phi_i^{\mathrm{MAP}}(y'_{\sim i}) = z\}$. Since $Y_i \to X_i \to Y_{\sim i}$, we first get $p_{X_i|Y_i,Y_{\sim i},\Phi_i^{\mathrm{MAP}}}(x_i|y_i,y_{\sim i},z) = \frac{p_{Y_i|X_i}(y_i|x_i)p_{X_i|Y_{\sim i},\Phi_i^{\mathrm{MAP}}}(x_i|y_{\sim i},z)}{\sum_{x'_i \in \mathcal{X}} p_{Y_i|X_i}(y_i|x'_i)p_{X_i|Y_{\sim i},\Phi_i^{\mathrm{MAP}}}(x'_i|y_{\sim i},z)}$. Since $X_i \to \Phi_i^{\mathrm{MAP}} \to Y_{\sim i}$ from Fact 2.4, we further get $p_{X_i|Y_{\sim i},\Phi_i^{\mathrm{MAP}}}(x_i|y_{\sim i},z) = p_{X_i|\Phi_i^{\mathrm{MAP}}}(x_i|z)$. Finally, substituting in the above equation, we get $p_{X_i|Y_i,Y_{\sim i},\Phi_i^{\mathrm{MAP}}}(x_i|y_i,y_{\sim i},z) = p_{X_i|Y_i,\Phi_i^{\mathrm{MAP}}}(x_i|y_i,z)$, i.e., $Y \to (Y_i, \Phi_i^{\mathrm{MAP}}) \to X_i$. □

**Fact 2.6** [LLR as Sufficient Statistic] Consider the channel $p_{Y_i|X_i}$ where $X_i$ and $Y_i$ are random variables, $X_i$ being binary. The LLR $Y_i = y(Y_i)$ is a sufficient statistic for estimating $X_i$.

*Proof.* The proof is similar to the one of Fact 2.4. It demonstrates $Y_i \to Y_i \to X_i$ using the set $S_z \triangleq \{y_i' : \phi_i(y_i') = z\}$. □

**Theorem 2.1** [Data Processing Inequality] If $X \to W \to V$, then $H(X|W) \leq H(X|V)$. Alternatively, $I(X,W|V) \leq I(X;W)$.

*Proof.* $X \to W \to V$ implies $H(X|W) = H(X|W,V) \leq H(X|V)$ since conditioning reduces uncertainty. Using the same argument, $I(X;W|V) = H(X|V) - H(X|V,W) = H(X|V) - H(X|W) \leq H(X) - H(X|W) = I(X;W)$. □

As a corollary, for any function $\phi$, consider $V = \phi(W)$. Then $X \to W \to V$ and the data processing inequality shows that $H(X|W) \leq H(X|V)$.

**Example 2.8** The previous remark shows that $H(X_i|Y_i) \leq H(X_i|y(Y_i))$. In addition, Fact 2.6 states that the LLR is a sufficient statistic for estimating $X_i$, i.e., $X_i \to y(Y_i) \to Y_i$, therefore $H(X_i|y(Y_i)) \leq H(X_i|Y_i)$ from the data processing theorem. Hence $H(X_i|Y_i) = H(X_i|y(Y_i))$.

**Example 2.9** The data processing theorem shows that $H(X_i|Y_{\sim i}) \leq H(X_i|\Phi_i^{\text{MAP}})$ (from Example 2.7). If $Y_i \to X_i \to Y_{\sim i}$ (e.g., for a memoryless channel), it also shows that $H(X_i|\Phi_i^{\text{MAP}}) \leq H(X_i|Y_{\sim i})$ because of Fact 2.4. Hence $H(X_i|\Phi_i^{\text{MAP}}) = H(X_i|Y_{\sim i})$.

**Example 2.10** The data processing theorem shows further that $H(X_i|Y) \leq H(X_i|Y_i,\Phi_i^{\text{MAP}})$ (from Example 2.7). If $Y_i \to X_i \to Y_{\sim i}$, it also shows $H(X_i|Y_i,\Phi_i^{\text{MAP}}) \leq H(X_i|Y)$ because of Fact 2.5. Hence $H(X_i|Y_i,\Phi_i^{\text{MAP}}) = H(X_i|Y)$.

**Example 2.11** [EXIT Upper Bound] A consequence of Example 2.9 is that for any function $\phi_i^{\text{DEC}}$, i.e., any estimator $\Phi_i^{\text{DEC}} \triangleq \phi_i^{\text{DEC}}(Y_{\sim i})$, $H(X_i|\Phi_i^{\text{MAP}}) \leq H(X_i|\Phi_i^{\text{DEC}})$.

Following the discussion of Section 2.2, Example 2.8 shows that channel post-processing does not deteriorate the information content of the channel output and can therefore be interpreted as part of the channel. A consequence is that the channel entropies computed in Example 2.4, Example 2.5 and Example 2.6 correspond to the true channel capacity (up to a change $I(X;Y) = 1 - H(X|Y)$ assuming equal priors). This, together with Fact 2.2, shows that the study of symmetric channels reduces to a study based on the symmetric channel density $\mathsf{a}(y)$.

**Lemma 2.2** [Channel Equivalence] Let $\mathsf{a}(y)$ be a symmetric density. The transition density $p_{Y|X}$ such that $p_{Y|X}(y|\pm 1) = \mathsf{a}(\pm y)$ describes a symmetric channel with associated $L$-density $\mathsf{a}(y)$.

*Proof.* First, the channel is symmetric since $p_{Y|X}(y|-1) = \mathsf{a}(-y) = p_{Y|X}(-y|+1)$. Second it has associated $L-$density $\mathsf{a}(y)$ since $\log \frac{p_{Y|X}(y|+1)}{p_{Y|X}(y|-1)} = \log \mathrm{e}^y = y$. □

## 2.5 Codes, Graphs and BP Estimator

Recall that a linear code of length $n$ is defined as the kernel of a $n \times m$ matrix, $m \leq n$, see [56]. Such a matrix has rank $n-k \leq m$ where $k$ is the code dimension; it is in general non-unique. A given parity-check matrix can be regarded as the incidence matrix of a hypergraph whose vertices or *variable nodes* represent the code components and whose hyperedges represent the parity-check constraint. A hypergraph can further be represented as a bipartite graph if we replace the hyperedges by *function nodes*: This graphical representation of a code where each function node is associated with a single parity-check constraint is called a *Tanner graph* [57]. Equivalently to a parity-check matrix, a Tanner graph also defines a code.

For example, the parity-check matrix

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

defines a code of length $n = 9$ and dimension $k = 5$ over the binary field $\mathbb{F}_2 \stackrel{\Delta}{=} \{0,1\}$. This binary $[9,5]$ linear code is equivalently defined by the (cycle-free) Tanner graph depicted in Figure 2.2.

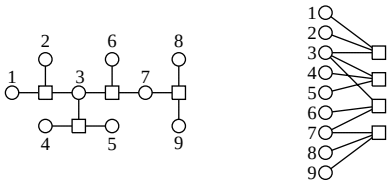



Figure 2.2: A Tanner tree.

Tanner graphs can be seen as a factorization of the code membership function if we associate a single-parity check code to each of its function nodes. In our example, $\mathbb{I}_{\{Hx^T=0\}} = \mathbb{I}_{\{x_1+x_2+x_3=0\}} \cdot \mathbb{I}_{\{x_3+x_4+x_5=0\}} \cdot \mathbb{I}_{\{x_3+x_6+x_7=0\}} \cdot \mathbb{I}_{\{x_7+x_8+x_9=0\}}$. In this respect Tanner graphs constitute a special case of *factor graphs*. Factor graphs [58–60, 67] are a handy tool for visualizing the factorization of a given function. It is sometimes advantageous to introduce auxiliary nodes to facilitate the factorization: such auxiliary nodes are associated with hidden variables such as state variables in trellis representations. A particular application of factor graphs concerns the description of algorithms used in estimation. They provide a very efficient way to reduce the computational complexity by exploiting the general distributive law over a (semi-)ring. See [75]. The estimation task will be performed *iteratively* using a *message-passing algorithm* on the factor tree. A standard message-passing algorithm in coding or statistical mechanics is the so-called Belief Propagation (BP), see [11, 76]. This algorithm, also known as the sum-product algorithm (for which BCJR [77], Turbo [62] or LDPC [10] decoding are particular instances), performs symbol MAP decoding on a tree. The BP algorithm will play a central role in our work. Let us therefore review the principles of message-passing decoding and, in particular, the definition of the *extrinsic* BP estimate. Recall that the MAP decision rule of Definition 2.7 maximizes

$$p_{X_i|Y}(\xi|y) \propto p_Y(y) p_{X_i|Y}(\xi|y) = \sum_{x_{\sim i}} p_{X_i,X_{\sim i},Y}(\xi,x_{\sim i},y) = \sum_{x_{\sim i}} p_{X_i,X_{\sim i}}(\xi,x_{\sim i}) p_{Y|X_i,X_{\sim i}}(y|\xi,x_{\sim i})$$

over $\xi \in \mathcal{X}$ ($\sigma$-additivity law of total probability and Bayes rule). We see from this general representation that an exponential number of terms $p_{X_i,X_{\sim i}}(\xi,x_{\sim i}) p_{Y|X_i,X_{\sim i}}(y|\xi,x_{\sim i})$ might be required to marginalize $p_{X_i,Y=y}(\xi) = p_Y(y) p_{X_i|Y}(\xi|y)$. Fortunately, because of the tree structure, a cascade of successive factorizations drastically reduces this computational complexity. In order to see this, assume $Y_i \to X_i \to Y_{\sim i}$ as in Section 2.3 and take the intrinsic factor out of the sum (distributive law in the ring of the reals) to get

$$p_{X_i|Y}(\xi|y) \propto p_Y(y) p_{X_i|Y}(\xi|y) = \underbrace{p_{Y_i|X_i}(y_i|\xi)}_{\text{intrinsic factor}} \cdot \underbrace{\sum_{x_{\sim i}} p_{X_i,X_{\sim i}}(\xi,x_{\sim i}) p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}|x_{\sim i})}_{\text{extrinsic factor}} .$$

In fact, the hypothesis $Y_i \to X_i \to Y_{\sim i}$ is embedded in the more general memoryless assumption which is assumed in the rest of the thesis. Now, assume that the extrinsic quantity $p_{X_i,X_{\sim i}}(\xi,x_{\sim i})$ (one function node) further factorizes into $K$ subfactors ($K$ function nodes) such that $p_{X_i,X_{\sim i}}(\xi,x_{\sim i}) = \prod_{k=1}^K f_k(\xi,x_{S_k})$ where $S_k \subseteq [n] \setminus \{i\}$ are pairwise disjoint. The distributive law permits us to write

$$\begin{aligned} p_{X_i|Y}(\xi|y) &\propto p_{Y_i|X_i}(y_i|\xi) \cdot \underbrace{\sum_{x_{\sim i}} \prod_{k=1}^K \left( f_k(\xi,x_{S_k}) p_{Y_{S_k}|X_{S_k}}(y_{S_k}|x_{S_k}) \right)}_{\text{extrinsic quantity}} \\ &= p_{Y_i|X_i}(y_i|\xi) \cdot \prod_{k=1}^K \left( \sum_{x_{\sim i}} f_k(\xi,x_{S_k}) p_{Y_{S_k}|X_{S_k}}(y_{S_k}|x_{S_k}) \right) . \end{aligned} \tag{2.1}$$

Furthermore, if each individual subfactor $f_k(\xi, x_{S_k})$ factorizes into $K_k$ subfactors, i.e., if $f_k(\xi, x_{S_k}) = f_k(x_{S_1^k}) \prod_{l_k=2}^{K_k} f_k(\xi, x_{S_{l_k}^k})$ where $S_{l_k}^k \subseteq S_k$ are pairwise disjoint, then

$$p_{X_i|Y}(\xi|y) \propto p_{Y_i|X_i}(y_i|\xi) \cdot \prod_{k=1}^{K} \Big( \sum_{x_{S_1^k}} f_k(x_{S_1^k}) p_{Y_{S_1^k}|X_{S_1^k}}(y_{S_1^k}|x_{S_1^k}) \underbrace{\Big( \sum_{x_{S_k \setminus S_{l_k}^k}} \prod_{l_k=+1}^{K_k} \big( f_k(\xi, x_{S_{l_k}^k}) p_{Y_{S_{l_k}^k}|X_{S_{l_k}^k}}(y_{S_{l_k}^k}|x_{S_{l_k}^k}) \big) \Big)}_{\text{new extrinsic quantity}} \Big).$$

We can now iterate the same procedure for the new extrinsic quantities. By successive iterations, we recursively describe the BP algorithm on a tree.

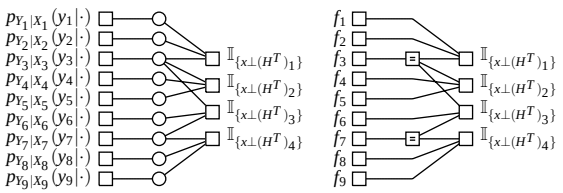



Figure 2.3: Factor graph representation. Left: Wiberg-style Factor graph. Right: Forney-style factor graph.

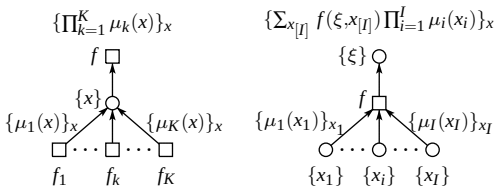



Figure 2.4: Message-passing rules. Left: variable node update (if variable node is a leaf, there are no incoming message functions $\mu_k(x)$ and $\{\mu(x) = 1\}_x$). Right: function node update (if function node is a leaf, there are no incoming functions $\mu_i$ and $\mu(x) = f(x)$).

Consider our running example and assume that the transmitted codewords are chosen uniformly at random from $\mathtt{C}$. Then $p_X(x) = \frac{1}{2^k} \mathbb{I}_{\{xH^T=0\}} \propto \prod_{k=1}^{4} \mathbb{I}_{\{x(H^T)_k=0\}}$ and we get

$$\begin{aligned}
\hat{x}_1^{\text{MAP}}(y) &\overset{(a)}{=} \text{argmax}_{x_1} \sum_{x_{\sim 1}} \prod_{j=1}^{9} p(y_j|x_j) \cdot \mathbb{I}_{\{x_1+x_2+x_3=0\}} \cdot \mathbb{I}_{\{x_3+x_4+x_5=0\}} \cdot \mathbb{I}_{\{x_3+x_6+x_7=0\}} \cdot \mathbb{I}_{\{x_7+x_8+x_9=0\}} \\
&\overset{(b)}{=} \text{argmax}_{x_1} p(y_1|x_1) \Big\{ \sum_{x_2,x_3} \mathbb{I}_{\{x_1+x_2+x_3=0\}} p(y_2|x_2) p(y_3|x_3) \cdot \Big[ \sum_{x_4,x_5} \mathbb{I}_{\{x_3+x_4+x_5=0\}} p(y_4|x_4) p(y_5|x_5) \Big] \cdot \\
&\qquad \Big[ \sum_{x_6,x_7} \mathbb{I}_{\{x_3+x_6+x_7=0\}} p(y_6|x_6) p(y_7|x_7) \Big( \sum_{x_8,x_9} \mathbb{I}_{\{x_7+x_8+x_9=0\}} p(y_8|x_8) p(y_9|x_9) \Big) \Big] \Big\}.
\end{aligned}$$

For each value of $x_1$, 6655 elementary operations (function evaluation, multiplication or addition) are required to determine the marginal in a brute force $(a)$. Some further thoughts show that this complexity reduces down to 601 elementary operations if one takes advantage of the distributive law $(b)$: The BP algorithm will operate recursively, evaluating first the quantity inside the bracket "( )" (in $4 \times 5 + 3$ operations), then the brackets "[ ]", and finally "{ }". This is better visualized by the propagation of *beliefs* through the factor graph in Figure 2.3 using the generic message-passing rules shown in Figure 2.4 and provided by Equation 2.1.

So far, the BP recursion has been written for a unique variable node but one can take advantage of a parallel processing of all variable nodes. In the remainder of the thesis (unless we explicitly use the equivalent peeling schedule, see Section 2.10), we choose the following time schedule. At iteration $\ell$

we simultaneously process all variable nodes, then all function nodes. The $L-$value

$$\phi_i^{\mathrm{BP}(\mathtt{G}),\ell}(y_{\sim i}) \triangleq \log \frac{\mu^\ell(+1,y_{\sim i})}{\mu^\ell(-1,y_{\sim i})}, \quad \text{is the } i^{\text{th}} \text{ BP estimate at iteration } \ell,$$

and $\{\mu^\ell(x_i,y_{\sim i})\}_{x_i}$ is given by the product of the messages coming from the neighboring function nodes at iteration $\ell$ (recall $\mu^{\ell=0}(+1,y_{\sim i}) = \mu^{\ell=0}(-1,y_{\sim i}) = 1$). The *BP decision* for the $i^{\text{th}}$ bit at the $\ell^{\text{th}}$ iteration is therefore $\hat{x}_i^{\mathrm{BP}(\mathtt{G}),\ell}(y) \triangleq \text{sign}\left(y_i + a_i + \phi_i^{\mathrm{BP}(\mathtt{G}),\ell}(y_{\sim i})\right)$.

Assume we are given a code $\mathtt{C}$. For notational simplicity, we skip the dependence of BP decoding on a particular graphical representation of $\mathtt{C}$, i.e., we use the superscript BP, instead of BP($\mathtt{G}$). For our running example,

$$\phi_1^{\mathrm{BP},0}(y_{\sim 1}) = 0, \qquad \phi_1^{\mathrm{BP},1}(y_{\sim 1}) = \log \frac{\sum_{x_2,x_3} \mathbb{I}_{\{x_2+x_3=0\}} p(y_2|x_2) p(y_3|x_3)}{\sum_{x_2,x_3} \mathbb{I}_{\{x_2+x_3=1\}} p(y_2|x_2) p(y_3|x_3)},$$

$$\phi_1^{\mathrm{BP},2}(y_{\sim 1}) = \log \frac{\sum_{x_2,x_3} \mathbb{I}_{\{x_2+x_3=0\}} p(y_2|x_2) p(y_3|x_3) [\sum_{x_4,x_5} \mathbb{I}_{\{x_4+x_5=x_3\}} p(y_4|x_4) p(y_5|x_5)] [\sum_{x_6,x_7} \mathbb{I}_{\{x_6+x_7=x_3\}} p(y_6|x_6) p(y_7|x_7)]}{\sum_{x_2,x_3} \mathbb{I}_{\{x_2+x_3=1\}} p(y_2|x_2) p(y_3|x_3) [\sum_{x_4,x_5} \mathbb{I}_{\{x_4+x_5=x_3\}} p(y_4|x_4) p(y_5|x_5)] [\sum_{x_6,x_7} \mathbb{I}_{\{x_6+x_7=x_3\}} p(y_6|x_6) p(y_7|x_7)]},$$

and $\phi_1^{\mathrm{BP},3}(y_{\sim 1}) = \phi_1^{\mathrm{MAP}}(y_{\sim 1})$ (cycle-free graph). In terms of LLRs, the dual rule (see Appendix 2.B and Example 2.14) reveals that the above expressions can be computed as $\phi_1^{\mathrm{BP},1}(y_{\sim 1}) = y_2 \boxplus y_3$, $\phi_1^{\mathrm{BP},2}(y_{\sim 1}) = y_2 \boxplus (y_3 + y_4 \boxplus y_5 + y_6 \boxplus y_7)$, and $\phi_1^{\mathrm{BP},3}(y_{\sim 1}) = y_2 \boxplus (y_3 + y_4 \boxplus y_5 + y_6 \boxplus (y_7 + y_8 \boxplus y_9)) = \phi_1^{\mathrm{MAP}}(y_{\sim 1})$ where the "boxplus function," denoted by "$\boxplus$", is defined in [63].

BP (for $\ell$ as large as the longest subtree) and MAP decoding are identical on a tree. However cycle-free graphs with a bounded state size do not appear 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 that only depends on the channel and the rate of the code [65, 78].

Therefore we will consider graphs with cycles as illustrated by the binary $[7,4]$ Hamming[5] code in Figure 2.5. First, let us show that the BP decoder is sub-optimal on such a graph. Assume that codewords are chosen uniformly at random from the $[7,4]$ Hamming code and transmitted through the channel BEC($\epsilon$). It is easy to check that all bit estimates have the same probability of erasure, i.e., $\forall i \in [7]$, $\Pr\{\hat{x}_i^{\mathrm{MAP}} = \text{❋}\} = \Pr\{\hat{x}_1^{\mathrm{MAP}} = \text{❋}\}$ (see further details in Chapter 3, in particular Lemma 3.5). Consider a decoder that decodes *up to* $d_{\min} - 1$ erasures. We call it a minimum-distance-based (MDB) decoder. For the $[7,4]$ Hamming code, we have $d_{\min} = 3$ so that the MDB decoder has extrinsic erasure probability[6] $\Pr\{\hat{x}_1^{\mathrm{MDB}}(y_{\sim 1}) = \text{❋}\} = 1 - \bar{\epsilon}^6 - 6\epsilon\bar{\epsilon}^5$. In fact, a MAP decoder can recover certain patterns beyond $d_{\min} - 1$ and a tedious but conceptually easy exercise (see also Chapter 3) shows that $\Pr\{\hat{x}_1^{\mathrm{MAP}}(y_{\sim 1}) = \text{❋}\} = 1 - \bar{\epsilon}^6 - 6\epsilon\bar{\epsilon}^5 - 12\epsilon^2\bar{\epsilon}^4 - 4\epsilon^3\bar{\epsilon}^3 = 3\epsilon^2 + 4\epsilon^3 - 15\epsilon^4 + 12\epsilon^5 - 3\epsilon^6$. Of course we have $\Pr\{\hat{x}_1^{\mathrm{MAP}}(y_{\sim 1}) = \text{❋}\} \leq \Pr\{\hat{x}_1^{\mathrm{MDB}}(y_{\sim 1}) = \text{❋}\}$, i.e., we

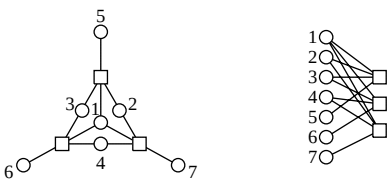



Figure 2.5: A $[7,4]$ Hamming Tanner graph.

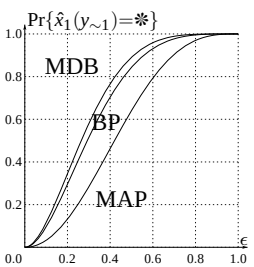



Figure 2.6: Comparison of the extrinsic erasure functions for the $[7,4,3]$ Hamming and various decoding algorithms when transmission takes place over the BEC.

[5] A $p$-ary $[p^r - 1, p^r - 1 - r]$ Hamming code is defined using the (parity-check) matrix whose columns are all non-zero $p$-ary $r$-tuples, see [56]. This construction implies that Hamming codes are perfect codes with distance $d_{\min} = 3$.

[6] Recall that for a single-parity check code of length $\mathtt{r}$, the *extrinsic erasure probability* is given by $\Pr\{\hat{x}_i^{\mathrm{MAP}}(y_{\sim i}) = \text{❋}\} \triangleq \Pr\{\hat{x}_i^{\mathrm{MAP}}(Y_i = \text{❋}, y_{\sim i}) = \text{❋}\} = \frac{\Pr\{\hat{x}_i^{\mathrm{MAP}} = \text{❋}\}}{\epsilon} = 1 - \bar{\epsilon}^{\mathtt{r}-1}$ where $\bar{\epsilon} = 1 - \epsilon$. This is true because the code can correct *exactly* $d_{\min} - 1 = 1$ erasure. For single-parity check codes, we have $\Pr\{\hat{x}_i^{\mathrm{MAP}}(y_{\sim i}) = \text{❋}\} = \Pr\{\hat{x}_i^{\mathrm{MDB}}(y_{\sim i}) = \text{❋}\}$.

find that the MDB decoder is *sub-optimal*. It is now natural to ask: How would a BP decoder perform? Some thought reveals that the extrinsic erasure probability is $\Pr\{x_1^{\mathrm{BP}}(Y_i=\divideontimes, y_{\sim 1})=\divideontimes\} = 12\epsilon^2 - 28\epsilon^3 + 27\epsilon^4 - 12\epsilon^5 + 2\epsilon^6$. See also [16] for the finite-length analysis of BP decoding. The performance curves of the respective decoders are compared in Figure 2.6. We see that the BP decoder on the considered graph is strictly sub-optimal. BP decoding performs only a *local* search; its sub-optimality is a general statement (implied by the data processing theorem).
Notice nevertheless that the BP performance curve in Figure 2.6 is not "too" poor, thus BP decoding is used in practice (in particular associated with sparse graphs). The general wisdom is to apply BP decoding to graphs with loops and to consider this type of decoding as a (typically) strictly sub-optimal attempt to perform maximum a posteriori (MAP) bit decoding. Therefore one would not expect any link between the BP and the MAP decoder, except for the obvious sub-optimality of the BP decoder... We will see that the actual typical behavior is more surprising!

## 2.6 Standard Notations for Iterative Coding Systems

To be concrete, most of the statements of this thesis will be exemplified using standard Low-Density Parity-Check (LDPC) codes introduced in [10]. However, the results extend to various scenarios, among others Generalized LDPC (GLDPC) code ensembles or multi-edge ensembles such as Turbo codes. Many statements will therefore be stated in a general form. LDPC codes were originally defined in [9, 10] as the kernel of a pseudo-random low-density parity-check matrix whose rows and columns have a fixed number of non-zero entries. Such a matrix is therefore *sparse* for large lengths $n$. More generally, iterative coding systems are described by sparse factor graphs.
An in-depth introduction to the analysis of LDPC ensembles are found, e.g., in [12–15]. Further references on LDPC and iterative coding analysis are [11, 25, 57, 62, 63, 67, 79–92]. We will use standard conventions; for convenience of the reader, and to settle notation, let us nevertheless 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 left and right degree distributions of the graph from the *edge* perspective (For Turbo codes, $\lambda(\mathtt{x})$ will be the distribution of the *systematic* information symbols, see Chapter 7). We consider the ensemble $\mathrm{LDPC}(n, \lambda, \rho)$ of such graphs[7] 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. The changes of representation are obtained via $\Lambda(\mathtt{x}) = \frac{\int_0^{\mathtt{x}} \lambda(u)\mathrm{d}u}{\int \lambda}$, $\Gamma(\mathtt{x}) = \frac{\int_0^{\mathtt{x}} \rho(u)\mathrm{d}u}{\int \rho}$, $\lambda(\mathtt{x}) = \Lambda'(\mathtt{x})/\Lambda'(1)$ and $\rho(\mathtt{x}) = \Gamma'(\mathtt{x})/\Gamma'(1)$. Notice that $\Lambda'(1) = \frac{1}{\int \lambda}$ is the average left (variable node) degree, $\Gamma'(1) = \frac{1}{\int \rho}$ is the average right (check node) degree. An important characteristic of the ensemble $\mathrm{LDPC}(n, \lambda, \rho) = \mathrm{LDPC}(n, \Xi)$ is the *design rate* $r_\Xi \triangleq 1 - \int \rho / \int \lambda = 1 - \Lambda'(1)/\Gamma'(1)$. Let $r_{\mathtt{G}}$ be the actual rate of an element of the ensemble $\mathrm{LDPC}(n, \Xi)$. The rate $r_{\mathtt{G}}$ is potentially larger than the design rate (as the associated parity-check matrix has potentially linearly dependent rows). However, when $n \to \infty$, in many cases, this rate is provably the design rate with high probability. This is formalized in the next section.

## 2.7 Asymptotic Rate and Design Rate

Consider a dd pair $\Xi$ and let $r_\Xi$ be the associated design rate. Consider the ensemble $\mathrm{LDPC}(n, \Xi)$ and let $\mathtt{G}$ be chosen uniformly at random from this ensemble with rate $r_{\mathtt{G}}$. For our purpose, we would like to know when the asymptotic average rate converges to the design rate, i.e., when

$$\lim_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[r_{\mathtt{G}}] = r_\Xi.$$

[7] Clarifications: First, the ensemble of graphs is in general slightly larger that the ensemble of corresponding codes. For example, a graph might have multiple edges or two graphs might represent the same code. We do not distinguish such cases because it is shown to have a negligible effect on the average ensemble performance. Second, codes (or graphs) in general cannot be constructed for any $n$ but only for any $n_m$ where $\{n_m\}_m$ is a sub-sequence of $\{n\}_n$. For the analysis, we only deal with the sequence $\{\mathrm{LDPC}(n_m \lambda, \rho)\}_m$ and the shorthand "$n \to \infty$" means in fact "$m \to \infty$."

At first view, one would expect that this statement holds for *any* LDPC ensemble. However this is not necessarily always the case, see discussion in Section 4.5. The next lemma asserts that, under some technical conditions, the actual rate of a random element of an ensemble is equal to the design rate with high probability when the blocklength $n \to \infty$.

**Lemma 2.3** [Design Rate versus Asymptotic Rate] Consider a dd pair $\Xi$ with associated design rate $r_\Xi$. Let us define the function

$$\Theta_\Xi(u) \triangleq -\Lambda'(1)\log_2\left[\frac{1+u\cdot v_u}{(1+u)(1+v_u)}\right] + \sum_{\mathtt{l}} \Lambda_{\mathtt{l}} \log_2\left[\frac{1+u^{\mathtt{l}}}{2(1+u)^{\mathtt{l}}}\right] + \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{\mathtt{r}} \Gamma_{\mathtt{r}} \log_2\left[1+\big(\frac{1-v_u}{1+v_u}\big)^{\mathtt{r}}\right]$$

using $v_u = \big(\sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}} u^{\mathtt{l}-1}}{1+u^{\mathtt{l}}}\big)/\big(\sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}}}{1+u^{\mathtt{l}}}\big)$. Let $\mathtt{G}(n)$ be chosen uniformly at random from the ensemble $\mathrm{LDPC}(n,\Xi)$, let $r_{\mathtt{G}(n)}$ denote its actual rate. If $\forall u \in (0,1)$, $\Theta_\Xi(u) \le 0$, then $r_{\mathtt{G}(n)}$ converges and

$$\lim_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[r_{\mathtt{G}}] = r_\Xi.$$

More precisely, $\exists B>0$, $\forall \xi>0$, $\exists n_{\xi,\Xi} \in \mathbb{N}$, such that $\forall n > n_{\xi,\Xi}$ we have $\Pr\{|r_{\mathtt{G}(n)} - r_\Xi| > \xi\} \le \mathrm{e}^{-B\xi n}$, and $\exists C>0$ such that $\forall n > n_{\xi,\Xi}$ we have $\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[r_{\mathtt{G}(n)} - r_\Xi] \le C\frac{\log n}{n}$.

*Proof.* Recall that for any $\mathtt{G} \in \mathrm{LDPC}(n,\Xi)$, we have $r_{\mathtt{G}} \ge r_\Xi$, and that Jensen's inequality reads $n r_\Xi \le \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}\big[n r_{\mathtt{G}}\big] = \frac{\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[\log_2 N_{\mathtt{G}}]}{n} \le \frac{\log_2 \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[N_{\mathtt{G}}]}{n}$. The idea of the proof is to use the first-order moment method and the Hayman approximation to derive an upper bound on the average rate of the ensemble $\mathrm{LDPC}(n,\Xi)$ when $n\to\infty$. If 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 rates close to the design rate.
Following [9,87,90,93–99] and using weight enumerator functions, we write that the expected number of codewords involving $E$ edges is given by

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

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 quoted papers it is known that, for a fixed $e \in [0,1]$, 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

$$\sum_{\mathtt{l}} \Lambda_{\mathtt{l}} \log_2(1+u^{\mathtt{l}}) - \Lambda'(1) e \log_2 u + \frac{\Lambda'(1)}{\Gamma'(1)} \sum_{\mathtt{r}} \Gamma_{\mathtt{r}} \log_2 q_{\mathtt{r}}(v) - \Lambda'(1) e \log_2 v - \Lambda'(1) h(e). \quad (2.2)$$

We aim at evaluating the exponent corresponding to the expected number of codewords, i.e., we want to determine $\lim_{n\to\infty} \frac{1}{n}\log_2\big(\mathbb{E}[N_{\mathtt{G}}]\big)$, where $N_{\mathtt{G}} = \sum_E N_{\mathtt{G}}(E)$. As there is only a linear number of "types" (numbers $E$) this exponent is equal to the supremum of (2.2) over all $0 \le e \le 1$. In summary, the sought after exponent is given by a stationary point of the function stated in Eq. (2.2) with respect to $u$, $v$ and $e$. Taking the derivative with respect to $e$ gives us $e = uv/(1+uv)$. If we substitute this expression for $e$ into Eq. (2.2), subtract the design rate $r(\Lambda,\Gamma)$, and rearrange the terms somewhat, we get the expression of $\Theta_\Xi(u)$. Next, if we take the derivative with respect to $u$ and solve for $v$, we get the expression for $v_u$. In summary, $\Theta_\Xi(u)$ is a function so that

$$\log_2 \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[N_{\mathtt{G}}] = o_n(n) + n\big(r_\Xi + \sup_{u\in[0,\infty)} \Theta_\Xi(u)\big).$$

In particular, by explicit computation we see that we always have $\Theta_\Xi(1) = 0$. The case $u=1$ corresponds to the exponent of codewords of weight $n/2$. Therefore, the condition that the global maximum of $\Theta_\Xi(u)$ is achieved at $u=1$ is equivalent to the condition that the expected weight enumerator is dominated by codewords of weight (close to) $n/2$. In this case, there exists $B>0$ such that $\forall \xi>0$, $\exists n_{\xi,\Xi} \in \mathbb{N}$, $\forall n > n_{\xi,\Xi}$,

$$\Pr\{r_{\mathtt{G}} \ge r_\Xi + \xi\} = \Pr\left\{N_{\mathtt{G}} \ge 2^{n(\xi - o_n(1))} \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[N_{\mathtt{G}}]\right\} \le e^{-Bn\xi},$$

where the last step follows from the Markov inequality if $B = (\log 2)/2$ and $\omega_n \leq \xi/2$ for any $n \geq n_0$. Moreover, since $r_{\mathtt{G}} \leq 1$, we get $\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[|r_{\mathtt{G}} - r_\Xi|] \leq \xi + e^{-Bn\xi}$, and the last claim follows by choosing $\xi = \log n/Bn$.

It now remains to show that $\Theta_\Xi(u)$ achieves its maximum over $[0,+\infty)$ in $[0,1]$. With this aim, first observe the following symmetries. The function $u \mapsto v_u$ enjoys the property $v_{1/u} = 1/v_u$ for any $u > 0$, e.g., $u \in (0,1)$ implies $v_u \in (0,1)$. In fact, the change $(u, v_u) \leftrightarrow (1/u, 1/v_u)$ corresponds to the change $e \leftrightarrow 1-e$, which indicates the symmetry around the half-weight codewords. Observe now $\Theta_\Xi(u) - \Theta_\Xi(1/u) = \frac{2\Lambda'(1)}{\Gamma'(1)} \sum_{\mathtt{r}:\mathtt{r}\text{ odd}} \Gamma_{\mathtt{r}} \log_2 \left[\frac{(1+v_u)^{\mathtt{r}}+(1-v_u)^{\mathtt{r}}}{(1+v_u)^{\mathtt{r}}-(1-v_u)^{\mathtt{r}}}\right] = \frac{2\Lambda'(1)}{\Gamma'(1)} \sum_{\mathtt{r}:\mathtt{r}\text{ odd}} \Gamma_{\mathtt{r}} \boxplus_{j=1}^{\mathtt{r}} \log(1/v_u) \overset{(a)}{\geq} 0$ for all $u \in (0,1)$ where $(a)$ is an equality if and only if $\Gamma_{\mathtt{r}} = 0$ for all odd degree $\mathtt{r}$. Therefore, if $\Theta_\Xi$ has a maximum in $u' > 1$, it has necessarily another maximum in $u = 1/u' < 1$. Since $\Theta_\Xi(1) = 1$, we finally have that $\lim_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[r_{\mathtt{G}}] = r_\Xi$ whenever $\Theta_\Xi(u) \leq 0$ over $(0,1)$. □

Discussion: First notice that, if the conditions of the lemma are fulfilled, then the function $\Theta_\Xi$ is locally concave around 1. If we use slightly stronger conditions, i.e., if we assume $\forall u \in (0,1)$, $\Theta_\Xi(u) < 0$, then the function is strictly concave in 1. In this case the function is also locally quadratic (i.e., locally "Gaussian"): This property is further investigated in [98] where it is assumed that the maximum of $\Theta_\Xi$ is unique and achieved for $u = 1$. In this case where $\forall u \in (0,1)$, $\Theta_\Xi(u) < 0$, it can even be specified that $\Pr\{nr_{\mathtt{G}(n)} = nr_\Xi + \delta\} = 1 - o_n(1)$ where $\delta = 0$ in general, and $\delta = 1$ if $\mathtt{x} \mapsto \Lambda(\mathtt{x})$ is an even function: This means that all parity-check equations (except one trivially obtained as the sum of all remaining parity-check equations) are linearly independent with high probability.

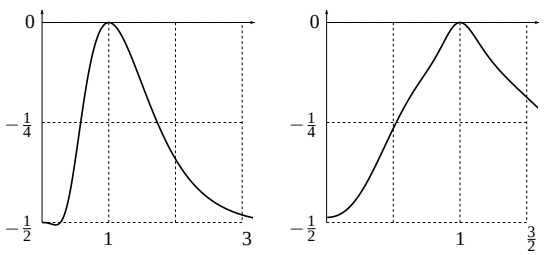



Figure 2.7: Characterization of the growth rate of the average weight via $u \mapsto \Theta_\Xi(u)$. Left: dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$ with design rate $r = \frac{1}{2}$. Right: dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\frac{2\mathtt{x}+3\mathtt{x}^2+4\mathtt{x}^{13}}{10}, \mathtt{x}^6)$ with design rate $r = \frac{19}{39}$.

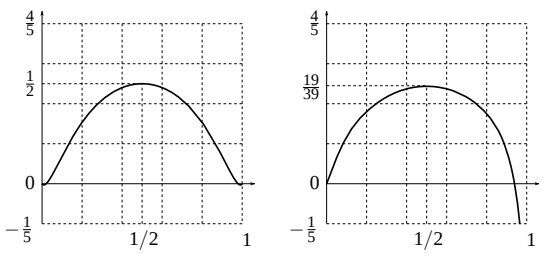



Figure 2.8: Growth rate of the average weight via the parametric curve $\{(\frac{uv_u}{1+uv_u}, \Theta_\Xi(u))\}_{u\in[0,+\infty)}$. Left: dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$ with design rate $r = \frac{1}{2}$. Right: dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\frac{2\mathtt{x}+3\mathtt{x}^2+4\mathtt{x}^{13}}{10}, \mathtt{x}^6)$ with design rate $r = \frac{19}{39}$.

Second, observe that the function $\Theta_\Xi(u)$ is essentially a re-parameterization for the growth rate of the average weight distribution.

This is illustrated in Figure 2.7. The parameter $u = 1$ corresponds to codewords of a relative weight of one-half. The standard picture for the growth rate of the average weight distribution is depicted in Figure 2.8. In the two considered examples, the maximum growth rate corresponds to codewords of a relative weight of one-half: The maximum of $u \mapsto \Theta_\Xi(u)$ is achieved for $u = 1$ and Lemma 2.3 asserts that the actual asymptotic rate is the design rate.

One further consequence of the characterization given by Lemma 2.3 is the following. The observation stated at the end of the proof shows that the growth rate of the average weight distribution is symmetric with respect to the line representing the half-weight codewords iff all parity-check nodes have even degree.

Lemma 2.3 is practical in the sense that it is a "plug and play" criterion to insure that the actual rate is the design rate with high probability for sufficiently large blocklengths. However, it is only a *sufficient* condition. It could happen that this condition is not fulfilled although the actual rate is the design rate with high probability. This would mean that the average growth rate can still be strictly below the growth rate of the average weight distribution obtained from the combinatorial first moment method.

Finally, let us show that for regular LDPC ensembles the actual rate always converges to the design rate.

**Theorem 2.2** [Asymptotic Rate for Regular Ensembles] Consider a regular dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^{\mathtt{l}-1}, \mathtt{x}^{\mathtt{r}-1})$. Let $r_{\mathtt{l},\mathtt{r}} = 1 - \frac{\mathtt{l}}{\mathtt{r}}$ be the associated design rate. Consider the ensemble $\mathrm{LDPC}(n, \mathtt{x}^{\mathtt{l}-1}, \mathtt{x}^{\mathtt{r}-1})$

and let $\mathtt{G}(n)$ be chosen uniformly at random from this regular ensemble. Let $r_{\mathtt{G}(n)}$ denote its actual rate. Then

$$\lim_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\mathtt{x}^{\mathtt{l}-1},\mathtt{x}^{\mathtt{r}-1})}[r_{\mathtt{G}}] = r_{\mathtt{l},\mathtt{r}}.$$

*Proof.* In the regular case the expression of $\Theta_{\Xi}(u)$ simplifies to

$$\Theta_{\Xi}(u) = \log\Big(\frac{1}{2}(1+u^{\mathtt{l}})^{1-\mathtt{l}}((1+u^{\mathtt{l}-1})^{\mathtt{r}} + (1-u^{\mathtt{l}-1})^{\mathtt{r}})^{\frac{1}{\mathtt{r}}}\Big).$$

Define $x \triangleq u^{\mathtt{l}-1}$. Then the condition $\Theta_{\Xi}(u) \le 0$, with strict inequality except for $u = 1$, is equivalent to $f(x,\mathtt{r}) \le g(x,\mathtt{l})$, with strict inequality except for $x = 1$, where $f(x,\mathtt{r}) \triangleq ((1+x)^{\mathtt{r}} + (1-x)^{\mathtt{r}})^{\frac{1}{\mathtt{r}}}$ and $g(x,\mathtt{l}) \triangleq 2^{\frac{1}{\mathtt{l}}}(1+x^{\frac{1}{\mathtt{l}-1}})^{\frac{\mathtt{l}-1}{\mathtt{l}}}$. We start by showing that for $\mathtt{r} \ge 2$ and $x \ge 0$, $f(x,\mathtt{r}) \le g(x,\mathtt{r})$, i.e., that the desired inequality is true for the choice $\mathtt{l} = \mathtt{r}$. To see this, consider the equivalent statement $2\sum_i \binom{\mathtt{r}}{2i} x^{2i} = f(x,\mathtt{r})^{\mathtt{r}} \le g(x,\mathtt{r})^{\mathtt{r}} = 2\sum_j \binom{\mathtt{r}-1}{j} x^{\frac{\mathtt{r}}{\mathtt{r}-1}j}$. For $\mathtt{r} = 2$ a direct check shows that the two sides are equal and the same is true for $x = 0$. Consider therefore the case $\mathtt{r} \ge 3$ and $x > 0$. First, cancel the factor 2 from both sides. Next, note that both series start with the term 1 and if $\mathtt{r}$ is even then the last term on both sides is $x^{\mathtt{r}}$. For each remaining term on the left of the form $\binom{\mathtt{r}}{2i}x^{2i}$, $2 \le 2i < \mathtt{r}$, there are exactly two terms and they have the form $\binom{\mathtt{r}-1}{2i-1}x^{\frac{(2i-1)\mathtt{r}}{\mathtt{r}-1}} + \binom{\mathtt{r}-1}{2i}x^{\frac{2i\mathtt{r}}{\mathtt{r}-1}}$ on the right. Now note that for $x > 0$, the function $\alpha \mapsto x^{\alpha}$ is convex for $\alpha > 0$ and that $\binom{\mathtt{r}-1}{2i-1} + \binom{\mathtt{r}-1}{2i} = \binom{\mathtt{r}}{2i}$. Therefore by Jensen,

$$\frac{\binom{\mathtt{r}-1}{2i-1}}{\binom{\mathtt{r}}{2i}}x^{\frac{(2i-1)\mathtt{r}}{\mathtt{r}-1}} + \frac{\binom{\mathtt{r}-1}{2i}}{\binom{\mathtt{r}}{2i}}x^{\frac{2i\mathtt{r}}{\mathtt{r}-1}} \ge \Big(x^{(\binom{\mathtt{r}-1}{2i-1}\frac{(2i-1)\mathtt{r}}{\mathtt{r}-1} + \binom{\mathtt{r}-1}{2i}\frac{2i\mathtt{r}}{\mathtt{r}-1})/\binom{\mathtt{r}}{2i}}\Big) = x^{2i}.$$

As we know that $f(x,\mathtt{r}) \le g(x,\mathtt{r})$ for $\mathtt{r} \ge 2$ and $x \ge 0$, the proof will be complete if we can show that $g(x,\mathtt{l})$ is a decreasing function in $\mathtt{l}$ and that it is strictly decreasing except for $x = 1$: we write $f(x,\mathtt{r}) \le g(x,\mathtt{r}) \le g(x,\mathtt{l})$, where the last inequality is strict for $x \ne 1$.
It remains to show that $g(x,\mathtt{l})$ is indeed decreasing in $\mathtt{l}$. Consider the related function $\tilde{g}_x(\mathtt{l}) \triangleq 2^{\frac{1}{\mathtt{l}}}(1+x^{\frac{1}{\mathtt{l}-1}})^{\frac{\mathtt{l}-1}{\mathtt{l}}}$ where $\mathtt{l}$ is now a real-valued variable. It is easy to check that the sign of $\frac{\mathrm{d}\tilde{g}_x(\mathtt{l})}{\mathrm{d}\mathtt{l}}$ is given by the opposite of the sign of $w(x,\mathtt{l}) \triangleq \mathtt{l}x^{\frac{1}{\mathtt{l}-1}}\log(x) + (\mathtt{l}-1)\left(1+x^{\frac{1}{\mathtt{l}-1}}\right)\left(\log(2) - \log(1+x^{\frac{1}{\mathtt{l}-1}})\right)$. Define $y \triangleq x^{1/(\mathtt{l}-1)}$ to get $w(x,\mathtt{l}) = \tilde{w}_{\mathtt{l}}(y) \triangleq (\mathtt{l}-1)\big(y\log(y) + (1+y)\log(2) - (1+y)\log(1+y)\big)$. Since $\tilde{w}'_1(y) = (\mathtt{l}-1)\log\frac{2y}{1+y}$, $\tilde{w}'_1(y) = \frac{1}{y(1+y)}$, we find that $\tilde{w}_{\mathtt{l}}(y)$ achieves its (unique) minimum in 1 such that $\tilde{w}_{\mathtt{l}}(1) = 0$, and $\tilde{w}_{\mathtt{l}}(y) > 0$ for $y \in (0,+\infty)\setminus\{1\}$. This shows that $\tilde{g}_x(\mathtt{l})$ is decreasing in $\mathtt{l}$ and concludes the proof. □

Discussion: With the same arguments as above, one can say that, in the case of such $(\mathtt{l},\mathtt{r})$-regular LDPC ensembles, we have $\Pr\{r_{\mathtt{G}(n)} = r_{\mathtt{l},\mathtt{r}}n + \nu\} = 1 - o_n(1)$ where $\nu = 1$ if $\mathtt{l}$ is even, and $\nu = 0$ otherwise.

## 2.8 Degraded Channels and Threshold

Once an ensemble and its corresponding asymptotic rate (or design rate) have been fixed, a natural approach for practical coding is to consider a family of channels ordered by some measure of the "noise" and to study at which *threshold* the "noise" prevents (asymptotically) the decoder from recovering the information. The channel family will be typically $\{\mathrm{BMSC}(\mathtt{p})\}_{\mathtt{p}}$ where $\mathtt{p}$ is a real-valued parameter, see Section 2.1. In most of the cases, $\mathtt{p}$ can be viewed as the channel entropy $\mathtt{h}$ such that an increase of $\mathtt{p}$ corresponds to some *degradation* of the transmission channel. The largest value of $\mathtt{p}$ that is compatible with a vanishing bit error probability will be the *threshold* associated with the considered decoding. Following [14, 15] let us formalize those concepts when the transition densities are assumed to exist.

**Definition 2.10** [Physical Degradation] Assume we are given two channels $p_{Z|X}, p_{Y|X}$ with input alphabet $\mathcal{X}$, and output alphabets $\mathcal{Z}$, respectively $\mathcal{Y}$. We say that $p_{Z|X}$ is physically degraded with respect to

$p_{Y|X}$ and denote $p_{Y|X} \prec p_{Z|X}$ if and only if there exists a joint distribution $p_{Y,X|Z}$ such that $X \to Y \to Z$.

A few remarks are in order. First, let us comment our definition of *channel degradation*. It is easy to see that *physical* degradation implies *stochastic* degradation [55]. In fact, most of the statements of this thesis are meaningful if we consider a channel $p_{Z|X}$ that is stochastically degraded with respect to $p_{Y|X}$. However, since we are mostly interested in marginals (or linear operators acting on marginals), we are free to think of the channel $p_{Z|X}$ as a physically degraded version of $p_{Y|X}$.
Second, let us justify the notation "$\prec$". Observe that two memoryless channels $p_{Y|X} \prec p_{Z|X}$ are such that $H(X|Y) \leq H(X|Z)$ because of the data processing theorem. Moreover, as shown in [65], if we consider coded transmission over these two channels such that $\Pr^{\text{MAP}}(p_{Y|X})$ and $\Pr^{\text{MAP}}(p_{Z|X})$ are the respective (bit or block) error probabilities associated with a MAP decoder, then $\Pr^{\text{MAP}}(p_{Y|X}) \leq \Pr^{\text{MAP}}(p_{Z|X})$. Third, observe that if the channels are binary and symmetric, then the channel $p_{Z|Y}$ itself is also symmetric, see [65].

**Definition 2.11** [Order implied by Physical Degradation] Consider the input alphabet $\mathcal{X}$ and the output alphabet $\mathcal{Y}$. Consider a family of memoryless channels with common input and output alphabets $\{p^{\mathfrak{p}}_{Y|X}\}_{\mathfrak{p}\in P}$ parameterized by $\mathfrak{p} \in P \subseteq \mathbb{R}$. This family is said to be ordered by physical degradation if and only if $\mathfrak{p}_1 < \mathfrak{p}_2$ implies $p^{\mathfrak{p}_1}_{Y|X} \prec \mathfrak{p}^{\mathfrak{p}_2}_{Y|X}$.

If a family parameterized by $-\mathfrak{p}$ is ordered by physical degradation, we will sometimes (when there is no risk of confusion) say that the family itself is ordered by physical degradation.

**Definition 2.12** [Ordered and complete Family] Consider the input alphabet $\mathcal{X}$ and the output alphabet $\mathcal{Y}$. Consider a family of memoryless channels with common input and output alphabets $\{p^{\mathfrak{p}}_{Y|X}\}_{\mathfrak{p}\in P}$ parameterized by $\mathfrak{p} \in P \subseteq \mathbb{R}$. If this family is ordered by physical degradation and if $\{\mathfrak{h}^{\mathfrak{p}}\}_{\mathfrak{p}} \triangleq \{H(X|Y(\mathfrak{p}))\}_{\mathfrak{p}}$ ranges from 0 to $H(X)$ (where $\mathfrak{p}$ describes $P$ and where $H(X|Y(\mathfrak{p}))$ is the conditional entropy associated with $p^{\mathfrak{p}}_{Y|X}$), then the family is said to be ordered and complete.

**Example 2.12** The channel families $\{\text{BEC}(\epsilon)\}_{\epsilon\in[0,1]}$, $\{\text{BSC}(\epsilon)\}_{\epsilon\in[0,1/2]}$, and $\{\text{BAWGNC}(\sigma)\}_{\sigma\in[0,\infty)}$ are all ordered and complete.

The notion of threshold is inherent to the notion of physical degradation. Let us now review different thresholds that characterize transmission over a complete and ordered family of memoryless symmetric channels, call it $\{p^{\mathfrak{p}}_{Y|X}\}_{\mathfrak{p}\in P}$. First, the ultimate limit is the Shannon threshold that we denote by $\mathfrak{p}^{\text{SH}} \triangleq \mathfrak{h}^{-1}(1-r_\infty)$ where $r_\infty$ indicates the (asymptotic) rate of transmission. For this rate the channel coding theorem (see [2,54,55]) shows that transmission at a vanishing (block) error probability (independently of the code and/or decoder) is not possible above this threshold.
The existence of a threshold phenomena concerning the MAP decoding of codes is discussed in [21,22] when the minimum distance of a sequence of linear codes of length $n$ tends to infinity when $n \to \infty$. Let us exemplify this notion for the case of an ensemble of LDPC codes characterized by the dd pair $\Xi$ from a node perspective.

**Definition 2.13** [MAP Threshold] Consider a dd pair $\Xi$ and assume that $\mathtt{G}$ is chosen uniformly at random from $\text{LDPC}(n,\Xi)$. Assume that transmission takes place over a complete and ordered family of BMS channels. The MAP threshold is defined as

$$\mathfrak{p}^{\text{MAP}} \triangleq \min\{\mathfrak{p} : \liminf_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}\left[H_{\mathtt{G}}(X|Y(\epsilon))/n\right] > 0\}.$$

Discussion: Observe that, with this definition, the inequality $\mathfrak{p}^{\text{MAP}} \leq \mathfrak{p}^{\text{SH}}$ is a rephrasing of the channel coding theorem (combined with the Fano inequality and the strong converse [2,55]). To see this, recall that the Fano inequality implies that the block error probability is (up to some fixed scaling) larger than the entropy rate, i.e., $\Pr\{\hat{x}^{\text{MAP}}_{[n]}(Y) \neq X\} \geq (H(X|Y)-1)/(nr_{\mathtt{G}})$. This implies that transmission at a vanishing (block) error probability (for this particular ensemble) is not possible *in average* above this threshold. Moreover, a stronger result is given by the strong converse. This states that transmission

is reliable below this threshold so that $\mathfrak{p}^{\text{MAP}} \leq \mathfrak{p}^{\text{SH}}$. Another reason to define the MAP threshold as above is more intuitive and considers the conditional entropy as a measure of the typical number of codewords compatible with a received vector. Let us now consider the operational meaning of the above definition for a *particular* instance of transmission. On the one hand, assume that $\mathfrak{p} < \mathfrak{p}^{\text{MAP}}$, then there exists a subsequence of blocklengths so that the *average* conditional entropy rate converges to zero. Assume that the conditional normalized entropy concentrates (this result is shown in Theorem 4.3). It follows that most of the codes in the corresponding ensembles have a conditional entropy rate smaller than any fixed constant. For sufficiently large blocklengths, a conditional entropy that grows sublinearly implies that the receiver can limit the set of hypothesis to a subexponential list that with high probability contains the correct codeword. Therefore, in this sense, reliable communication is possible. On the other hand, assume that $\mathfrak{p} > \mathfrak{p}^{\text{MAP}}$. In this case the conditional entropy rate stays bounded away from zero by a strictly positive constant for all sufficiently large blocklengths. If the conditional normalized entropy concentrates (Theorem 4.3), then this is not only true for the average over the ensemble but for most elements from the ensemble. It follows that with high probability, for most elements from the ensemble, reliable communication is not possible.

Notice that we set the hypothesis of a *complete* family simply in order to lay the emphasis on practical communication schemes. This hypothesis is however not strictly required by the above definition of threshold (an *ordered* family would be sufficient).

Similar to the MAP threshold, specific ensembles like LDPC ensembles exhibit a threshold phenomena when they are decoded using the BP algorithm. In this case, the behavior is well-defined as observed in [12–15]. Various (equivalent) definitions of the BP threshold are possible. We will use the following.

**Definition 2.14** [BP Threshold] Consider a dd pair $\Xi$ and assume that $\mathtt{G}$ is chosen uniformly at random from LDPC$(n,\Xi)$. Assume that transmission takes place over a complete and ordered family of BMS channels. The BP threshold is defined as

$$\mathfrak{p}^{\text{BP}} \triangleq \inf\{\mathfrak{p} : \lim_{\ell\to\infty}\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)} \frac{1}{n}\sum_{i=1}^{n} \Pr\{\hat{x}_i^{\text{BP},\ell}(Y) \neq X_i\} > 0\}.$$

Discussion: Let $P_b^{\text{MAP}} \triangleq \frac{1}{n}\sum_{i=1}^{n} \Pr\{\hat{x}_i^{\text{MAP}}(Y) \neq X_i\}$ denote the average symbol error probability and $h_2$ the binary entropy. The Fano inequality reads $h_2(P_b^{\text{MAP}}) \geq H(X|Y)/n$. Hence our definition of $\mathfrak{p}^{\text{BP}}$ since it has for straightforward consequence (using the sub-optimality of BP decoding shown in Example 2.9 after taking the limits) that $\mathfrak{p}^{\text{BP}} \leq \mathfrak{p}^{\text{MAP}}$.

**Example 2.13** [Thresholds over the BEC] Assume transmission takes place over $\{BEC(\epsilon)\}_{\epsilon\in[0,1]}$. The BP threshold is alternatively determined as $\epsilon^{\text{BP}} \triangleq \sup\{\epsilon \in [0,1] : \epsilon\lambda(1-\rho(1-\mathtt{x})) < \mathtt{x}, \forall \mathtt{x} \in (0,1]\}$. See [12–15] and Chapter 3. 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 $n \to \infty$. Otherwise, if $\epsilon \geq \epsilon^{\text{BP}}$, then a fixed fraction of bits remains erased after BP decoding when $n \to \infty$. The BP threshold associated with LDPC$(\mathtt{x}^2, \mathtt{x}^5)$ is $\epsilon^{\text{BP}} \approx 0.429$. Values for the MAP threshold were first obtained by the replica method in [28]. Some steps of the replica method are not rigorously justified and, in [37] a simple counting argument leading to an upper bound for this threshold is given. This argument is explained and sharpened in Section 4.2.2. In this thesis we will develop the viewpoint taken in [48] and we will see that the MAP threshold associated with LDPC$(\mathtt{x}^2, \mathtt{x}^5)$ is $\epsilon^{\text{MAP}} \approx 0.488$.

In the previous example, we have verified that $\epsilon^{\text{BP}} \approx 0.429 \leq \epsilon^{\text{MAP}} \approx 0.488 \leq \epsilon^{\text{SH}} = 0.5$. Note that the last inequality is obtained from the previous section where it is shown that the design rate equals the asymptotic rate for regular ensembles. As stated above, the inequalities

$$\mathfrak{p}^{\text{BP}} \leq \mathfrak{p}^{\text{MAP}} \leq \mathfrak{p}^{\text{SH}} \triangleq 1 - \liminf_{n\to\infty}\left(\mathbb{E}_{\text{LDPC}(n,\Xi)}[r_{\mathtt{G}}]\right) \leq 1 - r_{\Xi}$$

are true for any complete and ordered family of BMS channels. Furthermore, for BEC$(\epsilon)$, we show in Appendix 2.C that

$$\epsilon^{\text{BP}} \leq \epsilon^{\text{MAP}} \leq \min\{\epsilon^{\text{SH}}, \epsilon^{\text{SC}}\},$$

where $\epsilon^{\text{SH}}$ and $\epsilon^{\text{SC}} \triangleq \frac{1}{\lambda'(0)\rho'(1)}$ denote the Shannon and stability condition thresholds, respectively. A consequence of the Maxwell construction presented in this thesis is that the above relations will generalize naturally over any family of BMS channels.

## 2.9 Channel Smoothness

The order implied by physical degradation leads naturally to the notion of "differentiability" with respect to a measure of the degradation. More precisely, in the chapters ahead, especially Chapter 5, Chapter 6 and Chapter 7, we will often be concerned by the "differentiability" of certain quantities with respect to a given uncertainty measure. For example, we will study how the conditional entropy $H(X|Y)$ behaves when the noise in the channel varies. In order to ensure that the considered objects exist, we need to impose some regularity conditions on the channel family with respect to a given channel parameter. This can be done in various ways. We choose the following convention for practical reasons, see, e.g., [100].

**Definition 2.15** [Channel Smoothness] Consider the input alphabet $\mathcal{X}$ and the output alphabet $\mathcal{Y}$. Consider a family of memoryless channels with common input and output alphabets $\{p^{\mathfrak{p}}_{Y|X}(y|x)\}_{\mathfrak{p}\in P}$ parameterized by $\mathfrak{p} \in P \subseteq \mathbb{R}$. The channel family is said to be *smooth* with respect to $\mathfrak{p}$ if for all $x \in \mathcal{X}$ and all bounded continuously differentiable functions $f(y)$, the integral $\int f(y)p^{\mathfrak{p}}_{Y|X}(y|x)\mathrm{d}y$ exists and is a continuously differentiable function with respect to $\mathfrak{p}$, $\mathfrak{p} \in P$.

Discussion: If $\{\text{BMSC}(\mathfrak{p})\}$ is smooth, the derivative $\frac{\mathrm{d}}{\mathrm{d}\mathfrak{p}}\int f(y)p_{Y|X}(y|x)\mathrm{d}y$ exists and is a linear functional of $f$. It is therefore consistent to formally *define* the derivative of $p_{Y|X}(y|x)$ with respect to $\mathfrak{p}$ by setting $\frac{\mathrm{d}}{\mathrm{d}\mathfrak{p}}\int f(y)p_{Y|X}(y|x)\,\mathrm{d}y \triangleq \int f(y)\frac{\mathrm{d}p_{Y|X}(y|x)}{\mathrm{d}\mathfrak{p}}\,\mathrm{d}y$. In a large number of cases it is relatively easy to check that the channel family is smooth, for example, if $\mathcal{Y}$ is finite and the transition probabilities are differentiable functions of $\mathfrak{p}$, or if it admits a density with respect to the Lebesgue measure and the density is differentiable for each $y$. In these cases, our formal definition coincides with the ordinary derivative. Examples are the BMS channel families $\{\text{BEC}(\epsilon)\}_{\epsilon=0}^{1}$, $\{\text{BSC}(\epsilon)\}_{\epsilon\in[0,1/2]}$, and $\{\text{BAWGNC}(\sigma)\}_{\sigma\in[0,\infty)}$, which are then all smooth.
In the case of transmission over a BMSC, we will see that it is interesting and useful to parameterize the channels in such a way that the parameter reflects the channel entropy $\mathtt{h} \triangleq H(X|Y)$. More precisely, let $\{\mathtt{c}_{\mathfrak{p}}\}_{\mathfrak{p}}$ be a family of BMS channels characterized by their $L$-densities and such that the random input $X$ has equal priors. We then write this family of $L$-densities as $\{\mathtt{c}_{\mathtt{h}}\}_{\mathtt{h}}$ if $\mathtt{H}(\mathtt{c}_{\mathfrak{p}}) = \mathtt{h}$, where $\mathtt{H}$ is the entropy operator of Definition 2.5. Observe that, if $\mathtt{c}$ is a (symmetric) $L-$density, then

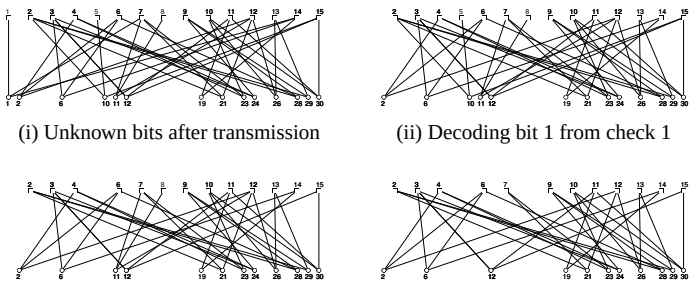
$$\mathtt{H}(\mathtt{c}) \triangleq \int_{-\infty}^{\infty}\mathtt{c}(y)\log_2(1+e^{-y})\mathrm{d}y = \int_{-\infty}^{\infty}\mathtt{c}(y)l(y)\mathrm{d}y = \int_0^{\infty}h_2\left(\frac{\mathrm{e}^{-y}}{1+\mathrm{e}^{-y}}\right)\mathtt{c}^{|L|}(y)\mathrm{d}y,$$

where $\mathtt{c}^{|L|}$ indicates the channel density in the $|D|-$domain, see Appendix 2.B. This integral always exists; it is continuously differentiable in $\mathfrak{p}$ when the family is smooth (If the channel does not admit a density, then this can also be seen by writing it in the equivalent form as Riemann-Stieltjes integral).

## 2.10 Peeling Decoder

In this thesis we will deal essentially with the BP schedule discussed in Section 2.5. However an alternative and equivalent description is presented in [12, 13] for the case of transmission over the BEC. A similar approach is found in [38]. The analysis of this alternative schedule is based on the Wormald method [101, 102]. It is very convenient to gain insight, for example, in the finite-length behavior of iterative decoding. This also illuminates the behavior above threshold and the notion of *residual graph*. Let us call such an implementation a *peeling* decoder and review the basic principles. This is illustrated in Figure 2.9.
Let $\mathtt{G}$ (with length $n$) be chosen uniformly at random from an ensemble characterized from a node perspective by the dd pair $\Xi \triangleq (\Lambda, \Gamma)$. Assume that transmission takes place over BEC($\epsilon$). The

peeling decoder proceeds as follows. A variable node is removed (together with all connected edges) as soon as it has received (either from the channel or from the incoming edges) at least one known message. At each iteration, a check node of degree one is chosen uniformly at random among all check nodes with degree one. This check node is further removed, as well as all connected edges. At the end of the decoding process, all check nodes have degree at least two: The decoder is in a *stopping set* as the one depicted in Figure 2.9 (iv). A stopping set defines a *residual graph* with a given degree profile.



(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

Figure 2.9: Code of length $n = 30$ and peeling decoder. At the decoder, the variable nodes which have received a non-erased bit are removed from the bipartite graph. The remaining graph is shown in (i). The peeling decoder determines successively bits 1, 10 and 11, until it gets stuck. The stopping set is shown in (iv).

Let $\mathtt{G}(\epsilon)$ denote such a particular graph obtained from $\mathtt{G}$ and transmission over BEC($\epsilon$). Let us further denote by $\Xi_{\mathtt{G}(\epsilon)}$ its degree profile and $\Lambda_{\mathtt{G}(\epsilon)}(1)n$ its length. It is easy to check, see [12, 13], that stopping sets are uniformly distributed over an ensemble of residual graphs once we have fixed the degree profile $\Xi_{\mathtt{G}(\epsilon)}$ and the length $\Lambda_{\mathtt{G}(\epsilon)}(1)n$. The degree profile of the residual graph $\Xi_{\mathtt{G}(\epsilon)}$ is a random quantity because of the channel randomness. However it is sharply concentrated around its expected value. In the asymptotic limit when $n \to \infty$ this expected value converges to a typical dd pair, call it $\Xi_\epsilon$.

The expected number of remaining variable nodes (parity-check nodes with degree at least 2) after $nt$ steps of the peeling algorithm, normalized by $n$, converges to $L_\epsilon^\delta \triangleq \epsilon\Lambda(1-\rho(1-\delta))$ (respectively, $R_\epsilon^\delta \triangleq (1-r_\Xi)\sum_{j\geq 2}\sum_i \Gamma_i\binom{i}{j}\delta^j(1-\delta)^{i-j}$), where $\delta = \delta(t) \in [0,1]$, sometimes called *state of the system*, parameterizes (smoothly) the decoding process. The limiting value for $\delta$ (i.e., once the peeling algorithm has terminated and is stuck in a stopping set) equals the fixed point of density evolution (see Section 3.2). In the limit $\delta = \mathtt{x}$, when there are no more check nodes with degree one, the total number of parity-check nodes remaining at the end of the decoding is then $R_\epsilon = \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{j\geq 2}\sum_i \Gamma_i\binom{i}{j}\mathtt{x}^j(1-\mathtt{x})^{i-j} = \frac{\Lambda'(1)}{\Gamma'(1)}\sum_i \Gamma_i\sum_{j\geq 2}\binom{i}{j}\mathtt{x}^j(1-\mathtt{x})^{i-j} = \frac{\Lambda'(1)}{\Gamma'(1)}\sum_i \Gamma_i(1-i\mathtt{x}(1-\mathtt{x})^{i-1}-(1-\mathtt{x})^i) = \Lambda'(1)(\int_{1-\mathtt{x}}^1 \rho(u)\mathrm{d}u - \mathtt{x}\rho(1-\mathtt{x}))$ while the number of variable nodes is $L_\epsilon = \epsilon\lambda(1-\rho(1-\mathtt{x}))$. Observe that the expected difference (divided by $n$) between the residual numbers of variable and check nodes is then

$$P(\mathtt{x}) \triangleq L_\epsilon - R_\epsilon = \epsilon\Lambda(\mathtt{y}(\mathtt{x})) - \frac{\Lambda'(1)}{\Gamma'(1)}(1-\Gamma(1-\mathtt{x})) + \Lambda'(1)\mathtt{x}(1-\mathtt{y}(1-\mathtt{x})), \tag{2.3}$$

where $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$. In Chapter 4 we will observe that $\epsilon = \epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ (at the fixed point of density evolution) and call the resulting polynomial *trial entropy*, because it indicates the number of remaining degrees of freedom of the linear system.

A more refined description of the residual graph is needed if we want to know whether or not the linear system of equations has full rank (i.e., whether or not the parity-check equations are independent with high probability). With this aim, we shall describe the expected degree distribution of the residual graph from a node perspective. Let us therefore introduce an unknown variable $\mathtt{z}$ in order to describe the degree distribution as a polynomial in $\mathtt{z}$. Similarly to the previous description of the total number of nodes, the expected (normalized with respect to the original graph) degree distribution of the variable nodes from a node perspective converges to $L_\epsilon(\mathtt{z}) \triangleq \epsilon\Lambda(\mathtt{z}\mathtt{y})$ while the (normalized with respect to the original graph) distribution of the check nodes converges to $R_\epsilon(\mathtt{z}) = \frac{\Lambda'(1)}{\Gamma'(1)}\sum_{j\geq 2}\sum_i \Gamma_i\binom{i}{j}(\mathtt{z}\mathtt{x})^j(1-\mathtt{x})^{i-j}$. A similar calculation as above shows that the expected degree distribution of the residual graph

typically has the form

$$\Xi_\epsilon = \big(\Lambda_\epsilon(\mathtt{z}), \Gamma_\epsilon(\mathtt{z})\big) \triangleq \left(\frac{\Lambda(\mathtt{zy})}{\Lambda(\mathtt{y})}, \frac{\Gamma(1-\mathtt{x}-\mathtt{xz}) - \Gamma(1-\mathtt{x}) - \mathtt{zx}\Gamma'(1-\mathtt{x})}{1-\Gamma(1-\mathtt{x}) - \mathtt{x}\Gamma'(1-\mathtt{x})}\right)$$

where $\mathtt{x}$ denotes the fixed point of density evolution (i.e., the largest solution of $\mathtt{x} = \epsilon\lambda(1-\rho(1-\mathtt{x}))$) when the channel parameter in $\epsilon$ and $\mathtt{y} \triangleq 1-\rho(1-\mathtt{x})$.[8]
In the sequel the dd pair associated with the residual graph combined with the technical condition of Lemma 2.3 will permit us to determine the asymptotic rate of the residual ensemble in which BP decoding gets stuck. This will be investigated in Chapter 4, where we will determine MAP thresholds for iterative coding systems.

## 2.11 Conclusion and Discussion

We have settled notations and conventions for the analysis presented in the following chapters. Markov chains and the order implied by physical degradation, linear functionals and asymptotic rates will play a central role in this thesis. We have introduced the main tools, mostly in the context of a *binary* input alphabet $\mathcal{X}$.

Binary input alphabets are indeed our main domain of investigation, but we will see, e.g., in Chapter 5, that many concepts of our analysis extend to the non-binary case. Think of the result stating that the LLR post-processing gives rise to an equivalent channel. In the *non-binary* case, the LLR mapping can be replaced by the canonical representation of the channel output $y \mapsto y(y) \triangleq \{p_{Y|X}(y|x)/z(y) : x \in \mathcal{X}\}$, where $z(y) \triangleq \sum_{x\in\mathcal{X}} p_{Y|X}(y|x)$ (discrete assumption). In this case, $y(y)$ belongs to the $(|\mathcal{X}|-1)$-dimensional simplex. In the binary case, the LLR representation is a particular parameterization of the one-dimensional simplex. Various alternatives are possible, for example the "soft bit" (or "difference") parameterization $\mathbb{E}[X|Y=y]$. See, e.g., [63, 65, 103]. Let $\{\mathtt{c}_\mathtt{p}\}_\mathtt{p}$ represent a family of BMS channels such that the random input $X$ has equal priors and such that there is a bijection between the channel entropy $\mathtt{h} \triangleq H(X|Y) = \mathtt{H}(\mathtt{c}_\mathtt{p}) = \mathtt{h}(\mathtt{p})$ and the channel parameter $\mathtt{p}$ (see Section 2.1). We then write this family of $L$-densities as $\{\mathtt{c}_\mathtt{h}\}_\mathtt{h}$. By some abuse of notation, we will sometimes, especially in Chapter 5 and Chapter 6, write BMSC$(\mathtt{h})$ instead of BMSC$(\mathtt{p})$ to denote a BMSC of parameter $\mathtt{p}$ with entropy $\mathtt{h}$.

In the next chapter, we will give a first motivation for the choice of the entropy as channel parameter. With this aim, we will present EXIT functions and their main properties.

# Appendix

## 2.A Proper Linear Codes

A random $X_i$ over the finite alphabet $\mathcal{X}$ is said to have equal priors if for all $x_i \in \mathcal{X}$ $\Pr\{X_i = x_i\} = \frac{1}{|\mathcal{X}|}$. Assume we are given a code $\mathtt{C}$. If $X$ is chosen uniformly at random from $\mathtt{C}$, the codewords are said to have equal priors. A "non-trivial" binary linear code is expected to be such that its symbols have equal priors when the codewords are equally likely. Next definition characterizes such codes.

[8] Observe that, if we adopt the convention of normalizing the dd pair with respect to $n$, then we get a non-standard dd pair $(L_\epsilon(\mathtt{z}), R_\epsilon(\mathtt{z})) = (\epsilon\Lambda(\mathtt{zy}), (\Lambda'(1)/\Gamma'(1))(\Gamma(1-\mathtt{x}-\mathtt{xz}) - \Gamma(1-\mathtt{x}) - \mathtt{zx}\Gamma'(1-\mathtt{x})))$ (whose coefficients do not sum to one). If we now adopt the convention of normalizing the dd pair with respect to the original graph (i.e., dividing the number of variable nodes by $n$ and the number of check nodes by $n(\Lambda'(1)/\Gamma'(1))$), then we get an alternative non-standard dd pair $(\mathcal{L}_\epsilon(\mathtt{z}), \mathcal{R}_\epsilon(\mathtt{z})) = \big(\epsilon\Lambda(\mathtt{zy}), \Gamma(1-\mathtt{x}-\mathtt{xz}) - \Gamma(1-\mathtt{x}) - \mathtt{zx}\Gamma'(1-\mathtt{x})\big)$ (whose coefficients do not sum to one).

**Definition 2.16** [Proper Linear Codes] A linear code $\mathtt{C}$ of length $n \geq 1$ is said to be *proper* if and only if its dual code $\mathtt{C}^{\perp}$ has minimum distance $d_{\min}^{\perp} > 1$, or equivalently, if and only if it possesses a generator matrix with no zero column.

Almost all linear codes used in practice are proper. We will often use proper linear codes for our statements. In a proper binary linear code, half the codewords take on the value $+1$ and half the value $-1$ in each given bit position. This is stated in the next fact, a basic exercise in information theory.

**Fact 2.7** Let $\mathtt{C}$ be a proper linear binary code of length $n$. Assume $X$ is chosen uniformly at random from $\mathtt{C}$, then $\forall \mathcal{S} \subseteq [n]$, $\forall i \in [n] \setminus \mathcal{S}$, $\forall (x_i, x_{\mathcal{S}}) \in \mathbb{F}_2^{1+|\mathcal{S}|}$, $\Pr\{X_i = x_i | X_{\mathcal{S}} = x_{\mathcal{S}}\} = \frac{1}{2}$.

Proper linear codes are often needed for technical reasons. For example, the proper code assumption will imply that our definition of the EXIT function in Chapter 3 is simply the complementary to one of the original definition in [33]. More important, the next lemma shows that proper codes preserve channel symmetry when a MAP decoder is considered. As it can be found in [14, 15, 65], this property has also for consequence that the densities appearing in density evolution are also symmetric. Let us review this result for our purpose.

**Lemma 2.4** [Symmetry, Linearity, and MAP Decoder] Let $\mathtt{C}$ be a proper linear binary code of length $n$. Assume $X$ is chosen uniformly at random from $\mathtt{C}$ and is passed through a BMS channel. Let $\mathtt{a}_i(z)$ denote the $L-$density associated with the $i^{\text{th}}$ MAP extrinsic estimate $\Phi_i^{\text{MAP}}$, i.e., the density of $\log \frac{p_{Y_{\sim i}|X_i}(y_{\sim i}|+1)}{p_{Y_{\sim i}|X_i}(y_{\sim i}|-1)}$ conditioned on $X_i = 1$. Then $\mathtt{a}_i(z)$ is symmetric, i.e., $\mathtt{a}_i(-z) = \mathrm{e}^z \mathtt{a}_i(-z)$.

*Proof.* The proof is virtually identical to the one in [65]. We simply need to distinguish between intrinsic and extrinsic part. With this aim, observe that $p_{Y_{\sim i}|X_i}(y_{\sim i}|\xi) p_{X_i}(\xi) = \int \sum_{x: x_i = \xi} p_{Y|X}(y|x) p_X(x) \mathrm{d}y_i = \left( \int p_{Y_i|X_i}(y_i|\xi) dy_i \right) \left( \frac{1}{|\mathtt{C}|} \sum_{x \in \mathtt{C}: x_i = \xi} p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}|x_{\sim i}) \right)$ such that $\frac{p_{Y_{\sim i}|X_i}(y_{\sim i}|+1)}{p_{Y_{\sim i}|X_i}(y_{\sim i}|-1)} = \frac{\sum_{x \in \mathtt{C}: x_i = +1} p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}|x_{\sim i})}{\sum_{x \in \mathtt{C}: x_i = -1} p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}|x_{\sim i})}$ where we have used the "memoryless" and "proper" assumptions. The proof follows similarly to [65] since the channel symmetry gives $p_{Y_{\sim i}|X_{\sim i}}(y|x) = p_{Y_{\sim i}|X_{\sim i}}(x_{\sim i} y_{\sim i} | \underline{1}) = p_{Y_{\sim i}|X_{\sim i}}(w_{\sim i} y_{\sim i} | w_{\sim i} x_{\sim i})$ for all $w, x \in \mathtt{C}$, where the vector product is the component-wise product and $\underline{1}$ the all-one codeword. □

Discussion: Observe that the assumptions in Lemma 2.4 could be weakened. First, the following symmetries of channel would be first sufficient: $\forall w \in \mathtt{C}$, $p_{Y_{\sim i}|X_{\sim i}}(y|w) = p_{Y_{\sim i}|X_{\sim i}}(w_{\sim i} y_{\sim i} | \underline{1})$, $Y_i \to X_i \to Y_{\sim i}$, and $p_{Y_i|X_i}$ symmetric. Second, the generator matrix of the code only needs to have non-zero $i^{\text{th}}$ column.

Because of Lemma 2.4, operations on conditionally independent $L$-values like "$+$", "$\boxplus$", or any other computation performed by a MAP decoder preserve symmetry.

## 2.B Duality and Change of Domain

Duality relationships play an important role in iterative coding. The first part of this section is a review of the well-known dual decoding rule presented in [86, 104]. This will lead us to another duality rule presented in [105, 106]. We will finally review some notations for the different dual representations.

The dual decoding rule is based on the MacWilliams identities, see [56, 107]. If $\mathtt{C}$ is a binary linear code of length $n$, recall that the multi-variate *extended MacWilliams polynomial* associated with $\mathtt{C}$, which we denote by $P_C(\mathtt{a}_{[n]}; \mathtt{b}_{[n]}) = P_C(\mathtt{a}_1, \cdots, \mathtt{a}_n; \mathtt{b}_1, \cdots, \mathtt{b}_n) \triangleq \sum_{x \in \mathtt{C}} \prod_{j \in [n]} \mathtt{a}_j^{(1+x_j)/2} \mathtt{b}_j^{(1-x_j)/2}$ is such that the following identity is satisfied.

**Theorem 2.3** [Extended MacWilliams Theorem] Let $\mathtt{C}$ be a binary linear code of length $n$ and $\mathtt{C}^{\perp}$ its dual. Then $P_C(\mathtt{a}_{[n]}; \mathtt{b}_{[n]}) = \frac{1}{|\mathtt{C}|} P_{C^{\perp}}(\mathtt{a}_1 + \mathtt{b}_1, \cdots, \mathtt{a}_n + \mathtt{b}_n; \mathtt{a}_1 - \mathtt{b}_1, \cdots, \mathtt{a}_n - \mathtt{b}_n)$.

Discussion: Observe that, for any $i \in [n]$, $P_C(\mathtt{a}_{[n]}; \mathtt{b}_{[n]}) = \mathtt{a}_i S_{i,\mathtt{C}}^{+1}(\mathtt{a}_{[n]}; \mathtt{b}_{[n]}) + \mathtt{b}_i S_{i,\mathtt{C}}^{-1}(\mathtt{a}_{[n]}; \mathtt{b}_{[n]})$ where we

use $S_{i,\mathtt{C}}^{\pm 1}(\mathtt{a}_1,\cdots,\mathtt{a}_n;\mathtt{b}_1,\cdots,\mathtt{b}_n) \triangleq \sum_{x\in\mathtt{C}:x_i=\pm 1}\prod_{j\in[n]\setminus\{i\}}\mathtt{a}_j^{(1+x_j)/2}\mathtt{b}_j^{(1-x_j)/2}$. Define further

$$r_i^{\mathbf{in}}(y_i) \triangleq \exp(-y_i) = \frac{p_{Y_i|X_i}(y_i|-1)}{p_{Y_i|X_i}(y_i|+1)} \text{ to be the } i^{\text{th}} \textit{ intrinsic} \text{ (inverse) ratio and,}$$

$$r_{i,\mathtt{C}}^{\mathbf{out}}(y_{[n]\setminus\{i\}}) \triangleq \exp(-\phi_i^{\text{MAP}}) = \frac{p_{X_i|Y_{[n]\setminus\{i\}}}(-1|y_{[n]\setminus\{i\}})}{p_{X_i|Y_{[n]\setminus\{i\}}}(+1|y_{[n]\setminus\{i\}})} \text{ to be the } i^{\text{th}} \text{ MAP } \textit{extrinsic} \text{ (inverse) ratio.}$$

With these conventions, the (bit) MAP decoding rule can be expressed as follows.

**Lemma 2.5** [MAP Decoding and Ratio Parameterization] Let $\mathtt{C}$ be a binary linear code of length $n$. Assume that $X$ is chosen uniformly at random from $\mathtt{C}$ and that transmission takes place over a BMS channel. Define the values $\{a_i \triangleq p_{Y_i|X_i}(y_i|+1)\}$, $\{a_i \triangleq p_{Y_i|X_i}(y_i|-1)\}$, then

$$r_i^{\mathbf{in}}(y_{[n]\setminus\{i\}}) = \frac{b_i}{a_i}, \qquad\qquad r_{i,\mathtt{C}}^{\mathbf{out}}(y_{[n]\setminus\{i\}}) = \frac{S_{i,\mathtt{C}}^{-1}(a_{[n]};b_{[n]})}{S_{i,\mathtt{C}}^{+1}(a_{[n]};b_{[n]})}.$$

*Proof.* Let us focus on the right identity. Given $(y_i,y_{\sim i})$, we use the $\sigma$-additivity to write $p_{X_i|Y_{\sim i}(\xi|y_{\sim i})} = \sum_{x:x_i=\xi} p_{X|Y_{\sim i}}(x|y_{\sim i}) \propto \sum_{x:x_i=+1} p(x,y_{\sim i})$. The channel is memoryless, therefore $p_{Y_i|X,Y_{\sim i}}(y_i|x,y_{\sim i}) = p_{Y_i|X_i}(y_i|\xi)$. This shows that $p_{X_i|Y_{\sim i}(\xi|y_{\sim i})} \propto \frac{1}{p_{Y_i|X_i}(y_i|\xi)}\sum_{x:x_i=\xi} p_{X|Y}(x,y) \propto \frac{1}{p_{Y_i|X_i}(y_i|\xi)}\sum_{x\in\mathtt{C}:x_i=\xi} p_{Y|X}(y|x)$ because the code has equal priors. We use the memoryless assumption again to factorize $p_{Y|X}(y|x)$. This concludes the proof. □

In the domain of the ratios, the discrete Fourier transform is equivalent to the involution $\mathcal{F}: r \mapsto \frac{1-r}{1+r}$ (such that $\mathcal{F}=\mathcal{F}^{-1}$). We use the notation $\mathcal{F}(r_{[n]}) \triangleq (\mathcal{F}(r_1),\cdots,\mathcal{F}(r_n))$ if $r_{[n]}$ is a vector. We can now state the dual decoding rule of [86, 104].

**Theorem 2.4** [MAP Dual Decoding] Let $\mathtt{C}$ be a binary linear code of length $n$ and $\mathtt{C}^\perp$ its dual. Assume that $X$ is chosen uniformly at random from $\mathtt{C}$ and that transmission takes place over a BMS channel. With the previous notations,

$$r_{i,\mathtt{C}}^{\mathbf{out}}(y_{[n]\setminus\{i\}}) = r_{i,\mathtt{C}}^{\mathbf{out}}(r_{[n]\setminus\{i\}}^{\mathbf{in}}), \qquad\qquad r_{i,\mathtt{C}}^{\mathbf{out}}(r_{[n]\setminus\{i\}}^{\mathbf{in}}) = \mathcal{F}\left(r_{i,\mathtt{C}^\perp}^{\mathbf{out}}[\mathcal{F}(r_{[n]\setminus\{i\}}^{\mathbf{in}})]\right).$$

*Proof.* The following are equivalent:

$$\begin{aligned}
r_{i,\mathtt{C}}^{\mathbf{out}}(r_{[n]\setminus\{i\}}^{\mathbf{in}}) = u \quad &\Longleftrightarrow \quad S_{i,\mathtt{C}}^{-1}(a_{[n]};b_{[n]}) - uS_{i,\mathtt{C}}^{+1}(a_{[n]};b_{[n]}) = 0\\
&\Longleftrightarrow \quad P_C(a_1,\cdots,\mathtt{a}_i=-u,\cdots,a_n;b_1,\cdots,\mathtt{b}_i=1,\cdots,b_n) = 0\\
&\overset{\text{Th. 2.3}}{\Longleftrightarrow} \quad P_{C^\perp}(a_1+a_1,\cdots,1-u,\cdots,1+a_n;a_1-b_1,\cdots,-1-u,\cdots,a_n-b_n) = 0\\
&\Longleftrightarrow \quad (-1-u)S_{i,\mathtt{C}^\perp}^{-1}(a_{[n]}+b_{[n]};a_{[n]}-b_{[n]}) + (1-u)S_{i,\mathtt{C}^\perp}^{+1}(a_{[n]}+b_{[n]};a_{[n]}-b_{[n]}) = 0\\
&\Longleftrightarrow \quad r_{i,\mathtt{C}^\perp}^{\mathbf{out}}[\mathcal{F}(r_{[n]\setminus\{i\}}^{\mathbf{in}})] = \mathcal{F}(u).
\end{aligned}$$ □

Discussion: Theorem 2.4 has various practical applications, for example it gives rise to a low-complex decoding of high rate codes in [63, 108]. In fact Theorem 2.4 (with the identity $2\tanh^{-1}(u) = \log\frac{1+u}{1-u}$) shows that the two implementations represented in Figure 2.10 are equivalent.
An illustration is given in Example 2.14 where the update rule for LLRs at the parity-check nodes (viewed as single parity-check codes) is obtained from the product of the (Fourier transform of the) ratios entering the corresponding variables nodes (viewed as dual codes, i.e., repetition codes).

**Example 2.14** [Rule at Variable and Function Nodes for LDPC Decoding] Assume the channel outputs $n+1$ $L$-values $y_1,\cdots,y_{n+1}$. For the $[n+1,1,n+1]$ repetition code, the $n+1^{\text{th}}$ MAP extrinsic estimate in the $L-$domain equals $\sum_{i=1}^n y_i$ (using the left scheme in Figure 2.10). For the $[n+1,n,2]$ single parity-check code, the $n+1^{\text{th}}$ MAP extrinsic estimate in the $L-$domain equals $\boxplus_{i=1}^n y_i \triangleq 2\tanh^{-1}\left(\prod_{i=1}^n \tanh(y_i/2)\right)$ (using the right scheme in Figure 2.10).


Figure 2.10: Two equivalent implementations of a decoder with $L$-values at input/output. The MAP decoder uses ratios as inputs/outputs, e.g., it is a simple product for the case of a repetition code. Left: Implementation based on the actual code and its MAP decoding. Right: Implementation in the dual domain.

Observe moreover that Theorem 2.4 shows also that the input ratios constitute a sufficient statistic for estimating $X_i$. See also Lemma 2.2. As shown in Figure 2.11, a channel can be equivalently defined by its $L/R/D$-density.

It is now natural to ask whether or not the pointwise duality described by Theorem 2.4 has a corollary in the domain of the densities. Let us explain this point in more detail. For any density $\mathsf{d}$ over $[-1,+\infty]$, define (when it exists) the linear operator $\mathcal{H}(\mathsf{d}) \stackrel{\Delta}{=} \int_{-1}^{+\infty} \mathsf{d}(r)\log_2(1+r)\mathrm{d}r$ and the density $\mathsf{d}^{\perp} = \frac{2}{(1+r)^2}\mathsf{d}(\frac{1-r}{1+r})$ over $[-1,+\infty]$ (density of the Fourier transforms). Then $\mathcal{H}(\mathsf{d}) = 1-\mathcal{H}(\mathsf{d}^{\perp})$. Therefore formally $\left(\mathcal{H}(\mathsf{d}), \mathcal{H}(\mathsf{d}_{i,\mathsf{C}}^{\mathbf{out}(\mathsf{d})})\right) = \left(1-\mathcal{H}(\mathsf{d}^{\perp}), 1-\mathcal{H}(\mathsf{d}_{i,\mathsf{C}^{\perp}}^{\mathbf{out}(\mathsf{d}^{\perp})})\right)$ by Theorem 2.4, where we assume that the previous quantities are well-defined, where $\mathsf{d}$ is the common distribution of the values $r_i^{\mathbf{in}}$, where $\mathsf{d}^{\perp}$ is the common distribution of the values $(\mathcal{F}(r_i^{\mathbf{in}}))$, and where $\mathsf{d}_{i,\mathsf{C}}^{\mathbf{out}(\mathsf{d})}$ is the distribution of the $r_{i,\mathsf{C}}^{\mathbf{out}}$'s with inputs $r_i^{\mathbf{in}}$.
Such a result is a typical duality result in the context of EXIT-like curves. Here duality means symmetry around $(1/2,1/2)$. Observe that, if $\mathsf{d}(z)$ is zero for $z \le 0$, then it defines a "true" transmission channel with entropy $\mathcal{H}(\mathsf{d}) = \mathtt{H}(\mathsf{d})$. Unfortunately the domain $I = [0,+\infty]$ (where $\mathsf{d}$ is non-zero) is mapped into $I^{\perp} = [-1,1]$, and the "dual" channel is not in general a "true" transmission channel. A possible exception is when the channel is BEC($\epsilon$): it has $I = \{0,1\}$ and the dual channel is BEC$(1-\epsilon)$ with $I = \{1,0\}$. This symmetry for the BEC will be stated in (the duality) Theorem 3.3 of Chapter 3.

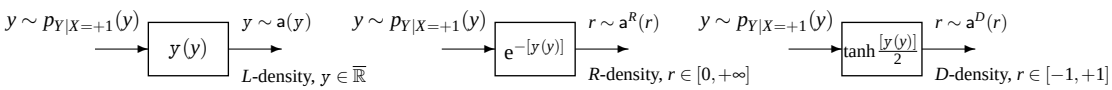



Figure 2.11: Equivalent channels. The $R$-domain and $D$-domain are dual (Fourier transform) of each other. Changes of domain can be obtained as, e.g., $\mathsf{a}^D(r) = \frac{2\mathsf{a}(2\tanh^{-1}(r))}{1-r^2}$ and $\mathsf{a}^R(r) = \frac{\mathsf{a}(-\log(r))}{r}$ where $\mathsf{a}(l)$ is the $L-$density. Left: $L$-density. Middle: $R$-density. Right: $D$-density.

Duality results, different from the previous symmetry, can further be derived. This is shown in the next lemma. This lemma states that the entropy at the output of a parity-check node plus the entropy at the output of a variable node (both with the same two inputs) is equal to the sum of the two input entropies. See [105, 106].

**Lemma 2.6** [Duality Rule For Entropy] Let $\mathsf{a}$ and $\mathsf{b}$ denote two symmetric $L-$densities. Let $X$ and $Y$ have $L$-densities $\mathsf{a}$ and $\mathsf{b}$. Consider the (symmetric) $L$-densities $\mathsf{a}\circledast\mathsf{b}$ and $\mathsf{a}\boxplus\mathsf{b}$, where $\mathsf{a}\circledast\mathsf{b}$ denotes the density of $X+Y$ and $\mathsf{a}\boxplus\mathsf{b}$ denotes the density of $X \boxplus Y$. Then $\mathtt{H}(\mathsf{a}\circledast\mathsf{b}) + \mathtt{H}(\mathsf{a}\boxplus\mathsf{b}) = \mathtt{H}(\mathsf{a}) + \mathtt{H}(\mathsf{b})$.

*Proof.* Let $Z$ have $L-$density $\mathsf{c}$. If $Z \stackrel{\Delta}{=} X \boxplus Y = 2\tanh^{-1}(\tanh(\frac{X}{2})\tanh(\frac{Y}{2}))$ such that $\mathsf{c} = \mathsf{a}\boxplus\mathsf{b}$, then

$$\begin{aligned}
\mathtt{H}(\mathsf{c}) &= \int_{-\infty}^{\infty} \mathsf{c}(z)\log_2(1+e^{-z})\mathrm{d}z \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathsf{a}(x)\mathsf{b}(y)\log_2\left(1+e^{-2\tanh^{-1}(\tanh(x/2)\tanh(y/2))}\right)\mathrm{d}x\,\mathrm{d}y \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathsf{a}(x)\mathsf{b}(y)\log_2\left(\frac{(1+e^{-x})(1+e^{-y})}{1+e^{-x-y}}\right)\mathrm{d}x\,\mathrm{d}y = \mathtt{H}(\mathsf{a}) + \mathtt{H}(\mathsf{b}) - \mathtt{H}(\mathsf{a}\circledast\mathsf{b}),
\end{aligned}$$

where $\mathtt{H}(\mathsf{a}\circledast\mathsf{b}) = \int_{-\infty}^{\infty}\Big(\int_{-\infty}^{\infty}\mathsf{a}(x)\mathsf{b}(y-x)\mathrm{d}x\Big)\log_2(1+e^{-y})\mathrm{d}y = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\mathsf{a}(x)\mathsf{b}(y)\log_2(1+e^{-x-y})\mathrm{d}x\,\mathrm{d}y$. □

Finally let us collect, once and for all, some remarks about the different *domains* (i.e., parameterizations of the decoder inputs) used in this thesis (Further information can be found in [65]). Most of the time we consider the $L$-density, i.e., the density representing the $L$-values under the all-one assumption. The associated channel with corresponding post-processing is depicted in Figure 2.11 (left picture). Several results in Chapter 5 are more easily presented in the $D$-domain, see Figure 2.11 (right picture). Let us give some conversion rules when the $L-$density $\mathsf{a}(y)$ is the reference density. If $y\sim\mathsf{a}(y)$, then $|y|\sim\mathsf{a}^{|L|}(|y|)$ where $\mathsf{a}^{|L|}(z)\triangleq(1+\mathrm{e}^{-z})\mathsf{a}(z)$ is the $|L|$-density. If $r\sim\mathsf{a}^{D}(r)=\frac{2\mathsf{a}(2\tanh^{-1}(r))}{1-r^2}$, then $|r|\sim\mathsf{a}^{|D|}(|r|)$ where $\mathsf{a}^{|D|}(z)\triangleq\frac{2}{1+z}\mathsf{a}^{D}(z)$ is the $|D|$-density. In the $D$-domain the channel symmetry reads $\mathsf{a}^{D}(-z)=\frac{1-z}{1+z}\mathsf{a}^{D}(z)$.

## 2.C Relations between Various Thresholds

The inequalities between BP, MAP and Shannon thresholds are trivial to see. It is more difficult to see how the threshold obtained from the stability condition is related to the previous quantities. This is relatively easy to show for the case of the erasure channel.

**Lemma 2.7** Assume that transmission takes places over BEC$(\epsilon)$. Given a dd pair $(\lambda,\rho)$, we have the relations

$$\epsilon^{\mathrm{BP}}\leq\epsilon^{\mathrm{MAP}}\leq\min\{\epsilon^{\mathrm{SH}},\epsilon^{\mathrm{SC}}\},$$

where $\epsilon^{\mathrm{SH}}$ and $\epsilon^{\mathrm{SC}}\triangleq\frac{1}{\lambda'(0)\rho'(1)}$ denote, respectively, the Shannon and stability condition thresholds.

*Proof.* As discussed in this chapter, $\epsilon^{\mathrm{BP}}\leq\epsilon^{\mathrm{MAP}}$ follows from the sub-optimality of BP decoding, Moreover, $\epsilon^{\mathrm{MAP}}\leq\epsilon^{\mathrm{SH}}\leq 1-r_{\Xi}$ follows from a rephrasing of the channel coding theorem and its strong converse. Finally $\epsilon^{\mathrm{MAP}}\leq\epsilon^{\mathrm{SC}}=1/(\lambda'(0)\rho'(1))$ can be proved through the following graph-theoretic argument. Assume, by contradiction that $\epsilon^{\mathrm{MAP}}>\epsilon^{\mathrm{SC}}$ and let $\epsilon$ be such that $\epsilon^{\mathrm{SC}}<\epsilon<\epsilon^{\mathrm{MAP}}$. Notice that $\epsilon^{\mathrm{SC}}<\epsilon$ is equivalent to $\epsilon\lambda'(0)\rho'(1)>1$. Consider now the peeling decoder and the residual graph once the received variable nodes have been erased. Focus on the subgraph of degree 2 variable nodes. This (bipartite) 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 as shown in [109]. 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 exists a codeword such that $x_i=1$: just set $x_j=1$ if $j$ belongs to some fixed loop through $i$ and 0 otherwise. As there is a finite fraction of such vertices $\liminf_{n\to\infty}(\mathbb{E}[H(X|Y)]/n)$ and therefore $\epsilon>\epsilon^{\mathrm{MAP}}$. We have reached a contradiction, therefore $\epsilon^{\mathrm{MAP}}\leq\epsilon^{\mathrm{SC}}$ as claimed. □

**Overview:** The definition and basic properties of EXIT functions are reviewed. EXIT functions are the starting point of this thesis.

# 3 | EXIT Functions

The rediscovery of iterative decoding [9,10] in [62,73,110] was heavily based on the notion of *extrinsic* estimate. The remaining uncertainty on an individual symbol given the information provided by all other received values is a natural measure of the performance associated with a code. This observation guides the definition of EXIT functions [33].

## 3.1 Definition and Linear Functional

EXIT functions [33] (see also [111–114]) measure the residual uncertainty associated with a given symbol based on the remaining observations. They were originally derived from the picture of a "soft-in soft-out" receiver [71, 72] and they can be considered as "transfer functions" because they give the residual uncertainty at the "exit" of the decoder.

**Definition 3.1** [EXIT Value] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over the channel $p_{Y|X}$. Let $Y$ be the received random vector of length $n$, and let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Consider $i \in [n]$. Define $h_i \triangleq H(X_i|Y_{\sim i}, \Omega)$. This estimator is called the $i^{\text{th}}$ EXIT value.

The concept of EXIT estimators is quite general. Nevertheless, as in the rest of the thesis, we focus on binary channels for notational simplicity. In this context, let us recall some notations from Chapter 2. Assume that the channel is memoryless. Then the random *extrinsic estimator* is $\Phi_i^{\text{MAP}} \triangleq \log\left(\frac{p_{X_i|Y_{\sim i},\Omega}(+1|Y_{\sim i},\Omega)}{p_{X_i|Y_{\sim i},\Omega}(-1|Y_{\sim i},\Omega)}\right)$ and takes on values $\phi_i^{\text{MAP}}(y_{\sim i},\omega) \triangleq \log\left(\frac{p_{X_i|Y_{\sim i},\Omega}(+1|y_{\sim i},\omega)}{p_{X_i|Y_{\sim i},\Omega}(-1|Y_{\sim i},\omega)}\right)$. We have seen in Chapter 2 that $\Phi_i^{\text{MAP}}$ is a sufficient statistic[1] for estimating $X_i$. This quite intuitive fact is used in the next lemma.

**Lemma 3.1** [(MAP) EXIT Value: Alternative Characterization] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over the channel $p_{Y|X}$. Let $Y$ be a received random vector of length $n$, and let $\Omega$ be a further observation of $X$. Assume that $\Omega \to X \to Y$ and $Y_i \to X_i \to Y_{\sim i}$. Then $h_i^{\text{MAP}} \triangleq h_i = H(X_i|\Phi_i^{\text{MAP}}, \Omega)$.

[1] Implied by the memoryless assumption, the hypothesis $Y_i \to X_i \to Y_{\sim i}$ suffices to define the extrinsic MAP estimator and to show that $\Phi_i^{\text{MAP}}$ is a sufficient statistic. This is demonstrated in Example 2.11 when there is no extra observation $\Omega$. This can be strengthened to include a fixed observation $\Omega$ as shown in Appendix 3.A. In the non-binary case (see Section 2.11) the statement stays literally unchanged, as well as Lemma 3.1.

In the previous lemma we wrote "(MAP)" EXIT value to emphasize that we can replace $Y_{\sim i}$ with the extrinsic estimate $\Phi_i^{\text{MAP}}$. The concept of EXIT estimators is more meaningful if we consider transmission over channels parametrized by a *common* parameter $\mathtt{p}$ so that the EXIT estimator becomes a function of a (typically, single-valued) variable $\mathtt{p}$. With this aim, let us consider transmission over the channel family $\{\{\text{BMSC}_i(\mathtt{h}_i(\mathtt{p}))\}_i\}_{\mathtt{p}}$ which means that the $i^{\text{th}}$ bit experiences the channel entropy $\mathtt{h}_i(\mathtt{p})$. A typical example is when, for all $i$, the channel entropy is identical, i.e., $\mathtt{h}_i(\mathtt{p}) = \mathtt{h}_1(\mathtt{p}) = \mathtt{h}(\mathtt{p})$, and monotonic with respect to $\mathtt{p}$, e.g., $\mathtt{h}_i(\mathtt{p}) = \mathtt{p}$.

**Definition 3.2** [(MAP) EXIT Function] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$. Let $Y$ be the received random vector of length $n$, and let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Define

$$h_i^{\text{MAP}}(\mathtt{h}_{\sim i}) \triangleq H(X_i|Y_{\sim i}(\mathtt{h}_{\sim i}),\Omega), \qquad h^{\text{MAP}}(\mathtt{h}_{\sim i}) \triangleq \frac{1}{n}\sum_{i=1}^{n} h_i^{\text{MAP}}(\mathtt{h}_{\sim i}) = \frac{1}{n}\sum_{i=1}^{n} H(X_i|Y_{\sim i}(\mathtt{h}_{\sim i}),\Omega).$$

The function $h^{\text{MAP}}$ ($h_i^{\text{MAP}}$) is the multi-variate EXIT (respectively, $i^{\text{th}}$ EXIT) function. If the individual channel entropies $\mathtt{h}_i = H(X_i|Y_i)$ are all parametrized by a scalar $\mathtt{p} \in P \subseteq \mathbb{R}$ such that $\mathtt{h}_i = \mathtt{h}_i(\mathtt{p})$, then $h_i^{\text{MAP}}(\mathtt{p}) \triangleq H(X_i|Y_{\sim i}(\mathtt{p}),\Omega)$, and $h^{\text{MAP}}(\mathtt{p}) \triangleq \frac{1}{n}\sum_{i=1}^{n} h_i^{\text{MAP}}(\mathtt{p}) = \frac{1}{n}\sum_{i=1}^{n} H(X_i|Y_{\sim i}(\mathtt{p}),\Omega)$ are simply the $i^{\text{th}}$ (MAP) EXIT function and (MAP) EXIT function, respectively.

Let us make two more remarks concerning Lemma 3.1. First, if no extra observation is added or if the underlying channel is symmetric, then the MAP estimator $\phi^{\text{MAP}}(Y_{\sim i})$ or $\phi^{\text{MAP}}(Y_{\sim i},\Omega)$ has a symmetric $L-$density. See Chapter 2 and Appendix 3.A. In the sequel we will assume this to be the case. Second, Lemma 3.1 permits us to enlarge the notion of (MAP) EXIT estimator and function. The definition extends naturally to *any* (extrinsic) decoder that is denoted by the shorthand DEC and whose associated estimator is $\Phi_i^{\text{DEC}} \triangleq \phi_i^{\text{DEC}}(Y_{\sim i},\Omega)$. An important example of extrinsic DEC estimator is the BP estimator if we define it as $\Phi_i^{\text{BP},\ell} \triangleq \phi_i^{\text{BP},\ell}(Y_{\sim i})$ at iteration $\ell$. Notice that, in our definition, $\phi_i^{\text{BP},\ell}(y_{\sim i})$ does not include the $i^{\text{th}}$ received $y_i$ by construction.[2]

**Definition 3.3** [EXIT Function] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$. Let $Y$ be the received random vector of length $n$, and let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Consider any estimator $\Phi_i^{\text{DEC}} = \phi_i^{\text{DEC}}(Y_{\sim i})$. Define

$$h_i^{\text{DEC}}(\mathtt{h}_{\sim i}) \triangleq H(X_i|\Phi_i^{\text{DEC}}(\mathtt{h}_{\sim i}),\Omega), \qquad h^{\text{DEC}}(\mathtt{h}_{\sim i}) \triangleq \frac{1}{n}\sum_{i=1}^{n} h_i^{\text{DEC}}(\mathtt{h}_{\sim i}) = \frac{1}{n}\sum_{i=1}^{n} H(X_i|\Phi_i^{\text{DEC}}(\mathtt{h}_{\sim i}),\Omega).$$

The function $h^{\text{DEC}}$ ($h_i^{\text{DEC}}$) is the multi-variate EXIT (respectively, $i^{\text{th}}$ EXIT) function associated with the extrinsic DEC estimator. If the individual channel entropies $\mathtt{h}_i = H(X_i|Y_i)$ are all parametrized by a scalar $\mathtt{p} \in P \subseteq \mathbb{R}$ such that $\mathtt{h}_i = \mathtt{h}_i(\mathtt{p})$, then the function becomes function of a single scalar parameter.

Discussion: Our definition of EXIT functions differs only in a trivial way from the original definition in [33]. More precisely, EXIT functions were originally defined as $I(X_i|Y_{\sim i}(\mathtt{h}(\mathtt{p})),\Omega) = H(X_i) - H(X_i|Y_{\sim i}(\mathtt{h}(\mathtt{p})),\Omega)$. If $X_i$ is binary and has equal priors, then $H(X_i) = 1$. In this case the EXIT curve $(\mathtt{h}(\mathtt{p}), h_i^{\text{MAP}}(\mathtt{h}(\mathtt{p})))$ according to our definition is simply the original one as introduced in [33] but flipped around the diagonal. In applications we deal mainly with *proper* binary linear codes (see Appendix 2.A) which satisfy $H(X_i) = 1$ for all $i$.

**Lemma 3.2** [(MAP) EXIT: Operational Characterization] Let $X$ be chosen uniformly at random from a proper binary linear code of length $n$. Assume that transmission takes place over the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$. Let $\mathtt{a}_i^{\text{MAP}}$ denote the density of $\Phi_i^{\text{MAP}} = \phi_i^{\text{MAP}}(Y_{\sim i})$ assuming that the all-one codeword

[2] If the (finite) graph $\mathtt{G}$ has cycles, then the "true" BP estimate (which is received by the $i^{\text{th}}$ variable node) is potentially a function of $y_i$. However, for a fixed number of iterations and in the limit of large blocklengths, the two BP estimates (i.e., based either on $(y_i, y_{\sim i})$ or on $(\ast, y_{\sim i})$) coincide with high probability, see, e.g., Chapter 4 and Chapter 6. The definition of the BP extrinsic estimate as a function of $y_{\sim i}$ simplifies the analysis.

was transmitted. Then

$$h_i^{\text{MAP}}(\mathtt{h}_{\sim i}) = \mathtt{H}(\mathtt{a}_i^{\text{MAP}}),$$

where $\mathtt{H}(\mathtt{a}) = \int \mathtt{a}(y)\log_2(1+\mathrm{e}^{-y})\mathrm{d}y = \mathbb{E}_Y[\log_2(1+\mathrm{e}^{-Y})]$ is the entropy operator of Definition 2.5.

*Proof.* Lemma 2.4 shows that assuming $X$ is chosen uniformly at random from a proper binary linear code $\mathtt{C}$, the binary channel $p_{\Phi_i|X_i}$ is symmetric. Further, note that $\Phi_i$ is already in the $L$-domain, therefore its density conditioned on $X_i = 1$ is already the $L$-density conditioned on $X_i = 1$. Assume temporarily that this density is equal to the density of $\Phi_i$ when the all-one codeword $\underline{1}$ is transmitted. Let $\mathtt{a}_i$ denote this density. From Lemma 2.1 and definition 2.5, we conclude that $H(X_i|\Phi_i) = \mathtt{H}(\mathtt{a}_i)$. It now remains to prove that $\mathtt{a}_i$ is equal to the density of $\Phi_i$ assuming that $\underline{1}$ was transmitted (which is already implicit in Lemma 2.4). To see this, note that, using the symmetry of the channel and the equal priors of the codewords (together with the fact that the code is proper), for a fixed $y$ we can write $p_{X_i|Y}(x_i|y) \propto \sum_{\tilde{x}\in\mathtt{C}:\tilde{x}_i=x_i} p_{Y|X}(y\tilde{x}|\underline{1})$ (the product involving the vectors $y$ and $\tilde{x}$ denotes the componentwise product). In a similar manner, if $x' \in \mathtt{C}$, then $p_{X_i|Y}(x_i|yx') \propto \sum_{\tilde{x}\in\mathtt{C}:\tilde{x}_i=x_i x'_i} p_{Y|X}(yx'\tilde{x}|x')$. Compare the density of the LLR assuming the codeword $\underline{1}$ was transmitted to the one assuming that the codeword $x'$ was transmitted. The claim follows by noting that for any received vector $y$, $p_{Y|X}(y|\underline{1}) = p_{Y|X}(yx'|x')$, and that in this case also $p_{Y|X}(y\tilde{x}|\underline{1}) = p_{Y|X}(yx'\tilde{x}|x')$. □

Discussion: The function $l(y) \triangleq \log_2(1+\mathrm{e}^{-y})$ is sometimes called *EXIT kernel* and the entropy operator is in fact an "EXIT operator."
Again (under the technical conditions used in the previous proof) we can enlarge the domain of applications of the previous lemma to include alternative estimators.

**Lemma 3.3** [EXIT: Operational Characterization] Let $X$ be chosen uniformly at random from a proper binary linear code of length $n$. Assume that transmission takes place over the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$. Consider an additional observation $\Omega$ such that $\Omega \to X \to Y$. Consider any estimator $\Phi_i^{\text{DEC}} = \phi_i^{\text{DEC}}(Y_{\sim i}, \Omega)$ that preserves channel symmetry. Let the density of $\Phi_i^{\text{DEC}}$ under the assumption that the all-one codeword was transmitted be $\mathtt{a}_i^{\text{DEC}}$. Then

$$h_i^{\text{DEC}}(\mathtt{h}_{\sim i}) = \mathtt{H}(\mathtt{a}_i^{\text{DEC}}),$$

where $\mathtt{H}(\mathtt{a}) = \int \mathtt{a}(y)\log_2(1+\mathrm{e}^{-y})\mathrm{d}y$ is the entropy operator (and $l(y) \triangleq \log_2(1+\mathrm{e}^{-Y})$ the EXIT kernel).

In the binary case, notice that the EXIT function is a quantity between 0 and 1. In most applications, it is be a non-decreasing function of the channel entropy $\mathtt{h}(\mathtt{p})$ as shown in the next fact.

**Fact 3.1** [Monotonicity over Ordered Channels] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$. Assume that all bits are transmitted over a channel $\text{BMSC}(\mathtt{p})$ and that for all $\mathtt{p}$ we have $\Omega \to X \to Y(\mathtt{p})$. If the channel family $\{\text{BMSC}(\mathtt{p})\}_{\mathtt{p}}$ is ordered and complete (see Section 2.8), then the function $h_i^{\text{MAP}}(\mathtt{h}) = H(X_i|Y_{\sim i}(\mathtt{h}), \Omega)$ is non-decreasing for $\mathtt{h} \in [0,1]$.

*Proof.* Fix $(\mathtt{h}_1, \mathtt{h}_2)$ such that $\mathtt{h}_1 < \mathtt{h}_2$. Since the family of channels is complete (i.e., $\mathtt{h}$ ranges from 0 to 1) and ordered by physical degradation, then $\exists \mathtt{p}_1, \mathtt{p}_2$ such that $\mathtt{p}_1 < \mathtt{p}_2$, $\mathtt{h}_1 = H(X|Y(\mathtt{p}_1))$ is the entropy of $\text{BMSC}(\mathtt{p}_1)$, $\mathtt{h}_2 = H(X|Y(\mathtt{p}_2))$ is the entropy of $\text{BMSC}(\mathtt{p}_2)$, and $\text{BMSC}(\mathtt{p}_2)$ is physically degraded with respect to $\text{BMSC}(\mathtt{p}_1)$. Since the channels are memoryless and degraded, we get $X_i \to Y_{\sim i}(\mathtt{p}_1) \to Y_{\sim i}(\mathtt{p}_2)$. Therefore $H(X_i|Y_{\sim i}(\mathtt{p}_1), \Omega) = H(X_i|Y_{\sim i}(\mathtt{p}_1), Y_{\sim i}(\mathtt{p}_2), \Omega) \leq H(X_i|Y_{\sim i}(\mathtt{p}_2), \Omega)$ using Markovity in the first equality and the fact that conditioning reduces entropy in the second. □

Discussion: Notice first that the EXIT monotonicity comes from the data processing inequality. Second, observe that, if the channel family is complete, then the MAP EXIT functions is a non-decreasing (possibly piecewise constant) mapping from $[0,1]$ to $[0,1]$. Observe also that a similar property is true for any $h^{\text{DEC}}$ associated with any estimator "DEC" that preserves the order implied by physical degradation.
Finally, the following characterization using the all-one codeword assumption is helpful for deriving explicit EXIT functions.

**Fact 3.2** [Alternative Characterization] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$. Assume that, for all $i$, the $i^{\text{th}}$ bit is transmitted over $\text{BMSC}_i(\mathtt{h}_i(\mathtt{p}))$ and $\Omega \to X \to Y$. Then $h_i^{\text{MAP}} = H(X_i|Y_{\sim i},\Omega) = \int_{y_{\sim i}} p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}|\underline{1}) H(X_i|Y_{\sim i}=y_{\sim i}X_{\sim i},\Omega)(\mathrm{d}y)_{\sim i}$ where the product of vectors is defined as the component-wise product.

*Proof.* Let us assume that $\mathcal{Y}$ is discrete. Under the BMSC assumption, the expansion of the entropy rule reads

$$H(X_i|Y_{\sim i},\Omega) = \sum_{y_{\sim i}} p_{Y_{\sim i}}(y_{\sim i}) H(X_i|Y_{\sim i}=y_{\sim i},\Omega) = \sum_{x_{\sim i}} p_{X_{\sim i}}(\underline{1}) \sum_{y_{\sim i}} p_{Y_{\sim i}|X_{\sim i}}(y_{\sim i}x_{\sim i}|\underline{1}) H(X_i|Y_{\sim i}=y_{\sim i},\Omega).$$

The proof can be concluded by the change of variable $y_{\sim i} \leftarrow y_{\sim i}x_{\sim i}$, followed by reordering of the sums. □

Assume that transmission takes place over the channel family $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}$ so that each bit is passed through the same BMS channel. Using the previous lemma it is relatively easy to see that repetition codes, single parity-check codes or cyclic codes (e.g., Hamming codes) have individual EXIT functions that are independent of the location $i$. This is investigated in [115, 116] where such codes (more exactly, codes for which the extrinsic density $\mathtt{a}_i^{\text{MAP}}$ is independent of $i$) are called *isotropic*. Let us now give simple examples of EXIT functions.

**Example 3.1** [EXIT Function for Maximum Distance Separable Codes over BMS Channels]

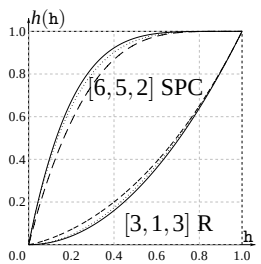



Figure 3.1: EXIT functions for $[3,1]$ R and $[6,5]$ SPC codes on $\text{BEC}(\mathtt{h})$ (solid), $\text{BSC}(\mathtt{h})$ (dashed) and $\text{BAWGNC}(\mathtt{h})$ (dotted).

Figure 3.1 shows the MAP EXIT curve $\mathtt{h} \mapsto h^{\text{MAP}}(\mathtt{h}) = h_1^{\text{MAP}}(\mathtt{h})$ for the $[3,1,3]$ repetition (R) code, as well as for the $[6,5,2]$ single parity-check (SPC) code over $\text{BEC}(\mathtt{h})$, $\text{BSC}(\mathtt{h})$, and $\text{BAWGNC}(\mathtt{h})$. E.g., over $\text{BSC}(\mathtt{h})$, the EXIT function for the $[n+1,n,2]$ single parity-check code is given by $h^{\text{MAP}}(\mathtt{h}) = h_1^{\text{MAP}}(\mathtt{h}) = h_2\Big(\frac{1-(1-2\epsilon)^n}{2}\Big)$, where $\epsilon = {h_2}^{-1}(\mathtt{h})$, and the EXIT function for the $[n+1,1,n+1]$ repetition code is given by $h^{\text{MAP}}(\mathtt{h}) = h_1^{\text{MAP}}(\mathtt{h}) = \sum_{i=0}^{n} \binom{n}{i} \bar{\epsilon}^{n-i}\epsilon^i \log_2\left(1+(\epsilon/\bar{\epsilon})^{n-2i}\right)$ where $\epsilon = {h_2}^{-1}(\mathtt{h})$ and $\bar{\epsilon} = 1-\epsilon$, over BAWGNC, the EXIT function for the $[n+1,1,n+1]$ repetition code is given by $h^{\text{MAP}}(\mathtt{h}) = h_1^{\text{MAP}}(\mathtt{h}) = \mathtt{H}(\mathtt{a}_{\text{BAWGNC}(\mathtt{h})}^{\circledast n})$ where $\mathtt{H}$ is the entropy operator introduced in Definition 2.5 and $\mathtt{a}_{\text{BAWGNC}(\mathtt{h})}$ is the Gaussian $L-$density.

In the rest of this chapter, when there is no risk of confusion, we skip the superscripts MAP for (MAP) EXIT functions.

## 3.2 EXIT Chart Method

EXIT functions were originally invented to be used in the so-called *EXIT charts*, see [33]. The purpose of EXIT charts is to provide a practical tool to design and optimize iterative coding systems. The original idea behind the EXIT chart *method* is to *approximate* the decoding process by a one-dimensional representation, see possible alternatives, e.g., in [45, 117–121]. This approximation is then visualized on the basis of EXIT charts.

We can justify this approach as follows. From a formal standpoint, the principle consists in *projecting* the average trajectory of density evolution onto a 2-dimensional plane using a *linear operator*. The trace of the decoding trajectory becomes a staircase function (assuming that we are using the BP schedule described in Section 2.5) between two EXIT curves. More precisely, consider the density evolution analysis, see [14, 15, 65]. Let us shortly review the main aspects. The *ensemble* performance of $\text{LDPC}(\lambda,\rho)$ is studied in *average* and in the *asymptotic* limit when first the blocklength $n$ tends to infinity and second the number of iterations $\ell$ goes to infinity. This average performance can be computed on the associated infinite tree, called the *computation tree*. *Concentration* results (see similar statements in Appendix 4.A) indicate further that this limiting object is the correct description of a particular instance of transmission with high probability (going to one when $n$ tends to infinity). The density evolution analysis is in general simplified by the all-one codeword assumption, the channel symmetry, and the decoder symmetry.

Let us exemplify this analysis in our context. Assume that the all-one codeword is transmitted over a BMSC using the dd pair $(\lambda(\mathtt{x}),\rho(\mathtt{x}))$. The channel outputs the $L-$density $\mathsf{c}$. Consider BP decoding and the average asymptotic behavior. During the decoding process, function or variable nodes locally perform a MAP decoding that preserves the symmetry of the $L$-densities (see Chapter 2 or [65]). It can easily be shown that this symmetry is also preserved in the dual domain (see Fourier transform in Appendix 2.B). Therefore the densities of all the intermediate messages are symmetric. Let us denote $\mathsf{a}_\ell$ ($\mathsf{b}_\ell$) to be the variable-to-function (function-to-variable, respectively) density at iteration $\ell$. The initial density is $\mathsf{a}_0=\Delta_0$, and for $\ell\geq 0$, $\mathsf{a}_{\ell+1}=\mathsf{c}\circledast\big(\sum_j\lambda_j\mathsf{b}_\ell^{\circledast(j-1)}\big)$ with $\mathsf{b}_\ell=\sum_j\rho_j\mathsf{a}_\ell^{\boxtimes(j-1)}$, where $\circledast$ denote the standard convolution and $\boxtimes$ the convolution in the dual domain. Figure 3.2 depicts a projection of the decoding process using the entropy (or EXIT) operator of Definition 2.5.

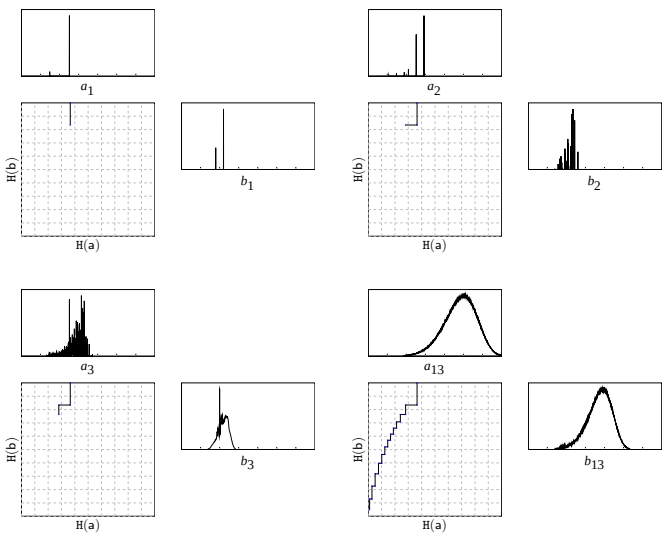



Figure 3.2: Density evolution and the associated EXIT points for the $(3,6)$-regular ensemble over BSC$(0.07)$ at BP iteration $\ell=1,2,3$ and $\ell>>1$.

More precisely, let $y_\ell\triangleq\mathtt{H}(\mathsf{b}_\ell)$ ($x_\ell\triangleq\mathtt{H}(\mathsf{a}_\ell)$) be the entropy of the messages emitted at the function nodes (variables nodes) at the $\ell^{\text{th}}$ iteration. The sequence $\{x_\ell,y_\ell\}_\ell$ is represented by a staircase function that reflects the trajectory of density evolution in the plane of the entropies. As an example consider transmission over BSC$(\epsilon=0.07)$ and the ensemble LDPC$(x^2,x^5)$: Figure 3.2 depicts the density evolution process and its projection in the plane of the entropies.

In general (up to a few notable exceptions such as the erasure channel), the densities $\mathsf{a}_\ell$ (or $\mathsf{b}_\ell$) do not have simple descriptions after a finite number of iterations. This makes density evolution difficult to handle analytically. In a similar manner, the entropies or EXIT functions associated with the true intermediate densities are difficult to handle analytically.

The main idea behind the EXIT chart method therefore is to replace at each iteration $\ell$ the intermediate densities in the density evolution process with an "equivalent" density chosen from some "suitable family of densities." The most "faithful" equivalence rule is to choose the element of the channel family that has *equal entropy*. We further "hope" that the convergence of iterative decoding is "robust" to such a replacement. This approximation is called the EXIT chart method. In other words, instead of tracking the full density evolution process and projecting it as a staircase function between two boundary EXIT curves (as in Figure 3.2), the EXIT chart method *conjectures* that two approximated boundary curves suffice to describe "faithfully" the decoding process.

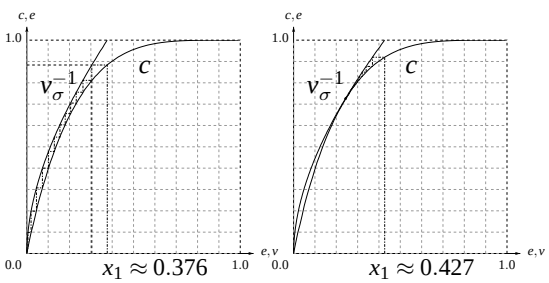



Figure 3.3: EXIT chart method over BAWGNC$(\sigma)$: (3,6)-regular ensemble in the Gaussian approximation for $c$ and $v$. Left: $\sigma=0.816$. Right: $\sigma=0.878$.

It remains to choose a "suitable family of densities" that we want to parametrize by a scalar. (This scalar is chosen to be the entropy in the context of the EXIT chart method. It could also be obtained, however, from other linear operators, see, e.g., Section 5.3). It is standard to choose a family of symmetric Gaussian densities; the resulting approximation is called the *Gaussian approximation*.

Let us explicitly write down the equations for this case according to the EXIT chart method. Assume that transmission takes place over BAWGNC$(\sigma)$ such that the $L$-density $\mathsf{c}$ is Gaussian with mean $2/\sigma^2$ and variance $4/\sigma^2$. Let $\mathsf{g}_m$ denote a generic $L$-density that is Gaussian with mean $m$ and variance $2m$,

and let $f(m) \stackrel{\triangle}{=} \mathtt{H}(\mathtt{g}_m)$ denote the associated entropy. With these notations, the channel has entropy $\mathtt{H}(\mathtt{c}) = \mathtt{H}(\mathtt{g}_{2/\sigma^2})$. First let $e$ be the entropy entering a variable node and define the function $v_\sigma(e) \stackrel{\triangle}{=} \sum_j \lambda_j f\big((j-1) f^{-1}(e) + \frac{2}{\sigma^2}\big)$. The function $v_\sigma$ describes the output entropy at a variable node. Consider a variable node of degree $i$: Assume that the entropy of the incoming message density equals $e$ and that this density is a symmetric Gaussian. Since all inputs are symmetric Gaussian $L$-densities, the output is a symmetric Gaussian as well. Such a Gaussian $L$-density is uniquely determined by its mean. By assumption the message density has mean $f^{-1}(e)$ and the channel density has mean $2/\sigma^2$. The mean of the output is therefore $(i-1) f^{-1}(x) + \frac{2}{\sigma^2}$. Hence the associated entropy is $f\big((i-1) f^{-1}(x) + \frac{2}{\sigma^2}\big)$ and the claim follows by averaging over the edge degrees. Now let $e$ denote the entropy entering a function node. Similar to $v$, a function $c(e)$ describes the output entropy at a check node. In practice, instead of the true $c(e)$, it is common to use the dual approximation[3] $c(x) \approx 1 - \sum_j \rho_j f\big((j-1) f^{-1}(1-e)\big)$. The postulate of the EXIT chart method is that the true entropy $x_\ell$ ($y_\ell$) at the output of the variable (function) nodes is well-approximated as the sequence $x_{\ell+1} \approx v_\sigma(c(x_\ell))$ where $x_0 = \mathtt{H}(\mathtt{c}) = \mathtt{H}(\mathtt{g}_{2/\sigma^2})$.

**Example 3.2** [EXIT Chart Method Applied to the (3,6)-Regular Ensemble] Strictly speaking, an EXIT chart is a diagram as shown in Figure 3.3. It shows the density evolution process according to the EXIT chart method for the two parameters $\sigma = 0.816$ and $\sigma = 0.878$. To construct this chart, plot $\{(\mathtt{h}, c(\mathtt{h}))\}_{\mathtt{h}\in[0,1]}$ which describes the entropy evolution at the function nodes and $\{(\mathtt{h}, v_\sigma^{-1}(\mathtt{h}))\}_{\mathtt{h}\in[0,1]} = \{(v_\sigma(\mathtt{h}), \mathtt{h})\}_{\mathtt{h}\in[0,1]}$ which describes the progress at the variable nodes. The approximate density evolution is now easily read off from this picture by constructing the staircase function associated with the recursive sequence $\tilde{x}_{\ell+1} = v_\sigma(c(\tilde{x}_\ell))$ with $\tilde{x}_1 = \mathtt{H}(\mathtt{c})$. For example $\tilde{x}_1 \approx 0.3765$ if $\sigma \approx 0.816$. According to the EXIT chart method, the entropy at the output of the function nodes is then $c(0.3765) \approx 0.8835$. We can construct this value graphically if we look for the intersection of the vertical line located at $0.3765$ with the curve $(e, c(e))$. This entropy now enters the variable nodes and according to the EXIT chart method the entropy at the output of the variable nodes is equal to $v_{\sigma=0.816}(c(0.3765)) = v_{\sigma=0.816}(0.8835) \approx 0.3045$. Again we can construct this value graphically using the function $v_\sigma^{-1}(e)$. If we iterate this procedure, the corner points of the resulting staircase function describe the progress of density evolution according to the approximated EXIT chart method. We see from Figure 3.3 that for $\sigma \approx 0.816$ the staircase function eventually reaches the point $(0,0)$, corresponding to successful decoding. This is no longer the case for $\sigma \approx 0.878$, therefore $\sigma \approx 0.878$ is the critical threshold according to the EXIT chart method. Note that this parameter differs only slightly from the true value of the BP threshold that is $\sigma^{\mathrm{BP}} \approx 0.88$.

The EXIT chart method is very popular because it gives immediate insight on how to *optimize* iterative coding systems and it is easily computed in practice. In the EXIT chart methodology the condition for progress is $v_\sigma(c(e)) < e$ at each iteration. This formulation is *linear* in $\lambda$ once the function nodes (and their distribution) are fixed. We can therefore optimize the left distribution by techniques from linear programming (In the same manner we could first fix $v_\sigma$ and then optimize $c$ using linear programming). Basically all known optimization methods so far rely to some degree on this simple principle.[4] See, e.g., [12–15, 33, 112, 122–130]. The excellent results given by EXIT optimizations in a first approach and the insight they provide indicate that EXIT functions might be more than a simple practical optimization tool. The next section provides a first step towards a theoretical justification.

## 3.3 Universal Bounds

Note that the first EXIT functions we depicted in Figure 3.1 were "ordered." More precisely, for a repetition code we get the highest extrinsic entropy at the output for the channel family $\{\mathrm{BSC}(\mathtt{h})\}_{\mathtt{h}\in[0,1]}$ and we get the lowest such entropy if we use instead the family $\{\mathrm{BEC}(\mathtt{h})\}_{\mathtt{h}\in[0,1]}$. Indeed, the next

[3] See also [118]. The dual approximation is motivated by the duality theorem which is exact for the BEC. The fact that here we use an approximation of the output entropy rather than an exact expression does little harm. The approximation appears to be accurate in practice and the EXIT method is anyway an approximate method. The small additional error incurred by using the dual approximation is therefore easily outweighed by the advantage of being able to write down a pleasing analytic expression.

[4] In Chapter 6 the direct optimization on the EBP GEXIT curve via linear programming is an alternative in order to optimize iterative coding systems.

theorem shows that these two families are the *least* and *most* "informative" families of channels over the whole class of BMSCs for a repetition code, as conjectured in [118] and proved in [105, 106, 131–134]. The roles are exactly exchanged at a check node.

**Theorem 3.1** [Extremes of Information Combining] Consider any two BMS channels represented by the $L$-densities $\mathsf{a}$ and $\mathsf{b}$. For $\mathtt{h} \in [0,1]$, let $\mathsf{d}_{\mathrm{BEC}(\mathtt{h})}$ and $\mathsf{d}_{\mathrm{BSC}(\mathtt{h})}$ be the $L$-densities associated with the BEC and BSC when the channel entropy is $\mathtt{h}$. Then

$$\mathtt{H}(\mathsf{a} \boxledast \mathsf{d}_{\mathrm{BSC}(\mathtt{H}(\mathsf{b}))}) \leq \mathtt{H}(\mathsf{a} \boxledast \mathsf{b}) \leq \mathtt{H}(\mathsf{a} \boxledast \mathsf{d}_{\mathrm{BEC}(\mathtt{H}(\mathsf{b}))}) = 1 - (1 - \mathtt{H}(\mathsf{a}))(1 - \mathtt{H}(\mathsf{d}_{\mathrm{BEC}(\mathtt{H}(\mathsf{b}))})),$$

$$\mathtt{H}(\mathsf{a})\mathtt{H}(\mathsf{d}_{\mathrm{BEC}(\mathtt{H}(\mathsf{b}))}) = \mathtt{H}(\mathsf{a} \circledast \mathsf{d}_{\mathrm{BEC}(\mathtt{H}(\mathsf{b}))}) \leq \mathtt{H}(\mathsf{a} \circledast \mathsf{b}) \leq \mathtt{H}(\mathsf{c} \circledast \mathsf{d}_{\mathrm{BSC}(\mathtt{H}(\mathsf{b}))}).$$

*Proof.* We only need to show the result for the parity-check $\boxledast$-convolution. The equivalent result for the regular $\circledast$-convolution follows from the duality rule for entropy in Lemma 2.6. For any cross-over probability $\epsilon$, let $\mathsf{c}_{\mathrm{BSC}(\epsilon)} = \mathsf{d}_{\mathrm{BSC}(h_2(\epsilon))}$ be the $L-$density associated with BSC$(\epsilon)$. Any BMS channel can be written as an infinite convex combination of BSCs. Therefore there exist two density functions $w_a(u)$ and $w_b(u)$ such that $\mathsf{a}(z) = \int_0^{\frac{1}{2}} w_a(u)\mathsf{c}_{\mathrm{BSC}(u)}(z)\mathrm{d}u$, and $\mathsf{b}(z) = \int_0^{\frac{1}{2}} w_b(u)\mathsf{c}_{\mathrm{BSC}(u)}(z)\mathrm{d}u$. Since the entropy operator $\mathtt{H}$ and the $\boxledast$-convolution are linear in their arguments, it follows that

$$\begin{aligned}\mathtt{H}(\mathsf{a} \boxledast \mathsf{b}) &= \int\int w_a(u_a) w_b(u_b) \mathtt{H}\big(\mathsf{c}_{\mathrm{BSC}(u_a)} \boxledast \mathsf{c}_{\mathrm{BSC}(u_a)}\big) \mathrm{d}u_a \mathrm{d}u_b \\ &= \int w_a(u_a) \left( \int w_b(u_b) h_2(u_b(1-2u_a)+u_a) \mathrm{d}u_b \right) \mathrm{d}u_a. \end{aligned} \tag{3.1}$$

where the last equality comes from $\mathtt{H}\big(\mathsf{c}_{\mathrm{BSC}(u_a)} \boxledast \mathsf{c}_{\mathrm{BSC}(u_a)}\big) = h_2(u_b(1-u_a)+u_a(1-u_b))$. To see this observe that, if BSC$(u_a)$ and BSC$(u_b)$ represent two densities entering a check node, then the output density is again a BSC with parameter $u_b(1-u_a)+u_a(1-u_b)$.
Now it suffices to use twice the convexity of $f(e) \triangleq h_2(h_2^{-1}(e)(1-2u_a)+u_a)$ (which was first proved in [135]) to conclude the proof. This is done as follows. First, after the change of variable $u_b \leftarrow e_b \triangleq h_2(u_b)$ where $e_b$ is the entropy of a BSC with cross-over probability $u_b$, the convexity of $f$ gives

$$\begin{aligned}\int_0^{\frac{1}{2}} w_b(u_b) h_2(u_b(1-2u_a)+u_a)\mathrm{d}u_b &= \int_0^1 \tilde{w}_b(e_b) h_2(h_2^{-1}(e_b)(1-2u_a)+u_a)\mathrm{d}e_b \\ &\geq h_2\left( h_2^{-1}(\int_0^1 \tilde{w}_b(e_b) e_b \mathrm{d}e_b)(1-2u_a)+u_a \right) \\ &= h_2\left( h_2^{-1}(\mathtt{H}(\mathsf{b}))(1-2u_a)+u_a \right), \end{aligned}$$

using the channel entropy $\int_0^1 \tilde{w}_b(e_b) e_b \mathrm{d}e_b = \mathtt{H}(\mathsf{b})$. From Eq. (3.1) we get $\mathtt{H}(\mathsf{a} \boxledast \mathsf{b}) \geq \mathtt{H}(\mathsf{a} \boxledast \mathsf{c}_{\mathrm{BSC}(\mathtt{H}(\mathsf{b}))})$. Second, the convexity of $f$ shows that any arc $\{e, f(e)\}$ lies under its chord, therefore if we consider the arc between the points $(e_b = 0, f(e_b) = h_2(u_a))$ and $(e_b = 1, f(e_b) = 1)$ we get $f(e) \leq h_2(u_a)(1-e)+e$ for any $e \in [0,1]$. Applied for $e_b = h_2(u_b)$, we then have the upper bound

$$h_2(u_b(1-2u_a)+u_a) \leq h_2(u_a)(1-e_b)+e_b = 1-(1-h_2(u_a))(1-h_2(u_b))$$

From Eq. (3.1) we finally get $\mathtt{H}(\mathsf{a} \boxledast \mathsf{b}) \leq 1-(1-\mathtt{H}(\mathsf{a}))(1-\mathtt{H}(\mathsf{b}))$. Notice finally that the BEC fulfills the property $\mathtt{H}(\mathsf{a} \circledast \mathsf{d}_{\mathrm{BEC}(\epsilon)}) = \mathtt{H}(\mathsf{a})\mathtt{H}(\mathsf{d}_{\mathrm{BEC}(\epsilon)})$ as we will show again in Lemma 3.4. □

Observe that $\mathsf{a}$ (or $\mathsf{b}$) can itself be the $\circledast/\boxledast$-convolution of any numbers of $L$-densities. In the framework of iterative decoding, this implies the following: If at a variable node we substitute an input density with a density representing a BSC with equal entropy, then the output entropy is decreased. The rule is reversed if instead we use a BEC with equal entropy or if we look at the check node side. Such extremal densities have many applications. In particular they are useful in deriving universal bound on thresholds. For example the idea to derive a universal lower bound on the BP threshold is the following. Consider the picture in Figure 3.3 where density evolution is seen as a staircase function between two fictitious EXIT curves. Instead of replacing these fictitious curves with the Gaussian

approximation as for the EXIT chart method, we can replace them by extremal EXIT curves obtained from the previous theorem, i.e., we consider that the intermediate inputs at the variable (parity-check) nodes densities are BSC (BEC) densities and we obtain a lower bound on the smallest channel entropy $\mathtt{c}$ under which we can guarantee that BP decoding is successful. For example, if we consider LDPC($x^2, x^5$) and BP decoding, then we can transmit reliably over any BMS channel with entropy $\mathtt{h} < 0.3643$. Further examples can be found, e.g., in [133].

## 3.4 EXIT Analysis for the Erasure Channel

In the previous section we saw that rigorous statements using the extremes on information combining could be obtained from EXIT charts. This gave us a way to quantify the maximum deviation of the EXIT chart method from the actual density evolution. Can we derive other rigorous statements? The original reason behind EXIT charts is that if an iterative coding scheme is composed of several component codes (e.g., serial or parallel concatenation), then we characterize each component by its individual EXIT curve. There is a particular case where the EXIT chart methodology is exact: For the BEC it is equivalent to the density evolution equations. Let us first derive further properties of EXIT functions when transmission takes place over the BEC. We will then present some consequences in the framework of the EXIT chart (i.e., density evolution) analysis.

### 3.4.1 Further Properties of EXIT Functions

Let $X$ be chosen with probability $p_X(x)$ from a code $\mathtt{C}$ of length $n$. Consider the memoryless family $\{\mathrm{BEC}_i(\epsilon_i)\}$ such that the $i^{\text{th}}$ bit is transmitted through $\mathrm{BEC}_i(\epsilon_i)$, and let $Y(\epsilon_{[n]})$ denote the received vector (typically, $\epsilon_i = \epsilon_1$ for all $i$). Let us list some useful characterizations of the EXIT function (which extend naturally to the non-binary erasure case). Recall that, in the binary erasure case, the extrinsic MAP estimate $\Phi_i^{\text{MAP}} \triangleq \phi_i^{\text{MAP}}(Y_{\sim i})$ is a LLR that takes on values $\pm\infty$ or 0. The MAP decision is $\hat{x}_i^{\text{MAP}}(y_{\sim i}) \triangleq \text{sign}(\phi_i^{\text{MAP}}(y_{\sim i}))$ if $\phi_i^{\text{MAP}}(y_{\sim i}) \neq 0$, and $\hat{x}_i^{\text{MAP}}(y_{\sim i}) = \text{❋}$ otherwise. Recall $\bar{\epsilon}_i \triangleq 1-\epsilon_i$ and $\epsilon \triangleq (\epsilon_1, \cdots, \epsilon_n)$.

**Lemma 3.4** [Various Characterizations] $h_i(\epsilon) \triangleq H(X_i|Y_{\sim i})$ is equivalent to the following:

(i) $h_i(\epsilon) \triangleq H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}))$

(ii) $h_i(\epsilon) \triangleq \Pr\{\hat{x}_i^{\text{MAP}}(Y_{\sim i}) = \text{❋}\}$

(iii) $h_i(\epsilon) \triangleq \sum_{\mathcal{K} \subseteq [n]\setminus\{i\}} \prod_{j \in [n]\setminus(\{i\}\cup\mathcal{K})} \epsilon_j \prod_{k\in\mathcal{K}} \bar{\epsilon}_k \, H(X_i|X_{\mathcal{K}})$

(iv) $h_i(\epsilon) \triangleq \dfrac{\partial H(X_i|Y)}{\partial \epsilon_i}$

If $\mathtt{C}$ is a binary linear code with parity-check (generator) matrix $H$ ($G$, respectively) from which $X$ is chosen uniformly at random, then $h_i(\epsilon) \triangleq H(X_i|Y_{\sim i})$ is also equivalent to the following:

(v) $h_i(\epsilon) \triangleq \sum_{\mathcal{E} \subseteq [n]\setminus\{i\}} \prod_{j\in\mathcal{E}} \epsilon_j \prod_{k\in[n]\setminus(\{i\}\cup\mathcal{E})} \bar{\epsilon}_k \, \left(1 + \mathrm{rk}(H_{\mathcal{E}}) - \mathrm{rk}(H_{\mathcal{E}\cup\{i\}})\right)$

(vi) $h_i(\epsilon) \triangleq \sum_{\mathcal{K} \subseteq [n]\setminus\{i\}} \prod_{j \in [n]\setminus(\{i\}\cup\mathcal{K})} \epsilon_j \prod_{k\in\mathcal{K}} \bar{\epsilon}_k \, \left(\mathrm{rk}(G_{\mathcal{K}\cup\{i\}}) - \mathrm{rk}(G_{\mathcal{K}})\right)$

*Proof.* Characterization (i) was discussed in Section 3.1. Characterization (ii) comes from considering $p_{X_i|Y_{\sim i}}$ as an erasure channel with erasure probability $\Pr\{\hat{x}_i^{\text{MAP}}(Y_{\sim i}) = ?\} = \Pr\{p_{X_i|Y_{\sim i}}(+1|Y_{\sim i}) = p_{X_i|Y_{\sim i}}(-1|Y_{\sim i})\}$. Characterization (iv) follows from a similar argument. We first write $H(X_i|Y) = \Pr\{\hat{x}_i^{\text{MAP}}(Y) = \text{❋}\} = \Pr\{Y_i = \text{❋}, \hat{x}_i^{\text{MAP}}(Y_{\sim i}) = \text{❋}\} = \epsilon_i h_i(\epsilon)$, then we take the partial derivative with respect to $\epsilon_i$. Characterization (iii) comes from the expansion of the conditional entropy in Fact 3.2. This implies (vi) since for a binary linear code with equal priors $H(X_i|X_{\mathcal{K}}) = \mathrm{rk}(G_{\mathcal{K}\cup\{i\}}) - \mathrm{rk}(G_{\mathcal{K}})$ (which is either 0 or 1 when the $i^{\text{th}}$ bit is reconstructible). Characterization (v) follows from a similar argument. □

Discussion: Each one of the above characterizations has its own merit. Nevertheless, as we will see in the remaining of this thesis, the most fundamental one is characterization (iv). Moreover, observe that, for notational simplicity, we have skipped the superscript MAP, as well as the potential observation $\Omega$ satisfying $\Omega \to X \to Y$. Notice, however, that characterization (iv) for example extends naturally to any extrinsic DEC estimator. We have indeed $h_i(\epsilon) = h_i^{\text{MAP}}(\epsilon) \triangleq \frac{\partial H(X_i|Y_i,\Phi_i^{\text{MAP}})}{\partial \epsilon_i}$. By extension, the DEC EXIT function will be characterized by $h_i^{\text{DEC}}(\epsilon) \triangleq \frac{\partial H(X_i|Y_i,\Phi_i^{\text{DEC}})}{\partial \epsilon_i}$.

**Example 3.3** In this example, let $\epsilon \in [0,1]$ be the scalar such that $\epsilon_i = \epsilon$ for all $i$. Figure 3.4 shows EXIT functions for some standard codes. In all these cases $\forall i$, $h_i(\epsilon) = h_1(\epsilon) = h(\epsilon)$. The $[n+1,1,n+1]$ repetition code has EXIT function $h_i(\epsilon) = \epsilon^n$. Its dual, the $[n+1,n,2]$ single parity-check code, has $h_i(\epsilon) = 1-(1-\epsilon)^n$. If we refer to characterization (ii), we see that the EXIT function for the $[7,4]$ Hamming code has already been depicted in Figure 2.6 of Chapter 2. Let us further illustrate characterizations (v) and (vi) with the self-dual $[8,4]$ extended Hamming code [136] as well as with the $[15,11]$ Hamming code and its dual. The $[8,4]$ self-dual code has $h(\epsilon) = 7\epsilon^3 - 21\epsilon^5 + 21\epsilon^6 - 6\epsilon^7$, the $[15,11]$ Hamming code has $h(\epsilon) = 7\epsilon^2 + 28\epsilon^3 - 49\epsilon^4 - 756\epsilon^5 + 3871\epsilon^6 - 9232\epsilon^7 + 13629\epsilon^8 - 13552\epsilon^9 + 9317\epsilon^{10} - 4396\epsilon^{11} + 1365\epsilon^{12} - 252\epsilon^{13} + 21\epsilon^{14}$, and its dual has $h^{\perp}(\epsilon) = 8\epsilon^7 - 28\epsilon^{11} + 42\epsilon^{13} - 21\epsilon^{14}$.

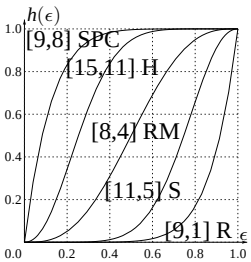



Figure 3.4: EXIT curves $(\epsilon, h(\epsilon))$ over BEC($\epsilon$): Single Parity-Check (SPC) code and Repetition (R) code (or SPC dual), Hamming (H) code and Simplex (S) code (or H dual). Observe that the $[8,4]$ first-order Reed Muller (RM) code or extended Hamming code is self-dual: Its curve is symmetric with respect to the point $(\frac{1}{2},\frac{1}{2})$.

The first characterization of Lemma 3.4 is used for practical computations in the EXIT chart method. The second characterization provides a somehow more intuitive insight into EXIT functions. For example, it is well-known that, over the BEC, a linear code $\mathtt{C}$ can detect and correct up to $d_{\min}-1$ erasures. If $d_{\min} \geq 2$, the punctured code $\ker(H_{[n]\setminus\{i\}})$ can therefore recover at least up to $d_{\min}-2$ erasures. When this is the case, i.e., when the entire extrinsic block is recovered, the intrinsic bit is also uniquely determined. Therefore the extrinsic (bit) erasure probability of the second characterization should have at least minimum degree $d_{\min}-1$. This is stated in (the next) Theorem 3.2.

**Theorem 3.2** [Minimum Distance Theorem] Let $\mathtt{C}$ be a proper binary linear code of length $n$ and minimum distance $d_{\min}$. The EXIT function $h_i(\epsilon)$, $i \in [n]$, is a multivariate polynomial of minimum degree at least $d_{\min}-1$ and the average EXIT function $h(\epsilon)$ has minimum degree *exactly* $d_{\min}-1$.

*Proof.* Consider characterization (v) of Lemma 3.4. Observe first that since the code is proper $h_i(\epsilon)$ is a non-zero multivariate polynomial. Let $\mathcal{E}$ be a subset of cardinality $|\mathcal{E}| < d_{\min}$. As any $d_{\min}-1$ columns of $H$ are linearly independent, it follows that $(1+\text{rk}(H_{\mathcal{E}})-\text{rk}(H_{\mathcal{E}\cup\{i\}}))$ is zero for any such subset $\mathcal{E}$. Therefore $h_i(\epsilon_1,\cdots,\epsilon_n)$ does not contain multivariate monomials of degree less than $d_{\min}-1$. Moreover, if $\mathcal{E}\cup\{i\}$ is chosen to correspond to the support of a minimum distance codeword then $(1+\text{rk}(H_{\mathcal{E}})-\text{rk}(H_{\mathcal{E}\cup\{i\}}))$ is one and this will contribute to a (monic) monomial of degree $d_{\min}-1$. Since these monic minimum degree terms cannot be canceled by any other terms, it follows that $h(\epsilon_1,\cdots,\epsilon_n)$ has minimum degree exactly $d_{\min}-1$. □

**Example 3.4** The EXIT functions in Figure 3.3 show that the minimum distance of the $[n+1,n]$ single parity-check code is 2; its dual has $d_{\min} = n+1$. The $[15,11]$ Hamming code has $d_{\min} = 3$; its dual has $d_{\min}^{\perp} = 8$. The $[8,4]$ extended Hamming has $d_{\min} = 4$. Examples will be shown in Chapter 7 where the *free distance* of a convolutional code is obtained.

The main interest of characterization (v) in Lemma 3.4 is when it is combined with characterization (vi) to give the duality theorem [32, 137]. Recall that, if $G$ if a generator matrix for $\mathtt{C}$, then its dual is the code $\mathtt{C}^\perp \triangleq \ker(G)$. Let us denote $h_i^\perp$ ($h^\perp$) the (average) EXIT function associated with $\mathtt{C}^\perp$.

**Theorem 3.3** [Duality Theorem] Assume $\mathtt{C}$ is a binary linear code with parity-check (generator) matrix $H$ ($G$, respectively) from which $X$ is chosen uniformly at random, then

$$h_i(\epsilon_1, \cdots, \epsilon_n) = 1 - h_i^\perp(1-\epsilon_1, \cdots, 1-\epsilon_n).$$

**Example 3.5** This property can be easily verified on EXIT functions from Example 3.3. For example, for the repetition code of length $n+1$, we have $h(1-\epsilon) = (1-\epsilon)^n = 1 - h^\perp(\epsilon)$ where $h^\perp(\epsilon) = 1-\bar{\epsilon}^n$ is the EXIT function of the single parity-check code. For the $[8,4]$ self-dual code, it can be verified that $h(\epsilon) = h^\perp(\epsilon)$.

Many other proofs of the duality theorem exist: E.g., a proof using the so-called *information functions* is used in [32]. A common trend of all proofs is that they exploit the relationship between a code and its dual. One of the key ingredients to prove the MacWilliams identities is a small exercise in algebra (presented in Appendix 3.B): It shows that, for any subset $\mathcal{S} \subseteq [n]$, $|\mathcal{S}| - \mathrm{rk}(G_\mathcal{S}) = n - k - \mathrm{rk}(H_{[n]\setminus\mathcal{S}})$ where $k$ is the dimension of $\mathtt{C}$. In fact, this statement, together with either characterization (iv) or (v) alone, would also suffice to prove (the duality) Theorem 3.3. It is therefore not surprising that the dual decoding rule according to Hartmann *et al.* [86, 104] (which is derived from the MacWilliams identities and is reviewed in Appendix 2.B) can also prove Theorem 3.3 directly from characterization (ii).
Note that characterization (vi) also permits us to state Lemma 3.5 which shows that, in many cases, the code is such that for all $i$, $h_i = h_1$ (see "isotropy" in Section 3.1).

**Lemma 3.5** Assume $\mathtt{C}$ is a binary linear code of length $n$ with parity-check (generator) matrix $H$ ($G$, respectively) from which $X$ is chosen uniformly at random. Assume that the channel family $\{\mathrm{BEC}(\epsilon_i)\}$ is such that for all $i$, $\epsilon_i = \epsilon \in [0,1]$. If $\forall \mathcal{S} \subseteq [n]$ $\mathrm{rk}(G_\mathcal{S}) = \mathrm{rk}(G_{[|\mathcal{S}|]})$, then $\forall i \in [n]$, $h_i(\epsilon) = h_1(\epsilon)$. Alternatively, if $\forall \mathcal{S} \subseteq [n]$ $\mathrm{rk}(H_\mathcal{S}) = \mathrm{rk}(H_{[|\mathcal{S}|]})$, then $\forall i \in [n]$, $h_i(\epsilon) = h_1(\epsilon)$.

So far we have listed the merits of all but one characterization in Lemma 3.4. All of the induced properties concern *individual* EXIT function $h_i$ but trivially translate to the *average* EXIT function $h = \frac{1}{n}\sum_{i=1}^n h_i$. Nevertheless, if we look at the average EXIT function $h(\mathtt{h})$, an alternative characterization emerges. This is probably the most fundamental property of EXIT functions over the BEC and will be stated as a theorem. From characterization (iv), observe that an alternative characterization is $h_i(\epsilon) = \frac{\partial H(X|Y)}{\partial \epsilon_i}$. To see this, use the chain rule to write $H(X|Y) = H(X_i|Y) + H(X_{\sim i}|Y, X_i) = H(X_i|Y) + H(X_{\sim i}|Y_{\sim i}, X_i)$ where the last equality comes from the memoryless nature of $\{\mathrm{BEC}_i(\epsilon_i)\}_i$. We finally get $\frac{\partial H(X|Y)}{\partial \epsilon_i} = \frac{\partial H(X_i|Y)}{\partial \epsilon_i} + 0$ and we can state this result for the BEC.

**Theorem 3.4** [General Area Theorem – BEC] Let $X$ be a binary random vector of length $n$ and assume that transmission takes place over a family $\{\mathrm{BEC}_i(\epsilon_i)\}_i$. If $\underline{h} \triangleq (h_1(\epsilon_{\sim 1}), \cdots, h_n(\epsilon_{\sim n}))$ denotes the vector composed of the $n$ individual EXIT functions, then $\underline{h}$ is the gradient of the conditional entropy, i.e., $\underline{h} = \nabla H(X|Y) \triangleq (\frac{\partial H(X|Y)}{\partial \epsilon_1}, \cdots, \frac{\partial H(X|Y)}{\partial \epsilon_n})$. Furthermore, if there exists a real-valued parameter $\mathfrak{p}$ such that the vector $\epsilon(\mathfrak{p}) = (\epsilon_1(\mathfrak{p}), \cdots, \epsilon_n(\mathfrak{p}))$ is differentiable in $\mathfrak{p}$, then $\underline{h} \cdot \frac{\mathrm{d}\epsilon(\mathfrak{p})}{\mathrm{d}\mathfrak{p}} = \nabla H(X|Y) \cdot \epsilon'(\mathfrak{p}) = \frac{\mathrm{d}H(X|Y(\mathfrak{p}))}{\mathrm{d}\mathfrak{p}}$ where "$\cdot$" denotes the standard scalar product. In particular, if a parameter $\mathfrak{p}$ can be chosen such that $\epsilon_i(\mathfrak{p}) = \mathfrak{p}$ for all $i$, then $h(\mathfrak{p}) = \frac{1}{n}\sum_{i=1}^n h_i(\epsilon_i) = \frac{\mathrm{d}H(X|Y)}{n\mathrm{d}\mathfrak{p}}$ where $h(\mathfrak{p})$ is the average EXIT function over $\mathrm{BEC}(\mathfrak{p})$.

Discussion: The particular case where $\epsilon_i(\mathfrak{p}) = \mathfrak{p}$ for all $i$ is basically equivalent to the original area theorem [32] (See Appendix 3.C for historical details). It is indeed trivial to deduce that in that case $\int_0^\mathfrak{p} h(\mathfrak{p})\mathrm{d}\mathfrak{p} = \frac{H(X|Y(\mathfrak{p}))}{n}$. For example, if we use a coded transmission and $\mathfrak{p} = 1$, then the area under the EXIT curve equals the rate of the code.

**Example 3.6** From the EXIT functions of Example 3.3, we get $\int_0^1 h(\epsilon)\mathrm{d}\epsilon = \int_0^1 \epsilon^n \mathrm{d}\epsilon = \frac{1}{n+1}$ for the

repetition code of length $n+1$ and $\int_0^1 h(\epsilon)\mathrm{d}\epsilon = \int_0^1 1-(1-\epsilon)^n \mathrm{d}\epsilon = \frac{n}{n+1}$ for the single parity-check code of length $n+1$. In a similar manner, the area under the EXIT curve is $11/15$ for the Hamming code, $4/15$ for the Simplex code, and $1/2$ for the self-dual Reed Muller code.

Theorem 3.4 is more general than the original area theorem because it allows us to consider *any* smooth path of the channel space. If we change the set of all channels $\mathrm{BEC}_i$s from some starting state $A$ characterized by $\{\epsilon_i^A\}_i$ to some final state $B$ characterized by $\{\epsilon_i^B\}_i$, then the total change of entropy $H(X|Y)$ between $A$ and $B$ is independent[5] of the smooth way we follow and equals the sum of "local changes in entropy" at each position. By "local change in entropy", we mean the variation of uncertainty at a bit position due to the variation of all channels, i.e., $h_i(\mathfrak{p})\epsilon_i'(\mathfrak{p})$ at the $i^{\text{th}}$ position. The total change of $H(X|Y)$ along different paths between the initial state $A$ and the final state $B$ is of course the same, but the individual contributions as $h_i(\mathfrak{p})\epsilon_i'(\mathfrak{p})$ might differ. This is illustrated by the next two examples.

**Example 3.7** [Contribution of Individual EXIT Functions] Consider the $[2,1]$ repetition code. Assume first that the channel family is $\{\mathrm{BEC}_i(\epsilon_i=\mathfrak{p})\}_{i\in\{1,2\}}$, i.e., each individual channel is parametrized by the same real-valued parameter $\mathfrak{p}\in[0,1]$. It is easy to see that the change of entropy $H(X|Y(\mathfrak{p}))$ is $H(X)-0=1$ when the common channel entropy $\mathfrak{p}$ varies from 0 to 1. Further we have $h_1(\mathfrak{p})=h_2(\mathfrak{p})=\mathfrak{p}$ so that $\int_0^1 h_i(\mathfrak{p})\mathrm{d}\mathfrak{p}=\frac{1}{2}$ for $i=1,2$. This means that for this parametrization both positions contribute one-half to the total change of entropy rate. Assume now that the channel family is $\{\mathrm{BEC}_1(\epsilon_1=\min(1,\mathfrak{p})),\mathrm{BEC}_2(\epsilon_2=\max(0,1-\mathfrak{p}))\}$ where $\mathfrak{p}$ ranges from 0 to 2, i.e., we change each individual channel entropy from 0 to 1 successively and not simultaneously. The initial and final state are the same as before; therefore the change of entropy rate is again 1. The contribution of the first channel to this total change of entropy is given by $\int_0^2 h_1(\mathfrak{p})\epsilon_1'(\mathfrak{p})\mathrm{d}\mathfrak{p}=\int_0^2 0\mathrm{d}\mathfrak{p}=0$ while the contribution of the second channel is $\int_0^2 h_2(\mathfrak{p})\epsilon_2'(\mathfrak{p})\mathrm{d}\mathfrak{p}=\int_1^2 \mathrm{d}\mathfrak{p}=1$. In other words, the uncertainty of the first position contributes to zero, whereas the second position contributes to one to the total change of conditional entropy.

The freedom of choosing any path between $A$ and $B$ is again exploited in Example 3.8. This is a pleasing example that provides an alternative way to compute a particular area which will be called area under the EBP EXIT curve in the next sections.

**Example 3.8** [Area Theorem and EBP EXIT Curve] Consider the $[5,3]$ code whose parity-check matrix is formed by the two row vectors $(1,1,1,0,0)$ and $(1,0,0,1,1)$. Consider the function $\epsilon: x\mapsto \frac{x}{(1-(1-x)^2)^2}$ defined (by continuity) over $[0,1]$ and let the channel family be $\{\mathrm{BEC}_1(\epsilon(\mathfrak{p})),\{\mathrm{BEC}_i(\mathfrak{p})\}_{i\neq 1}\}$ where $\mathfrak{p}$ ranges from 0 to 1 ($\epsilon(\mathfrak{p})$ ranges from $1/2$ to 1). The local change in entropy at the first position is $I_1 \triangleq \int_{\mathfrak{p}=0}^1 h_1(\mathfrak{p})\mathrm{d}\epsilon(\mathfrak{p})$. This integral can be easily computed in this example. As a game, assume however that we are not allowed to compute it directly. Is there any other way to obtain its value? The answer is affirmative if we take advantage of the general area theorem. The general area theorem states that $H(X)=I_1+\sum_{i=2}^5 h_i(\mathfrak{p})\mathrm{d}\mathfrak{p}$. Now since the code is a tree and because of the particular choice of parametrization (which in fact corresponds to the fixed-points of density evolution), it is easy to check that $h_i(\mathfrak{p})=1-(1-\mathfrak{p})^2$ such that $\int_0^1 h_i(\mathfrak{p})\mathrm{d}\mathfrak{p}=2/3$ for all $i\neq 1$. Then $I_1=H(X)-4\frac{2}{3}=3-8/3=1/3$. We will later see in Chapter 4 and Chapter 6 that $1/3$ is in fact the design rate of the LDPC ensemble whose computation tree of depth 1 is the considered $[5,3]$ code.

Finally, note also that the additional observation $\Omega$ such that $\Omega\to X\to Y$ can also be included in the statements of Theorem 3.4. This general form is given in Chapter 5 and exemplified in Chapter 7. A few notes on the "history" of the area theorem are collected in Appendix 3.C. In the next chapter, we will present what is perhaps the most fundamental use of (G)EXIT functions. But before, let us review some properties obtained from the previous theorems.

[5]This fact is evident in our context where all functions are differentiable. However in the history of thermodynamics, this kind of result has long been deduced from empirical observations that, e.g., have postulated the equivalence between "work" and "heat" (*first principle* of thermodynamics).

### 3.4.2 EXIT Charts

Density evolution [14, 15] or equivalently the analysis of the peeling algorithm [12, 13, 38] reveals that the asymptotic behavior of (G)LDPC ensembles is characterized by $f_\epsilon(\mathtt{x}) \triangleq \epsilon\lambda(\mathtt{y}(\mathtt{x}))$ with $\mathtt{y}(\mathtt{x}) = 1-\rho(1-\mathtt{x})$ for LDPC ensembles. The function $f_\epsilon(\mathtt{x})$ represents the evolution of the fraction of erased messages emitted by the variable nodes when transmission takes place over BEC($\epsilon$). The system is said to be in state $\mathtt{x}$ when $\mathtt{x}$ is the current fraction of erased messages. In the BP implementation, the fraction of erased messages is then given by the sequence $\mathtt{x}_{\ell+1} = f_\epsilon(\mathtt{x}_\ell)$ with $\mathtt{x}_0 = 1$. Various graphical representations of this recursive sequence are possible. Figure 3.5 shows three such standard representations: The decoding process corresponds graphically to a staircase function bounded below by $f_\epsilon(\mathtt{x})$ and bounded above by $\mathtt{x}$ in the classical (middle) picture. It is possible and helpful to represent $f_\epsilon(\mathtt{x})$ as the composition of two non-decreasing (therefore invertible) functions, one which represents the extrinsic entropy emitted at the variable nodes (i.e., the EXIT function for a repetition code), the other which represents the extrinsic entropy emitted by the function nodes (e.g., the EXIT function for a single parity-check code in the case of a LDPC ensemble). Let us therefore use $v_\epsilon(\mathtt{x}) \triangleq \epsilon\lambda(\mathtt{x})$ and $c(\mathtt{x}) \triangleq \mathtt{y}(\mathtt{x})$ to write $f_\epsilon(\mathtt{x}) = v_\epsilon(c(\mathtt{x}))$. The sequence of erased messages emitted by a function node is then $\mathtt{y}_\ell = c(\mathtt{x}_\ell)$, it is $\mathtt{x}_{\ell+1} = v_\epsilon(\mathtt{y}_\ell)$ at a variable node ($\mathtt{y}_0 = 1$ , $\mathtt{x}_0 = 1$). Recall that the condition for convergence reads $f_\epsilon(\mathtt{x}) < \mathtt{x}$ for $\mathtt{x} \in (0,1)$ which can be written as $c(\mathtt{x}) < v_\epsilon^{-1}(\mathtt{x})$ for $\mathtt{x} \in (0,1)$. In other words, the function $c(\mathtt{x})$ has to lie strictly below $v_\epsilon^{-1}(\mathtt{x})$ over $(0,1)$. The BP threshold $\epsilon^{\text{BP}}$ is the supremum of all numbers $\epsilon$ for which this condition is fulfilled. Note that the local condition around $\epsilon = 0$ reads $c'(0) \leq \frac{\mathrm{d}v_\epsilon^{-1}(\mathtt{x})}{\mathrm{d}\mathtt{x}}|_{\mathtt{x}=0} = \frac{1}{\epsilon\lambda'(0)}$. This is of course the stability condition $\rho'(1)\lambda'(0) \leq \frac{1}{\epsilon}$ for LDPC ensembles when $c'(0) = \rho'(1)$.

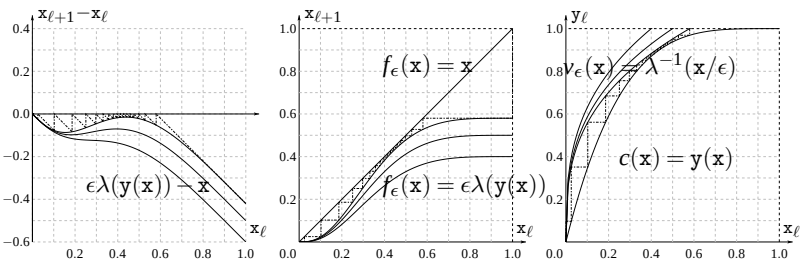



Figure 3.5: Progress of density evolution: Three equivalent pictures represent the asymptotic decoding of LDPC($\mathtt{x}^3, \mathtt{x}^4$) over BEC($\epsilon = 0.58$). Left: Original analysis in [12–15]. Middle: Classical representation. Right: EXIT chart. We have $v_\epsilon^{-1}(\mathtt{x}) = (\mathtt{x}/\epsilon)^3$ for $\epsilon = 0.4,\ 0.5$ and $0.58$, and $c(\mathtt{x}) = 1-(1-\mathtt{x})^4$. The BP threshold is $\epsilon^{\text{BP}} \approx 0.6001$. The evolution of the decoding is represented for $\epsilon = 0.58$, i.e., slightly below threshold.

### 3.4.3 Matching Condition

Consider the EXIT chart associated with a given LDPC ensemble or, more generally, GLDPC ensemble. The case of multi-edge ensembles such as Turbo codes will be considered, for completeness, in Chapter 7. A GLDPC ensemble is characterized by a variable node distribution $\lambda(\mathtt{x})$ and a collection of function nodes such that the EXIT function $c(\mathtt{x})$ represents the extrinsic entropy at the output of the function nodes when transmission takes place over BEC($\mathtt{x}$). All function nodes are assumed to be MAP decoded; $c(\mathtt{x})$ is therefore averaged over all degrees and all possible types of function nodes. For example, if we considered LDPC($\lambda,\rho$), then $c(\mathtt{x}) = \sum_i \lambda_i c_i(\mathtt{x})$ where $c_i(\mathtt{x}) \triangleq 1-(1-\mathtt{x})^{i-1}$ is the EXIT function associated with the $[i, i-1]$ single parity-check code. In the same manner, $v_\epsilon(\mathtt{x})$ is the average EXIT function associated with the variable nodes when transmission takes place over BEC($\epsilon$). We have $v_\epsilon(\mathtt{x}) = \epsilon\lambda(\mathtt{x})$. The area theorem states that the area "under the curve" $c(\mathtt{x})$ equals the rate of the average function node, call it $r_c$. For example, it is $\frac{I'(1)-1}{I'(1)} = 1-\int\rho$ for the case of the ensemble LDPC($\lambda,\rho$). (Note that, for LDPC ensembles, this integral can also be directly computed). In a similar manner, the area "to the left of the curve" $v_\epsilon(\mathtt{x})$ is equal to $\epsilon/\Lambda'(1) = \epsilon\int\lambda$. A necessary condition for successful BP decoding is that the two curves $v_\epsilon(\mathtt{x})$ and $c(\mathtt{x})$ do not cross. In this case the areas do not

overlap and we get the following necessary condition for successful BP decoding:

$$1-\epsilon\int_0^1 \lambda(\mathtt{x})\mathrm{d}\mathtt{x}-\int_0^1 c(\mathtt{x})\mathrm{d}\mathtt{x}>0, \qquad \text{or } 1-\frac{1-r_c}{\int\lambda}<C(\epsilon)\triangleq 1-\epsilon.$$

In other words, the design rate $r(\lambda,c)\triangleq 1-\frac{1-r_c}{\int\lambda}$ of any GLDPC ensemble that, for increasing block lengths, allows successful BP decoding over BEC($\epsilon$) cannot surpass the channel capacity. This necessary condition is called *matching condition* and arises similarly in the context of multi-edge ensembles, see Chapter 7. Although the matching condition itself is trivial, its derivation is constructive because it shows how the Shannon limit enters in the calculation of the asymptotic performance of iterative coding system. In particular, it shows that in order to achieve capacity, the two EXIT curves have to be perfectly matched. We will exemplify this point in the next subsection.
Notice that an argument very similar to the one above is introduced in [34, 138] (albeit not using the language and geometric interpretation of EXIT functions and applying a slightly different range of integration). It was the first bound on the performance of iterative systems in which the Shannon capacity appeared explicitly using only quantities of density evolution. A substantially more general version of this bound can be found in [32, 137, 139]. See also [47, 61]. The extension to parallel turbo schemes is addressed in [36, 47] and discussed in Chapter 7.
A generalization of the matching condition to BMSCs will be presented in Chapter 6.

### 3.4.4 Capacity-Achieving Sequences

The quantity $r-\epsilon=(1-\epsilon)-(1-r)$, which is the limiting *additive gap to capacity* shown by the matching condition, can be further quantified. Observe the EXIT chart in Figure 3.6, which represents the case of transmission just below the BP threshold. At channel parameter $\epsilon=\epsilon^{\mathrm{BP}}$, the two EXIT functions are tangent in $(\mathtt{x}^{\mathrm{BP}},\mathtt{y}^{\mathrm{BP}})$ and the EXIT chart gives a graphical representation of the limiting gap to capacity: The additive gap $C(\epsilon^{\mathrm{BP}})-r$ where $C(\epsilon^{\mathrm{BP}})\triangleq 1-\epsilon^{\mathrm{BP}}$ is indeed represented by the entire white area $\mathcal{D}$ such that

$$C(\epsilon^{\mathrm{BP}})-r=\epsilon^{\mathrm{SH}}-\epsilon^{\mathrm{BP}}=\frac{\mathcal{D}}{\int\lambda}, \tag{3.2}$$

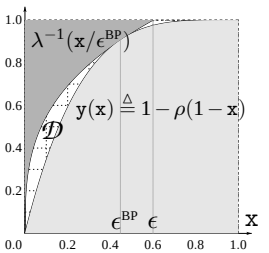



Figure 3.6: Additive gap to capacity for the ensemble LDPC($\mathtt{x}^3,\mathtt{x}^4$).

where $\frac{1}{\int\lambda}=\Lambda'(1)$ is the average left degree. In other words, the area $\mathcal{D}$ is the area between the left EXIT curve $\mathtt{x}\mapsto\lambda^{-1}(\mathtt{x}/\epsilon^{\mathrm{BP}})$ (at the BP threshold) and the right EXIT curve $\mathtt{x}\mapsto c(\mathtt{x})$ that is bounded away by the unit square, see, e.g., [32]. This expression has a straightforward consequence: the fact that "good" iterative coding schemes do not require the use of "good" component codes (i.e., codes with an associated EXIT function which becomes a step function, see Appendix 7.A). In order to make the gap to capacity as small as possible, one natural method would be to consider a curve $c(x)$ and to look if its inverse function has a Taylor expansion with positive coefficients. See [12, 13, 122]. After some work, we hope to make the gap to capacity (as well as the matching of the curves) very small. It is shown in [35, 138] that no fixed dd pair $(\lambda,\rho)$ has zero (multiplicative) gap to capacity (where the multiplicative gap to capacity is $(C(\epsilon^{\mathrm{BP}})-r)/C(\epsilon^{\mathrm{BP}})$). We then have to work with *sequences* of ensembles.
The next lemma presents such a construction: this is a variation from the standard right-concentrated capacity-achieving sequences presented in [34, 140].

**Lemma 3.6** [Right-Regular Capacity Achieving Sequence] Consider a fixed degree $\mathtt{r}>2$ and let $\rho_{\mathtt{r}}(\mathtt{x})\triangleq\mathtt{x}^{\mathtt{r}-1}$ represent a right degree distribution. Assume that transmission takes place over BEC($\epsilon$). Define

$$\tilde{\lambda}(\mathtt{x})\triangleq 1-(1-\mathtt{x})^{\frac{1}{\mathtt{r}-1}}=\sum_{i=2}^{\infty}\overbrace{\binom{\frac{1}{\mathtt{r}-1}}{i-1}(-1)^i}^{\tilde{\lambda}_i>0}x^{i-1}, \quad \lambda_{d_{\mathtt{r}}}(\mathtt{x})\triangleq\frac{1}{\epsilon}\sum_{i=2}^{d_{\mathtt{r}}}\tilde{\lambda}_i\mathtt{x}^{i-1}+\tilde{\lambda}_{L_{\mathtt{r}}}\mathtt{x}^{L_{\mathtt{r}}-1},$$

where $d_{\mathtt{r}} > 2$. Then, there exists $d_{\mathtt{r}}$ satisfying $\sum_{i=2}^{d_{\mathtt{r}}} \tilde{\lambda}_i \leq \epsilon$ and $L_{\mathtt{r}} >> d_{\mathtt{r}}$ sufficiently large such that $(\lambda_{\mathtt{r}}(\mathtt{x}), \rho_{\mathtt{r}}(\mathtt{x}))$ is a valid dd pair. This pair allows for asymptotically erasure-free transmission at design rate $r_{\mathtt{r}} = 1 - \frac{1}{\mathtt{r}\sum_{i=2}^{d_{\mathtt{r}}} \tilde{\lambda}_i / i} + o_{L_{\mathtt{r}}}(1)$ (where $o_{L_{\mathtt{r}}}(1)$ is arbitrarily small). Furthermore, $\lim_{\mathtt{r}\to\infty} r_{\mathtt{r}} = 1 - \frac{\mathtt{r}}{\mathtt{r}-1}\epsilon$, which shows that erasure-free transmission is (asymptotically) possible arbitrarily close to capacity.

Discussion: Let us first indicate that the proof of this lemma follows from a few geometric considerations using the convexity of $\lambda^{-1}$, the concavity of $c(x)$ and the fact that the upper part of the residual area $\mathcal{D}$ tends to zero geometrically as $\mathrm{O}((1-\epsilon)^{\mathtt{r}-1})$. Second, observe that, in order to adjust the weight of the left degree distribution, we chose to put all the weight to a very high (think of it as "infinite") degree. This is a minor modification of the original right-regular construction [34, 35, 140] that distributes the weight over *all* coefficients [34,35,140]. However in both cases the first coefficients of the Taylor expansion are used to construct a sequence that performs close to capacity. In our case, these coefficients are perfectly matched. This might not be the optimum choice in terms, e.g., of complexity, but this is somehow closer to what a linear optimization program would find out, see, e.g., [128].

## 3.5 Conclusion and Discussion

This chapter has presented EXIT functions and their main properties. Most of those properties are only valid for the particular case of the BEC. For this channel, the EXIT chart methodology permit us to derive capacity-achieving sequences of dd pairs. Such sequences are obtained by matching the EXIT functions of the individual component codes. From a theoretical point of point, this is done by using the Taylor expansion of one (fixed) individual EXIT curve as shown in the previous section. From a practical point of view, when we aim at optimizing a given iterative coding system in order to approach channel capacity, we will read off the "bottlenecks" between the two individual EXIT curves.

Although this might not be an optimal trade off between performance versus complexity, the sequence presented in Lemma 3.6 shows already that one can read off the "bottlenecks" in the decoding process.

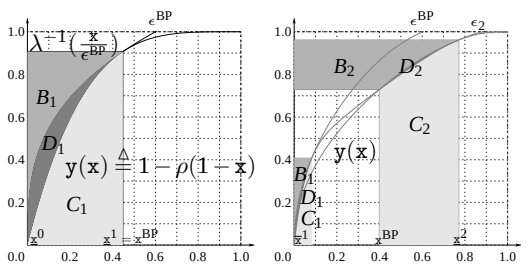



Figure 3.7: Graphical interpretation of Theorem 4.10 (dynamic level). Left: Ensemble with dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^3)$ (one-jump) and transmission at $\epsilon = \epsilon^{\mathrm{BP}}$. Right: Ensemble with $\lambda(\mathtt{x}) = 0.78\mathtt{x}^2 + 0.1\mathtt{x}^3 + 0.12\mathtt{x}^{14}$ and $\mathtt{y}(\mathtt{x})$ obtained from 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 (edge perspective) at $\epsilon = \epsilon^{\mathrm{BP}}$ and at $\epsilon = \epsilon_2$.

The (limiting) individual EXIT curves associated with this sequence match perfectly. This is not the case for iterative coding systems encountered in practice. However it is possible to improve the performance of the system by identifying the critical points. Once these critical regions have been identified, the individual component codes can be changed appropriately to improve the performance of the system (see also Example 7.2 in Chapter 7). The degree of freedom for this improvement is linked to the area gap (called $\mathcal{D}$ in Figure 3.6) between the individual EXIT functions. The derivation of provable capacity-approaching or capacity-achieving ensembles is a first application of EXIT charts over the BEC. Other applications are possible; an important one is when we want to give (at least in certain cases) lower bounds on the number of iterations for successful decoding. But the application of EXIT functions, perhaps the most surprising, is obtained when we look at a *single* EXIT curve (and not a "chart") on the erasure channel. This topic is addressed in the next chapter. We will apply the area theorem to the EXIT function that describes the average performance of the overall LDPC ensemble. Surprisingly this will permit us to refine the statement of Eq. (3.2). In other words, we will see that the area $\mathcal{D}$ in Figure 3.6 can be itself divided into two parts where the subarea "below $\mathtt{x}^{\mathrm{BP}}$" (denoted by $D_1$ on the left picture in Figure 3.7) 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-matching problem, but now "below" $\mathtt{x}^{\mathrm{BP}}$. See Figure 3.7... and the next chapter!

In the next chapter, we will use EXIT functions to present the strong relationship between iterative (BP) and optimal (MAP) decoding when transmission takes place over the BEC. The second part of the thesis starting at Chapter 5 (with the introduction of GEXIT functions) will extend the properties and applications of EXIT functions to general BMSCs.

# Appendix

## 3.A Technical Clarifications on the Additional Observation $\Omega$

So far we have used several times the hypothesis $Y_i \to X_i \to Y_{\sim i}$. As already discussed in Section 2.4, it is implied by a more general assumption, the memoryless nature of the channel $p_{Y_{[n]}|X_{[n]}}$. In fact, if a channel is memoryless, then $\forall \mathcal{S} \subseteq [n]$, $Y_{\mathcal{S}} \to X_{\mathcal{S}} \to Y_{[n]\setminus\mathcal{S}}$.
In order to include cases such as, e.g., parallel concatenation in our framework, we consider a further observation $\Omega$ such that $\Omega \to X \to Y$ as in Definition 3.1. The next fact is needed to enlarge the domain of application of Example 2.9 to such cases.

**Fact 3.3** Assume $Y_i \to X_i \to Y_{\sim i}$ and $\Omega \to X \to Y$. Then $Y_i \to X_i \to (Y_{\sim i}, \Omega)$.

*Proof.* Assume for simplicity that the channels are discrete such that we have

$$\begin{aligned} p(y_i|x_i,y_{\sim i},\omega) &= \frac{p(y_i,\omega|x_i,y_{\sim i})}{p(\omega,x_i,y_{\sim i})} = \frac{p(y_i|x_i,y_{\sim i})p(\omega|x_i,y_{\sim i},y_i)}{p(\omega|x_i,y_{\sim i})} \\ &\overset{(a)}{=} p(y_i|x_i)\frac{p(\omega|x_i,y)}{p(\omega|x_i,y_{\sim i})} = p(y_i|x_i)\frac{\sum_{x_{\sim i}} p(\omega,x_{\sim i}|x_i,y)}{p(\omega|x_i,y_{\sim i})} \\ &= p(y_i|x_i)\frac{\sum_{x_{\sim i}} p(\omega|x,y)p(x_{\sim i}|x_i,y)}{p(\omega|x_i,y_{\sim i})} \overset{(b)}{=} p(y_i|x_i)\frac{\sum_{x_{\sim i}} p(\omega|x)p(x_{\sim i}|x_i,y)}{p(\omega|x_i,y_{\sim i})} \end{aligned}$$

where $(a)$ uses $Y_i \to X_i \to Y_{\sim i}$ and $(b)$ uses $\Omega \to X \to Y$. The denominator can be written as $p(\omega|x_i,y_{\sim i}) = \sum_{x_{\sim i}} p(\omega,x_{\sim i}|x_i,y_{\sim i}) = \sum_{x_{\sim i}} p(\omega|x,y_{\sim i})p(x_{\sim i}|x_i,y_{\sim i}) = \sum_{x_{\sim i}} p(\omega|x)p(x_{\sim i}|x_i,y_{\sim i})$ where the last equality comes from $p(\omega|x,y_{\sim i}) = \frac{p(\omega,y_{\sim i}|x)}{p(y_{\sim i}|x)} = \frac{\sum_{y_i} p(\omega,y|x)}{p(y_{\sim i}|x)} = \frac{\sum_{y_i} p(\omega|x,y)p(y|x)}{p(y_{\sim i}|x)} = p(\omega|x)$ with $\Omega \to X \to Y$. The denominator can further be written $\sum_{x_{\sim i}} p(\omega|x)p(x_{\sim i}|x_i,y_{\sim i}) = \sum_{x_{\sim i}} p(\omega|x)p(x_{\sim i}|x_i,y)$ observing $Y_i \to X_i \to Y_{\sim i}$. We finally obtain $p(y_i|x_i,y_{\sim i},\omega) = p(y_i|x_i)$. □

Discussion: Assuming $Y_i \to X_i \to Y_{\sim i}$ and $\Omega \to X \to Y$, we also have $p(y_i|x_i,\omega) = p(y_i|x_i)$. This means that the channel $p_{\Omega=\omega}(y_{[n]}|x_{[n]})$ itself is memoryless. The observation $\Omega$ plays the role of an additional (and independent) channel observation, i.e., we could formally define a received extrinsic vector $\tilde{Y}_{\sim i} = (Y_{\sim i}, \Omega)$.
The main consequence of Fact 3.3 is that it shows that the random variable $\phi_i^{\text{MAP}}(Y_{\sim i}, \Omega)$ constitutes a sufficient statistic for estimating $X_i$. This follows from similar considerations to those leading to Example 2.9. Therefore (with a slight abuse of notation) $H(X_i|Y_{\sim i},\Omega) = H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}),\Omega) = H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i},\Omega))$. Moreover observe that $Y_i$ and $\Phi_i^{\text{MAP}}$ are conditionally independent random variables.

## 3.B A Touch of Algebra

The following simple exercise in linear algebra is used several times in this thesis.

**Fact 3.4** Consider a $[n,k]$ linear code. Assume it possesses a parity-check matrix $H$ and a generator matrix $G$. Then, for any subset $\mathcal{S} \subseteq [n]$, we have $|\mathcal{S}| - \mathrm{rk}(G_{\mathcal{S}}) = (n-k) - \mathrm{rk}(H_{[n]\setminus\mathcal{S}})$.

*Proof.* Since $\mathtt{C} = \ker(H) = \mathrm{Vect}(\{(G_{i_1} G_{i_2} \cdots G_{i_n})\}_{1\le i\le k})$, linear combinations of rows do not change the rank. Consider a generator matrix $G$ and choose a subset $\mathcal{S} \subseteq [n]$. The sub-matrix $G_{\mathcal{S}}$ has $|\mathcal{S}|$ columns of rank $\mathrm{rk}(G_{\mathcal{S}})$. Therefore one could find a new generator matrix $G'$ of the subspace $\mathtt{C}$ such that $G'_{\mathcal{S}} = [Q^T\ 0]^T$ has $|\mathcal{S}|$ columns of rank $\mathrm{rk}(G_{\mathcal{S}})$ with a $\mathrm{rk}(G_{\mathcal{S}}) \times |\mathcal{S}|$ sub-matrix $Q$ of same rank (0 can be an empty submatrix). Consider the dual matrix $Q^\perp$ of minimum rank such that $Q(Q^\perp)^T = 0$. The rank formula says $\mathrm{rk}(Q) + \dim[\mathrm{Ker}(Q)] = \mathrm{rk}(Q) + \mathrm{rk}(Q^\perp) = |\mathcal{S}|$. Therefore $Q^\perp$ (with $|\mathcal{S}|$ columns) has rank $|\mathcal{S}| - \mathrm{rk}(G_{\mathcal{S}})$. Completing the basis, we can find two submatrices $U$ and $V$ to form a $(n-k)\times n$ matrix $H'$ such that $G'H'^T = 0$ with $H'_{[n]\setminus\mathcal{S}} = [0\ V^T]^T$ and $H'_{\mathcal{S}} = [Q^\perp\ U^T]^T$. The matrix $[U\ V]$ is a $(n - k + \mathrm{rk}G_{\mathcal{S}} - |\mathcal{S}|) \times n$ matrix. We then conclude that $\mathrm{rk}(H_{[n]\setminus\mathcal{S}}) = \mathrm{rk}(H'_{[n]\setminus\mathcal{S}}) = n - k + \mathrm{rk}(G_{\mathcal{S}}) - |\mathcal{S}|$. □

## 3.C A Brief History of Area Theorems

The first work with the flavor of the area theorem appears in [34, 138, 140]. The main differences between this first work and the area theorem are that, first, the range of integration in [140] is slightly different from the one in [32], and, second, the results are mainly rooted by dynamical considerations (i.e., by the design of capacity-achieving schemes). The first explicit statement that connects the *area* under the EXIT function to an invariant quantity (the rate of the code) comes later in [137,139,141,142] and is published in [32]. The use of the area theorem for parallel concatenation is treated in [36, 47].
The general area theorem, which we stated in this chapter, generalizes the original version in [32]. The fundamental difference between the two is in the proof technique.
The first partial justification of the area theorem is given in [141]. It uses the chain rule and Riemann sums. (This is an idea similar to the one used in Appendix 7.B for the bi-infinite trellis). A more formal result is presented for linear codes in [142]: it consists of taking the integral of characterization (v) or (vi) of Lemma 3.4 to get a difference of two sums whose terms cancel pair-wise. An alternative (slightly more general, but similar in essence) proof is provided in [32, 137, 139]. Let us present this version in the following.

**Theorem 3.5** [Ashikhmin et al. Area Theorem] Let $X$ be a binary vector of length $n$ chosen uniformly at random from a code $\mathtt{C}$. Let $Y(\epsilon)$ be the result of passing $X$ through $\mathrm{BEC}(\epsilon)$. Let $\Omega$ be a further observation of $X$ so that $\Omega \to X \to Y$. Then

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

*Proof.* We get

$$\begin{aligned}
\sum_{i\in[n]} \int_0^1 H(X_i \,|\, Y_{\sim i}(\epsilon), \Omega)\mathrm{d}\epsilon &\overset{(a)}{=} \sum_{i\in[n]} \sum_{\mathcal{S}\subseteq[n]\setminus\{i\}} H(X_i \,|\, X_{\mathcal{S}}, \Omega) \int_0^1 (1-\epsilon)^{|\mathcal{S}|}\epsilon^{n-1-|\mathcal{S}|}\mathrm{d}\epsilon \\
&\overset{(b)}{=} \sum_{i\in[n]} \sum_{\mathcal{S}\subseteq[n]\setminus\{i\}} H(X_i \,|\, X_{\mathcal{S}}, \Omega) \frac{(n-1-|\mathcal{S}|)!|\mathcal{S}|!}{n!} \\
&\overset{(c)}{=} \sum_{s=0}^{n-1} \sum_{\mathcal{S}\subseteq[n]:|\mathcal{S}|=s} \sum_{i\in[n]\setminus\mathcal{S}} \sum_{\pi\in\Pi_{\mathcal{S}}} \frac{(n-1-s)!}{n!} H(X_i \,|\, X_{\pi(\mathcal{S})}, \Omega) \\
&= \sum_{s=0}^{n-1} \sum_{\mathcal{S}\subseteq[n]:|\mathcal{S}|=s} \sum_{\pi\in\Pi_{\mathcal{S}}} \sum_{i\in[n]\setminus\mathcal{S}} \frac{(n-1-s)!}{n!} H(X_i \,|\, X_{\pi(\mathcal{S})}, \Omega) \\
&\overset{(d)}{=} \sum_{s=0}^{n-1} \sum_{\iota\in\Upsilon_{[s]\to[n]}} \sum_{i\in[n]\setminus\iota([s])} \frac{(n-1-s)!}{n!} H(X_i \,|\, X_{\iota([s])}, \Omega)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{s=0}^{n-1} \sum_{\iota \in \Upsilon_{[s+1]\to[n]}} \frac{(n-1-s)!}{n!} H(X_{\iota(s+1)} \,|\, X_{\iota([s])}, \Omega) \\
&\overset{(e)}{=} \sum_{s=0}^{n-1} \sum_{\iota \in \Upsilon_{[s+1]\to[n]}} \sum_{\pi \in \Pi_{[n]\setminus\iota([s])}} \frac{1}{n!} H(X_{\iota(s+1)} \,|\, X_{\iota([s])}, \Omega) \\
&\overset{(f)}{=} \sum_{s=0}^{n-1} \sum_{\pi \in \Pi_{[n]}} \frac{1}{n!} H(X_{\pi(s+1)} \,|\, X_{\pi([s])}, \Omega) \overset{(g)}{=} H(X \,|\, \Omega),
\end{aligned}
$$

where $(a)$ uses Lemma 3.4 and characterization (iii), $(b)$ is the integration of the Beta function $B(u,v) = \int_0^1 \epsilon^{u-1}(1-\epsilon)^{v-1}\mathrm{d}\delta = \frac{(u-1)!(v-1)!}{(u+v-1)!}$, $(c)$ is obtained by switching the sums and denoting $\Pi_{\mathcal{S}}$ the group of the permutations over $\mathcal{S}$ (there exists $|\mathcal{S}|!$ such permutations), $(d)$ uses the notation $\Upsilon_{\mathcal{S}\to[n]}$ for the set of all injections of a subset $\mathcal{S}$ into the set $[n]$, $(e)$ uses again the notation $\Pi_{[n]\setminus\mathcal{S}}$ for the group of the permutations over the set $[n]\setminus\mathcal{S}$, $(f)$ constructs permutations over $[n]$ by rearranging the $s+1$ first elements in an initial stage and finally combining the remaining ones, and $(g)$ uses the chain rule for entropy $H(X|\Omega) = \sum_{s=1}^{n}(X_i|X_1, X_2, \cdots, X_{s-1}, \Omega)$ and the $n!$ ways of writing down this rule such that $H(X|\Omega) = \frac{1}{n!}\sum_{s=1}^{n}\sum_{\pi\in\Pi_{[n]}} H(X_{\pi(s)}|X_{\pi([s-1])}, \Omega)$. □

Observe that the observations $Y$ and $\Omega$ represent what were called, in the original theorem [137], the "extrinsic" information and the "channel," respectively.
Let us now show how our formulation relates to the original statement, i.e., let us make the bridge between the original area theorem and (the general area) Theorem 3.4. In Theorem 3.5 the integration ranges from zero (perfect channel) to one (no information conveyed). The following is a trivial extension.

**Theorem 3.6** [Area Theorem] Let $X$ be a binary vector of length $n$ chosen uniformly at random from a code $\mathtt{C}$. Let $Y(\epsilon)$ be the result of passing $X$ through BEC$(\epsilon)$. Let $\Omega$ be a further observation of $X$ so that $\Omega \to X \to Y$. Then

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

*Proof.* Let $Y^{(1)}$ be the result of passing $X$ through BEC$(\epsilon)$ and $Y^{(2)}$ be the result of passing $X$ through BEC$(\epsilon^*)$. Let $\Omega$ be the additional observation of $X$. Applying Theorem 3.5, with $Y = Y^{(1)}$ and with additional observation $(Y^{(2)}, \Omega)$, we have $p_{\Omega,Y^{(2)}\,|\,X,Y^{(1)}}(\omega, y^{(2)} \,|\, x, y^{(1)}) = p_{\Omega,Y^{(2)}\,|\,X}(\omega, y^{(2)} \,|\, x)$, 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 $\epsilon^*$. 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 3.6 follows. □

It suffices now to allow each $X_i$ to be passed through a different channel BEC$(\epsilon_i)$ to obtain (the general area) Theorem 3.4 by differentiating $H(X|Y, \Omega)$.

**Overview:** The relationship between MAP and BP decoding is described in the setting of transmission over the BEC and infinite blocklengths. An (almost) complete characterization is given.

# 4 The Bridge between MAP and BP Decoding

As it was shown in the previous chapter, EXIT functions are handy tools for visualizing the decoding process. Various consequences, for example, on complexity issues or code optimization, have been pointed out. Perhaps more surprising and more fundamental is the fact that, for the erasure channel, EXIT functions connect the performance of a code under MAP decoding to that under BP decoding. The reason is that they contain in essence a conservation law (the general area theorem) on the entropy. A construction reminiscent of the Maxwell construction in thermodynamics (see Chapter 1) constitutes the bridge between MAP and BP decoding.
This chapter deals with transmission over BEC($\epsilon$), where $\epsilon$ denotes the erasure probability.

## 4.1 Asymptotic EXIT Functions

Let $\mathtt{C}$ be a binary linear code of length $n$. Assume that we choose a codeword $X$ uniformly at random from $\mathtt{C}$. Let $Y(\epsilon)$ be the result of transmitting $X$ over BEC($\epsilon$). Let $\mathtt{G}$ be a (fixed) graphical representation of the code and consider the BP schedule described in Section 2.5. Assume that we use the extrinsic BP estimate at the $\ell^{\text{th}}$ iteration, i.e., consider $\phi_i^{\text{BP},\ell}(Y_{\sim i})$ (which is independent of the $i^{\text{th}}$ received symbol). Define the $i^{\text{th}}$ BP EXIT function at iteration $\ell$ to be $h_i^{\text{BP},\ell} \triangleq H(X_i|\phi_i^{\text{BP},\ell}(Y_{\sim i}))$ as stated in Definition 3.3. Using (the data processing) Theorem 2.1 and Example 2.9 we see that the BP EXIT function belongs to the general class of upper bounds on the MAP EXIT function. Formally,

$$h_i^{\text{MAP}}(Y_{\sim i}) \triangleq H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i})) = H(X_i|Y_{\sim i}) \leq H(X_i|\phi_i^{\text{BP},\ell}(Y_{\sim i})) = h_i^{\text{BP},\ell}(Y_{\sim i}).$$

At first glance it seems that not much more than this inequality can be stated about the relationship between MAP and BP decoding. However, in the asymptotic limit and for sparse graphs, a fundamental connection between these two quantities appears. Therefore we now turn our attention to the (average) performance of such large graphs.

**Definition 4.1** [(MAP) EXIT Function over BEC($\epsilon$)] The MAP EXIT function associated with the dd pair $\Xi$ is defined as

$$h^{\text{MAP}}(\epsilon) \triangleq \limsup_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}\big[\frac{1}{n}\sum_{i=1}^{n} H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}(\epsilon)))\big],$$

where the expectation is over instances of graph $\mathtt{G}$ taken uniformly at random from LDPC($n,\Xi$), $X$ denotes a codeword chosen uniformly at random from $\mathtt{G}$, $Y(\epsilon)$ is the result of transmitting $X$ over BEC($\epsilon$), and $\phi_i^{\text{MAP}}(Y_{\sim i})$ is the $i^{\text{th}}$ extrinsic MAP estimate.

Discussion: Taking the average over all positions $i$ is not essential in this definition. In fact we can also write $h^{\text{MAP}}(\epsilon) = \limsup_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}\big[H_{\mathtt{G}}(X_1|Y_{\sim 1}(\epsilon))\big]$ since the quantity is averaged over all graphs in LDPC$(n,\Xi)$ (and therefore all possible permutations of columns). A more fundamental observation is that we consider the average EXIT function (over the ensemble of graphs). The practical interest of this technique is justified in Appendix 4.A. This is done in the usual manner by showing that the particular instances $\frac{1}{n}\sum_{i=1}^{n} H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))$ concentrate around their expected value. Finally note that we use the lim sup instead of the ordinary limit because it is not obvious a priori that the ordinary limit indeed exists. Towards the end of this chapter we will show that in many cases the ordinary limit is a well-defined object.

**Definition 4.2** [BP EXIT Function over BEC($\epsilon$)] The BP EXIT function associated with the dd pair $\Xi$ is defined as

$$h^{\text{BP}}(\epsilon) \triangleq \lim_{\ell\to\infty}\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}\big[\frac{1}{n}\sum_{i=1}^{n} H(X_i|\phi_i^{\text{BP},\ell}(Y_{\sim i}(\epsilon))\big],$$

where the expectation is over instances of graph $\mathtt{G}$ taken uniformly at random from LDPC$(n,\Xi)$, $X$ denotes a codeword chosen uniformly at random from $\mathtt{G}$, $Y(\epsilon)$ is the result of transmitting $X$ over BEC($\epsilon$), and $\phi_i^{\text{BP},\ell}(Y_{\sim i})$ is the $i^{\text{th}}$ extrinsic BP estimate at iteration $\ell$.

Contrary to $h^{\text{MAP}}$, this object is well-defined and can be computed easily in a parametric way.

**Theorem 4.1** The BP EXIT function associated with the dd pair $\Xi$ is given by $h^{\text{BP}}(\epsilon) = \max\{0, \mathcal{H}^{\text{EBP}}(\epsilon)\}$ where $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$ for LDPC$(n,\lambda,\rho)$ and[1] $\mathcal{H}^{\text{EBP}}(\epsilon) \triangleq \{\Lambda(\mathtt{y}(\mathtt{x})) : \mathtt{x}\in[0,1], \epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))} = \epsilon\}$.

*Proof.* Standard arguments from density evolution, see [12–15, 65], show that if we first let $n\to\infty$ and second $\ell\to\infty$, then the erasure probability emitted by the variable nodes converges to the value that we get if we run density evolution on an infinite tree. This limit, call it $\mathtt{x}$, is the largest fixed-point of the density evolution equations. More precisely, recall that the fixed-point condition reads $\mathtt{x} = \epsilon\lambda(\mathtt{y}(\mathtt{x}))$ where $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$ for the dd pair $(\lambda,\rho)$ over BEC($\epsilon$). The formal characterization of the asymptotic behavior is easy to understand: The graph $\mathtt{G}$ is locally a tree with high probability. Therefore, it can be shown that, for a fixed (large) number of iterations, when $n\to\infty$, the erasure probability after BP decoding on the actual graph becomes equal (with probability one) to the erasure probability after BP decoding on the associated infinite tree or *computation* tree. (This argument extends naturally to GLDPC ensembles since the computation tree remains the same if we replace check nodes by more complex constraints). Solving the fixed-point equation for $\epsilon$, we get $\epsilon(\mathtt{x}) = \mathtt{x}/\lambda(\mathtt{y}(\mathtt{x}))$, $\mathtt{x}\in(0,1]$. In other words, for each non-zero fixed-point $\mathtt{x}$ of density evolution, there is a unique channel parameter $\epsilon$. At this fixed-point the erasure probability emitted by the function nodes is $\mathtt{y}(\mathtt{x})$, therefore the extrinsic erasure probability, i.e., the BP EXIT function equals $\Lambda(\mathtt{y}(\mathtt{x}))$. □

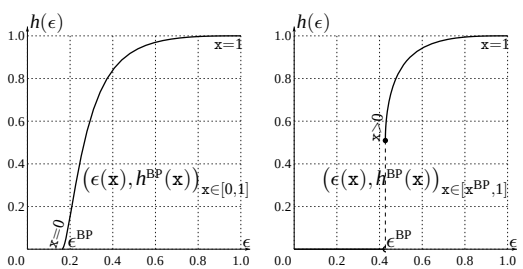



Figure 4.1: BP EXIT functions. Left: LDPC$(\lambda(\mathtt{x}) = \mathtt{x}, \rho(\mathtt{x}) = \frac{\mathtt{x}^3+4\mathtt{x}^7}{5})$, with $\epsilon^{\text{BP}} = \epsilon^{\text{SC}} = \frac{5}{31} \approx 0.1613$. Right: LDPC$(\lambda(\mathtt{x}) = \frac{\mathtt{x}+4\mathtt{x}^3}{5}, \rho(\mathtt{x}) = \frac{\mathtt{x}^3+4\mathtt{x}^7}{5})$, with $\epsilon^{\text{BP}} \approx 0.4273$ (at $\mathtt{x}^{\text{BP}} \approx 0.2524$) and $\epsilon^{\text{SC}} = \frac{25}{31} \approx 0.8065$.

Discussion: If $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ increases over the whole interval $[0,1]$, then the BP EXIT curve is given in parametric form by $\big(\epsilon(\mathtt{x}), \Lambda(1-\rho(1-\mathtt{x}))\big)$. An example is depicted in Figure 4.1 (left). Note that the value $\epsilon(0) = \epsilon^{\text{SC}} \triangleq \frac{1}{\epsilon\lambda'(0)y'(0)}$ indicates the stability condition threshold. For some ensembles, e.g., regular cycle-code ensembles with dd pair $(\lambda(x)=x, \rho(x))$, $\epsilon(\mathtt{x})$ is indeed increasing[2] over the whole range $[0,1]$, but this is not true in general. For the general case, the domain of definition of the parameter $\mathtt{x}$ reduces to a subset $\mathcal{D}\subseteq[0,1]$ that is smaller than the full interval $[0,1]$. The domain $\mathcal{D}$ describes all possible values for the fixed-point $\mathtt{x}$ of density evolution when BP decoding is not successful. Standard (simple) examples are LDPC with $\mathcal{D} = [\mathtt{x}^{\text{BP}}, 1]$

[1] The functions which define $\mathcal{H}^{\text{EBP}}(\epsilon)$ are composed from polynomials; therefore this set contains a finite number of elements.

[2] This follows from the fact that $\mathtt{y}(\mathtt{x}) = 1-\rho(1-\mathtt{x})$ is concave with $\mathtt{y}(0) = 0$.

like in Figure 4.1 (right) with $0 < \mathtt{x}^{\text{BP}} \triangleq \text{argmin}_{\mathtt{x}\in[0,1]}\{\epsilon(\mathtt{x})\} \approx 0.2524$. Such LDPC examples have an associated EXIT function with *one* discontinuity (which appears at the BP threshold). This "jump" is, in the vocabulary of thermodynamics, a phase transition. Regular LDPC codes (except cycle-codes) are examples of ensembles that have a single jump at the BP threshold (regular LDPC cycle-codes have no jump and are such that $\epsilon^{\text{BP}} = \epsilon^{\text{SC}}$).

**Lemma 4.1** [BP EXIT Function for Regular LDPC Ensembles] The BP EXIT function associated with the dd pair $(\mathtt{x}^{\mathtt{l}-1}, \mathtt{x}^{\mathtt{r}-1})$ is given in parametric form by

$$h^{\text{BP}}(\epsilon) = \begin{cases} (\epsilon, 0), & \epsilon \in [0, \epsilon^{\text{BP}}), \\ (\frac{\mathtt{x}}{(1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}-1}}, (1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}}), & \mathtt{x} \in \mathcal{D} = [\mathtt{x}^{\text{BP}}, 1] \leftrightarrow \epsilon \in [\epsilon^{\text{BP}}, 1], \end{cases}$$

where $\mathtt{x}^{\text{BP}}$ denotes the location of the unique minimum of $\epsilon(\mathtt{x}) = \frac{\mathtt{x}}{(1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}-1}}$ in the range $[0,1]$ and $\epsilon^{\text{BP}} = \epsilon(\mathtt{x}^{\text{BP}})$. Moreover, $\mathtt{x}^{\text{BP}} = 0$ if and only if $\mathtt{l} = 2$, otherwise $\mathtt{x}^{\text{BP}} > 0$.

*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 any minimum of $\epsilon(\mathtt{x})$ must be a root of $q(\mathtt{x}) \triangleq 1 + ((\mathtt{l}-1)(\mathtt{r}-1)-1)(1-\mathtt{x})^{\mathtt{r}-1} - (\mathtt{l}-1)(\mathtt{r}-1)(1-\mathtt{x})^{\mathtt{r}-2}$. Using Descartes rule of signs, we see that there are either exactly two or no roots for $1-\mathtt{x} \geq 0$. Such a root is at $\mathtt{x} = 0$. It remains to locate the second root. Observe that $q(0) = 0, q(1) = 1$, $q'(0) = -(\mathtt{l}-2)(\mathtt{r}-1)$. If $\mathtt{l} > 2$, then $q'(0) < 0$, and the existence of a root in $(0,1)$ is shown by the intermediate value theorem. If $\mathtt{l} = 2$, then $q'(0) = 0$, and therefore $q(\mathtt{x}) > 0$ for $\mathtt{x} \in (0,1]$ (otherwise $q(\mathtt{x})$ would cross the x-axis at least twice according to the intermediate value theorem and would have strictly more than two roots). Therefore, there is exactly one root in $[0,1]$ which we call $\mathtt{x}^{\text{BP}}$. Finally, $\epsilon'(1) > 0$ and $\epsilon'(0) \leq 0$ show that $\mathtt{x}^{\text{BP}}$ is a minimum of $\epsilon(\mathtt{x})$. Further, $\epsilon(\mathtt{x})$ is decreasing over $[0, \mathtt{x}^{\text{BP}}]$ and increasing over $[\mathtt{x}^{\text{BP}}, 1]$. □

More complex examples have several phase transitions. This is typically the case for "practical" codes obtained after optimization. Let $J$ denote the number of such "jumps" (more precisely, the number of discontinuities of the BP EXIT curve obtained from density evolution). For example, the ensemble depicted in Figure 4.2 (right) has $J = 2$. The BP EXIT function is given in parametric form by

$$h^{\text{BP}}(\epsilon) = \begin{cases} (\epsilon, 0), & \epsilon \in [0, \epsilon^{\text{BP}}), \\ (\frac{\mathtt{x}}{(1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}-1}}, (1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}}), & \mathtt{x} \in \mathcal{D} = \bigcup_{i\in\{0\}\cup[J]} [\underline{\mathtt{x}}^i, \overline{\mathtt{x}}^i) \cup \{1\} \leftrightarrow \epsilon \in [\epsilon^{\text{BP}}, 1], \end{cases}$$

where the subdivision $0 < \underline{\mathtt{x}}^1 < \overline{\mathtt{x}}^1 < \cdots < \underline{\mathtt{x}}^J < \overline{\mathtt{x}}^J = 1$ characterizes the discontinuities of the BP EXIT function. The $J$ discontinuities appear at the points $\epsilon_j \triangleq \epsilon(\underline{\mathtt{x}}^j) = \epsilon(\overline{\mathtt{x}}^{j-1})$ for $j \in [J]$. The considered example has $J = 2$ but the previous characterization holds in general. Let us formally[3] define $\underline{\mathtt{x}}^j$ recursively as $\underline{\mathtt{x}}^j = \max\left\{\mathtt{x} \in (\underline{\mathtt{x}}^{j-1}, 1) : \mathtt{x} \text{ minimizes } \epsilon(\mathtt{x}) \text{ over } (\underline{\mathtt{x}}^{j-1}, 1) \text{ and } \epsilon(\mathtt{x}) \text{ is locally strictly convex}\right\}$, with $\underline{\mathtt{x}}^0 \triangleq 0$. This procedure will determine a finite number of discontinuities $J$. The definition of $\overline{\mathtt{x}}^j$ is then simply, $\overline{\mathtt{x}}^j \triangleq \min\left\{\mathtt{x} \in (\underline{\mathtt{x}}^j, 1) : \epsilon(\mathtt{x}) = \epsilon(\underline{\mathtt{x}}^{j+1})\right\}$ for all $j \in \{0\} \cup [J-1]$ and $\overline{\mathtt{x}}^J \triangleq 1$. Note that the BP threshold is given by $\epsilon^{\text{BP}} \triangleq \epsilon(\mathtt{x}^{\text{BP}})$ where $\underline{\mathtt{x}}^{\text{BP}} \triangleq \max\left\{\mathtt{x} \in (\underline{\mathtt{x}}^0 = 0, 1) : \mathtt{x} \text{ minimizes } \epsilon(\mathtt{x}) \text{ over } (\underline{\mathtt{x}}^0 = 0, 1)\right\}$. It is possible, for example in the case of cycle-codes, that $J = 0$. In this case $0 = \underline{\mathtt{x}}^0 = \overline{\mathtt{x}}^0 = \mathtt{x}^{\text{BP}}$, and the BP threshold equals the stability condition threshold $\epsilon^{\text{SC}} \triangleq \epsilon(0) < 1$. A more curious example is when the BP threshold equals the stability condition but $J \geq 1$. In this case $0 = \underline{\mathtt{x}}^0 = \mathtt{x}^{\text{BP}}$ but $\overline{\mathtt{x}} > 0$, and a jump occurs for $\mathtt{x} = \overline{\mathtt{x}}$. An example of this is depicted in Figure 4.2 (left).

So far we have characterized the (asymptotic average) BP EXIT function. Can we provide a similar characterization for the MAP EXIT curve? We know at least one fundamental property of the MAP EXIT function, which is the value of its integral (from the area theorem). This – combined with the obvious sub-optimality of BP decoding – will give us a way to characterize the MAP EXIT function in

[3] We use "max" and "min" in order to eliminate trivial (not connected) points of the EXIT curve: this technicality can be ignored for simplicity. Moreover, the fact that the subdivision exists and is unique follows from the fact that $\epsilon(\mathtt{x})$ is an analytic and differentiable function for $\mathtt{x} \in (0,1]$.

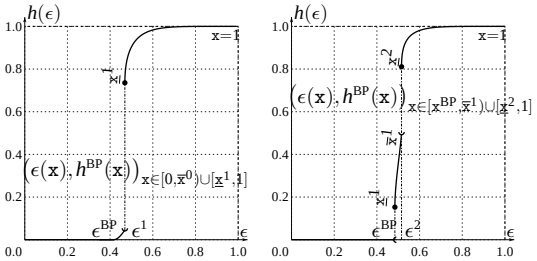



Figure 4.2: BP EXIT functions. Left: LDPC($\lambda(\mathtt{x}) = \frac{4\mathtt{x}+6\mathtt{x}}{10}, \rho(\mathtt{x}) = \mathtt{x}^6$) such that $\epsilon^{\text{BP}} = \epsilon^{\text{SC}} = \frac{5}{12} \approx 0.4167$ is obtained for $\mathtt{x} = \underline{\mathtt{x}}^0 = 0$, i.e., $\epsilon^{\text{BP}} = \epsilon^{\text{SC}}$. Moreover the number of discontinuities is $J = 1$. For $\mathtt{x} = \overline{\mathtt{x}}^0 \approx 0.04828$, i.e., at $\epsilon^1 \approx 0.4691$, a discontinuity appears and $\mathtt{x}$ "jumps" to $\underline{\mathtt{x}}^1 \approx 0.3309$. Right: LDPC($\lambda(\mathtt{x}) = \frac{3x+3x^2+4x^{13}}{10}, \rho(\mathtt{x}) = x^6$) such that $\epsilon^{\text{BP}} = \epsilon^1 \approx 0.48437$ is obtained for $\mathtt{x} = \mathtt{x}^{\text{BP}} = \underline{\mathtt{x}}^1 \approx 0.09904$. Moreover the number of discontinuities is $J = 2$, one is at $\epsilon = \epsilon^1 = \epsilon^{\text{BP}}$ and the second is at $\epsilon = \epsilon^2 \approx 0.51553$. The function is then piece-wise continuous, first between $\epsilon = 0$ and $\epsilon = \epsilon^{\text{BP}}$, second when the parameter $\mathtt{x}$ is between $\mathtt{x}^{\text{BP}} = \underline{\mathtt{x}}^1$ and $\overline{\mathtt{x}}^1 \approx 0.22156$, and third when $\mathtt{x}$ is between $\underline{\mathtt{x}}^2 \approx 0.37016$ and $\overline{\mathtt{x}}^2 = 1$.

many cases. Since the integral "under" the curve $(\epsilon(\mathtt{x}), \Lambda(\mathtt{y}(\mathtt{x})))$ (that is called the EBP EXIT curve) will appear frequently in the subsequent section, it is worth to compute it once and for all. This is easily done by applying integration by parts twice (see Appendix 4.B for details). We call this integral the *trial entropy*, a choice which was first indicated in Chapter 2 and Eq. (2.3) and which will hopefully become clear in the remainder of this chapter.

**Definition 4.3** [Trial Entropy] Consider a dd pair $(\lambda, \rho)$, define $\mathtt{y}(\mathtt{x}) \triangleq 1 - \rho(1-\mathtt{x})$ and $\epsilon(\mathtt{x}) = \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$. The associated *trial entropy* is defined as the polynomial

$$P(\mathtt{x}^*) \triangleq \int_0^{\mathtt{x}^*} \Lambda(\mathtt{y}(\mathtt{x}))\epsilon'(\mathtt{x})\mathrm{d}\mathtt{x} = \epsilon(\mathtt{x}^*)\Lambda(\mathtt{y}(\mathtt{x}^*)) + \Lambda'(1)\mathtt{x}^*(1-\mathtt{y}(\mathtt{x}^*)) - \frac{\Lambda'(1)}{\Gamma'(1)}(1-\Gamma(1-\mathtt{x}^*)).$$

**Lemma 4.2** [BP/MAP EXIT Function for LDPC Ensembles with $J = 0$] Consider a dd pair $(\lambda, \rho)$ such that $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ with $\mathtt{y}(\mathtt{x}) \triangleq 1 - \rho(1-\mathtt{x})$ is non-decreasing over $[0,1]$. (In other words, the associated number of discontinuities is $J = 0$.) Then the MAP and BP EXIT functions are equal and are given in parametric form by

$$h^{\text{MAP}}(\epsilon) = h^{\text{BP}}(\epsilon) = \begin{cases} (\epsilon, 0), & \epsilon \in [0, \epsilon^{\text{BP}}), \\ (\epsilon(\mathtt{x}), \Lambda(\mathtt{y}(\mathtt{x})), & \mathtt{x} \in \mathcal{D} = [0,1] \ \leftrightarrow \ \epsilon \in [\epsilon^{\text{BP}}, 1], \end{cases}$$

where $\epsilon^{\text{BP}} = \epsilon(0)$. Moreover the expected conditional entropy rate converges and

$$\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X|Y(\epsilon))/n] = \int_0^{\epsilon} h^{\text{MAP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon} = P(\mathtt{x}^{\epsilon}),$$

where $\mathtt{x}^{\epsilon}$ is the unique non-zero root of $\mathtt{x} = \epsilon\lambda(\mathtt{y}(\mathtt{x}))$.

*Proof.* Using the upper bound discussed in Example 2.9, we know that for any $\mathtt{G} \in \text{LDPC}(n,\lambda,\rho)$ and any $\ell \in \mathbb{N}$ we have $\frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}(\epsilon))) \le \frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{BP}(\mathtt{G}),\ell}(Y_{\sim i}(\epsilon)))$. Therefore

$$r_{\lambda,\rho} \le r_{\mathtt{G}} = \int_0^1 \frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}(\epsilon)))\mathrm{d}\epsilon \le \int_0^1 \frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{BP}(\mathtt{G}),\ell}(Y_{\sim i}(\epsilon)))\mathrm{d}\epsilon$$

If we take first the expectation over the ensemble LDPC$(n,\lambda,\rho)$, then the limsup when $n \to \infty$, and finally the limit when $\ell \to \infty$, the Fatou-Lebesgue theorem shows

$$r_{\lambda,\rho} \le \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon \le \int_0^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon.$$

A direct computation gives $\int_0^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = P(1) = r_{\lambda,\rho}$. Therefore $\int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon = \int_0^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = r_{\lambda,\rho}$. Since $h^{\text{BP}}(\epsilon)$ is continuous over $[0,1]$ (because $J=0$) and $h(\epsilon)$ is non-decreasing, it must be true that $h^{\text{MAP}}(\epsilon) = h^{\text{BP}}(\epsilon)$ for $\epsilon \in [0,1]$.

It remains to show that $\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[\frac{H_{\mathtt{G}}(X|Y(\epsilon))}{n}]$ exists (and is equal to the trial entropy). We have seen that $\lim_{n\to\infty} \int_0^1 \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\epsilon = \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$, i.e., the asymptotic average rate converges to the design rate. This implies more generally, that for any subset $U \subseteq [0,1]$ we have the equality $\text{limsup}_{n\to\infty} \int_U \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\epsilon = \int_U h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$. This is true because the left-hand side is at least as large as the right-hand side (Fatou-Lebesgue), and because we must have equality when $U=[0,1]$. Therefore,

$$\begin{aligned}
\text{limsup}_{n\to\infty} \int_0^{\epsilon} \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\tilde{\epsilon} &= \int_0^{\epsilon} h^{\text{MAP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon} = r_{\lambda,\rho} - \int_{\epsilon}^1 h^{\text{MAP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon} \\
&= r_{\lambda,\rho} - \text{limsup}_{n\to\infty} \int_{\epsilon}^1 \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\tilde{\epsilon} \\
&= \text{liminf}_{n\to\infty} \int_0^1 \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\tilde{\epsilon} \\
&\quad - \text{limsup}_{n\to\infty} \int_{\epsilon}^1 \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\tilde{\epsilon} \\
&= \text{liminf}_{n\to\infty} \int_0^{\epsilon} \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\tilde{\epsilon},
\end{aligned}$$

which shows that the limit exists and $\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[\frac{H_{\mathtt{G}}(X|Y(\epsilon))}{n}] = \int_0^{\epsilon} h^{\text{MAP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon}$. □

Discussion: In words, the previous lemma means that BP decoding is asymptotically MAP decoding whenever the BP threshold is given by the stability condition ($J=0$). In this case, the three thresholds coincide, i.e., $\epsilon^{\text{BP}} = \epsilon^{\text{MAP}} = \epsilon^{\text{SC}} \triangleq \frac{1}{\lambda'(0)\mathtt{y}'(0)} = \frac{1}{\lambda'(0)\rho'(1)}$. This happens, for example, for cycle-codes that have $\lambda(\mathtt{x}) = \mathtt{x}$.

**Example 4.1** For the dd pair $(\lambda(\mathtt{x}) = \mathtt{x}, \rho(\mathtt{x}))$, i.e., for an ensemble of LDPC cycle-codes, we get $\epsilon^{\text{MAP}} = 1/\rho'(1)$. For example, when the ensemble is regular with dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}, \mathtt{x}^{\mathtt{r}-1})$, then $\epsilon^{\text{MAP}} = 1/(\mathtt{r}-1)$.

Standard LDPC ensembles, such as regular ensembles, have a typically discontinuous BP EXIT function. In this case, a direct computation of the trial entropy (see Lemma 4.10) shows that the area under the BP EXIT function is strictly larger than the design rate. Therefore, for $J \geq 1$ and from the area theorem, we expect that the MAP EXIT function will not be point-wise equal to the BP EXIT function. Let us first focus on a class of ensembles which have a unique discontinuity ($J=1$) that occurs at the BP threshold $\epsilon^{\text{BP}}$. For technical reasons, the notion of *residual graph* introduced in Section 2.10 will appear below. Recall that the largest root of $\mathtt{x} = \epsilon\lambda(\mathtt{y}(\mathtt{x}))$, which we denote by $\mathtt{x}^{\epsilon}$, is the fixed-point of density evolution when transmission takes place over BEC($\epsilon$). If $\mathtt{x}^{\epsilon} > 0$ (i.e., above BP threshold), then BP decoding gets stuck in a stopping set, which is asymptotically described by the *residual graph*.

**Lemma 4.3** [MAP EXIT Function for LDPC Ensembles with $J=1$ at the Threshold] Consider a dd pair $(\lambda,\rho)$ such that $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ (with $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$) is non-decreasing over $[\mathtt{x}^{\text{BP}}, 1]$ (with $\mathtt{x}^{\text{BP}} \triangleq \text{argmin}_{[0,1]}(\epsilon(\mathtt{x}))$). Let $\mathtt{x}^{\epsilon} \in (0,1)$ be the largest root of $\mathtt{x} = \epsilon\lambda(\mathtt{y}(\mathtt{x}))$, and let $(\lambda_{\epsilon}, \rho_{\epsilon})$ be the dd pair of the corresponding residual graph. If there exists a channel parameter $\epsilon^* \in (0,1)$ with corresponding $\mathtt{x}^* \triangleq \mathtt{x}^{\epsilon^*}$ (largest root of $\mathtt{x} = \epsilon^*\lambda(\mathtt{y}(\mathtt{x}))$) such that $P(\mathtt{x}^*) = 0$, and if $\forall u \in (0,1)$, $\Theta_{\lambda_{\epsilon^*},\rho_{\epsilon^*}}(u) \leq 0$ (where $\Theta_{\lambda_{\epsilon^*},\rho_{\epsilon^*}}$ is defined in Section 2.3), then the MAP EXIT function is given in parametric form by

$$h^{\text{MAP}}(\epsilon) \overset{\text{a.e.}}{=} \begin{cases} (\epsilon, 0), & \epsilon \in [0, \epsilon^{\text{MAP}}], \\ (\epsilon(\mathtt{x}), \Lambda(\mathtt{y}(\mathtt{x})), & \mathtt{x} \in \mathcal{D} = (\mathtt{x}^{\text{MAP}}, 1] \ \leftrightarrow \ \epsilon \in (\epsilon^{\text{MAP}}, 1], \end{cases}$$

where $\epsilon^{\text{MAP}} \triangleq \epsilon(\mathtt{x}^{\text{MAP}})$ with $\mathtt{x}^{\text{MAP}} \triangleq \mathtt{x}^*$. Moreover the expected conditional entropy rate converges and

$$\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X|Y(\epsilon))/n] = \int_0^{\epsilon} h^{\text{MAP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon} = P(\mathtt{x}^{\epsilon}).$$

*Proof.* We prove the lemma by establishing three results.

$$\text{(i) } \forall \epsilon \in [0,1] \quad h^{\text{MAP}}(\epsilon) \le h^{\text{BP}}(\epsilon) \qquad \text{(ii) } \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon \ge r_{\lambda,\rho} \qquad \text{(iii) } \forall \epsilon \in [0,\epsilon^*) \quad h^{\text{MAP}}(\epsilon) = 0$$

Let us first see how the lemma follows from these observations. We write

$$r_{\lambda,\rho} \overset{\text{(ii)}}{\le} \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon \overset{\text{(iii)}}{=} \int_{\epsilon^*}^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$$

and observe that the evaluation of the integral under the BP EXIT function (see, e.g., Definition 4.3) gives $\int_{\epsilon^*}^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = P(1) - P(\mathtt{x}^*) = P(1) = r_{\lambda,\rho}$. This shows $\int_{\epsilon^*}^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon \le \int_{\epsilon^*}^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$ We then use (i) to see that the opposite inequality is true as well, therefore $\int_{\epsilon^*}^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = \int_{\epsilon^*}^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$. Since $h^{\text{BP}}$ is continuous over $[\epsilon^*,1]$ and $h^{\text{MAP}}(\epsilon)$ is non-decreasing it must be in fact true that $h^{\text{MAP}}(\epsilon) = h^{\text{BP}}(\epsilon)$ for $\epsilon \in (\epsilon^{\text{MAP}},1]$.

As for in (previous) Lemma 4.2, we want now to show formally that $\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[\frac{H_{\mathtt{G}}(X|Y(\epsilon))}{n}]$ exists and is equal to the trial entropy. By hypothesis (using $\epsilon = 1$ and Lemma 2.3) we see that the actual rate converges to the design rate. This means that $\limsup_{n\to\infty} \int_0^1 \mathbb{E}[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|\phi_i^{\text{MAP}})]\mathrm{d}\epsilon = \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon$. The rest of the proof follows strictly similar steps as the second part of the proof of Lemma 4.2.

It finally remains to show the three steps of the proof.

(i) Using the upper bound discussed in Example 2.9, we know that for any $\mathtt{G} \in \text{LDPC}(n,\lambda,\rho)$ and any $\ell \in \mathbb{N}$ we have $\frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{MAP}}(Y_{\sim i}(\epsilon))) \le \frac{1}{n}\sum_{i=1}^n H(X_i|\phi_i^{\text{BP},\ell}(Y_{\sim i}(\epsilon)))$. If we first take the expectation over the ensemble $\text{LDPC}(n,\lambda,\rho)$, then the limsup when $n\to\infty$, and finally the limit when $\ell\to\infty$, we get

$$\forall \epsilon \in [0,1] \quad h^{\text{MAP}}(\epsilon) \le h^{\text{BP}}(\epsilon). \tag{4.1}$$

(ii) For any $\mathtt{G} \in \text{LDPC}(n,\lambda,\rho)$, by the area theorem we have $\frac{H_{\mathtt{G}}(X)}{n} = \int_0^1 \frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))\mathrm{d}\epsilon$. If we take the expectation over the elements of the ensemble and the limit when $n\to\infty$, then $\frac{H_{\mathtt{G}}(X)}{n}$ converges to the design rate $r_{\lambda,\rho}$. To see this use the hypothesis that $\Theta_{\lambda,\rho}(u)$ achieves its unique maximum at $u = 1$ and Lemma 2.3. Therefore we can write

$$r_{\lambda,\rho} = \lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\big[\int_0^1 \frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))\mathrm{d}\epsilon\big] = \lim_{n\to\infty} \int_0^1 \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\big[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))\big]\mathrm{d}\epsilon.$$

Since the integrand is upper bounded (by 1), the Fatou-Lebesgue theorem shows

$$\begin{aligned} r_{\lambda,\rho} &= \lim_{n\to\infty} \int_0^1 \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\big[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))\big]\mathrm{d}\epsilon \\ &\le \int_0^1 \text{limsup}_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\big[\frac{1}{n}\sum_{i=1}^n H_{\mathtt{G}}(X_i|Y_{\sim i}(\epsilon))\big]\mathrm{d}\epsilon = \int_0^1 h^{\text{MAP}}(\epsilon)\mathrm{d}\epsilon. \end{aligned}$$

(iii) Let $\epsilon > \epsilon^{\text{BP}}$ denote the channel parameter, let $\mathtt{x}^{\epsilon}$ denote the corresponding fixed point of density evolution, and define $\mathtt{y}^{\epsilon} \triangleq 1-\rho(1-\mathtt{x}^{\epsilon})$. At this fixed point the expected dd pair of the residual graph, call it $\Xi_{\epsilon} = (\Lambda_{\epsilon}, \Gamma_{\epsilon})$ from a node perspective, has the form

$$\Xi_{\epsilon} = \big(\Lambda_{\epsilon}(\mathtt{z}), \Gamma_{\epsilon}(\mathtt{z})\big) \triangleq \left(\frac{\Lambda(\mathtt{zy})}{\Lambda(\mathtt{y})}, \frac{\Gamma(1-\mathtt{x}-\mathtt{xz}) - \Gamma(1-\mathtt{x}) - \mathtt{zx}\Gamma'(1-\mathtt{x})}{1-\Gamma(1-\mathtt{x}) - \mathtt{x}\Gamma'(1-\mathtt{x})}\right)$$

as shown in Section 2.10. Therefore the expected dd pair has design rate $r_{\Xi_\epsilon} = 1 - \frac{\Lambda'_\epsilon(1)}{\Gamma'_\epsilon(1)} = P(x)$. Let us first notice that $P(\mathtt{x})$ has a unique root in $(\epsilon^{\text{BP}}, 1]$, which is $\mathtt{x}^*$, such that $\forall \mathtt{x} > \mathtt{x}^*$, $P(\mathtt{x}) > 0$. To see this observe that $\epsilon(\mathtt{x})$ increasing over $(\mathtt{x}^{\text{BP}}, 1]$ implies that $P(\mathtt{x})$ increases over $(\epsilon^{\text{BP}}, 1]$.

Consider $\epsilon = \epsilon(\mathtt{x}^*) \leftrightarrow \mathtt{x} = \mathtt{x}^*$. In this case, since the assumptions in Lemma 2.3 are fulfilled by hypothesis, we find that the expected residual graph (normalized by $n$) has full rank. Since $P(\mathtt{x}^*) = 0$, we see that, for this parameter, the residual graph has in expectation the same number of variable nodes as check nodes. We therefore conclude that a MAP decoder can completely decode all bits with high probability. This means that the normalized conditional entropy must be zero. Since the conditional entropy is non-decreasing, we conclude that $\epsilon^* = \epsilon(\mathtt{x}^*) = \epsilon^{\text{MAP}}$ and that $\forall \epsilon \leq \epsilon^{\text{MAP}}$, $\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)} \frac{H_{\mathcal{G}}(X|Y(\epsilon))}{n} = 0$. This implies that $h^{\text{MAP}}(\epsilon)$ must be zero for $\epsilon \in [0, \epsilon^{\text{MAP}})$ (otherwise we would reach a contradiction via the area theorem). □

Discussion: First, note that Lemma 4.3 applies to regular ensembles LDPC$(\mathtt{x}^{\mathtt{l}-1}, \mathtt{x}^{\mathtt{r}-1})$ with $\mathtt{l} \geq 3$. Unfortunately, we are not able to provide a sufficiently compact and elegant proof to write such a general statement (although the proof is not difficult in essence). In the remainder of this chapter, we will simply provide examples for which the technical condition is fulfilled. Observe that this technical condition is easy to check so that it can be viewed as a "plug and play" criterion. Nevertheless it is worth recalling that the method based on Lemma 2.3 provides only a sufficient condition. Based on Lemma 2.3, this condition guarantees the system to be full rank. In theory we could relax this criterion and simply ask for a full rank system. Our last remark is more technical and concerns the point (iii) of the proof. Formally we are only interested in the *average* behavior of the residual graph, and the asymptotic typical dd pair suffices to describe this expected residual graph. We will nevertheless see in Section 4.2.2 that the method we used (i.e., the assumptions in Lemma 2.3) is "robust" to variations of individual degree profiles. The dd pair of a particular residual graph is indeed itself a random variable and we will see that the approach is still valid in this (practical) context.

Let us give some examples with $J = 1$ for which Lemma 4.3 applies.

**Example 4.2** For the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^3)$, we obtain $\epsilon^{\text{MAP}} = \frac{102-7\sqrt{21}}{108} \approx 0.647426$. Note that this dd pair has rate $1/4$ so that the MAP threshold should be compared to the Shannon threshold $3/4 = 0.75$.

**Example 4.3** For the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$, define $a \triangleq \frac{7 \cdot 5^{\frac{2}{3}}}{\left(11+6\sqrt{51}\right)^{\frac{1}{3}}}$ and $b \triangleq \left(55 + 30\sqrt{51}\right)^{\frac{1}{3}}$, then $\overline{\epsilon}^{\text{MAP}} = \frac{7-\sqrt{-1-a+b}-\sqrt{-2+a-b+\frac{4}{\sqrt{-1-a+b}}}}{6\left(-1+\left(-\frac{1}{6}+\frac{\sqrt{-1-a+b}}{6}+\frac{\sqrt{-2+a-b+\frac{4}{\sqrt{-1-a+b}}}}{6}\right)^5\right)^2} \approx 0.4882$. The Shannon threshold for this ensemble is $\frac{1}{2}$.

**Example 4.4** The following table compares the thresholds for various ensembles. The threshold of the first ensemble is given by the stability condition. Its exact value is $7/28 \approx 0.1786$.

| $\lambda(\mathtt{x})$ | $\rho(\mathtt{x})$ | $\epsilon^{\text{BP}}$ | $\epsilon^{\text{MAP}}$ | $\epsilon^{\text{SH}}$ |
|---|---|---|---|---|
| $\mathtt{x}$ | $\frac{2\mathtt{x}^5+3\mathtt{x}^6}{5}$ | 0.1786 | 0.1786 | 0.3048 |
| $\frac{7\mathtt{x}^2+2\mathtt{x}^3+1\mathtt{x}^4}{10}$ | $\frac{2\mathtt{x}^5+3\mathtt{x}^6}{5}$ | 0.4236 | 0.4948 | 0.5024 |
| $\frac{2857\mathtt{x}+3061.47\mathtt{x}^2+4081.53\mathtt{x}^9}{10000}$ | $\mathtt{x}^6$ | 0.4804 | 0.4935 | 0.5000 |
| $\frac{7.71429\mathtt{x}^2+2.28571\mathtt{x}^7}{10}$ | $\mathtt{x}^4$ | 0.5955 | 0.6979 | 0.7000 |
| $\frac{9\mathtt{x}^2+\mathtt{x}^7}{10}$ | $\mathtt{x}^7$ | 0.3440 | 0.3899 | 0.4000 |

## 4.2 Two (Tight) Bounds on the MAP Threshold

Let us now look at the general case. Although we will not be able to give a complete characterization, we will see that the ideas introduced in Lemma 4.3 carry over to a much wider setting. Let us come back to the points that have been used in the proof of Lemma 4.3.
Points (i) and (ii) (area theorem combined with BP sub-optimality) give an upper bound on the MAP threshold. This bound is obtained from a global upper bound on the MAP EXIT function.
Point (iii) (counting argument) is specific to the BEC. The counting argument provides a similar upper bound on the MAP threshold as the area theorem and can be further strengthened to show that the upper bound is in fact tight.
In Lemma 4.3, for $J = 1$ and under a few specific hypotheses, the complementarity[4] of the two upper bound techniques is the key ingredient that permits us to describe (not only at the threshold) the MAP EXIT curve. In this section, we clarify what we can gain from those two techniques. In other words, let us see up to what extent we are able to characterize MAP EXIT functions, in particular for ensembles with $J \geq 1$.

### 4.2.1 Upper Bound via Area Theorem and Data Processing

The key argument here is to associate the area theorem with the inequality $h^{\text{MAP}}(\epsilon) \leq h^{\text{BP}}(\epsilon)$ (see Eq. (4.1)) which shows the obvious sub-optimality[5] of BP decoding. Because of the area theorem, the integral under $h^{\text{MAP}}(\epsilon)$ is equal to (or potentially larger than) the asymptotic rate $r_{\infty,\Xi} \triangleq \liminf_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}[r_{\mathtt{G}}]$ (see (ii) in the proof of Lemma 4.3). In fact, if we ignore issues concerning the existence of limits, we expect that the integral under $h^{\text{MAP}}(\epsilon)$ equals the asymptotic rate. Let us write this as a lemma.

**Lemma 4.4** [Upper Bound via Area Theorem] Consider a dd pair $\Xi$. Let $h^{\text{BP}}$ denote the associated BP EXIT function, $r_\Xi$ denote the design rate, and $r_{\infty,\Xi} \triangleq \liminf_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\Xi)}[r_{\mathtt{G}}]$ denote the asymptotic rate. Choose $r \in [r_\Xi, r_{\infty,\Xi}]$. Let $\epsilon^*$ be the unique number in $[\epsilon^{\text{BP}}, 1]$ such that $\int_{\epsilon^*}^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = r$. Then $\epsilon^{\text{MAP}} \leq \epsilon^*$.

Discussion: Note first that if in addition $\epsilon^* = \epsilon^{\text{BP}}$ then $\epsilon^{\text{MAP}} = \epsilon^{\text{BP}}$, and in fact $\forall \epsilon \in [0,1]$ $h^{\text{BP}}(\epsilon) = h^{\text{MAP}}(\epsilon)$ and $r_{\infty,\Xi} = r_\Xi$. In the same manner, if $\epsilon^* = \epsilon^{\text{MAP}}$, then $\forall \epsilon > \epsilon^{\text{MAP}}$ $h^{\text{BP}}(\epsilon) = h^{\text{MAP}}(\epsilon)$ and $r_{\infty,\Xi} = r$. Second, a crucial observation is in order: The upper bounding technique used in Lemma 4.4 is not specific to the BEC case and we will see that it extends trivially to general BMS channels in Chapter 6.
Let us now choose $r = r_\Xi$. In that case, an upper bound on the MAP threshold is found as $\epsilon^* = \epsilon(\mathtt{x}^*)$ where $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ and $\mathtt{x}^*$ is a root of the trial entropy under some conditions. This is formalized in the next lemma.

**Lemma 4.5** [Upper Bound via Area Theorem – Explicit Characterization] Consider a dd pair $\Xi$. 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 (trial entropy)

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

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

This upper bound was found to be tight in Example 4.2, Example 4.3, and Example 4.4 of (previous) Section 4.1. However this is not always the case, as shown by the following counterexample. We have seen in Lemma 4.2 that the BP and MAP EXIT functions are point-wise equal if $J = 0$. This shows that if $J = 0$, then the MAP threshold is also given by the stability condition. The next example shows that the converse is not necessary true. BP and MAP thresholds can be equal and given by the stability condition, although their respective EXIT functions are not point-wise equal.

[4] The first two points of the proof of Lemma 4.3 were introduced in [48]. The third point of the proof of Lemma 4.3 is a sharpened version of [37]. The bound tightness can be shown under some technical conditions.
[5] The inequality $h^{\text{MAP}}(\epsilon) \leq h^{\text{BP}}(\epsilon)$ is formally obtained from the data processing inequality in Chapter 2

**Example 4.5** Consider the dd pair $(\lambda(\mathtt{x}),\rho(\mathtt{x}))=(\frac{4\mathtt{x}+6\mathtt{x}^6}{10},\mathtt{x}^6)$ and the corresponding LDPC ensemble with design rate $r_{\lambda,\rho}=1/2$. Using Lemma 2.3 we can check that $r_{\lambda,\rho}=r_{\infty,\Xi}$. 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 $\mathtt{x}=\underline{\mathtt{x}}^0=0$. Figure 4.2 (left) describes the BP EXIT function corresponding to this ensemble. Since the BP threshold is determined by the stability condition, we obtain $\epsilon^{\text{MAP}}=\epsilon^{\text{BP}}\approx 0.4167$ from Appendix 2.C. (An alternative explanation will be given by the counting argument of Section 4.2.2.) This is true despite the fact that the integral under the BP EXIT is strictly larger than $r_{\lambda,\rho}=r_{\infty,\Xi}$ (see Appendix 4.B).

More generally, the BP EXIT function has many discontinuities ($J\geq 2$), this happens when $\epsilon(\mathtt{x})$ has more than one local minimum in $(\mathtt{x}^{\text{BP}},1]$. In those cases, the simple upper bound stated in Lemma 4.4 can no longer provide a tight bound. In the next subsection, or alternatively in Section 4.4, it is shown that this upper bound can be further refined as follows.

**Lemma 4.6** [Upper Bound via Maxwell Construction – Explicit Characterization] Consider a dd pair $\Xi$. Define the polynomial $\mathtt{y}(\mathtt{x})\stackrel{\Delta}{=}1-\rho(1-\mathtt{x})$ and, for $\mathtt{x}\in(0,1]$ the function $\epsilon(\mathtt{x})\stackrel{\Delta}{=}\frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$. Let $\mathtt{x}^*$ be a root of the polynomial (trial entropy)

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

in the interval $[\mathtt{x}^{\text{BP}},1]$. Assume that there exists no $\tilde{\mathtt{x}}\in(\mathtt{x}^*,1]$ such that $\epsilon(\tilde{x})=\epsilon(\mathtt{x}^*)$. Collect all such $\mathtt{x}^*$ in the subset $\mathcal{S}^*\stackrel{\Delta}{=}\{\mathtt{x}^*:P(\mathtt{x}^*)=0,\ \nexists\tilde{\mathtt{x}}\in(\mathtt{x}^*,1]\ \epsilon(\tilde{x})=\epsilon(\mathtt{x}^*)\}$. Then $\epsilon^{\text{MAP}}\leq\epsilon^*=\epsilon(\min\mathcal{S}^*)$

Discussion: Observe that the upper bound of Lemma 4.6 is obtained from the integration of the *parametric* curve $(\epsilon(\mathtt{x}),\Lambda(\mathtt{y}(\mathtt{x})))$ which will be called EBP EXIT curve in the sequel.
In the next subsection, we provide a sufficient condition for tightness in Lemma 4.4 and Lemma 4.6. For a wide class of dd pairs the upper bound of Lemma 4.4, or at least the one of Lemma 4.6, is indeed tight. Nevertheless it might happen that there exists $\tilde{\mathtt{x}}\in(\mathtt{x}^*,1]$ such that $\epsilon(\tilde{x})=\epsilon(\mathtt{x}^*)$. In this case we expect the bound provided by Lemma 4.6 not to be tight. This will be further discussed in Section 4.3.

### 4.2.2 Tightness via Counting Argument

From Section 2.10 we know that the typical dd pair associated with the residual graph has the form

$$\Xi_\epsilon=\big(\Lambda_\epsilon(\mathtt{z}),\Gamma_\epsilon(\mathtt{z})\big)\stackrel{\Delta}{=}\left(\frac{\Lambda(\mathtt{zy})}{\Lambda(\mathtt{y})},\frac{\Gamma(1-\mathtt{x}-\mathtt{xz})-\Gamma(1-\mathtt{x})-\mathtt{zx}\Gamma'(1-\mathtt{x})}{1-\Gamma(1-\mathtt{x})-\mathtt{x}\Gamma'(1-\mathtt{x})}\right),$$

where $\mathtt{x}$ denotes the largest solution of $\mathtt{x}=\epsilon\lambda(1-\rho(1-\mathtt{x}))$ when the channel parameter is $\epsilon$ and $\mathtt{y}\stackrel{\Delta}{=}1-\rho(1-\mathtt{x})$. The associated design rate is $r_{\Xi_\epsilon}\stackrel{\Delta}{=}P(\mathtt{x})$. The corresponding function $\Theta_{\Xi_\epsilon}(u)$ of Lemma 2.3 gives a sufficient condition for the rate of the residual graph to be asymptotically the design rate. This technical condition will be used in the next theorem that is the main result of this subsection.

**Lemma 4.7** [Residual Uncertainty] Consider a dd pair $\Xi=(\Lambda,\Gamma)$. Let $\mathtt{G}$ be chosen uniformly at random from LDPC$(n,\Lambda,\Gamma)$. Assume that transmission takes place over BEC$(\epsilon)$ and let $H_{\mathtt{G}}(X|Y)$ be the conditional entropy associated with $\mathtt{G}$. Let $\Xi_\epsilon=(\Lambda_\epsilon,\Gamma_\epsilon)$ be the dd pair associated with the residual graph. Consider $\Theta_{\Xi_\epsilon}(u)$. If $\Theta_{\Xi_\epsilon}(u)$ achieves its global maximum as a function of $u\in[0,\infty)$ at $u=1$, with $\Theta''_{\Xi_\epsilon}(1)<0$, and that $\epsilon\notin\{\epsilon_j,j\in[J]\}$. Then

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}_{\text{LDPC}(n,\Xi)}[H_{\mathtt{G}}(X|Y(\epsilon))]=P(\mathtt{x}^\epsilon)\,,\tag{4.2}$$

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

*Proof.* Assume that transmission takes place over BEC$(\epsilon)$ using the code $\mathtt{G}$. We follow Section 2.10 and denote by $\mathtt{G}(\epsilon)$ the (random) residual graph after BP decoding and by $r_{\mathtt{G}(\epsilon)}$ its rate. It is straightforward to verify that, over the erasure channel, BP decoding does not exclude any codeword compatible with the received vector. This means that $H_{\mathtt{G}}(X|y(\epsilon))=nr_{\mathtt{G}(\epsilon)}$ where $y(\epsilon)$ is a particular received vector

that has led to the residual graph $\mathtt{G}(\epsilon)$. Recall that the design rate of the typical dd pair of the residual graph is $r_{\Xi_\epsilon} = P(\mathtt{x})$. Observe

$$\frac{1}{n}\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[H_{\mathtt{G}}(X|Y)] = \sum_{\tilde{\Xi}=(\tilde{\Lambda},\tilde{\Gamma})} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} \cdot \mathbb{E}_{\mathrm{LDPC}(n\tilde{\Lambda}(1),\tilde{\Xi})}[r_{\mathtt{G}(\epsilon)}],$$

where the expectations are taken with respect to codes chosen uniformly at random in the index set. By assumption $\Theta_{\Xi_\epsilon}(u)$ achieves its global maximum at $u = 1$, with $\Theta''_{\Xi_\epsilon}(1) < 0$, and $\Theta_{\Xi_\epsilon}(1) = 0$. Therefore we can find a constant $\delta > 0$ such that $\Theta_{\Xi_\epsilon}(u) \leq -\delta(1-u)^2$ for $u \in [0,1]$. We use now Appendix 4.C and Lemma 4.12 to find $\xi > 0$ such that, for any dd pair $\tilde{\Xi}$ with $d(\tilde{\Xi}, \Xi_\epsilon) \leq \xi$, we have $\Theta_{\tilde{\Xi}}(u) \leq -\delta(1-u)^2/2$ for $u \in [0,1]$. Let $\mathcal{N}_\xi$ denote the closed ball $\mathcal{N}_\xi \triangleq \{\tilde{\Xi} : d(\tilde{\Xi}, \Xi_\epsilon) \leq \xi\}$. In words $\mathcal{N}_\xi$ is the set of dd pairs $\tilde{\Xi}$ such that $d(\tilde{\Xi}, \Xi_\epsilon) \leq \xi$ where $d$ denotes the $L_1$ distance (which is defined as follows: $\forall \Xi^a = (\Lambda^a, \Gamma^a), \forall \Xi^b = (\Lambda^b, \Gamma^b)$, $d(\Xi_a, \Xi_b) = \sum_{\mathtt{l}} |\Lambda^a_{\mathtt{l}} - \Lambda^b_{\mathtt{l}}| + \sum_{\mathtt{r}} |\Gamma^a_{\mathtt{r}} - \Gamma^b_{\mathtt{r}}|$). Then

$$\begin{aligned}
\frac{1}{n}\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[H_{\mathtt{G}}(X|Y)] &\overset{(a)}{\leq} \sum_{\tilde{\Xi}\in\mathcal{N}_\xi} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} \cdot \mathbb{E}_{\mathrm{LDPC}(n\tilde{\Lambda}(1),\tilde{\Xi})}[r_{\mathtt{G}(\epsilon)}] + \Pr\{\Xi_{\mathtt{G}(\epsilon)} \notin \mathcal{N}_\xi\} \\
&\overset{(b)}{\leq} \sum_{\tilde{\Xi}\in\mathcal{N}_\xi} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} \cdot \mathbb{E}_{\mathrm{LDPC}(n\tilde{\Lambda}(1),\tilde{\Xi})}[r_{\mathtt{G}(\epsilon)}] + o_n(\xi),
\end{aligned}$$

where $(a)$ follows from $r_{\mathtt{G}(\epsilon)} \leq 1$ and $(b)$ uses Appendix 4.C and Lemma 4.11 to get $\lim_{n\to\infty} o_n(\xi) = 0$. The main step of the proof is now to apply Lemma 2.3 to any ensemble whose dd pair is in $\mathcal{N}_\xi$ (since they all fulfill the required technical conditions). We get

$$\begin{aligned}
\left|\frac{1}{n}\mathbb{E}[H_{\mathtt{G}}(X|Y)] - r_{\Xi_\epsilon}\right| &\leq \sum_{\tilde{\Xi}\in\mathcal{N}_\xi} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} |\tilde{\mathbb{E}}[r_{\mathtt{G}(\epsilon)}] - r_{\tilde{\Xi}}| + \sum_{\tilde{\Xi}\in\mathcal{N}_\xi} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} |r_{\tilde{\Xi}} - r_{\Xi_\epsilon}| + o_n(\xi) \\
&\leq \sum_{\tilde{\Xi}\in\mathcal{N}_\xi} \Pr\{\Xi_{\mathtt{G}(\epsilon)} = \tilde{\Xi}\} |r_{\tilde{\Xi}} - r_{\Xi_\epsilon}| + o'_n(\xi)
\end{aligned}$$

where $o'_n(\xi) = o_n(\xi) + C\log n/n$. Because of the continuity of the expression of the design rate, notice that there exist $B > 0$ such that for any pair $\Xi^a$, $\Xi^b$ we have $|r_{\Xi^a} - r_{\Xi^b}| \leq B d(\Xi^a, \Xi^b)$. Therefore,

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

Observe that $\xi$ can be chosen arbitrarily small, which concludes the proof. □

One consequence of Lemma 4.7 is that it permits us to compute the exact MAP threshold whenever the required conditions are verified. The next corollary gives an explicit characterization.

**Corollary 4.1** [Characterization of the MAP Threshold] Consider a dd pair $\Xi = (\Lambda, \Gamma)$. Let $\mathtt{G}$ be chosen uniformly at random from $\mathrm{LDPC}(n, \Lambda, \Gamma)$. Assume that transmission takes place over $\mathrm{BEC}(\epsilon)$ such that $\epsilon \notin \{\epsilon_j : j \in [J]\}$. Assume that $\mathtt{x}^\epsilon > 0$ is the fixed point of density evolution. Assume that $P(\mathtt{x}^\epsilon) = 0$, $\Theta_{\Xi_\epsilon}(u) \leq 0$ for $u \in [0, +\infty)$, and $\Theta''_{\Xi_\epsilon}(1) < 0$. Let $\mathcal{W} \triangleq \{u \in [0, +\infty) : u \neq 1, \Theta_{\Xi_\epsilon}(u) = 0\}$, if, for any $u \in \mathcal{W}$, $\frac{\partial \Theta_{\Xi_\epsilon}(u)}{\partial \epsilon} < \frac{\partial \Theta_{\Xi_\epsilon}(1)}{\partial \epsilon}$, then $\epsilon^{\mathrm{MAP}} = \epsilon$.

*Proof.* Let us first claim that there exists $\delta > 0$ such that the hypothesis of Lemma 4.7 is verified for any $\tilde{\epsilon} \in (\epsilon, \epsilon + \delta)$, and let us see how we conclude the proof. For any $\tilde{\epsilon} \in (\epsilon, \epsilon + \delta)$ let $\mathtt{x}^{\tilde{\epsilon}}$ be the associated fixed point of density evolution. Then $\lim_{n\to\infty} \frac{1}{n}\mathbb{E}[H(X|Y(\tilde{\epsilon}))] = P(\mathtt{x}^{\tilde{\epsilon}})$. Moreover $P(\mathtt{x}^\epsilon) = 0$ by hypothesis. Using the definition of the trial entropy as the integral of $\Lambda(\mathtt{y}(\mathtt{x}^{\tilde{\epsilon}}))$ with respect to $\epsilon(\mathtt{x}^{\tilde{\epsilon}})$, we get $\frac{\mathrm{d}P(\mathtt{x}^{\tilde{\epsilon}})}{\mathrm{d}\tilde{\epsilon}} = \Lambda(\mathtt{y}(\mathtt{x}^{\tilde{\epsilon}})) > 0$ for any $\tilde{\epsilon} > \epsilon$. Therefore $P(\mathtt{x}^{\tilde{\epsilon}}) > 0$ for any $\tilde{\epsilon} > \epsilon$. This implies $\epsilon^{\mathrm{MAP}} \leq \epsilon$. On the other hand $(\mathbb{E}[H(X|Y(\tilde{\epsilon}))])/n$ is increasing in $\tilde{\epsilon}$. This implies that $\forall \tilde{\epsilon} \in [0, \epsilon]$ $\lim_{n\to\infty} \frac{1}{n}\mathbb{E}_{\mathrm{LDPC}(n,\Xi)}[H_{\mathtt{G}}(X|Y(\tilde{\epsilon}))] = 0$ which in turn implies $\epsilon^{\mathrm{MAP}} \geq \epsilon$ and, therefore, $\epsilon^{\mathrm{MAP}} = \epsilon$.
It remains to prove the claim. By assumption $\epsilon$ is a continuity point of the BP EXIT function. Therefore the residual dd pair $\Xi_{\tilde{\epsilon}}$ is also continuous at $\tilde{\epsilon} = \epsilon$. Using Appendix 4.C and Lemma 4.12, we see that it

implies that, for any $\xi > 0$, there exists $\delta > 0$ such that $\forall \tilde{\epsilon} \in [\epsilon, \epsilon+\delta)$, $\forall u \in [0,1]$, $|\Theta_{\Xi_{\tilde{\epsilon}}}(u) - \Theta_{\Xi_\epsilon}(u)| \leq \xi(1-u)^2$. Since $\Theta''_{\Xi_{\tilde{\epsilon}}}(1) < 0$, this shows that, if $\delta$ is small enough, $u = 1$ is a local maximum of $\Theta_{\Xi_{\tilde{\epsilon}}}(u)$. The hypothesis $\frac{\partial \Theta_{\Xi_\epsilon}(u)}{\partial \epsilon}$ for $u \in \mathcal{W}$ indicates that it is in fact a global maximum. □

Discussion: The conditions of the previous corollary are relatively easy to verify. This has been done for all the exact values of MAP threshold given in this thesis. See, e.g., Example 4.4. Unfortunately, it has two main weaknesses. First, the technical condition is only a sufficient condition. Second, it only applies to the phase transition occurring at the MAP threshold and does not characterize further potential phase transitions of the MAP decoding. The picture carried by the Maxwell construction in the next chapter (although it can be only partially proved) will give us more insight into the complete behavior of a MAP decoder. It will then be very instructive to use the previous results to see which cases we can typically recover. This will be the subject of Example 4.8 in the next chapter.

## 4.3 Maxwell Construction and EBP EXIT Curve

It is pleasing, but also surprising, that (at least in many cases) the MAP performance can be derived from the behavior under BP decoding. We will see that it has an interpretation that is analog to the Maxwell construction in statistical mechanics. More interestingly, this interpretation has an operational counterpart, called the *Maxwell decoder* in the next section. The central object of this section will be the Extended BP (EBP) EXIT curve.

**Definition 4.4** [EBP EXIT Curve over BEC($\epsilon$)] The EBP EXIT curve associated with the dd pair $\Xi$ is given in parametric form by

$$(\epsilon, h^{\text{EBP}}) = (\epsilon(\mathtt{x}), h^{\text{EBP}}(\mathtt{x})) \triangleq \left( \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}, \Lambda(\mathtt{y}(\mathtt{x})) \right), \qquad \mathtt{x} \in (0,1],$$

where $\mathtt{y}(\mathtt{x}) \triangleq 1 - \rho(1-\mathtt{x})$.

Discussion: We see from the characterization of the BP EXIT function in Section 4.1 that the EBP EXIT curve is the envelope of the BP EXIT function.
The BP EXIT function is obtained when $\mathtt{x}$ corresponds to a fixed point of density evolution (i.e., the largest solution of $\mathtt{x} = \epsilon\lambda(\mathtt{y}(\mathtt{x}))$). However the EBP curve also contains branches that correspond to *unstable* fixed points of the density evolution equations (when $\epsilon(\mathtt{x})$ decreases) or to fixed points that are *stable* but *not achieved by density evolution* (considering the standard BP algorithm). Figure 4.3 shows the EBP EXIT functions corresponding to the first two dd pairs presented in this chapter. The EBP EXIT curve has its own area theorem connecting the integral of the curve to the (design) rate of the ensemble.

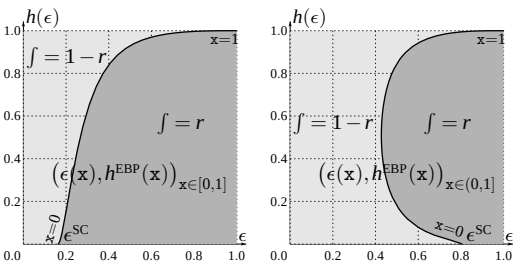



Figure 4.3: EBP EXIT functions. Left: LDPC($\lambda(\mathtt{x}) = x, \rho(\mathtt{x}) = \frac{x^3+4x^7}{5}$) with $\epsilon^{\text{SC}} = \frac{5}{31}$. Right: LDPC($\lambda(\mathtt{x}) = \frac{x+4x^3}{5}, \rho(\mathtt{x}) = \frac{x^3+4x^7}{5}$) with $\epsilon^{\text{SC}} = \frac{25}{31}$.

This might be surprising at first glance because, afterall, the EBP EXIT curve is defined in terms of the (in general) suboptimal (BP) decoder, whereas the MAP EXIT curve to which the classical area theorem applies concerns optimal (MAP) decoding. There is nevertheless a connection between the two area theorems. This is shown in Example 3.8 where an alternative proof for the EBP area theorem is given for the case $\epsilon^{\text{SC}} \leq 1$.

**Theorem 4.2** [EBP Area Theorem – BEC] Consider a dd pair $(\lambda, \rho)$ and transmission over the BEC. Let $r_{\lambda,\rho}$ be the design rate associated with the dd pair $(\lambda, \rho)$, then the EBP EXIT curve satisfies

$$\int_0^1 h^{\text{EBP}}(\mathtt{x}) \mathrm{d}\epsilon(\mathtt{x}) = r_{\lambda,\rho}.$$

*Proof.* A direct computation gives $\int_0^1 h^{\text{EBP}}(\mathtt{x})\mathrm{d}\epsilon(\mathtt{x}) = P(1) - P(0) = P(1) = r_{\lambda,\rho}$ using the trial entropy defined in 4.3. □

**Example 4.6** [EBP EXIT Curve for the (3,6) Ensemble] Figure 4.4 shows the EBP EXIT curve corresponding to the regular dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$.

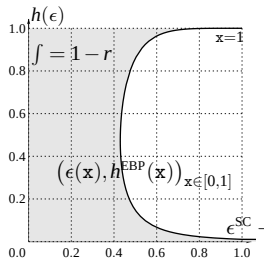



Figure 4.4: EBP EXIT function for the ensemble LDPC($\lambda(\mathtt{x}) = \mathtt{x}^2, \rho(\mathtt{x}) = \mathtt{x}^5$).

Note that for small values of $\mathtt{x}$, the EBP curve goes "outside" the unit box. This is a consequence of $\lambda'(0)\rho'(1) = 0 < 1$: for small values of $\mathtt{x}$ we have $\epsilon\lambda(1-\rho(1-\mathtt{x})) = \epsilon\lambda'(0)\rho'(1)\mathtt{x} + o(\mathtt{x}^2) = o(\mathtt{x}^2)$. Therefore, $\epsilon(\mathtt{x}) \overset{\mathtt{x}\to 0}{\to} 1/(\lambda'(0)\rho'(1)) = \infty$. But in general, even for ensembles for which $\lambda'(0)\rho'(1) > 1$, part of the EBP curve might have "$\epsilon$" coordinates larger than one. Since part of the EBP EXIT curve lies outside the unit box it is slightly more convenient here to regard the complement of this area, which is shown in grey. As predicted by Theorem 4.2, the grey area is equal to $1 - r_{\mathtt{x}^2,\mathtt{x}^5} = 1 - 3/6 = 1/2$.

Let us now combine Theorem 4.2 (the area theorem for the EBP EXIT curve) with Lemma 4.4, which gives a (provably tight) upper bound on the MAP threshold. This combination gives rise to the *Maxwell construction*. Rather than directly giving a formal description, let us first explain it by means of an example.

**Example 4.7** [Maxwell Construction for the (3,6) Ensemble] Figure 4.5 shows the Maxwell construction for the regular dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$.

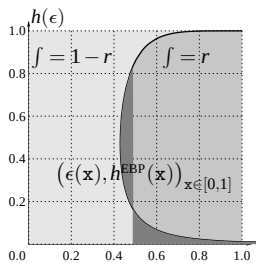



Figure 4.5: Maxwell construction for the ensemble LDPC($\lambda(\mathtt{x}) = \mathtt{x}^2, \rho(\mathtt{x}) = \mathtt{x}^5$).

This construction is as follows. Consider the associated EBP EXIT curve. Take a vertical line and adjust its position in such a way that the area to the left of the line and bounded to the left by the EBP EXIT curve is equal to the area to the right of this line and bounded above by the EBP EXIT curve. These two areas are shown in dark grey in Figure 4.5. The claim is that the unique such location of the vertical line is exactly at $\epsilon = \epsilon^{\text{MAP}}$! This is a straightforward consequence of Theorem 4.2 and Lemma 4.4 and is obtained by a direct computation whenever the MAP threshold upper bound is shown to be tight.

In the next chapter, we will give an operational interpretation of the latter two areas. In short, the balance between the two areas will be viewed as a balance between entropies. Although we will only be able to entirely characterize the right part of this balance, we conjecture that a local Maxwell construction based on the EBP EXIT curve applies at each jump of the true MAP EXIT function.
Let us therefore describe a general recursive procedure to construct this non-decreasing function, which we call *Maxwell function* and denote by $h^{\text{Maxwell}}(\epsilon)$. The Maxwell function $h^{\text{Maxwell}}(\epsilon)$ is expected to be the true MAP performance curve. This construction is again reminiscent of the Maxwell construction in thermodynamics when multiple phase transitions are observed.

Notice first (see, e.g., Appendix 4.B) that instead of $\mathtt{x} \mapsto \epsilon(\mathtt{x})$, we can alternatively consider $h \mapsto \epsilon(h) \triangleq \left(\frac{\mathtt{y}^{-1}\circ\Lambda^{-1}}{\lambda\circ\Lambda^{-1}}\right)(h)$. The function $\epsilon(h)$ describes equivalently the EBP EXIT curve. This view facilitates the description of the following recursive procedure where we "walk" on the EBP EXIT function in the direction of increasing $\mathtt{x}$. Each time we can do so, we replace a "S"-shaped part of the EBP EXIT curve $(\epsilon(\mathtt{x}), h^{\text{EBP}}(\mathtt{x}))$ by a straight (vertical) transition in order to locally satisfy the Maxwell construction. It can happen that the final number of discontinuities of the Maxwell curve differs from $J$ (which is the number of discontinuities of the BP EXIT function).
More formally, let us define recursively a sequence of functions $\epsilon^{\text{Max}(j)}(h)$, or equivalently a sequence of curves $(\epsilon^{\text{Max}(j)}, h^{\text{Max}(j)}(\epsilon^{\text{Max}(j)}))$ so that, after a finite number of steps (equal to $J$ in the standard

case), the stationary limit (more precisely and if needed, the minimum between one and its stationary limit) is the Maxwell curve $\epsilon^{\text{Maxwell}}(h) \leftrightarrow h^{\text{Maxwell}}(\epsilon)$. Choose $\epsilon^{\text{Max}(j=0)}(h) \triangleq \epsilon(h)$ (if needed, we might consider $h > 1$, see discussion on Figure 4.15 in Section 4.5). Construct the curve $\epsilon^{\text{Max}(j+1)}(h)$ as follows. Assume that the derivative of $\epsilon^{\text{Max}(j)}(h)$ changes signs exactly $p$ times (the value 0 being considered both negative and/or positive). If $p \geq 1$, then consider the subdivision $0 \leq \underline{h}^1 < \overline{h}^1 < \underline{h}^2 < \overline{h}^2 < \cdots < \underline{h}^{\lfloor \frac{p+1}{2} \rfloor} < \overline{h}^{\lfloor \frac{p+1}{2} \rfloor} < \underline{h}^{\lfloor \frac{p+1}{2} \rfloor + 1} \triangleq 1$ such that $\epsilon^{\text{Max}(j)}(h)$ is (strictly) decreasing over any interval $[\underline{h}^i, \overline{h}^i]$, and non-decreasing over any interval $[\overline{h}^i, \underline{h}^{i+1}]$ for $i \in [\lfloor \frac{p+1}{2} \rfloor]$. For $h \in (\underline{h}^1, \overline{h}^1)$, let $\mathcal{S}_h \triangleq \{\tilde{h} \in [0, +\infty) : \epsilon^{\text{Max}(j)}(\tilde{h}) = \epsilon^{\text{Max}(j)}(h)\}$ and let $h_a \triangleq \max\{\tilde{h} \in \mathcal{S}_h : \tilde{h} < h\}$, $h_b \triangleq \min\{\tilde{h} \in \mathcal{S}_h : \tilde{h} > h\}$. Then there is a unique $h$ (to see this imagine that $h^{\text{EBP}}(\mathrm{x})$ describes the interval from increasing values of $\mathrm{x}$) and associated $h_a, \mathrm{x}_a, h_b, \mathrm{x}_b$ such that $\int_{x_a}^{x_b} h^{\text{Max}(j)}(x) \mathrm{d}\epsilon(x) = 0$ (for construction $\epsilon^{\text{Max}(1)}(h)$, we think of the line $(-\infty, \lim_{\mathrm{x}\to 0}(\epsilon(\mathrm{x})]$ as part of the EBP EXIT curve). Define $\epsilon^{\text{Max}(j+1)}(h) \triangleq \epsilon^{\text{Max}(j)}(h_b)$ for any $h \in (h_a, h_b)$, and $\epsilon^{\text{Max}(j+1)}(h) \triangleq \epsilon^{\text{Max}(j)}(h)$ otherwise. Once the procedure has terminated, the function $\epsilon^{\text{Maxwell}}(h)$ is well-defined over $(0,1]$, it takes values in $[0,1]$ and is such that it is constant over $J'$ distinct non-trivial intervals, which we denote by $I_j \triangleq (\underline{h}'_j, \overline{h}'_j]$.

**Definition 4.5** [Maxwell (EXIT) Function] The *Maxwell function* associated with the dd pair $\Xi$ is denoted by $h^{\text{Maxwell}}(\epsilon)$. It is defined for $h \in [0,1]$ such that it is the inverse of the function $\epsilon^{\text{Maxwell}}(h)$ when $h \notin I_j$ for $j \in [J']$ and it is zero for $\epsilon \in [0, \min_h\{\epsilon^{\text{Maxwell}}(h)\}]$.

Discussion: Note that this function is not always continuous at $\epsilon = 1$ (see discussion on Figure 4.15 in Section 4.5). Nevertheless, it is in general the case so that, by construction, the Maxwell function fulfills an area theorem and the area under this function equals the design rate.

We conjecture that the Maxwell function describes the MAP performance of iterative coding systems. Let us see, with an example with multiple jumps, how far we can prove this conjecture. In fact, in many cases, we can only formally prove the first local Maxwell construction.

**Example 4.8** [Maxwell versus MAP EXIT Function for an Ensemble with 2 Jumps] Consider the dd pair $(\lambda(\mathrm{x}), \rho(\mathrm{x})) = (\frac{3\mathrm{x}+3\mathrm{x}^2+4\mathrm{x}^{13}}{10}, \mathrm{x}^6)$ and refer to Figure 4.6. The corresponding BP EXIT curve was shown in detail in Figure 4.2. A further discussion of this ensemble will be found in Example 4.10.

The recursive procedure described above "walks" on the EBP EXIT curve in the direction of increasing $\mathrm{x}$ and allows us to construct the Maxwell function. Let us now see where we can show that the Maxwell function coincides with the true MAP EXIT function. With this aim, we will "walk" on the EBP EXIT curve in the direction of decreasing $\mathrm{x}$ and we will apply Theorem 4.7, keeping in mind that the total integral defined by the EBP EXIT curve (i.e., the double "S"-shaped curve that describes all fixed points of the density evolution equations) equals the design rate.

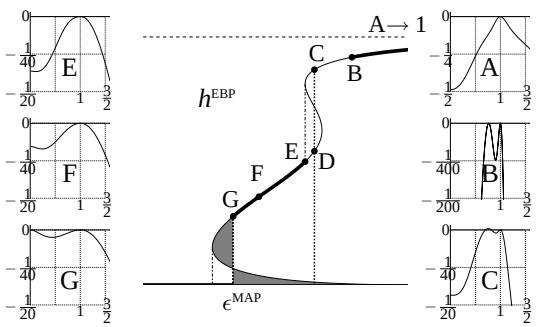



Figure 4.6: EBP EXIT function for a double-jump ensemble and function $\Theta_{\Xi}(u)$ for the typical residual ensemble in A, B, C, E, F and G.

We start with $\epsilon_A = 1$ (point A). The residual degree distribution corresponds of course to the ensemble itself. As shown in Figure 4.6 (top right picture) the hypotheses of Lemma 4.7 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 $r = 19/39 \approx 0.4872$. Now decrease $\epsilon$ smoothly. The conditions of Lemma 4.7 stay fulfilled until we get to $\epsilon_B \approx 0.5313$ (point B). At this point a second global maximum of the function $\Theta_{\Xi}(u)$ occurs. As shown in Figure 4.6 (left pictures), the hypotheses of Lemma 4.7 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, Lemma 4.7 suffices 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 can verify that a balance condition (i.e., a local Maxwell construction shown in dark grey) applies "at the jump G" (MAP threshold). But the straightforward application of Lemma 4.7 does not provide us with means of determining $h^{\text{MAP}}$ between the points B and E. Intuitively, $h^{\text{MAP}}$ should go from B to C (which corresponds to $\epsilon^C \approx 0.5156$). At this point one would hope that a second 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.
When Lemma 4.7 applies, it suffices to show 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 we conjecture that the matrix corresponding to the residual graph takes, after some suitable swapping of columns and rows, the generic form $\begin{pmatrix} U & V \\ 0 & W \end{pmatrix}$, where $W$ is a full rank square matrix of dimension $\epsilon_C(\Lambda(\mathtt{y}_C) - \Lambda(\mathtt{y}_D))$. The MAP decoder can therefore solve the part of the equation corresponding to the submatrix $W$.

In the next chapter, we provide an operational meaning of the Maxwell construction. We describe the so-called Maxwell decoder that performs MAP decoding. Instead of looking at the balance of the two dark grey areas shown in Figure 4.7 (left picture) we can consider the balance of the two dark grey areas shown in the middle and the right picture in Figure 4.7. These two areas differ only by a constant from the previous areas.

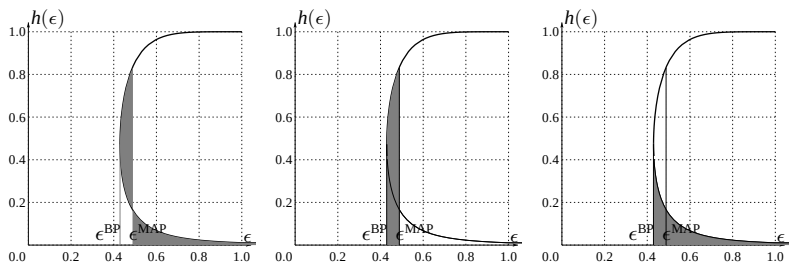



Figure 4.7: Maxwell construction for the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$ at $\epsilon = \epsilon^{\text{MAP}}$. Left: Because the MAP threshold $\epsilon^{\text{MAP}}$ is found when the two dark grey areas are in balance. Middle: The dark grey area is proportional to the total number of independent variables that the Maxwell decoder introduces (in other words, the number of guesses that a sequential Maxwell decoder has to perform). Right: The dark grey area is proportional to the total number of independent equations that are obtained during the decoding process and are used to resolve variables (in other words, the number of contradictions that a sequential Maxwell decoder will achieve).

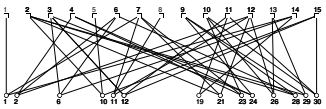
The Maxwell decoder provides an operational interpretation and further justifies our conjecture that the Maxwell function is the MAP EXIT function. A consequence of this general conjecture is that it implies a second conjecture of practical interest, i.e.,

$$\epsilon^{\text{MAP}} = \min\{\{\epsilon^* \in (0,1] : \epsilon^* = \epsilon(\mathtt{x}^*), P(\mathtt{x}^*) = 0\} \cup \{\epsilon^{\text{SC}}\}\}.$$

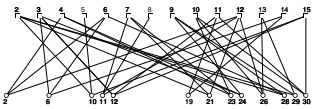
## 4.4 Maxwell Decoder

Inspired by the statistical mechanics analogy, we explain the balance condition that determines the phase transition of the MAP EXIT function by analyzing a "BP/peeling decoder with guessing." The state of the algorithm can then be associated with a point moving along the EBP EXIT curve. One consequence of this analysis is a proof of Lemma 4.6. Because of this balance condition, we term this decoder the Maxwell (M) decoder. Note that a similar algorithm is discussed in [143] although it is used for some more practical[6] concerns. Similar ideas can be also found in practical implementations of iterative decoding.

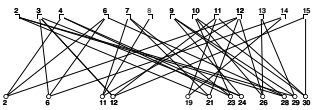
[6] Of course, for practical concerns, it might be advantageous not to guess a bit uniformly at random.



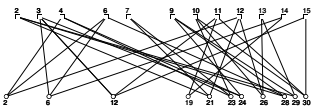
(i) Unknown bits after transmission



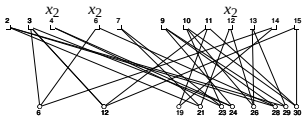
(ii) Decoding bit 1 from check 1



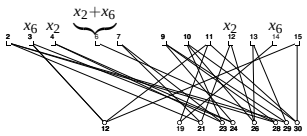
(iii) Decoding bit 10 from check 5



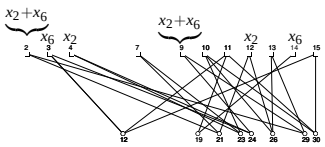
(iv) Decoding bit 11 from check 8



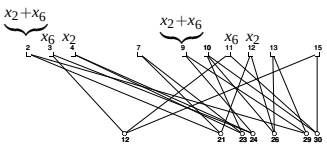

(v) Introducing $x_2$ (Guessing bit 2)



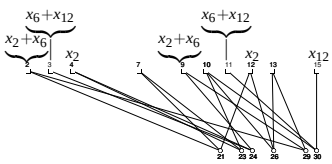

(vi) Introducing $x_6$ (Guessing bit 6)



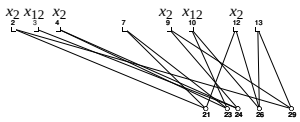

(vii) Decoding bit 28 ($x_2+x_6$) from check 6



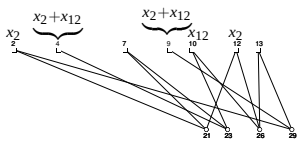

(viii) Decoding bit 19 ($x_6$) from check 14



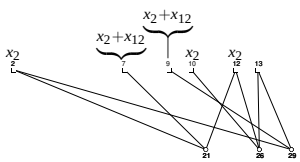

(ix) Introducing $x_{12}$ (Guessing bit 12)



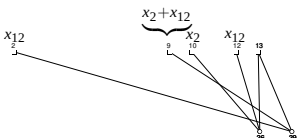

(x) Decoding bit 30 ($x_6+x_{12}=x_{12}$) from checks 11 and 15 $\longrightarrow x_6=0$



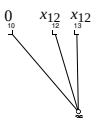

(xi) Decoding bit 24 ($x_{12}$) from check 3




(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 $\longrightarrow x_2=0$

Figure 4.8: Code of length $n=30$ and Maxwell decoder. Assume that the all-zero codeword has been transmitted. Whereas the peeling decoder in Figure 2.9 gets stuck in the stopping set (iv), the Maxwell decoder succeeds in decoding all bits. The final step (xv) is indeed as follows: Decoding bit 26 $\longrightarrow x_{12}=0$. The three successive resolutions $x_6=0$, $x_2=0$, and $x_{12}=0$ can be seen as contradictions in a pure "peeling with guessing" implementation.

Given a received word from BEC($\epsilon$), the M decoder proceeds iteratively as does the standard peeling decoder described in Section 2.10. At each time step a parity-check equation involving a single undetermined variable is chosen at random 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 from the set of yet undetermined bits and a binary (symbolic) variable $x_i$ representing the value of bit $i$ is associated with this position. The decoder proceeds further as if position $i$ was known with symbolic value $x_i$. This means that messages consist not only in values `0` or `1` but in general contain (linear combinations of) symbolic variables. In other words, the messages are really binary linear equations that state how some quantities can be expressed in terms of other quantities. It can happen that, during the decoding process, a yet undetermined variable is connected to several degree-one nodes. It will then receive a message describing its value from each of these connected degree-one check nodes. Of course, all these messages describe the *same* value (recall that over the BEC, no errors occur). Therefore, if at least one of these messages contains a symbolic variable, then the condition that all these messages describe the same value gives rise to linear equations that have to be fulfilled. Whenever this happens, the decoder resolves this set of equations with respect to some of the previously introduced variables $x_i$ and eliminates those resolved variables in the whole system. The decoding process finishes once the residual graph is empty. By definition of the process, the decoder always terminates. At this point there are two possibilities. Either all introduced variables $\{x_i\}_{i\in I}$, $I \subseteq [n]$, were resolved at some later stage of the decoding process (a special case of this being that no such variables ever had to be introduced, i.e., when the peeling decoder is successful). In this case, each bit has an associated value (either `0` or `1`) and this is the only solution compatible with the received information. In other words, the decoded word is the MAP estimate. The other possibility is that there are some undetermined variables $\{x_i\}_{i\in I}$ remaining. In this case each variable node either has already a specific value (`0` or `1`) or by definition of the decoder can be expressed as linear combination of the variables $\{x_i\}_{i\in I}$. In such a case each realization (choice) of $\{x_i\}_{i\in I} \in \{0,1\}^{|I|}$ gives rise to a valid codeword and all codewords compatible with the received information are the result of a particular choice. In other words, we have accomplished a complete list decoding, so that $|I|$ equals the conditional entropy $H(X|Y(\epsilon))$. All this is better illustrated in Figure 4.8. This shows an example where the MAP decoder succeeds in recovering all transmitted bits whereas the peeling decoder does not.

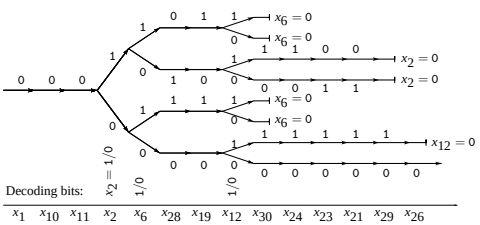



Figure 4.9: List decoding performed by the M decoder in Figure 4.8. When the M decoder gets stuck in a stopping set (before potentially introducing a new variable, i.e., guessing a new bit), then the surviving paths are compatible with the sent codeword.

Analogously to the usual peeling/BP decoder for the erasure channel, the M decoder admits two equivalent descriptions: either as a *sequential* (i.e., node-by-node in the spirit of the peeling algorithm described in Section 2.10), or as a *message-passing* algorithm (i.e., based on the general BP schedule described in Section 2.5).

The sequential approach is more intuitive and we chose it to introduce the M decoder. In particular, we can think of the M decoder as a peeling algorithm with guessing such that various simultaneous copies of the decoding are performed. In this implementation, each time that a bit $i$ is guessed, the decoder duplicates[7] the decoding process and doubles the number of running copies. If a copy encounters a contradiction (when a variable node receives non-erased messages from several check nodes which are inconsistent), then the corresponding path terminates. This intuitive approach is illustrated in Figure 4.9.

The message-passing approach allows for a simpler analysis. We follow this point of view in the sequel. The main novelty of the new analysis is a second channel parameter, call it $\gamma$, which represents the fraction of introduced variables (independent or not).

[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.

### 4.4.1 Message-Passing with Storing

Consider a variable node of index $i$. Assume it receives the channel value $\mu_i^\epsilon$ from a memoryless symmetric[8] channel with output alphabet $\{0, ❋, g\}$. More precisely, each variable node receives $\mu_i^\epsilon = 0$ with probability $1-\epsilon$, $\mu_i^\epsilon = ❋$ with probability $\epsilon(1-\gamma)$ and $\mu_i^\epsilon = g$ with probability $\epsilon\gamma$. The parameter $\gamma$ represents the fraction of introduced variables (i.e., the fraction of performed guesses).

The new message-passing algorithm employs left-to-right messages $\mu^x$ and right-to-left messages $\mu^y$, all of which take values in $\{0, ❋, g\}$. The meaning of the 0 message and the ❋ message follows naturally from the classical BP setting. A 0 message indicates a known variable,[8] a g message indicates that it carries one or a linear combination of introduced variables (i.e., a 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 $\mu_i = g$ as being shorthand for a non-empty set (or list) of indices $I_i = \{j_1, \ldots, j_k\}$. This set (or list) indicates that $x_i$ is expressible as $x_i = x_{j_1} + \cdots + x_{j_k}$, i.e., as a linear combination of introduced variables (guessed bits).

We can now write the update rules at the parity-check and variable nodes.

(i) Refer to Figure 4.10 and consider the update rule at a parity-check node of degree $r$. Assume that the index set for the $(r-1)$ messages that enter the check node is $\mathcal{R} = [r-1]$. Then

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

With respect to the classical iterative decoder, the only new rule is the one that leads to $\mu^y = g$. The reason is as follows: Assume that for all $i \in \mathcal{R}$ we have either $\mu_i^x = 0$ or $\mu_i^x = g$ and that at least one such message is 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, $\mu^y = g$ in this case. Operationally, we have $r-1$ lists (or sets) $I_1, \ldots, I_{r-1}$ (at least one of which is non-empty) entering the check node. The outgoing list $I^y$ is obtained as the *union* of the incoming lists, where indices that occur an even number of times in the incoming lists are eliminated. The list $I^y$ provides a resolution rule for $x_1 + \cdots + x_{r-1}$, and therefore for the variable connected to the outgoing edge.

In the above description 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 will see, this assumption has no influence on the proof of Lemma 4.6.

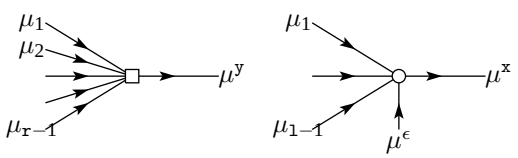



Figure 4.10: Update rule. Left: Function node. Right: Variable node.

(ii) Refer to Figure 4.10 and consider the update rule at a variable node of degree $l$. Assume that the index set for the $l-1$ messages entering the variable node is $\mathcal{L} = [l-1] \cup \{\epsilon\}$. Then

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

Once again, it should be enough to explain the rule that leads to $\mu^x = g$. Recall that 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 0, and at least one is g, then the outgoing message is g. Operationally, this means that the outgoing list is equal to *one* of the incoming

[8] Recall that the analysis is simplified by the symmetry of the channel and intermediate densities, which allows us to make the all-zero codeword assumption, see Lemma 2.4. With this assumption, the known variables and messages are equal to 0.

non-empty lists. For example, if the bit itself has been guessed (i.e., $\mu_i^{\epsilon} = \mathtt{g}$) and all other incoming messages are ❋ then the outgoing message is $\{i\}$.
From the messages we can obtain estimates of the transmitted bits. Let $\hat{\mu}_i$ denote the estimate corresponding to the $i^{\text{th}}$ variable node. In order to obtain these estimates we apply the same rule as for the variable node update with incoming messages corresponding to all of the neighboring check nodes. Formally, for a degree $\mathtt{l}$ variable node, we use $\mathcal{L} = [\mathtt{l}] \cup \{\epsilon\}$ instead of $\mathcal{L} = [\mathtt{l}-1] \cup \{\epsilon\}$.
The consistency of the estimates implies a set of linear *conditions*[9] on the guessed variables. Consider all messages $\mu_i$ entering the variable node $i$ and the associated (possibly empty) lists $I_i = \{j_1^i, \cdots, j_k^i\}$. 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

$$x_{j_1^i} + \cdots + x_{j_k^i} = \mathtt{0} \qquad \mod 2\,. \tag{4.3}$$

   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

$$x_{j_1^i} + \cdots + x_{j_k^i} = x_{j_1^l} + \cdots + x_{j_k^l} \qquad \mod 2\,. \tag{4.4}$$

   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 condition involves uniquely guessed bits (Recall that, if bit $i$ is guessed, then $\mu_i^{\epsilon} = \mathtt{g}$, and the variable $x_i$ is propagated). It can happen that a particular condition is either linearly dependent upon previous ones or is 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. Given a set of guesses, any subset of those 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.
Notice that, as the number of iterations increases, a given message can change its status according to one of the transitions ❋ $\rightarrow$ g, and $\mathtt{g}/\mathtt{0} \rightarrow \mathtt{0}$. Therefore the algorithm will stop after a finite number of iterations.

### Density Evolution Analysis

Let us now perform a density evolution analysis as in [14, 15]. Let $\mathtt{x}_{\mu^{\mathtt{x}}}^t$ ($\mathtt{y}_{\mu^{\mathtt{y}}}^t$) denote the probability that a left-to-right (right-to-left) message at iteration $\ell$ is equal to $\mu^{\mathtt{x}} \in \{\mathtt{0}, ❋, \mathtt{g}\}$.

| (i) Function node: | (ii) Variable node: |
|---|---|
| $\mathtt{y}_{\mathtt{0}}^{\ell} = \rho(\mathtt{x}_{\mathtt{0}}^{\ell})$, | $\mathtt{x}_{\mathtt{0}}^{\ell+1} = 1 - \epsilon\lambda(\mathtt{y}_{\mathtt{g}}^{\ell} + \mathtt{y}_{❋}^{\ell})$, |
| $\mathtt{y}_{❋}^{\ell} = 1 - \rho(\mathtt{x}_{\mathtt{0}}^{\ell} + \mathtt{x}_{\mathtt{g}}^{\ell}) = 1 - \rho(1 - \mathtt{x}_{❋}^{\ell})$, | $\mathtt{x}_{❋}^{\ell+1} = (1-\gamma)\epsilon\lambda(\mathtt{y}_{❋}^{\ell})$, |
| $\mathtt{y}_{\mathtt{g}}^{\ell} = 1 - \mathtt{y}_{\mathtt{0}}^{\ell} - \mathtt{y}_{❋}^{\ell} = \rho(\mathtt{x}_{\mathtt{0}}^{\ell} + \mathtt{x}_{\mathtt{g}}^{\ell}) - \rho(\mathtt{x}_{\mathtt{0}}^{\ell})$ | $\mathtt{x}_{\mathtt{g}}^{\ell+1} = \epsilon\lambda(\mathtt{y}_{\mathtt{g}}^{\ell} + \mathtt{y}_{❋}^{\ell}) - (1-\gamma)\epsilon\lambda(\mathtt{y}_{❋}^{\ell})$ |

Observe that, as expected, the sequences $\mathtt{x}_{❋}^{\ell}$ ($\mathtt{y}_{❋}^{\ell}$) and $\mathtt{x}_{❋}^{\ell} + \mathtt{x}_{\mathtt{g}}^{\ell}$ ($\mathtt{y}_{❋}^{\ell} + \mathtt{y}_{\mathtt{g}}^{\ell}$) satisfy the same density evolution equations as the fractions of erased messages in the standard BP decoder with erasure probabilities $\epsilon(1-\gamma)$ and $\gamma$ respectively. When $\ell \rightarrow \infty$, density evolution converges to a fixed point. To settle our notation, we write $(\mathtt{x}_{\mathtt{0}}^{\ell}, \mathtt{x}_{❋}^{\ell}, \mathtt{x}_{\mathtt{g}}^{\ell}) \underset{\ell\rightarrow\infty}{\longrightarrow} \big(\mathtt{x}_{\mathtt{0}}^{\infty}(\epsilon,\gamma), \mathtt{x}_{❋}^{\infty}(\epsilon,\gamma), \mathtt{x}_{\mathtt{g}}^{\infty}(\epsilon,\gamma)\big)$ and equivalently $(\mathtt{y}_{\mathtt{0}}^{\ell}, \mathtt{y}_{❋}^{\ell}, \mathtt{y}_{\mathtt{g}}^{\ell}) \underset{t\rightarrow\infty}{\longrightarrow} \big(\mathtt{y}_{\mathtt{0}}^{\infty}(\epsilon,\gamma), \mathtt{y}_{❋}^{\infty}(\epsilon,\gamma), \mathtt{y}_{\mathtt{g}}^{\infty}(\epsilon,\gamma)\big)$. Notice that $\mathtt{x}_{❋}^{\infty}(\epsilon,\gamma)$ satisfies the equation $\mathtt{x} = \epsilon(1-\gamma)\lambda(1-\rho(1-\mathtt{x}))$, and $\mathtt{x}_{\mathtt{0}}^{\infty}(\epsilon,\gamma) = \mathtt{x}_{\mathtt{0}}^{\infty}(\epsilon)$ satisfies the equation $u = \epsilon\lambda\big(1-\rho(1-u)\big)$ where $u \triangleq 1 - \mathtt{x}$. When $\ell \rightarrow \infty$, the algorithm provides estimates of the transmitted bits. Let us denote $\hat{\mu}_i^{\infty}$ the estimate associated with the $i^{\text{th}}$ variables nodes. The residual graph at the fixed point has the following structure. The variable

[9] *Conditions* are equivalent in the present setting to *contradictions*. If one thinks of guessed bits as i.i.d. uniformly random in $\{\mathtt{0}, \mathtt{1}\}$ then each new independent condition, see Eq. (4.3) and Eq. (4.4), is satisfied with probability $1/2$.

nodes such that $\hat{\mu}_i^\infty = \text{❋}$ or $\hat{\mu}_i^\infty = \text{g}$ form a stopping set: this is the largest stopping set contained in the set of variable nodes for which $\mu_i^\epsilon = \text{❋}$ or $\mu_i^\epsilon = \text{g}$. Further, the set of variable nodes such that $\hat{\mu}_i^\infty = \text{❋}$ form a stopping set contained in the previous set: this is the largest stopping set contained in the set $\mu_i^\epsilon = \text{❋}$.
In the remainder of our analysis, given a node in the bipartite graph, we will compute expectations with respect to the limiting ($\ell = \infty$) incoming messages. In those computations, we will consider that the messages are i.i.d. random variables distributed according to $\left(\mathrm{x}_0^\infty, \mathrm{x}_\text{❋}^\infty, \mathrm{x}_\text{g}^\infty\right)$ for the left-to-right messages and $\left(\mathrm{y}_0^\infty, \mathrm{y}_\text{❋}^\infty, \mathrm{y}_\text{g}^\infty\right)$ for the right-to-left messages. As long as $(\epsilon,\gamma)$ is such that $\epsilon(1-\gamma) \notin \{\epsilon_j : j \in [J]\}$ (i.e., as long as it corresponds to a continuity point of the BP EXIT function so that $\left(\mathrm{x}_0^\infty(\epsilon,\gamma), \mathrm{x}_\text{❋}^\infty(\epsilon,\gamma), \mathrm{x}_\text{g}^\infty(\epsilon,\gamma)\right)$ is continuous in $(\epsilon,\gamma)$), this is justified by the following argument. First consider messages after a finite number of iterations $\ell$. For $n$ large enough they are independent because the Tanner graph is locally a tree with high probability. Since $\epsilon(1-\gamma) \notin \{\epsilon_j : j \in [J]\}$, the number of messages that change after the $\ell^{\text{th}}$ iteration is bounded by $n \cdot o_\ell(\ell)$ with $\lim_{\ell\to\infty} o_\ell(\ell) = 0$. This argument is essentially the same as in Lemma 4.11.

### 4.4.2 Entropy Balance

An analysis using density evolution can be applied to the considered message-passing setting. However, density evolution itself does not deal with the storing of variables. Although (strictly speaking) density evolution describes *locally* the decoding behavior, the evolution of the number of resolved variable is *global*. Therefore we need an additional (global) description of the system similar to the Gibbs free energy in thermodynamics. We will see that it is relatively easy to determine the number of introduced variables (guesses). The number of resolutions (contradictions) is more difficult and we can only give an upper bound (which we expect to be tight) on this number.

#### Introduction (Guessing) Work

In the M decoder, we can introduce variables (i.e., guesses) at our convenience since the algorithm (when entirely performed) realizes a complete list decoding.

**Guessing Strategy:** Let us denote by $\hat{\mu}_i(\infty,\gamma)$ the final estimate for variable $i$ assuming that a fraction $\gamma$ of variables have been introduced (i.e., a fraction of $\gamma$ guesses, potentially dependent, have been performed). We opt for the following strategy: we increase step-by-step the fraction $\gamma$ of guessed bits. We assume that the message-passing decoding gets stuck at each step. Let us choose an explicit notation and let $\Delta\gamma$ denote such a (very small) step. Set first $\gamma = 0$. Start with the messages received via BEC$(\epsilon)$ and apply message-passing decoding until the algorithm gets stuck. Then consider each of the bits not yet determined and and set $\mu_i^\epsilon = \text{g}$ independently for each of them with probability $\Delta\gamma/(1-\gamma)$. Set $\gamma \stackrel{\Delta}{=} \gamma + \Delta\gamma$. Apply the message-passing decoder until it gets stuck. This procedure is iterated until all variables have been either guessed or decoded.
The derivation of the number of guesses becomes simple (and a real implementation more efficient!) if we take $\Delta\gamma \to 0$. This limit is always taken after $n \to \infty$. We will see that the algorithm alternates between the following two phases that are well separated. In the "guessing phase" the algorithm guesses a small fraction of bits and processes the consequences that do not propagate too far and essentially stay local. In the "contradiction phase" 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 $\Delta\gamma$.

**Useful Guesses:** Consider a point $(\epsilon,\gamma)$. Assume it does not correspond to a discontinuity point of the BP EXIT curve. Consider a variable node $i$, $i \in [n]$. The corresponding estimate provided by the M decoder is $\hat{\mu}_i(\infty,\gamma)$. Consider now moving to point $(\epsilon,\gamma+\Delta\gamma)$ ($\Delta\gamma << 1$) as follows. Assume the variable $i$ is chosen independently with probability $\Delta\gamma/(1-\gamma)$ to be guessed. If $\hat{\mu}_i(\infty,\gamma) = \text{❋}$, the channel observation on $i$ is changed from $\mu_i^\epsilon = \text{❋}$ to $\mu_i^\epsilon = \text{g}$ and the counter of newly guessed variables is increased by one. By linearity of the expectation, we get

$$\mathbb{E}[\Delta\mathbb{G}] = \frac{1}{n}\sum_{i\in[n]} \Pr\{i \text{ is chosen}\}\Pr\{\mu_i(\infty,\gamma) = \text{❋}\} = \frac{\Delta\gamma}{1-\gamma}\,\epsilon(1-\gamma)\Lambda(\mathrm{y}_\text{❋}^\infty) = \epsilon\Lambda(\mathrm{y}_\text{❋}^\infty)\Delta\gamma\,.$$

Notice that this computation assumes $n \to \infty$ and $\ell \to \infty$ afterwards. Recall that, once $\gamma$ has been changed into $\gamma + \Delta\gamma$ (introducing the $n\Delta\mathbb{G}$ new guesses), the message-passing decoder is started again until a new fixed-point is reached.

**Confirmation (Contradiction) Work**

At each step of the described strategy, it may happen that several g messages are transmitted to the same variable node $i$. Each of these lists corresponds to a distinct resolution rule for the variable $x_i$. Their convergence on the same node imposes some non-trivial conditions on the variables, which appear in the resolution rules. In this paragraph, we will estimate the number of independent conditions by exploiting (the somehow intuitive) Lemma 4.15 in Appendix 4.D. This lemma shows how to upper bound the number of contradictions via a *local* counting. Formally, we will use Lemma 4.15 directly with the original graph. This supposes that (i) we do not count contradictions generated at variable nodes receiving at least one 0 message (either from the channel or from the graph) and (ii) we count at the check node only those edges whose incoming messages are not 0. With these two conventions one can check that Lemma 4.15 holds for a general graph including degree-one check nodes as well as variable nodes that are known.[10]

Let $(\epsilon, \gamma)$ be a non-discontinuity point and denote by $n\mathbb{C}$ the number of contradictions as estimated by the right-hand side of Eq. (4.10) of Lemma 4.15 (this estimate is in fact an upper bound on the actual number of conditions). The first term counts the number of conditions arising at that node. We get

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

where $\mathbb{I}_A$ is the indicator function for the event $A$ and where $n_{\mathtt{g}}$, $n_0$, and $n_*$ count the number of incoming g, 0, and ✻ messages. Here the limits $n \to \infty$ and $\ell \to \infty$ are understood and $\mathbb{E}_{\mathtt{l}}$ denotes expectation with respect to the multinomial variables $n_0, n_{\mathtt{g}}, n_*$ with sum $\mathtt{l}$ and parameters $\mathtt{y}_0^\infty, \mathtt{y}^\infty, \mathtt{y}_*^\infty$. Note that we use the indicator function $\mathbb{I}_{n_0=0}$ because (as previously indicated) we consider only nodes "in the residual graph" (i.e., nodes that have not been determined in the standard BP phase as a consequence of the received bits). Let us temporarily adopt the shorthand $\mathtt{y}_0, \mathtt{y}, \mathtt{y}_*$ for $\mathtt{y}_0^\infty, \mathtt{y}_{\mathtt{g}}^\infty, \mathtt{y}_*^\infty$ for the density of the right-to-left messages (and the corresponding one for left-to-right messages). We get

$$\mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} \max(|\mathcal{L}_{i,\mathtt{g}}|-1,0)\right] = \epsilon(1-\gamma)\left\{\Lambda'(\mathtt{y}_*+\mathtt{y}_{\mathtt{g}})\mathtt{y}_{\mathtt{g}} - \Lambda(\mathtt{y}_*+\mathtt{y}_{\mathtt{g}}) + \Lambda(\mathtt{y}_*)\right] + \epsilon\gamma\Lambda'(\mathtt{y}_*+\mathtt{y}_{\mathtt{g}})\mathtt{y}_{\mathtt{g}}. \tag{4.5}$$

Let us now evaluate the correction term in Eq. (4.10) of Lemma 4.15. Consider a check node $j$. Assume that its "residual" degree is $\mathtt{r}'_j$. I.e., $\mathtt{r}'_j$ counts the number of edges whose incoming messages are not zero. If the corresponding $\mathtt{r}'_j$ outgoing messages are all g (equivalently, the $\mathtt{r}'_j$ incoming messages are all g), then the same condition has been overcounted $\mathtt{r}'_j - 1$ times. Let $\mathcal{C}$ denote the set of such check nodes. We have

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

where $\mathbb{E}_{\mathtt{r}}$ denotes expectation with respect to 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 easy to compute the above expectations, we get

$$\mathbb{E}\left[\frac{1}{n}\sum_{j\in\mathcal{C}}(\mathtt{r}'_j-1)\right] = \frac{\Lambda'(1)}{\Gamma'(1)}\Big(\Gamma'(1-\mathtt{x}_*)\mathtt{x}_{\mathtt{g}} - \Gamma(1-\mathtt{x}_*) + \Gamma(1-\mathtt{x}_*-\mathtt{x}_{\mathtt{g}})\Big). \tag{4.6}$$

[10] Notice that in Lemma 4.15 we assume $\mu_i^\epsilon \in \{\mathtt{g}, ✻\}$. In order to make contact with this assumption we could first run the standard BP decoder until no further progress can be made. We could then directly apply Lemma 4.15 to the residual graph. The disadvantage of this method 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 density evolution equations. Alternatively we can use Lemma 4.15 directly to the original graph under conventions (i) and (ii).

Take the difference between Eq (4.5) and Eq. (4.6), a few algebraic manipulations reveal finally that $\mathbb{E}[\mathbb{C}] = F(\mathtt{x}, \epsilon, \gamma)$, where

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

and where $\mathtt{x}$ is shorthand for the vector $(\mathtt{x}_*, \mathtt{x}_\mathtt{g}, \mathtt{x}_0, \mathtt{y}_*, \mathtt{y}_\mathtt{g}, \mathtt{y}_0)$.
We can now change $\gamma \to \gamma + \Delta\gamma$ and compute the number of new conditions on the newly guessed variables. This computation is similar to the previous description. Call $\Delta\mathbb{C}$ the upper bound on this number provided by Eq. (4.10) of Lemma 4.15. Repeating the previous 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 two distinct cases depending on the continuity or not of $\mathtt{x}^\infty(\epsilon,\gamma')$ in any $\gamma' \in [\gamma, \gamma+\Delta\gamma]$.
(i) The function $\mathtt{x}^\infty(\epsilon,\gamma')$ is continuous (hence analytic) over $[\gamma, \gamma+\Delta\gamma]$. Its Taylor expansion shows

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

where the second equality follows from evaluating the gradient of $F$ at $\mathtt{x}' = \mathtt{x}^\infty(\epsilon, \gamma+\Delta\gamma)$ (a direct calculation shows that the gradient vanishes at this point).
(ii) The interval $[\gamma, \gamma+\Delta\gamma]$ includes a discontinuity point (i.e., a jump) at $\gamma_\mathrm{j}$. Observe that $\underline{\mathtt{x}}^{j+1} = \lim_{\gamma\downarrow\gamma_\mathrm{j}} \mathtt{x}^\infty(\epsilon,\gamma)$ and $\overline{\mathtt{x}}^{j} = \lim_{\gamma\uparrow\gamma_\mathrm{j}} \mathtt{x}^\infty(\epsilon,\gamma)$ with the notations of Section 4.1. Then

$$\mathbb{E}[\Delta\mathbb{C}] = F(\underline{\mathtt{x}}^{j+1}, \epsilon, \gamma_\mathrm{j}) - F(\overline{\mathtt{x}}^{j}, \epsilon, \gamma_\mathrm{j}) + O_{\Delta\gamma}(\Delta\gamma).$$

**Work Balance**

Recall our guessing strategy. For $\gamma = 0$, the received message is first decoded with the standard message-passing (BP) decoder. Each variable is further chosen independently with probability $\Delta\gamma/(1-\gamma)$ and is guessed if it has not yet been determined (possibly in terms of former guesses). The M decoder is then applied until it gets stuck. 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 the final estimate is $\hat{\mu}_i(\infty,\gamma) \in \{\mathtt{0}, \mathtt{g}\}$ for all $i$. Without loss of generality, let us assume 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

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

where the last sums runs over the jump positions $\gamma_\mathrm{j}$ and $\Delta F_\mathrm{j} \leq F(\underline{\mathtt{x}}^{j+1}, \epsilon, \gamma_\mathrm{j}) - F(\overline{\mathtt{x}}^{j}, \epsilon, \gamma_\mathrm{j})$ indicates the discontinuity of $F$ at those positions. Finally, notice that $H_\mathtt{G}(X|Y)$ does not depend upon $\Delta\gamma$ and we can therefore take the limit $\Delta\gamma \to 0$ discarding terms in $O_{\Delta\gamma}(\Delta\gamma)$. Moreover $\mathtt{y}_*(\gamma,\epsilon) = \mathtt{y}(\mathtt{x}^{\epsilon(1-\gamma)})$ where $\mathtt{x}^{\epsilon(1-\gamma)}$ is the fixed point of density evolution at erasure probability $\epsilon(1-\gamma)$, therefore

$$\int_0^1 \epsilon\Lambda(\mathtt{y}_*(\gamma,\epsilon))\,\mathrm{d}\gamma = \int_0^1 \epsilon h^{\mathrm{BP}}(\epsilon(1-\gamma))\mathrm{d}\gamma = \int_0^\epsilon h^{\mathrm{BP}}(\tilde{\epsilon})\mathrm{d}\tilde{\epsilon} = \int_0^\epsilon \Lambda(\mathtt{y}(\tilde{\epsilon}))\,\mathrm{d}\tilde{\epsilon}.$$

This quantity is the area under the BP curve. It is depicted in dark grey in Figure 4.7 (middle). Finally, consider a discontinuity point $\epsilon_\mathrm{j} = (1-\gamma_\mathrm{j})\epsilon$ so that $\underline{\mathtt{x}}^{j+1}$ and $\overline{\mathtt{x}}^{j}$ are the corresponding fixed points of density evolution (just above and below the jump, see Section 4.1). Then

$$\Delta F_\mathrm{j} = P(\overline{\mathtt{x}}^{j}) - P(\underline{\mathtt{x}}^{j+1})$$

where $P_\epsilon(\mathtt{x})$ is the trial entropy of Definition 4.3 so that $\Delta F_\mathrm{j}$ is the area delimited by the EBP EXIT curve and a vertical line through the jump. This area is depicted in dark grey in Figure 4.7 (right).

The conditional entropy rate is therefore asymptotically equal to (or larger than) the integral associated with the BP EXIT function minus the area corresponding to each jump $\Delta F_{\mathrm{j}}$. The analysis furnished by the M decoder is more precise[11] than the upper bounding technique leading to Lemma 4.4. It shows that the presence of jumps not only degrades the *average* performance (of BP decoding versus MAP decoding), but, more precisely, each jump *coincides* with a local loss of performance. This proves Lemma 4.6. More generally, we can now draw a complete picture of the entropy balance, which is illuminated by the M decoder.

### 4.4.3 Results

The Maxwell decoder provides a fundamental *interpretation* for the balance of areas that we described in Section 4.3. Let us first summarize the previous analysis with a lemma. We will give two illustrations in the next section.

**Lemma 4.8** [Maxwell Interpretation] Consider a dd pair $(\lambda(\mathtt{x}),\rho(\mathtt{x}))$ and the associated EBP EXIT curve $h^{\text{EBP}}$. Let the subdivision $0<\underline{\mathtt{x}}^1<\overline{\mathtt{x}}^1<\cdots<\underline{\mathtt{x}}^J<\overline{\mathtt{x}}^J=1$ describe the discontinuities of the BP EXIT function, which means that $J$ discontinuities appear at the locations $\epsilon_j \triangleq \epsilon(\underline{\mathtt{x}}^j)=\epsilon(\overline{\mathtt{x}}^{j-1})$ for $j\in[J]$ (see Characterization of Theorem 4.1). Let $\mathtt{G}$ be chosen uniformly at random in $\mathrm{LDPC}(n,\lambda,\rho)$. Assume that transmission takes place over $\mathrm{BEC}(\epsilon)$, and consider the M decoder. Define $\mathtt{y}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$, $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$, $\mathtt{x}^\epsilon$ the largest fixed-point of $\mathtt{x}=\epsilon\lambda(\mathtt{y}(\mathtt{x}))$, and $\epsilon^{\text{BP}}$ the BP threshold at location $\mathtt{x}^{\text{BP}}\geq 0$. Let us further define $I \triangleq [\mathtt{x}^{\text{BP}},\overline{\mathtt{x}}^1]\cup\left(\cup_{j\in[J]}[\underline{\mathtt{x}}^j,\overline{\mathtt{x}}^j]\right)$ (with $\overline{\mathtt{x}}^1 \triangleq 1$ if $J=0$) such that $\mathtt{x}\in I \overset{\text{a.e.}}{\leftrightarrow} \epsilon(\mathtt{x})\in[\epsilon^{\text{BP}},1]$. In the same manner, let us define its complement $\mathcal{C} \triangleq [0,1]\setminus I$ ($\mathcal{C}$ is possibly empty). Let $S(\mathtt{G},\ell)$ denote the size of the residual graph at the $\ell^{\text{th}}$ iteration (including introduction of variables, i.e., including guesses). Let $\mathbb{G}(\mathtt{G},\ell)$ denote the number of introduced variables (guesses). Let $\mathbb{C}(\mathtt{G},\ell)$ denote the number of resolutions (contradictions), and let $\mathtt{H}(\mathtt{G},\ell)$ be the number of unresolved variables. Choose $\mathtt{x}\in[0,\mathtt{x}^\epsilon]$, which we call *state of the system*, then

$$\lim_{n\to\infty}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[S(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n]=s(\mathtt{x}), \qquad \limsup_{n\to\infty}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[\mathbb{C}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n]\leq \mathtt{c}(\mathtt{x}),$$
$$\lim_{n\to\infty}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[\mathbb{G}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n]=\mathtt{i}(\mathtt{x}), \qquad \liminf_{n\to\infty}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[\mathtt{H}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n]\geq \overline{h}(\mathtt{x}),$$

where the asymptotic characters are

$$s(\mathtt{x}) \triangleq \min\{1,\epsilon(\mathtt{x})h^{\text{EBP}}(\mathtt{x})\}, \qquad \mathtt{i}(\mathtt{x}) \triangleq \int_{u\in[\mathtt{x},\mathtt{x}^\epsilon]\cap I} h^{\text{EBP}}(u)\mathrm{d}\epsilon(u),$$
$$\mathtt{c}(\mathtt{x}) \triangleq \min\left\{\mathtt{i}(\mathtt{x}), -\int_{u\in[\mathtt{x},\mathtt{x}^\epsilon]\cap\mathcal{C}} h^{\text{EBP}}(u)\mathrm{d}\epsilon(u)+h^{\text{EBP}}(\mathtt{x})\int_{u\in[\mathtt{x},\mathtt{x}^\epsilon]\cap\mathcal{C}}\mathrm{d}\epsilon(u)\right\},$$
$$\overline{h}(\mathtt{x}) \triangleq \mathtt{i}(\mathtt{x})-\mathtt{c}(\mathtt{x})=\max\left\{0,\int_{u\in[\mathtt{x},\mathtt{x}^\epsilon]} h^{\text{EBP}}(u)\mathrm{d}\epsilon(u)-h^{\text{EBP}}(\mathtt{x})\int_{u\in[\mathtt{x},\mathtt{x}^\epsilon]\cap\mathcal{C}}\mathrm{d}\epsilon(u)\right\}.$$

Discussion: Numerous remarks are in order. First, as discussed in the previous section, the M decoding decomposes in distinct phases that correspond to either introducing variables (guessing phase) or resolving equations (contradiction phase). In general,[12] the number of phases coincides with the number of discontinuities of the BP EXIT curve. Furthermore (at least in cases where the inequalities are shown to be equalities, and the limsup and liminf are well-defined), the individual instances $\left(S(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n,\mathtt{C}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n,\mathtt{I}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n,\overline{h}(\mathtt{G},\lfloor \mathtt{x}n\rfloor)/n\right)$ concentrate around this asymptotic limit. Finally, observe that, in the above formulation, formulas are similar to those of the original Maxwell construction in thermodynamics (see Section 1.1). The insight is indeed similar. Where the Van der Waals curve explains the balance between the *energy* gained and spent in the system (see Gibbs free energy in the introduction), the EBP EXIT curve explains the balance between the extrinsic (information from the code) and intrinsic (information from the channel) *entropy* at a variable node.

[11] This can also be compared to a related result shown in Appendix 4.B for the corresponding quantities (area) at a dynamic level. The upper bound on the number of contradictions corresponds to the area $D_1$ in the EXIT chart depicted in Figure 3.7.

[12] Exceptions are when the number of discontinuities of the BP EXIT function is different from the one of the Maxwell function. In this case, the number of phases is in fact equal to the number of discontinuities of the Maxwell function.

Consider a dd pair that fulfills the (sufficient) criterion of Lemma 4.7 and assume $\epsilon \geq \epsilon^{\text{MAP}}$, then Lemma 4.8 provides a complete picture of the Maxwell decoding (as a consequence, a complete description of the MAP performance). This is the case of the dd pair shown in (the first) Example 4.9 of the next section. Furthermore Example 4.9 will suggest that the picture remains valid for $\epsilon < \epsilon^{\text{MAP}}$. For some cases like in Example 4.8, Lemma 4.7 does not apply and we are not able to describe entirely the MAP EXIT curve. Although the Maxwell analysis in this case is not more successful than Lemma 4.7, it strongly suggests that the intuitive picture is true. This is moreover confirmed by the experiments in (the second and last) Example 4.10 of Section 4.4.4.

### 4.4.4 Experiments

The Maxwell decoder provides an *operational* interpretation for the balance of the areas described in Section 4.3. This means that it is relatively easy to implement (although its exponential complexity makes the implementation somehow tedious).

**Example 4.9** [Regular-$(3,6)$ LDPC ensemble] Consider the regular dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\mathtt{x}^2, \mathtt{x}^5)$ for which the Maxwell function is proved to be the MAP EXIT function.

Recall that, as discussed after Lemma 4.3, more generally the Maxwell function for regular LDPC ensembles can be proved to be the MAP EXIT function. In such a case, the asymptotic characters predicted by Lemma 4.8 are exact for $\epsilon \geq \epsilon^{\text{MAP}}$. Figure 4.11 compares the evolution of the number of unresolved variables as a function of the fraction of bits determined by the decoding process (i.e, one minus the size) as predicted by Lemma 4.8 with empirical samples for $\epsilon = 1.0, \epsilon = 0.5, \epsilon = \epsilon^{\text{MAP}} \approx 0.4882$ and $\epsilon = 0.46 \in (\epsilon^{\text{BP}}, \epsilon^{\text{MAP}})$. We observe a good agreement of the practical samples with the predicted curves, even for the last case ($\epsilon = 0.46 < \epsilon^{\text{MAP}}$) for which the tightness of $\mathtt{c}(\mathtt{x})$ is not guaranteed to be tight.

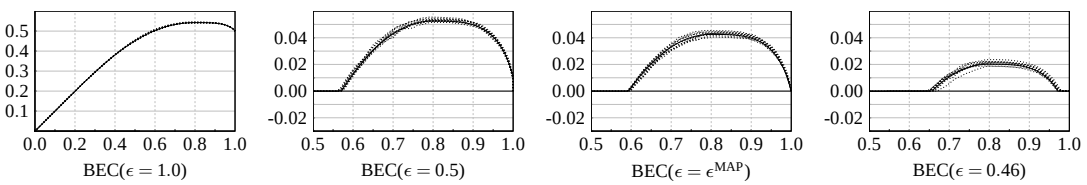



Figure 4.11: Comparison of the number of unresolved variables for the Maxwell decoder: asymptotic prediction (solid curves) versus samples for $n = 10000$ (dashed curves). The channel parameters are $\epsilon = 1.0, \epsilon = 0.50, \epsilon = \epsilon^{\text{MAP}} \approx 0.4882$ and $\epsilon^{\text{BP}} < \epsilon = 0.46 < \epsilon^{\text{MAP}}$. Note that the case associated with the channel parameter $\epsilon = 0.46$ is not entirely covered by Lemma 4.3 and Lemma 4.8. Nevertheless, there seems to be a good experimental agreement with the predicted curve.

Let us now exemplify the construction of the asymptotic curve. Figure 4.12 shows the number of unresolved variables (i.e., the number of running copies) as a function of the fraction of bits determined by the decoding process for $\epsilon = 0.46$. This means that transmission takes place over BEC($\epsilon = 0.46$), i.e., we fix the channel parameter $\epsilon$ 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 density evolution. At this point (point O in Figure 4.12), 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 variables (guesses) that the M decoder has to introduce (perform) 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. Because we are operating below the MAP threshold, (essentially) all guesses are eventually confirmed and the M decoder comes to a halt.

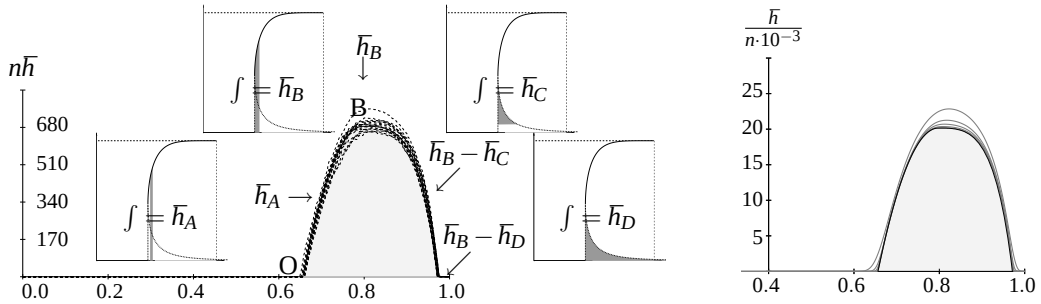



Figure 4.12: M decoder applied to the $(3,6)$-regular LDPC ensemble. Left: Experiments for 15 channel and code realizations with $\epsilon = 0.46$ and blocklength $n = 34 \cdot 10^3$ are shown (dashed curves) together with the analytic asymptotic curve (solid curve). The inserts show how the entropy profile (number of unresolved variables) can be constructed from the EXIT curve. The fraction of introduced variables (guesses) is shown in the 2 left-most inserts and the fraction of resolutions (contradictions) is shown in the 2 right inserts. Right: Expected asymptotic entropy profile (number of unresolved variables) as a function of the fraction of determined bits at $\epsilon = 0.46$ (solid curve) and empirical average profiles (grey 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).

**Example 4.10** [Standard Double-Jump LDPC Ensemble] Consider the dd pair $(\lambda, \rho) = (\frac{3x+3x^2+4x^{13}}{10}, x^6)$, which was previously investigated in Example 4.8.

Recall that corresponding LDPC ensemble has design rate $r = \frac{19}{39} \approx 0.4872$ and its BP EXIT curve has two jumps. In Example 4.8 we have discussed how large parts of the MAP EXIT curve can be constructed from Lemma 4.7. The MAP threshold is $\epsilon^{\text{MAP}} \approx 0.4913$ (at $\text{x}^{\text{MAP}} \approx 0.1434$). According to the Maxwell construction, the second MAP discontinuity is conjectured to occur at $\epsilon^{\text{MAP},2} \approx 0.5186$ (at $\underline{\text{x}}^{\text{MAP},2} \approx 0.2378$, $\overline{\text{x}}^{\text{MAP},2} \approx 0.4121$) .

Figure 4.13 shows the evolution of the fraction of unresolved variables for $\epsilon = 0.5313$. This corresponds to the point $B$ in Example 4.8, 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 empirically the Maxwell construction seems to be valid over the whole range.

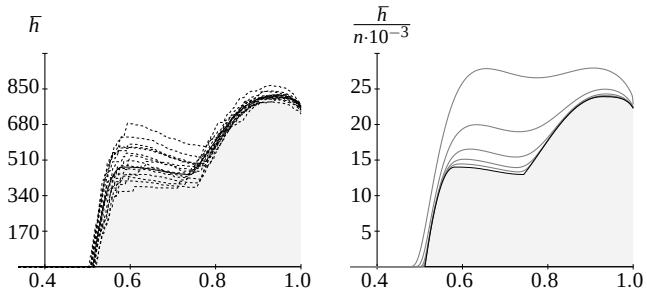



Figure 4.13: M decoder applied to our standard double-jump LDPC ensemble. Asymptotic entropy profile at $\epsilon = 0.5313$ (point $B$ in Example 4.8). Left: 15 channel and code realizations with blocklength $n = 34000$ are shown (dashed curves) together with the analytic asymptotic curve (solid curve). Right: Convergence of the average entropy curves (grey 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).

## 4.5 Conclusion and Discussion

We have seen in this chapter that the Maxwell construction associated with a suitable curve makes the bridge between the optimal MAP decoding and the iterative BP decoding. More precisely, the curve that plays the role of the Van der Waals equation is the EBP EXIT curve. The MAP phase transition is expected to be obtained from the EBP EXIT curve via a (local) Maxwell-type construction. The underlying law of a transfer of *energy* translates to a transfer of *entropy*. Operationally (when implemented as a Maxwell decoder), this transfer is expressed in terms of guesses and contradictions.

It is relatively intuitive to understand the meaning of the fraction of guesses. What about the fraction of contradictions? In a standard BP decoder, it might happen that a variable node receives twice (or more) the same message. During the decoding process (when the state of the system describes unstable fixed points) the total information a variable node receives minus the information it needs to be known represents the fraction of contradictions.[13]

In the next chapter, we will develop tools (called GEXIT functions) that will permit us to extend many of the previous observations to more general BMS channels. For the BEC, a few curiosities or peculiarities follow directly from our observations.

A first curiosity is when we investigate the analogy with thermodynamics. In practice it is possible to warm up water slightly above $100^oC$ (metastable regime). Surprisingly, a standard BP decoder works naturally in such a metastable regime if $\epsilon \in (\epsilon^{\text{BP}}, \epsilon^{\text{MAP}})$. Moreover, it is possible to think of a (purely theoretical) M decoder with "negative" guess, i.e., a BP decoder with a *unrevealing* device instead of a *guessing* device. This decoder is expected to follow hidden (stable, then metastable) branches of the EBP curve (i.e., it is expected to describe $\min(1, \mathcal{H}^{\text{EBP}}(\epsilon))$).

A second curiosity is based on the following example where the number of BP jumps and the number of MAP jumps are different. Refer to the dd pair $(\lambda, \rho) = (\frac{3\mathtt{x}+3\mathtt{x}^2+14\mathtt{x}^{50}}{20}, \mathtt{x}^{15})$ whose EBP EXIT curve is depicted in Figure 4.14. The BP EXIT curve has a single jump at $\epsilon^{\text{BP}} \approx 0.3531$ ($\mathtt{x}^{\text{BP}} \approx 0.3008$). Unfortunately Lemma 4.7 shows the tightness of the M construction only up to point A (at $\epsilon \approx 0.5063$ in Figure 4.14) . But it is quite natural to conjecture that the MAP EXIT curve has two singularities, namely at $\epsilon^{\text{MAP}} \approx 0.3986$ ($\mathtt{x}^{\text{MAP}} \approx 0.0340$) and at $\epsilon^{(\text{MAP},2)} \approx 0.4855$ ($\overline{\mathtt{x}}^{(\text{MAP},2)} \approx 0.1096$) as shown in Figure 4.14. This is validated by the M decoder that gives a residual entropy (as a fraction of the blocklength) of $\frac{\bar{h}}{n} \approx 0.0121$ 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.

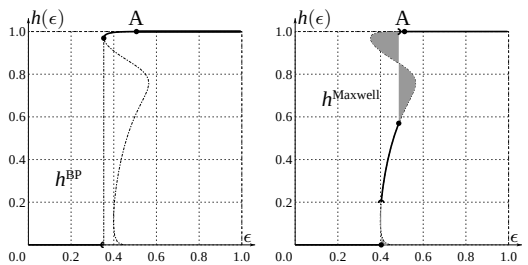



Figure 4.14: EBP EXIT function and Maxwell construction for the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\frac{3\mathtt{x}+3\mathtt{x}^2+14\mathtt{x}^{50}}{20}, \mathtt{x}^{15})$ with design rate $r_\Xi = \frac{311}{566} \approx 0.5495$. The numbers of BP and "MAP" (Maxwell) jumps (respectively, $J$ and $J'$) are different. Left: BP EXIT function with $J = 1$. Right: Maxwell construction with with $J' = 2$.

This indicates that, between the two conjectured MAP phase transitions, the M decoder follows the part of the EBP EXIT function that is "hidden" from the BP decoder. The Maxwell construction is conjectured to hold in this case.

This example provides some hints on the relationship between design rate and asymptotic rate.

First, imagine that we use the typical residual dd pair obtained from the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\frac{3\mathtt{x}+3\mathtt{x}^2+14\mathtt{x}^{50}}{20}, \mathtt{x}^{15})$ when transmission occurs at $\epsilon = 0.5$. Figure 4.15 depicts this "pathological" case where the EBP curve goes out of the unit box. The sufficient condition obtained from $\Theta_\Xi$ is not fulfilled for the new dd pair. We conjecture, however, that the Maxwell construction holds: This means that we conjecture that the design rate is still the actual rate.

Second, imagine now that we use the typical residual dd pair obtained from the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x})) = (\frac{3\mathtt{x}+3\mathtt{x}^2+14\mathtt{x}^{50}}{20}, \mathtt{x}^{15})$ when transmission occurs at a point located slightly to the left of the second phase

[13] This view also makes the link between the EBP curve and the dynamical (EXIT chart) representation of the decoding process (see also the conclusion of Chapter 3).

discontinuity of the Maxwell function. It corresponds to a state $\mathtt{x}$ with $P(\mathtt{x}) < 0$, i.e., it corresponds to a LDPC ensemble with "negative" design rate. However, we can choose this point such that a "hidden" branch of the EBP curve lies below the initial point. If the Maxwell construction holds, then the actual rate of the system is positive and the code is well-defined.
In fact, there is an even more explicit case which shows that there are ensembles for which the design rate and the actual rate are different. Imagine an ensemble with an EBP EXIT curve almost similar to the one in Figure 4.15 but such that (i) the BP threshold is given by the stability condition and (ii) the area outside the unit box is larger than the top grey area. Then, one can think of an example with design rate equal to zero (if the area outside the unit box is equal to the top grey area plus the bottom area inside the unit box). However, a simple application of Appendix 2.C shows that the MAP threshold is equal to the BP threshold. We further expect (by a simple use of the area theorem) the true asymptotic rate of the ensemble to be strictly positive[14] and therefore different from the design rate!

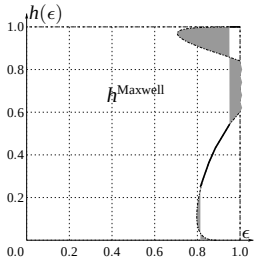



Figure 4.15: Maxwell construction for the dd pair obtained from $(\lambda(\mathtt{x}),\rho(\mathtt{x})) = (\frac{3\mathtt{x}+3\mathtt{x}^2+14\mathtt{x}^{50}}{20},\mathtt{x}^{15})$ at $\epsilon = 0.5$ ($r_\Xi \approx 0.098976$).

More than a third curiosity, we would like to point out a last but very important remark. Beyond a global theory for our observation, a further interesting research would be the analysis of more general combinatorial search problems through a suitable Maxwell construction. An example consists in the problem of satisfiability of random sparse linear systems ("XORSAT" problem) considered in [38, 144]. This problem is extremely close to the topic of this thesis; for example, a Maxwell construction can be performed for the Poisson distribution considered in [38]. The counting argument presented in Chapter 4 is in fact closely related to the approach of these papers. Therefore we hope that ideas presented in our work can be used to analyze the behavior of simple resolution algorithms (for which numerical results are presented in [145]).

# Appendix

## 4.A Concentration of Entropy

The MAP performance of sparse graph codes in the asymptotic limit is investigated in this thesis. In practice, our task is made much easier by realizing that we can restrict our study to the *average* of such a performance. More precisely, let $\mathtt{G} = \mathtt{G}(n)$ be chosen uniformly at random from $\mathrm{LDPC}(n,\lambda,\rho)$, assume that $X$ is chosen uniformly at random from $\mathtt{G}$, $Y$ is the received word, and let $H_{\mathtt{G}(n)}(X|Y)$ be the associated conditional entropy. The following theorems hold for general BMS channels.

**Theorem 4.3** [Concentration of Conditional Entropy] Let $\mathtt{G}(n)$ be chosen uniformly at random from $\mathrm{LDPC}(n,\lambda,\rho)$. Assume that $\mathtt{G}(n)$ is used to transmit over a BMS channel and let $H_{\mathtt{G}(n)} \triangleq H_{\mathtt{G}(n)}(X\,|\,Y)$ be the associated conditional entropy. Then for any $\xi > 0$, $\Pr\left\{|H_{\mathtt{G}(n)} - \mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}\left[H_{\mathtt{G}(n)}\right]| > n\xi\right\} \leq 2\,\mathrm{e}^{-nB\xi^2}$, where $B = 1/(2(\mathtt{r}_{\max}+1)^2(1-r))$ and where $\mathtt{r}_{\max}$ is the maximum check node degree.

*Proof.* The proof uses the standard technique. We first construct a (Doob's) martingale with bounded differences and then apply the Hoeffding-Azuma inequality. The complete proof can be found in [31]. It is reported in an adapted and streamlined form in the following arguments.
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's) martingale $Z_t \triangleq \mathbb{E}[H_{\mathtt{G}(n)}|\mathtt{G}_t]$. The martingale property $\mathbb{E}[Z_{t+1}\,|\,Z_0,\ldots,Z_t] = Z_t$ follows by construction. Let us write $Z_t = Z(\mathtt{G}_t)$ to stress that $Z_t$ is a (deterministic) function of the

[14] This would be consistent with the hypothesis of non-uniform priors.

random variable $\mathtt{G}_t$. Then $Z_0 = \mathbb{E}[H_{\mathtt{G}(n)}]$ is the expected conditional entropy over the code ensemble, and $Z_m = H_{\mathtt{G}(n)} \triangleq H_{\mathtt{G}(n)}(X|Y)$ is the conditional entropy for a random code $\mathtt{G}$. Therefore Theorem 4.3 follows from the Hoeffding-Azuma inequality, once we bound the difference $|Z_{t+1} - Z_t|$.
It remains to bound the difference $|Z_{t+1} - Z_t|$. Assume for the sake of definiteness that parity-check equations 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 equations 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 that differ only in the $(t+1)^{\text{th}}$ check. Let $\mathtt{G}$ be a code uniformly distributed over LDPC$(\lambda,\rho,n)$ 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 as described in [14]. This procedure results in the "pairing up" of graphs. In order to obtain the desired result, it is now enough to show that $|H_{\mathtt{G}(n)}(X|Y) - H_{\mathtt{G}'(n)}(X|Y)| \le \alpha$, for some constant $\alpha$ independent of $n$. 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 parity-check constraint $x_{i_1} + \dots + x_{i_\mathtt{r}} = 0$. Define the corresponding parity bit $\tilde{x} = x_{i_1} + \dots + x_{i_\mathtt{r}}$, Then $H_\mathtt{G}(X|Y) = H_\mathtt{G}(X|\tilde{X},Y) + H_\mathtt{G}(\tilde{X}|Y) - H_\mathtt{G}(\tilde{X}|X,Y) = H_\mathtt{G}(X|\tilde{X}=0,Y) + H_\mathtt{G}(\tilde{X}|Y) = H_{\mathtt{G}+1}(X|Y) + H_\mathtt{G}(\tilde{X}|Y)$. The second equality follows by using the channel symmetry and the fact $H_\mathtt{G}(\tilde{X}|X,Y) = 0$. 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 $|H_\mathtt{G}(X|Y) - H_{\mathtt{G}+1}(X|Y)| \le 1$. Recall that $\mathtt{G}$ and $\mathtt{G}'$ differ in at most $(\mathtt{r}+1)$ parity-check equations, where $\mathtt{r}$ is upper bounded by $\mathtt{r}_{\max}$, the maximum check node degree. Therefore the previous equation, which states $|H_\mathtt{G}(X|Y) - H_{\mathtt{G}+1}(X|Y)| \le 1$, implies $|H_\mathtt{G}(X|Y) - H_{\mathtt{G}'}(X|Y)| \le (\mathtt{r}+1)$. This concludes the proof. □

Let us now consider the concentration of the MAP EXIT function. Characterization (iv) in Lemma 3.4 implies that the (G)EXIT curve is equivalently defined as $\frac{\mathrm{d}H_{\mathtt{G}(n)}(X|Y(\epsilon))}{n\mathrm{d}\epsilon}$ when transmission takes place over BEC$(\epsilon)$. We state this concentration result in a somehow more general form when the channel is any BMS channel.

**Theorem 4.4** [Concentration of MAP GEXIT Function] Let $\mathtt{G}$ be chosen uniformly at random from LDPC$(n,\lambda,\rho)$ and let $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ denote a family of BMS channels ordered by physical degradation (with BMS$(\epsilon')$ physically degraded with respect to BMS$(\epsilon)$ whenever $\epsilon' > \epsilon$) and smooth[15] with respect to $\epsilon$. Assume that $\mathtt{G}$ is used to transmit over BMS$(\epsilon)$. Let $H_{\mathtt{G}(n)} \triangleq H_\mathtt{G}(X|Y)$ be the associated conditional entropy. Denote by $\frac{\mathrm{d}H_{\mathtt{G}(n)}}{\mathrm{d}\epsilon}$ the derivative[15] of $H_{\mathtt{G}(n)}$ with respect to $\epsilon$ 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 $\epsilon$. Then, for any $\epsilon \in J$ and $\xi > 0$ there exists an $\alpha_\xi > 0$ such that, for $n$ large enough, $\Pr\left\{\left|\frac{\mathrm{d}H_{\mathtt{G}(n)}}{\mathrm{d}\epsilon} - \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\left[\frac{\mathrm{d}H_{\mathtt{G}(n)}}{\mathrm{d}\epsilon}\right]\right| > n\xi\right\} \le \mathrm{e}^{-n\alpha_\xi}$. 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 $\alpha_\xi > A\xi^4$.

*Proof.* Let $\mathfrak{h}_n(\epsilon) \triangleq \frac{1}{n}H_{\mathtt{G}(n)}(X|Y)$ be the entropy rate, let $\mathfrak{h}'_n(\epsilon) \triangleq \frac{\mathrm{d}H_{\mathtt{G}(n)}(X|Y)}{n\mathrm{d}\epsilon}$ be its derivative, and let $\overline{\mathfrak{h}}_n(\epsilon) \triangleq \frac{1}{n}\mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[H_\mathtt{G}(X|Y)]$ be its expected value. Since the channel family $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ is smooth and ordered by physical degradation, $\mathfrak{h}_n(\epsilon)$ is a differentiable convex function of $\epsilon \in I$. Therefore

$$\frac{1}{\Delta}[\mathfrak{h}_n(\epsilon) - \mathfrak{h}_n(\epsilon-\Delta)] \le \mathfrak{h}'_n(\epsilon) \le \frac{1}{\Delta}[\mathfrak{h}_n(\epsilon+\Delta) - \mathfrak{h}_n(\epsilon)], \tag{4.7}$$

for any $\Delta > 0$ such that $[\epsilon-\Delta, \epsilon+\Delta] \in I$. Because of Theorem 4.3, we also have $\frac{1}{\Delta}[\overline{\mathfrak{h}}_n(\epsilon) - \overline{\mathfrak{h}}_n(\epsilon-\Delta) - 2\tilde{\xi}] \le \mathfrak{h}'_n(\epsilon) \le \frac{1}{\Delta}[\overline{\mathfrak{h}}_n(\epsilon+\Delta) - \overline{\mathfrak{h}}_n(\epsilon) + 2\tilde{\xi}]$, 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 $\epsilon$). By averaging (4.7) over the code $\mathtt{G}$, and subtracting it from the last equation, we get $|\mathfrak{h}'_n(\epsilon) - \overline{\mathfrak{h}}'_n(\epsilon)| \le \frac{1}{\Delta}[\overline{\mathfrak{h}}_n(\epsilon+\Delta) - 2\overline{\mathfrak{h}}_n(\epsilon) + \overline{\mathfrak{h}}_n(\epsilon-\Delta) + 2\tilde{\xi}]$. Now by using the convexity of $\overline{\mathfrak{h}}_n(\epsilon)$ and fixing $\Delta = \tilde{\xi}^{1/2}$ we get

[15] See Chapter 5: The derivative $\frac{\mathrm{d}H_{\mathtt{G}(n)}}{\mathrm{d}\epsilon}$ exists because of the explicit calculation presented in Chapter 5

$|\mathfrak{h}'_n(\epsilon) - \overline{\mathfrak{h}}'_n(\epsilon)| \leq [\overline{\mathfrak{h}}'_n(\epsilon+\tilde{\xi}^{1/2}) - \overline{\mathfrak{h}}'_n(\epsilon-\tilde{\xi}^{1/2})] + 2\tilde{\xi}^{1/2}$. The functions $\overline{\mathfrak{h}}_n$ are differentiable and convex and (by hypothesis) they converge to $\overline{\mathfrak{h}}(\epsilon) = h^{\text{MAP}}(\epsilon) = \lim_{n\to\infty} \frac{1}{n}\mathbb{E}H_{\mathtt{G}(n)}(X_1|Y_{\sim 1})$ which is differentiable in $J$. It is a standard result in convex analysis, see, e.g., [146], that the derivatives $\overline{\mathfrak{h}}'_n$ converge to $\overline{\mathfrak{h}}'$ uniformly in $J$. Therefore, there exists a sequence $\delta_n \to 0$, such that $|\mathfrak{h}'_n(\epsilon) - \overline{\mathfrak{h}}'_n(\epsilon)| \leq [\overline{\mathfrak{h}}'(\epsilon+\tilde{\xi}^{1/2}) - \overline{\mathfrak{h}}'(\epsilon-\tilde{\xi}^{1/2})] + \delta_n + 2\tilde{\xi}^{1/2}$ 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 $[\overline{\mathfrak{h}}'(\epsilon+\tilde{\xi}^{1/2}) - \overline{\mathfrak{h}}'(\epsilon-\tilde{\xi}^{1/2})] + 2\tilde{\xi}^{1/2} < \xi/2$. Then the thesis holds with $\alpha_\xi = B\tilde{\xi}_*^2(\xi)/2$. In particular, if $\mathfrak{h}(\epsilon)$ is twice differentiable with respect to $\epsilon \in J$, then $[\overline{\mathfrak{h}}'(\epsilon+\tilde{\xi}^{1/2}) - \overline{\mathfrak{h}}'(\epsilon-\tilde{\xi}^{1/2})] \leq \tilde{A}\tilde{\xi}^{1/2}$ and $\tilde{\xi}_*(\xi) \geq \tilde{A}'\xi^2$. □

Notice that Theorem 4.4 has two extra hypotheses with respect to Theorem 4.3. First, we assumed that the channel family $\{\text{BMS}(\epsilon)\}_{\epsilon\in I}$ is ordered by physical degradation. This ensures that $\frac{\mathrm{d}H_{\mathtt{G}(n)}}{\mathrm{d}\epsilon}$ is non-negative. This condition is trivially satisfied for the family $\{\text{BEC}(\epsilon)\}_{\epsilon\in[0,1]}$. More generally, we can let $\epsilon$ be any differentiable and increasing function of the erasure probability that takes values from zero to one. The second condition, namely the existence and differentiability of the expected entropy per bit in the limit, is instead crucial. The asymptotic MAP EXIT function may have jumps which coincide with discontinuities in the derivative of the conditional entropy. At a jump $\epsilon_*$, the value of the EXIT function may vary dramatically when passing from one element $\mathtt{G}$ of the ensemble to the other. Some (a finite fraction) of the codes $\mathtt{G}$ will perform well, and have an EXIT function close to the asymptotic value at $\epsilon_* - \delta$, whereas others (a finite fraction) may have an EXIT function close to the asymptotic value at $\epsilon_* + \delta$, for $\delta > 0$.

**Theorem 4.5** [Concentration of BP EXIT Curve] Let $\mathtt{G}$ be chosen uniformly at random from the ensemble LDPC$(n,\lambda,\rho)$. Assume that $\mathtt{G}$ is used to transmit over a BMS channel and let $\Phi_i^{\text{BP}(\mathtt{G}),\ell} = \phi_i^{\text{BP}(\mathtt{G}),\ell}(Y_{\sim i})$ denote the extrinsic estimate (conditional mean) of $X_i$ produced by the BP decoder after $\ell$ iterations. Then, for all $\xi > 0$, there exists $\alpha_\xi > 0$, such that

$$\Pr\left\{\left|\sum_{i=1}^{n}\left(H(X_i|\Phi_i^{\text{BP}(\mathtt{G}),\ell}) - \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\left[H(X_i|\Phi_i^{\text{BP}(\mathtt{G}),\ell})\right]\right)\right| > n\xi\right\} \leq \mathrm{e}^{-\alpha_\xi n}.$$

*Proof.* The proof is virtually identical to those given in [12, 14] where the bit/block error probability is considered. □

## 4.B Area and BP EXIT Function

The trial entropy is computed using integration by parts twice. This elementary computation appears frequently in this chapter. Hence let us give some more details. Observe that the function $\mathtt{x} \mapsto h^{\text{EBP}} \triangleq \Lambda(\mathtt{y}(\mathtt{x}))$ is composed of two functions $\mathtt{y}$ and $\Lambda$ that 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 $[0,1]$. Then the values $\epsilon(\mathtt{x}) \triangleq \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}$ can be equivalently described by $\epsilon(h) \triangleq \left(\frac{\mathtt{y}^{-1}\circ\Lambda^{-1}}{\lambda\circ\Lambda^{-1}}\right)(h)$.

**Lemma 4.9** Given a dd pair $(\lambda,\rho)$ and any $\mathtt{x}_a, \mathtt{x}_b \in [0,1]$, define $\epsilon(h) \triangleq \left(\frac{\mathtt{y}^{-1}\circ\Lambda^{-1}}{\lambda\circ\Lambda^{-1}}\right)(h)$, $h^{\text{EBP}}(\mathtt{x}) \triangleq (\Lambda \circ \mathtt{y})(\mathtt{x})$, $h_a \triangleq h^{\text{EBP}}(\mathtt{x}_a)$, and $h_b \triangleq h^{\text{EBP}}(\mathtt{x}_b)$. Then

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

Alternatively, define $\epsilon_a \triangleq \epsilon(h_a) = \frac{\mathtt{x}_a}{\lambda(\mathtt{y}(\mathtt{x}_a))}$ and $\epsilon_b \triangleq \epsilon(h_b) = \frac{\mathtt{x}_b}{\lambda(\mathtt{y}(\mathtt{x}_b))}$. Then

$$\int_{\mathtt{x}_a}^{\mathtt{x}_b} h^{\text{EBP}}(\mathtt{x})\mathrm{d}\epsilon(\mathtt{x}) = \Lambda'(1)\Big(\epsilon_b\int_0^{\mathtt{y}(\mathtt{x}_b)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \epsilon_a\int_0^{\mathtt{y}(\mathtt{x}_a)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \mathtt{x}_b\mathtt{y}(\mathtt{x}_b) + \mathtt{x}_a\mathtt{y}(\mathtt{x}_a) + \int_{\mathtt{x}_a}^{\mathtt{x}_b} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x}\Big).$$

*Proof.* This first equation follows from integration by parts after having parametrized $\epsilon(h) = \frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))} = \epsilon(\mathtt{x})$, $h = h^{\text{EBP}}(\mathtt{x})$ and observing that $\epsilon(\mathtt{x}) \cdot \frac{\mathrm{d}h^{\text{EBP}}(\mathtt{x})}{\mathrm{d}\mathtt{x}} = \frac{\mathtt{x}}{\lambda \circ \mathtt{y}(\mathtt{x})} \cdot \frac{(\lambda \circ \mathtt{y})(\mathtt{x}) \cdot \mathtt{y}'(\mathtt{x})}{\int \lambda} = \frac{\mathtt{x}\mathtt{y}'(\mathtt{x})}{\int \lambda}$. Notice that $\frac{1}{\int \lambda} = \Lambda'(1)$ is the average right degree. Finally, integrate by part $\int_{\mathtt{x}_a}^{\mathtt{x}_b} h^{\text{EBP}}(\epsilon(\mathtt{x}))\epsilon'(\mathtt{x})\mathrm{d}\mathtt{x}$ and use the first equation to get the second equation. □

The previous lemma allows us to compute the total area under the BP EXIT function.

**Lemma 4.10** [Area under BP EXIT Function] Consider a dd pair $(\lambda, \rho)$, the area under the associated BP EXIT curve is

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

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.* A straightforward computation gives

$$\begin{aligned}
&\int_0^1 h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon = \int_0^{\epsilon^{\text{BP}}} h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon + \sum_{i=1}^{J} \int_{\epsilon^i}^{\epsilon^{i+1}} h^{\text{BP}}(\epsilon)\mathrm{d}\epsilon \\
&\overset{(a)}{=} 0 + \frac{1}{\int \lambda} \sum_{i=1}^{J} \left( \left[ \epsilon(\mathtt{x}) \int_0^{h(\mathtt{x})} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} \right]_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} - \left[ \mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} + \int_{\underline{\mathtt{x}}^i}^{\overline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right) \\
&= \frac{\left( \int_0^1 \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \sum_{i=1}^{J} \left[ \epsilon(\mathtt{x}) \int_0^{h(\mathtt{x})} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \right) - \left( 1 - \sum_{i=1}^{J} \left[ \mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \right) + \left( \int_0^1 \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} - \sum_{i=1}^{J} \int_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right)}{\int \lambda} \\
&\overset{(b)}{=} \frac{\int \lambda - 1 + \int \mathtt{y}}{\lambda} + \frac{1}{\int \lambda} \sum_{i=1}^{J} \left( \left[ \mathtt{x}\mathtt{y}(\mathtt{x}) \right]_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} - \epsilon^i \int_{\mathtt{y}(\overline{\mathtt{x}}^{i-1})}^{\mathtt{y}(\underline{\mathtt{x}}^i)} \lambda(\mathtt{y})\mathrm{d}\mathtt{y} - \int_{\overline{\mathtt{x}}^{i-1}}^{\underline{\mathtt{x}}^i} \mathtt{y}(\mathtt{x})\mathrm{d}\mathtt{x} \right) \qquad (4.8)
\end{aligned}$$

where $(a)$ comes from Lemma 4.9 and $(b)$ uses the fact that $\epsilon^i = \epsilon(\overline{\mathtt{x}}^{i-1}) = \epsilon(\underline{\mathtt{x}}^i)$. □

Discussion: First observe that Lemma 4.10 quantifies the average sub-optimality of BP decoding compared to MAP decoding (if we assume that the asymptotic average rate is the design rate). The area under the BP EXIT function is trivially larger than or equal to the design rate because the $D_i$s are non-negative. This seems to indicate that performance loss occurs at each BP phase transition.
Second, Lemma 4.10 has a pleasing geometric interpretation that goes back to the asymptotic analysis using EXIT charts (or density evolution). This has been discussed in the previous chapter.

## 4.C Technical Lemmas for Counting Argument

We collect here a few technical tools that characterize the dd pair of the residual graph. They show that there is no discontinuous behavior implied by the randomness of the residual graph.

**Lemma 4.11** Consider a dd pair $\Xi$ and transmission over BEC($\epsilon$) such that $\epsilon \notin \{\epsilon_j : j \in [J]\}$. Let $\mathtt{G}(\epsilon)$ denote the residual graph obtained after BP decoding and let $\Xi_{\mathtt{G}(\epsilon)}$ denote its dd pair. Let $\Xi_\epsilon$ denote the typical dd pair. Then, for any $\xi > 0$, $\lim_{n\to\infty} \Pr\{d(\Xi_{\mathtt{G}(\epsilon)}, \Xi_\epsilon) \geq \xi\} = 0$ where $d$ denotes the $L_1$ distance, i.e., $\forall \Xi^a = (\Lambda^a, \Gamma^a), \forall \Xi^b = (\Lambda^b, \Gamma^b)$, we have $d(\Xi_a, \Xi_b) = \sum_{\mathtt{l}} |\Lambda_{\mathtt{l}}^a - \Lambda_{\mathtt{l}}^b| + \sum_{\mathtt{r}} |\Gamma_{\mathtt{r}}^a - \Gamma_{\mathtt{r}}^b|$.

*Proof.* Let $\mathtt{G}(\epsilon, \ell)$ denote the residual graph after $\ell$ iterations of the BP decoder, and let $\Xi_{\mathtt{G}(\epsilon,\ell)}$ be the associated dd pair. Moreover let $\Xi_{\epsilon,\ell}$ be the typical degree distribution pair of $\mathtt{G}(\epsilon, \ell)$. An explicit expression for this typical dd pair is easily obtained from Section 2.10 if we take $\delta = \mathtt{x}_\ell$ where $\mathtt{x}_\ell$ is an intermediate fraction of left-to-right erased messages obtained from density evolution ($\mathtt{y}_\ell$ is the fraction of right-to-left messages).

From the triangle inequality, we get

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

We claim that

$$\text{(i)} \ \lim_{\ell\to\infty} d(\Xi_{\mathtt{G}(\epsilon,\ell)}, \Xi_{\mathtt{G}(\epsilon)}) = 0 \quad \text{(ii)} \ \lim_{n\to\infty} \mathbb{E}[d(\Xi_{\epsilon,\ell}, \Xi_{\mathtt{G}(\epsilon,\ell)})] = 0 \quad \text{(iii)} \ \lim_{\ell\to\infty}\lim_{n\to\infty} \mathbb{E}[d(\Xi_\epsilon, \Xi_{\epsilon,\ell})] = 0$$

This will imply the thesis via Markov inequality. Since $\lim_{\ell\to\infty}\lim_{n\to\infty}\mathbb{E}\, d(\Xi_\epsilon, \Xi_{\mathtt{G}(\epsilon)}) = 0$ and $d(\Xi_\epsilon, \Xi_{\mathtt{G}(\epsilon)})$ does not depend upon $\ell$, we get

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

It remains to prove the three inequalities. (i) is a trivial consequence of the convergence to the fixed point of density evolution. $\lim_{\ell\to\infty} \mathtt{x}_\ell = \mathtt{x}$, $\lim_{\ell\to\infty} \mathtt{y}_\ell = \mathtt{y}$, together with the continuity of the dd pair in $\mathtt{x}, \mathtt{y}$. (ii) follows from the general concentration analysis in [14].
In order to prove (iii), consider the $i^{\text{th}}$ variable node of the residual graph. Assume we change its received value, and update all the messages consequently. Consider the edges whose distance from variable node $i$ is larger than $\ell$, and denote by $W_i^{(\ell)}$ the number of messages on such edges that change value after that the $i^{\text{th}}$ received symbol has been changed. It is clear that $\mathbb{E}[d(\Xi_\epsilon, \Xi_{\epsilon,\ell})] \le \mathbb{E}[W_i^{(\ell)}]$. The limit $\lim_{n\to\infty}\mathbb{E}[W_i^{(\ell)}]$ can be computed through a branching process analysis. The calculation is similar to the one in [147] and we do not reproduce it here. The 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 $\mathbb{E}[W_i^{(\ell)}] \le A b^\ell$. We conclude the proof by noticing that the condition $\epsilon\lambda'(\mathtt{y})\rho'(1-\mathtt{x}) < 1$ is satisfied whenever $\epsilon$ is larger than $\epsilon^{\text{BP}}$ and not equal to a discontinuity point $\epsilon_j$ for $j \in [J]$. □

**Lemma 4.12** Consider the function $\Theta_\Xi(u)$ defined in Lemma 2.3. There exists $A > 0$ such that for any two dd pairs $\Xi$ and $\tilde{\Xi}$ we have $\forall u \in [0,1]$, $|\Theta_\Xi(u) - \Theta_{\tilde{\Xi}}(u)| \le A d(\Xi, \tilde{\Xi})(1-u)^2$.

*Proof.* Let us write $\Theta_\Xi(u) = \Theta_\Xi^{(1)}(u) + \Theta_\Xi^{(2)}(u) + \Theta_\Xi^{(3)}(u)$ where the $\Theta_\Xi^{(j)}$'s for $j = 1,2,3$ are the three terms appearing in $\Theta_\Xi(u)$. The claim can be proved for each of the three terms separately. We will restrict ourselves to $\Theta_\Xi^{(1)}(u)$. The derivation is almost identical for the two other terms. Start by noticing that for any $u \in [0,1]$ and any dd pair we have $\frac{1}{2} \le \sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}}}{1+u^{\mathtt{l}}} \le 1$, and $\sum_{\mathtt{l}} \frac{\lambda_{\mathtt{l}} u^{\mathtt{l}-1}}{1+u^{\mathtt{l}}} \le 1$. Now fix two dd pairs $\Xi$ and $\tilde{\Xi}$. Let $v(u)$ and $\tilde{v}(u)$ be the corresponding functions. We get

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

Using these inequalities, some calculus shows that

$$1 \ge v(u), \tilde{v}(u) \ge 1 - 2\,\mathtt{l}_{\max}(1-u), \qquad |v(u) - \tilde{v}(u)| \le 3\,\mathtt{l}_{\max}^2\,(1-u)\, d(\Xi, \tilde{\Xi}) .$$

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

$$|f(u,v)| \le \frac{(1-u)(1-v)}{\log 2}, \qquad |f(u,v) - f(u,\tilde{v})| \le \frac{(1-u)}{\log 2}|v - \tilde{v}| .$$

Using these observations we finally obtain

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

where $A_1 = (2\mathtt{l}_{\max}^2 + 3\mathtt{l}_{\max}^3)/\log 2$. This concludes the proof for $\Theta_\Xi^{(1)}(u)$. The variations of $\Theta_\Xi^{(2)}$ and $\Theta_\Xi^{(3)}$ are bounded analogously. □

Discussion: From Chapter 2 we know that the function $\Theta_\Xi$ defined in Lemma 2.3 takes its maximas on $[0,1]$. The previous lemma is therefore sufficient to describe the regularity of $\Theta_\Xi$ over $[0,\infty)$.

# 4.D Maxwell Decoder: Tree and Elementary Consequences

The analysis of a Maxwell decoder is simplified if the graph is a tree or a forest. Recall from Section 2.5 that in this case the standard message-passing (BP) decoder (or its peeling version) performs MAP decoding. Some elementary results are therefore stated when the graph is a tree. This is very instructive and leads to a key result, given in Lemma 4.15.

**Lemma 4.13** [Sequential M Decoder and Number of Guesses] Consider a homogeneous[16] system of binary linear equations with $k$ 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 with this system is a tree. Then the sequential M decoder performs exactly $k$ guesses during the decoding process and all these guesses are independent.

*Proof.* Since the Tanner graph is a tree, the system itself (represented by a $m \times n$ matrix $H$) has full rank $m = n - k \geq 0$. Moreover there exists a submatrix that is the identity matrix $I_m$. It is therefore straightforward (e.g., by induction and Gaussian elimination) to see that we need to fix exactly $n - m = k$ variables to solve the system. This is done sequentially (hence a proof by induction) by the M decoder. We provide a more descriptive proof in [52]. □

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 4.14** [Number of Independent Guesses] Consider a homogeneous system of binary linear equations with $k$ 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 with this system is a tree and that it contains no check nodes of degree one. Then the number of *independent* guesses performed 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, let $\mathtt{l}_i^{\mathtt{g}}$ denote the number of incoming guessed (g) messages at variable node $i$ (including, if applicable, the guess of the bit itself), and let $\mathcal{C}_{\mathtt{g}}$ be the subset of all check nodes whose incoming messages are all guessed (g). Then

$$k = \mathbb{G} - \sum_{i \in \mathcal{V}} (\mathtt{l}_i^{\mathtt{g}} - 1) + \sum_{i \in \mathcal{C}_{\mathtt{g}}} (\mathtt{r}_i - 1). \tag{4.9}$$

*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 performed 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 perform the $k$ independent guesses and then apply the BP decoder. At the end of this phase all bits are known. Further, from Lemma 4.13 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, e.g., guess 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 $\mathtt{e}$ 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

[16] Recall that without loss of generality we can make the all-zero codeword assumption for our analysis. Therefore we consider a linear system with the right side equal to zero

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 one hand, the outgoing message along edge $\mathtt{e}$ flips to $\mathtt{g}$, 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 $l \in \mathcal{V}$ that are connected to check node $j$ but for node $i$ (the corresponding message was already a $\mathtt{g}$). In total this decreases the right-hand side of the counting formula by $\mathtt{r}_j - 1$. □

Each part of (the counting) Eq. (4.9) has a pleasing interpretation. As stated, $\mathbb{G}$ is the total number of performed guesses. If a variable node has $\mathtt{l}^{\mathtt{g}}$ incoming $\mathtt{g}$ 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 that the $\mathbb{G}$ guesses have to fulfill. But not all these conditions are linearly independent.

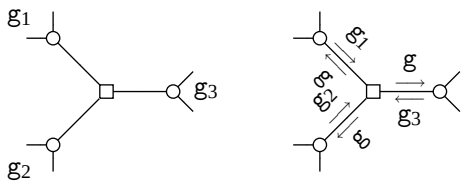



Figure 4.16: Computation of the number of linearly independent conditions. Each of the incoming edges corresponds to a list. To keep things simple and without loss of generality, assume that $I_i = \{i\}$. The three outgoing lists are then $I_1 = \{2,3\}$, $I_2 = \{1,3\}$, and $I_3 = \{1,2\}$. Compare the incoming and outgoing lists at the first node: we get the condition $x_1 = x_2 + x_3$. But exactly the same condition appears at the second and third nodes. In general, a check node of degree $\mathtt{r}$ whose incoming messages are all $g$, generates $\mathtt{r} - 1$ linearly dependent conditions.

Consider Figure 4.16. If a check node of degree $\mathtt{r}$ has all of its incoming messages equal to $\mathtt{g}$ then the $\mathtt{r}$ equations that correspond to the $\mathtt{r}$ outgoing messages are identical, i.e., $\mathtt{r} - 1$ of them are linearly dependent. The last term in Eq. (4.9) therefore corrects the over-counting of dependent conditions.

**Example 4.11** Consider a code whose Tanner graph is a tree and whose leaves are all 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 $\mathtt{g}$. Let us determine the number of independent guesses at the end of the decoding process using the counting formula (4.10). Note that for each leaf node we have $\mathtt{l}^{\mathtt{g}} = 2$ (one guess and one additional incoming $\mathtt{g}$ 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 of leaf nodes, so that $\mathbb{G} = n_l$, we obtain that the number of independent guesses equals

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

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 4.15** [Lower Bound on Independent Guesses] Consider a homogeneous system of binary linear equations with $k$ 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 with this system contains no check nodes of degree one. Let $\mathbb{G}$ denote the total number of guesses performed by the M decoder, let $\mathtt{l}_i^{\mathtt{g}}$ denote the number of incoming g messages at variable node $i$ (including the guess if this node has been guessed), and let $\mathcal{C}_{\mathtt{g}}$ be the subset of all check nodes all of whose incoming messages are g. Then

$$k \geq \mathbb{G} - \sum_{i \in \mathcal{V}} (\mathtt{l}_i^{\mathtt{g}} - 1) + \sum_{i \in \mathcal{C}_{\mathtt{g}}} (\mathtt{r}_i - 1). \tag{4.10}$$

**Overview:** A new tool called the GEXIT function is developed. EXIT functions on the BEC become a particular instance of GEXIT functions and the general area theorem extends to BMS channels.

# 5 | GEXIT Functions

In the previous two chapters we have seen that EXIT functions are powerful tools for analyzing iterative decoding over the BEC. However, for more general channels, their theoretical interest is restricted. This is mainly due to the fact that they do not fulfill the area theorem. An easy way to see this is to consider the extremality properties presented in Theorem 3.1. For example, for a $[n+1,n]$ single parity-check code over BSC($\epsilon$), the area under the EXIT function is $A_n \triangleq \int_0^1 h^{\text{MAP}}(\mathtt{h})\mathrm{d}\mathtt{h} = \int_0^{1/2} h_2(\frac{1-(1-2\epsilon)^n}{2})\mathrm{d}\epsilon$ as discussed in Example 3.1. This gives $A_2 \approx 0.643704 < 2/3$ for $n = 2$. *Generalized* EXIT (GEXIT) functions are an extension of the EXIT concept to general BMS channels: GEXIT functions satisfy the area theorem by definition and share most of the basic properties with EXIT functions.

## 5.1 Definition and Linear Functional

The concept of GEXIT functions extends to non-binary channels in a natural way as shown in Appendix 5.B. Nevertheless, in order to bring out the main message of this thesis in a simple way, we focus here on the binary case.

Before defining a measure that fulfills the general area theorem by assumption, let us ask the question: What property of EXIT functions makes them fulfill the area theorem on the erasure channel? We have seen that the answer follows trivially from characterization (iv) of Lemma 3.4, which states that the EXIT function over the BEC coincides with the derivative of the conditional entropy $H(X|Y)$. Let us therefore *define* the GEXIT function using this characterization. Of course, some technical hypotheses are required to ensure that the involved objects exist. They are, for example, implied by the *smoothness* of the channel as defined in Section 2.9.

**Definition 5.1** [(MAP) GEXIT Function] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over a BMS channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$, i.e., for any $i$ the $i^{\text{th}}$ bit is passed through a BMS channel parameterized by a single scalar that is the channel entropy $\mathtt{h}_i \triangleq H(X_i|Y_i) \in P_i \subseteq \mathbb{R}$. Let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Consider $i \in [n]$. Under the hypothesis that the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i \in P_i}$ is smooth, define

$$g_i^{\text{MAP}}(\mathtt{h}) \triangleq \frac{\partial H(X_i|Y_i, \phi_i^{\text{MAP}}(Y_{\sim i}), \Omega)}{\partial \mathtt{h}_i},$$

where $\phi_i^{\text{MAP}}$ is the (extrinsic) MAP estimator defined in Chapter 2. The function $g_i^{\text{MAP}}$ is the $i^{\text{th}}$ GEXIT

function. If for all $i \in [n]$ the family $\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i}$ is smooth, then $g^{\text{MAP}} \triangleq \frac{1}{n}\sum_i g_i^{\text{MAP}}$ is the (uniformly averaged) GEXIT function, and $\underline{g}^{\text{MAP}} \triangleq (g_1^{\text{MAP}}, \cdots, g_n^{\text{MAP}})$ is the GEXIT vector.

Discussion: Note first that, without loss of generality, we have chosen to parameterize with respect to the channel entropy $\mathtt{h}_i = H(X_i|Y_i)$, but various alternative parameterizations are possible. Also, recall from Chapter 2 (in particular Example 2.10) and Chapter 3 (Appendix 3.A) that $H(X_i|Y_i, \phi_i^{\text{MAP}}(Y_{\sim i}), \Omega) = H(X_i|Y, \Omega)$.

**Theorem 5.1** [General Area Theorem – BMSC] Let $X$ be a binary random vector of length $n$ and assume that transmission takes place over a smooth family $\{\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i \in P_i}\}_i$, i.e., for any $i$ the $i^{\text{th}}$ bit is passed through a (smooth family of) BMS channel(s) parameterized by $\mathtt{h}_i \in P_i \subseteq \mathbb{R}$. Let $Y$ be the received vector and let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Then $\underline{g}^{\text{MAP}} \triangleq (g_1^{\text{MAP}}, \cdots, g_n^{\text{MAP}}) = \nabla H(X|Y,\Omega) \triangleq (\frac{\partial H(X|Y,\Omega)}{\partial \mathtt{h}_1}, \cdots, \frac{\partial H(X|Y,\Omega)}{\partial \mathtt{h}_n})$. Furthermore, if there exists a real-valued parameter $\mathfrak{p}$ such that the vector $\mathtt{h}(\mathfrak{p}) = (\mathtt{h}_1(\mathfrak{p}), \cdots, \mathtt{h}_n(\mathfrak{p}))$ is differentiable with respect to $\mathfrak{p}$, then $\underline{g}^{\text{MAP}} \cdot \frac{\mathrm{d}\mathtt{h}(\mathfrak{p})}{\mathrm{d}\mathfrak{p}} = \nabla H(X|Y) \cdot \mathtt{h}'(\mathfrak{p}) = \frac{\mathrm{d}H(X|Y(\mathfrak{p}))}{\mathrm{d}\mathfrak{p}}$ where "$\cdot$" denotes the standard scalar product. In particular, if a parameter $\mathfrak{p}$ can be chosen such that $\mathtt{h}_i(\mathfrak{p}) = \mathfrak{p}$ for all $i$, then $g^{\text{MAP}}(\mathfrak{p}) = \frac{1}{n}\sum_{i=1}^n g_i^{\text{MAP}}(\mathtt{h}_i) = \frac{\mathrm{d}H(X|Y,\Omega)}{n\mathrm{d}\mathfrak{p}}$ where $g(\mathfrak{p})$ is the average GEXIT function over $\text{BMSC}(\mathfrak{p})$.

*Proof.* The chain rule for entropy reads $H(X|Y,\Omega) = H(X_i|Y_i, Y_{\sim i}, \Omega) + H(X_{\sim i}|Y, X_i, \Omega)$ which gives $H(X|Y,\Omega) = H(X_i|Y_i, Y_{\sim i}, \Omega) + H(X_{\sim i}|Y_{\sim i}, X_i, \Omega)$ due to the memoryless nature of the channel and $\Omega \to X \to Y$. By taking the partial derivative with respect to $\mathtt{h}_i$, we get the result. □

As for EXIT functions, the definition of GEXIT function extends naturally to any extrinsic estimator $\phi_i^{\text{DEC}}(Y_{\sim i})$.

**Definition 5.2** [GEXIT Function] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over a BMS channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$. Let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Let $\Phi_i^{\text{DEC}} = \phi_i^{\text{DEC}}(Y_{\sim i})$ represent any estimator on $X_i$ based on $Y_{\sim i}$. Consider $i \in [n]$. Under the hypothesis that the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i \in P_i}$ is smooth, define

$$g_i^{\text{DEC}}(\mathtt{h}) \triangleq \frac{\partial H(X_i|Y_i, \phi_i^{\text{DEC}}(Y_{\sim i}), \Omega)}{\partial \mathtt{h}_i}.$$

The function $g_i^{\text{DEC}}$ is the $i^{\text{th}}$ GEXIT function associated with the extrinsic DEC estimator. If for all $i \in [n]$ the family $\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i}$ is smooth, then $g^{\text{DEC}} \triangleq \frac{1}{n}\sum_i g_i^{\text{DEC}}$ is the corresponding averaged GEXIT function. If the individual channel entropies $\mathtt{h}_i = H(X_i|Y_i)$ are all parameterized by a scalar $\mathfrak{p} \in P \subseteq \mathbb{R}$ such that $\mathtt{h}_i = \mathtt{h}_i(\mathfrak{p})$, then the functions become functions of a single scalar parameter.

The entropy operator (see Definition 2.5) is used to facilitate the evaluation of EXIT functions in Chapter 3. Assume that the BMS channel is represented by its $L-$density $\mathtt{a}$. The associated entropy is then obtained as $H(X|Y) = \mathtt{H}(\mathtt{a}) = \mathbb{E}_Y[l(Y)]$ where $y \mapsto l(y) = \log_2(1+\mathrm{e}^{-y})$ is called the *EXIT kernel* (here in the $L$-domain). As a consequence the entropy operator acts as an "EXIT operator." Lemma 3.2 shows that the $i^{\text{th}}$ EXIT function can be computed as $h_i^{\text{MAP}}(\mathtt{h}_{\sim i}) = \mathbb{E}_Y[l(Y)]$. A similar linear functional exists for GEXIT functions. Its associated *GEXIT kernel* is a channel-dependent function that measures the response of the environment to small noise perturbations. In other words, the GEXIT kernel reflects the dependency of the GEXIT measure on the intrinsic channel.

**Lemma 5.1** [MAP GEXIT: Operational Characterization] Let $X$ be chosen uniformly at random from a proper binary linear code of length $n$. Assume that transmission takes place over a BMS channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_i$ equivalently represented by the family of $L$-densities $\{\mathtt{c}\}_i$. Consider $i \in [n]$. Assume that the channel family $\{\text{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i \in P_i}$ is smooth and let $\mathtt{a}_i^{\text{MAP}}$ denote the $L-$density associated with $\Phi_i^{\text{MAP}}$. Then

$$g_i^{\text{MAP}}(\mathtt{h}_i, \mathtt{h}_{\sim i}) = \mathtt{G}(\mathtt{c}_i, \mathtt{a}_i^{\text{MAP}})$$

where $\mathtt{G}(\mathtt{c},\mathtt{a}) \triangleq \int \mathtt{a}(y)[\int \frac{\mathrm{d}\mathtt{c}_i(w)}{\mathrm{d}\mathtt{h}_i}\log_2(1+\mathrm{e}^{-w-y})\mathrm{d}w]\mathrm{d}y$ is called GEXIT operator. The function $y \mapsto l^{\mathtt{c}_i} \triangleq \int \frac{\mathrm{d}\mathtt{c}_i(w)}{\mathrm{d}\mathtt{h}_i}\log_2(1+\mathrm{e}^{-w-y})\mathrm{d}w$ is the GEXIT kernel associated with the channel $\mathtt{c}_i$.

*Proof.* The channel outputs values in the $L-$domain. Let $\mathtt{Y}_i$ denote the random LLR that the channel outputs, and let $\Phi_i$ be shorthand for the extrinsic MAP estimate. We have seen in Chapter 2 and Chapter 3 that the channel symmetry is preserved under addition of $L-$densities and under MAP decoding. Therefore

$$H(X_i|Y) = H(X_i|\mathtt{Y}_i, \Phi_i) = H(X_i|\mathtt{Y}_i + \Phi_i) = \mathtt{H}(\mathtt{a}_i \circledast \mathtt{c}_i),$$

where the convolution $\mathtt{a}_i \circledast \mathtt{c}_i$ gives the density of the estimate $\mathtt{Y}_i + \Phi_i$ since, by hypothesis, $Y_i$ and $\Phi_i$ are conditionally independent. Further

$$\mathtt{H}(\mathtt{a}_i \circledast \mathtt{c}_i) = \int_{\tilde{y}} (\mathtt{a}_i \circledast \mathtt{c}_i)(\tilde{y}) l(\tilde{y}) \mathrm{d}\tilde{y} = \int_{\tilde{y}} \int_w \mathtt{a}_i(\tilde{y} - w) \mathtt{c}_i(w) l(\tilde{y}) \mathrm{d}w \mathrm{d}\tilde{y} = \int_y \int_w \mathtt{a}_i(y) \mathtt{c}_i(w) l(y + w) \mathrm{d}w \mathrm{d}y.$$

Since the extrinsic estimate does not depend upon $\mathtt{h}_i$, we get

$$\frac{\partial H(X_i|Y_i, \Phi_i)}{\partial \mathtt{h}_i} = \frac{\partial}{\partial \mathtt{h}_i} \int_y \mathtt{a}_i(y) \int_w \mathtt{c}_i(w) l(y + w) \mathrm{d}w \mathrm{d}y = \int_y \mathtt{a}_i(y) \int_w \frac{\mathrm{d}}{\mathrm{d}\mathtt{h}_i}(\mathtt{c}_i)(w) l(y + w) \mathrm{d}w \mathrm{d}y$$

which concludes the proof since $l(y + w) = \log_2(1 + \mathrm{e}^{-y-w})$. □

If $\Phi_i$ denotes any symmetric estimator $\Phi_i^{\mathrm{DEC}}$, then the proof of Lemma 5.1 applies in a more general context.

**Lemma 5.2** [GEXIT: Operational Characterization] Let $X$ be chosen uniformly at random from a proper binary linear code of length $n$. Assume that transmission takes place over a BMS channel family $\{\mathrm{BMSC}_i(\mathtt{h}_i)\}_i$ equivalently represented by the family of $L$-densities $\{\mathtt{c}\}_i$. Consider an additional observation $\Omega$ such that $\Omega \to X \to Y$. Consider $i \in [n]$. Consider any estimator $\Phi_i^{\mathrm{DEC}} = \phi_i^{\mathrm{DEC}}(Y_{\sim i}, \Omega)$ that preserves channel symmetry. Let the density of $\Phi_i^{\mathrm{DEC}}$ under the assumption that the all-one codeword was transmitted be $\mathtt{a}_i^{\mathrm{DEC}}$. Assume that the channel family $\{\mathrm{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i \in P_i}$ is smooth. Then

$$g_i^{\mathrm{DEC}}(\mathtt{h}_i, \mathtt{h}_{\sim i}) = \mathtt{G}(\mathtt{c}_i, \mathtt{a}_i^{\mathrm{DEC}})$$

where $\mathtt{G}(\mathtt{c}, \mathtt{a}) \triangleq \int \mathtt{a}(y) [\int \frac{\mathrm{d}\mathtt{c}_i(w)}{\mathrm{d}\mathtt{h}_i} \log_2(1 + \mathrm{e}^{-w-y}) \mathrm{d}w] \mathrm{d}y$ is the GEXIT operator (and $l^{\mathtt{c}_i} \triangleq \int \frac{\mathrm{d}\mathtt{c}_i(w)}{\mathrm{d}\mathtt{h}_i} \log_2(1 + \mathrm{e}^{-w-y}) \mathrm{d}w$ the GEXIT kernel associated $\mathtt{c}_i$).

Discussion: The following remark also applies to the kernel defined for EXIT functions. Consider a generic kernel $l(z)$ (for example an EXIT or a GEXIT kernel). Because of the symmetry property of $L$-densities, for any such $l(z)$, we can write $\int_{-\infty}^{\infty} \mathtt{a}(z) l(z) \, \mathrm{d}z = \int_0^{\infty} \mathtt{a}(z)(l(z) + \mathrm{e}^{-z} l(-z)) \mathrm{d}z = \int_0^{\infty} \mathtt{a}^{|L|}(z) \frac{l(z) + \mathrm{e}^{-z} l(-z)}{1 + \mathrm{e}^{-z}} \mathrm{d}z$. This means that an expression for the kernel is uniquely specified on the absolute value domain $[0, \infty]$ (or $|L|$-domain, see Appendix 2.B), but that for each $z \in [0, \infty]$ we can split the weight of a (kernel) function $l(z)$ in any desired way between $+z$ and $-z$ so that $l(z) + \mathrm{e}^{-z} l(-z)$ equals the desired fixed value. In the remainder of this section, we will use this degree of freedom to bring some kernels into a more convenient form and we will sometimes omit to mention that they are *equivalent* representations. Let us therefore define formally this equivalency relationship.

**Definition 5.3** [Equivalent Kernel] Consider two functions $l_1(y)$ and $l_2(y)$ over $\mathbb{R}$. If $\forall y \in [0, \infty)$, $l_1(y) + \mathrm{e}^{-y} l_1(-y) = l_2(y) + \mathrm{e}^{-y} l_2(-y)$, then $l_1(y)$ and $l_2(y)$ are said to be equivalent kernels.

To be more concrete, let us present some examples of GEXIT kernels and GEXIT curves. Consider a smooth family $\{\{\mathrm{BMSC}_i(\mathfrak{p})\}_{\mathfrak{p} \in P|}\}_i$, i.e., a family of smooth (family of) channels parameterized by a common $\mathfrak{p} \in P$. As we have already remarked, the GEXIT functions $g_i(\mathfrak{p})$ allow us to "locally" measure the change of the conditional entropy of a system. This property is the essence of GEXIT functions. For example, it is apparent in the representation of Lemma 5.1 where we see that the local measurement has two components: (i) the kernel that depends on the derivative of the channel seen at the given position and (ii) the distribution $\mathtt{a}_i$, which encapsulates all our ignorance about the code behavior with respect to the $i^{\mathrm{th}}$ position. This representation is very intuitive. If we improve the observation of a particular bit (derivative of the channel with respect to the parameter), then the amount by which the conditional entropy of the overall system changes clearly depends on how well

this particular bit was already known via the code constraints and the observations of the other bits (extrinsic posterior density). For example, if the bit was already perfectly known, then the additional extrinsic observation afforded will be useless, whereas if nothing was known about the bit, one would expect that the additional reduction in entropy of this bit fully translates into a reduction of the entropy of the overall system. In the next three examples we compute the kernels $I^{\mathsf{c}_{\mathrm{BMSC}(\mathtt{h})}}(z)$, where the family of $L$-density $\{\mathsf{c}_{\mathrm{BMSC}(\mathtt{h})}\}_{\mathtt{h}}$ represents the channel families $\{\mathrm{BEC}(\mathtt{h})\}_{\mathtt{h}}$, $\{\mathrm{BSC}(\mathtt{h})\}_{\mathtt{h}}$, or $\{\mathrm{BAWGNC}(\mathtt{h})\}_{\mathtt{h}}$. We made the choice to parameterize the channel family by the entropy $\mathtt{h}$ in order to measure the "progress per d$\mathtt{h}$", and in the sequel to measure "exchanges of entropy". If we consider an alternative parameterization $\mathtt{p}$ such that $\mathtt{h}=\mathtt{h}(\mathtt{p})$, then the GEXIT kernel is simply obtained via the normalization[1]

$$I^{\mathsf{c}_{\mathrm{BMSC}(\mathtt{h})}}(z)=\frac{\int_{-\infty}^{\infty}\frac{\mathrm{d}\mathsf{c}_{\mathrm{BMSC}(\mathtt{h}(\mathtt{p}))}(w)}{\mathrm{d}\mathtt{p}}\log_2(1+\mathrm{e}^{-z-w})\,\mathrm{d}w}{\int_{-\infty}^{\infty}\frac{\mathrm{d}\mathsf{c}_{\mathrm{BMSC}(\mathtt{h}(\mathtt{p}))}(w)}{\mathrm{d}\mathtt{p}}\log_2(1+\mathrm{e}^{-w})\,\mathrm{d}w}. \tag{5.1}$$

**Example 5.1** [GEXIT Kernel, $L$-Domain – $\{\mathrm{BEC}(\mathtt{h})\}_{\mathtt{h}}$] Consider the family $\{\mathsf{c}_{\mathrm{BEC}(\mathtt{h})}\}_{\mathtt{h}}$ where the parameter $\mathtt{h}$ denotes both, the channel (intrinsic) entropy, i.e., $\mathtt{h}(\mathtt{p})=\mathtt{p}$, and the cross-over erasure probability, i.e., $\epsilon=\mathtt{p}$. A quick calculation shows that $I^{\mathsf{c}_{\mathrm{BEC}(\mathtt{h})}}(z)=\log_2(1+\mathrm{e}^{-z})=l(z)$. In other words, the GEXIT kernel associated with the family $\{\mathrm{BEC}(\mathtt{h})\}_{\mathtt{h}}$ is the standard EXIT kernel.

**Example 5.2** [GEXIT Kernel, $L$-Domain – $\{\mathrm{BSC}(\mathtt{h})\}_{\mathtt{h}}$] Consider the family $\{\mathsf{c}_{\mathrm{BSC}(\mathtt{h})}\}_{\mathtt{h}}$ parameterized by the channel entropy $\mathtt{h}$. Some calculus reveals that

$$I^{\mathsf{c}_{\mathrm{BSC}(\mathtt{h})}}(z)=\log\left(\frac{1+\frac{1-\epsilon}{\epsilon}\mathrm{e}^{-z}}{1+\frac{\epsilon}{1-\epsilon}\mathrm{e}^{-z}}\right)/\log\left(\frac{1-\epsilon}{\epsilon}\right),$$

where $\epsilon=h_2{}^{-1}(\mathtt{h})$. For a fixed $z\in\mathbb{R}$ and $\mathtt{h}\to 0$, the kernel converges to 1 as $1+z/\log(\epsilon)$, whereas the limit when $\mathtt{h}\to 1$ is equal to $\frac{2}{1+\mathrm{e}^{z}}$.

**Example 5.3** [GEXIT Kernel, $L$-Domain – $\{\mathrm{BAWGNC}(\mathtt{h})\}_{\mathtt{h}}$] Consider the family $\{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h})}\}_{\mathtt{h}}$ parameterized by the channel entropy $\mathtt{h}$. If the noise has variance $\sigma^2$, then a convenient parameterization is $\mathtt{p}\triangleq 2/\sigma^2$. This means that $\mathtt{h}=\mathtt{H}(\mathsf{c}_{\mathrm{BAWGNC}(\sigma^2=2/\mathtt{p})})$. After some steps of calculus shown in Appendix 5.C and Lemma 5.8, we get

$$I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h})}}(z)=\left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w-\mathtt{p})^2}{4\mathtt{p}}}}{1+\mathrm{e}^{w+z}}\mathrm{d}w\right)/\left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w-\mathtt{p})^2}{4\mathtt{p}}}}{1+\mathrm{e}^{w}}\mathrm{d}w\right).$$

In Appendix 5.C we also give alternative representations and/or interpretations of this kernel. In particular, we discuss the relationship with the formulation presented in [40, 148] using a connection to the MMSE detector, as well as the formulation in [41] based on the Nishimori identity.

One convenient feature of standard EXIT functions is that they are fairly similar for a given code across the whole range of BMS channels. Is this still true for GEXIT functions? The extrinsic densities are the same as for the computation of EXIT functions. But now, the kernels are also functions of the channel. Let us therefore compare the shape of the various kernels. As indicated in Definition 5.3 it is most convenient to compare the kernels not in the $L$-domain but rather in a domain where the kernel is uniquely defined, e.g., the $|D|$-domain of Appendix 2.B. A change of variables shows that in general the $L$-domain kernel $l^{\mathsf{c}}(\cdot)$ and the associated $|D|$-domain kernel, denote it by $|d|^{\mathsf{c}}(\cdot)$, are linked by

$$|d|^{\mathsf{c}}(s)=\frac{1-s}{2}l^{\mathsf{c}}(\log\frac{1-s}{1+s})+\frac{1+s}{2}l^{\mathsf{c}}(\log\frac{1+s}{1-s}). \tag{5.2}$$

**Example 5.4** [GEXIT Kernel, $|D|$-Domain – $\{\mathrm{BEC}(\mathtt{h})\}_{\mathtt{h}}$] We get $|d|^{\mathsf{c}_{\mathrm{BEC}(\mathtt{h})}}(s)=h_2((1+s)/2)$.

[1] To see this formula, refer to the proof of Lemma 5.2 and use $\frac{\partial H}{\partial \mathtt{h}}=\frac{\partial H}{\partial \mathtt{p}}/\frac{\partial \mathtt{h}}{\partial \mathtt{p}}$.

**Example 5.5** [GEXIT Kernel, $|D|$-Domain – $\{\mathrm{BSC}(\mathtt{h})\}_{\mathtt{h}}$] Some calculus shows that $|d|^{\mathsf{c}_{\mathrm{BSC}(\mathtt{h}(\epsilon))}}(s) = 1+\frac{s}{\log((1-\epsilon)/\epsilon)}\log\left(\frac{1+2\epsilon s-s}{1-2\epsilon s+s}\right)$. The limiting values are $\lim_{\mathtt{h}\to 1}|d|^{\mathsf{c}_{\mathrm{BSC}(\mathtt{h})}}(s) = 1-s^2$, and $\lim_{\mathtt{h}\to 0}|d|^{\mathsf{c}_{\mathrm{BSC}(\mathtt{h})}}(s) = 1$.

**Example 5.6** [GEXIT Kernel, $|D|$-Domain – $\{\mathrm{BAWGN}(\mathtt{h})\}_{\mathtt{h}}$] With Example 5.3 we get

$$|d|^{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h}(\epsilon))}}(s) = \sum_{i\in\{-1,+1\}}\left(\int_{-\infty}^{+\infty}\frac{(1-s^2)\mathrm{e}^{-\frac{(w-\mathtt{p})^2}{4\mathtt{p}}}}{(1+is)+(1-is)\mathrm{e}^{w}}\mathrm{d}w\right)\Big/\left(\int_{-\infty}^{+\infty}\frac{2\mathrm{e}^{-\frac{(w-\mathtt{p})^2}{4\mathtt{p}}}}{1+\mathrm{e}^{w}}\mathrm{d}w\right).$$

As shown in Appendix 5.C, the limiting values are the same as for the BSC, i.e., $\lim_{\mathtt{h}\to 1}|d|^{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h})}}(s) = 1-s^2$, and $\lim_{\mathtt{h}\to 0}|d|^{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h})}}(s) = 1$.

In Figure 5.1 we compare the EXIT kernel (which is also the GEXIT kernel for the BEC) with the GEXIT kernels for BSC($\mathtt{h}$) and BAWGNC($\mathtt{h}$) in the $|D|$-domain for several channel parameters. These kernels are distinct but quite similar. In particular, for $\mathtt{h} = 0.5$ the GEXIT kernel with respect to BAWGNC($\mathtt{h}$) is hardly distinguishable from the regular EXIT kernel. The GEXIT kernel for the BSC shows more variation.

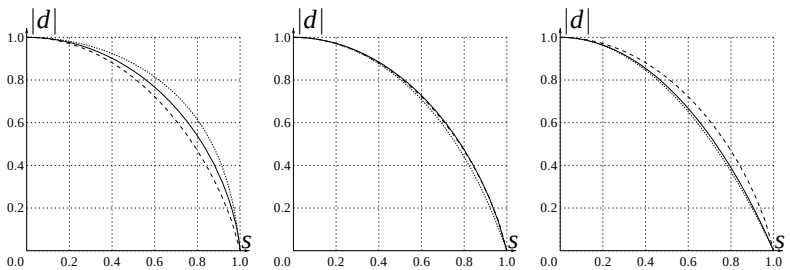



Figure 5.1: Comparison of the kernels $|d|^{\mathsf{c}_{\mathrm{BEC}(\mathtt{h})}}(s)$ (dashed line) with $|d|^{\mathsf{c}_{\mathrm{BSC}(\mathtt{h})}}(s)$ (dotted line) and $|d|^{\mathsf{c}_{\mathrm{BAWGNC}(\mathtt{h})}}(s)$ (solid line) at channel entropy rate $\mathtt{h} = 0.1$ (left), $\mathtt{h} = 0.5$ (middle) and $\mathtt{h} = 0.9$ (right).

Let us now give a few examples of GEXIT curves. Recall that the considered codes are isotropic, see Chapter 3. Therefore $g^{\mathrm{MAP}} = g_i^{\mathrm{MAP}}$ for all $i \in [n]$.

**Example 5.7** [Repetition Code] Consider the $[n,1,n]$ repetition code. Let $\{\mathsf{c}_{\mathtt{h}}\}_{\mathtt{h}}$ characterize a smooth family of BMS channels. The GEXIT function for the $[n,1,n]$ repetition code is then given by $g^{\mathrm{MAP}}(\mathtt{h}) = \frac{\mathrm{d}}{\mathrm{d}\mathtt{h}}\mathtt{H}(\mathsf{c}_{\mathtt{h}}^{\circledast n})$. An explicit expression over BEC($\mathtt{h}$) is $g^{\mathrm{MAP}}(\mathtt{h}) = \mathtt{h}^n = h^{\mathrm{MAP}}(\mathtt{h})$ where $h^{\mathrm{MAP}}(\mathtt{h})$ is the EXIT function. As a further example over BSC($\mathtt{h}$), $g^{\mathrm{MAP}}(\mathtt{h})$ is given in parametric form by

$$\left(h_2(\epsilon), \frac{\sum_{j=\pm 1} j\sum_{i=1}^{n}\binom{n}{i}\epsilon^i\bar{\epsilon}^{n-i}\log\left(1+(\epsilon/\bar{\epsilon})^{n-2i-j}\right)}{n\log(\bar{\epsilon}/\epsilon)}\right),$$

where $\epsilon = {h_2}^{-1}(\mathtt{h})$ and $\bar{\epsilon} \triangleq 1-\epsilon$.

**Example 5.8** [Single Parity-Check Code] Consider the dual code of the previous example, i.e., the $[n,n-1,2]$ parity-check code. Some calculations show that over BSC($\mathtt{h}$) the GEXIT function $g^{\mathrm{MAP}}(\mathtt{h})$ is given in parametric form by

$$\left(h_2(\epsilon), 1-(1-2\epsilon)^{n-1}\frac{\log\left(\frac{1+(1-2\epsilon)^n}{1-(1-2\epsilon)^n}\right)}{\log\left(\frac{1-\epsilon}{\epsilon}\right)}\right).$$

No simple analytic expressions are known for the case of transmission over the BAWGNC.

Figure 5.2 compares EXIT to GEXIT curves for some repetition codes and their dual.

**Example 5.9** [Hamming Code] Consider the $[7,4,3]$ Hamming code. When transmission takes place over BEC($\epsilon$), a tedious but conceptually simple exercise shows that the EXIT function is $h^{\mathrm{MAP}}(\epsilon) =$

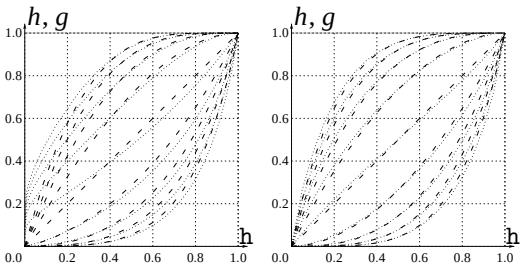



Figure 5.2: The EXIT (dashed line) and GEXIT (dotted line) function of the $[n,1,n]$ repetition code and the $[n,n-1,2]$ parity-check code, $n \in \{2,3,4,5,6\}$. Left: Transmission takes place over BSC($\mathtt{h}$). Right: Transmission takes place over BAWGNC($\mathtt{h}$).

$3\epsilon^2 + 4\epsilon^3 - 15\epsilon^4 + 12\epsilon^5 - 3\epsilon^6$, see Chapter 3. In a similar way, using the derivative of the conditional entropy, one can give an analytic expression for the GEXIT function assuming transmission takes place over the BSC. Both expressions are evaluated in Figure 5.3 (left). A comparison between GEXIT and EXIT functions for the Hamming code and the BSC is shown in Figure 5.3 (right).

**Example 5.10** [Simplex Code] Consider finally the dual of the Hamming code, i.e., the $[7,3,4]$ Simplex code. For transmission over BEC($\epsilon$), we have $h^{\text{MAP}}(\epsilon) = 4\epsilon^3 - 6\epsilon^5 + 3\epsilon^6$. Figure 5.3 compares GEXIT and EXIT functions for this code when transmission takes place over the BEC and over the BSC.

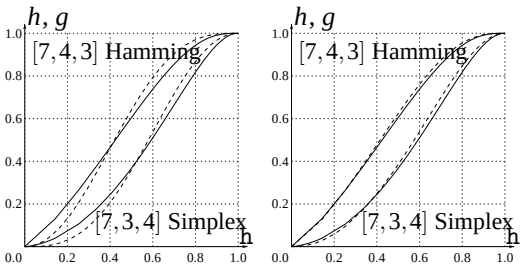



Figure 5.3: Comparison of the GEXIT functions for the $[7,4,3]$ Hamming code and its dual. Left: Comparison between GEXIT functions when transmitting over the BEC (dashed line) and over the BSC (solid line). Right: Comparison between GEXIT (solid line) and EXIT (dashed line) functions when transmission takes place over the BSC.

## 5.2 Further Properties of GEXIT Functions

We derive in this section a few further properties of GEXIT functions. We show that GEXIT functions share many characteristics with EXIT functions (except of course the extremality property since the area under the GEXIT curves is independent of the channel). One such fundamental property is the partial order imposed by physical degradation.

Let us first examine how the GEXIT measure is related to the extrinsic bit error probability. This will, in Chapter 6, justify the definition of the MAP threshold stated in Chapter 2. Assuming that the potential probability mass at zero of a channel $L$-density is equally distributed on both sides of zero, the error probability is obtained by integrating the negative part of this channel density. If the $L$-density is symmetric, we can further define the resulting error probability operator as follows, $\mathfrak{E}(\mathsf{a}) \triangleq \frac{1}{2}\int_{-\infty}^{\infty} \mathsf{a}(z) e^{-(|z/2|+z/2)} \mathrm{d}z$. This definition avoids dealing with a potential probability mass at zero.

**Lemma 5.3** [GEXIT Kernel and Bounds on GEXIT Functions] Consider a smooth family of BMS channels characterized by their family of $L$-densities $\{\mathsf{c}_{\text{BMSC}(\mathtt{h})}\}_{\mathtt{h}}$. Let $|d|^{\mathsf{c}_{\text{BMSC}(\mathtt{h})}}(z)$ be the associated GEXIT kernel in the $|D|-$domain. Then $|d|^{\mathsf{c}_{\text{BMSC}(\mathtt{h})}}(z) : [0,1] \to [0,1]$ is non-decreasing and concave. Moreover, $1 - z \leq |d|^{\mathsf{c}_{\text{BMSC}(\mathtt{h})}}(z) \leq 1$, therefore, if $\mathsf{a}$ is a symmetric $L$-density, then $2\,\mathfrak{E}(\mathsf{a}) \leq$

$\mathtt{G}(\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})},\mathtt{a}) = \int_{-\infty}^{\infty} I^{\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})}}(z)\mathtt{a}(z)\mathrm{d}z \leq 1\,.$

*Proof.* In Appendix 5.A, we show that $|d|^{\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})}}(z)$ is non-increasing and concave. The upper bound follows from $|d|^{\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})}}(z) < |d|^{\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})}}(z=0) = 1$. The lower bound is proved in a similar way by using concavity and observing that $|d|^{\mathtt{c}_{\mathrm{BMSC}(\mathtt{h})}}(z=1) = 0$. The final claim now follows from the fact that the $|D|$-domain kernel associated with $\mathfrak{E}$ is equal to $(1-z)/2$. □

Discussion: If $\mathtt{a}$ represents the extrinsic $L-$density, then $\mathfrak{E}(\mathtt{a})$ is the extrinsic error probability.
The next theorem shows that GEXIT functions preserve partial order implied by physical degradation. It is a powerful property when used in the next chapter to give an upper bound on the MAP threshold of iterative coding systems. Before stating this theorem let us derive an elementary lemma from (the data processing) Theorem 2.1.

**Lemma 5.4** [Second Order Data Processing] Let $X,Y,Y',\Phi,\Phi'$ be random vectors. If $X \to Y \to Y'$, $X \to \Phi \to \Phi'$, and $(Y,Y') \to X \to (\Phi,\Phi')$, then $H(X|Y',\Phi) - H(X|Y,\Phi) \leq H(X|Y',\Phi') - H(X|Y,\Phi')$. Alternatively, $I(X;Y|Y',\Phi) \leq I(X;Y|Y',\Phi')$.

*Proof.* It is easy to check that $H(X|Y',\Phi) - H(X|Y,\Phi) \leq H(X|Y',\Phi') - H(X|Y,\Phi')$, $I(X;Y|Y',\Phi) \leq I(X;Y|Y',\Phi')$, or $I(X;Y|Y',\Phi',\Phi) \leq I(X;Y|Y',\Phi')$ are equivalent statements. Given $(y',\phi')$, the inequality $I(X;Y|Y'=y',\Phi'=\phi',\Phi) \leq I(X;Y|Y'=y',\Phi'=\phi')$ is a simple application of the data processing theorem if we have $Y \to X \to \Phi$ conditioned on $(Y'=y',\Phi'=\phi')$. It remains to demonstrate this hypothesis to conclude the proof. The formula $p(y,\phi|x,y',\phi') = p(y|x,y',\phi')p(z|x,y',\phi')$ follows from $p(\phi|x,y',\phi') = p(\phi|x,y,y',\phi')$. This last identity can be shown by first applying the Bayes rule, then expanding all terms in the order $x,\phi',y$, and $y'$, further canceling common terms and, finally, repeatedly using the assumptions $X \to Y \to Y'$, $X \to \Phi \to \Phi'$, and $(Y,Y') \to X \to (\Phi,\Phi')$. □

**Theorem 5.2** [GEXIT Monotonicity] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over a BMS channel family $\{\mathrm{BMSC}_i(\mathtt{h}_i)\}_i$. Consider $i \in [n]$. Assume that the channel family $\{\mathrm{BMSC}_i(\mathtt{h}_i)\}_{\mathtt{h}_i}$ is smooth and degraded with respect to $\mathtt{h}_i$. Consider two extrinsic estimators $\Phi_i \stackrel{\triangle}{=} \phi(Y_{\sim i})$ and $\Phi_i' \stackrel{\triangle}{=} \phi'(Y_{\sim i})$ such that $X \to \Phi_i \to \Phi_i'$. Then

$$\frac{\partial H(X_i|Y_i,\Phi_i)}{\partial \mathtt{h}_i} \leq \frac{\partial H(X_i|Y_i,\Phi_i')}{\partial \mathtt{h}_i}.$$

*Proof.* The partial derivative is known to exist a.e., therefore the statement is equivalent to saying that, for any $\mathtt{h}_i' > \mathtt{h}_i$, we have $H(X_i|Y_i(\mathtt{h}_i'),\Phi_i) - H(X_i|Y_i(\mathtt{h}_i),\Phi_i) \leq H(X_i|Y_i(\mathtt{h}_i'),\Phi_i') - H(X_i|Y_i(\mathtt{h}_i'),\Phi_i')$, where $Y_i(\mathtt{h}_i)$ ($Y_i(\mathtt{h}_i')$) is the result of passing $X_i$ through $\mathrm{BMSC}(\mathtt{h}_i)$ ($\mathrm{BMSC}(\mathtt{h}_i')$, respectively). Since

$$\begin{aligned} X \to Y_i(\mathtt{h}_i) \to Y_i(\mathtt{h}_i') &\qquad \text{from channel physical degradation} \\ X \to \Phi_i \to \Phi_i' &\qquad \text{by hypothesis} \\ (Y_i(\mathtt{h}_i),Y_i(\mathtt{h}_i')) \to X \to (\Phi_i,\Phi_i') &\qquad \text{from channel memoryless assumption} \end{aligned}$$

The proof is concluded by using Lemma 5.4 and the obvious substitutions. □

Discussion: Note first that we restricted Theorem 5.2 to channels that are *binary* and *symmetric*. As shown by the proof, those two hypotheses are in fact not required and the result holds in the more general context of memoryless channels parameterized by a single scalar. Moreover observe that Lemma 5.4 plays for GEXIT functions the same role as the data processing inequality does for EXIT functions. Its consequence, i.e., Theorem 5.2, is also used to prove the monotonicity of the function over a degraded channel family and the relative "sub-optimality" of BP decoding versus MAP decoding.

**Corollary 5.1** [Monotonicity over Ordered Channels] Let $X$ be a binary vector of length $n$ chosen with probability $p_X(x)$. Assume that transmission takes place over a BMS channel family $\{\mathrm{BMSC}_i(\mathtt{h})\}_i$ where the common parameter $\mathtt{h}$ indicates the channel entropy. Consider $i \in [n]$. Assume that the channel family $\{\mathrm{BMSC}_i(\mathtt{h})\}_{\mathtt{h}}$ is smooth and degraded with respect to $\mathtt{h}$. Then $g_i^{\mathrm{MAP}}(\mathtt{h})$ is non-decreasing

in $\mathtt{h}$. Moreover, if the family is complete, then $g_i^{\text{MAP}}(0) = 0$ and $g_i^{\text{MAP}}(1) = 1$. The same is true for any GEXIT function $g_i^{\text{DEC}}(\mathtt{h})$ associated with an extrinsic estimator $\phi_i^{\text{DEC}}(Y_{\sim i})$ that preserves partial ordering imposed by physical degradation.

*Proof.* That $g_i^{\text{MAP}}(\mathtt{h})$ is non-decreasing follows from Theorem 5.2 using the substitutions $\Phi_i = \Phi^{\text{MAP}}(\mathtt{h}_i)$ and $\Phi_i' = \Phi^{\text{MAP}}(\mathtt{h}_i')$. If $\mathtt{h} = 0$, then the associated $L$-density corresponds to a "delta at infinity" (this is an easy consequence of the minimum distance being at least 2). If $\mathtt{h} = 1$ then the corresponding $L$-density is a "delta at zero." The same argument using Theorem 5.2 holds for any estimator $\phi_i^{\text{DEC}}(Y_{\sim i})$ if it preserves partial ordering imposed by physical degradation. □

The minimum distance theorem has already been observed for EXIT functions over the BEC. Let us see its general version.

**Theorem 5.3** [Minimum Distance Theorem] Let $\mathtt{C}$ be a proper binary linear code of length $n$ and minimum distance $d_{\min}$. Assume that transmission takes place over an ordered and complete smooth BMS channel family $\{\{\text{BMSC}_i(\mathtt{h})\}_i\}_{\mathtt{h}\in[0,1]}$. Then, for all $k < d_{\min}$, we have $\frac{\mathrm{d}^{k-1} g^{\text{MAP}}(\mathtt{h})}{\mathrm{d}\mathtt{h}^{k-1}}|_{\mathtt{h}=0} = 0$.

*Proof.* From Definition 5.2, the expression for the GEXIT function $g^{\text{MAP}} = \frac{1}{n}\sum_i g_i^{\text{MAP}}$ is given by $g^{\text{MAP}}(\mathtt{h}) = \frac{1}{n}\frac{\mathrm{d}}{\mathrm{d}\mathtt{h}} H(X|Y)$. Therefore $\frac{\mathrm{d}^{k-1}}{\mathrm{d}\mathtt{h}^{k-1}} g^{\text{MAP}}(\mathtt{h}) = \frac{1}{n}\frac{\mathrm{d}^k}{\mathrm{d}\mathtt{h}^k} H(X|Y(\mathtt{h}))$. Formally $\frac{\mathrm{d}}{\mathrm{d}\mathtt{h}} = \sum_i \frac{\mathrm{d}\mathtt{h}_i}{\mathrm{d}\mathtt{h}}\frac{\partial}{\partial \mathtt{h}_i} = \sum_i \frac{\partial}{\partial \mathtt{h}_i}$ such that

$$\frac{\mathrm{d}^{k-1}}{\mathrm{d}\mathtt{h}^{k-1}} g^{\text{MAP}}(\mathtt{h}) = \frac{1}{n}\sum_{i_1\dots i_k} \frac{\partial^k}{\partial \mathtt{h}_{i_1}\cdots\partial \mathtt{h}_{i_k}} H(X|Y).$$

In order to evaluate this expression for $\mathtt{h} = 0$ such that $\mathtt{h}_i = 0$ for all $i$, we can of course choose to first set $h_i$ to 0 for all bits that are not differentiated over. We get the expression

$$\frac{\mathrm{d}^{k-1}}{\mathrm{d}\mathtt{h}^{k-1}} g^{\text{MAP}}(\mathtt{h})\Big|_{\mathtt{h}_1=0,\cdots,\mathtt{h}_n=0} = \frac{1}{n}\sum_{i_1\dots i_k} \frac{\partial^k}{\partial \mathtt{h}_{i_1}\cdots\partial \mathtt{h}_{i_k}} H(X|Y_{i_1}(\mathtt{h}_{i_1})\dots Y_{i_k}(\mathtt{h}_{i_k}), X_{[n]\setminus\{i_1,\dots,i_k\}})\Big|_{\mathtt{h}_{i_1}=0,\cdots,\mathtt{h}_{i_k}=0}$$

where the terms $\frac{\partial^k}{\partial \mathtt{h}_{i_1}\cdots\partial \mathtt{h}_{i_k}} H(X|Y_{i_1}(\mathtt{h}_{i_1}),\cdots,Y_{i_k}(\mathtt{h}_{i_k}), X_{[n]\setminus\{i_1,\dots,i_k\}})$ need to be evaluated at $\mathtt{h}_{i_1} = \cdots = \mathtt{h}_{i_k} = 0$. If the code has minimum distance strictly larger than $k$, then any $n-k$ bits determine the whole codeword and $H(X|Y_{i_1}(\mathtt{h}_{i_1})\dots Y_{i_k}(\mathtt{h}_{i_k}), X_{\sim i_1\dots i_k}) = 0$. This concludes the proof. □

Finally we present a notion of duality different from the algebraic one in Appendix 2.B. This new notion is mainly operational; an application will be presented in the next chapter.

**Lemma 5.5** [GEXIT and Dual GEXIT] Let $X$ be a vector chosen with probability $p_X(x)$ from a binary code $\mathtt{C}$ of length $n$ and rate $r_{\mathtt{C}}$, and such that $p_{X_i}(x_i) = 1/2$ for all $i$. Assume that transmission takes place over a complete and smooth BMS family $\{\{\text{BMSC}_i(\mathtt{h}_{\mathtt{c}}(\mathfrak{p}))\}_i\}_{\mathfrak{p}}$ whose equivalent family of $L$-densities is $\{\mathtt{c}_{\mathfrak{p}}\}_{\mathfrak{p}}$. The entropy associated with $\mathtt{c}_{\mathfrak{p}}$ is $\mathtt{h}_{\mathtt{c}}(\mathfrak{p}) \in [0,1]$, and the standard GEXIT function is represented in parametric form by $\left\{\left(\mathtt{h}_{\mathtt{c}}(\mathfrak{p}), \frac{1}{n}\sum_{i\in[n]} \frac{\partial H(X_i|Y_i,\Phi_i^{\text{MAP}}))}{\partial \mathtt{h}_{\mathtt{c}}}(\mathfrak{p})\right)\right\}_{\mathfrak{p}}$. In a symmetric manner let $\{\mathtt{a}_{\mathfrak{p}}\}_{\mathfrak{p}}$ denote the family formed by (uniformly averaged) extrinsic MAP $L-$densities, and let $\mathtt{h}_{\mathtt{a}}(\mathfrak{p})$ be the entropy associated with $\mathtt{a}_{\mathfrak{p}}$. Then $\{\mathtt{a}_{\mathfrak{p}}\}$ is a smooth and complete family, and we define the *dual GEXIT curve* in parametric form by $\left\{\left(\frac{1}{n}\sum_{i\in[n]} \frac{\partial H(X_i|Y_i,\Phi_i^{\text{MAP}}))}{\partial \mathtt{h}_{\mathtt{a}}}(\mathfrak{p}), \mathtt{h}_{\mathtt{a}}(\mathfrak{p})\right)\right\}_{\mathfrak{p}}$. For both, standard and dual EXIT curve, the total area under the curve equals $r_{\mathtt{C}}$ over the range $[0,1]$.

*Proof.* By definition the first curve represents the standard GEXIT function. Let us focus on the second curve, i.e., the dual GEXIT curve: The only statement that requires a proof concerns the area under this curve. Consider the channel $p_{\Phi_i(\mathfrak{p})|X_i}$ where $\Phi_i(\mathfrak{p})$ is the extrinsic MAP estimate, and let $\mathtt{h}_{\mathtt{a}_i}(\mathfrak{p}) \triangleq H(X_i|\Phi_i(\mathfrak{p}))$ denote its entropy (extrinsic entropy or EXIT entropy). Consider the channel $p_{Y_i(\mathfrak{p})|X_i}$ where $Y_i(\mathfrak{p})$ is the intrinsic estimate, and let $\mathtt{h}_{\mathtt{c}_i}(\mathfrak{p}) \triangleq H(X_i|Y_i(\mathfrak{p}))$ denotes its entropy (intrinsic entropy). By assumption $\mathfrak{p}$ parameterizes the complete channel family $\{p_{\Phi_i(\mathfrak{p})|X_i}\}_{\mathfrak{p}}$, i.e., it is in a one-to-one correspondence with the channel entropy $\mathtt{h}_{\mathtt{c}_i}(\mathfrak{p})$ which ranges from 0 to 1, see Section 2.8.

Therefore $\mathtt{h}_{\mathtt{c}_i}$, as well as $\mathtt{h}_{\mathtt{a}_i}$ (because of the monotonicity of the EXIT function and $p_{X_i}(x_i) = 1/2$), are possible reparameterizations of the system over $[0,1]$. Furthermore,

$$\frac{\mathrm{d}}{\mathrm{d}\mathtt{p}} H(X_i|Y_i(\mathtt{p}),\Phi_i(\mathtt{p})) = \frac{\partial H(X_i|Y_i(\mathtt{h}_{\mathtt{c}_i}),\Phi_i(\mathtt{h}_{\mathtt{a}_i}))}{\partial \mathtt{h}_{\mathtt{a}_i}} \frac{\mathrm{d}\mathtt{h}_{\mathtt{a}_i}(\mathtt{p})}{\mathrm{d}\mathtt{p}} + \frac{\partial H(X_i|Y_i(\mathtt{h}_{\mathtt{c}_i}),\Phi_i(\mathtt{h}_{\mathtt{a}_i}))}{\partial \mathtt{h}_{\mathtt{c}_i}} \frac{\mathrm{d}\mathtt{h}_{\mathtt{c}_i}(\mathtt{p})}{\mathrm{d}\mathtt{p}}. \tag{5.3}$$

First, sum this identity over all $i$, divide by $n$, notice that the intrinsic density is independent of the location $i$, and consider the average extrinsic density. Integrate now this relationship over the whole range of $\mathtt{p}$, which goes from "perfect" (channel) to "useless" (channel). The integral on the left-hand side equals 1. On the right-hand side the first term corresponds to the standard GEXIT function and its area equals $r_{\mathtt{C}}$ by the area theorem. The roles of the two densities are exchanged for the second term so that it corresponds to the GEXIT curve $\left\{ \left(\mathtt{h}_{\mathtt{a}}(\mathtt{p}), \frac{1}{n}\sum_{i\in[n]} \frac{\partial H(X_i|Y_i,\Phi_i^{\mathrm{MAP}}))}{\partial \mathtt{h}_{\mathtt{a}}}(\mathtt{p})\right)\right\}_{\mathtt{p}}$. Since the sum of the two areas equals one and the area under the standard GEXIT curve equals $r_{\mathtt{C}}$, it follows that the area under the second curve equals $1-r_{\mathtt{C}}$. Finally, note that if we consider the inverse of the second curve by exchanging the two coordinates, then the area under this curve is equal to $1-(1-r_{\mathtt{C}}) = r_{\mathtt{C}}$. □

Discussion: Note first that both curves are "comparable" in the sense that the first component measures the channel $\mathtt{c}$ and the second argument measures the extrinsic density $\mathtt{a}$. The difference between the two lies in the choice of measure that is applied to each component.
Second, from an operational point of view, it is more convenient to work with linear operators (assuming that $\mathtt{C}$ is a proper binary linear code). In this case, whereas the standard GEXIT curve is given in parametric form by $\{\mathtt{H}(\mathtt{c}_{\mathtt{p}}), \mathtt{G}(\mathtt{c}_{\mathtt{p}},\mathtt{a}_{\mathtt{p}})\}$, the *dual* GEXIT curve is given in parametric form by $\{\mathtt{G}(\mathtt{a}_{\mathtt{p}},\mathtt{c}_{\mathtt{p}}), \mathtt{H}(\mathtt{a}_{\mathtt{p}})\}$. In this operational representation, an alternative proof follows from the derivative of $\mathtt{H}(\mathtt{c}_{\mathtt{p}} \circledast \mathtt{a}_{\mathtt{p}})$ which represents the total bit entropy conditioned on the observations. We further get the formula

$$\mathrm{d}\mathtt{H}(\mathtt{a}_{\mathtt{p}} \circledast \mathtt{c}_{\mathtt{p}}) = \mathtt{G}(\mathtt{c}_{\mathtt{p}},\mathtt{a}_{\mathtt{p}})\mathrm{d}\mathtt{H}(\mathtt{c}_{\mathtt{p}}) + \mathtt{G}(\mathtt{a}_{\mathtt{p}},\mathtt{c}_{\mathtt{p}})\mathrm{d}\mathtt{H}(\mathtt{a}_{\mathtt{p}})$$

which is the operational form of Eq. (5.3). The left-hand side is the total entropy variation; it decomposes into a term due to the variation of the intrinsic entropy and a term due to the variation of the extrinsic entropy.

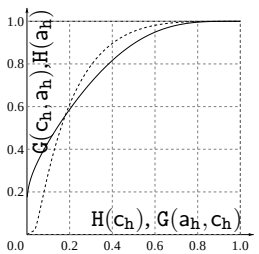



Figure 5.4: Standard and dual GEXIT function: $[5,4,2]$ single parity-check codes code and transmission over BSC$(\mathtt{h})$.

From the isotropy property discussed in Chapter 3, we know that the individual extrinsic densities coincide in many cases with the average extrinsic density. This is the case for single parity-check or repetition codes. An example is given in Figure 5.4, which shows the standard GEXIT function and the dual GEXIT function for the $[5,4,2]$ single parity-check code and transmission over the BSC. Although the two curves have quite distinct shapes, the area under the two curves remains the same.
In the next chapter, the duality notion is used to show that, inherently, iterative coding systems cannot surpass capacity.

## 5.3 GEXIT Charts and Matching Condition

The upper bound on the MAP threshold, which we stated in Chapter 4 for the erasure channel, and which we will state in the next chapter for general BMS channels, cannot be larger than the Shannon threshold $1-r$. This implies that iterative coding systems do not allow to communicate reliably above channel capacity. Of course, this is a straightforward consequence of Shannon's channel coding theorem. However, although the final result is trivial, the method of proof is well worth the effort because it shows how capacity enters in the calculation of the performance of iterative coding systems. There is an even more satisfying way to show why we can not surpass capacity: This is the *matching condition* introduced for the BEC in Section 3.4.3. In the remainder of this section, we extend the matching condition to general BMS channels and, with this aim, we introduce *GEXIT charts*. The interest in this "matching" approach is three-fold. First, compared to our upper bounding technique, it does not require the assumption of communication over a smooth channel family. Second, it is based on a *dynamical* description of the decoding process, and therefore uses only quantities appearing in density

evolution (and not just fixed points). Third, component codes (and their "matching") play a crucial role in the optimization of coding schemes for practical issues.

In order to follow the proof technique of Section 3.4.3, we need a suitable one-dimensional representation of density evolution, see Section 3.2. Such a convenient *chart*, similar to EXIT charts but that takes further advantage of an area theorem, is the *GEXIT chart* that measures the exact intermediate densities of the decoding process and uses the GEXIT operator. Motivated by the geometric statement observed for the BEC and the relationship between the derivative of the mutual information and the MMSE introduced in [40, 148], a similar chart for BMS channels is proposed in [46]. Assuming that the input densities to the component codes are Gaussian, this chart again fulfills the area theorem. In order to apply the MMSE chart in the context of iterative coding the authors propose to approximate the intermediate densities that appear in density evolution by "equivalent" Gaussian densities. This was an important first step in generalizing the matching condition to the whole class of BMS channels. In the following we show how to overcome the need for making the Gaussian approximation by using GEXIT functions and interpolating intermediate densities.

Let us first review the case of transmission over BEC($\mathtt{h}$) using a dd pair $(\lambda,\rho)$ as presented in Section 3.4.3. In this case, density evolution is equivalent to the EXIT chart method and the condition for successful decoding under BP reads $c(\mathtt{x}) \stackrel{\triangle}{=} 1-\rho(1-\mathtt{x}) \leq \lambda^{-1}(\mathtt{x}/\mathtt{h}) \stackrel{\triangle}{=} \nu_{\mathtt{h}}^{-1}(\mathtt{x})$. The area under the curve $c(\mathtt{x})$ is equal to $1-\int\rho$ and the area to the left of the curve $\nu_{\mathtt{h}}^{-1}(\mathtt{x})$ is equal to $\mathtt{h}\int\lambda$. A necessary condition for successful BP decoding is then that these two areas do not overlap. Since the total area equals 1 we get the necessary condition $\mathtt{h} \leq \frac{\int\rho}{\int\lambda} = 1-r_{\lambda,\rho}$. In other words, the design rate $r_{\lambda,\rho}$ of any LDPC ensemble which, for increasing block lengths, allows for successful decoding over BEC($\mathtt{h}$), cannot surpass the Shannon limit $1-\mathtt{h}$.

By turning this bound around, we can find conditions under which iterative systems achieve capacity as discussed in Section 3.4.4. In particular, it shows that the two component EXIT curves have to be matched perfectly. Indeed, all currently known capacity-achieving dd pairs for the BEC can be derived by starting with this perfect matching condition and working backwards. Let us now show that, by using component GEXIT functions, the matching condition holds in the general case. This might in the future serve as a starting point to find capacity-achieving (or at least capacity-approaching) dd pairs for general BMSCs. (Observe that, if the design rate is shown to approach capacity, then necessarily, the actual asymptotic rate is potentially larger and therefore does at least as well). We need one preliminary definition.

**Definition 5.4** [Interpolating Channel Families] Consider a dd pair $(\lambda,\rho)$ and transmission over a BMSC characterized by its $L$-density $\mathsf{c}$. Let $\mathsf{a}_0 = \Delta_0$ and $\mathsf{a}_1 = \mathsf{c}$ and set $\mathsf{a}_\alpha$, $\alpha \in [0,1]$, to $\mathsf{a}_\alpha = (1-\alpha)\mathsf{a}_0 + \alpha\mathsf{a}_1$. The *interpolating density evolution families* $\{\mathsf{a}_\alpha\}_{\alpha=0}^{\infty}$ and $\{\mathsf{b}_\alpha\}_{\alpha=0}^{\infty}$ are defined as $\mathsf{b}_\alpha = \sum_i \rho_i \mathsf{a}_\alpha^{\boxplus(i-1)}$ and $\mathsf{a}_{\alpha+1} = \sum_i \lambda_i \mathsf{c}\circledast\mathsf{b}_\alpha^{\circledast(i-1)}$ for $\alpha \geq 0$.

Discussion: First note that, with the conventions of Section 3.2, $\mathsf{a}_\ell$ ($\mathsf{b}_\ell$), $\ell \in \mathbb{N}$, represents the sequence of $L$-densities of density evolution emitted by the variable (check) nodes in the $\ell$-th iteration. By starting density evolution not only with $\mathsf{a}_1 = \mathsf{c}$ (or equivalently $\mathsf{a}_0 = \Delta_0$) but with all possible convex combinations of $\Delta_0$ and $\mathsf{c}$, this discrete sequence of densities is completed to form a continuous family of densities ordered by physical degradation. The fact that the densities are ordered by physical degradation can be seen as follows: note that the computation tree for $\mathsf{a}_\alpha$ can be constructed by taking the standard computation tree of $\mathsf{a}_{\lceil\alpha\rceil}$ and independently erasing the observation associated with each variable leaf node with probability $\lceil\alpha\rceil - \alpha$. It follows that we can convert the computation tree of $\mathsf{a}_\alpha$ to that of $\mathsf{a}_{\alpha-1}$ by erasing all observations at the leaf nodes and by independently erasing each observation in the second (from the bottom) row of variable nodes with probability $\lceil\alpha\rceil - \alpha$. The same statement is true for $\mathsf{b}_\alpha$. Moreover, if $\lim_{\ell\to\infty} \mathtt{H}(\mathsf{a}_\ell) = 0$, i.e., if BP decoding is successful in the limit of large blocklengths, then the families are both complete.

**Example 5.11** [Density Evolution and Interpolation] Consider transmission over BSC($\epsilon = 0.07$) using a $(3,6)$-regular ensemble. Figure 3.2 in Section 3.2 depicts the density evolution process for this case. Density evolution gives rise to the sequences of densities $\{\mathsf{a}_\ell\}_{\ell=0}^{\infty}$, and $\{\mathsf{b}_\ell\}_{\ell=0}^{\infty}$. Figure 5.5 shows the interpolation of these sequences for the choices $\alpha = 1.0, 0.95, 0.9$ and $0.8$ and the complete family with $\alpha \in [0,1]$: the resulting densities are projected onto a two-dimensional chart using the EXIT operator.

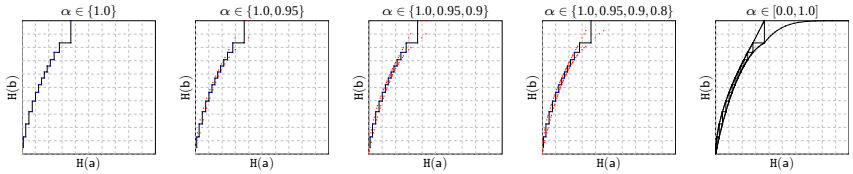



Figure 5.5: Interpolation of densities using the method of Definition 5.4 and EXIT representation.

**Lemma 5.6** [Matching Condition] Consider a dd pair $(\lambda,\rho)$ and transmission over a BMS channel characterized by its $L$-density $\mathsf{c}$ so that density evolution converges to $\mathcal{A}_\infty$. Let $\{\mathsf{a}_\alpha\}_{\alpha=0}^{\infty}$ and $\{\mathsf{b}_\alpha\}_{\alpha=0}^{\infty}$ denote the interpolated families as defined in Definition 5.4. Then the two GEXIT curves $\{\mathsf{H}(\mathsf{a}_\alpha),\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_\alpha)\}$, which depicts the GEXIT curve for parity-check nodes, and $\{\mathsf{H}(\mathsf{a}_{\alpha+1}),\mathsf{G}(\mathsf{a}_{\alpha+1},\mathsf{b}_\alpha)\}$, which depicts the inverse of the dual GEXIT curve for variable nodes, do not cross and faithfully represent density evolution. Further, the area under the "check node" GEXIT function is equal to $1-\int\rho$ and the area to the left of the "inverse dual variable node" GEXIT function is equal to $\mathsf{H}(\mathsf{c})\int\lambda$. It follows that $r_{\lambda,\rho}\leq 1-\mathsf{H}(\mathsf{c})$, i.e., the design rate can not exceed the Shannon limit.

*Proof.* On the one hand, note that $\{\mathsf{H}(\mathsf{a}_\alpha),\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_\alpha)\}$ is the standard GEXIT curve representing the action of the check nodes: $\mathsf{a}_\alpha$ denotes the density of the messages entering the check nodes and $\mathsf{b}_\alpha$ represents the density of the corresponding output messages. On the other hand, $\{\mathsf{H}(\mathsf{a}_{\alpha+1}),\mathsf{G}(\mathsf{a}_{\alpha+1},\mathsf{b}_\alpha)\}$ is the inverse of the dual GEXIT curve corresponding to the action at the variable nodes: $\mathsf{b}_\alpha$ represents the density of the messages entering the variable nodes and $\mathsf{a}_{\alpha+1}$ denotes the output density.
The fact that the two curves do not cross can be seen as follows. Fix an entropy value. This entropy value corresponds to a density $\mathsf{a}_\alpha$ for a unique value of $\alpha$. The fact that $\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_\alpha)\leq\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_{\alpha-1})$ now follows from the fact that $\mathsf{b}_\alpha\prec\mathsf{b}_{\alpha-1}$ and that for any symmetric $\mathsf{a}_\alpha$ this relationship is preserved by applying the GEXIT functional according to Theorem 5.2.
The statements regarding the areas of the two curves follow from the general area theorem and Lemma 5.5. The bound on the achievable rate follows in the same manner as for the BEC: the total area of the GEXIT box equals one and the two curves do not overlap and have areas $1-\int\rho$ and $\mathsf{H}(\mathsf{c})$. Therefore $1-\int\rho+\mathsf{H}(\mathsf{c})\int\lambda\leq 1$, which concludes the proof. □

We see that the matching condition still holds for general BMSCs. There are a few important differences between the general case and the simple case of transmission over the BEC. For the BEC, the intermediate densities are always the BEC densities that are *independent of the degree distribution*. This, of course, enormously simplifies the task. Further, for the BEC, given the two EXIT curves, the progress of density evolution is simply given by a staircase function bounded by the two EXIT curves. For a general BMSC, this staircase function still has vertical pieces, but the "horizontal" pieces have in general a non-vanishing slope. This is true because the $y$-axis for the "check node" step measures $\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_\alpha)$, but in the subsequent "inverse variable node" step it measures $\mathsf{G}(\mathsf{a}_{\alpha+1},\mathsf{b}_\alpha)$. Therefore, one should think of two sets of labels on the $y$-axis, one measuring $\mathsf{G}(\mathsf{a}_\alpha,\mathsf{b}_\alpha)$, and the second one measuring $\mathsf{G}(\mathsf{a}_{\alpha+1},\mathsf{b}_\alpha)$. The "horizontal" step then consists of first switching from the first $y$-axis to the second (so that the labels correspond to the same density $\mathsf{b}_\alpha$) and then drawing a horizontal line until it crosses the "inverse variable node" GEXIT curve. The "vertical" step stays as before, i.e., it really corresponds to drawing a vertical line. This is certainly best clarified by a simple example.

**Example 5.12** [GEXIT Chart] Consider the $(3,6)$-regular ensemble and transmission over $\mathrm{BSC}(0.07)$. The corresponding illustrations are shown in Figure 5.6. The two pictures on the left show the standard GEXIT curve for the check node side and the dual GEXIT curve corresponding to the variable node side. In order to use these two curves in the same chart, it is convenient to consider the inverse function for the variable node side. In the right-most picture (GEXIT chart) both curves are shown together with the "staircase" like function that represents density evolution. The two curves do not overlap and both have areas equal to the rate.

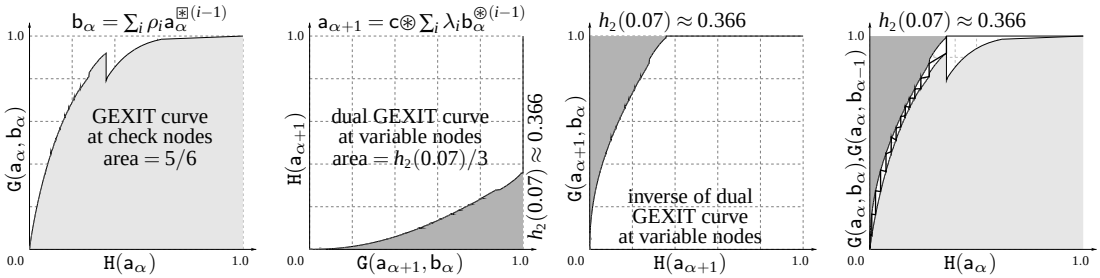



Figure 5.6: Faithful representation of density evolution by two non-overlapping component-wise GEXIT functions that represent the "actions" of the check nodes and variable nodes, respectively. As for the dynamical picture over the BEC, the area between the two curves is proportional to the additive gap to capacity.

As remarked in the last section, one potential use of the matching condition is to find capacity approaching dd pairs. Let us quickly outline a further potential application. Assuming that we have found a sequence of capacity-achieving degree distributions, how does the number of required iterations scale as we approach capacity? It has been conjectured that the number of required iterations scales like $1/\delta$, where $\delta$ is the gap to capacity. This conjecture is based on the geometric picture that is implied by the matching condition. To make things simple, imagine the two GEXIT curves as two parallel lines, let us say both at a 45 degree angle, a certain distance apart, and think of density evolution as a staircase function. From the previous results, the area between the lines is proportional to $\delta$. Therefore, if we half $\delta$, the distance between the lines has to be halved and one would expect that we need twice as many steps. Obviously, the above discussion was based on a number of simplifying assumptions. It remains to be seen if this conjecture can be proved rigorously.

## 5.4 Conclusion and Discussion

We have seen in this chapter how to prove the empirical area rules observed for EXIT curves. The price to pay was to replace the EXIT function by the GEXIT function. GEXIT functions have a fundamental meaning that goes back to the normalized conditional entropy and are, fortunately, as simple to compute as standard EXIT functions.

This thesis is mainly dedicated to analyzing the relationship between MAP and BP decoding. This relationship appears in the limit of large blocklengths and is based on EXIT functions in the erasure case. In the next chapter we will see how GEXIT functions apply in the general framework. We will begin by deriving an upper bound on the MAP threshold (which we conjecture to be tight) based on the general area theorem. We further present a general EBP GEXIT curve and prove elements for generalizing the Maxwell construction to BMSCs.

# Appendix

## 5.A GEXIT Kernel and Concavity

We provide a direct calculus proof of Theorem 5.2, exploiting the explicit representation provided by Lemma 5.2. As a byproduct we show that the GEXIT kernel in the $|D|$-domain is non-increasing and concave. This fact is also used in the proof of Lemma 5.3.

For our purpose it is convenient to represent all quantities in the $|D|$-domain. Let $\{\mathsf{c}_\mathtt{h}{}^{|D|}\}_\mathtt{h}$ denote the family of $|D|$-densities characterizing the channel family $\{\mathrm{BMSC}(\mathtt{h})\}_\mathtt{h}$. Let $|d|^{\mathrm{BMSC}(\mathtt{h})}(w)$ denote the GEXIT kernel in the $|D|$-domain as introduced in Eq. (5.2). We can rewrite it in the form

$$|d|^{\mathrm{BMSC}(\mathtt{h})}(w) = \int_0^1 \frac{\partial \mathsf{c}_\mathtt{h}{}^{|D|}(z)}{\partial \mathtt{h}} \alpha(z,w)\mathrm{d}z,$$

where $\alpha(z,w) = \frac{1}{4}\sum_{i,j=\pm 1}(1+iz)(1+jw)\beta(iz,jw)$ with $\beta(z,w) = \log_2\left(1+e^{-2\tanh^{-1}(z)}e^{-2\tanh^{-1}(w)}\right)$. Finally, let $\mathsf{a}^{|D|}$ and $\mathsf{b}^{|D|}$ denote the two symmetric densities in the $|D|$-domain. The claim of the theorem is then equivalent to the statement that the GEXIT functional $\int_0^1 |d|^{\mathrm{BMSC}(\mathtt{h})}(w)\mathsf{a}^{|D|}(w)\mathrm{d}w$ preserves the partial order implied by physical degradation. This means that if $\mathsf{a}^{|D|} \prec \mathsf{b}^{|D|}$ then

$$\int_0^1 |d|^{\mathrm{BMSC}(\mathtt{h})}(w)a^{|D|}(w)\mathrm{d}w \leq \int_0^1 |d|^{\mathrm{BMSC}(\mathtt{h})}(w)b^{|D|}(w)\mathrm{d}w.$$

It is shown in [65] that a $|D|$-domain kernel preserves the partial order implied by physical degradation if it is non-increasing and concave on $[0,1]$, i.e., if its first two derivatives are non-positive. Therefore we need to show that $\int_0^1 \frac{\mathrm{d}\mathsf{c}_\mathtt{h}{}^{|D|}(z)}{\mathrm{d}\mathtt{h}} \frac{\partial^i \alpha(z,w)}{\partial w^i}\mathrm{d}z \leq 0$, for $i = 1,2$. By the same theorem in [65] the above condition is verified if both $\frac{\partial^i \alpha(z,w)}{\partial w^i}$ for $i = 1,2$, are convex and non-decreasing. This in turn is true if $\frac{\partial^{i+j}\alpha(z,w)}{\partial w^i \partial z^j} \geq 0$ for $i,j = 1,2$. Now some further calculus shows that

$$\begin{aligned}\frac{\partial \alpha(z,w)}{\partial w} &= \frac{1}{2}\sum_{i=\pm 1} iz\log_2(1+iwz) - \frac{1}{2}\sum_{i=\pm 1} i\log_2(1+iw),\\ \log(2)\frac{\partial^2 \alpha(z,w)}{\partial w^2} &= \frac{z^2}{1-w^2z^2} - \frac{1}{1-w^2}.\end{aligned}$$

Note that the last identity implies that $\frac{\partial^2\alpha(z,w)}{\partial w^2}$ has a positive expansion in $z$ (except for the constant term). Therefore the derivatives $\frac{\partial^{2+i}\alpha(z,w)}{\partial w^2 \partial z^i}$, $i = 1,2$, are both positive and by symmetry of the function $\alpha(z,w)$ in its arguments $z$ and $w$ so is $\frac{\partial^3\alpha(z,w)}{\partial w \partial z^2}$. Finally,

$$\log(2)\frac{\partial^2\alpha(z,w)}{\partial w \partial z} = \frac{1}{2}\ln\frac{1+wz}{1-wz} + \frac{wz}{1-w^2z^2} = 2wz\sum_{i\geq 0}\frac{(i+1)(w^2z^2)^i}{2i+1},$$

which has a positive Taylor series expansion as well. This confirms our claim that the GEXIT kernel preserves the partial order implied by physical degradation.

## 5.B Non-Binary GEXIT Functions

Consider a (not necessarily binary) input alphabet $\mathcal{X}$. The concept of GEXIT functions extends naturally to the non-binary (non-symmetric) case.

**Definition 5.5** [GEXIT Function over $\mathcal{X}$] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$ from $\mathcal{X}^n$. Assume that the channel $p_{Y|X}$ is memoryless. Assume moreover that $Y_i$ is the result of passing $X_i$ through a channel $p^{\epsilon_i}_{Y_i|X_i}$ parameterized by $\epsilon_i \in [0,1]$. Consider any (extrinsic) estimator $\Phi_i^{\mathrm{DEC}}(Y_{\sim i})$ taking value on the $(|\mathcal{X}|-1)$-dimensional simplex. Let $\Omega$ be a further observation of $X$ such that $\Omega \to X \to Y$. Consider $i \in [n]$. If $\{p^{\epsilon_i}_{Y_i|X_i}\}_{\epsilon_i}$ is smooth, then the $i^{\text{th}}$ GEXIT function associated with the given channel parameterization and given (extrinsic) estimator is defined as

$$g_i^{\mathrm{DEC}}(\epsilon) \triangleq \frac{\partial H(X_i|Y_i,\Phi_i^{\mathrm{DEC}},\Omega)}{\partial \epsilon_i},$$

where the entropy uses the natural logarithm (i.e., log instead of the base two logarithm denoted by $\log_2$).

Assume that the considered extrinsic estimator is any sufficient statistic of $X_i$ given $Y_{\sim i}$. For example, if the channel input alphabet is finite and discrete, one may take $\phi_i^{\text{MAP}}(y_{\sim i}) = \{p_{X_i|Y_i}(x_i|y_{\sim i});\, x_i \in \mathcal{X}\}$, which takes value on the $(|\mathcal{X}|-1)$-dimensional simplex, or any parameterization of it (see Section 2.11). Then, if the (MAP) GEXIT function is defined for all $i$, and if all individual channels are parameterized in a smooth way by a common parameter $\mathfrak{p}$, i.e., $\epsilon_i = \epsilon_i(\mathfrak{p})$, $i \in [n]$, then again the general area theorem holds. Notice that, in a slight generalization of the notion of GEXIT function, this definition considers the GEXIT function as a function of the channel parameter rather than the channel entropy.

**Lemma 5.7** [GEXIT Function for General Memoryless Channels] Let $X$ be a vector of length $n$ chosen with probability $p_X(x)$. Assume that the (discrete) channel $p_{Y|X}$ is memoryless. Assume moreover that $Y_i$ is the result of passing $X_i$ through a channel $p^{\epsilon_i}_{Y_i|X_i}$ parameterized by $\epsilon_i \in [0,1]$. Consider any extrinsic estimator $\Phi_i^{\text{DEC}}$ taking value on the $(|\mathcal{X}|-1)$-dimensional simplex. If $\{p^{\epsilon_i}_{Y_i|X_i}\}_{\epsilon_i}$ is smooth, then the $i^{\text{th}}$ GEXIT function associated with the considered channel parameterization is given by

$$g_i^{\text{DEC}}(\epsilon) = \int_{\phi_i^{\text{DEC}}, y_i} \sum_{x_i} p(x_i)\, p(\phi_i^{\text{DEC}}|x_i)\, \frac{\mathrm{d}}{\mathrm{d}\epsilon_i} p^{\epsilon_i}(y_i|x_i) \cdot \log \left\{ \sum_{x_i'} \frac{p(x_i'|\phi_i^{\text{DEC}})\, p(y_i|x_i')}{p(x_i|\phi_i^{\text{DEC}})\, p(y_i|x_i)} \right\} \mathrm{d}y_i \mathrm{d}\phi_i^{\text{DEC}}.$$

*Proof.* We first expand the conditional entropy

$$\begin{aligned} H(X_i|\Phi_i^{\text{DEC}}, Y_i) &= -\int_{\phi_i^{\text{DEC}}, y_i} \sum_{x_i} p(x_i, \phi_i^{\text{DEC}}, y_i) \log(p(x_i|\phi_i^{\text{DEC}}, y_i)) \mathrm{d}y_i \mathrm{d}\phi_i^{\text{DEC}} \\ &= -\int_{\phi_i^{\text{DEC}}, y_i} \sum_{x_i} p(x_i)\, p(\phi_i^{\text{DEC}}|x_i)\, p(y_i|x_i) \cdot \log \left\{ \frac{p(x_i|\phi_i^{\text{DEC}})\, p(y_i|x_i)}{\sum_{x_i' \in \mathcal{X}} p(x_i'|\phi_i^{\text{DEC}})\, p(y_i|x_i')} \right\} \mathrm{d}y_i \mathrm{d}\phi_i^{\text{DEC}}. \end{aligned}$$

This form has the advantage that the dependence of $H(X_i|\Phi_i^{\text{DEC}}, Y_i)$ upon the channel at position $i$ is completely explicit. Let us therefore differentiate the above expression with respect to $\epsilon_i$, the parameter that governs the transition probability $p(y_i|x_i)$. The terms obtained by differentiating with respect to the channel *inside* the log vanish. For instance, when differentiating the $p(y_i|x_i)$ at the numerator, we get $-\int_{\phi_i^{\text{DEC}}, y_i} \sum_{x_i} p(x_i) p(\phi_i^{\text{DEC}}|x_i) \frac{\mathrm{d}p(y_i|x_i)}{\mathrm{d}\epsilon_i} \mathrm{d}y_i \mathrm{d}\phi_i^{\text{DEC}} = -\int_{\phi_i^{\text{DEC}}} \sum_{x_i} p(x_i) p(\phi_i^{\text{DEC}}|x_i) \frac{\mathrm{d}}{\mathrm{d}\epsilon_i} \int_{y_i} p(y_i|x_i) \mathrm{d}y_i \mathrm{d}\phi_i^{\text{DEC}} = 0$. When differentiating with respect to the *outer* $p(y_i|x_i)$ we get the stated result. □

Consider a BMSC, it is now a straightforward exercise to (re)derive Lemma 5.2. See also [53].

## 5.C GEXIT Kernel for Gaussian Channels

This appendix contains a few useful results concerning the GEXIT kernel for Gaussian channels.

**Lemma 5.8** [Characterization of GEXIT Kernel, $L$-Domain – $\{\text{BAWGNC}(\mathtt{h})\}$] Consider the family $\{\mathsf{c}_{\text{BAWGNC}(\mathtt{h}=\mathtt{h}(\sigma))}\}$ of BAWGN channels, where $\mathtt{h}$ denotes the channel entropy. Recall from Chapter 2 that the channel is modeled as $Y = X + Z$, where $X$ takes values $x \in \mathcal{X} = \{-1, +1\}$ and $Z$ is Gaussian with zero mean and variance $\sigma^2$. Then the following represents *equivalent* kernels:

$$\text{(i)}\ \ l^{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}}(z) = \left( \mathrm{e}^{-z} \int_{-\infty}^{+\infty} \frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{(\cosh(\frac{w-z}{2}))^2} \mathrm{d}w \right) \Big/ \left( \int_{-\infty}^{+\infty} \frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{(\cosh(\frac{w}{2}))^2} \mathrm{d}w \right),$$

$$\text{(ii)}\ \ l'^{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}}(z) = \frac{1 - \mathbb{E}[\mathbb{E}[X|Y, \Phi = z]^2]}{1 - \mathbb{E}[\mathbb{E}[X|Y]^2]}, \qquad \text{(iii)}\ \ l''^{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}}(z) = \frac{1 - \mathbb{E}[\mathbb{E}[X|Y, \Phi = z]|X = +1]}{1 - \mathbb{E}[\mathbb{E}[X|Y]|X = +1]}.$$

Hereby, $\Phi$ denotes a further observation of $X$ conditionally independent of $Y$: It is the result of passing $X$ through a symmetric channel, and it is assumed to be in the LLR form (if we use coding, $\Phi$ represents the extrinsic estimate of $X$).

Discussion: This lemma provides several equivalent representations of the kernel for the BAWGN channel. The expression (ii) shows the relationship between conditional entropy and minimum mean-square error (MMSE) estimator (see footnote in Section 2.3). To see this, observe first that the denominator is a ($z$ independent) scaling factor that depends on our parameterization of the channel through its entropy $\mathfrak{h}$. Second, observe that the numerator $1-\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]^2]=\mathbb{E}[\mathbb{E}[X^2|Y,\Phi=z]-\mathbb{E}[X|Y,\Phi=z]^2]$ is the MMSE estimator (which in this framework includes the decoding estimate $z$). This relationship, which connects a fundamental information theoretic quantity to a measure widely-used in signal processing, was first observed in [40, 148]. In the above lemma, the channel inputs are binary. In Lemma 5.10 we give an alternative way of deriving $I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)$ in the more general context of non-binary channel inputs. The form (iii) provides a further simplification. This expression, in which the numerator shows the magnetization was first stated in [41] using the Nishimori identity (in the context of coding, this identity was first discussed in [31]).

Before proving Lemma 5.8, let us recall the following elementary fact used several times in the proof Lemma 5.8. Let $p_{Y|X}(y|x)$ be a BMSC and let $f(y)$ be a measurable function. If $f(y)$ is even, then

$$\mathbb{E}[f(Y)]=\mathbb{E}[f(Y)|X=+1]. \tag{5.4}$$

*Proof of Lemma 5.8.* The channel $L-$density is given by $\mathsf{c}(w)\triangleq\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}(w)=\frac{\sigma}{\sqrt{8\pi}}\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}$.
(i) The kernel as stated in Eq. (5.1) is expressed in terms of the derivative of $\mathsf{c}(w)$ with respect to the channel parameter. Let us use the channel parameterization $\mathfrak{p}\triangleq 2/\sigma^2$. We get a pleasing analytic expression because, for the Gaussian case, we can express this derivative via the identity $\frac{\partial\mathsf{c}(w)}{\partial\mathfrak{p}}=-\frac{\partial\mathsf{c}(w)}{\partial w}+\frac{\partial^2\mathsf{c}(w)}{\partial w^2}$. Then using twice integration by parts (as in [41]), we get

$$\begin{aligned}
I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)\cdot\log(2)\cdot\frac{\partial\mathfrak{h}}{\partial\mathfrak{p}}&=\int_{-\infty}^{+\infty}\frac{\partial\mathsf{c}(w)}{\partial\mathfrak{p}}\log(1+\mathrm{e}^{-w-z})\mathrm{d}w\\
&=\int_{-\infty}^{+\infty}\frac{\partial\mathsf{c}(w)}{\partial w}\frac{\mathrm{e}^{-w-z}}{1+\mathrm{e}^{-w-z}}\mathrm{d}w-\int_{-\infty}^{+\infty}\mathsf{c}(w)\frac{\mathrm{e}^{-w-z}}{1+\mathrm{e}^{-w-z}}\mathrm{d}w\\
&=\int_{-\infty}^{+\infty}\mathsf{c}(w)\frac{-1}{(1+\mathrm{e}^{w+z})^2}\mathrm{d}w\;=\frac{-\mathrm{e}^{-z}}{4}\int_{-\infty}^{+\infty}\frac{\mathsf{c}(-w)}{(\cosh(\frac{w+z}{2}))^2}\mathrm{d}w.
\end{aligned}$$

The computation of $\frac{\partial\mathfrak{h}}{\partial\mathfrak{p}}$ is exactly the same if we set $z=0$. Therefore,

$$I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)\triangleq\left(\mathrm{e}^{-z}\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w-\mathfrak{p})^2}{4\mathfrak{p}}}}{(\cosh(\frac{w-z}{2}))^2}\mathrm{d}w\right)\Big/\left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w-\mathfrak{p})^2}{4\mathfrak{p}}}}{(\cosh(\frac{w}{2}))^2}\mathrm{d}w\right).$$

(ii) First, we claim that the previous expression can be written as

$$I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)=\mathrm{e}^{-z}\frac{1-\mathbb{E}[\mathbb{E}[X|Y,\Phi=-z]^2]}{1-\mathbb{E}[\mathbb{E}[X|Y]^2]}.$$

To see this, observe that

$$w+z\overset{(a)}{=}\log\frac{p_{\frac{2Y}{\sigma^2}|X}(w|+1)}{p_{\frac{2Y}{\sigma^2}|X}(w|-1)}+\log\frac{p_{\Phi|X}(z|+1)}{p_{\Phi|X}(z|-1)}\overset{(b)}{=}\log\frac{p_{\frac{2Y}{\sigma^2},\Phi|X}(w,z|+1)}{p_{\frac{2Y}{\sigma^2},\Phi|X}(w,z|-1)}\overset{(c)}{=}\log\frac{p_{X|\frac{2Y}{\sigma^2},\Phi}(+1|w,z)}{p_{X|\frac{2Y}{\sigma^2},\Phi}(-1|w,z)},$$

where $(a)$ comes from the definition of $w$ and $z$ in Lemma 5.8, $(b)$ from the independence of $Y$ and $\Phi$ when $X$ is given, and where $(c)$ is the Bayes rule using $p_X(+1)=p_X(-1)=\frac{1}{2}$. Therefore,

$$\tanh(\frac{w+z}{2})=\frac{1-\mathrm{e}^{-w-z}}{1+\mathrm{e}^{-w-z}}=\frac{p_{X|\frac{2Y}{\sigma^2},\Phi}(+1|w,z)-p_{X|\frac{2Y}{\sigma^2},\Phi}(-1|w,z)}{p_{X|\frac{2Y}{\sigma^2},\Phi}(+1|w,z)+p_{X|\frac{2Y}{\sigma^2},\Phi}(-1|w,z)}=\mathbb{E}[X|y(Y)=w,\Phi=z]. \tag{5.5}$$

This "soft bit" is a bit estimate in the $D$-domain and Eq. (5.5) is in fact a well-known relationship. Observe now that $1-(\tanh(\frac{w+z}{2}))^2 = \frac{1}{(\cosh(\frac{w+z}{2}))^2}$, therefore

$$I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) = \mathrm{e}^{-z}\frac{1-\int_{-\infty}^{\infty}\mathsf{c}(w)(\tanh(\frac{w+z}{2}))^2\mathrm{d}w}{1-\int_{-\infty}^{\infty}\mathsf{c}(w)(\tanh(\frac{w}{2}))^2\mathrm{d}w} = \mathrm{e}^{-z}\frac{1-\mathbb{E}[(\tanh(\frac{(2Y)/(\sigma^2)+z}{2}))^2|X=+1]}{1-\mathbb{E}[(\tanh(\frac{Y}{\sigma^2})^2|X=+1]},$$

and the claim follows because, as discussed in Eq. (5.4), we can drop in the last expression the conditioning on $X=+1$.
Second, the kernel is in general not unique in the $L$-domain and we can use this degree of freedom to get equivalent kernels, see Definition 5.3. Denote $f(z) \triangleq \frac{1-\mathbb{E}[\mathbb{E}[X|Y,\Phi=-z]^2]}{1-\mathbb{E}[\mathbb{E}[X|Y]^2]}$ and observe that $I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) = \exp(-z)f(z)$ with this notation. For any symmetric density $\mathsf{a}(z)$, the function $I^{'\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) \triangleq f(-z)$ is also a valid kernel for the $L$-domain since $\int_{-\infty}^{+\infty}\mathsf{a}(z)\mathrm{e}^{-z}f(z)\mathrm{d}z = \int_{-\infty}^{+\infty}\mathsf{a}(z)f(-z)\mathrm{d}z$. Therefore, we get the equivalent kernel

$$I^{'\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) = \frac{1-\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]^2]}{1-\mathbb{E}[\mathbb{E}[X|Y]^2]} = \left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{(\cosh(\frac{w+z}{2}))^2}\mathrm{d}w\right)\Big/\left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{(\cosh(\frac{w}{2}))^2}\mathrm{d}w\right).$$

(iii) For any symmetric random variable $L$, a straightforward exercise shows that $\mathbb{E}[\tanh(L/2)] = \mathbb{E}[(\tanh(L/2))^2]$. See, e.g, [31, 41]. Applied to the random variable $y(Y) = \log\frac{p(Y|+1)}{p(Y|-1)} = \frac{2}{\sigma^2}Y$ that is symmetric given $X=+1$, this gives us

$$\begin{aligned}\mathbb{E}[\mathbb{E}[X|Y]^2] = \mathbb{E}[\tanh(y(Y)/2)^2] &\overset{(5.4)}{=} \mathbb{E}[\tanh(y(Y)/2)^2|X=+1]\\ &= \mathbb{E}[\tanh(y(Y)/2)|X=+1] = \mathbb{E}[\mathbb{E}[X|Y]|X=+1].\end{aligned}$$

Therefore the denominator of $I^{'\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)$ can be easily written as $1-\mathbb{E}[\mathbb{E}[X|Y]^2] = 1-\mathbb{E}[\mathbb{E}[X|Y]|X=+1]$. We cannot use directly this argument for the term $\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]^2] = \mathbb{E}[\tanh(\frac{Y}{\sigma^2}+\frac{z}{2})^2]$ at the numerator (the random variable $\frac{2}{\sigma^2}Y+z$ being not symmetric). However, we can look for an *equivalent* kernel. This is easily done by observing that the values $z$ can be provided by the *symmetric* random variable $\Phi$ given $X=+1$. The sum of two symmetric random variables is again symmetric (see Chapter 2), therefore $\hat{y}(Y,\Phi) \triangleq \frac{2}{\sigma^2}Y+\Phi$ is a symmetric random variable given $X=+1$. As above, we can now use the fact that $\mathbb{E}[\tanh(\hat{y}(Y,\Phi)/2)|X=+1] \overset{(5.4)}{=} \mathbb{E}[(\tanh(\hat{y}(Y,\Phi)/2))^2|X=+1]$ to obtain $\mathbb{E}[\mathbb{E}[X|Y,\Phi]^2] = \mathbb{E}[\mathbb{E}[X|Y,\Phi]|X=+1]$. Therefore,

$$I^{''\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) = \frac{1-\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]|X=+1]}{1-\mathbb{E}[\mathbb{E}[X|Y]|X=+1]} = \left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{1+\mathrm{e}^{w+z}}\mathrm{d}w\right)\Big/\left(\int_{-\infty}^{+\infty}\frac{\mathrm{e}^{-\frac{(w\sigma^2-2)^2}{8\sigma^2}}}{1+\mathrm{e}^{w}}\mathrm{d}w\right)$$

is an equivalent kernel (but pointwise different from $I^{\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)$ and $I^{'\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)$). The last equality comes from the fact that $1-\mathbb{E}[X|Y=y,\Phi=z] = 1-\tanh(\frac{y+z}{2}) = \frac{2}{1+\mathrm{e}^{y+z}}$. □

One remark about the previous lemma and its proof is in order. Observe that $I^{''\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z)$ uses the conditional expectation $\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]|X=+1]$. By channel symmetry, we have $\mathbb{E}[\mathbb{E}[X|Y,\Phi=z]|X=+1] = \mathbb{E}[\tanh(\frac{Y}{\sigma^2}+z)|X=+1] = \mathbb{E}[\tanh(-\frac{Y}{\sigma^2}+z)|X=-1] = \mathbb{E}[X\tanh(\frac{Y}{\sigma^2}+zX)] = \mathbb{E}[X\mathbb{E}[X|Y,\Phi=zX]]$. Now, using the kernel equivalencies to replace the conditioning $\Phi=zX$ by $\Phi=z$, we obtain the equivalent kernel

$$I^{'''\mathsf{c}_{\mathrm{BAWGNC}(\mathfrak{h})}}(z) = \frac{1-\mathbb{E}[X\mathbb{E}[X|Y,\Phi=z]]}{1-\mathbb{E}[X\mathbb{E}[X|Y]]},$$

where the conditioning $X=+1$ has been dropped. In fact, this last expression can also be proved directly by using the form (ii), the kernel equivalencies and the relationship $\mathbb{E}[X\mathbb{E}[X|Y]] = \mathbb{E}[\mathbb{E}[X|Y]^2]$ that comes from the definition of the conditional expectation.

GEXIT and EXIT curves are in general very similar. The next lemma illuminates this fact: it shows that, in the limit of small SNR, the kernel for the BAWGNC behaves similarly to the kernel for the BSC discussed in Example 5.2.

**Lemma 5.9** [Limiting Behavior of GEXIT Kernel] Consider the family $\{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}\}$ of BAWGN channels, where $\mathtt{h}$ denotes the channel entropy: The additive noise $N$ in the model $Y = X + N$ is Gaussian with zero-mean and variance $\sigma^2$. Then

$$\text{(i)} \lim_{\sigma\to\infty} |d|^{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}}(s) = 1 - s^2, \qquad \text{(ii)} \quad \lim_{\sigma\to 0} |d|^{\mathsf{c}_{\text{BAWGNC}(\mathtt{h})}}(s) = 1.$$

In the $|D|$-domain, the kernels are ordered between those two extremal functions.

*Proof.* First recall the transform formula (5.2) and $2\tanh^{-1}(s) = \log\frac{1+s}{1-s}$.
(i) Characterization (iii) of Lemma 5.8 shows that $I^{\mathsf{c}}(2\tanh^{-1}(s)) = \frac{1-\int_{-\infty}^{+\infty}\mathsf{c}(l)\tanh(l/2+\tanh^{-1}(s))\mathrm{d}l}{1-\int_{-\infty}^{+\infty}\mathsf{c}(l)\tanh(l/2)\mathrm{d}l}$. Let us restrict ourselves to the study of the term $I_\sigma(s) \triangleq \int_{-\infty}^{+\infty}\mathsf{c}(l)\tanh(l/2+\tanh^{-1}(s))\mathrm{d}l$. When $\sigma^2\to\infty$, then the distribution of the channel inputs in the $L$-domain $\mathsf{c}(l) = \frac{\sigma}{2\sqrt{2\pi}}\exp(-\frac{\sigma^2(l-2/\sigma^2)^2}{8})$ becomes a Dirac centered in 0 (since its variance $4/\sigma^2\to 0$). For any function continuous in 0, e.g., for the function $k_s : l \mapsto \tanh(l/2+\tanh^{-1}(s))$, one can indeed replace, without committing much error when $\sigma^2\to\infty$, the integral $\int_{-\infty}^{+\infty}\mathsf{c}(l)k_s(l)\mathrm{d}l$ by $\int_{-\infty}^{+\infty}\mathsf{c}(l)k_s(0)\mathrm{d}l$. See, e.g., [149] for further details. Therefore $I_\sigma(z) \underset{\sigma\to\infty}{\longrightarrow} \tanh(0/2+\tanh^{-1}(s)) = s$. Using Eq. (5.2), we get $|d|^{\mathsf{c}}(s) = \frac{1-s}{2}\frac{1+s}{1} + \frac{1+s}{2}\frac{1-s}{1} = 1-s^2$.
(ii) The case $\sigma\to 0$ corresponds to the full knowledge of the channel input. The kernel in the $|D|$-domain converges pointwise to 1. In this case $\mathsf{c}(l)$ becomes a "Dirac at infinity" and a similar argument as for (i) can be applied.
Finally, observe that, for a fixed $\mathtt{h}\in(0,1)$, the kernels in the $|D|-$domain are ordered because of Theorem 5.2 and the fact that the Gaussian family is ordered. □

So far we have restricted ourselves to the case of binary inputs. But the non-binary case is not much harder. This is presented in Lemma 5.10.

**Lemma 5.10** [AWGN($\mathtt{h}$)] Consider a length $n$ code. Assume transmission takes place over a family $\{AWGNC(\mathtt{h}_i)\}_{i\in[n]}$ where there is a global parameter $\epsilon$ such that $\mathtt{h}_i(\epsilon) = \mathtt{h}(\epsilon)$ is the entropy associated with the $i^{\text{th}}$ channel for all $i\in[n]$. Let this parameter be $\epsilon = -2\mathsf{snr} \triangleq -\frac{2}{\sigma^2}$. Then

$$g_i^{\text{MAP}}(\mathtt{G},\epsilon) = \mathbb{E}\left[\mathbb{E}[X_i^2|Y] - \mathbb{E}[X_i|Y]^2\right].$$

In other words, the derivative of the conditional entropy with respect to the particular parameter $\epsilon$ is equal to the minimum mean-square error estimator.

*Proof.* We will prove the result in general settings when the input alphabet $\mathcal{X}$ can be any subset of $\mathbb{R}$. Temporarily, let $\tilde{Y} = X + \tilde{N}$ represent our running Gaussian channel model. $\tilde{N}$ is the additive white Gaussian noise with zero-mean and variance $\sigma^2$. Now let us normalize this model by $\sigma^2$ to obtain the equivalent model $Y = \sqrt{\mathsf{snr}}X + N$ where $\mathsf{snr} = \frac{1}{\sigma^2}$ and $N$ is an additive white Gaussian noise with zero-mean and unit-variance. In order to be a sufficient statistic, the extrinsic MAP estimate $\phi_i = \phi_i(y_{\sim i})$ can no longer be a log-likelihood ratio but, in general, a function of $x_i$, i.e., $\phi_i : x \mapsto \phi_i(y_{\sim i}, x)$. Using Lemma 5.7 it follows that

$$g_i^{\text{MAP}}(\epsilon) = \int_{\phi_i,y_i,x_i} p(x_i)p(\phi_i\,|\,x_i)\frac{\mathrm{d}}{\mathrm{d}\epsilon}p(y_i|x_i)\cdot\log\left(\int_{x_i'}\frac{p(x_i'|\phi_i)p(y_i|x_i')}{p(x_i|\phi_i)p(y_i|x_i)}\mathrm{d}x_i'\right)\mathrm{d}x_i\mathrm{d}y_i\mathrm{d}\phi_i.$$

To simplify the computations, a few remarks are in order. First recall that we have chosen $\epsilon$ to be $\epsilon = -2\mathsf{snr} = \frac{-2}{\sigma^2}$. Second, observe that the Gaussian density permits us to write $\frac{\mathrm{d}p(y_i|x_i)}{\mathrm{d}\epsilon} = \frac{x_i}{\sqrt{\mathsf{snr}}}\frac{\mathrm{d}}{\mathrm{d}y_i}p(y_i|x_i)$.

Therefore, integrating by parts with respect to $y_i$, we get

$$g_i^{\text{MAP}}(\epsilon) = \int_{\phi_i,y_i,x_i} p(x_i)p(\phi_i\,|\,x_i)\frac{x_i}{\sqrt{\mathsf{snr}}}p(y_i|x_i)\cdot\frac{\mathrm{d}}{\mathrm{d}y_i}\left\{\log\left(\int_{x_i'}\frac{p(x_i'|\phi_i)p(y_i|x_i')}{p(x_i|\phi_i)p(y_i|x_i)}\mathrm{d}x_i'\right)\right\}\mathrm{d}x_i\mathrm{d}y_i\mathrm{d}\phi_i$$
$$= -\int_{\phi_i,y_i,x_i} p(x_i)p(\phi_i\,|\,x_i)\frac{x_i}{\sqrt{\mathsf{snr}}}p(y_i|x_i)\cdot\frac{\int_{x_i'}\sqrt{\mathsf{snr}}(x_i'-x_i)p(x_i'|\phi_i)p(y_i|x_i')\mathrm{d}x_i'}{\int_{x_i'}p(x_i'|\phi_i)p(y_i|x_i')\mathrm{d}x_i'}\mathrm{d}x_i\mathrm{d}y_i\mathrm{d}\phi_i,$$

after having used $\frac{\mathrm{d}p(y_i|x_i')}{\mathrm{d}y_i} = \frac{\mathrm{d}p_{Z_i}(y_i-\sqrt{\mathsf{snr}}x_i')}{\mathrm{d}y_i} = -(y_i-\sqrt{\mathsf{snr}}x_i')p(y_i|x_i')$. Let us now re-order as $p(x_i'|\phi_i)p(y_i|x_i') = p(x_i'|\phi_i,y_i)p(y_i|\phi_i)$ and use (with a slight abuse of notation) $\frac{y_i+\phi_i}{\sqrt{\mathsf{snr}}} = \mathbb{E}_{X_i}[X_i|\phi_i,y_i]$ to get

$$g_i^{\text{MAP}}(\epsilon) = -\int_{\phi_i,y_i,x_i} p(x_i)p(\phi_i\,|\,x_i)x_i p(y_i|x_i)\cdot\frac{p(y_i|\phi_i)(\frac{(y_i+\phi_i)}{\sqrt{\mathsf{snr}}}-x_i)}{p(y_i|\phi_i)}\mathrm{d}x_i\mathrm{d}y_i\mathrm{d}\phi_i$$
$$= \int_{\phi_i,y_i} p(\phi_i,y_i)\cdot\int_{x_i} p(x_i|y_i,\phi_i)\left(x_i^2-\frac{(y_i+\phi_i)x_i}{\sqrt{\mathsf{snr}}}\right)\mathrm{d}x_i\mathrm{d}y_i\mathrm{d}\phi_i$$
$$= \int_{\phi_i,y_i} p(\phi_i,y_i)\cdot\left(\mathbb{E}_{X_i}\left[X_i^2|\phi_i,y_i\right]-\mathbb{E}_{X_i}\left[X_i|\phi_i,y_i\right]^2\right)\mathrm{d}y_i\mathrm{d}\phi_i.$$

This concludes our proof since $\Phi_i$ is a sufficient statistic for $Y_{\sim i}$. □

Discussion: Imagine now we are considering binary inputs, then $\mathbb{E}[X_i^2|Y]=1$. In this case, using the fact that for a measurable and even function $f(y)$ we have $\mathbb{E}_Y[f(Y)] = \mathbb{E}_{Y|X=1}[f(Y)]$, it is possible to use Lemma 5.10 to re-derive Lemma 5.8.

A standard relationship, called de Bruijn's identity, is equivalent to the above connection between conditional entropy and MMSE for the Gaussian case. This is shown in the next section.

## 5.D A Long History of Gaussian Channels

The connection between minimum mean-square error and mutual information over Gaussian channels is due to [40, 148, 150]. This observation is pleasing (and somewhat surprising) because it connects a quantity well-used in detection theory to a fundamental information-theoretic measure – two notions which are a priori independent. The result has motivated further research, see, e.g., [42].
In hindsight, it is interesting to note that the relationship presented in [148] is – together with several alternative formulations, see [41, 151–155] – equivalent to the de Bruijn's identity (meaning one can prove one from the other one, or vice versa).
A formal treatment can be found in [154], see also [156]. For appropriately well-behaved functions (see, e.g., Section 2.9 ), let us show, as in [150], how the derivation of the relationship between minimum mean-square error and mutual information derive from the de Bruijn's identity.
Refer to [55, pp. 494-496] or [151]. Recall that de Bruijn's identity between entropy and Fisher information can be stated as follows.

**Lemma 5.11** [de Bruijn's Identity] Consider the channel $Y = X + Z$ where the Gaussian noise has zero-mean and variance $\sigma^2$. Assume $X$ has finite variance. Then $\frac{\mathrm{d}H(Y)}{\mathrm{d}\sigma} = \frac{1}{2}J(Y)$, where $J(Y) \triangleq \int f(y)\left(\frac{\frac{\mathrm{d}f(y)}{\mathrm{d}y}}{y}\right)^2 \mathrm{d}y$ is the Fisher information associated with a random variable $Y$ with density $f(y)$.

Let us now show how to get the desired relationship from the de Bruijn's identity.
First, observe that $I(X;Y) = H(X) - H(X|Y)$ and $I(X;Y) = H(Y) - H(Y|X) = H(Y) - H(Z)$ imply

$$\frac{\mathrm{d}H(X|Y)}{\mathrm{d}(\sigma^2)} = \frac{\mathrm{d}H(Z)}{\mathrm{d}(\sigma^2)} - \frac{\mathrm{d}(Y)}{\mathrm{d}(\sigma^2)} \tag{5.6}$$

where $\frac{\mathrm{d}H(Z)}{\mathrm{d}(\sigma^2)} = \frac{1}{2\sigma^2}$ since $H(Z) = \frac{1}{2}\log(2\pi\mathrm{e}\sigma^2)$. Without loss of generality and to make the connection with the expressions of this thesis, let us further assume that $X$ is a binary random variable with equal

priors. Then $f(y) = \frac{1}{2}p_Z(y|-1) + \frac{1}{2}p_Z(y|+1)$, such that $\frac{\mathrm{d}f(y)}{\mathrm{d}y} = \frac{-1-y}{2\sigma^2}p_Z(y|-1) + \frac{1-y}{2\sigma^2}p_Z(y|+1)$, and $\frac{p_Z(y|1)}{p_Z(y|-1)} = \exp(\frac{2y}{\sigma^2})$. We get

$$\sigma^2 \cdot \frac{\frac{\mathrm{d}f(y)}{\mathrm{d}y}}{f(y)} = \big((-1-y) + (1-y)\mathrm{e}^{\frac{2y}{\sigma^2}}\big)/\big(1+\mathrm{e}^{\frac{2y}{\sigma^2}}\big) = \tanh(y/(\sigma^2)) - y,$$
$$= \mathbb{E}[X|Y=y] - y = \mathbb{E}[X-Y|Y=y],$$

where the last equality has been shown, e.g., in Eq. (5.5). It remains to estimate the Fisher information

$$\begin{aligned} J(Y) &= \frac{1}{\sigma^4}\mathbb{E}\big[\mathbb{E}[X-Y|Y]^2\big] \overset{(a)}{=} \frac{1}{\sigma^4}\mathbb{E}\big[\mathbb{E}[X|Y]^2 - 2XY + Y^2\big] \\ &= \frac{1}{\sigma^4}\left(\mathbb{E}[\mathbb{E}[X|Y]^2] + \sigma^2 - \mathbb{E}[X^2]\right) = \frac{1}{\sigma^2} - \frac{1}{\sigma^4}\mathbb{E}[X^2 - \mathbb{E}[X|Y]^2], \\ &\overset{(b)}{=} \frac{1}{\sigma^2} - \frac{1}{\sigma^4}\mathbb{E}[(X-\mathbb{E}[X|Y])^2], \end{aligned} \tag{5.7}$$

where $(a)$ uses the fact that $\mathbb{E}[Y\mathbb{E}[X|Y]] = \mathbb{E}[XY]$ by definition of the conditional expectation, and $(b)$ uses the fact that $\mathbb{E}[\mathbb{E}[X|Y]^2] = \mathbb{E}[X\mathbb{E}[X|Y]]$ by again definition of the conditional expectation. Since $X^2 = 1 = \mathbb{E}[X^2|Y]$ in the binary case, Eq. (5.7) gives the (binary) GEXIT kernel. Finally, with Lemma 5.11 and Eq. (5.6), we obtain $\frac{\mathrm{d}H(X|Y)}{\mathrm{d}(\sigma^2)} = \frac{1}{2\sigma^2} - \frac{1}{2}J(Y) = \frac{1}{2\sigma^4}\mathbb{E}[\mathbb{E}[(X-\mathbb{E}[X|Y])^2]$. If we take the parameter $\mathsf{snr} = \frac{1}{\sigma^2}$ ($\mathsf{snr}$ being in this case the associated signal to noise ratio), we get $\frac{\mathrm{d}H(X|Y)}{\mathrm{d}\mathsf{snr}} = -\frac{1}{2}\mathbb{E}[(X-\mathbb{E}[X|Y])^2]$, where the right-hand side term $\mathbb{E}[(X-\mathbb{E}[X|Y])^2] = \mathbb{E}[\mathbb{E}[X^2|Y] - \mathbb{E}[X|Y]^2]$ is the minimum mean-square error. See also [70].

**Overview:** The Maxwell construction is extended to BMS channels. Unfortunately many interesting questions are left open.

# 6 MAP versus BP for Memoryless Symmetric Channels

As it is shown in the previous chapter, GEXIT functions share many properties with EXIT functions. It is therefore natural to ask if the connection between MAP and BP decoding also carries over to the more general context of memoryless symmetric channels. The upper bound on the MAP threshold presented in Chapter 4 extends in a simple way to this framework. Furthermore, we will see that a Maxwell construction holds in general if we look at a suitable *EBP GEXIT curve.*
This chapter deals with transmission over BMSC($\mathtt{h}$), where $\mathtt{h}$ denotes the channel entropy. The channel family is in general assumed to be smooth, ordered and complete.

## 6.1 Asymptotic GEXIT Functions

Let $\mathtt{C}$ be a binary linear code of length $n$. Assume that we choose a codeword $X$ uniformly at random from $\mathtt{C}$. Let $Y(\mathtt{h})$ be the received vector when transmission takes place over a smooth and ordered family $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}_{\mathtt{h}}$. Let $\mathtt{G}$ be a (fixed) graphical representation of the code and consider the BP schedule described in Section 2.5. Assume that we use the extrinsic BP estimate at the $\ell^{\text{th}}$ iteration, i.e., consider $\phi_i^{\text{BP},\ell}(Y_{\sim i})$. Define the $i^{\text{th}}$ BP GEXIT function at iteration $\ell$ to be $g_i^{\text{BP},\ell} \triangleq \frac{\partial}{\partial \mathtt{h}_i} H(X_i|Y_i, \phi_i^{\text{BP},\ell}(Y_{\sim i}))$ (see Definition 5.2). By analogy with Section 4.1, we state that

$$g_i^{\text{MAP}}(\mathtt{h}) \triangleq \frac{\partial}{\partial \mathtt{h}_i} H(X_i|Y_i, \phi_i^{\text{MAP}}(Y_{\sim i})) = \frac{\partial}{\partial \mathtt{h}_i} H(X_i|Y) \leq \frac{\partial}{\partial \mathtt{h}_i} H(X_i|Y_i, \phi_i^{\text{BP},\ell}(Y_{\sim i})) = g_i^{\text{BP},\ell}(\mathtt{h}).$$

Although the above inequality in the setting of EXIT functions treated in Section 4.1 is quite intuitive, its above counterpart for GEXIT functions requires a slightly more elaborate argument. This is shown in Theorem 5.2. For the BEC this inequality is the first step for showing the fundamental connection between MAP and BP decoding that appears in the asymptotic limit of large blocklengths when considering sparse graph codes. We follow a similar path as in Chapter 4 and turn our attention to the (average) performance of such large graphs.

**Definition 6.1** [(MAP) GEXIT Function over BMSC($\mathtt{h}$)] Assume that transmission takes place over a smooth family $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}_{\mathtt{h}}$. The MAP GEXIT function associated with the dd pair $(\lambda, \rho)$ is defined as

$$g^{\text{MAP}}(\mathtt{h}) \triangleq \limsup_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)} \left[ \frac{1}{n} \sum_{i=1}^{n} \frac{\partial H(X_i|Y_i(\mathtt{h}_i), \Phi_i^{\text{MAP}}(\mathtt{h}_{\sim i}))}{\partial \mathtt{h}_i} \Big|_{\mathtt{h}_1=\mathtt{h},\cdots,\mathtt{h}_n=\mathtt{h}} \right],$$

where the expectation is over instances of graph $\mathtt{G}$ taken uniformly at random from LDPC$(n,\lambda,\rho)$, $X$ denotes a codeword chosen uniformly at random from $\mathtt{G}$, $Y(\mathtt{h})$ is the result of transmitting $X$ over BMSC$(\mathtt{h})$, and $\Phi_i^{\text{MAP}}(\mathtt{h}_{\sim i}) = \phi_i^{\text{MAP}}(Y_{\sim i})$ is the $i^{\text{th}}$ extrinsic MAP estimate.

Discussion: Similar observations as in Section 4.1 are in order. First we can also write $g^{\text{MAP}}(\mathtt{h}) = \limsup_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\left[g_1^{\text{MAP}}(\mathtt{h})\right]$ since the quantity is averaged over all graphs in LDPC$(n,\lambda,\rho)$. Moreover, we consider the average (over graphs from a given ensemble) of all GEXIT functions. This is justified in Appendix 4.A where we show that both, entropy rate (via Theorem 4.3) and MAP GEXIT function (via Theorem 4.4), concentrate around their average. Finally, note that we use the lim sup, instead of the ordinary limit, in order to work with a well-defined limiting object. Proving the existence of the limit seems to be a difficult task. As discussed in Chapter 4, i.e., even in the simple case of transmission over the erasure channel, the existence of the corresponding limit is not known, in general, but only follows from the explicit construction of the Maxwell decoder in all the cases where the Maxwell construction can be shown to result in MAP performance.

**Definition 6.2** [BP EXIT Function over BEC$(\epsilon)$] Assume that transmission takes place over a smooth family $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}_{\mathtt{h}}$. The BP GEXIT function associated with the dd pair $(\lambda,\rho)$ is defined as

$$g^{\text{BP}}(\mathtt{h}) \triangleq \lim_{\ell\to\infty}\lim_{n\to\infty} \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\left[\frac{1}{n}\sum_{i=1}^{n} \frac{\partial H(X_i|Y_i(\mathtt{h}_i), \Phi_i^{\text{BP},\ell}(\mathtt{h}_{\sim i}))}{\partial \mathtt{h}_i}\Big|_{\mathtt{h}_1=\mathtt{h},\cdots,\mathtt{h}_n=\mathtt{h}}\right],$$

where the expectation is over instances of graph $\mathtt{G}$ taken uniformly at random from LDPC$(n,\lambda,\rho)$, $X$ denotes a codeword chosen uniformly at random from $\mathtt{G}$, $Y(\mathtt{h})$ is the result of transmitting $X$ over BMSC$(\mathtt{h})$, and $\Phi_i^{\text{BP},\ell}(\mathtt{h}_{\sim i}) = \phi_i^{\text{BP},\ell}(Y_{\sim i})$ is the $i^{\text{th}}$ extrinsic BP estimate at iteration $\ell$.

Discussion: Contrary to $g^{\text{MAP}}$ the existence of the BP GEXIT function is well-established. This follows from density evolution, see (next) Theorem 6.1 (ii). The fact that the "average" has a practical meaning is also justified by Theorem 6.1 (i).

**Theorem 6.1** [Limit and Concentration of BP GEXIT Functions] Consider a dd pair $(\lambda,\rho)$ and the sequence $\{\text{LDPC}(n,\lambda,\rho)\}_n$. Assume that transmission takes place over a smooth family $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}_{\mathtt{h}}$. Choose an element $\mathtt{G}(n)$ of length $n$ uniformly at random in LDPC$(n,\lambda,\rho)$. Let $g^{\text{BP}(\mathtt{G}(n)),\ell}(\mathtt{h}) = \frac{1}{n}\sum_{i=1}^{n} g_i^{\text{BP}(\mathtt{G}(n)),\ell}(\mathtt{h})$ denote the associated (averaged) BP GEXIT function at iteration $\ell$. Then

(i) $\forall \xi > 0, \exists \alpha_\xi > 0, \exists N \in \mathbb{N}, \forall n > N,\ \Pr\left\{\left|g^{\text{BP}(\mathtt{G}(n)),\ell}(\mathtt{h}) - \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[g^{\text{BP}(\mathtt{G}(n)),\ell}(\mathtt{h})]\right| > n\xi\right\} \le \mathrm{e}^{-\alpha_\xi n}$,

(ii) The limits $g^{\text{BP},\ell}(\mathtt{h}) = \lim_{n\to\infty}\mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}[g^{\text{BP}(\mathtt{G}(n)),\ell}(\mathtt{h})]$ and $g^{\text{BP}}(\mathtt{h}) = \lim_{\ell\to\infty} g^{\text{BP},\ell}(\mathtt{h})$ exist.

*Proof.* (i) The proof of the concentration is along the same lines as the proof in [65], which shows the concentration of the probability of error under BP decoding, or the proof in Appendix 4.A, which relates to the concentration of the BP EXIT function. We will therefore skip the details.
(ii) Note that for a fixed iteration number $\ell$, the distribution of $\Phi_i^{\text{BP}(\mathtt{G}(n)),\ell}$ (where the node $i$ is chosen uniformly at random in $[n]$), assuming that the all-one codeword was sent, converges (at a speed of $1/n$) to the corresponding distribution of density evolution obtained from the corresponding spanning tree, denote it by $\mathsf{a}_\ell$. See, e.g., [65]. The result now follows by noting that $g^{\text{BP},\ell}$ is the result of applying a bounded linear operator to the distribution $\mathsf{a}_\ell$, see Lemma 5.2 and Section 2.9. □

For simple codes, such as single parity-check codes or repetition codes, EXIT or GEXIT functions are relatively easy to compute. Lemma 3.3 or Lemma 5.2 give an operational way to determine the quantities via the corresponding EXIT or GEXIT linear operators. In general though, it is not a trivial matter to determine the density of $\Phi_i^{\text{MAP}}$ required for the calculation. What we "can" easily compute in practice are the BP estimates. In the asymptotic limit the extrinsic BP estimates are obtained from density evolution, and $g^{\text{BP},\ell}$ and $g^{\text{BP}}$ have a convenient representation in terms of the asymptotic extrinsic BP densities. More precisely, the bounded linear operator of Lemma 5.2 shows that

$$g^{\text{BP},\ell}(\mathtt{h}) = \int_{-\infty}^{\infty} \mathsf{a}^{\text{BP},\ell}(z) I^{\mathsf{c}_{\text{BMSC}(\mathtt{h})}}(z)\,\mathrm{d}z, \qquad g^{\text{BP}}(\mathtt{h}) = \int_{-\infty}^{\infty} \mathsf{a}^{\text{BP}}(z) I^{\mathsf{c}_{\text{BMSC}(\mathtt{h})}}(z)\,\mathrm{d}z,$$

where $\mathsf{c}_{\mathrm{BMSC}(\mathtt{h})}$ is the channel density, and where $\mathsf{a}^{\mathrm{BP},\ell}$ is the limiting density of $\Phi_i^{\mathrm{BP}(\mathtt{G}(n)),\ell}$ (where the node $i$ is chosen uniformly at random in $[n]$) under the all-one codeword assumption as $n$ tends to infinity and averaged over $\mathrm{LDPC}(n,\lambda,\rho)$. This density can easily be computed by density evolution. In a similar manner, $\mathsf{a}^{\mathrm{BP}}$ denotes the corresponding fixed-point density of density evolution.

Figure 6.1 shows BP GEXIT functions for a sample of regular LDPC ensembles. They are compared with the corresponding BP EXIT functions. We see that the curves are quite similar.

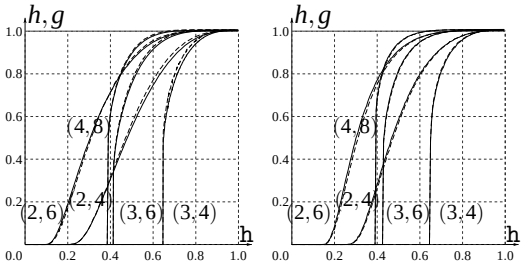



Figure 6.1: BP GEXIT (solid curves) versus BP EXIT (dashed curves) for several regular LDPC ensembles. Left: BSC($\mathtt{h}$). Right: BAWGNC($\mathtt{h}$).

## 6.2 Upper Bound on the MAP Threshold

Consider a complete and ordered family $\{\mathrm{BMSC}(\mathtt{h})\}_{\mathtt{h}}$ and a dd pair $(\lambda,\rho)$. Recall from Section 2.8 that the MAP threshold is defined as $\mathtt{h}^{\mathrm{MAP}} \triangleq \min\{\mathtt{h}\in[0,1] : \liminf_{n\to\infty} \mathbb{E}_{\mathtt{G}}[H(X\,|\,Y(\mathtt{h}))]/n > 0\}$. As discussed in Section 2.8, this definition captures the notion of threshold for the *bit* error probability. This (MAP) threshold is the value of the channel entropy $\mathtt{h}$ at which the considered GEXIT function becomes non-negative (see Lemma 5.3).

We now follow the method presented in Section 4.2.1 to derive an upper bound (which we conjecture to be tight in many cases) on the MAP threshold. We need two intermediate results in order to extend Lemma 4.4 to BMS channels. (i) The first one is of course the general area theorem. (ii) The second is the following asymptotic version of Theorem 5.2.

**Lemma 6.1** [Upper Bound $g^{\mathrm{MAP}} \le g^{\mathrm{BP}}$] Consider a dd pair $(\lambda,\rho)$ and transmission over a smooth and ordered family $\{\mathrm{BMSC}_i(\mathtt{h}_i=\mathtt{h})\}_{\mathtt{h}}$. Let $g^{\mathrm{MAP}}(\mathtt{h})$ and $g^{\mathrm{BP}}(\mathtt{h})$ denote the corresponding asymptotic MAP and BP GEXIT functions. Then $g^{\mathrm{MAP}}(\mathtt{h}) \le g^{\mathrm{BP}}(\mathtt{h})$.

*Proof.* From Theorem 5.2 we know that, for any $\mathtt{G}\in\mathrm{LDPC}(n,\lambda,\rho)$ and $\ell\in\mathbb{N}$, we have $g_{\mathtt{G}}^{\mathrm{MAP}}(\mathtt{h}) \le g_{\mathtt{G}}^{\mathrm{BP},\ell}(\mathtt{h})$. If we take first the expectation over the elements of the ensemble, then the lim sup on both sides with respect to $n$, and finally the limit $\ell\to\infty$, we get the desired result. □

**Theorem 6.2** [Upper Bound on MAP Threshold] Consider a dd pair $(\lambda,\rho)$ with design rate $r_{\lambda,\rho}$. Assume that transmission takes place over a complete and ordered smooth family $\{\mathrm{BMSC}_i(\mathtt{h}_i=\mathtt{h})\}_{\mathtt{h}}$. Let $g^{\mathrm{BP}}(\mathtt{h})$ denote the associated BP GEXIT function. Then $\liminf_{n\to\infty}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X|Y(\mathtt{h}))]/n \ge r_{\lambda,\rho} - \int_{\mathtt{h}}^{1} g^{\mathrm{BP}}(\mathtt{h}')\,\mathrm{d}\mathtt{h}'$. Furthermore, if $\overline{\mathtt{h}}$ denotes the largest positive scalar so that

$$\int_{\overline{\mathtt{h}}}^{1} g^{\mathrm{BP}}(\mathtt{h})\,\mathrm{d}\mathtt{h} = r_{\lambda,\rho},$$

then $\mathtt{h}^{\mathrm{MAP}} \le \overline{\mathtt{h}}$, where $\mathtt{h}^{\mathrm{MAP}}$ denotes the MAP threshold.

*Proof.* The asymptotic rate is potentially larger than the design rate. Therefore

$$r_{\lambda,\rho} - \liminf_{n\to\infty} \frac{\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X|Y(\mathtt{h}))]}{n} \le \limsup_{n\to\infty} \frac{1}{n}\mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X\,|\,Y(1)) - H_{\mathtt{G}}(X\,|\,Y(\mathtt{h}))]$$
$$\overset{\text{(i)}}{=} \limsup_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}\Big[\int_{\mathtt{h}}^{1} g_{\mathtt{G}}^{\mathrm{MAP}}(\mathtt{h}')\,\mathrm{d}\mathtt{h}'\Big],$$

where (i) is obtained from the general area theorem. We can exchange the expectation and the integral by Fubini's theorem since $g_{\mathtt{G}}^{\mathrm{MAP}}$ is measurable and bounded by 0 and 1. We can furthermore exchange the limit and the integral by the Fatou-Lebesgue lemma so that

$$\liminf_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[H(X|Y(\mathtt{h}))]/n \ge r_{\lambda,\rho} - \int_{\mathtt{h}}^{1} g^{\mathrm{MAP}}(\mathtt{h}')\,\mathrm{d}\mathtt{h}' \overset{\text{(ii)}}{\ge} r_{\lambda,\rho} - \int_{\mathtt{h}}^{1} g^{\mathrm{BP}}(\mathtt{h}')\mathrm{d}\mathtt{h}',$$

where (ii) follows from Lemma 6.1. It remains to show how to derive an upper bound on the MAP threshold. This follows from the observation that the right-hand side of the last inequality is non-decreasing in $\mathtt{h}$. Therefore $\limsup_{n\to\infty} \mathbb{E}_{\mathrm{LDPC}(n,\lambda,\rho)}[H_{\mathtt{G}}(X|Y(\mathtt{h}))]/n$ is bounded away from 0 for any $\mathtt{h} > \overline{\mathtt{h}}$. Combined with the definition of $\mathtt{h}^{\mathrm{MAP}}$, this concludes the proof. □

**Example 6.1** The following table presents the upper bounds on the MAP threshold for transmission over BSC($\mathtt{h}$) as derived from Theorem 6.2 for a few regular ensembles with dd pair $(\lambda(\mathtt{x}),\rho(\mathtt{x})) = (\mathtt{x}^{\mathtt{l}-1},\mathtt{x}^{\mathtt{r}-1})$. The corresponding thresholds were first computed using the (non-rigorous) replica method from statistical mechanics in [157]. In [31], they were shown to be upper bounds for $\mathtt{r}$ even, using an interpolation technique. The present proof applies also to the case of odd $\mathtt{r}$. It can be proved that the three characterizations of the threshold are indeed equivalent, i.e., they give *exactly* the same value.

| $\mathtt{l}$ | $\mathtt{r}$ | $\mathtt{h}^{\mathrm{BP}}$ | $\overline{\mathtt{h}}^{\mathrm{MAP}}$ | $\overline{\mathtt{h}}^{\mathrm{MAP}}$ ( [24, 158]) | $\mathtt{h}^{\mathrm{SH}}$ |
|---|---|---|---|---|---|
| 3 | 4 | 0.6507(5) | 0.7417(1) | 0.743231 | 3/4 |
| 3 | 5 | 0.5113(5) | 0.5800(3) | 0.583578 | 3/5 |
| 3 | 6 | 0.4160(5) | 0.4721(5) | 0.476728 | 1/2 |
| 4 | 6 | 0.5203(5) | 0.6636(2) | 0.663679 | 2/3 |

Also shown is the result of the information theoretic upper bound given in [24], which in turn is an improved version of the bound developed in [158]. For the specific case of transmission over BSC($\epsilon$) and regular LDPC ensembles this upper bound on the MAP threshold is given by $h_2(\overline{\epsilon})$, where $\overline{\epsilon}$ is the unique positive root of the equation $\mathtt{r}h_2(\epsilon) = \mathtt{l}h_2((1-(1-2\epsilon)^{\mathtt{r}})/2)$.

## 6.3 Maxwell Construction and EBP GEXIT Curve

As discussed in Chapter 4 for the case of transmission over the BEC, the fundamental relationship that appears in the limit of large blocklengths between the MAP and the BP decoder is best described in terms of the *Extended* BP (EBP) EXIT curve. For the BEC this curve is given in parametric form by $\Big(\frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))}, \Lambda(1-\rho(1-\mathtt{x}))\Big)$, where $\mathtt{x}$ takes values in a union of a finite number of intervals $I \subseteq [0,1]$ such that $\mathtt{x} \le \lambda(1-\rho(1-\mathtt{x}))$, see, e.g., Lemma 4.8. Such an explicit characterization is in general not available for non-trivial BMS channels.

### 6.3.1 EBP GEXIT Curve

The families $\{\mathtt{f}_{\mathtt{x}}\}_{\mathtt{x}} \triangleq \{\mathrm{BEC}(\mathtt{x})\}_{\mathtt{x}}$ and $\{\mathtt{c}_{\mathtt{x}}\}_{\mathtt{x}} \triangleq \{\mathrm{BEC}(\frac{\mathtt{x}}{\lambda(1-\rho(1-\mathtt{x}))})\}_{\mathtt{x}}$, $\mathtt{x} \in I$, have the property that, for each $\mathtt{x} \in I$, $\mathtt{f}_{\mathtt{x}}$ constitutes a fixed-point density (of density evolution) for the channel $\mathtt{c}_{\mathtt{x}}$. Furthermore, both channel families are *smooth* and satisfy $\mathtt{H}(\mathtt{f}_{\mathtt{x}}) = \mathtt{x}$. Moreover, if $\epsilon^{\mathrm{SC}} \triangleq \frac{1}{\lambda'(0)\rho'(1)} < 1$, then $I = [0,1]$ and the families are *complete* (i.e., $\mathtt{x}$ and $\mathtt{c}_{\mathtt{x}}$ describe the full range $[0,1]$).

**Definition 6.3** [Complete Fixed-Point Family] Consider a dd pair $(\lambda,\rho)$. The families $\{\mathsf{f}_\mathtt{x}\}_\mathtt{x}$ and $\{\mathsf{c}_\mathtt{x}\}_\mathtt{x}$, $\mathtt{x}\in[0,1]$, are said to form a *complete fixed-point family* for $(\lambda,\rho)$ if

(i) there exists a complete and ordered family $\{\mathrm{BMSC}(\mathtt{h})\}_\mathtt{h}$ such that $\forall\mathtt{x}\in[0,1]$, $\mathsf{c}_\mathtt{x}\in\{\mathrm{BMSC}(\mathtt{h})\}_\mathtt{h}$

(ii) for each $\mathtt{x}\in[0,1]$, $\mathsf{f}_\mathtt{x}$ is a fixed-point density with respect to the dd pair $(\lambda,\rho)$ and the channel $\mathsf{c}_\mathtt{x}$; this means that for each $\mathtt{x}\in[0,1]$, $\mathsf{f}_\mathtt{x}=\mathsf{c}_\mathtt{x}\circledast\lambda(\rho(\mathsf{f}_x))\triangleq\mathsf{c}_\mathtt{x}\circledast\sum_j\lambda_j\left(\sum_k\rho_k(\mathsf{f}_\mathtt{x})^{\boxplus(k-1)}\right)^{\circledast(j-1)}$

(iii) $\{\mathsf{f}_\mathtt{x}\}_\mathtt{x}$ and $\{\mathsf{c}_\mathtt{x}\}_\mathtt{x}$ are smooth with respect to $\mathtt{x}$

(iv) $\mathtt{H}(\mathsf{f}_\mathtt{x})=\mathtt{x}$.

The previous characterization of a complete fixed-point family permits us to define the EBP GEXIT curve in the general case.

**Definition 6.4** [EBP GEXIT Curve] Recall Definition 6.3. Let $\mathsf{a}_\mathtt{x}(y)\triangleq\Lambda(\rho(\mathsf{f}_\mathtt{x}))$. The EBP GEXIT curve is given in parametric form by $(\mathtt{h}(\mathtt{x}),g^{\mathrm{EBP}}(\mathtt{x}))\triangleq(\mathtt{H}(\mathsf{c}_\mathtt{x}),\mathtt{G}(\mathsf{c}_\mathtt{x},\mathsf{a}_\mathtt{x}))$, where $\mathtt{H}$ is the entropy operator and $\mathtt{G}$ the GEXIT operator.

Discussion: Several remarks are in order. First, notice that the function $g^{\mathrm{BP}}$ coincides a.e. with the "envelope" of the EBP GEXIT curve.
Second, notice that we have used $\mathtt{x}$ to parameterize the channel families and the function $g^{\mathrm{EBP}}(\mathtt{x})$ and we have assumed that $\mathtt{H}(\mathsf{f}_\mathtt{x})=\mathtt{x}$ (rather than $\mathtt{H}(\mathsf{c}_\mathtt{x})=\mathtt{x}$). The reason is that, in general, the EBP GEXIT curve is not a single-valued function of the *channel* entropy, but is a single-valued function of the *fixed-point* entropy. By using the parameter $\mathtt{x}$ (and not the channel entropy), we remind ourselves that the channel $\mathsf{c}_\mathtt{x}$ is the channel that belongs to the family of fixed-point densities $\{\mathsf{f}_\mathtt{x}\}$ (and not a channel $\mathsf{c}_\mathtt{h}$ defined uniquely by a fixed channel entropy). *Complete* fixed-point families do not always exist. If, for instance, $\lambda_2=0$, then $\mathtt{x}$ cannot be chosen arbitrarily close to 0. This is easily seen for transmission over the BEC because, in this case, the stability condition threshold is infinite.
Third, it is not immediately obvious that for a given dd pair $(\lambda,\rho)$ and a complete and ordered family $\{\mathrm{BMSC}(\mathtt{h})\}_\mathtt{h}$, a (complete or incomplete) fixed-point family always exists, or that it is unique. For the BEC we have an explicit formula for the family, but in the general case the existence is far from trivial. We will get back to this point in the sequel.
One important application of EBP GEXIT curves is that they encode the connection between MAP and BP decoding. As mentioned above, the BP GEXIT function is obtained as the "envelope" of the EBP GEXIT curve. More precisely, one has to choose, for each value of the channel entropy $\mathtt{h}$, the branch of the EBP curve whose GEXIT value is the largest, as stated in Theorem 4.1 for the BEC. In Chapter 4, we have seen many cases where the Maxwell function (i.e., the function obtained from the EBP GEXIT using the Maxwell construction, see Definition 4.5) is proved to coincide a.e. with the MAP GEXIT function.

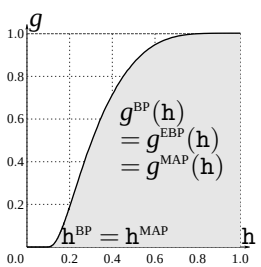



Figure 6.2: Cycle-code ensemble with dd pair $(x,x^5)$. The EBP GEXIT curve, BP GEXIT curve and MAP GEXIT curve coincide.

In a strictly similar way as in Chapter 4, we construct a *Maxwell function* from the EBP GEXIT curve for general BMS channels. We conjecture that the Maxwell function coincides with the MAP GEXIT functions a.e. for a general BMSC.
Let us first say that (beyond the simple BEC case in Chapter 4), we can further (almost) prove this conjecture in the following case. If the BP GEXIT does not jump, i.e., if it is a non-decreasing continuous function, then the BP and EBP curves are equal. (For any value of the channel entropy $\mathtt{h}$, a single fixed point density – apart from the "delta at infinity" – is found. Also: a single fixed point density exists for each value of the density entropy $\mathtt{x}$). Using Corollary 6.1 it is further possible to show that the BP, EBP, and Maxwell GEXIT curves in fact coincide. For example, consider the dd pair $(\lambda,\rho)=(x,x^5)$ and the corresponding LDPC ensemble with design rate $r=2/3$. Assume that transmission takes place over the family $\{\mathrm{BSC}(\epsilon)\}$. Recall that for this code the BP threshold is given by the stability condition. From Figure 6.2 we see that, according to the numerical calculation, the EBP GEXIT curve is a monotone function. It follows that the EBP GEXIT is equal to the BP GEXIT curve for this example.

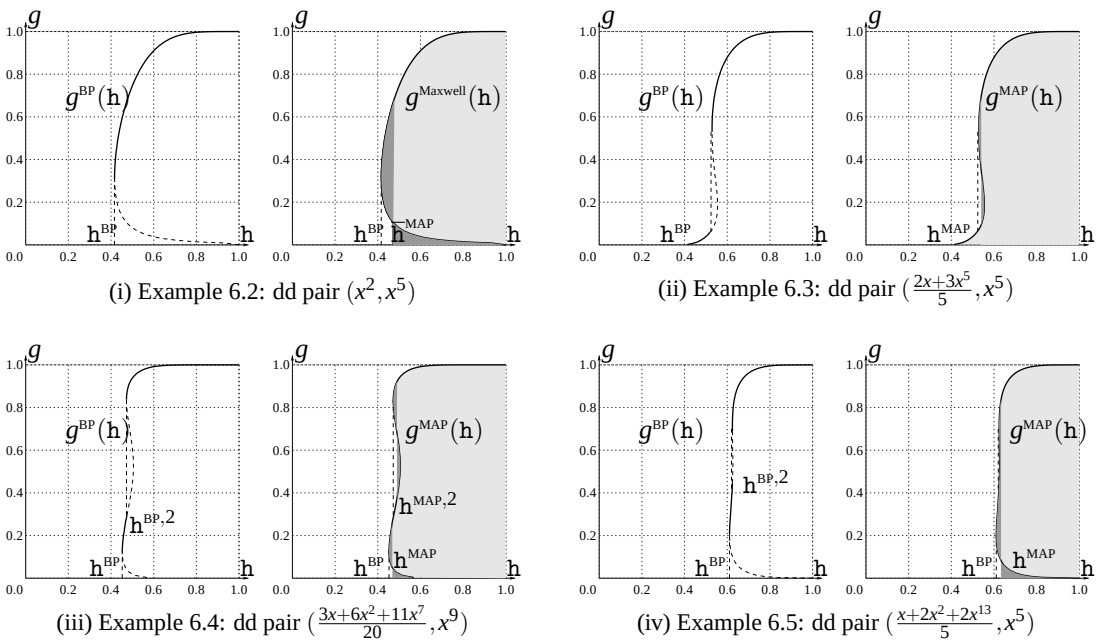
A few typical examples are presented below. In each of them the complete fixed-point family is computed via a *numerical* procedure explained in the next section.

**Example 6.2** [(3,6) LDPC Ensemble – BSC] Consider the dd pair $(\lambda,\rho)=(x^2,x^5)$ and the corresponding LDPC ensemble with design rate $r_{3,6}=1/2$. We assume that transmission takes place over the family $\{\mathrm{BSC}(\mathtt{h}(\epsilon))\}$. Figure 6.3 (i) shows on the left the EBP GEXIT curve and the corresponding BP GEXIT curve, which has one jump. The picture on the right shows the conjectured MAP GEXIT curve according to the Maxwell construction. For this ensemble, we have $\mathtt{h}^{\mathrm{BP}}\approx 0.416$. The (conjectured) MAP threshold implied by the Maxwell construction coincides with the upper bound provided by Theorem 6.2 that reads $\overline{\epsilon}^{\mathrm{MAP}}\approx 0.472$.

**Example 6.3** [LDPC$(2/5x+3/5x^5,x^5)$ – BSC] Consider the dd pair $(\lambda,\rho)=(2/5x+3/5x^5,x^5)$ and the corresponding LDPC ensemble with design rate $r_{\lambda,\rho}=4/9$. We assume that transmission takes place over the family $\{\mathrm{BSC}(\mathtt{h}(\epsilon))\}$. Figure 6.3 (ii) shows on the left the EBP GEXIT curve and the corresponding BP GEXIT curve, which has one jump. The picture on the right shows the conjectured MAP GEXIT curve according to the Maxwell construction. The BP threshold is given by the stability condition. As a consequence of this and our conjecture on the Maxwell function, we find $\mathtt{h}^{\mathrm{BP}}=\mathtt{h}^{\mathrm{MAP}}$ .

**Example 6.4** [LDPC$(\frac{3x+6x^2+11x^{17}}{20},x^9)$ – BSC] Consider the dd pair $(\frac{3x+6x^2+11x^{17}}{20},x^9)$. We assume that transmission takes place over the family $\{\mathrm{BSC}(\mathtt{h}(\epsilon))\}$. Figure 6.3 (iii) shows on the left the EBP GEXIT curve and the corresponding BP GEXIT curve (with two jumps). The picture on the right shows the conjectured MAP GEXIT curve (with two jumps) according to the Maxwell construction.

**Example 6.5** [$(\frac{x+2x^2+2x^{13}}{5},x^5)$ – BSC] Consider the dd pair $(\frac{x+2x^2+2x^{13}}{5},x^5)$ and the corresponding LDPC ensemble. We assume that transmission takes place over the family $\{\mathrm{BSC}(\mathtt{h}(\epsilon))\}$. Figure 6.3 (iv) shows on the left the EBP GEXIT curve and the corresponding BP GEXIT curve (with two jumps). The picture on the right shows the conjectured MAP GEXIT curve (with one jump) according to the Maxwell construction.




(i) Example 6.2: dd pair $(x^2,x^5)$

(ii) Example 6.3: dd pair $(\frac{2x+3x^5}{5},x^5)$

(iii) Example 6.4: dd pair $(\frac{3x+6x^2+11x^7}{20},x^9)$

(iv) Example 6.5: dd pair $(\frac{x+2x^2+2x^{13}}{5},x^5)$

Figure 6.3: Examples of LDPC ensembles over the BSC with the (conjectured) MAP GEXIT function. Left: BP and EBP GEXIT. Right: Maxwell GEXIT.

### 6.3.2 EBP Computation

Let us now explain how the EBP GEXIT curves of the previous examples have been computed. Notice that ordinary density evolution at a fixed initial channel parameter cannot be applied. First, EBP curves include "unstable branches" where the GEXIT curve is a decreasing function of the channel entropy. These branches are expected to correspond to locally unstable fixed point densities of the density evolution equations. This is confirmed analytically for the BEC, and numerically for a generic BMS channel. As a consequence, these fixed points cannot be approximated by iterating density evolution with a generic initial condition (if we consider a *static* channel). Moreover, for a fixed channel parameter, multiple locally stable fixed point densities might coexist. Therefore (if we consider a *static* channel) different initial conditions are required to achieve each of these densities by density evolution.

Different ways for constructing these initial conditions can be imagined. In this thesis, however, we overcome this issue by noticing that EBP GEXIT curves are naturally parameterized by the intermediate densities, and in particular by the *entropy of the fixed point density*. More precisely, consider a smooth and degraded family $\{\mathrm{BMSC}(\mathtt{h})\}$ and $\mathtt{x} \in [0,1]$. Then, we expect that there exists at most one value of the channel parameter $\mathtt{h} = \mathtt{h}(\mathtt{x})$ and one density $\mathsf{f}_{\mathtt{x}}$, such that $\mathtt{H}(\mathsf{f}_{\mathtt{x}}) = \mathtt{x}$ and $(\mathsf{c}_{\mathtt{x}} \triangleq \mathrm{BMSC}(\mathtt{h}(\mathtt{x})), \mathsf{f}_{\mathtt{x}})$ forms a pair of fixed point densities. This naturally suggests running density evolution *at fixed density entropy*. Let us denote by $T_{\mathtt{h}}$ the ordinary density evolution operator at fixed channel $\mathrm{BMSC}(\mathtt{h})$. Formally $T_{\mathtt{h}}(\mathsf{a}) \triangleq \mathsf{c} \circledast \lambda(\rho(\mathsf{a}))$ where $\mathsf{c} \triangleq \mathrm{BMSC}(\mathtt{h})$. For any $\mathtt{x} \in [0,1]$, we define the density evolution operator at fixed entropy $\mathtt{x}$, $R_{\mathtt{x}}$ as $R_{\mathtt{x}}(\mathsf{a}) \triangleq T_{\mathtt{h}(\mathtt{a},\mathtt{x})}(\mathsf{a})$ where $\mathtt{h}(\mathtt{a},\mathtt{x})$ is the solution of $\mathtt{H}(T_{\mathtt{h}}(\mathsf{a})) = \mathtt{x}$ if such a solution exists, otherwise $\mathtt{H}(T_{\mathtt{h}}(\mathsf{a}))$ is undefined. Since, for a given $\mathsf{a}$, the family $T_{\mathtt{h}}(\mathsf{a})$ is ordered by physical degradation, $\mathtt{H}(T_{\mathtt{h}}(\mathsf{a}))$ is a non-decreasing function of $\mathtt{h}$. Therefore the equation $H(T_{\mathtt{h}}(\mathsf{a})) = \mathtt{x}$ has at most one solution. Furthermore, since the channel family $\mathrm{BMSC}(\mathtt{h})$ is smooth, $\mathtt{H}(T_{\mathtt{h}}(\mathsf{a}))$ is continuous. Note that $\mathtt{H}(T_0(\mathsf{a})) = 0$, i.e., if the channel is noiseless, then the output density at a variable nodes is noiseless as well. Therefore, a necessary and sufficient condition for a solution $\mathtt{h}(\mathtt{a},\mathtt{x})$ to exist (assuming that the family $\{\mathrm{BMSC}(\mathtt{h})\}_{\mathtt{h}}$ is complete) is that $\mathtt{H}(T_1(\mathsf{a})) = \mathtt{H}(\lambda(\rho(\mathsf{a}))) \geq \mathtt{x}$.

Any fixed point of $R_{\mathtt{x}}$, i.e. any $\mathsf{f}$ such that $\mathsf{f} = R_{\mathtt{x}}(\mathsf{f})$, is also a fixed point of ordinary density evolution for $\mathrm{BMSC}(\mathtt{h})$ with $\mathtt{h} = \mathtt{h}(\mathsf{f},\mathtt{x})$, and corresponds to a point on the EBP GEXIT curve. Furthermore, if a sequence of densities such that $\mathsf{a}_{\ell+1} = R_{\mathtt{x}}(\mathsf{a}_\ell)$ converges (weakly) to a density $\mathsf{f}$, then $\mathsf{f}$ is a fixed point of $R_{\mathtt{x}}$, with entropy $\mathtt{x}$. This motivates the following numerical procedure.

(i) Set the initial condition $\mathsf{a}_0 \triangleq \mathrm{BMSC}(\mathtt{x})$.

(ii) For $\ell \geq 0$ compute $\mathsf{a}_{\ell+1} = R_{\mathtt{x}}(\mathsf{a}_\ell)$. (Possible implementations are based on sampling or on Fourier Transform. Due to the monotonicity of $\mathtt{H}(T_{\mathtt{h}}(\mathsf{a}_\ell))$ in $\mathtt{h}$, the value of $\mathtt{h}(\mathsf{a}_\ell,\mathtt{x})$ can be determined efficiently by bisection.)

(iii) The current estimate of the EBP GEXIT curve is given in parametric form by $(\mathtt{h}_\ell, g_\ell^{\mathrm{EBP}})$, where $\mathtt{h}_\ell \triangleq \mathtt{h}(\mathsf{a}_\ell,\mathtt{x})$ is the estimate of the channel entropy, and

$$g_\ell^{\mathrm{EBP}} \triangleq \mathtt{G}(\mathrm{BMSC}(\mathtt{h}_\ell), \mathsf{b}_\ell) = \int_{-\infty}^{\infty} \mathsf{b}_\ell(y)\, l^{\mathrm{BMSC}(\mathtt{h}_\ell)}(y)\, \mathrm{d}y, \qquad \text{with } \mathsf{b}_\ell \triangleq \Lambda(\rho(\mathsf{a}_\ell)).$$

(iv) Halt when some convergence criterion is met and return the current estimate $(\mathtt{h}_\ell, g_\ell^{\mathrm{EBP}})$. (In practical implementations one can require that a properly defined distance between $\mathsf{a}_\ell$ and $\mathsf{a}_{\ell+1}$ becomes smaller than a certain threshold.)

The described procedure is found to converge rapidly in practice. Moreover, the limit is found to be (within numerical precision) independent of the initial condition $\mathsf{a}_0$. Proving these statements for this particular procedure seems a challenging task (notice that unlike in ordinary density evolution, the sequence $\{\mathsf{a}_\ell\}$ is in general not ordered by physical degradation). However it is possible to show that, if $\mathtt{x}$ is such that $R_{\mathtt{x}}$ is "well defined", then this procedure has at least one fixed point. This is shown in Appendix 6.A based on a new application of the extremes of information combining presented in Theorem 3.1.

### 6.3.3 EBP Area Theorem

Recall from Chapter 4 that a key ingredient for proving the Maxwell construction over the BEC is (the EBP area) Theorem 4.2. This theorem states that the integral associated with the EBP curve equals the design rate. Combined with the upper bound on the MAP threshold (based on the standard area theorem), it shows the Maxwell construction in the various cases where the upper bound is proved to be tight.

Let us assume that the EBP GEXIT curve obtained from the previous procedure "behaves well" so that we can compute the associated integral. What is the value of this integral? Is it again equal to the design rate of the considered dd pair? The original proof of the EBP area theorem follows from a straightforward computation. It is therefore not possible to proceed in a similar fashion for BMS channels because no analytic expression of the EBP curve is available in general. Let us therefore look at the alternative proof presented in Example 3.8 for BEC($\epsilon$) under the hypothesis that $\epsilon^{\text{SC}} \leq 1$. Following this proof we can extend the EBP area theorem to general memoryless symmetric channels.

**Theorem 6.3** [EPP Area Theorem – BMSC] Consider a dd pair $(\lambda, \rho)$ and transmission over the smooth and ordered family $\{\text{BMSC}(\mathtt{h})\}$. Let $g^{\text{EBP}}$ denote the corresponding EBP GEXIT function. Assume that the corresponding $\{\mathsf{f}_\mathtt{x}\}_\mathtt{x}$ and $\{\mathsf{c}_\mathtt{x}\}_\mathtt{x}$, $\mathtt{x} \in [0,1]$, form a *complete fixed-point family*. Then

$$\int_0^1 g^{\text{EBP}}(\mathtt{x})\mathrm{d}\mathtt{h}(\mathtt{x}) = r_{\lambda,\rho} \stackrel{\triangle}{=} 1 - \frac{\int \rho}{\int \lambda}.$$

*Proof.* We proceed as in Example 3.8. 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 Figure 6.4.

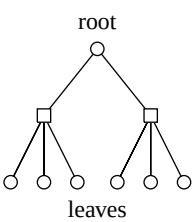



Figure 6.4: Spanning tree of depth one for the $(2,4)$-regular ensemble.

Let us assume that the bit associated with the root node is passed through the channel characterized by $\mathsf{c}_\mathtt{x}$, while those associated with the leaf nodes are passed through a channel characterized by $\mathsf{f}_\mathtt{x}$. Apply the general area theorem. Let $X = (X_1, \ldots, X_{1+\mathtt{l}\times(\mathtt{r}-1)})$ be the transmitted random codeword whose values are chosen uniformly at random from the tree code. Let $Y(\mathtt{x})$ be the result of passing the bits of $X$ through their respective channels with parameter $\mathtt{x}$. Note that $H(X|Y(\mathtt{x}=1)) - H(X|Y(\mathtt{x}=0)) = H(X)$. This follows since by assumption the fixed-point family is complete. In particular this implies that the channel for $\mathtt{x}=0$ is the "noiseless" channel so that $H(X|Y(\mathtt{x}=0))=0$. By the general area theorem, this difference is equal to the sum of the integrals of the individual $g_i^{\text{MAP}}$ curves, where the integral extends from $\mathtt{x}=0$ to $\mathtt{x}=1$. There are two types of individual $g_i^{\text{MAP}}$ curves, namely the one associated with the root node, call it $g_\mathtt{r}$, and the $\mathtt{l}(\mathtt{r}-1)$ ones associated with the leaf nodes, call them $g_\mathtt{l}^{\text{MAP}}$. To summarize, the general area theorem states

$$H(X) = \int_0^1 g_\mathtt{r}^{\text{MAP}}(\mathtt{x})\,\frac{\mathrm{d}\mathtt{h}(\mathtt{x})}{\mathrm{d}\mathtt{x}}\mathrm{d}\mathtt{x} + \mathtt{l}(\mathtt{r}-1)\int_0^1 g_\mathtt{l}^{\text{MAP}}(\mathtt{x})\mathrm{d}\mathtt{x}.$$

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, $\int_0^1 g_\mathtt{l}^{\text{MAP}}(\mathtt{x})\mathrm{d}\mathtt{x} = \int_0^1 1-\rho(1-\mathtt{x})\mathrm{d}\mathtt{x} = (\mathtt{r}-1)/\mathtt{r}$. This follows by applying the area theorem once again to a $[\mathtt{r},1,\mathtt{r}-1]$ single parity-check code. Collecting these observations and solving for $\int_0^1 g_\mathtt{r}^{\text{MAP}}(\mathtt{x})\,\frac{\mathrm{d}\mathtt{h}(\mathtt{x})}{\mathrm{d}\mathtt{x}}\mathrm{d}\mathtt{x}$, we get

$$\int_0^1 g_\mathtt{r}^{\text{MAP}}(\mathtt{x})\,\frac{\mathrm{d}\mathtt{h}(\mathtt{x})}{\mathrm{d}\mathtt{x}}\mathrm{d}\mathtt{x} = 1-\mathtt{l}/\mathtt{r} = r_{\mathtt{l},\mathtt{r}},$$

as claimed since $g_\mathtt{r}^{\text{MAP}} = g^{\text{EBP}}$. 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 $\Lambda$ and each edge emanating from this root node is connected to a check node whose degree is chosen according to the edge degree distribution $\rho$. As before, leaf nodes experience the channel characterized by $\mathsf{f}_x$, whereas the root node experiences the channel characterized by $\mathsf{c}_x$. We apply the general area theorem to each such choice and average with the respective probabilities. □

As in Chapter 4 for the erasure channel, this result imposes some strong constraints on BP GEXIT functions and their connection to Maxwell functions. The next corollary is an example where we can show that the Maxwell curve is a.e. equal to the MAP GEXIT curve under the assumption that the fixed point density family is smooth (and complete).

**Corollary 6.1** Consider communication over the complete and ordered smooth family $\{\mathrm{BMSC}(\mathtt{h})\}_{\mathtt{h}}$, $\mathtt{h} \in [0,1]$ using codes chosen uniformly at random from the ensemble $\mathrm{LDPC}(n,\lambda,\rho)$. Assume that the BP fixed point family $\{\mathrm{BMSC}(\mathtt{h}),\mathtt{a}_{\mathtt{h}}\}$ is smooth and complete. Then MAP GEXIT and BP GEXIT functions coincide for $\mathtt{h} \in [0,1]$.

*Proof.* The hypotheses of Theorem 6.3 are satisfied, therefore $\int_0^1 g^{\mathrm{BP}}(\mathtt{h})\,\mathrm{d}\mathtt{h} = r_{\lambda,\rho}$ Further, by the general area theorem (and applying Fubini's theorem and Fatou's lemma as in the proof of Theorem 6.2) $\int_0^1 g^{\mathrm{MAP}}(\mathtt{h})\,\mathrm{d}\mathtt{h} \geq r_{\lambda,\rho}$. We conclude the proof by noticing that $g^{\mathrm{MAP}}(\mathtt{h}) \leq g^{\mathrm{BP}}(\mathtt{h})$ for every $\mathtt{h} \in [0,1]$ as shown in 6.1. □

Discussion: Unfortunately, for a given dd pair $(\lambda,\rho)$, proving that the hypotheses of the previous corollary hold, in particular that the family is complete, seems to be a challenging task. Of course, numerical computations suggest that this is in fact the case in examples like cycle-code ensembles, see, e.g., Figure 6.2. The existence of a fixed-point pair $(\mathtt{f}_{\mathtt{x}},\mathtt{c}_{\mathtt{x}})$ for each value of $\mathtt{x} = \mathtt{H}(\mathtt{f}_{\mathtt{x}})$ is demonstrated in many cases by Theorem 6.4 (see Appendix 6.A). Some partial analytic results are further presented in [53] to show that the corresponding channel densities are smooth.

## 6.4 Conclusion and Discussion

We have seen the first steps to prove the fundamental connection between MAP and BP decoding for general BMSCs. The central character is the EBP GEXIT curve based on which a Maxwell-type construction can be performed. The resulting curve is conjectured to represent the MAP GEXIT curve.

More precisely, via the numerical procedure of Section 6.3, we were able to obtain densities that describe "unstable" or "hidden stable" branches of the EBP GEXIT curve. Notice first that we could imagine alternative ways to compute these branches and the EBP GEXIT in general. For example, we could modify the BP standard algorithm to include a dynamical channel parameter so that we could follow even the unstable branches (this idea is similar to the "unrevealing" algorithm described in conclusion of Chapter 4). Moreover, the existence, uniqueness and regularity of the EBP GEXIT curve obtained in this chapter needs to be formally established. Further investigations need therefore to be performed on this topic. They will certainly deal with the fixed-point theory (e.g., Brouwer and Schauder theorems) for deriving formal properties of regularity. Alternative approaches (e.g., via a modified BP algorithm such as the M decoder for the BEC) should also be investigated in order to answer the following question. What is the *interpretation* of the Maxwell construction in this general context? Is there any *operational* meaning of this construction, i.e., what is the equivalent of the M decoder for a generic BMS channel?

Let us here summarize what we are able to prove so far. Using GEXIT functions, we have proved an upper bound on the MAP threshold. This bound is conjectured to be tight for a class of ensembles that includes regular LDPC ensembles. Moreover, using EBP GEXIT curves, we have derived some constraints on the relationship between MAP and BP decoding over general BMS channels. These constraints lead us naturally to postulate that the Maxwell construction holds in the general framework of memoryless symmetric channels. This is shown in many cases over the erasure channel.

A natural question arises: Does the coincidence of the BP and MAP GEXIT curves mean that the BP and MAP estimates are equal? Of course, this is true below BP threshold, see [14, 15], but we wonder whether the same is true if we assume that the two GEXIT functions coincide. Perhaps surprising, the answer is positive. We show indeed in [50] that, if the BP and MAP GEXIT functions are equal for $\mathtt{h}$, then, for any given sparse graph code, the average mean square error between extrinsic BP and MAP soft bits, i.e., between $\tanh(\frac{\phi_i^{\mathrm{BP}}(y_{\sim i})}{2})$ and $\tanh(\frac{\phi_i^{\mathrm{MAP}}(y_{\sim i})}{2})$ (see [63]), tends to zero when first $n \to \infty$ and

second $\ell \to \infty$. This implies a rather strict notion of the "correctness" of BP decoding since this shows that BP decoding should be able to reconstruct the full information about $X_i$, given a received vector.

The general area theorem and its consequences might have far wider implications. This has been discussed in Chapter 4 in the erasure case where we have seen that potential applications concern optimization theory. The next (and last) chapter presents some further applications in the field of coding theory.

# Appendix

## 6.A Existence of EBP GEXIT Points

The existence of an EBP GEXIT curve associated with the procedure described in this chapter can be partially demonstrated. We show in this appendix that there exists at least one EBP GEXIT value for each entropy parameter $\mathtt{x}$. Whereas it shows the existence of a EBP GEXIT curve obtained from the considered procedure, it does not show, e.g., that the curve is smooth.

**Theorem 6.4** Consider a dd pair $(\lambda,\rho)$, $\mathtt{x} \in [0,1]$, and let $R_{\mathtt{x}}$ be the corresponding density evolution operator at fixed density entropy $\mathtt{x}$ for the complete and ordered smooth family $\{\mathrm{BMSC}(\mathtt{h})\}_{\mathtt{h}}$. If $\mathtt{H}(\lambda(\rho(\mathtt{a}))) \geq \mathtt{x}$ for any density $\mathtt{a}$ with $\mathtt{H}(\mathtt{a}) = \mathtt{x}$, then there exists at least one density $\mathtt{f}$ such that $R_{\mathtt{x}}(\mathtt{f}) = \mathtt{f}$. Equivalently, $\mathtt{H}(\mathtt{f}) = \mathtt{x}$ and there exists $\mathtt{h} \in [0,1]$ such that $\mathtt{f}$ is a fixed point of density evolution for the channel $\mathrm{BMSC}(\mathtt{h})$.

*Proof.* Consider the space $\mathsf{S}_{\mathtt{x}}$ of $L$-densities $\mathtt{a}$ such that $H(\mathtt{a}) = \mathtt{x}$. Any element in $\mathsf{S}_{\mathtt{x}}$ is a probability measure on the completed real line, satisfying the symmetry condition (formally $\mathtt{a}(-x) = e^{-x}\mathtt{a}(x)$). Vice versa, any such probability measure (to be denoted formally by its "density" $\mathtt{a}$) with $\mathbb{E}[\log(1+e^{-x})] = \mathtt{x}$ corresponds to a unique element of $\mathsf{S}_{\mathtt{x}}$. Notice that the completed linear line $\mathbb{R}_\infty \triangleq [-\infty,+\infty]$ is a compact metric space (we can for instance identify it with $[-1,1]$ through the mapping $x \mapsto \tanh(x/2)$ and use the euclidean metric on $[-1,1]$). Therefore, the space of probability measure on $\mathbb{R}_\infty$ is sub-sequentially compact under the weak topology by Prohorov's theorem [68]. Both the symmetry condition and $H(\mathtt{a}) = \mathtt{x}$ are closed under the same topology, and therefore $\mathsf{S}_{\mathtt{x}}$ is compact as well.

Let $\mathsf{BL}$ be the space of bounded Lipschitz functions on $\mathbb{R}_\infty$ (as above, we identify $\mathbb{R}_\infty$ with $[-1,1]$ and consider the Lipschitz condition with respect to the induced distance) with the corresponding norm $||\cdot||_{\mathsf{BL}}$. The space of probability measures on $\mathbb{R}_\infty$ can be viewed as a convex subset of the dual space $\mathsf{BL}^*$, and the topology induced by the dual norm $||\cdot||^*_{\mathsf{BL}}$ coincides with the weak topology (see [68, Chap.III,§7]). As a consequence $\mathsf{S}_{\mathtt{x}}$ is a compact convex subspace of a normed linear space. By hypothesis, the mapping $\mathtt{a} \mapsto R_{\mathtt{x}}(\mathtt{a})$ is well defined for any $\mathtt{a} \in \mathsf{S}_{\mathtt{x}}$ and maps $\mathsf{S}_{\mathtt{x}}$ into itself. Furthermore, it is easily seen to be continuous with respect to the weak topology. This is a consequence of the Lipschitz continuity of the functions $(y_1,\ldots,y_1) \mapsto y_1 + \cdots + y_1$ and $(y_1,\ldots,y_{\mathtt{r}-1}) \mapsto \boxplus_{i=1}^{\mathtt{r}-1} y_i$ Therefore $R_{\mathtt{x}}$ is compact and, by Schauder's theorem (see [159, Chap.4]) it has at least one fixed point. □

Note that the procedure considered to compute the EBP GEXIT curve, as well as Theorem 6.4, holds unchanged if the entropy functional $\mathtt{H}(\cdot)$ is substituted by any continuous linear functional that preserves the partial order implied by physical degradation.

## 6.B Bounds on the EBP GEXIT Curve

In order to check the hypotheses of Theorem 6.4, it is useful to prove bounds on the entropy of fixed point pairs $(\mathtt{f},\mathtt{c})$. We start by recalling upper and lower bounds on the entropy of $T_{\mathtt{h}}(\mathtt{a})$, which follows from the extremes of information combining.

**Lemma 6.2** [Lower Bound and Upper Bound] Consider a dd pair $(\lambda,\rho)$ and transmission over BMSC$(\mathtt{h})$.
(i) Lower bound: Define

$$\underline{l}(\mathtt{x}) \triangleq \lambda(\mathtt{x}), \quad \underline{r}(\mathtt{x}) \triangleq \sum_i \rho_i h_2\Big(\frac{1-(1-2\epsilon(\mathtt{x}))^{i-1}}{2}\Big),$$

where $\epsilon(\mathtt{x}) \triangleq {h_2}^{-1}(\mathtt{x})$. If $\mathtt{a}$ is a $L$-density with $\mathtt{H}(\mathtt{a})=\mathtt{x}$, then $\mathtt{H}(T_{\mathtt{h}}(\mathtt{a})) \geq \mathtt{h}\,\underline{l}(\underline{r}(\mathtt{x}))$.
(ii) Upper bound: Define

$$\overline{l}(\mathtt{h},\mathtt{x}) \triangleq \sum_i \lambda_i f_{i-1}(\mathtt{h},\mathtt{x}), \qquad \overline{r}(\mathtt{x}) \triangleq 1-\rho(1-\mathtt{x})$$

where $f_i(\mathtt{h},\mathtt{x}) \triangleq \sum_{k\in\{\pm1\}}\sum_{j=0}^{i}\binom{i}{j}(1-\epsilon(\mathtt{x}))^j\epsilon(\mathtt{x})^{i-j}a_k(\mathtt{h})\cdot\log_2\Big(1+\frac{\epsilon(\mathtt{x})^{2j-i}a_{-k}(\mathtt{h})}{(1-\epsilon(\mathtt{x}))^{2j-i}a_k(\mathtt{h})}\Big)$, $a_{+1}(\mathtt{h}) \triangleq 1-\epsilon(\mathtt{h})$, $a_{-1}(\mathtt{h}) \triangleq \epsilon(\mathtt{h})$, and $\epsilon(\mathtt{h}) \triangleq h_2^{-1}(\mathtt{h})$ as above. If $\mathtt{a}$ is a $L$-density with $\mathtt{H}(\mathtt{a})=\mathtt{x}$, then $\mathtt{H}(T_{\mathtt{h}}(\mathtt{a})) \leq \overline{l}(\mathtt{h},\overline{r}(\mathtt{x}))$.

*Proof.* The extremes of EXIT functions (see [105,106,131,134]) have been presented in Theorem 3.1. Moreover expressions for EXIT functions on the BEC and BSC have been derived in Chapter 3.
(i) Following Theorem 3.1, for fixed $\mathtt{H}(\mathtt{a})$ and $\mathtt{H}(\mathtt{b})$, $\mathtt{a}\circledast\mathtt{b}$ has minimum entropy if $\mathtt{a}$ and $\mathtt{b}$ are the densities corresponding to a BEC. But, for the convolution at a parity-check node the minimum is achieved when the input densities correspond to a BSC. The lemma follows by applying these bounds to random variable and check nodes with degree distributions given by $\lambda$ and $\rho$.
(ii) The roles are BEC and BSC are simply exchanged and a similar proof applies by Theorem 3.1. □

This result can be used to check the hypotheses of Theorem 6.4. We deduce that, if $\underline{l}(\underline{r}(\mathtt{x})) \geq \mathtt{x}$ for some $\mathtt{x}\in[0,1]$, then there exists a fixed point pair $(\mathtt{f},\mathtt{c})$ with $\mathtt{H}(\mathtt{f})=\mathtt{x}$ and $\mathtt{c}=\text{BMSC}(\mathtt{h})$ for some $\mathtt{h}$. For instance, for cycle-codes (i.e., for $\lambda(x)=x$) this implies that such a fixed point pair $(\mathtt{f},\mathtt{c})$ exists for any $\mathtt{H}(\mathtt{f})=\mathtt{x}\in[0,1]$.

**Theorem 6.5** [Bounds on EXIT Function] Consider a dd pair $(\lambda,\rho)$ and transmission over the ordered family $\{\text{BMSC}(\mathtt{h})\}_{\mathtt{h}}$. Define the functions

$$\underline{L}(\mathtt{x}) \triangleq \Lambda(\mathtt{x}), \quad \overline{L}(\mathtt{x}) \triangleq \sum_i \Lambda_i f_i(1,\mathtt{x}),$$

and $f(\mathtt{x},\mathtt{x}') \triangleq \max\{\mathtt{h}:\overline{l}(\mathtt{h},\mathtt{x}')=\mathtt{x}\}$ (with the convention $f(\mathtt{x},\mathtt{x}')=0$, if the set is empty). Let $\mathtt{f}$ denote any fixed point of density evolution, i.e., $\mathtt{f}=T_{\mathtt{h}}(\mathtt{f})$. If $\mathtt{H}(\mathtt{f})=\mathtt{x}$ then

$$f(\mathtt{x},\overline{r}(\mathtt{x})) \leq \mathtt{h} \leq \mathtt{x}/\underline{l}(\underline{r}(\mathtt{x})), \qquad \underline{L}(\underline{r}(\mathtt{x})) \leq h^{\text{EBP}} \leq \overline{L}(\overline{r}(\mathtt{x})).$$

In other words, the entropy parameters of any fixed points of density evolution, and so in particular the function $h^{\text{EBP}}$, are contained in the union of rectangles as given above.

*Proof.* The first two inequalities follow from Lemma 6.2. From Lemma 6.2 (i) we get $\mathtt{x}=H(\mathtt{f})=H(T_{\mathtt{h}}(\mathtt{f}))\geq\mathtt{h}\underline{l}(\underline{r}(\mathtt{x}))$, which gives the upper bound on $\mathtt{h}$. Analogously, Lemma 6.2 (ii) implies $\mathtt{x}\geq\overline{l}(\mathtt{h},\overline{r}(\mathtt{x}))$. Since $\overline{l}(\mathtt{h},\overline{r}(\mathtt{x}))$ is monotonically increasing in $\mathtt{h}$, this relation can be inverted.
Given the fixed point $\mathtt{f}$, the corresponding EXIT entropy at variable nodes is $h^{\text{EBP}}=\mathtt{H}(L(\rho(\mathtt{f})))$. The bounds are obtained as in the proofs of Lemma 6.2. □

Discussion: The bounds given above are by no means the best possible. First, the given bounds are "universal" in the sense that they are valid for *all* channel distributions. Better bounds for any specific channel family can be derived by taking the actual input distribution into account. Even in the universal case, slightly better bounds can be given by taking into account that at the variable node before convolution with the channel, the incoming message density cannot be of arbitrary shape but that it is already the convolution of several message densities. Second, tighter bounds on the extremes of information combining have been derived in [132] and can be translated to give tighter bounds on EXIT functions, albeit at the price of more complex expressions. Finally, by using a similar technique one can also give bounds on the entropy versus GEXIT parameter of any fixed point with respect to any smooth channel family.

**Example 6.6** [LDPC$(2/5x+3/5x^5,x^5)$] Consider again the dd pair $(\lambda,\rho)=(2/5x+3/5x^5,x^5)$.

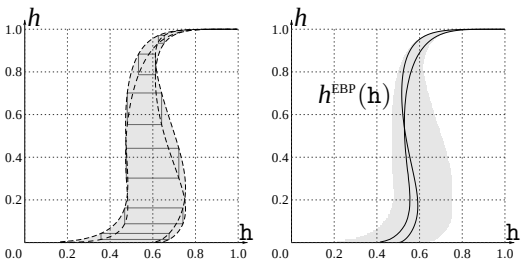



Figure 6.5: Left: Construction of bounding region for all EBP EXIT curves for the dd pair $(\lambda,\rho)=(2/5x+3/5x^5,x^5)$. Right: The EBP EXIT curves for transmission over the BSC and the BEC families.

Figure 6.5 shows on the left the construction of the bounded region (union of rectangles) that contains all EBP GEXIT curves. The dashed lines represent the individual curves traced out by the corner points of the rectangles. On the right, this is compared to the actual EBP GEXIT curves for transmission over the BSC and the BEC families (solid lines).

For many LDPC ensembles Theorem 6.4 ensures the existence of a fixed point pair $(\mathsf{f}_\mathtt{x},\mathsf{c}_\mathtt{x})$ for each value of $\mathtt{x}=H(\mathsf{f}_\mathtt{x})$. However, in order to apply (the EBP area) Theorem 6.3, we need the hypothesis of a smooth family with respect to the parameter $\mathtt{x}$. The fact that this is indeed the case is strongly suggested by the numerical computation of the EBP curve, however a complete characterization is not available. We report some partial analytic results in [53] using the Battacharyya operator. Since the Battacharyya functional is, for most channel families, a smooth function of the channel parameter, then regularity with respect to the Battacharyya operator translates into regularity with respect to the (G)EXIT operator (or any functional that preserves partial order implied by physical degradation).

**Overview:** Turbo codes are part of a number of standards. They can be seen as a particular instance of a multi-edge structure. It is natural to investigate how the Maxwell construction applies to these ensembles.

# 7 | Turbo Codes

Although they were discovered in the early days of information theory [10], LDPC codes have begun to have an impact on coding theory only since the second half of the nineties.
It was the discovery of *Turbo codes* [62] that ignited again the interest for iterative coding systems, which had been long forgotten. Original Turbo codes employ parallel concatenation (see [160]) in combination with a very large interleaver. A similar idea was presented at the same time in [110]. The so-called "Turbo principle" signified a revolution in coding theory.
Although we have illustrated our results using LDPC ensembles, the underlying principles apply to a wide range of systems defined on *sparse graphs* and equivalent statements are expected to hold in large generality. This is exemplified in this chapter using in particular the example of Turbo schemes in Section 7.2. Turbo schemes are instances of multi-edge coding systems (see [161]) for which different types of edge-message densities co-exist. We start by an example where the component codes are replaced by complex (linear) constraints whereas the (average) edge densities remain from a single type.

## 7.1 MAP Thresholds for GLDPC Codes

To give a first example, consider GLDPC ensembles and the case of transmission over the BEC. GLDPC codes were introduced in [57], and further investigated in [79, 80]. GLDPC codes are LDPC codes whose check nodes are replaced by more complex linear constraints. In other words, the parity-check matrix of a GLDPC code is constructed from a suitable LDPC matrix where each non-zero element on a row is replaced by a non-zero column vector (chosen uniformly at random from the parity-check matrix of a so-called *component code*), and each zero element is replaced by a zero vector. The analysis of GLDPC ensemble is therefore similar to the one of LDPC ensembles. In fact, many of our previous results are stated in such a way that they apply directly to GLDPC ensembles. Moreover, notice that it suffices that the "suitable" LDPC matrix fulfills the criteria of Lemma 2.3 in order to ensure that the asymptotic rate of the considered GLDPC ensemble is equal to the design rate (assuming that the component codes have a full rank parity-check matrix).
The right-to-left erasure probability (or MAP EXIT function) often depends on the edge type (except when all component codes are *isotropic*, see Chapter 3). For GLDPC ensembles, we consider the average over all types of nodes and all types of edges: Formally, the (MAP) EXIT function is $\mathtt{y}(\mathtt{x}) \triangleq \mathbb{E}[\frac{1}{\mathtt{r}} \sum_{i=1}^{\mathtt{r}} \mathtt{y}_i(\mathtt{x})]$, where $\mathtt{r}$ is the length of a particular component code and where the expectation is

taken with respect to the proportion of component codes. The distribution $\lambda$ can be freely chosen but must satisfy the design rate constraint $r_{\lambda,\mathtt{y}} = 1 - \frac{1-\int \mathtt{y}}{\int \lambda}$ where $\int \mathtt{y}$ is the rate of the average component code (area theorem). Therefore, equivalently to the dd pair $(\lambda(\mathtt{x}), \rho(\mathtt{x}))$ for LDPC codes, the pair $(\lambda(\mathtt{x}), \mathtt{y}(\mathtt{x}))$ suffices to describe the BP decoding of the GLDPC ensemble in the asymptotic limit. A few computations lead, in general, to an expression for the right component EXIT function $\mathtt{y}(\mathtt{x})$, see Chapter 3 or [32].

For example, consider GLDPC ensembles using $[2^p-1, 2^p-p-1, 3]$ binary Hamming codes as component codes. Since $\mathbb{E}[d_{\min}] \geq 3$, the BP EXIT function has at least one discontinuity at the BP threshold, and the EBP EXIT curve is given in parametric form by $(\epsilon, h^{\text{EBP}}) = \left(\frac{\mathtt{x}}{\lambda(\mathtt{y}(\mathtt{x}))}, \Lambda(\mathtt{y}(\mathtt{x}))\right)$. In general, $\epsilon^{\text{BP}} \neq \epsilon^{\text{MAP}}$ since the BP threshold is not given by the stability condition when the right component code has $d_{\min} \geq 3$ as shown in Appendix 7.A. 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 $(\lambda(\mathtt{x}), \mathtt{y}(\mathtt{x})) = (\mathtt{x}, 3\mathtt{x}^2 + 4\mathtt{x}^3 - 15\mathtt{x}^4 + 12\mathtt{x}^5 - 3\mathtt{x}^6)$. The second example uses the $[15,11,3]$ Hamming code. It can be observed that these standard GLDPC ensembles have relatively "poor" BP thresholds compared to the corresponding MAP thresholds. In the third example, $d_{\min}$ is no longer $> 2$ since we choose, in the node perspective, a mixture composed by 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 one discontinuity at the BP threshold.

| $\lambda(\mathtt{x})$ | $\mathtt{y}(\mathtt{x})$ | $\epsilon^{\text{BP}}$ | $\epsilon^{\text{MAP}}$ | $\epsilon^{\text{SH}}$ |
|---|---|---|---|---|
| $\mathtt{x}$ | $[7,4,3]$ | 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 |

## 7.2 MAP Thresholds for Turbo Codes

As a second example, we apply our upper bound on the MAP threshold to the case of Turbo codes [62, 63,73,110,162]. Without loss of generality, we exemplify this case via the following (standard) class of *bi-dimensional parallel Turbo codes*. Consider a binary rational function $G(D) = p(D)/q(D)$ of degree $m$ with $q_0 = 1$, see [163]. Fix a length $n$. Fix a permutation $\mathtt{G}$ over $[n]$ (which is chosen uniformly at random from $\Pi_{[n]}$ the set of permutations over $[n]$). Consider a vector $x_{[n]}$, which represents $n$ *systematic* bits. We append this vector with $m$ more zeros (termination) and we pass the resulting vector $x^{(s)} \stackrel{\Delta}{=} (x_{[n]}, 0, \cdots, 0)$ (or *sequence* of systematic bits) through the filter $G(D)$: we get a first sequence of $n+m$ *parity* bits (terminated convolutional code) that we call $x^{(p_1)}$. We now permute the $n$ bits of $x$ (i.e., we consider the binary vector $(x_{\mathtt{G}(1)}, \cdots, x_{\mathtt{G}(n)})$), then append them with $m$ more zeros to obtain the vector $(x_{\mathtt{G}(1)}, \cdots, x_{\mathtt{G}(n)}, 0, \cdots, 0)$ that is passed through $G(D)$, to get a second sequence of $n+m$ parity bits that we call $x^{(p_2)}$. The described procedure to encode the "last" bits is called *termination*, see, e.g., [164]. Alternatives are *truncation* or, more elegant, *tail-biting* (see, [136, 165–168]), for which the asymptotic analysis remains unchanged. The natural rate of the resulting parallel concatenated Turbo code is $1/3$ (if we neglect the border effects which vanish like $O_n(1/n)$). In the sequel we focus mainly on this class of standard Turbo code ensembles "à la Berrou-Glavieux", which we denote by $\text{PTurbo}\big(G(D) = \frac{p(D)}{q(D)}, \mathtt{x}, 0, n\big)$.

Further refinements are possible. For example we can puncture the code by erasing uniformly at random (with probability $\pi$) bits from the parity parts so that we can adjust the rate as desired. We can also use different filters (for example $G_1(D)$ and $G_2(D)$ so that the EXIT functions are complementary, see Example 7.2); in the case where several different filters $G_i(D)$ are used, the filter $G(D)$ denotes the *average* filter. More generally, we can consider irregular Turbo codes, see, e.g., [169], by using a distribution $\lambda(\mathtt{x})$ acting on the systematic bits, and then filtering them with the (possibly average) filter $G(D)$ to encode a parity part that we further puncture with probability $\pi$. Let $\text{PTurbo}\big(G(D), \lambda(\mathtt{x}), \pi, n\big)$ denote such a generic ensemble. This ensemble has design rate[1] $r_{\Lambda'(1),\pi} = (1 + \Lambda'(1)(1-\pi))^{-1}$. Elements of this ensemble are distinct if the associated permutations $\mathtt{G}$ are distinct. For our analysis we

[1] The considered component codes have rate $r_{\mathtt{C}} = 1/2$ before puncturing of the parity parts. In general, we can use component codes of *any* rate $r_{\mathtt{C}}$ so that the Turbo ensemble has design rate $r_{r_{\mathtt{C}},\Lambda'(1),\pi} = r_{\mathtt{C}} \big/ \big(r_{\mathtt{C}} + \Lambda'(1)(1-r_{\mathtt{C}})(1-\pi)\big)$.

consider instances of codes that are chosen uniformly at random from the ensemble.

For sake of clarity, let us present the factor graph associated with a standard bi-dimensional Turbo codes of rate $r = 1/3$, i.e., let us consider PTurbo($\frac{p(D)}{q(D)}, \lambda(\mathtt{x}) = \mathtt{x}, \pi = 0, n$). Recall that the mapping of a convolutional encoder at time $i$ is determined by the current state of the corresponding *trellis*, which we denote by $\sigma_i^{(j)}$ for the $j^{\text{th}}$ encoder ($j \in \{1,2\}$). Assume that the vector $(x^{(s)}, x^{(p_1)}, x^{(p_2)})$ is transmitted through $\{\text{BMSC}_i(\mathtt{h}_i = \mathtt{h})\}_{i\in[3(n+m)]}$ so that a corresponding vector $(y^{(s)}, y^{(p_1)}, y^{(p_2)})$ is received. Then, for $i \in [n+m]$, the MAP rule maximizes over $x_i \in \{0,1\}$ the quantity

$$p(x_i^{(s)}|y^{(s)}, y^{(p_1)}, y^{(p_2)}) = \sum_{\sim x_i^{(s)}} p(x^{(s)}, x^{(p_1)}, x^{(p_2)}, \sigma^{(1)}, \sigma^{(2)}|y^{(s)}, y^{(p_1)}, y^{(p_2)})$$

$$= \sum_{\sim x_i^{(s)}} \Big( \prod_{j=1}^{n+m} p(x_j) p(y_j^{(\mathfrak{s})}|x_j^{(\mathfrak{s})}) \prod_{\mathfrak{c}\in\{1,2\}} p(y_j^{(p_\mathfrak{c})}|x_j^{p_{(\mathfrak{c})}}) \Big) \Big( \prod_{\mathfrak{c}\in\{1,2\}} p(\sigma_0^{(\mathfrak{c})}) \prod_{j=1}^{n+m} p(x_j^{(p_\mathfrak{c})}, \sigma_j^{(\mathfrak{c})}|x^{(s)}_{\mathtt{G}^{\mathfrak{c}-1}(j)}, \sigma_{j-1}^{(\mathfrak{c})}) \Big),$$

where $\mathtt{G}(j) \triangleq j$ for $j \in \{m+1, \cdots, n+m\}$.

The corresponding factor graph is depicted in Figure 7.1.
As explained in Section 2.5, standard rules for message-passing decoding apply on this graph. Note that we consider here the following traditional scheduling of the messages. Figure 7.1 shows explicitly two subgraphs (vertical "line" on the left, vertical "line" on the right) that correspond to the trellises of length $n+m$ for each particular component code. The Turbo decoding schedule is governed by the point of view of component codes (see [57]). Each trellis is first processed entirely, then messages are passed to the second trellis. Each time a message is processed by a trellis, the iteration counter $\ell$ is increased by one. In other words, we decode the component convolutional code(s) using the BCJR algorithm of [77]. This will give rise to (G)EXIT chart representations of density evolution where the (G)EXIT functions are associated with the component codes.

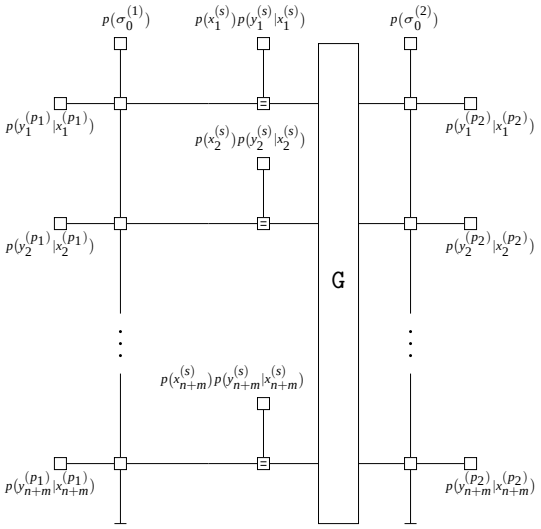



Figure 7.1: Forney-style factor graph of a parallel concatenated Turbo code.

The standard principles of density evolution (concentration around ensemble average, analysis on the computation tree, channel symmetry, see Chapter 2 and Chapter 3) can be applied. For the tree-like assumption of the computation procedure, we consider a windowed BCJR decoding (i.e., a processing up to a depth of $w$ trellis sections of each side of the considered node) of the component codes. Under this assumption the local graph is a tree with probability converging to one (for a fixed number of iterations $\ell > 0$ when $n \to \infty$). If we let first $w \to \infty$, and then $\ell \to \infty$, we can ignore the border effects of the trellis processing. See also [170]. Studying the stationary behavior of the Markov chain, i.e., investigating a bi-infinite trellis, suffices to perform density evolution analysis. Figure 7.2 depicts such a bi-infinite trellis: We define the following functionals $\mathsf{y}_{G(D)}^{(s)}$ and $\mathsf{y}_{G(D)}^{(p)}$ acting on a pair of systematic/parity densities $(\mathsf{x}, \mathsf{c}^\pi)$. The extrinsic "systematic" density that the bi-infinite trellis outputs is given by $(\mathsf{x}, \mathsf{c}^{(\pi)}) \mapsto \mathsf{y}_{G(D)}^{(s)}(\mathsf{x}, \mathsf{c}^{(\pi)})$. In a similar manner, let $(\mathsf{x}, \mathsf{c}^{(\pi)}) \mapsto \mathsf{y}_{G(D)}^{(p)}(\mathsf{x}, \mathsf{c}^{(\pi)})$ represent the extrinsic "parity" density.

Let us first state several equivalency relationships that decrease the number of cases one has to investigate. These relationships can be obtained from a small exercise considering either an *equivalent* code

(ignoring the border effects because of the bi-infinite trellis) or the structure of a trellis section. To a binary polynomial $p(D)$ with $p_0 = 1$ we associate the *reversed* polynomial $\overset{\circlearrowleft}{p}(D) = D^{\deg(P)} p(1/D)$. This definition extends to a binary rational function $G(D) = \frac{p(D)}{q(D)}$ with $p_0 = q_0 = 1$ by setting $\overset{\circlearrowleft}{G}(D) \triangleq \frac{\overset{\circlearrowleft}{p}(D)}{\overset{\circlearrowleft}{q}(D)}$.

**Lemma 7.1** [Equivalence of Encoders] Consider a convolutional encoder defined by a binary rational function $G(D) \triangleq \frac{p(D)}{q(D)}$ with $q_0 = 1$. Consider the two associated functionals $\mathsf{y}^{(s)}_{G(D)}(\cdot,\cdot)$ and $\mathsf{y}^{(p)}_{G(D)}(\cdot,\cdot)$. Then, for any pair of $L$-densities $(\mathsf{a},\mathsf{b})$, $\forall \mathsf{y}_{G(D)} \in \{\mathsf{y}^{(s)}_{G(D)}, \mathsf{y}^{(p)}_{G(D)}\}$, and $\forall j \geq 1$,

(i) $\mathsf{y}_{G(D^j)}(\mathsf{a},\mathsf{b}) = \mathsf{y}_{G(D)}(\mathsf{a},\mathsf{b})$, (ii) $\mathsf{y}_{G(D)}(\mathsf{a},\mathsf{b}) = \mathsf{y}_{D^j G(D)}(\mathsf{a},\mathsf{b})$,

(iii) $\mathsf{y}_{G(D)}(\mathsf{a},\mathsf{b}) = \mathsf{y}_{\overset{\circlearrowleft}{G}(D)}(\mathsf{a},\mathsf{b})$, if $p_0 = 1$, (iv) $\mathsf{y}^{(\mathrm{s})}_{G(D)}(\mathsf{a},\mathsf{a}) = \mathsf{y}^{(\mathrm{p})}_{G^{-1}(D)}(\mathsf{a},\mathsf{a})$, if $p_0 = 1$.

Let us now run density evolution applied to an ensemble PTurbo($G(D) = \frac{p(D)}{q(D)}, \lambda(\mathsf{x}), \pi, n$). Assume that transmission takes place over a BMS channel with associated $L$-density $\mathsf{c}$.

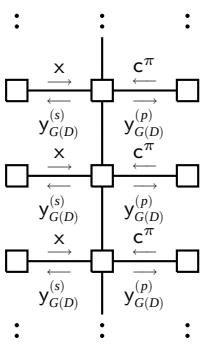



Figure 7.2: Bi-infinite trellis: "Systematic" variable nodes are received from a BMSC with $L$-density $\mathsf{x}$. "Parity" nodes are received from a BMSC with $L$-density $\mathsf{c}^\pi$. After trellis processing, "systematic" nodes experience the extrinsic density $\mathsf{y}^{(s)}_{G(D)}(\mathsf{x},\mathsf{c}^{(\pi)})$ while "parity" nodes experience the extrinsic density $\mathsf{y}^{(p)}_{G(D)}(\mathsf{x},\mathsf{c}^{(\pi)})$.

Let $\mathsf{c}^\pi$ denote the $L$-density emitted from the "parity" nodes; this notation indicates that the density is obtained from the concatenation of $\mathsf{c}$ with BEC($\pi$). Let $\mathsf{x}_\ell$ denote the $L$-density emitted from the "systematic" nodes towards the trellis at iteration $\ell$. Then $\mathsf{x}_0 = \Delta_0$, and for $\ell \geq 0$,

$$\mathsf{x}_{\ell+1} = \mathsf{c} \circledast \lambda\Big(\mathsf{y}^{(s)}_{G(D)}(\mathsf{x}_\ell, \mathsf{c}^\pi)\Big).$$

In general, densities obtained from the described density evolution process "live" in a high dimensional space. This makes an exact computation of the extrinsic density obtained from the functional $\mathsf{y}_{G(D)}(\cdot,\cdot)$ cumbersome. In practice, except in the BEC case (see Appendix 7.B), we determine these densities by sampling.

In order to upper bound the MAP threshold, it remains to compute the BP GEXIT function associated with a particular ensemble. This curve represents the performance of the overall Turbo code once the fixed point of density evolution has been achieved. Let $\mathsf{x}_\infty$ denote the fixed point density emitted from the trellis towards the "systematic" nodes. The BP GEXIT function is given in parametric form by

$$\left(\mathrm{H}(\mathsf{c}), \int \left(r_{\Lambda'(1),\pi} \Lambda\big(\mathsf{y}^{(s)}_{G(D)}(\mathsf{x}_\infty, \mathsf{c}^\pi)\big) + (1 - r_{\Lambda'(1),\pi}) \mathsf{y}^{(p)}_{G(D)}(\mathsf{x}_\infty, \mathsf{c}^\pi)\right)(z) l^{\mathsf{c}}(z) \mathrm{d}z\right)$$

where $r_{\Lambda'(1),\pi} \triangleq \frac{1}{1+\Lambda'(1)(1-\pi)}$ is the design rate of the ensemble.

As in Chapter 4 and Chapter 6 we find an upper bound for the MAP threshold by moving a vertical line from the right to the left, starting at 1 until the area under the BP GEXIT curve is equal to $r_{\Lambda'(1),\pi}$. The BP GEXIT curves for the ensemble of rate $1/2$ parallel Turbo codes with $G(D) = \frac{1+D+D^2+D^3+D^4}{1+D^4}$, $\pi = \frac{1}{2}$, $\lambda(x) = x$ and transmission over the BAWGNC is shown in Figure 7.3. The BP threshold is $\overline{\mathrm{h}}^{\mathrm{BP}} \approx 0.473$ and our (expected tight) upper bound on the MAP threshold is $\overline{\mathrm{h}}^{\mathrm{MAP}} \approx 0.488$ (close to the Shannon threshold which is $\mathrm{h}^{\mathrm{SH}} = 0.5$).

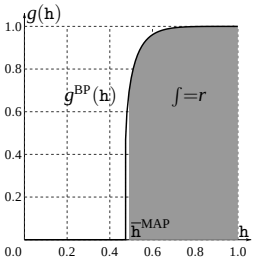



Figure 7.3: BP GEXIT function for the Berrou-Glavieux code over the BAWGNC.

For completeness, we give a few exact results for the BEC case in the next table, where the symbol † indicates the tightness of the stability condition and the last row ($S$) represents a standard serial concatenated scheme.

| $\lambda(\mathtt{x})$ | $\mathtt{y}(\mathtt{x})$ | $\epsilon^{\mathrm{BP}}$ | $\bar{\epsilon}^{\mathrm{MAP}}$ | $\epsilon^{\mathrm{SH}}$ |
|---|---|---|---|---|
| $\mathtt{x}$ | $\frac{1+D^2}{1+D+D^2}$ | 0.6428 | 0.6554 | 0.6667 |
| $\mathtt{x}$ | $\frac{1+D+D^2}{1+D^2}$ | $0.6478^{\dagger}$ | 0.6523 | 0.6667 |
| $\mathtt{x}$ | $\frac{1+D+D^3}{1+D^2+D^3}$ | 0.6369 | 0.6621 | 0.6667 |
| $\mathtt{x}$ | $\frac{1+D+D^3}{1+D^2+D^3}, \frac{1+D+D^3}{1+D}$ | 0.6481 | 0.6570 | 0.6667 |
| $\mathtt{x}$ | $\frac{1+D+D^3}{1+D^2+D^3}, \pi = \frac{1}{2}$ | 0.4651 | 0.4864 | 0.5 |
| $\frac{55\mathtt{x}+45\mathtt{x}^9}{100}$ | $\frac{1+D+D^3}{1+D^2+D^3}, \pi = \frac{68}{100}$ | 0.4825 | 0.4950 | 0.5 |
| $\mathtt{x}$ | $\frac{1+D^2}{1+D+D^2}$ ($S$) | 0.6896 | 0.7484 | 0.75 |

## 7.3 Conclusion and Discussion

Although most of the thesis is concerned with LDPC ensembles, in this brief chapter, we have seen that the basic ideas carry over to more general ensembles such as GLDPC or Turbo ensembles.

Alternative examples of ensembles could be given and discussed; peculiarities (for example the fact that the BP and MAP thresholds can be arbitrarily far apart, and nevertheless still be connected by the Maxwell construction as shown in Appendix 7.C) can be specified; related subjects such as Markovian channels and the computation of their capacity can also be investigated (see, e.g., [171–179]). In fact, our concepts apply to a much wider setting. Ramifications in domains like optimization and the "XORSAT" problem (as discussed in Chapter 4) would be further examples.

# Appendix

## 7.A Properties of GLDPC Ensembles

Let us give some examples of basic properties. Without loss of generality, let us exemplify two statements for the case of transmission over the BEC. Contrary to LDPC or Turbo ensembles, GLDPC ensembles in general have infinite stability condition threshold. This is shown using the minimum distance theorem, see Chapter 3 for the case of the BEC.

**Lemma 7.2** [Stability Condition] Consider a variable node degree distribution $\lambda(\mathtt{x}) = \sum_{i=2}^{\mathtt{l}} \lambda_i \mathtt{x}^{i-1}$ (from an edge perspective) and a family of component codes whose averaged minimum distance is $\leq d_{\min}$. Let $\mathtt{y}(\mathtt{x})$ denote the EXIT function associated with the family of function nodes (or component codes) where the function is uniformly averaged over the edges. Consider the recursive sequence $\mathtt{x}_{\ell+1} = \epsilon\lambda(\mathtt{y}(\mathtt{x}_\ell))$ with $\mathtt{x}_0 = 1$.
[Necessity] If $d_{\min} = 2$ and $\lambda'(0)\mathtt{y}'(0) > \frac{1}{\epsilon}$ then $\exists\xi = \xi(\lambda, \mathtt{y}, \epsilon) \in (0,1]$, such that, $\forall \ell$, $\mathtt{x}_\ell = \mathtt{x}_\ell(\epsilon) > \xi$.
[Sufficiency] If $d_{\min} = 2$ and $\lambda'(0)\mathtt{y}'(0) < \frac{1}{\epsilon}$ or if $d_{\min} \geq 3$ then $\exists\xi = \xi(\lambda, \mathtt{y}, \epsilon) \in (0,1]$, such that if, for some $\ell$, $\mathtt{x}_\ell = \mathtt{x}_\ell(\epsilon) \leq \xi$ then $\mathtt{x}_\ell \to 0$ as $\ell \to \infty$.

Although it was surprising in the early years of Turbo codes, it is now well-known that the choice of "good" component codes does not necessarily help (if we do not consider complexity or finite-length issues) when we aim at optimizing iterative coding system. This can be seen as a direct implication of the formula $C(\epsilon) - r = \frac{\mathcal{D}}{\int\lambda}$; this is formalized in the next lemma.

**Lemma 7.3** ["Good" Component Code Paradox] Consider a sequence of GLDPC ensembles, which we denote by $\{\mathrm{GLDPC}_n(n, \lambda_n(\mathtt{x}), c_n(\mathtt{x}))\}_n$ where $c_n(\mathtt{x})$ is the (MAP) EXIT function associated with the averaged mixture of component codes for the ensemble $\mathrm{GLDPC}_n$. Assume that the component mixture has a fixed rate $r_c = \int_0^1 c_n(x)\mathrm{d}x \in (1 - \int\lambda_n, 1)$. Let $r_{\lambda_n} \triangleq 1 - \frac{1-r_c}{\int\lambda_n} \in (0,1)$ be the design rate of the ensemble and $\epsilon_n^{\mathrm{BP}}$ be the associated BP threshold. If the sequence of component codes is such that, for $\mathtt{x} < 1 - r_c$, $c_n(\mathtt{x})$ decreases and $c_n(\mathtt{x}) \overset{n\to\infty}{\longrightarrow} 0$ (component codes achieve capacity), then the limiting gap to capacity $\liminf_{n\to\infty}[C(\epsilon_n^{\mathrm{BP}}) - r_{\lambda_n}]$ is lower-bounded by $1 - r_c > 0$.

*Proof.* As the function $\lambda_n^{-1}(\mathtt{x}/\epsilon)$ is concave, the area $\mathcal{D}$ is, in the limit of large $n$, at least as large as the area of the triangle $((0,0),(1-r_c,0),(1-r_c,1))$, which is $\frac{1-r_c}{2}$. Therefore $C(\epsilon_n^{\mathrm{BP}}) - r_{\lambda_n} = \mathcal{D}\Lambda'(1) \geq 2\mathcal{D}$. □

# 7.B Turbo Codes over the BEC

A further observation $\Omega$ such that $Y \to X \to \Omega$ has been included in the hypotheses of the general area theorem in Chapter 6. This additional observation $\Omega$ allows us to extend the area theorem to GEXIT charts and parallel concatenated systems. For simplicity, and because of the elegant closed-form expressions for EXIT functions of convolutional codes, let us exemplify this extension with the BEC case.

## A Simplified Matching Condition

We first introduce some code restrictions that allow us to apply the area theorem in a very simple way: For example the area under the "systematic" EXIT function associated with convolutional codes of rate $1/2$ (see Appendix 7.B) will be equal to $\epsilon$ (if the parity bits are transmitted through BEC($\epsilon$)).
Let $\mathtt{C}$ be a proper $[n,k]$ binary linear code with rate $r_{\mathtt{C}} = k/n$, and consider $X$ chosen uniformly at random from $\mathtt{C}$.

**Definition 7.1** For $\Delta \subseteq [n]$, we say that the pair $(\Delta, [n] \setminus \Delta)$ is a $\mathtt{C}$-*compatible partition* of $[n]$ if $H(X_\Delta) = k$ and $H(X_{[n]\setminus\Delta}) = n - |\Delta|$.

Discussion: If $\mathtt{C}$ has generator matrix $G$, then the partition $(\Delta, [n] \setminus \Delta)$ is $\mathtt{C}$-compatible if $\mathrm{rk}(G_\Delta) = k$ and $\mathrm{rk}(G_{[n]\setminus\Delta}) = |[n] \setminus \Delta|$. Note that $\Delta = [n]$ is a trivial $\mathtt{C}$-compatible partitioning set. The $\mathtt{C}$-*compatibility* is a code (not an encoding) characteristic. However the view of $\Delta$ as a systematic[2] encoding part is underlying and we will use it for parallel concatenation.

**Lemma 7.4** Consider a systematic generator matrix $G$ for $\mathtt{C}$, $\Delta \subseteq [n]$, and assume that $(\Delta, [n] \setminus \Delta)$ is a $\mathtt{C}$-compatible partition of $[n]$. If $\Delta$ represents the systematic part of $\mathtt{C}$, then $r_{\mathtt{C}} \geq \frac{1}{2}$.

[2] Recall that an encoder is *systematic* if the associated generator matrix admits a $k \times k$ identity submatrix

*Proof.* Consider the submatrix $G_\Delta = I_k$. If $\Delta$ is a $\mathtt{C}$-compatible partition of $[n]$ then $\mathrm{rk}(G_\Delta) = k$ and $\mathrm{rk}(G_{[n]\setminus\Delta}) = n-k$. Then $k = \mathrm{rk}(G) \geq \mathrm{rk}(G_{[n]\setminus\Delta}) = n-k$, which leads to $r_\mathtt{C} \geq \frac{1}{2}$. □

**Example 7.1** Consider a systematic binary Hamming code $\mathtt{C}^p$ of length $n = 2^p - 1$ for which the subset $\Delta \subseteq [n]$ denotes the systematic part. Then the partition $(\Delta, [n]\setminus\Delta)$ is $\mathtt{C}^p$-compatible. Clearly $\mathrm{rk}(G_\Delta) = k$. Lemma 3.4 gives $\mathrm{rk}(G_{[n]\setminus\Delta}) = \mathrm{rk}(H_\Delta)$. Since $\mathtt{C}^p$ is a Hamming code, $H_\Delta$ is formed by all non-zero non-canonical $p$-tuples, and $H_{[n]\setminus\Delta}$ is formed by the canonical basis of $\{\mathtt{0},\mathtt{1}\}^p$. Any canonical $p$-tuple can be obtained by adding the all-one column of $H_\Delta$ and the corresponding column of $H_\Delta$ with only one zero coordinate. Therefore $H_\Delta$ generates $\{\mathtt{0},\mathtt{1}\}^p$ and $\mathrm{rk}(H_\Delta) = p = n-k$.

Not only Hamming codes but many "good" codes have a compatible systematic part. In particular, almost all convolutional codes encountered in practice in their terminated and truncated block versions have this property.

**Lemma 7.5** [Area Theorem and "Compatible" Set] Assume that $Y$ is the result of passing $X$ through the channel family $\{\mathrm{BEC}_i(\epsilon_i)\}_{i\in[n]}$. Let $(\Delta, [n]\setminus\Delta)$ be a $\mathtt{C}$-compatible partition of $[n]$. If there is a channel parameter pair $(\mathrm{x},\epsilon)$ such that $\forall i \in \Delta$, $\epsilon_i = \mathrm{x}$, $\forall i \in [n]\setminus\Delta$, $\epsilon_i = \epsilon$, then $\int_0^1 \frac{1}{|\Delta|}\sum_{i\in\Delta} h_i^{\mathrm{MAP}}(\mathrm{x},\epsilon)\mathrm{d}\mathrm{x} = \left(1 - \frac{n-k}{|\Delta|}\right) + \left(\frac{n}{|\Delta|} - 1\right)\epsilon$.

*Proof.* Using the non-systematic part as a further observation obtained from $\mathrm{BEC}(\epsilon)$, we expand the result provided by the standard area theorem so that $\int_0^1 \frac{1}{|\Delta|}\sum_{i\in\Delta} h_i^{\mathrm{MAP}}(\mathrm{x},\epsilon)\mathrm{d}\mathrm{x} = \frac{1}{|\Delta|}\sum_{\mathcal{P}\subseteq[n]\setminus\Delta} \epsilon^{n-|\Delta|-|\mathcal{P}|}(1-\epsilon)^{|\mathcal{P}|} H(X_\Delta|X_\mathcal{P})$. For all $\mathcal{P} \subseteq [n]\setminus\Delta$, by definition of a $\mathcal{C}$-compatible partition, we have $H(X_\Delta|X_\mathcal{P}) = n - |\mathcal{P}|$. It suffices to use the Newton binomial to conclude the proof. □

If the component codes have a rate larger than $1/2$, then "good" component codes for iterative parallel concatenation require a compatible systematic part. This is a straightforward application of the matching condition and shows that in this case the "compatibility" is no longer a restriction. This is the case of convolutional codes presented in Appendix 7.B. Let us now present a simplified version of the matching condition when we deal with parallel concatenation and component codes of rate $\geq \frac{1}{2}$. Consider a systematic code $\mathtt{C}$ with rate $r_\mathtt{C}$ and length $n$ whose systematic bits are a compatible partition. Consider an ensemble $\mathrm{PTurbo}(\mathtt{C},\lambda,n,\pi=1)$, i.e., the ensemble of parallel concatenated Turbo codes that use $\mathtt{C}$ as a component code. This ensemble has rate $r_{\lambda,\mathtt{C}} = \frac{r_\mathtt{C}}{r_\mathtt{C} + (1-r_\mathtt{C})\Lambda'(1)}$. Consider the EXIT chart method over $\mathrm{BEC}(\epsilon)$, see Chapter 3: we plot the density evolution process as a staircase function between the curve $\lambda(\mathrm{x}/\epsilon)$ and the curve $\mathrm{y}_\mathtt{C}^{(s)}(\mathrm{x},\epsilon)$. By $\mathtt{C}$-compatibility, we get $\int \mathrm{y}_\mathtt{C}^{(s)}(\mathrm{x},\epsilon)\mathrm{d}\mathrm{x} = 2 - \epsilon + \frac{1}{r_\mathtt{C}}(\epsilon - 1)$. If the two EXIT functions do not overlap, then some calculation reveals that the area between the two is

$$\mathcal{D} = 1 - \frac{\epsilon}{\Lambda'(1)} - (2 - \epsilon + \frac{1}{r_\mathtt{C}}(\epsilon-1)) = \frac{1}{\Lambda'(1)}\frac{(1-\epsilon) - r_{\lambda,\mathtt{C}}}{r_{\lambda,\mathtt{C}}}.$$

In other words, it is proportional to the multiplicative gap to capacity (recall that it was the additive gap to capacity for GLDPC ensembles). In order for the communication to be asymptotically error-free, the matching condition again reads the necessary condition $\mathcal{D} > 0$ which says $r_{\lambda,\mathtt{C}} < 1 - \epsilon$.

## Closed-Form EXIT Functions for Convolutional Codes

For the BEC we can derive compact and exact expressions for the EXIT function of a convolutional code. We will then be able to provide an analytic expression for the (upper bounds on) MAP thresholds of (parallel concatenated) Turbo codes as shown in the last table of Section 7.2. The derivation of closed-form expressions for EXIT functions on the BEC answers a question asked in [32]

The functionals acting on the pair of densities can be computed exactly for the BEC. In this case, density evolution assigns all the mass to only a *finite* number of state-probability vectors and density evolution collapses to determining how the relative probability mass for each such vector changes as a function of the iteration. The number of such state probability vectors is found to be bounded by the following lemma.

**Lemma 7.6** [Pascal-Like Triangle] Consider a binary convolutional code defined by the generator $[1, G(D) = \frac{p(D)}{q(D)}]$ with degree $m$ and $q_0 = 1$. The maximum number $|S(m)|$ of distinct state probability vectors is given by $|S(m)| = \sum_{p=1}^{m+1} C_p^m$ where the numbers $C_p^m$ are obtained from the following recursion in $(p,n)$, $\forall n \geq 1,\ \forall p \in \{0,1,\cdots,n\}$, $C_{p+1}^{n+1} = C_p^n + 2^{p+1} C_{p+1}^n$, with $C_0^n = 1$ and $C_n^n = 1$.

This is best explained by an example. See also [47, 180]. Consider the recursive component convolutional code with rate $\frac{1}{2}$ and generator $[1, \frac{1+D^2}{1+D+D^2}]$. It has memory $m = 2$ (hence 4 states) and will be employed in a parallel concatenated Turbo code with rate $\frac{1}{3}$. Consider the BCJR algorithm for which the forward recursion (see [181]) is the $\alpha$-*recursion* and the backward recursion is called $\beta$-*recursion*. The final combining is called $\gamma$-*recursion*. We run density evolution on PTurbo$(\frac{1+D^2}{1+D+D^2}, x, \pi = 0)$ over BEC$(\epsilon)$.

Under the bi-infinite trellis and the all-one codeword hypotheses, possible state probability vectors at a trellis section of time *i* belong to the set $\{(1,0,0,0), (\frac{1}{2},\frac{1}{2},0,0), (\frac{1}{2},0,\frac{1}{2},0), (\frac{1}{2},0,0,\frac{1}{2}), (\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4})\}$. These vectors correspond to the 5 states of a Markov chain for the $\alpha$-recursion. Denoting $\mathtt{x}$ to be the erasure probability associated with "systematic" nodes, the transition probability matrix is

$$P^{(\alpha)}(\mathtt{x},\epsilon) = \begin{pmatrix} 1-\mathtt{x}\epsilon & \mathtt{x}\epsilon & 0 & 0 & 0 \\ (1-\mathtt{x})(1-\epsilon) & 0 & \mathtt{x}(1-\epsilon) & (1-\mathtt{x})\epsilon & \mathtt{x}\epsilon \\ 0 & 1 & 0 & 0 & 0 \\ (1-\mathtt{x})(1-\epsilon) & 0 & (1-\mathtt{x})\epsilon & \mathtt{x}(1-\epsilon) & \mathtt{x}\epsilon \\ 0 & (1-\mathtt{x})(1-\epsilon) & 0 & 0 & 1-(1-\mathtt{x})(1-\epsilon) \end{pmatrix}$$

The steady-state probability vector representing these 5 states satisfies $\hat{\pi}^{(\alpha)}(\mathtt{x},\epsilon)P^{(\alpha)}(\mathtt{x},\epsilon) = \hat{\pi}^{(\alpha)}(\mathtt{x},\epsilon)$. It is $\hat{\pi}^{(\alpha)}(\mathtt{x},\epsilon) = \left(\frac{(1-\mathtt{x})(1-\epsilon)}{\mathtt{x}\epsilon}, \frac{1-\mathtt{x}-\mathtt{x}\epsilon}{1+\epsilon-\mathtt{x}}, \frac{\mathtt{x}-\mathtt{x}^2+\epsilon^2-\mathtt{x}\epsilon(1-2\mathtt{x}+2\epsilon)}{1+\epsilon-\mathtt{x}}, \frac{(1-\mathtt{x})\epsilon}{1+\epsilon-\mathtt{x}}, \frac{\mathtt{x}\epsilon}{1-\epsilon-\mathtt{x}+\epsilon\mathtt{x}}\right)$. A similar work can be performed to get the stationary vector $\hat{\pi}^{(\beta)}(\mathtt{x},\epsilon)$ associated with the $\beta$-recursion. It suffices to combine $\hat{\pi}^{(\alpha)}(\mathtt{x},\epsilon)$ and $\hat{\pi}^{(\beta)}(\mathtt{x},\epsilon)$ to get the desired output from the $\gamma$-recursion. This gives a *closed-form* expression for the extrinsic erasure probability. This compact form is sometimes quite simple, e.g., $\mathtt{y}^{(s)}_{\frac{1}{1+D}}(\mathtt{x},\epsilon) = \frac{\epsilon\mathtt{x}(2-2\epsilon+\mathtt{x}\epsilon)}{(1-\epsilon(1-\mathtt{x}))^2}$.

In fact, BCJR decoding of a finite-length trellis over BEC$(\epsilon)$ gives rise to EXIT functions that converge uniformly to the limiting EXIT function obtained from the previous method. Therefore, not surprisingly, many finite-length statements extend to bi-infinite trellises. Some more thought shows that the integral under the EXIT function associated with the "systematic" nodes is $\epsilon$. This was an initial intuition for the area theorem. More precisely, for a (convolutional) code of length n with systematic bits passed through BEC(x) and parity bits passed through BEC$(\epsilon)$, we see that $\frac{H(X|Y(\mathtt{x}),\Omega(\epsilon))}{n} = \frac{1}{n}\sum_{i=1}^n H(X_i|Y,X_1,\cdots,X_{i-1},\Omega) = \frac{\mathtt{x}}{n}\sum_{i=1}^n \mathtt{y}^{(s)}_{\mathtt{C}_n}(\mathtt{x}(1-\frac{i}{n}),\epsilon) \underset{n\to\infty}{\longrightarrow} \int_0^{\mathtt{x}} \mathtt{y}^{(s)}_{\mathtt{C}_n}(\tilde{\mathtt{x}},\epsilon)\mathrm{d}\tilde{\mathtt{x}}$ where the first equality comes from a similar averaging as in the proof of Theorem 3.5 and where the last inequality is obtained as a Riemann sum in combination with uniform convergence. Some more thought shows that the minimum distance theorem applied to the closed-form expression of the EXIT function now gives the *free distance* of the convolutional code.

**Example 7.2** [BP Thresholds for Rate $1/3$ Parallel Turbo Codes with Memory $m = 3$] By performing an exhaustive search we have collected all thresholds for standard ensembles PTurbo$(G(D), \lambda(\mathtt{x}) = \mathtt{x}, \pi = 0)$ using the same $m \leq 3$ rational function $G(D)$ for the two parity sequences (bi-dimensional *symmetric* Turbo code). For example, using $G_1(D) = \frac{1+D+D^3}{1+D^2+D^3}$ (UMTS generator), we found $\epsilon^{\text{BP}} \approx 0.6369$. Using $G_2(D) = \frac{1+D^2+D^3}{1+D+D^2}$ (BN-LD generator, see [182]), we found $\epsilon^{\text{BP}} \approx 0.6444$.

Now, using the UMTS filter $G_1(D)$ for the first sequence of parity bits combined with the BN-LD filter $G_2(D)$ for the second sequence of parity bits (this forms a bi-dimensional *asymmetric* Turbo code), we get $\epsilon^{\text{BP}} = \frac{(26+6\sqrt{33})^{\frac{2}{3}} + 2(26+6\sqrt{33})^{\frac{1}{3}} - 8}{6(26+6\sqrt{33})^{\frac{1}{3}}} \approx 0.648$, which exceeds all other BP thresholds found for bi-dimensional *symmetric* Turbo codes with the given memory.

## 7.C Difference between MAP and BP Threshold

Fix $r \in (0,1) \cap \mathbb{Z}$. Consider a sequence[3] of dd pairs $\{(\lambda(x), \rho(x)) = (x^{\mathtt{l}-1}, x^{\frac{1}{1-r}-1})\}_{\mathtt{l} \geq 2}$ with fixed design rate $r_{\mathtt{l},\mathtt{r}} = r$. Ensembles associated with this sequence are regular LDPC code ensembles. We have seen in Fact 4.1 that such ensembles have at most one jump. Moreover, as discussed after Lemma 4.3, our bound on the MAP threshold is expected (and can be shown) to be tight for any regular LDPC ensemble.

It is already shown in [83] 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 (non-rigorous) replica method, an explicit asymptotic expansion of the MAP threshold is given in [28].

Let us show here how to prove this fact using our machinery. The fact that the (tight upper bound on the) MAP threshold $\epsilon^{\text{MAP}}(\mathtt{l})$ converges to the Shannon threshold is shown in Fact 7.2. On the contrary, as stated in Fact 7.1, the BP threshold $\epsilon^{\text{MAP}}(\mathtt{l})$ 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 still be constructed from the corresponding (E)BP EXIT curve! This is illustrated in Figure 7.4 and the proofs are given in the sequel.

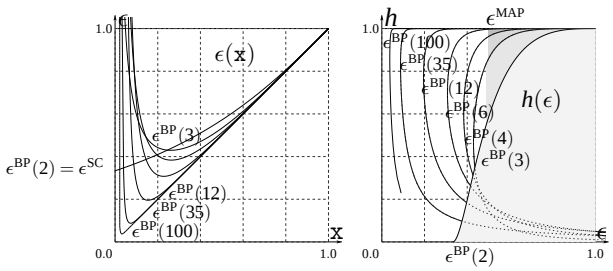



Figure 7.4: Regular LDPC Ensembles with design rate $r = \frac{1}{2}$. Left: Channel entropy function $\mathtt{x} \mapsto \epsilon^{(\mathtt{l})}(\mathtt{x})$. Right: EBP EXIT curve $h^{(\mathtt{l})}(\epsilon) \longleftrightarrow \epsilon^{(\mathtt{l})}(h)$. The depicted ensembles are, in decreasing 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 MAP threshold goes to the Shannon limit 0.5.

**Lemma 7.7** For a fixed non-negative $\mathtt{x} \in (0,1]$, denote $\epsilon^{(\mathtt{l})}(\mathtt{x}) \triangleq \frac{\mathtt{x}}{(1-(1-\mathtt{x})^{\frac{1}{1-r}-1})^{\mathtt{l}-1}}$. Then $\epsilon^{(\mathtt{l})}(\mathtt{x}) \underset{\mathtt{l}\to\infty}{\longrightarrow} \mathtt{x}$.

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

**Fact 7.1** [Limiting BP Threshold for Regular LDPC Ensembles] Fix $r \in (0,1) \cap \mathbb{Z}$. Consider the sequence $(x^{\mathtt{l}-1}, x^{\frac{1}{1-r}-1})\}_{\mathtt{l}\geq 2}$ with fixed design rate $r$. Then $\epsilon^{\text{BP}}(\mathtt{l}) \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$.

*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 7.7, we get $\exists \mathtt{l}_0 \in \mathbb{N},\ \forall \mathtt{l} \geq \mathtt{l}_0\ \ \epsilon^{(\mathtt{l})}(\frac{\xi}{2}) \leq \frac{\xi}{2} + \frac{\xi}{2}$. 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 $\xi$ meaning $\epsilon^{\text{BP}}(\mathtt{l}) \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$. □

Instead of studying the parameterized EBP EXIT $h(\mathtt{x}) \triangleq (1-(1-\mathtt{x})^{\mathtt{r}-1})^{\mathtt{l}}$, we work directly with the inverse mapping $h \mapsto \mathtt{x}(h) \triangleq 1-[1-h^{\frac{1}{\mathtt{l}}}]^{\frac{1}{\mathtt{r}-1}}$ and we use $\epsilon(h) = \frac{1-[1-h^{\frac{1}{\mathtt{l}}}]^{\frac{1}{\mathtt{r}-1}}}{h^{\frac{1-\mathtt{l}}{\mathtt{l}}}}$ for $h \in (0,1]$.

**Lemma 7.8** For a fixed $h \in (0,1)$, we have $\epsilon(h) = \frac{1-(1-h^{\frac{1}{\mathtt{l}}})^{\frac{r-1}{1-r+1}}}{h^{\frac{1-\mathtt{l}}{\mathtt{l}}}} \underset{\mathtt{l}\to\infty}{\longrightarrow} 0$.

[3] By convention we consider only the elements for which $\frac{1}{1-r} - 1$ is a well-defined integer.

*Proof.* The second term of the numerator goes to 1 since $\log(1-h^{\frac{1}{1}}) = \frac{\log h}{1} + \log(\frac{1}{h^{\frac{1}{1}}} - 1) = \frac{\log h}{1} + \log(\frac{-\log h}{1} + o(\frac{1}{1}))$ such that $\frac{r-1}{1-1+r}[\frac{\log h}{1} + \log(\frac{-\log h}{1} + o(\frac{1}{1}))] \underset{1\to\infty}{\longrightarrow} 0$. The lemma follows from the fact that the denominator behaves as $h^{\frac{1-1}{1}} \underset{1\to\infty}{\sim} h > 0$. □

Discussion: Notice that $\epsilon(h)$ does not uniformly converge to 0 on $(0,1)$ since, e.g., $\int_0^1 \epsilon(h)\mathrm{d}h = 1-r \neq 0$.

**Fact 7.2** [Limiting MAP Threshold for Regular LDPC Ensembles] Fix $r \in (0,1) \cap \mathbb{Z}$. Consider the sequence $(x^{1-1}, x^{\frac{1}{1-r}-1})\}_{1\geq 2}$ with fixed design rate $r$, then $\epsilon^{\text{MAP}}(1) \underset{1\to\infty}{\longrightarrow} \epsilon^{\text{SH}} = 1-r > 0$.

*Proof.* First, obviously $0 \leq \epsilon^{\text{SH}} - \epsilon^{\text{MAP}}(1)$. Second, from algebraic considerations, we have

$$\begin{aligned}
\epsilon^{\text{SH}} - \epsilon^{\text{MAP}}(1) &= (1-r) - \epsilon^{\text{MAP}}(1) = \left(1 - \int_{\epsilon^{\text{MAP}}(1)}^1 h^{(1)}(\epsilon)\mathrm{d}\epsilon\right) - \left(1 - \int_{\epsilon^{\text{MAP}}(1)}^1 \mathrm{d}\epsilon\right) \\
&= \int_{\epsilon^{\text{MAP}}(1)}^1 [1 - h^{(1)}(\epsilon)]\mathrm{d}\epsilon \leq \int_{\epsilon^{\text{BP}}(1)}^1 [1 - h^{(1)}(\epsilon)]\mathrm{d}\epsilon = \int_{h^{\text{BP}}(1)}^1 [\epsilon^{(1)}(h) - h^{\text{BP}}(1)]\mathrm{d}h \\
&\leq \int_{h^{\text{BP}}(1)}^1 \epsilon^{(1)}(h)\mathrm{d}h \leq \int_0^1 \tilde{\epsilon}^{(1)}(h)\mathrm{d}h,
\end{aligned}$$

where $h^{\text{BP}}(1) \triangleq \epsilon^{-1}(\epsilon^{\text{BP}}(1))$ and $\tilde{\epsilon}^{(1)}(h) \triangleq \begin{cases} \epsilon^{(1)}(h), & \text{if } h \in [h^{\text{BP}}(1), 1), \\ \epsilon^{\text{BP}}(1), & \text{if } h \in (0, h^{\text{BP}}(1)). \end{cases}$

Fact 7.1 and Lemma 7.8 give $\tilde{\epsilon}^{(1)} \underset{1\to 0}{\longrightarrow} 0$ for $h \in (0,1)$ which, followed by the application of the dominated convergence theorem to the sequence $\tilde{\epsilon}^{(1)} \leq 1$, shows that $\lim_{1\to\infty} \int_0^1 \tilde{\epsilon}^{(1)}(h)\mathrm{d}h = 0$. This completes the proof. □

# Bibliography

[1] A. K. Khandekar and R. J. McEliece, "On the complexity of reliable communication on the erasure channel," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Washington, USA), p. 1, June 24–29 2001.

[2] R. G. Gallager, *Information theory and reliable communication*. Wiley, 1968.

[3] J. D. V. der Waals, *Over de Continuiteit van den Gas- en Vloeistoftoestand (On the continuity of the gaseous and liquid states)*. PhD thesis, Leiden University, Holland, June 1873.

[4] J. C. Maxwell, *Theory of Heat*. London, United Kingdom: Longmans, Green and Co., 4th ed., 1875.

[5] J. L. Sengers, *How Fluids Unmix; Discoveries by the School of Van der Waals and Kamerlingh Onnes*. Amsterdam, Holland: Koninklijke Nederlandse Akademie van Wetenschappen, 2002.

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

[7] M. Graetzel and P. Infelta, *The bases of chemical thermodynamics*, vol. 1 / 2. Parkland, Florida: Universal publishers, 2000.

[8] B. Jancovici, *Thermodynamique et physique statistique*. Paris, France: Nathan, 1996.

[9] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. 8, pp. 21–28, Jan. 1962.

[10] R. G. Gallager, *Low-density parity-check codes*. Cambridge, Massachusetts: MIT Press, 1963.

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

[12] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 569–584, Feb. 2001.

[13] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inform. Theory*, vol. 47, pp. 585–598, Feb. 2001.

[14] 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.

[15] 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.

[16] C. Di, D. Proietti, T. Richardson, E. Telatar, and R. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1570–1579, June 2002.

[17] A. Orlitsky, K. Viswanathan, and J. Zhang, "Stopping set distribution of LDPC code ensembles," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Yokohama, Japan), p. 123, June 29 - July 4 2003.

[18] A. Montanari, "Finite-size scaling and metastable states of good codes," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), 2001.

[19] A. Amraoui, A. Montanari, T. Richardson, and R. Urbanke, "Finite-length scaling for iteratively decoded LDPC ensembles," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2003.

[20] A. Amraoui, A. Montanari, T. Richardson, and R. Urbanke, "Finite-length scaling for iteratively decoded LDPC ensembles." submitted to IEEE Trans. Inform. Theory, June 2004.

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

[22] J. Tillich and G. Zémor, "The Gaussian isoperimetric inequality and decoding error probabilitites for the Gaussian channel," *IEEE Trans. Inform. Theory*, vol. 50, pp. 328–331, Feb. 2004.

[23] I. Sason and S. Shamai, "Improved upper bounds on the ML decoding error probability of parallel and serial concatenated turbo codes via their ensemble distance spectrum," *IEEE Trans. Inform. Theory*, vol. 46, pp. 24–47, Jan. 2000.

[24] G. Wiechman and I. Sason, "Improved bounds on the parity-check density and achievable rates of binary linear block codes with applications to LDPC codes." *submitted to IEEE Trans. Inform. Theory*, arXiv:cond-math/cond-mat/0505057.

[25] N. Sourlas, "Spin-glass models as error-correcting codes," *Nature*, vol. 339, pp. 693–695, June 1989.

[26] N. Sourlas, "Spin glasses, error-correcting codes and finite-temperature decoding," *Europhysics Letters*, vol. 25, pp. 159–164, 1994.

[27] A. Montanari and N. Sourlas, "Statistical mechanics and turbo codes," in *Proc. of the International Symposium on Turbo Codes and Related Topics*, (Brest, France), pp. 63–66, Sept. 2000.

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

[29] S. Franz, M. Leone, A. Montanari, and F. Ricci-Tersenghi, "The dynamic phase transition for decoding algorithms," *Physical Review E*, vol. 66, no. 046120, 2002.

[30] F. Guerra and F. L. Toninelli, "Quadratic replica coupling in the Sherrington-Kirkpatrick mean field spin glass model." arXiv:cond-math/0201091 v2, Mar. 2002.

[31] A. Montanari, "Tight bounds for LDPC and LDGM codes under MAP decoding," *IEEE Trans. Inform. Theory*, vol. 51, pp. 3221 – 3246, Sept. 2005.

[32] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel property," *IEEE Trans. Inform. Theory*, vol. 50, pp. 2657–2673, Nov. 2004.

[33] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.

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

[35] P. Oswald and Shokrollahi, "Capacity-achieving sequences for the erasure channel," *IEEE Trans. Inform. Theory*, vol. 48, pp. 3017–3028, Dec. 2002.

[36] 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), p. 266, June 29–July 4 2003.

[37] 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), pp. 63–66, Dec. 4–7 2001.

[38] M. Mézard, F. Ricci-Tersenghi, and R. Zecchina, "Two solutions to diluted p-spin models and XORSAT problems," *J. of Stat. Phys.*, vol. 111, pp. 505–533, 2003. arXiv:cond-math/cond-mat/0207140.

[39] D. Guo, S. Shamai, and S. Verdú, "Mutual information and conditional mean estimation in Poisson channels," in *Proc. of the IEEE Inform. Theory Workshop*, (San Antonio, TX, USA), Oct. 24–29 2004.

[40] D. Guo, S. Shamai, and S. Verdú, "Mutual information and minimum mean-square error in Gaussian channels," *IEEE Trans. Inform. Theory*, vol. 51, pp. 1261–1882, Apr. 2005.

[41] N. Macris, "Correlation inequalities: a useful tool in the theory of LDPC codes," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Adelaide, Australia), Sept. 4–9 2005.

[42] M. Zakai, "On mutual information, likelihood ratios, and estimation error for the additive Gaussian channel," *IEEE Trans. Inform. Theory*, vol. 51, pp. 3017–3024, Sept. 2005.

[43] D. Guo, S. Shamai, and S. Verdú, "Additive non-Gaussian noise channels: Mutual information and conditional mean estimation," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Adelaide, Australia), Sept. 4–9 2005.

[44] 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, TX, USA), Oct. 24–29 2004.

[45] M. Tüchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proc. of the Int. ITG Conf. on Source and Channel Coding*, (Berlin, Germany), Jan. 2002.

[46] K. Bhattad and K. R. Narayanan, "An MSE based transfer chart to analyze iterative decoding schemes," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2004.

[47] C. Méasson and R. Urbanke, "Asymptotic analysis of turbo codes over the binary erasure channel," in *Proc. of the 12th Joint Conference on Communications and Coding*, (Saas Fee, Switzerland), March 2002.

[48] C. Méasson and R. Urbanke, "An upper-bound on the ML thresholds of LDPC ensembles over the BEC," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2003.

[49] 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, IL, USA), June 27 – July 2 2004.

[50] C. Méasson, A. Montanari, T. Richardson, and R. Urbanke, "Maximum a posteriori decoding and turbo codes for general memoryless channels," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Adelaide, Australia), Sept. 4–9 2005.

[51] C. Méasson, A. Montanari, and R. Urbanke, "Why we can not surpass capacity: the matching condition," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2005.

[52] C. Méasson, A. Montanari, and R. Urbanke, "Maxwell's construction: The hidden bridge between iterative and maximum a posteriori decoding." submitted to IEEE Trans. Inform. Theory, June 2005.

[53] C. Méasson, A. Montanari, T. Richardson, and R. Urbanke, "The generalized area theorem and some of its consequences." submitted to IEEE Trans. Inform. Theory, Nov. 2005.

[54] C. E. Shannon, "A mathematical theory of communication," *Bell System Tech. J.*, vol. 27, pp. 379–423, 623–656, July and Oct. 1948.

[55] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[56] F. J. MacWilliams and N. J. Sloane, *The theory of error-correcting codes*. North-Holland, 1977.

[57] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. 27, pp. 533–547, Sept. 1981.

[58] N. Wiberg, *Codes and Decoding on General Graphs*. PhD thesis, Linköping University, S-581 83, Linköping, Sweden, 1996.

[59] F. R. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[60] G. D. Forney, "Codes on graphs: Normal realizations," *IEEE Trans. Inform. Theory*, vol. 47, Feb. 2001.

[61] G. D. Forney, "Lecture notes." MIT, 2005.

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

[63] 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.

[64] J. Hagenauer, "Lecture notes." TUM, 2005.

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

[66] İ. E. Telatar, "Lecture notes." EPFL, 2005.

[67] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *European Trans. on Telecommunications*, vol. 6, pp. 513–526, Sep./Oct. 1995.

[68] A. N. Shiryaev, *Probability*. Springer, 1996.

[69] O. Lévêque, "Lecture notes." EPFL, 2005.

[70] O. Gallay, "Mutual information and mimimum mean-square error in Gaussian channels." Student project, EPFL (Advisor: O. Lévêque), 2005.

[71] J. Hagenauer, "Soft-in/soft-out: The benefits of using soft-decisions in all stages of digital receivers," in *Proceedings of the 3rd International Workshop on DSP Techniques applied to Space Communications*, (ESTEC, Noordwijk, Netherlands), Sept. 1992.

[72] J. Hagenauer, *Soft is better than hard*. Kluwer Publication, 1994. Proceedings in Communications, Coding and Cryptology, (Blahut, D. and Costello, D./Eds.).

[73] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, 1996.

[74] C. Berrou and A. Glavieux, "Reflections on the prize paper "near optimum error correcting coding and decoding: Turbo codes"," *IEEE Inform. Theory Society Newsletter*, vol. 48, no. 2, 1998.

[75] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Inform. Theory*, vol. 46, pp. 325–343, Mar. 2000.

[76] J. Pearl, "Fusion, propagation, and structuring in belief networks," *Artificial Intelligence*, vol. 29, pp. 241–288, 1986.

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

[78] T. Etzion, A. Trachtenberg, and A. Vardy, "Which codes have cycle-free Tanner graphs?," *IEEE Trans. Inform. Theory*, vol. 45, pp. 2173 – 2181, Sept. 1999.

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

[80] 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.

[81] V. Zyablov and M. Pinsker, "Estimation of the error-correction complexity of Gallager low-density codes," *Problemy Peredachi Informatsii*, vol. 11, pp. 23–26, Jan. 1975.

[82] G. A. Margulis, "Explicit constructions of graphs without short cycles and low density codes," *Combinatorica*, vol. 2, pp. 71–78, 1982.

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

[84] S. Benedetto and G. Montorsi, "Performance evaluation of turbo-codes," *Electronics Letters*, vol. 31, no. 3, pp. 163–165, 1995.

[85] D. Divsalar, S. Dolinar, F. Pollara, and R. J. McEliece, "Transfer function bounds on the performance of turbo codes." TDA Progress Report 42-122, Communications Systems and Research Section, California Institute of Technology, 1995.

[86] G. Battail, M. Decouvelaere, and P. Godlewski, "Replication decoding," *IEEE Trans. Inform. Theory*, vol. 25, pp. 332–345, May 1979.

[87] P. Flajolet and R. Sedgewick, "The average case analysis of algorithms: Saddle point asymptotics," tech. rep., RR 2376, 1994.

[88] E. A. Bender and L. B. Richmond, "Central and local limit theorems applied to asymptotic enumeration II: multivariate generating functions," *J. Combin. Theory*, vol. A 34, pp. 255–265, 1983.

[89] A. Shamir and J. Spencer, "Sharp concentration of the chromatic number on random graphs $G_{n,p}$," *Combinatorica*, vol. 7, pp. 121–129, 1987.

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

[91] K. Azuma, "Weighted sums of certain dependent random variables," *Tohoku Mathematical Journal*, vol. 19, pp. 357–367, 1967.

[92] W. Hoeffding, "Probability inequalitites for sums of bounded random variables," *Journal of the American Statistical Association*, vol. 58, pp. 13–30, 1963.

[93] G. Miller and D. Burshtein, "Asymptotic enumeration method for analyzing LDPC codes," *IEEE Trans. Inform. Theory*, vol. 50, pp. 1115–1131, June 2004.

[94] 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.

[95] S. L. Litsyn and V. S. Shevelev, "Distance distribution in ensembles of irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. IT–49, pp. 3140 –3159, Dec. 2003.

[96] 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), p. 50, June 24–29 2001.

[97] C. Di, T. Richardson, and R. Urbanke, "Weight distribution of low-density parity-check codes," *IEEE Trans. Inform. Theory*, 2004. submitted to IEEE Trans. Inform. Theory.

[98] C. Di, *Asymptotic and finite-length analysis of low-density parity-check codes*. PhD thesis, EPFL, Lausanne, Switzerland, 2004. Number 3072.

[99] 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), p. 102, June 27 – July 2 2004.

[100] R. L. Wheeden and A. Zygmund, *Measure and integral*. New York, USA: Marcel Dekker, 1977.

[101] N. C. Wormald, "Differential equations for random processes and random graphs," *Ann. Appl. Probab.*, vol. 5, pp. 1217–1235, 1995.

[102] N. C. Wormald, "The differential equation method for random graph processes and greedy algorithms." Notes on lectures given a the Summer School on Randomized Algorithms in Antonin, Poland, 1997.

[103] J. Berkmann, "On turbo decoding of nonbinary codes," *IEEE Commun. Let.*, vol. 2, pp. 94–96, 1998.

[104] C. Hartmann and L. Rudolph, "An optimum symbol-by-symbol decoding rule for linear codes," *IEEE Trans. Inform. Theory*, vol. 22, pp. 514–517, Sept. 1976.

[105] I. Land, P. Hoeher, S. Huettinger, and J. B. Huber, "Bounds on information combining," in *Proc. of the Int. Symposium on Turbo Codes and Related Topics*, (Brest, France), Sept. 2003.

[106] I. Sutskover, S. Shamai, and J. Ziv, "Extremes of information combining," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), 2003.

[107] F. J. MacWilliams, "A theorem on the distribution of weights in a systematic code," *Bell System Tech. J.*, vol. 42, pp. 79–94, 1963.

[108] S. Riedel, "Symbol-by-symbol MAP decoding algorithms for high-rate convolutional codes that use reciprocal dual codes," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 175–185, Feb. 1998.

[109] B. Bollobás, *Random Graphs*. Cambridge University Press, 2001.

[110] J. Lodge, R. Young, P. Hoeher, and J. Hagenauer, "Separable map "filters" for the decoding of product and concatenated codes," in *Proceedings of ICC'93*, (Geneve, Switzerland), pp. 1740–1745, May 1993.

[111] S. ten Brink, "Convergence of iterative decoding," *Electronics Letters*, vol. 35, pp. 806–808, May 1999.

[112] S. ten Brink, "Iterative decoding trajectories of parallel concatenated codes," in *Proc. of the Int. ITG Conf. on Source and Channel Coding*, pp. 75–80, Jan. 2000.

[113] S. ten Brink, "Iterative decoding for multicode CDMA," in *Proc. IEEE VTC*, vol. 3, pp. 1876–1880, May 1999.

[114] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart," *AEU Int. J. Electron. Commun.*, vol. 54, pp. 389–398, 2000.

[115] S. Vialle and J. Boutros, "Performance limits of concatenated codes with iterative coding," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Sorrento, Italy), p. 150, June 25–30 2000.

[116] S. Vialle, *Construction et analyse de nouvelles structures de codage de canal adaptées au traitement itératif*. PhD thesis, ENST, Paris, France, Dec. 2000.

[117] D. Burshtein and G. Miller, "Bounds on the performance of belief propagation decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 112–122, Jan. 2002.

[118] S.-Y. Chung, *On the construction of some capacity-approaching coding schemes*. PhD thesis, MIT, Cambridge, Massachusetts, 2000.

[119] S.-Y. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 657–670, Feb. 2001.

[120] S.-Y. Chung, T. Richardson, and R. Urbanke, "Gaussian approximation for sum-product decoding of low-density parity-check codes," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Sorrento, Italy), p. 318, June 25–30 2000.

[121] J. Hagenauer, "The EXIT chart – Introduction to extrinsic information transfer in iterative decoding," in *Proc. of the European Sign. Proc. Conf.*, (Vienna, Austria), pp. 1541–1548, Sept. 2004.

[122] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proceedings of the* 29*th annual ACM Symposium on Theory of Computing*, pp. 150–159, 1997.

[123] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman, "Analysis of low-density codes and improved designs using irregular graphs," in *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pp. 249–258, 1998.

[124] S.-Y. Chung, J. Forney, G. D., T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, pp. 58–60, Feb. 2001.

[125] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Trans. Commun.*, vol. 50, pp. 754–767, May 2002.

[126] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, pp. 670–678, Apr. 2004.

[127] M. Tüchler and J. Hagenauer, "EXIT charts of irregular codes," in *Proc. of the Conf. Inf. Science and Systems*, (Princeton University, NJ, USA), Mar. 2002.

[128] A. Amraoui (personal communication: lthcwww.epfl.ch/research/ldpcopt/).

[129] A. Schaefer, N. Görtz, and J. Hagenauer, "Analysis tools for iterative source-channel decoding," in *Proc. of the Int. Symposium on Turbo Codes and Related Topics*, (Brest, France), Sept. 2003.

[130] F. E. Flegbo, "Iterative coding over the fading channel." Student project, EPFL (Advisor: C. Méasson), 2003.

[131] S. Huettinger and J. B. Huber, "Information processing and combining in channel coding," in *Proc. of the Int. Symposium on Turbo Codes and Related Topics*, (Brest, France), Sept. 2003.

[132] I. Sutskover, S. Shamai, and J. Ziv, "Extremes of information combining," *IEEE Trans. Inform. Theory*, vol. 51, no. 4, pp. 1313–1325, 2005.

[133] I. Land, *Reliability Information in Channel Decoding*. PhD thesis, TUK, Germany, 2005.

[134] I. Land, P. Hoeher, S. Huettinger, and J. B. Huber, "Bounds on information combining," *IEEE Trans. Inform. Theory*, vol. 51, no. 2, pp. 612–619, 2005.

[135] A. D. Wyner and J. Ziv, "A theorem on the entropy of certain binary sequences and applications: Part I," *IEEE Trans. Inform. Theory*, vol. 19, pp. 769–777, Nov. 1973.

[136] A. R. Calderbank, G. D. Forney, and A. Vardy, "Minimal tail-biting trellises: the Golay code and more," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1435–1455, July 1999.

[137] 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), p. 115, June 30 – July 5 2002.

[138] P. Oswald and A. Shokrollahi, "Capacity achieving sequences for the erasure channel," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Washington, USA), p. 48, June 24–29 2001.

[139] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: a model and two properties," in *Proc. of the Conf. Inf. Science and Systems*, (Princeton University, NJ, USA), Mar. 2002.

[140] A. Shokrollahi, "New sequences of linear time erasure codes approaching the channel capacity," in *Proceedings of AAECC-13, Lecture Notes in Computer Science 1719*, no. 1719 in Lecture Notes in Computer Science, pp. 65–76, Springer Verlag, 1999.

[141] S. ten Brink, "Exploiting the chain rule of mutual information for the design of iterative decoding schemes," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2001.

[142] A. Ashikhmin, G. Kramer, and S. ten Brink, "Code rate and the area under extrinsic information transfer curves," in *submission to IEEE Int. Symposium on Inform. Theory*, (Lausanne, Switzerland), 2002.

[143] 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, pp. 439–454, Mar. 2004.

[144] 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.

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

[146] R. Rockafellar, *Convex Analysis*. Princeton, New Jersey: Princeton University Press, 1970.

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

[148] D. Guo, S. Shamai, and S. Verdú, "Mutual information and MMSE in Gaussian channels," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Chicago, IL, USA), p. 349, June 27 – July 2 2004.

[149] A. H. Zemanian, *Distribution Theory and Transform Analysis*. New York, USA: Dover Publications, 1965.

[150] D. Guo, *Gaussian Channels: Information, Estimation and Multiuser Detection*. PhD thesis, Princeton University, 2004.

[151] A. J. Stam, "Some inequalities satisfied by the quantities of information of Fisher and Shannon," *Inform. and Control*, vol. 2, pp. 101–112, 1959.

[152] N. M. Blachman, "The convolution inequality for entropy powers," *IEEE Trans. Inform. Theory*, vol. 11, pp. 267–271, July 1965.

[153] T. E. Duncan, "On the calculation of the mutual information," *SIAM J. Appl. Math.*, vol. 19, pp. 215–220, July 1986.

[154] A. R. Barron, "Entropy and the central limit theorem," *The Annals of Probability*, vol. 14, pp. 336–342, Jan. 1986.

[155] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing*. Oxford Science Publications, 2001.

[156] M. Costa, "A new entropy power inequality," *IEEE Trans. Inform. Theory*, vol. 31, pp. 751–760, Nov. 1985.

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

[158] D. Burshtein, M. Krivelevich, S. L. Litsyn, and G. Miller, "Upper bounds on the rate of LDPC codes," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2437–2449, Sept. 2002.

[159] R. F. Brown, *A topological introduction to nonlinear analysis*. Birkhäuser, 1993.

[160] G. D. Forney, *Concatenated Codes*. PhD thesis, MIT, Cambridge, Massachusetts, 1966.

[161] T. Richardson and T. Urbanke, "Multi-edge type LDPC codes." submitted to IEEE Trans. Inform. Theory, 2004.

[162] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Proceedings of ICC'94*, pp. 645–649, May 1994.

[163] R. Johannesson and K. S. Zigangirov, *Fundamentals of convolutional coding*. IEEE Press, 1998.

[164] M. Bossert, *Kanalcodierung*. Stuttgart, Germany: B. G. Teubner, 1998. (English ed.: *Channel Coding for Telecommunications*. New York, USA: Wiley, 1999).

[165] R. Koetter and A. Vardy, "The structure of tail-biting trellises: Minimality and basic principles," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1877–1901, Sept. 2003.

[166] S. M. Aji, G. B. Horn, and R. J. McEliece, "On the convergence of iterative decoding on graphs with a single cycle," in *Proc. of the Conf. Inf. Science and Systems*, (Princeton University, NJ, USA), Mar. 1998.

[167] J. B. Anderson and S. M. Hladik, "Tail-biting MAP decoders," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 297–302, Feb. 1998.

[168] S. Riedel and C. Weiss, "The Golay convolutional code–Some application aspects," *IEEE Trans. Inform. Theory*, vol. 45, pp. 2191–2199, Sept. 1999.

[169] J. Boutros, G. Caire, E. Viterbo, H. Sawaya, and S. Vialle, "Turbo code at 0.03 dB from capacity limit," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Lausanne, Switzerland), p. 56, June 30 – July 5 2002.

[170] P. Brémaud, *Markov chains, Gibbs fields, Monte Carlo simulation, and queues*. New York, USA: Springer-Verlag (2nd printing), 2001.

[171] C. Douillard, A. Picart, M. Jézéquel, P. Didier, C. Berrou, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Trans. on Telecommunications*, vol. 6, pp. 507–511, Sept. 1995.

[172] T. Ktari, "Iterative decoding and the inter-symbol interference channel." Student project, EPFL (Advisor: C. Méasson), 2004.

[173] F. Michaud, "Channel modeling and implementation of low-density parity-check codes." Student project, EPFL (Advisor: C. Méasson, R. Urbanke), 2002.

[174] C. Neuberg, "Gilbert-Elliott channel and iterative decoding." Student project, TUD (Advisor: C. Méasson), 2004.

[175] E. Perron, "Hidden Markov model based on Clarke's fading channel." Student project, EPFL (Advisor: C. Méasson), 2003.

[176] I. Ravot, "Iterative coding over the Gilbert-Elliott channel." Student project, EPFL (Advisor: C. Méasson), 2004.

[177] P. Reymond, "Inter-symbol interference channels: capacity and iterative algorithms." Student project, EPFL (Advisor: C. Méasson), 2005.

[178] D. Arnold, H.-A. Loeliger, and P. O. Vontobel, "Computation of information rates from finite-state source/channel models," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2002.

[179] A. Goldsmith, T. Holliday, and P. Glynn, "Entropy and mutual information for Markov channels with general inputs," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 2002.

[180] T. Richardson and R. Urbanke, "Thresholds for turbo codes," in *Proc. of the IEEE Int. Symposium on Inform. Theory*, (Sorrento, Italy), p. 317, June 25–30 2000.

[181] G. D. Forney, "The forward-backward algorithm," in *Proc. of the Allerton Conf. on Commun., Control and Computing*, (Monticello, IL, USA), Oct. 1996.

[182] P. C. Massey and D. J. Costello, "New developments in asymmetric turbo codes," in *Proc. of the Int. Symposium on Turbo Codes and Related Topics*, (Brest, France), Sept. 2000.

Cyril Méasson
French-citizen – April 3, 1974 – cyril.measson@gmail.com

## PhD Thesis

2001-2006 Communication Theory Lab, EPFL, Lausanne, Switzerland
*Conservation laws for coding*

## Teaching Activities

EPFL, Lausanne, Switzerland

Assistant
*Information theory and coding* (E. Telatar), winter semester 2004-2005
*Information theory* (B. Faltings, J.C. Chappelier), winter semesters 2002-2003 / 2003-2004
*Modern coding theory* (R. Urbanke), summer semesters 2001 / 2002
*Advanced digital communications* (R. Urbanke), winter semesters 2000-2001 / 2001-2002

Advisor
P. Reymond, *Inter-symbol interference channels: capacity and iterative algorithms*, semester project for EPFL, winter 2004-2005
C. Neuberg, *Gilbert-Elliott channel and iterative decoding*, semester project for TU Dresden, Germany, summer 2004
T. Ktari, *Iterative decoding and the inter-symbol interference channel*, semester project for EPFL, winter 2003-2004.
I. Ravot, *Iterative coding over the Gilbert-Elliott channel*, semester project for EPFL, summer 2004
E. Perron, *Hidden Markov model based on Clarke's fading channel*, semester project for EPFL, winter 2002-2003
F.E. Flegbo, *Iterative coding over the fading channel*, semester project for EPFL, winter 2002-2003

## Publications

Journal
C. Méasson, A. Montanari, T. Richardson, R. Urbanke, "The general area theorem and some of its consequences," sub. to IEEE Trans. on Inform. Th., Nov. 2005
C. Méasson, A. Montanari, R. Urbanke, "Maxwell's construction: The hidden bridge between iterative and maximum a posteriori decoding," sub. to IEEE Trans. on Inform. Th., Jun. 2005
J. Hagenauer, E. Offer, C. Méasson, M. Mörz, "Decoding and equalization with analog non-linear networks," *European Trans. on Telecomm.*, Vol. 10, No 6, pp 659-680, Nov./Dec. 1999

Conference
C. Méasson, A. Montanari, R. Urbanke, "Why we can not surpass capacity: the matching condition," *Proc. 43th Annual Allerton Conference on Communication, Control and Computing*, Monticello, USA, Oct. 2005
C. Méasson, A. Montanari, T. Richardson, R. Urbanke, "MAP decoding and turbo codes for general memoryless channels," *Proc. ISIT*, Adelaide, Australia, Sep. 2005
C. Méasson, A. Montanari, T. Richardson, R. Urbanke, "Life above threshold: from list decoding to area theorem and MSE," *Proc. ITW*, San Antonio, USA, Oct. 2004
C. Méasson, A. Montanari, R. Urbanke,"Maxwell's construction: the hidden bridge between maximum-likelihood and iterative decoding," *Proc. ISIT*, Chicago, USA, Jun./Jul. 2004
C. Méasson, R. Urbanke, "An upper-bound on the ML threshold of LDPC Ensembles over the BEC," *Proc. 41th Annual Allerton Conference on Communication, Control and Computing*, Monticello, USA, Oct. 2003
C. Méasson, R. Urbanke, "Further analytic properties of EXIT-like curves and applications," *Proc. ISIT*, Yokohama, Japan, Jun./Jul. 2003
C. Méasson, R. Urbanke, "Asymptotic analysis of turbo codes over the binary erasure channel," *Proc. 12th Joint Conference on Communications and Coding*, Saas Fee, Switzerland, Mar. 2002

Miscellaneous
C. Méasson, *Analog decoding on graphs with cycles*, Munich University of Technology, Institute for Communications Engineering, MSc thesis, DA 1352, Feb. 1999
C. Méasson, "Analog decoding," *Proc. Quellen- und Kanalcodierung für digitale Kommunikationssysteme*, Universität Erlangen-Nürnberg – TU München, Kurs 2 der Ferien-Akademie 1998, pp 237-250, Sarntal – Südtirol, Sep. 1998

Previous Employment

| | |
|---|---|
| 1999-2000 | Lucent Technologies Inc., Bell Laboratories, Murray Hill, USA |
| 3 months | Visiting scientist in the Mathematics for Communications Dep. and Physics Dep. |
| 6 months | Consultant in the Wireless Research Dep.: testing for an analog 0.25$\mu$m IC decoder |
| 1998-2000 | Inst. for Comm. Engineering, Munich University of Technology, Germany |
| 14 months | Research Engineer: developments on analog decoding and turbo equalization |
| 1997 | European Aeronautic Defence and Space EADS Company, Ottobrunn, Germany |
| 4 months | Trainee: test-system for an earth observation satellite |
| 1996 | Employee (3 months) at Swatch SMH Group, Microelectronic, Marin, Switzerland |
| 1995 | Operator (3 months) at France Telecom, Saint-Etienne, France |

Education

| | |
|---|---|
| 2000-2001 | Graduate School in Communications Systems, EPFL, Lausanne, Switzerland |
| 1995-1998 | MSc degree (*Ingénieur*) with European exchange program, ENSEEIHT, Toulouse, France |
| 10 months | MSc thesis (*Diplomarbeit*): *Analog decoding on graphs with cycles* |
| | Inst. for Communications Eng., Munich University of Technology, Germany |
| 6 months | Math. Dep. and Electrical Eng. Dep., Berlin University of Technology, Germany |
| 20 months | Inst. for Electronics and Signal Processing, ENSEEIHT, Toulouse, France |
| 1992-1995 | Lycée Claude Fauriel, Saint-Etienne, France |
| | Undergraduate Studies (*Mathématiques supérieures et spéciales*) |