Non-Binary LDPC Codes and EXIT Like Functions

July 2008

Non-Binary LDPC Codes and EXIT Like Functions

Vishwambhar Rathi

EPFL - Ecole Polytechnique Fédérale de Lausanne

July 2008

Thesis No. 4111 (July 2008)

Thesis presented to the faculty of computer and communication sciences for obtaining the degree of Docteur ès Sciences

Accepted by the jury:

R. Urbanke
Thesis directors

E. Soljanin
Expert

A. Loeliger
Expert

A. Shokrollahi
Expert

B. Rimoldi
President of the jury

Ecole Polytechnique Fédérale de Lausanne, 2008

Abstract

In the first part of this thesis we are interested in the asymptotic performance analysis of Non-Binary Low-Density Parity-Check (NBLDPC) codes over the Binary Erasure Channel (BEC) decoded via the suboptimal Belief Propagation (BP) decoder as well as the optimal Maximum-A-Posteriori (MAP) decoder.

Previously, NBLDPC codes defined with respect to a finite field have been studied in the literature. Unfortunately, for these codes the asymptotic analysis of the BP decoder in terms of density evolution is cumbersome since the involved densities "live" in a high dimensional space. To alleviate this problem, we introduce ensembles defined with respect to the general linear group. For these ensembles the density evolution equations can be written in a compact form. We compute thresholds for different alphabet sizes for various NBLDPC ensembles. Surprisingly, the threshold is not a monotonic function of the alphabet size. We also conjecture the stability condition for NBLDPC ensembles defined with respect to finite fields over any binary memoryless symmetric channel.

We then consider the performance of NBLDPC codes under MAP decoding when transmission takes place over the BEC. Towards this goal, we generalize the concepts of the peeling decoder and stopping sets to NBLDPC codes. Using these concepts, we give a combinatorial characterization of decoding failures for NBLDPC codes decoded via the BP decoder. We use the decoding failure criterion and the density evolution analysis to compute the asymptotic residual degree distribution for NBLDPC codes. The rate of this residual ensemble gives the conditional entropy of the NBLDPC code. To compute the rate of the residual ensemble we generalize the criterion of [1] to the non-binary setting, which, when satisfied, guarantees that almost every code in the ensemble has a rate equal to the design rate. We observe that the Maxwell construction of [1], relating the performance of MAP and BP decoding via the Extended BP GEXIT (EBP GEXIT [2]) function, holds in the setting of NBLDPC codes.

The weight and stopping set distribution of a code are related to its performance under the MAP and the BP decoder respectively. We first concentrate on the weight distribution of regular NBLDPC ensembles. We derive the average weight distribution of regular NBLDPC ensembles. Using this, we derive the average of equivalent binary weight distribution of regular NBLDPC ensembles. We show that as the alphabet size becomes larger the equivalent binary weight distribution converges to a straight line up to a critical value of the normalized weight. Beyond the critical value, it converges to the weight distribution of Gallager's random parity check ensemble.

The *average* weight distribution of binary LDPC ensembles has been extensively studied in the literature. Much less is known about the probability distribution of this quantity. In particular, the question of the concentration of the weight distribution around the ensemble average has not been addressed. We compute the variance of the weight distribution of binary regular LDPC ensembles. Using this and the second moment method we obtain bounds on the probability that a randomly chosen code has its weight distribution close to the ensemble average. We show that a large fraction of the total number of codes have their weight distribution close to the average. We apply the same technique for the stopping set distribution of binary regular LDPC ensembles. Again we show that a large fraction of total number of codes have their stopping set distribution close to the ensemble average.

In the last part of this thesis we address the question of the existence of the EBP GEXIT function. The EBP GEXIT function plays a fundamental role in the asymptotic analysis of sparse graph codes. For transmission over the BEC using binary LDPC codes, the analytic properties of the EBP GEXIT function are relatively simple and well understood. The case of general BMS channel is much harder and even the existence of the curve is not known in general. We introduce some tools from non-linear analysis which can be useful to prove the existence of EXIT like curves in some cases. The main tool is the Krasnoselskii-Rabinowitz (KR) bifurcation theorem.

Keywords: LDPC codes, non-binary low-density parity-check codes, density evolution, MAP decoding performance, weight distribution, stopping set distribution, second moment method, EBP GEXIT, bifurcation theorem.

Résumé

Dans la première partie de cette thèse nous nous intéressons à l'analyse asymptotique de la performance des codes Low-Density Parity-Check non-binaires (NBLDPC) sur le canal à effacement binaire (BEC) décodées à l'aide d'un décodeur Belief Propagation (BP) sousoptimal ainsi que d'un décodeur optimal Maximum-A-Posteriori (MAP).

Dans la littérature, les codes NBLDPC définis à l'égard d'un corps fini sont très populaires. Malheureusement, pour ces codes l'analyse asymptotique du décodeur BP en termes d'évolution de densité est lourde puisque les densités impliquées "habitent" dans un espace à haute dimension. Pour remédier à ce problème, nous introduisons des ensembles définis à l'égard du groupe linéaire général. Pour ces ensembles, les équations d'évolution de la densité peuvent être écrites sous une forme compacte. Nous calculons des seuils pour les différentes tailles d'alphabet et pour des ensembles NBLDPC variés. Etonnamment, le seuil n'est pas une fonction monotone de la taille de l'alphabet. Nous posons également la conjecture de la condition de stabilité pour les ensembles NBLDPC définis à l'égard de corps finis sur tout canal binaire symétrique sans mémoire.

Nous examinons ensuite examiner les performances des codes NBLDPC sous décodage MAP lorsque la transmission se déroule sur le BEC. À cette fin, nous généralisons les concepts du décodeur "peeling" et des "stopping sets" pour les codes NBLDPC. En utilisant ces concepts, nous donnons une caractérisation combinatoire pour les échecs de décodage des codes NBLDPC décodés via le décodeur BP. Nous utilisons le critére d'échec de décodage et l'analyse de l'évolution de la densité pour calculer la distribution asymptotique des degrés résiduels pour les codes NBLDPC. Le taux de cet ensemble résiduel donne l'entropie conditionnelle du code NBLDPC. Pour calculer le taux résiduel de l'ensemble, nous généralisons le critère [1] au cas non-binaire, garantissant que presque tous les codes de l'ensemble ont le taux égal au taux de construction. Nous obsérvons également que la construction de Maxwell [1], appliquée à l'analyse des performances MAP et BP par la fonction d'GEXIT BP étendue (EBP GEXIT [2]) donne des bons résultats.

Les distributions du poids et des ensembles d'arrêt d'un code sont respectivement liées avec les performances des décodeurs MAP et BP. Nous nous focusons d'abord sur la distribution du poids des ensembles de codes NBLDPC réguliers. En l'utilisant dans la suite, nous dérivons la distribution équivalente du poids binaire moyenne pour les codes NBLDPC réguliers. Nous démontrons que cette distribution converge vers la distribution du poids des codes de Gallager avec la taille de l'alphabet.

La distribution du poids moyenne des codes LDPC binaires a été très étudiée dans la litérature. Nous connaissons beaucoup moins sur la distribution de probabilités de cette quantité. Notamment, la question de concentration de la distribution du poids autour de la moyenne de l'ensemble n'a pas été encore explorée. Nous calculons la variance de la distribution du poids pour les codes LDPC binaires réguliers. En l'utilisant par la suite avec la technique des moments séconds, nous obtenons des bornes sur la probabilité que la distribution du poids d'un code choisi arbitrairement d'un ensemble est proche de la distribution du poids moyenne de cet ensemble. Nous démontrons qu'une large fraction des codes de l'ensemble ont leur distribution du poids proche de la moyenne. Nous utilisons la même technique pour trouver la distribution des ensembles d'arrêt des codes LDPC binaires réguliers. Nous démontrons le résultat similaire dans ce cas, qu'une large fraction des codes de l'ensemble ont leur distributions des ensembles d'arrêt proches de la moyenne.

Dans la dernère partie de la thèse, nous étudions la question de l'existance de la fonction GEXIT EBP. Cette fonction joue un rôle fondamental dans l'analyse asymptotique des codes définis par des graphes. Quand nous supposons la transmission sur le canal binaire à effacements à l'aide des codes LDPC binaires, les propriétés de la fonction GEXIT EBP sont rélativement simples et bien étudiées. Le cas des canaux binaires symmètriques sans mémoire est plus compliqué et même l'existance de cette courbe n'est pas connue dans le cas général. Nous introduisons quelques techniques issues de l'analyse non-linéaire qui peuvent être utiles pour prouver l'exisatnce des courbes de type EXIT dans certains cas particuliers. L'outil principal est le théorème de bifurcation de Krasnoselkii-Rabinowitz (KR).

Keywords: codes LDPC, codes LDPC non-binaires, évolution de densité, performances du décodeur MAP, distribution du poids, distribution des ensembles d'arrêt, méthode des moments séconds, courbe GEXIT EBP, théorème de bifurcation.

Acknowledgments

I thank my advisor, Professor Rüdiger Urbanke, from the bottom of my heart, for his immense help, support, and patience. Without his guidance, this thesis would not have been possible. I am indebted to him for the amount of learning and education I have received from him.

I am grateful to Prof. Professor Hans-Andrea Loeliger, Prof. Bixio Rimoldi, Prof. Amin Shokrollahi, and Dr. Emina Soljanin for agreeing to be member of my thesis committee.

During my stay at EPFL, I was very fortunate to be in a group led by exceptional professors. I thank Prof. Suhas Diggavi for his teachings, guidance, and the interactions I had with him, Prof. Bixio Rimoldi for his teaching and for introducing me to skiing, and Prof. Emre Telatar for the wonderful courses he taught, which were a delight to attend. It was a great fun and learning experience interacting with other senior members of the group, Olivier and Nicolas, for which I am thankful to them.

I am grateful to Muriel ($\mu\mathbb{R}$) for her immense help regarding administrative work, inside as well as outside EPFL and helping me getting around french. I thank Yvonne and Francoise for their help regarding administrative work. I thank Damir, John, and Denice for providing the flawless computer network in our lab.

I was fortunate to be surrounded by great colleagues and friends. I thank Anwitaman, Dinkar, Manish, Partha, Purushottam, Ramp, Rajesh, Reshmi, Sibi, Sanket, Seema, Satish, Shrinivas, Anmol, and Vasu for making my stay in Switzerland a wonderful and joyful experience and for their friendship. It was a great fun having chats with Shashi, Nadim, Chandak, Mahendra and Siddhartha. I cherish the friendship I have with them. I thank Abdel and Cyril for their friendship, help and swimming lessons. I would like to thank all the members of our lab for their friendship and help, Etienne, Tarek, Peter, Jérémie, Iryna, Marius, Ayfer, Eren, Shan-Yuan, Aslan, Changyen, Christine, Stephane, Sohail, Henry, Vojislav, Jasper, Harm, Amin, Mohammed.

Finally, I thank my family, my parents, my wife, sister, uncles, aunts, cousins for their love, support and affection. Most importantly, I dedicate this work to my grand parents.

Contents

Abstract **i**

Résumé **iii**

Acknowledgments **v**

Contents **vii**

1 Introduction **1**
1.1 Outline 3

2 BP Performance of Non-Binary LDPC Codes **5**
2.1 Motivation and Previous Work 5
2.2 Preliminaries and Notation 8
2.3 Belief Propagation Decoder 10
2.4 Density Evolution for the BEC 12
2.4.1 Density Evolution for EGL (λ, ρ, m) 15
2.4.2 Density Evolution for the Ensemble EGF (λ, ρ, m) . . . 22
2.5 Stability Condition 23
2.6 Conclusion 25

3 MAP Performance of Non-Binary LDPC Codes over the BEC **27**
3.1 Introduction 27
3.2 Preliminaries 28
3.3 Peeling Decoder 30
3.4 Residual Degree Distribution and Counting Argument 34
3.4.1 Calculating the Check Node Distribution for $m = 2$ . . 39
3.5 Equality between Conditional Entropy and Rate 41
3.5.1 Example 47
3.6 Conclusion 48
3.A Proof of Lemma 13 49
3.B Proof of Lemma 15 50
3.C Proof of Lemma 19 50

4 Weight and Stopping Set Distribution 55
4.1 Motivation and Summary of Results 55
4.2 Preliminaries 58
4.2.1 Definitions and Notations 58
4.2.2 Second Moment Method 59
4.3 Expectation of Weight Distribution 59
4.4 Second Moment of Weight Distribution 66
4.5 Moment Calculations for Stopping Set Distribution 72
4.6 Discussion 73
4.A Proof of Lemma 30 74

5 Existence Proofs of Some EXIT Like Functions 77
5.1 Introduction 77
5.2 Definitions and Theorem Related to the Existence of Fixed Points 80
5.3 Examples 83
5.4 Conclusion 88
5.A Some Useful Lemmas 88

6 Conclusion 91

Bibliography 93
Curriculum Vitae 97

1 Introduction

The focus in the first part of this thesis is on the performance analysis of Non-Binary Low-Density Parity-Check (NBLDPC) codes when transmission takes place over Binary Memoryless Symmetric (BMS) channels. We consider two main decoding algorithms: the locally optimal (but globally suboptimal) Belief Propagation (BP) decoder and the globally optimal Maximum-A-Posteriori (MAP) decoder. As the analysis for general BMS channels is difficult even in the binary case, we restrict ourselves to the Binary Erasure Channel (BEC).

NBLDPC codes are defined with respect to an underlying algebraic structure. This algebraic structure can be a group, ring or finite field. In the literature, codes defined with respect to a finite field have been widely studied [3, 4, 5]. Like in the binary case, there is no method known to evaluate their asymptotic performance efficiently for transmission over general BMS channels. However, unlike the binary case, no efficient method of evaluating their asymptotic performance is known even for transmission over the BEC. The main reason for this lack of analytical result is, as we will see in Chapter 2, that the number of messages arising in the BP decoder grows as $\Theta\left(q^{\log(q)}\right)$, where q is the size of the alphabet. So it becomes difficult to keep track of the message probabilities as q increases. To circumvent this problem, we propose NBLDPC codes defined with respect to the general linear group. We will show in Chapter 2 that for these codes the number of message probabilities which we need to keep track of only grows as $\Theta\left(\log(q)\right)$. Due to this dimensionality reduction the asymptotic analysis of these codes is significantly easier.

We then generalize the concepts of peeling decoder and stopping sets to non-binary codes and prove the equivalence of the BP and the peeling decoder. Using these generalizations, we evaluate the MAP decoding performance of NBLDPC codes over the BEC by extending the arguments of [1] from binary to the non-binary setting. For the MAP decoding, the performance metrics of interest are the GEXIT function and the conditional entropy [2]. The method to compute the conditional entropy is the following. We show that the BP decoder when applied to the NBLDPC ensemble results in an appropriately defined ensemble called *residual ensemble*. We compute the degree distribution of this residual ensemble and determine its rate. We then show that this rate equals the conditional entropy of the codeword given the observations at the receiver. The technically most challenging part in this sequence is the computation of the rate of the residual ensemble. In order to accomplish this task we generalize the sufficiency condition of [1] to the non-binary setting. This condition guarantees that almost all codes in the residual ensemble have a rate equal to the design rate. Finally, we observe that the Maxwell construction of [1], which relates the derivative of conditional entropy with respect to the channel erasure probability and BP EXIT function, holds in the wider setting of NBLDPC codes.

The rate depends on the total number of codewords in a code which in turn depends on its weight distribution. We derive the average weight distribution of regular NBLDPC ensembles. As we consider transmission over the BEC, the *binary* weight distribution of the NBLDPC code is of importance. By binary weight distribution we mean the weight distribution of the non-binary code when the code is viewed as a binary code. More precisely, the alphabet size of the non-binary codes we consider are given by 2^m for a positive integer m. So we can represent each symbol by m bits. Thus each codeword is also a binary codeword. We show that as the alphabet size becomes larger the average binary weight distribution converges to a straight line up to a critical value of the normalized weight. Beyond the critical value, it converges to the weight distribution of Gallager's random parity check ensemble [6].

We are then interested in the question whether the weight distribution of a code concentrates around its ensemble average. Due to technical difficulties we answer this question only for regular binary LDPC ensembles. Using the second moment method we obtain lower bounds on the probability that a randomly chosen code has its weight distribution close to the ensemble average.

The last question we consider is of importance to prove the conjectured relationship between the asymptotic MAP and the BP decoding performance. The EBP GEXIT function introduced in [2] is a key ingredient in this conjectured relationship which is furnished by the Maxwell construction. The EBP GEXIT function is a parametric function of a suitably defined set of fixed points of the density evolution map. It is not known that the EBP GEXIT function forms a smooth curve. However, numerical calculations suggest this to be true. We propose a theorem from non-linear analysis called Krasnoselskii-Rabinowitz (KR) theorem. Using the KR theorem, we prove the existence of EBP GEXIT like functions in many cases.

In the following section we give the outline of the thesis.

1.1 Outline

Chapter 2 We describe the BP decoder for non-binary alphabets. Then we prove the concentration of the message probability around the ensemble average. We derive the density evolution equations for ensembles over the general linear group and finite fields when transmission takes place over the BEC. We discuss the stability condition for non-binary LDPC defined with respect to the finite field when transmission takes place over any BMS channel.

Chapter 3 We generalize the concepts of peeling decoder and stopping sets to non-binary codes. We then show that the final estimates of the peeling decoder and the BP decoder are equal. Then we prove that, conditioned on the residual degree distribution the residual graph is uniformly distributed. Finally, we generalize the arguments of [1] to the non-binary case to prove the concentration of the rate around the design rate. This enables us to compute the conditional entropy of the code.

Chapter 4 We derive the average weight distribution and the equivalent average binary weight distribution of regular non-binary LDPC ensembles defined with respect to the general linear group and a finite field. We estimate the variance of the weight distribution of regular binary LDPC ensembles. We show that a large fraction of codes have their weight distribution concentrated around the ensemble average. The same result is shown to hold also for the stopping sets of regular binary LDPC ensembles.

Chapter 5 We give some relevant notations and definitions from non-linear analysis. We then introduce the Krasnoselskii-Rabinowitz (KR) theorem which can be useful to prove existence of fixed point. We demonstrate its applicability by considering various ensembles.

Chapter 6 We conclude with some discussion, open problems and future research directions.

BP Performance of Non-Binary LDPC Codes 2

The chapter is organized in the following way: In Section 2.1 we give the motivation for considering non-binary codes and discuss previous work. Preliminaries and definitions are given in Section 2.2. Section 2.3 describes the message-passing decoder for non-binary alphabets. In Section 2.4 we derive the density evolution equations for ensembles over the general linear group and finite fields. We discuss the stability condition in Section 2.5 for non-binary alphabets over any BMS channel. We conclude in Section 2.6 with future research directions.

2.1 Motivation and Previous Work

It is well known that using binary LDPC ensembles for transmission over BMS channels, one can construct codes which achieve rates seemingly arbitrarily close to capacity. In particular, for the BEC there are provable capacity-achieving degree distributions obtained in [7, 8, 9] based on the method of density evolution. The method of density evolution was generalized to any BMS channel in [10]. This generalization made it possible to construct codes for a given BMS channel which can achieve rates very close to the capacity [11, 12]. It should be noted however that good LDPC codes for BMS channels other than the BEC are obtained by numerical optimization. The problem of finding explicitly capacity-achieving degree distribution for general BMS channels is still open. In either case, the main property of the capacity achieving/approaching degree distributions for the standard LDPC ensembles of [7] is that the underlying parity-check matrix gets denser and denser as the gap to capacity is reduced [13].

Although *binary* ensembles are conjectured to constitute a powerful enough class to achieve capacity, it is nevertheless worth exploring the potential of non-binary ensembles. Clearly, by considering non-binary alphabets we add one more degree of freedom in our code design. This added degree of freedom might be of particular value for small lengths (as we will see later, non-binary codes typically have a better error floor performance). Not surprisingly we will see in Section 2.3 that the BP decoding complexity of NBLDPC codes is higher than that of equivalent binary codes. By "equivalent" binary code we mean a binary code with the same degree distribution and same length (measured in the number of bits). Though the BP decoding complexity of non-binary codes is higher than their binary counterpart, they can offer better performance. Thus we can have performance versus complexity tradeoffs by considering non-binary alphabets. For example, in Fig 2.1 we plot the bit error probability of the BP decoder versus the channel noise for the $(2, 3)$ ensemble and different alphabet sizes. As we can see from this figure, the bit error probability performance of the $(2, 3)$ ensemble improves with the alphabet size.

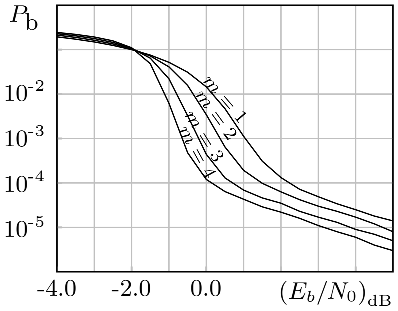
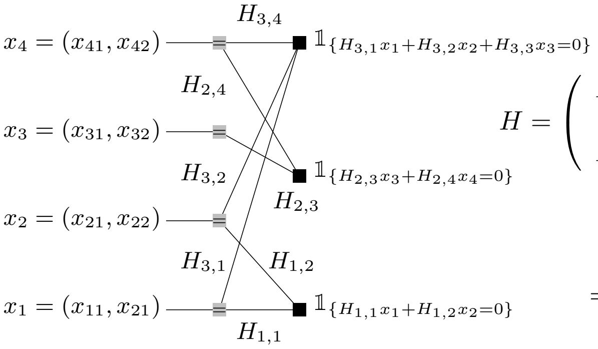


Figure 2.1: Performance of regular $(2, 3)$ ensemble with alphabet size 2^m over BAWGNC(σ) with binary length 4320.

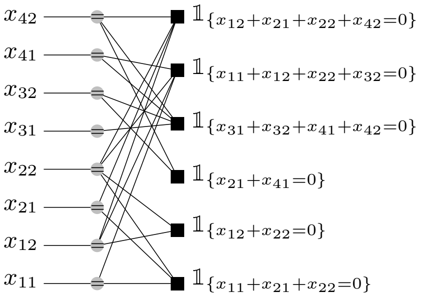
To achieve capacity the standard approach is to fix the alphabet size (to binary) and to increase the density of the underlying graph. We take an alternative route here. Suppose we fix the degree distribution and let the alphabet size increase. Let us consider an example. Fig. 2.2 shows the Forney-Style Factor Graph (FSFG [14]) of a simple code over GF(4) together with its corresponding parity-check matrix. The addition and multiplication operations are performed modulo the polynomial $1+z+z^2$. The parity-check matrix is equivalent to the binary parity-check matrix shown in Fig. 2.3. E.g., the constraint over GF(4), $x_1 + (1 + z)x_2 = 0$ is equivalent to the two binary constraints $x_{11} + x_{21} + x_{22} = 0$ and $x_{12} + x_{22} = 0$ and so on. The corresponding binary FSFG and the binary parity-check matrix are shown in Fig. 2.3. Again, we can find an equivalent binary code corresponding to a code defined with respect to the general linear group. The general linear group over the binary field of dimension m is the set of all $m \times m$ invertible matrices. For example, the following is a parity-check matrix over the general linear group of dimension



$$H = \begin{pmatrix} H_{1,1} & H_{1,2} & 0 & 0 \\ 0 & 0 & H_{2,3} & H_{2,4} \\ H_{3,1} & H_{3,2} & 0 & H_{3,4} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 1+z & 0 & 0 \\ 0 & 0 & z & z \\ 1+z & z & 0 & 1 \end{pmatrix}$$

Figure 2.2: The FSFG of a simple code over $\mathrm{GF}(4)$ and its associated parity-check matrix H.


x_{42}
x_{41}
x_{32}
x_{31}
x_{22}
x_{21}
x_{12}
x_{11}
$\mathbb{1}_{\{x_{12}+x_{21}+x_{22}+x_{42}=0\}}$
$\mathbb{1}_{\{x_{11}+x_{12}+x_{22}+x_{32}=0\}}$
$\mathbb{1}_{\{x_{31}+x_{32}+x_{41}+x_{42}=0\}}$
$\mathbb{1}_{\{x_{21}+x_{41}=0\}}$
$\mathbb{1}_{\{x_{12}+x_{22}=0\}}$
$\mathbb{1}_{\{x_{11}+x_{21}+x_{22}=0\}}$

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

Figure 2.3: The FSFG and its associated parity-check matrix corresponding to the equivalent binary code of the code given in Fig. 2.2.

two:

$$H = \begin{pmatrix} H_{1,1} & H_{1,2} & 0 & 0 \\ 0 & 0 & H_{2,3} & H_{2,4} \\ H_{3,1} & 0 & 0 & H_{3,4} \end{pmatrix}$$

where,

$$H_{1,1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad H_{1,2} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \quad H_{2,3} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix},$$

$$H_{2,4} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad H_{3,1} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \quad H_{3,4} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Its equivalent binary matrix is given by:

$$H_b = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

Each constraint in the matrix H is equivalent to two constraints in the equivalent binary matrix H_b. E.g., the constraint $H_{1,1}x_1 + H_{1,2}x_2 = 0$ is equivalent to $x_{11} + x_{21} + x_{22} = 0$ and $x_{12} + x_{21} = 0$.

One might hope that by increasing the alphabet size the equivalent binary matrix becomes better and better. Thus an increase in the alphabet size might yet yield another way of achieving capacity. As we will see, the relationship between alphabet size and performance is not a simple one and an increase in the underlying alphabet does not necessarily lead to increased performance. Nevertheless, there are many unexplored degrees of freedom in the system design.

The possibility of using non-binary alphabets for LDPC codes was already proposed by Gallager in his landmark PhD thesis [3]. The fact that by using non-binary alphabets, the performance of LDPC codes over BMS channels can be improved was first reported by Davey and MacKay [4]. They showed by specific examples that NBLDPC codes can perform significantly better than their binary counterparts for the BMS channels. Hu showed that even the performance of cycle codes can be improved considerably with non-binary alphabets [5]. In [15], Sridhara and Fuja have designed codes over certain rings and groups for coded modulation based on the principle of NBLDPC codes. Also, the design of LDPC code construction using “lifting” of multi-edge type designs [16, 17] and the related framework of protographs [18] can be seen as NBLDPC ensembles. In [19] Bennatan and Burshtein presented an analysis of random coset NBLDPC codes over arbitrary discrete-memoryless channels. They generalized the notion of symmetry and proved stability condition for these codes. Also they presented an EXIT chart analysis based on Gaussian approximation for random coset codes. Declercq and Fossorier presented various message passing decoding algorithms for NBLDPC codes in [20]. An analysis of tail-biting trellis based on NBLDPC codes was presented in [21] by Andriyanova and Tillich.

2.2 Preliminaries and Notation

We consider transmission over a BMS channel using NBLDPC ensembles. We denote the set of symbols of the codeword by $\mathcal{S}$. As we consider transmission over a binary-input channel it is natural to require that the cardinality of $\mathcal{S}$ is a power of 2. We denote the cardinality of $\mathcal{S}$ by $|\mathcal{S}| = q = 2^m$. Thus we can think of each symbol as a binary m-tuple. In order to transmit a symbol over a BMS channel we transmit the bits representing this symbol. We define the NBLDPC ensemble in an analogous way as in the binary case. First we define an ensemble of bipartite graphs. It is defined in terms of a pair of *degree distributions* (d.d.), $(\lambda(x) = \Sigma_i \lambda_i x^{i-1}, \rho(x) = \Sigma_j \rho_j x^{j-1})$, shorthand (λ, ρ). The coefficient λ_i (ρ_j) denotes the fraction of the total number of edges connected to a variable (check) node of degree i (j). A variable (check) node of degree i (j) has i (j) sockets for its connected edges. Given a pair (λ, ρ) of d.d. and the block length n, an unlabeled *ensemble* of bipartite graphs $\mathbb{G}(n, \lambda, \rho)$ is defined in the following manner. Label the sockets on the variable and check node side in an arbitrary but fixed way. Consider a permutation π on the number of edges. Join the socket i on the variable node side to the socket $\pi(i)$ on the check node side. The ensemble $\mathbb{G}(n, \lambda, \rho)$ contains all such bipartite graphs generated by all the possible permutations on the number of edges with uniform probability. Now we assign a bijective linear mapping $f : \mathcal{S} \mapsto \mathcal{S}$ to each edge of every bipartite graph in the ensemble $\mathbb{G}(n, \lambda, \rho)$. The mappings are chosen uniformly at random from a set of mappings $\mathcal{F}$. The resulting ensemble of labeled bipartite graphs is the NBLDPC ensemble. The check nodes of a bipartite graph from the NBLDPC ensemble represent parity-check equations of the form

$$\sum_i f_i(x_i) = 0, \tag{2.1}$$

where the $\{x_i\}, x_i \in \mathcal{S}$, are the variables which participate in the parity check equation. Note that a code defined by parity-check equations of the form as in Eqn(2.1) is linear as the mappings f_i are linear. The *design rate* of an ensemble with d.d. (λ, ρ) is the same as in the binary case:

$$r = 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx}.$$

Let $x = (x_1, \ldots, x_n)$ denote a codeword over $\mathcal{S}$. We can think of each symbol as a binary m-tuple. Hence, we can equivalently think of the codeword as a binary codeword of length nm, $x = (x_{11}, \ldots, x_{1m}, \ldots, x_{n1}, \ldots, x_{nm})$. To transmit the codeword x over a BMS channel, we transmit its binary components and let the corresponding received word be $y = (y_{11}, \ldots, y_{1m}, \ldots, y_{n1}, \ldots, y_{nm})$.

In this thesis we will consider two variants of NBLDPC ensembles: Ensembles over finite fields and ensembles over the general linear group. For ensembles over finite fields, $\mathcal{S} = \text{GF}(2^m)$, and the mappings f are of the form $f(x) = \omega x$, where $\omega \in \text{GF}^*(2^m)$, the multiplicative group of $\text{GF}(2^m)$. Hence, by some abuse of notation $\mathcal{F} = \text{GF}^*(2^m)$. This implies that $|\mathcal{F}| = 2^m - 1$. We will denote an LDPC ensemble of blocklength n over $\text{GF}(2^m)$ with d.d. (λ, ρ) by $\text{EGF}(n, \lambda, \rho, m)$. The ensemble over the general linear group is denoted by $\text{EGL}(n, \lambda, \rho, m)$. We denote the general linear group over the binary field of dimension m by GL_2^m. GL_2^m is the set of all $m \times m$ invertible matrices over the binary field. Note that the number of distinct invertible matrices over the binary field is $\prod_{l=0}^{m-1} (2^m - 2^l)$, [22]. The symbol set $\mathcal{S}$ of ensemble $\text{EGL}(n, \lambda, \rho, m)$ is the vector space GF_2^m of dimension m over the binary field.

The mappings on the edges are given by $f(b) = Wb$, where $b \in \mathrm{GF}_2^m$ and $W \in \mathrm{GL}_2^m$. We will be interested in the number of distinct subspaces of dimension k of the vector space GF_2^m. This number is known as the Gaussian binomial coefficient. We denote it by $\begin{bmatrix} m \\ k \end{bmatrix}$, and it is given by (see [22], pp. 443):

$$\begin{bmatrix} m \\ k \end{bmatrix} = \begin{cases} 1, & \text{if} \quad k = 0 \ \text{ or } \ k = m, \\ \prod_{l=0}^{k-1} \frac{2^m - 2^l}{2^k - 2^l}, & \text{otherwise.} \end{cases} \tag{2.2}$$

For a subspace V of the vector space GF_2^m, we denote its dual subspace by $\mathrm{V}^\perp$. The dual subspace is defined as:

$$\mathrm{V}^\perp = \left\{ \beta \in V^\perp : \sum_{i=1}^{m} \alpha_i \beta_i = 0, \forall \alpha \in \mathrm{V} \right\},$$

where the summation is with respect to the binary field. Note that the dimension of V and $\mathrm{V}^\perp$ satisfies the following relation [23]:

$$\dim(\mathrm{V}) + \dim\left(\mathrm{V}^\perp\right) = m.$$

The dual of the sum of two subspaces V_1 and V_2 is equal to the intersection of $\mathrm{V}_1^\perp$ and $\mathrm{V}_2^\perp$ i.e.

$$(\mathrm{V}_1 + \mathrm{V}_2)^\perp = \mathrm{V}_1^\perp \cap \mathrm{V}_2^\perp.$$

If a subspace V has the basis vectors $B = \{b_1, \dots, b_k\}$, then we denote the set of basis vectors for $\mathrm{V}^\perp$ by $B^\perp = \{b_1^\perp, \dots, b_k^\perp\}$. We denote the linear span of a set of vectors B by $\mathrm{Span}(B)$. We denote the null space of a matrix M by $\mathrm{nullspace}(M)$ and its row space by $\mathrm{rowspace}(M)$.

2.3 Belief Propagation Decoder

The messages in the belief propagation decoder are vectors of length $|\mathcal{S}|$. The α^{th} component of a message Ψ, denoted by $\psi(\alpha)$ where $\alpha \in \mathcal{S}$, is equal to the posteriori probability that the symbol is α. An iteration of the BP algorithm consists of four steps. The first step corresponds to processing the messages at the variable node side. The second step processes the messages according to edge labels. The third step is the check node side operation. Finally, the fourth step transforms the messages directed towards variable nodes according to edge labels. Note that in the first iteration the variable node processing corresponds to sending messages based on channel observations. We now describe these steps in detail.

1. **Initial Message:** The initial message $\Psi_{x,\mathtt{c}}^{(1,1)}$ from a variable node x to a connected check node $\mathtt{c}$ is the posteriori probability distribution of symbols $(\mathrm{P}(x = \alpha_0|y), \dots, \mathrm{P}(x = \alpha_{q-1}|y))$, where $\alpha_i \in \mathcal{S}$, $i = 0, \dots, q-1$. Note that the first superscript in the message $\Psi_{x,\mathtt{c}}^{(1,1)}$ represents the

iteration number and the second superscript represents the step number in that iteration.

2. **Edge Action:** Before the messages reach the check nodes, we need to consider the permutations induces by the edge labels and their associated mapping $f(x)$. Note that labels induce permutations since the mappings f are invertible. More precisely, if the edge label is f then the message vector $\Psi_{x,\mathtt{c}}^{(l,1)} = \left(\psi_{x,\mathtt{c}}^{(l,1)}(\alpha_0), \ldots, \psi_{x,\mathtt{c}}^{(l,1)}(\alpha_{q-1})\right)$ gets permuted to the message $\Psi_{x,\mathtt{c}}^{(l,2)} = \left(\psi_{x,\mathtt{c}}^{(l,1)}\left(f^{-1}(\alpha_0)\right), \ldots, \psi_{x,\mathtt{c}}^{(l,1)}\left(f^{-1}(\alpha_{q-1})\right)\right)$.

3. **Check Node Action:** The operation on the check node side is the convolution of the incoming messages. Lets consider a check node $\mathtt{c}$. Let its degree be 3 for the sake of simplicity. Let x, y, and z be the connected variables and lets consider the outgoing message along the edge to z as a function of the incoming messages along the edges connected to x and y. The outgoing message towards variable z is then

$$\begin{aligned} \psi_{\mathtt{c},z}^{(l,3)}(\alpha) &= \sum_{\beta,\gamma\in\mathcal{S}} \mathbb{1}_{\{\alpha+\beta+\gamma=0\}} \psi_{x,\mathtt{c}}^{(l,2)}(\beta)\psi_{y,\mathtt{c}}^{(l,2)}(\gamma) \\ &= \sum_{\beta\in\mathcal{S}} \psi_{x,\mathtt{c}}^{(l,2)}(\beta)\psi_{y,\mathtt{c}}^{(l,2)}(-\alpha-\beta), \end{aligned} \tag{2.3}$$

where,

$$\mathbb{1}_{\{\delta=0\}} = \begin{cases} 1, & \text{if} \quad \delta = 0, \\ 0, & \text{otherwise.} \end{cases}$$

In a brute force manner, the above summation can be accomplished with complexity $O(q^2)$. However, note that $\psi_{\mathtt{c},z}^{(l,3)}(-\alpha)$ is given in terms of a *convolution* of two message vectors, where the index calculations are done with respect to the additive group of GF (2^m). Note that the vector space GF_2^m and finite field GF (2^m) are isomorphic groups with respect to addition. Hence, as suggested in [24, 10], we can use Fourier transforms to accomplish this convolution in an efficient manner. Since the message size is 2^m, the Fourier transform is particularly simple. Write an element of $\mathcal{S}$ as an m-tuple with components in GF(2), $\alpha = (\alpha_1, \ldots, \alpha_m)$. Let $\Psi = \left(\psi(\alpha)_{\alpha\in\mathcal{S}}\right)$ denote a vector whose components are taking values in $\mathbb{C}$. Let $\Phi = (\phi(\alpha))_{\alpha\in\mathcal{S}}$ denote its Fourier transform. The corresponding Fourier transform pair is

$$\phi(\alpha) = \sum_{\beta} \psi(\beta)(-1)^{-\alpha.\beta^T}, \quad \psi(\alpha) = \frac{1}{2^m}\sum_{\beta}\phi(\beta)(-1)^{\alpha.\beta^T}, \tag{2.4}$$

where $\alpha.\beta^T$ is the dot product of the binary representation of α and β. Thus the check node operation is given by the component wise multiplication of the Fourier transform of incoming messages and by taking the inverse Fourier transform of the result.

4. **Inverse Edge Action:** Messages are again permuted to take care of the mappings. But in this case the permutation is the inverse of the permutation in step 2. More precisely, if the edge label is f then the message vector $\Psi_{\mathsf{c},x}^{(l,3)} = \left(\psi_{\mathsf{c},x}^{(l,3)}(\alpha_0), \ldots, \psi_{\mathsf{c},x}^{(l,3)}(\alpha_{q-1})\right)$ gets permuted to the message $\Psi_{\mathsf{c},x}^{(l,4)} = \left(\psi_{\mathsf{c},x}^{(l,3)}(f(\alpha_0)), \ldots, \psi_{\mathsf{c},x}^{(l,3)}(f(\alpha_{q-1}))\right)$.

5. **Variable Node Action:** The messages $\Psi^{(l+1,1)}$ from the variable nodes are the component-wise multiplication of the incoming messages $\Psi^{(l,4)}$ and the initial message $\Psi^{(1,1)}$. In order to represent the result as a probability distribution, we normalize the result.

2.4 Density Evolution for the BEC

In this section we analysis the asymptotic performance of the BP decoder for NBLDPC codes when transmission takes place over the BEC. In order to simplify the analysis, we show as in [10] that the all-zero codeword assumption holds. This implies that the error probability is the same for the BP decoder in the cases when the all-zero codeword is transmitted and when any other codeword is transmitted. We prove this in the following lemma for any BMS channel.

Lemma 1. *Consider transmission over a BMS channel using an NBLDPC code. Then the conditional symbol error probability of the BP decoder is independent of the transmitted codeword.*

Proof. We use the BPSK mapping for the transmission of bits: $\{0 \leftrightarrow 1, 1 \leftrightarrow -1\}$. Under the BPSK mapping any BMS channel can be represented in the product form $Y_t = X_t Z_t$, where X_t is the transmitted bit at time t, Z_t is the noise and Y_t is the output of the channel. Note that under the BPSK mapping the original all-zero codeword is mapped into the all-one codeword. By some abuse of notation we continue to speak about the "all-zero" codeword. The addition of two symbols $\alpha_i = \{\alpha_{i1}, \ldots, \alpha_{ism}\}$ and $\alpha_j = \{\alpha_{j1}, \ldots, \alpha_{jm}\}$ over GF(2) is $\alpha_i \oplus \alpha_j = \{\alpha_{i1} \oplus \alpha_{j1}, \ldots, \alpha_{im} \oplus \alpha_{jm}\}$. This is equivalent to the bit-wise product $\alpha_i \alpha_j = \{\alpha_{i1}\alpha_{j1}, \ldots, \alpha_{im}\alpha_{jm}\}$ after the BPSK mapping.

We would like to compare the progress of the iterative decoder when the all-zero codeword and a codeword $x = \{x_1, \ldots, x_n\}$ is transmitted assuming that the noise realization is the same in both cases. For a BMS channel

$$P(y_{ij}|x_{ij}) = P(-y_{ij}| - x_{ij}). \tag{2.5}$$

This implies $P(y_{ij}|x_{ij}) = P(y_{ij}x_{ij}|1)$. Hence, for the same noise realization

$$P(\mathbf{y}|1) = \prod_{i=1}^{n}\prod_{j=1}^{m} P(Z_{ij}|1), \quad P(\mathbf{y}|x) = \prod_{i=1}^{n}\prod_{j=1}^{m} P(x_{ij}Z_{ij}|1).$$

We show a one-to-one mapping between the messages flowing along the edges in the two cases. Consider a variable node x, an edge e emanating from x, and the corresponding check node $\mathtt{c}$. We assume that the mapping f is associated to the edge e. We recall that $f : \mathcal{S} \to \mathcal{S}$ is one-to-one and linear, i.e., for $\alpha, \beta \in \mathcal{S}$

$$f(\alpha \oplus \beta) = f(\alpha) \oplus f(\beta). \tag{2.6}$$

As f is linear, $f(0) = 0$. We denote the messages of the decoder corresponding to all-zero case by $\Gamma = (\gamma(\alpha_0), \ldots, \gamma(\alpha_{q-1}))$ and the messages for decoding the other codeword are denoted by $\Psi = (\psi(\alpha_0), \ldots, \psi(\alpha_{q-1}))$.

1. The initial message with respect to the $\{\pm 1\}$ mapping of the bits is given by

$$\psi_{x,\mathtt{c}}^{(1,1)}(\alpha) = \mathrm{P}(\alpha)\frac{\mathrm{P}(xZ|\alpha)}{\sum_{\beta\in\mathcal{S}} \mathrm{P}(\beta)\,\mathrm{P}(xZ|\beta)} \overset{x\beta=\beta'}{=} \mathrm{P}(\alpha)\frac{\mathrm{P}(x\alpha Z|1)}{\sum_{\beta'\in\mathcal{S}} \mathrm{P}(x\beta')\,\mathrm{P}(\beta' Z|1)},$$

   and similarly

$$\gamma_{x,\mathtt{c}}^{(1,1)}(\alpha) = \mathrm{P}(\alpha)\frac{\mathrm{P}(\alpha Z|1)}{\sum_{\beta'\in\mathcal{S}} \mathrm{P}(\beta')\,\mathrm{P}(\beta' Z|1)}.$$

   We have $\mathrm{P}(x\beta') = \mathrm{P}(\beta')$ as we assume that all codewords have equal probability and the code is linear. Hence $\psi_{x,\mathtt{c}}^{(1,1)}(\alpha) = \gamma_{x,\mathtt{c}}^{(1,1)}(x\alpha)$. In the $\{0,1\}$ mapping, this translates to $\psi_{x,\mathtt{c}}^{(1,1)}(\alpha) = \gamma_{x,\mathtt{c}}^{(1,1)}(x \oplus \alpha)$. For the rest of the proof we use the $\{0,1\}$ mapping of the bits.

2. In the l^{th} iteration after the edge action the message satisfies

$$\psi_{x,\mathtt{c}}^{(l,2)}(\alpha) = \psi_{x,\mathtt{c}}^{(l,1)}\left(f^{-1}(\alpha)\right), \quad \gamma_{x,\mathtt{c}}^{(l,2)}(\alpha) = \gamma_{x,\mathtt{c}}^{(l,1)}\left(f^{-1}(\alpha)\right).$$

   As $\psi_{x,\mathtt{c}}^{(l,1)}(\alpha) = \gamma_{x,\mathtt{c}}^{(l,1)}(x \oplus \alpha)$, and using the linearity of f described in (2.6), we get

$$\psi_{x,\mathtt{c}}^{(l,2)}(\alpha) = \gamma_{x,\mathtt{c}}^{(l,2)}(\alpha \oplus f(x)). \tag{2.7}$$

3. After the check node step we look at the message coming towards the variable node x. Let $x_1, \ldots, x_{\mathtt{r}-1}$ be the other variable nodes connected to the check node $\mathtt{c}$, with $f_1, \ldots, f_{\mathtt{r}-1}$ denoting the mappings associated to the edges. The β^{th} component of the Fourier transform of the message $\Psi_{\mathtt{c},x}^{(l,3)}$ is the component-wise multiplication of the β^{th} components of the Fourier transforms of $\Psi_{x_1,\mathtt{c}}^{(l,2)}, \ldots, \Psi_{x_{\mathtt{r}-1},\mathtt{c}}^{(l,2)}$. By taking the inverse Fourier

transform we get

$$\begin{aligned}
\psi^{(l,3)}_{\mathtt{c},x}(\alpha) &= \frac{1}{2^m}\sum_{\beta}(-1)^{\beta.\alpha^T}\prod_{i=1}^{\mathtt{r}-1}\underbrace{\sum_{\beta_i}(-1)^{\beta.\beta_i^T}\psi^{(l,2)}_{x_i,\mathtt{c}}(\beta_i)}_{\beta^{\text{th}}\text{ component of the FT of }\Psi^{(l,2)}_{x_i,\mathtt{c}}}, \\
&\overset{(a)}{=} \frac{1}{2^m}\sum_{\beta,\beta_1,\dots,\beta_{\mathtt{r}-1}}\prod_{i=1}^{\mathtt{r}-1}(-1)^{\beta.(\alpha\oplus\sum_{i=1}^{\mathtt{r}-1}\beta_i)^T} \\
&\qquad \gamma^{(l,2)}_{x_i,\mathtt{c}}(\beta_i\oplus f_i(x_i)), \\
&\overset{\beta_i\equiv\beta_i\oplus f_i(x_i)}{=} \frac{1}{2^m}\sum_{\beta,\beta_1,\dots,\beta_{\mathtt{r}-1}}(-1)^{\beta.(\alpha\oplus\sum_{i=1}^{\mathtt{r}-1}(\beta_i\oplus f_i(x_i)))} \\
&\qquad \prod_{i=1}^{\mathtt{r}-1}\gamma^{(l,2)}_{x_i,\mathtt{c}}(\beta_i), \\
&\overset{\sum f(x_i)=f(x)}{=} \frac{1}{2^m}\sum_{\beta,\beta_1,\dots,\beta_{\mathtt{r}-1}}(-1)^{\beta.(\alpha\oplus f(x)\oplus\sum_{i=1}^{\mathtt{r}-1}\beta_i)^T} \\
&\qquad \prod_{i=1}^{\mathtt{r}-1}\gamma^{(l,2)}_{x_i,\mathtt{c}}(\beta_i).
\end{aligned}$$

where in (a) we use (2.7). For $x=0$, the above equation gives

$$\gamma^{(l,3)}_{\mathtt{c},x}(\alpha) = \frac{1}{2^m}\sum_{\beta,\beta_1,\dots,\beta_{\mathtt{r}-1}}(-1)^{\beta.(\alpha\oplus\sum_{i=1}^{\mathtt{r}-1}\beta_i)^T}\prod_{i=1}^{\mathtt{r}-1}\gamma^{(l,2)}_{x_i,\mathtt{c}}(\beta_i).$$

Hence, we see that

$$\psi^{(l,3)}_{\mathtt{c},x}(\alpha) = \gamma^{(l,3)}_{\mathtt{c},x}\left(f(x)\oplus\alpha\right). \tag{2.8}$$

4. After the inverse edge action,

$$\psi^{(l,4)}_{\mathtt{c},x}(\alpha) = \psi^{(l,3)}_{\mathtt{c},x}\left(f(\alpha)\right), \quad \gamma^{(l,4)}_{\mathtt{c},x}(\alpha) = \gamma^{(l,3)}_{\mathtt{c},x}\left(f(\alpha)\right).$$

As $\psi^{(l,3)}_{\mathtt{c},x}(\alpha) = \gamma^{(l,3)}_{\mathtt{c},x}\left(f(x)\oplus\alpha\right)$ and using the linearity of f, we get

$$\psi^{(l,4)}_{\mathtt{c},x}(\alpha) = \gamma^{(l,4)}_{\mathtt{c},x}(\alpha\oplus x). \tag{2.9}$$

5. At the variable node we perform component-wise multiplication of all the incoming messages and the initial message. All of them satisfy the shift property of (2.9). Hence

$$\psi^{(l+1,1)}_{x,\mathtt{c}}(\alpha) = \gamma^{(l+1,1)}_{x,\mathtt{c}}(x\oplus\alpha).$$

This implies that $\psi^{(l+1,1)}_{x,\mathtt{c}}(x) = \gamma^{(l+1,1)}_{x,\mathtt{c}}(0)$. Thus the probability of correct decoding is the same at the completion of an iteration of message passing.

Hence we can analyze the performance of the BP decoder assuming that the all-zero codeword is transmitted. □

2.4.1 Density Evolution for EGL (λ, ρ, m)

In order to derive the density evolution equations for the ensemble EGL (λ, ρ, m) we need to find the set of messages which arise in the BP decoder. In the following lemma we characterize all the messages which appear in the BP decoder.

Lemma 2 (Message Space Characterization)**.** *Consider transmission over the BEC using the NBLDPC ensemble EGL (λ, ρ, m). The messages arising in the BP decoder satisfy the following properties:*

1. *All the non-zero entries in a message Ψ are equal.*
2. *Let $V = \{\alpha \in \mathrm{GF}_2^m : \psi(\alpha) \neq 0\}$. Then V is subspace of GF_2^m.*
3. *The Fourier transform Φ of a message Ψ has the property:*

$$\phi(\alpha) = \begin{cases} 1, & \textit{if } \alpha \in V^{\perp}, \\ 0, & \textit{otherwise}, \end{cases}$$

where $V^{\perp}$ is the dual subspace of V. Thus the total number of messages is equal to $\sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix}$.

Proof. We prove that messages satisfy these properties at every step of an iteration.

1. Consider a variable node in the bipartite graph. Without loss of generality assume that its initial k bits are erased. The posteriori probability of a symbol α in the initial message $\Psi^{(1)}$ is $\frac{1}{2^k}$ if its bit representation satisfies the equations $\alpha_{k+1} = 0, \ldots, \alpha_m = 0$. We write these system of equations in matrix form as $Mu = 0$, where

$$M_{ij} = \begin{cases} 1, & \text{if} \quad i = j \quad \text{and} \quad i > k, \\ 0, & \qquad \text{otherwise.} \end{cases}$$

Thus all the non-zero elements of message Ψ are equal and they belong to V = nullspace(M). Hence the claim 1 and 2 of the lemma are true for the initial message. Recall that the α^{th} component of the Fourier transform of a message is give by

$$\phi(\alpha) = \sum_{\beta \in \mathrm{GF}_2^m} (-1)^{\alpha.\beta^T} \psi(\beta) = \sum_{\beta \in \mathrm{V}} (-1)^{\alpha.\beta^T} \frac{1}{2^k}. \tag{2.10}$$

Clearly from (2.10), if $\alpha \in \mathrm{V}^{\perp}$ then $\alpha.\beta^T = 0$ for $\forall \beta \in \mathrm{V}$. Thus all the terms add up and we have $\phi(\alpha) = 1$. If $\alpha \notin \mathrm{V}^{\perp}$, then $(-1)^{\alpha.\beta^T} = 1$ for half the terms in the summation in (2.10) and for the other half $(-1)^{\alpha.\beta^T} = -1$. Thus $\phi(\alpha) = 0$ for $\alpha \notin \mathrm{V}^{\perp}$.

2. Assume that the message is acted on by an edge with edge label W. Let $u' = Wu$ or $u = W^{-1}u'$. Thus $MW^{-1}u' = 0$. This implies that after the edge action, the non zero entries of the permuted message correspond to the solution space of $MW^{-1}u' = 0$. Hence again by the same reasoning as in the previous step, all the three claims of the lemma hold. Note that the dimension of the nullspace(M) is equal to the dimension of the nullspace$\left(MW^{-1}\right)$.

3. Now at the check node side, we take component-wise multiplication of the Fourier transform of the incoming messages. For the sake of simplicity consider a check node of degree three. Let $\mathrm{V}_1^\perp, \mathrm{V}_2^\perp$ be subspaces corresponding to the two Fourier transform. So the subspace corresponding to the multiplication will be $\mathrm{V}_1^\perp \cap \mathrm{V}_2^\perp$. Now again by the same reasoning as in the step 1, all the three claims hold. Note that $\left(\mathrm{V}_1^\perp \cap \mathrm{V}_2^\perp\right)^\perp = \mathrm{V}_1 + \mathrm{V}_2$. So the operation on the check node side corresponds to taking the linear span of the subspaces corresponding to incoming messages.

4. The proof for inverse edge action is the same as the edge action (step 2).

5. At the variable node we have the component-wise multiplication of the incoming messages. For the sake of simplicity consider a degree two variable node. Let V_1 be the subspace corresponding the incoming message and V be the subspace corresponding to the initial message. The resultant message will have the subspace of non-zero entries as $\mathrm{V} \cap \mathrm{V}_1$. Again all the three claims of the lemma hold.

This characterizes the message space and there are $\sum_{i=0}^{m} \begin{bmatrix} m \\ i \end{bmatrix}$ different messages as there are $\begin{bmatrix} m \\ i \end{bmatrix}$ different messages of dimension i. □

Motivated by Lemma 2 we introduce the following terminology. We say that the dimension of a message Ψ, call it $\dim(\Psi)$, is k if the number of non-zero entries of Ψ is 2^k. We associate to a message Ψ of dimension k, an $(m-k) \times m$ matrix M and its subspace V. We have the following relations,

$$\mathrm{V} = \text{nullspace}\,(M)\,, \qquad \mathrm{V}^\perp = \text{rowspace}\,(M)\,.$$

With the aid of Lemma 2, we can now write down the density evolution equations for $\mathrm{EGL}\,(\lambda, \rho, m)$. Since we are choosing all the elements of GL_2^m uniformly, so after the edge action any message of dimension k is mapped to any other message of dimension k with uniform probability. More precisely, consider a message Ψ of dimension k and choose an element of GL_2^m (edge label) uniformly at random. It is not very hard to see that the edge action will map the message Ψ with uniform probability to any of the $\begin{bmatrix} m \\ k \end{bmatrix}$ messages of dimension k. It follows that we need not keep track of the probability of each message but that it suffices to keep track of the total probability of the class of all messages of dimension k. Since a message can have $m+1$ possible dimensions, density evolution can be expressed as an $m+1$ dimensional recursion. Note that a correct decoding implies that the probability of the subspace of zero dimension (it contains only the zero element) converges to one.

In the next lemma, by using the arguments of [10] we show that the empirical densities of the messages concentrate around the density evolution equations.

Lemma 3. *Consider transmission over the BEC using an element of the regular ensemble* $EGL\left(n, \mathtt{l}, \mathtt{r}, m\right)$. *Over the probability space of the elements in* $EGL\left(n, \mathtt{l}, \mathtt{r}, m\right)$ *and all the channel realizations, let* Z_i *denote the number of edges carrying a message of dimension* i *after variable node operation in the* l^{th} *iteration. Let* $P_v^{(l)}\left(i\right)$ *be the probability under the tree assumption that a randomly chosen edge is carrying a message of dimension* i *after the variable node operation. Then, there exist positive constants* $\beta = \beta\left(\mathtt{l}, \mathtt{r}, m, l\right)$ *and* $\gamma = \gamma\left(\mathtt{l}, \mathtt{r}, m, l\right)$ *such that*

1. *For any* $\epsilon > 0$ *we have*

$$Pr\left\{\left|Z_i - \mathbb{E}\left[Z_i\right]\right| > n\,\mathtt{l}\epsilon/2\right\} \leq 2e^{-\beta\epsilon^2 n}. \tag{2.11}$$

2. *For any* $\epsilon > 0$ *and* $n > \frac{2m\gamma}{\epsilon}$ *we have*

$$\left|\mathbb{E}\left[Z_i\right] - n\,\mathtt{l}\,P_v^{(l)}\left(i\right)\right| < n\,\mathtt{l}\epsilon/2. \tag{2.12}$$

3. *For any* $\epsilon > 0$ *and* $n > \frac{2m\gamma}{\epsilon}$ *we have*

$$Pr\left\{\left|Z_i - n\,\mathtt{l}\,P_v^{(l)}\left(i\right)\right| > n\,\mathtt{l}\epsilon\right\} \leq 2e^{-\beta\epsilon^2 n}. \tag{2.13}$$

Proof. The proof is very similar to proof of [10, Thm 2]. Hence we will be brief here. Note that Eqn(2.13) is an immediately consequence of Eqn(2.11) and Eqn(2.12). The proof of Eqn(2.12) is exactly the same as the proof of the corresponding claim in the binary setting proved in [10, Thm 2].

In order to prove Eqn(2.11) we define a sequence of equivalence relations on Δ, the probability space which consist of the tuple (G, R), where G is the graph and R is the input to the decoder. Let $=_j$, $0 \leq j \leq \left(2\mathtt{l}+1\right)n =: k$ be a sequence of equivalence relations on Δ which satisfies that if $(G', R') =_j \left(G'', R''\right)$ then $(G', R') =_{j-1} \left(G'', R''\right)$. Next we define $Z_i^0, Z_i^1, \ldots, Z_i^m$ by

$$Z_i^j(G, R) := \mathbb{E}\left[Z(G', R')|(G', R') =_j (G, R)\right].$$

By construction $Z_0, Z_1, \ldots, Z_k$ is a Doob's Martingale. The bounds on the difference $|Z_i^{j+1}(G, R) - Z_i^j(G, R)|$ are the same as that in [10, Thm 2] when we reveal the socket connection and decoder input. When we reveal the edge label, the dimension of the message does not changes by changing the label. Hence the constants during the process of revealing the labels are zero. Thus we prove Eqn(2.11). □

In the next lemma we derive the density evolution equations for the ensemble $\mathrm{EGL}(\lambda, \rho, m)$.

Lemma 4 (Density Evolution for $\mathrm{EGL}(\lambda, \rho, m)$)**.** *Consider the NBLDPC ensemble $EGL(\lambda, \rho, m)$. Let $P_v^{(l)}(k, \mathfrak{l})$ be the probability that a randomly chosen message is of dimension k after the edge action connected to a variable node of degree $\mathfrak{l}$ (i.e., a message just before the check node processing). Similarly, $P_c^{(l)}(k, \mathfrak{r})$ denotes the probability that a randomly chosen message is of dimension k after the inverse edge action connected to a check node of degree $\mathfrak{r}$ (i.e., a message just before the variable node processing). Then we have the following recursive relationships between different probabilities on the check node side:*

$$P_c^{(l)}(k,3) = \sum_{i=0}^{k} P_v^{(l)}(i) \sum_{j=k-i}^{k} \frac{\begin{bmatrix} m-i \\ m-k \end{bmatrix} \begin{bmatrix} i \\ k-j \end{bmatrix} 2^{(k-i)(k-j)}}{\begin{bmatrix} m \\ m-j \end{bmatrix}} P_v^{(l)}(j), \tag{2.14}$$

$$P_c^{(l)}(k,\mathfrak{r}) = \sum_{i=0}^{k} P_c^{(l)}(i, \mathfrak{r}-1) \sum_{j=k-i}^{k} \frac{\begin{bmatrix} m-i \\ m-k \end{bmatrix} \begin{bmatrix} i \\ k-j \end{bmatrix} 2^{(k-i)(k-j)}}{\begin{bmatrix} m \\ m-j \end{bmatrix}} P_v^{(l)}(j), \tag{2.15}$$

where $P_v^{(l)}(i)$ is the average over the variable node degree distribution,

$$P_v^{(l)}(i) = \sum_{\mathfrak{l}} \lambda_{\mathfrak{l}} P_v^{(l)}(i, \mathfrak{l}).$$

The equations on the variable node side for the probabilities in $(l+1)^{th}$ iteration are:

$$P_v^{(l+1)}(k,2) = \sum_{i=k}^{m} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \sum_{j=k}^{m-i+k} \frac{\begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} 2^{(i-k)(j-k)}}{\begin{bmatrix} m \\ j \end{bmatrix}} P_c^{(l)}(j), \tag{2.16}$$

$$P_v^{(l+1)}(k,\mathfrak{l}) = \sum_{i=k}^{m} P_v^{(l+1)}(i, \mathfrak{l}-1) \sum_{j=k}^{(m-i+k)} \frac{\begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} 2^{(i-k)(j-k)}}{\begin{bmatrix} m \\ j \end{bmatrix}} P_c^{(l)}(j), \tag{2.17}$$

where $P_c^{(l)}(j)$ is the average over the variable node degree distribution,

$$P_c^{(l)}(j) = \sum_{\mathfrak{r}} \rho_{\mathfrak{r}} P_c^{(l)}(j, \mathfrak{r}).$$

Proof. Consider a check node of degree 3. We know from Lemma 2 that the subspace V corresponding to the resulting message Ψ is the sum of subspaces V_1 and V_2 corresponding to the incoming messages Ψ_1 and Ψ_2, i.e., $\mathrm{V} = \mathrm{V}_1 + \mathrm{V}_2$, or $\mathrm{V}^\perp = \mathrm{V}_1^\perp \cap \mathrm{V}_2^\perp$. We would like to find out the probability that $\dim(\mathrm{V}) = k$ or $\dim\left(\mathrm{V}^\perp\right) = m - k$. Assume that $\dim(\mathrm{V}_1) = i$ and $\dim(\mathrm{V}_2) = j$. This happens with probability $\mathrm{P}_v^{(l)}(i)$ and $\mathrm{P}_v^{(l)}(j)$ respectively. There are $\begin{bmatrix} m-i \\ m-k \end{bmatrix}$ distinct subspaces of dimension $m-k$ in $\mathrm{V}_1^\perp$. We need to count how many subspaces of dimension $m-j$ there are such that $\dim\left(\mathrm{V}_1^\perp \cap \mathrm{V}_2^\perp\right) = m-k$. Let $B_1^\perp = \{b_1^\perp, \dots, b_{m-k}^\perp, b_{m-k+1}^\perp, \dots, b_{m-i}^\perp\}$ be the basis for V_1. We would like to construct a basis set for V_2 such that the first $m-k$ vectors are $\{b_1^\perp, \dots, b_{m-k}^\perp\}$. The $(m-k+1)^{\text{th}}$ vector can be chosen in $\left(2^m - 2^{m-i}\right)$ ways. We subtract 2^{m-i} as we can not choose any vector from $\mathrm{V}_1^\perp$. Similarly $(m-k+l+1)^{\text{th}}$ vector can be chosen in $\left(2^m - 2^{m-i+l}\right)$ ways. This gives the total number of possible choices as $\prod_{l=0}^{k-j-1}\left(2^m - 2^{m-i+l}\right)$. But not all the choices give different subspaces. There are $\prod_{l=0}^{k-j-1}\left(2^{m-j} - 2^{m-k+l}\right)$ choices which gives the same subspace. Hence there are $\prod_{l=0}^{k-j-1}\frac{\left(2^m - 2^{m-i+l}\right)}{\left(2^{m-j} - 2^{m-k+l}\right)} = \begin{bmatrix} i \\ k-j \end{bmatrix} 2^{(k-i)(k-j)}$ choices for $\mathrm{V}_2^\perp$ for a fixed $\mathrm{V}_1^\perp$. The factor $\begin{bmatrix} m \\ m-j \end{bmatrix}$ in the denominator is the total number of subspaces of dimension $m-j$. This explains (2.14).

Now the equation for a check node of degree $\mathtt{r}$ can be derived recursively. Assume that we know $\mathrm{P}_c^{(l)}(i, \mathtt{r}-1)$. Then by the same argument as for the check node of degree 3, we obtain (2.15).

The derivation of the equation for the variable node side is very similar. Consider a variable node of degree 2. Assume that the initial message V_1 is a subspace of dimension i. This happens with probability $\binom{m}{i}\epsilon^i (1-\epsilon)^{m-i}$. Now fix this subspace. Let us assume that the message coming from an edge is of dimension j whose subspace is V_2. This happens with probability $\mathrm{P}_c^{(l)}(j)$. Now we would like to find out how many subspaces of dimension j give an intersection of dimension k with a subspace of dimension i. This can be computed in exactly the same way as for the check node of degree 3 by replacing $\mathrm{V}^\perp, \mathrm{V}_1^\perp, \mathrm{V}_2^\perp$ by $\mathrm{V}, \mathrm{V}_1, \mathrm{V}_2$. Thus we obtain (2.16) by replacing i, j, k in the right hand side of (2.14) by $m-i, m-j, m-k$, respectively. The equation (2.17) is derived similarly. □

In Table 2.4.1, we list the thresholds for various ensembles. Note that for the ensemble with d.d. pair $\lambda(y) = y$ and $\rho(y) = y^2$, initially the threshold increases rapidly (as m is increased). Unfortunately it reaches a peak at $m = 6$ and then starts decreasing. For the ensemble with d.d. pair $\lambda(y) = 0.5y + 0.5y^4$ and $\rho(y) = y^5$, the threshold increases by moving from $m = 1$ to $m = 2$, but after that it starts decreasing. For $\lambda(y) = y^2$ and $\rho(y) = y^3$ the thresholds already start decreasing by moving from the binary case to an alphabet of size 4. We have observed for various other ensembles that if there are no degree

2 variable nodes then the threshold already starts decreasing by moving from the binary case to an alphabet of size 4.

$\lambda(y) = y, \rho(y) = y^2$
$\epsilon^{\text{sh}} \approx 0.6667$

m	ϵ^{IT}	ϵ^{stab}
1	0.5	0.5
2	0.5775	0.5811
3	0.6183	0.651
4	0.6369	0.7075
5	0.6446	0.7518
6	0.6464	0.7864
7	0.6453	0.8135
8	0.6425	0.8349
15	0.616	0.9097

$\lambda(y) = 0.5y + 0.5y^4, \rho(y) = y^5$
$\epsilon^{\text{sh}} \approx 0.4762$

m	ϵ^{IT}	ϵ^{stab}
1	0.4	0.4
2	0.4487	0.4832
3	0.4353	0.5605
4	0.4194	0.6266

$\lambda(y) = y^2,\ \rho(y) = y^3$
$\epsilon^{\text{sh}} = 0.75$

m	ϵ^{IT}	ϵ^{stab}
1	0.6474	1.0
2	0.6348	1.0
3	0.6192	1.0

Table 2.1: Thresholds for the ensemble EGL (λ, ρ, m) for various degree distributions. ϵ^{stab} is computed from Lemma 5 by equating the left hand side of Eqn(2.18) to unity.

We can obtain analytical upper bounds on the BP threshold by analyzing the stability condition of the density evolution map. The stability of density evolution map implies that when we give a small perturbation around the fixed point corresponding to the error-free decoding, the recursion converges back to the error-free fixed point. In the next lemma we give the stability condition.

Lemma 5. *Consider the ensemble EGL (λ, ρ, m). The fixed point of the density evolution equations corresponding to error-free decoding is stable if*

$$\lambda_2 \rho'(1) \max_{k=1,\ldots,m} \left(\sum_{i=k}^{m} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \frac{\begin{bmatrix} i \\ k \end{bmatrix}}{\begin{bmatrix} m \\ k \end{bmatrix}} \right) < 1, \tag{2.18}$$

where the density evolution equations are given in Lemma 4.

Proof. The probabilities $\mathrm{P}_v(i)(\mathrm{P}_c(i))$, $i \in \{0, \ldots, m\}$ sum to one. So we can express the probability $\mathrm{P}_v(0)(\mathrm{P}_c(0))$ in terms of $\mathrm{P}_v(i)(\mathrm{P}_c(i)), i \in \{1, \ldots, m\}$. Thus we can write the density evolution recursion as an m dimensional recursion. For correct decoding the probabilities $\mathrm{P}_v(i)$, $i \in \{1, \ldots, m\}$ converge to zero. Consider a small perturbation of the all zero vector. Let us denote the perturbed vector by $\underline{\delta} = (\delta_1, \ldots, \delta_m)$. By Eqns(2.14, 2.15), for the input $\underline{\delta}$, the output of the check node side up to the first order term is the vector

$$\rho'(1)\,(\delta_1, \dots, \delta_m)\,.$$

For the variable node side, observe that only the degree two variable nodes contribute to the first order term. So we change the order of summation in Eqn(2.16) and rewrite it as

$$\mathrm{P}_v^{(l+1)}(k, 2) = \sum_{j=k}^{m} \left(\sum_{i=k}^{m-j+k} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \frac{\begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} 2^{(i-k)(j-k)}}{\begin{bmatrix} m \\ j \end{bmatrix}} \right) \mathrm{P}_c^{(l)}(j).$$

This shows that for the input vector $\rho'(1)\,(\delta_1, \dots, \delta_m)$, the k^{th} component of the output of the variable node side is

$$\lambda_2 \rho'(1) \sum_{j=k}^{m} \left(\sum_{i=k}^{m-j+k} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \frac{\begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} 2^{(i-k)(j-k)}}{\begin{bmatrix} m \\ j \end{bmatrix}} \right) \delta_j.$$

Thus the linearized output of the density evolution map is given by

$$M\underline{\delta},$$

where the matrix M is an $m \times m$ matrix with entries

$$M_{kj} = \begin{cases} \lambda_2 \rho'(1) \sum_{i=k}^{m-j+k} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \frac{\begin{bmatrix} i \\ k \end{bmatrix} \begin{bmatrix} m-i \\ j-k \end{bmatrix} 2^{(i-k)(j-k)}}{\begin{bmatrix} m \\ j \end{bmatrix}}, & j \geq k, \\ 0, & j < k. \end{cases} \tag{2.19}$$

The stability of the density evolution map around the all-zero vectors is governed by the eigenvalues of the matrix M. More specifically, the density evolution map is stable if the largest absolute eigenvalue of the matrix M is less than one. As M is upper triangular, its eigenvalues are equal to its diagonal entries. The entries of the matrix M are positive. Thus the stability condition is given by

$$\lambda_2 \rho'(1) \max_{k=1,\dots,m} \left(\sum_{i=k}^{m} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} \frac{\begin{bmatrix} i \\ k \end{bmatrix}}{\begin{bmatrix} m \\ k \end{bmatrix}} \right) < 1,$$

as the left hand side is the largest eigenvalue. □

Note: The stability condition given in Eqn(2.18) for $m = 1$ is identical to the stability condition for binary LDPC codes derived in [12].

2.4.2 Density Evolution for the Ensemble EGF (λ, ρ, m)

The ensemble EGF (λ, ρ, m) is a subset of the ensemble EGL (λ, ρ, m). We prove this and we characterize the set of messages which arise during the BP decoding of the ensemble EGF (λ, ρ, m) in the following lemma.

Lemma 6 (EGF (λ, ρ, m) as a subset of EGL (λ, ρ, m))**.** *The mapping $f(\alpha) = \omega\alpha$, where $\omega \in \mathrm{GF}^*(2^m)$ and $\alpha \in \mathrm{GF}(2^m)$ is equivalent to a mapping $g(b) = Wb$, where $b \in \mathrm{GF}_2^m$ and $W \in \mathrm{GL}_2^m$. Hence all the messages in the belief propagation decoder are equivalent to the subspaces of the vector space GF_2^m. In fact all the possible subspaces of the vector space GF_2^m do arise in the belief propagation decoder.*

Proof. First we note that the additive group of GF (2^m) and the vector space GF_2^m are isomorphic. Also the mapping $f(\alpha) = \omega\alpha$ is a linear mapping. Hence from the isomorphism of GF (2^m) and GF_2^m, we can think of the mapping f as $f : \mathrm{GF}_2^m \mapsto \mathrm{GF}_2^m$. Every linear one-to-one mapping from a vector space to the same vector space can be represented by an invertible matrix [23]. Hence every mapping in EGF (λ, ρ, m) is equivalent to a mapping in EGL (λ, ρ, m). Thus from lemma 2, we know that every message is equivalent to a subspace of GF_2^m.

In order to show that all the possible subspaces arise in the message passing decoder, we consider a variable node where the first bit is erased. The initial message is $\Psi = \left\{\frac{1}{2}, \frac{1}{2}, 0, \ldots, 0\right\}$. After the edge action, this message can be mapped to all the other possible messages (subspace) of dimension 1. The operation at the check node is to take the sum of subspaces. Now all the possible subspace can be generated by taking the sum of subspaces of dimension 1. This shows that all the possible subspaces of vector space GF_2^m arise in the belief propagation decoder. □

For the derivation of density evolution equations, we observe that for $m \leq 3$, after the edge action a message of same dimension gets mapped to any other message of same dimension by equal number of mappings. Hence we can again combine the probability of the messages of the same dimension and do the density evolution over the dimension of the messages. Thus the density evolution equations for the ensemble EGF(λ, ρ, m) is the same as that of EGL(λ, ρ, m) for $m \leq 3$ as given in Lemma 4. However for $m > 3$, it is no longer true that any message of the same dimension gets mapped to any other message of same dimension. The set of messages of given a dimension are partitioned into several orbits under the action of the label group. In order to accomplish density evolution we need to keep track of each orbit. This quickly becomes quite cumbersome. Also note that the performance of the ensemble *does* in general depend on the primitive element which one chooses to represent the field. I.e., two isomorphic fields do not in general yield identical equations. More precisely, if we run density evolution for a fixed number of iterations then the performance of two isomorphic fields is in general strictly different. An example is GF (32) with fields defined with respect to the irreducible polynomials

$1+z^3+z^5$ and $1+z+z^2+z^3+z^5$. The difference though between these two fields is very small and the resulting difference in the threshold is of the order of 10^{-4}. Note also that within this precision, the ensembles EGF(λ, ρ, m) and EGL(λ, ρ, m) seem to have approximately the same threshold.

2.5 Stability Condition

For the general case, an analysis in terms of density evolution is in principle possible but practically difficult. Even for codes over GF(4) densities already "live" in $\mathbb{R}^3$. The BP threshold can be computed numerically by Monte Carlo methods in the same way as this is done in the setting of turbo codes, [25].

Slightly less ambitious, one can investigate the behavior of density evolution close to the desired fixed-point and derive a stability condition. Rather than giving a full-fledged analysis let us derive here the stability condition via an asymptotic analysis of the number of small weight codewords. Recall that a binary codeword is called *minimal* it there is no other codeword whose support set is a proper subset of the support set of this codeword. The support set of a codeword is the set of its non-zero positions. For a non-binary codeword we ask in addition that the left-most position is equal to one. The stability condition we conjecture is inspired by the following phenomenon which happens for binary LDPC codes as described in [17, Sec. 6.6]. Let $P(x)$ be the generating function counting the number of expected minimal constant weight codewords in the limit of infinite blocklength. As shown in [17], the generating function $P(x)$ is given by

$$P(x) = -\frac{1}{2}\log\left(1-\lambda'(0)\rho'(1)x\right).$$

Then the convergence condition for $P(x)$ when evaluated at $x = \mathfrak{B}(\mathsf{a})$ gives the stability condition

$$\mathfrak{B}(\mathsf{a})\lambda'(0)\rho'(1) < 1,$$

where $\mathfrak{B}(\mathsf{a})$ is the Battacharya parameter for channel log-likelihood density a,

$$\mathfrak{B}(\mathsf{a}) = \int_{-\infty}^{\infty} \mathsf{a}(y)e^{-\frac{y}{2}}dy.$$

We will use this analogy to derive the stability condition for NBLDPC ensemble EGF (λ, ρ, m). In the next lemma we derive upper and lower bounds on the generating function of the expected number of binary minimal codewords for non-binary ensembles.

Lemma 7. *Consider the ensemble EGF(λ, ρ, m). Let $P(x) = \sum_{w\geq 0} p_w x^w$ denote the generating function counting the number of expected minimal binary (low-weight) codewords in the limit of large blocklengths. More precisely, if $N(\mathtt{G}, w)$ denotes the number of minimal binary codewords of weight w in the graph $\mathtt{G}$, then $p_w \triangleq \lim_{n\to\infty} \mathbb{E}_{\mathtt{G}\in EGF(\lambda,\rho,m)}[N(\mathtt{G}, w)]$. Define*

$$\tilde{P}(x) \triangleq -\frac{1}{2(2^m-1)}\log\left(1-\lambda'(0)\rho'(1)\frac{(1+x)^m-1}{2^m-1}\right). \tag{2.20}$$

Then

$$\tilde{P}(x) \leq P(x) \leq (2^m - 1)\tilde{P}(x), \tag{2.21}$$

where the inequality is understood to hold pointwise for the coefficients of the respective Taylor series.

Proof. Let d denote the weight of the codeword over GF (2^m). This means that for such a codeword there are d non-zero positions of the variable nodes in the graph. Look at the subgraph induced by these d variable nodes. We claim that it has to form a stopping-set (in the sense of binary codes), i.e., no check node contained in this residual graph can have degree one (this is true since a single edge entering a check node and carrying a non-zero message can not satisfy this check node). From the analysis of binary codes we know that in the limit of large block lengths the expected number of minimal stopping sets of weight d tends to $(\lambda'(0)\rho'(1))^d/(2d)$ and that these stopping sets all correspond to cycles (in the bipartite graph) of length $2d$. Consider now a cycle of length $2d$. Such a cycle involves d variable nodes, d check nodes and $2d$ edges. Assign to each of the d variable nodes a non-zero binary m-tuple. What is the probability that this assignment corresponds to a codeword? We claim that this probability is equal to $1/(2^m - 1)^d$. To see this claim, first reveal the edge labels along the cycle for all *even* edges. This means: start traversing the cycle in a chosen direction, and reveal the edge label along every second edge. Consider now the first check node. If α and β are the labels along the two connected edges and if x and y are the corresponding bit assignments at the connected variables, $x, y, \alpha, \beta \in$ GF (2^m), then the check is fulfilled iff $\alpha x = \beta y$. Since x and y are already fixed, and also exactly one of the labels has been already fixed, this means that there is exactly 1 out of $2^m - 1$ choices which leads to a fulfilled check node. The same argument applies to all d check nodes, which confirms the claim. The above argument is valid for all non-zero bit assignments at the variable node. The generating function which counts the binary weight of all non-zero bit assignments at a variable node is $(1+x)^m - 1$. It follows therefore that a cycle of weight d gives in expectation rise to $\text{coef}\{\big(\frac{(1+x)^m-1}{2^m-1}\big)^d, x^w\}$ binary codewords of weight w.

First consider the codewords over GF (2^m). Due to the linearity of the code, each cycle which gives rise to at least one codeword, gives rise to exactly $2^m - 1$ codewords (multiplication by all non-zero field elements). Exactly one of those has its left-most element equal to 1, and is therefore minimal. We are interested in binary minimal codewords though. Although it is possible to write down the exact such number we will not need it for our purpose. It is sufficient to note that there are at least as many *binary* minimal codewords as there are minimal codewords over GF (2^m) and that there are at most $2^m - 1$ more. This gives rise to the two stated bounds. □

Note that because of Eqn(2.21), the convergence behavior of the generating function for minimal weight binary codewords $P(x)$ and $\tilde{P}(x)$ defined in Eqn(2.20) is identical. By the analogy with binary LDPC codes, the convergence condition of the generating function $\tilde{P}(x)$ when evaluated at $\mathfrak{B}(\mathsf{a})$

gives the stability condition for NBLDPC codes. We now present this as a conjecture.

Conjecture 1 (Stability Condition for EGF (λ, ρ, m)). *Consider the ensemble EGF(λ, ρ, m). Assume that transmission takes place over a BMS channel with log-likelihood density* a *and associated Battacharya constant* $\mathfrak{B}(\mathsf{a})$. *If*

$$\lambda'(0)\rho'(1)\frac{(1+\mathfrak{B}(\mathsf{a}))^m - 1}{2^m - 1} > 1, \tag{2.22}$$

then the fixed-point corresponding to zero-error probability is not stable.

Note that Eqn(2.22) agrees with Eqn(2.18) for the case $m = 2, 3$ for transmission over the BEC. This supports the conjecture as for $m = 2, 3$, the density evolution recursion is identical for the ensemble EGL (λ, ρ, m) and EGF (λ, ρ, m).

2.6 Conclusion

We have investigate the performance of NBLDPC ensembles. In particular, assuming that transmission takes place over the BEC(ϵ), we have given a compact representation of the density evolution equations for the ensemble EGL (λ, ρ, m) and we have discussed the stability condition.

Let us state here what we consider to be some of the most interesting questions that remain unanswered. From the examples we have investigated, it seems that for a fix degree distribution the threshold is a unimodal function of the alphabet size. If there are sufficiently many degree-two variable nodes the threshold initially rises and eventually decays again as m is increased. Otherwise it decrease right away.

There is one degree of freedom which was already suggested in [4] and which we have not considered so far. In all our analysis we assumed a uniform distribution on the edge labels. In out setting it is natural to allow a non-uniform distribution on the edge labels in such a way that the distribution respects the underlying algebraic structure. E.g., the ensemble EGF (λ, ρ, m) can be considered a special case of the ensemble EGL (λ, ρ, m) where we put a uniform distribution on the labels corresponding to field elements and zero weight on all other labels. Obviously there are many degrees of freedom that could be explored.

It is hoped that by a proper exploitation of these degrees of freedom one can find yet another way of approaching capacity, adding to our understanding of capacity approaching iterative coding schemes.

MAP Performance of Non-Binary LDPC Codes over the BEC

3

The chapter is organized in the following way. An introduction is given in Section 3.1. In Section 3.2 we give the definitions and notations. We discuss the peeling decoder, stopping constellations and the equivalence of the peeling and the BP decoder for the NBLDPC codes in Section 3.3. In Section 3.4 we compute the residual degree distribution of the NBLDPC ensemble and derive the expression for the average of total number of codewords in a residual ensemble. A sufficient condition is derived in Section 3.5, which, when satisfied, guarantees that the asymptotic rate of the residual ensemble is equal to its design rate. We then show the equality between the rate of the residual ensemble and the conditional entropy of the transmitted codewords. Finally we conclude in Section 3.6 with a discussion.

3.1 Introduction

Our aim is to investigate the MAP decoding performance of NBLDPC codes when transmission takes place over the BEC. So, we will assume in the rest of this chapter that transmission takes place over the BEC. Towards examining the MAP decoding performance of NBLDPC codes, our approach is to generalize the arguments of [1] from the binary to the non-binary setting.

Consider the binary case. The basic idea of [1], relating the MAP performance to the BP performance is the following. For the binary case there are two decoders, the BP algorithm and the peeling decoder. The peeling decoder was introduced in [7]. Although these two decoders look different, they are in fact identical if their final estimates on bits are considered. The peeling decoder associates to every code and erasure set a residual graph. If the erasure probability is below the BP threshold then the residual graph is empty as all the bits are known almost surely. It was shown in [7] that if the erasure probability is above the BP threshold then for most instances of the erasure set and graph, the residual graph has a degree distribution close to the *average residual degree distribution*. It was also shown that conditioned on the residual degree distribution, the induced probability distribution is uniform over all the graphs with the given degree distribution.

The entropy of the transmitted codeword conditioned on the channel observation is given by the logarithm of the number of codewords which are compatible with the observation. Thus the idea of bounding the conditional entropy for an erasure probability above the BP threshold is the following (for erasure probability below the BP threshold, the conditional entropy is zero). As we discussed the peeling decoder almost surely results in an LDPC ensemble whose degree distribution is given by the average residual degree distribution. Thus the normalized logarithm of the number of codewords which are compatible with the typical channel observations is lower bounded by the design rate of the average residual ensemble. The design rate only gives a lower bound since some check equations might be dependent. A criterion was derived in [1] which, when satisfied, guarantees that the lower bound is tight, i.e., the actual rate of the ensemble is equal to its design rate.

We will essentially follow the same sequence of arguments for the non-binary case. The BP decoder is defined in Chapter 2. We define the peeling decoder and stopping sets for the non-binary case. We refer to stopping sets as *stopping constellations* to distinguish them from the binary setting. Then we show the equivalence of the peeling and the BP decoder by showing that both of them get stuck in the largest stopping constellation. We show how the average residual degree distribution can be computed by using the fixed points of the density evolution of the BP decoder. Then we show that conditioned on the degree distribution of the residual graphs, all the graphs which are compatible with this degree distribution have uniform probability. We generalize the criterion of [1] to the non-binary case which, when satisfied, shows that almost all the codes in the ensemble have their rate equal to the design rate. If this criterion is satisfied by the average residual degree distribution, then we show that the conditional entropy of the transmitted codeword is equal to the design rate of the average residual ensemble. Here we assume that the degree distribution of a random residual graph is concentrated around the average residual degree distribution. We also observe that the Maxwell construction, relating the performance of the MAP and the BP decoder, also holds in the setting of non-binary LDPC codes.

3.2 Preliminaries

We consider the NBLDPC ensemble $\mathrm{EGL}\,(n, \lambda, \rho, m)$ ($\mathrm{EGL}\,(\lambda, \rho, m)$ in the asymptotic limit) defined in Chapter 2. We recall from Chapter 2, Lemma 2 that for transmission over the BEC, the messages which arise during BP decoding have a very specific form. The messages are real vectors of length 2^m and the α^{th} component of the message gives the posteriori probability that the corresponding symbol is α, $\alpha \in \mathrm{GF}_2^m$. The following properties of messages were proved in Chapter 2, Lemma 2:

1. The non-zero entries of a message are all equal.

2. The indices corresponding to non-zero entries of a message form a subspace of GF_2^m. Thus each message is equivalent to a subspace. This means that if the entries corresponding to α and β, $\alpha, \beta \in \mathrm{GF}_2^m$, are non-zero, then so is the entry corresponding to $\alpha + \beta$.

3. The variable-node side operation is equivalent to taking the intersection of the subspaces corresponding to the incoming messages.

4. The check-node side operation is equivalent to taking the sum of the subspaces corresponding to the incoming messages.

Based on these properties, we say that the dimension of a message Ψ, call it $\dim(\Psi)$, is k if the number of non-zero entries of Ψ is 2^k. By a slight abuse of notation, we denote the subspace of indices corresponding to non-zero entries of a message Ψ also by Ψ. Note that the subspaces corresponding to initial messages have a specific form. An initial message of dimension k has the set of basis vectors $\{e_{i_1}, \ldots, e_{i_k}\}$, where e_i is a vector of length m with i^{th} component equal to 1 and the remaining components equal to zero. This corresponds to the message where the bits $i_1, \ldots, i_k$ are erased and the rest of them are known. We denote the set of initial messages by $\mathcal{M}_{\text{ini}}$ and $\mathcal{M}$ denotes the set of all messages of the BP decoder. Also, we denote the set of neighbours of a node x by $\mathcal{N}(x)$.

We denote the number of mappings belonging to GL_2^m which map a given subspace of dimension k to another subspace of dimension k by $g(m, k)$. It is given by:

$$g(m,k) = |\,\mathrm{GL}_2^m\,| \begin{bmatrix} m \\ k \end{bmatrix}^{-1}, \tag{3.1}$$

where $\begin{bmatrix} m \\ k \end{bmatrix}$ is the Gaussian binomial coefficient (defined in Chapter 2, Eqn(2.2)) denoting the number of different subspaces of dimension k of GF_2^m. We obtain Eqn(3.1) by noticing that the number of elements in GL_2^m which map a subspace V to V_1 is equal to the number of elements in GL_2^m which map V to V_2, where the dimensions of V, V_1 and V_2 are equal to k.

We will use the following two simple facts $f\,(A \cap B) = (fA) \cap (fB)$, $f\,(A + B) = (fA) + (fB)$, where $f \in \mathrm{GL}_2^m$ and A, B are subspaces of GF_2^m.

In the next section we define the peeling decoder and stopping constellations. Then we prove that the BP and the peeling decoder get stuck in the largest stopping constellation compatible with the channel output.

3.3 Peeling Decoder

Assign to each variable node $\mathtt{v}$ a subspace belonging to the set of initial messages. We call this subspace the *state* of the variable node; denote it by $E_{\mathtt{v}}$, where $E_{\mathtt{v}} \in \mathcal{M}_{\text{ini}}$. We say that a check node $\mathtt{c}$ is *active* if there exists $\mathtt{v} \in \mathcal{N}(\mathtt{c})$ such that $E_{\mathtt{v}} \cap f_{\mathtt{vc}}^{-1}\left(\sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} E_i\right)$ is a strict subset of $E_{\mathtt{v}}$, where $f_{i\mathtt{c}}$ is the mapping on the edge connecting variable node $i \in \mathcal{N}(\mathtt{c})$ to check node $\mathtt{c}$. We call $\mathtt{v}$ and $\mathtt{c}$ an active pair.

The *channel state assignment* $C = \{C_{\mathtt{v}}\}_{\mathtt{v} \in \mathcal{V}}$ is the state assignment corresponding to channel erasures. Thus $C_{\mathtt{v}}$ is the subspace corresponding to the channel erasures for the variable node $\mathtt{v}$. We now define the peeling decoder.

1. Initially all the variable nodes are assigned the channel state C.

2. Consider an active pair: let $\mathtt{v}$ be the variable node and $\mathtt{c}$ be the check node. Set $E_{\mathtt{v}} = E_{\mathtt{v}} \cap f_{\mathtt{vc}}^{-1}\left(\sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} E_i\right)$.

3. If there is no active pair, terminate. Otherwise repeat step (2).

Note that in the binary setting, the possible states of variable nodes are 0 and 1. Hence an active pair of nodes correspond to a check node which has the corresponding variable node as the only variable node with state 1 (erased). The remaining attached variable nodes are assigned the state 0 (known). This corresponds to a check node of degree 1 in the setting of the peeling decoder for the binary codes [7].

We now define the stopping constellation for NBLDPC codes.

Definition 1. *A stopping constellation is an assignment of states $E = \{E_{\mathtt{v}}\}_{\mathtt{v} \in \mathcal{V}}$, where $E_{\mathtt{v}} \in \mathcal{M}_{\text{ini}}$ and $\mathcal{V}$ is the set of variable nodes. The state assignment E is such that there are no active pairs of nodes, i.e., for every variable node $\mathtt{v}$ and each check node $\mathtt{c}$ connected to $\mathtt{v}$, $E_{\mathtt{v}} \subseteq f_{\mathtt{vc}}^{-1}\left(\sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} E_i\right)$.*

We say that a check node $\mathtt{c}$ satisfies the *decoding failure criterion* with respect to the state assignment E if

$$E_{\mathtt{v}} \subseteq f_{\mathtt{vc}}^{-1}\left(\sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} E_i\right), \forall \mathtt{v} \in \mathcal{N}(\mathtt{c}). \tag{3.2}$$

Next we define the union G of two stopping constellations $E = \{E_{\mathtt{v}}\}_{\mathtt{v} \in \mathcal{V}}$ and $F = \{F_{\mathtt{v}}\}_{\mathtt{v} \in \mathcal{V}}$ as $G_{\mathtt{v}} = E_{\mathtt{v}} + F_{\mathtt{v}}$, $\mathtt{v} \in \mathcal{V}$, where $E_{\mathtt{v}} + F_{\mathtt{v}}$ is the sum of subspaces $E_{\mathtt{v}}, F_{\mathtt{v}}$. We claim that G is also a stopping constellation. We want to prove that $G_{\mathtt{v}} \subseteq f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} f_{i\mathtt{c}}G_i\right)$, $\forall\mathtt{c}\in\mathcal{N}(\mathtt{v})$, $\forall\mathtt{v}\in\mathcal{V}$. Now,

$$\begin{aligned}
f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} f_{i\mathtt{c}}G_i\right) &= f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} f_{i\mathtt{c}}\left(E_i+F_i\right)\right),\\
&= f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} \left(f_{i\mathtt{c}}E_i+f_{i\mathtt{c}}F_i\right)\right),\\
&= f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} f_{i\mathtt{c}}E_i\right)\\
&\quad +f_{\mathtt{vc}}^{-1}\left(\sum_{i\in\mathcal{N}(\mathtt{c})\backslash\mathtt{v}} f_{i\mathtt{c}}F_i\right),\\
&\supseteq E_{\mathtt{v}}+F_{\mathtt{v}}=G_{\mathtt{v}}.
\end{aligned}$$

We say that a stopping constellation E is a subset of state assignment S if

$$E_{\mathtt{v}}\subseteq S_{\mathtt{v}}, S_{\mathtt{v}}\in\mathcal{M}_{\mathrm{ini}}, \forall\mathtt{v}\in\mathcal{V}.$$

As the union of two stopping constellations is a stopping constellation, there is a unique largest stopping constellation which is a subset of a state assignment. In the following lemma we prove that the peeling decoder gets stuck in the largest stopping constellation which is a subset of the channel state assignment.

Lemma 8. *Consider transmission over the BEC using a NBLDPC code which is decoded by the peeling decoder. After the termination of the peeling decoder, the final state assignment to the variable nodes is the largest stopping constellation E which is a subset of the channel state assignment C.*

Proof. Let F be the final state assignment. First note that F is a stopping constellation as this is the only condition for the peeling decoder to terminate. Now $F_{\mathtt{v}}\subseteq C_{\mathtt{v}}$, $\forall\mathtt{v}\in\mathcal{V}$, which follows from the fact that every update of the state of $\mathtt{v}$ by the peeling decoder satisfies this property. We prove by contradiction that F is the largest stopping constellation which is a subset of C. If E is the largest stopping constellation, then F must be a subset of E. Otherwise, by taking the union of E and F we could obtain a stopping constellation larger than E which is also a subset of C. Hence F is a strict subset of E. Now, consider the variable node $\mathtt{v}$ which is the first node whose state becomes smaller than $E_{\mathtt{v}}$ when it is acted on by the check node $\mathtt{c}$. This is only possible if there are nodes in $\mathcal{N}(\mathtt{c})$ whose state is a subset of their respective states corresponding to E. But that is not possible as $\mathtt{v}$ is the first node for which this happens. Hence we prove the claim that $F=E$. □

We now prove the following lemma which will be useful when proving that the BP decoder gets stuck in the largest stopping constellation which is a subset of the channel state assignment.

Lemma 9. *Let $E = \{E_v\}$, $v \in \mathcal{V}$, be the largest stopping constellation contained in the channel state assignment C. Then during BP decoding, the state E_v is a subset of all the outgoing and incoming messages to the variable node $\mathtt{v}$. Also the state assignment E is a subset of the BP state assignment B, where $B = \{B_v\}_{v \in \mathcal{V}}$. B_v is the final estimate of the node $\mathtt{v}$ from the BP decoder.*

Proof. Recall the notation from Chapter 2, Section 2.3 that the superscripts of the message $\Psi^{(l,i)}$ indicate the iteration number l and the step number i, $i \in \{1, \ldots, 4\}$ in the l^{th} iteration.

The initial message from a variable node $\mathtt{v}$ is $C_{\mathtt{v}}$ and $E_{\mathtt{v}} \subseteq C_{\mathtt{v}}$ by definition of E. Also, the incoming message from a check node $\mathtt{c}$ to a variable node $\mathtt{v}$ in the first iteration is

$$\Psi^{(1,4)}_{\mathtt{c},\mathtt{v}} = f^{-1}_{\mathtt{vc}} \left(\sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} C_i \right).$$

This implies that $\Psi^{(1,4)}_{\mathtt{c},\mathtt{v}}$ contains $E_{\mathtt{v}}$ as $E_i \subseteq C_i$, $i \in \mathcal{N}(\mathtt{c})$ and E forms a stopping constellation. Now, by induction we can prove the desired result. Assume that $\Psi^{(l-1,1)}_{\mathtt{v},\mathtt{c}}$ and $\Psi^{(l-1,4)}_{\mathtt{c},\mathtt{v}}$ contain the state $E_{\mathtt{v}}$, $\forall \mathtt{v} \in \mathcal{V}$, $\forall \mathtt{c} \in \mathcal{N}(\mathtt{v})$. This implies that $\Psi^{(l,1)}_{\mathtt{v},\mathtt{c}}$ and $\Psi^{(l,4)}_{\mathtt{c},\mathtt{v}}$ contain $E_{\mathtt{v}}$ as E is a stopping constellation,

$$\Psi^{(l,1)}_{\mathtt{v},\mathtt{c}} = C_{\mathtt{v}} \cap \cap_{j \in \mathcal{N}(\mathtt{v}) \setminus \mathtt{c}} \Psi^{(l-1,4)}_{j,\mathtt{v}},$$

and

$$\Psi^{(l,4)}_{\mathtt{c},\mathtt{v}} = f^{-1}_{\mathtt{vc}} \sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} \Psi^{(l,1)}_{i,\mathtt{c}} \supseteq f^{-1}_{\mathtt{vc}} \sum_{i \in \mathcal{N}(\mathtt{c}) \setminus \mathtt{v}} f_{i\mathtt{c}} E_i \supseteq E_{\mathtt{v}}.$$

The second claim of the lemma can be proved by noting that

$$B_{\mathtt{v}} = C_{\mathtt{v}} \cap \left(\cap_{i \in \mathcal{N}(\mathtt{v})} \Psi^{(4)}_{i,\mathtt{v}} \right)$$

and as $E_{\mathtt{v}} \subseteq C_{\mathtt{v}}$, $E_{\mathtt{v}} \subseteq \Psi^{(4)}_{i,\mathtt{v}}$, $\forall i \in \mathcal{N}(\mathtt{v})$. So, we prove that $E_{\mathtt{v}} \subseteq B_{\mathtt{v}}$. Here we have dropped the superscript corresponding to iteration number to denote the converged messages. □

Thus from the previous lemma we see that the BP estimate of a variable node $\mathtt{v}$ satisfies $B_{\mathtt{v}} \supseteq E_{\mathtt{v}}$. If we prove that the BP estimates of variable nodes also form a stopping constellation then $B_{\mathtt{v}} = E_{\mathtt{v}}$, as E is the largest stopping constellation contained in the channel state assignment C, B contains E, and C contains B. In the following lemma we prove that the final estimate B is a stopping constellation.

Lemma 10. *Consider transmission over the BEC using NBLDPC code which is decoded by the BP decoder. Let $\Psi^{(i)}$ denote the message in the i^{th} step of an iteration of BP decoding when the decoder has converged. Let $\Psi^{(4)}_{\mathtt{c},v}$ be the incoming message to the variable node $\mathtt{v}$ from check node $\mathtt{c}$ and $\Psi^{(1)}_{v,\mathtt{c}}$ be* *the message from $\mathtt{v}$ to $\mathtt{c}$. Let C be the channel state assignment. Then the estimates $B_{\mathtt{v}} = C_{\mathtt{v}} \cap_{i \in \mathcal{N}(\mathtt{v})} \Psi^{(4)}_{i,\mathtt{v}}, \forall \mathtt{v} \in \mathcal{V}$ form a stopping constellation. Hence by Lemma 9, the estimate B is the largest stopping constellation contained in C.*

Proof. We need to prove that $B_{\mathtt{v}} \subseteq f_{\mathtt{vc}}^{-1} \sum_{i \in \mathcal{N}(\mathtt{c}) \backslash \mathtt{v}} f_{i\mathtt{c}} B_i$. For the sake of simplicity assume that the check node $\mathtt{c}$ is of degree 3 and let $\mathcal{N}(\mathtt{c}) = \{\mathtt{v}, \mathtt{v}_1, \mathtt{v}_2\}$. Also note that $B_{\mathtt{v}} = \Psi^{(4)}_{\mathtt{c},\mathtt{v}} \cap \Psi^{(1)}_{\mathtt{v},\mathtt{c}}$ as $\Psi^{(1)}_{\mathtt{v},\mathtt{c}} = C_{\mathtt{v}} \cap \left(\cap_{i \in \mathcal{N}(\mathtt{v}) \backslash \mathtt{c}} \Psi^{(4)}_{i,\mathtt{v}} \right)$. So we need to prove that

$$f_{\mathtt{vc}}^{-1} \left(f_{\mathtt{v}_1\mathtt{c}} \left(\Psi^{(4)}_{\mathtt{c},\mathtt{v}_1} \cap \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} \right) + f_{\mathtt{v}_2\mathtt{c}} \left(\Psi^{(4)}_{\mathtt{c},\mathtt{v}_2} \cap \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \right) \right) \supseteq \Psi^{(4)}_{\mathtt{c},\mathtt{v}} \cap \Psi^{(1)}_{\mathtt{v},\mathtt{c}}.$$

As the decoder has converged,

$$\begin{aligned} \Psi^{(4)}_{\mathtt{c},\mathtt{v}} &= f_{\mathtt{vc}}^{-1} \left(f_{\mathtt{v}_1\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} + f_{\mathtt{v}_2\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \right), \\ \Psi^{(4)}_{\mathtt{c},\mathtt{v}_1} &= f_{\mathtt{v}_1\mathtt{c}}^{-1} \left(f_{\mathtt{vc}} \Psi^{(1)}_{\mathtt{v},\mathtt{c}} + f_{\mathtt{v}_2\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \right), \\ \Psi^{(4)}_{\mathtt{c},\mathtt{v}_2} &= f_{\mathtt{v}_2\mathtt{c}}^{-1} \left(f_{\mathtt{v}_1\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} + f_{\mathtt{vc}} \Psi^{(1)}_{\mathtt{v},\mathtt{c}} \right). \end{aligned}$$

This implies that

$$\begin{aligned} f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_1\mathtt{c}} \left(\Psi^{(4)}_{\mathtt{c},\mathtt{v}_1} \cap \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} \right) &= f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_1\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_2\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \right), \\ f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_2\mathtt{c}} \left(\Psi^{(4)}_{\mathtt{c},\mathtt{v}_2} \cap \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \right) &= f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_2\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}} \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_1\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}} \right). \end{aligned}$$

Let $P = f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_1\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_1,\mathtt{c}}$, $Q = f_{\mathtt{vc}}^{-1} f_{\mathtt{v}_2\mathtt{c}} \Psi^{(1)}_{\mathtt{v}_2,\mathtt{c}}$. Then we need to prove that

$$\left(P \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + Q \right) \right) + \left(Q \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + P \right) \right) \supseteq \Psi^{(1)}_{\mathtt{v},\mathtt{c}} \cap (P + Q). \tag{3.3}$$

To prove this, let $d \in \Psi^{(1)}_{\mathtt{v},\mathtt{c}} \cap (P + Q)$, then $d \in \Psi^{(1)}_{\mathtt{v},\mathtt{c}}$. Also $d = p + q$, where $p \in P, q \in Q$. This implies that $p \in \Psi^{(1)}_{\mathtt{v},\mathtt{c}} + Q$ and $q \in \Psi^{(1)}_{\mathtt{v},\mathtt{c}} + P$. Hence $p \in \left(P \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + Q \right) \right)$ and $q \in \left(Q \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + P \right) \right)$. Hence $p + q \in \left(P \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + Q \right) \right) + \left(Q \cap \left(\Psi^{(1)}_{\mathtt{v},\mathtt{c}} + P \right) \right)$. This proves Eqn(3.3) and also implies that B is a stopping constellation. From Lemma 9 it follows that B is the largest stopping constellation contained in C. □

In the next section we define the ensemble of codes resulted by the BP decoder which we call residual ensemble. We then compute the design rate and expectation of total number of codewords of the residual ensemble. Then we show how we can derive the residual degree distribution from the fixed points of density evolution for the BP decoder. This will enable us to compute the conditional entropy when the block length tends to infinity.

3.4 Residual Degree Distribution and Counting Argument

In the previous section we saw that the BP decoder assigns the state $B_{\mathtt{v}}$ to a variable node $\mathtt{v}$, $B_{\mathtt{v}} \in \mathcal{M}_{\text{ini}}$, $\mathtt{v} \in \mathcal{V}$. We say that the *BP state* of $\mathtt{v}$ is $B_{\mathtt{v}}$. Thus in the resulting *residual graph* every variable node is characterized by its degree and state. So, the *residual degree distribution* on the variable node side is given by $\Omega = \{\Omega_{\mathtt{l}V}\}$, where $\Omega_{\mathtt{l}V}$ denotes the fraction of variable nodes which has degree $\mathtt{l}$ and the symbols corresponding to these variable nodes in the set of codewords can take values only in the subspace V, $V \in \mathcal{M}_{\text{ini}}$. Similarly, we define $\Omega_{\mathtt{l}k}$ to be the fraction of variable nodes which can only take values in a subspace of dimension k. More precisely,

$$\Omega_{\mathtt{l}k} = \sum_{V:\dim(V)=k, V\in\mathcal{M}_{\text{ini}}} \Omega_{\mathtt{l}V}.$$

From Lemma 10 we know that the BP state assignment is such that every check node satisfies the decoding failure criterion. In order to define the degree distribution of the check node side, we define the set $\mathcal{S}_{\mathtt{r}}$ which consists of all the $\mathtt{r}$-tuples of subspaces which satisfy the decoding failure criterion. More precisely,

$$\mathcal{S}_{\mathtt{r}} = \left\{ (V_1, \dots, V_{\mathtt{r}}) : V_i \in \mathcal{M}, V_i \subseteq \sum_{j=1, j\neq i}^{\mathtt{r}} V_j, \forall i \in \{1, \dots, \mathtt{r}\} \right\}. \tag{3.4}$$

Using the definition of $\mathcal{S}_{\mathtt{r}}$ in Eqn(3.4), we define the residual check node degree distribution $\Phi = \{\Phi_{\mathtt{r}s}\}$. For $s \in \mathcal{S}_{\mathtt{r}}$, $\Phi_{\mathtt{r}s}$ denotes the fraction of check nodes with degree $\mathtt{r}$ and for every such check node $\mathtt{c}$, the state of its neighboring variable nodes when acted by the corresponding edge labels satisfy $s = \{V_1, \dots, V_{\mathtt{r}}\}$. More precisely, $V_i = f_{\mathtt{v}_i\mathtt{c}} B_{\mathtt{v}_i}$, where $\mathtt{v}_i \in \mathcal{N}(\mathtt{c})$, $B_{\mathtt{v}_i}$ is the BP state of $\mathtt{v}_i$ and $f_{\mathtt{v}_i\mathtt{c}}$ is the edge label of the edge connecting $\mathtt{v}_i$ and $\mathtt{c}$. We say that the check node $\mathtt{c}$ is of type $(\mathtt{r}, s)$ and its i^{th} socket is restricted to the subspace V_i. We do not distinguish between two types of check nodes $(\mathtt{r}, s_1)$ and $(\mathtt{r}, s_2)$, where s_2 is identical to s_1 up to some permutation. Hence only one arbitrary but fixed representative is included in $\mathcal{S}_{\mathtt{r}}$.

We denote the ensemble of all the residual graphs with block length n and degree distribution (Ω, Φ) by $\text{REGL}(n, \Omega, \Phi, m)$ ($\text{REGL}(\Omega, \Phi, m)$ in the asymptotic limit). Thus the ensemble $\text{REGL}(n, \Omega, \Phi, m)$ has $n\Omega_{\mathtt{l}V}$ variable nodes of degree $\mathtt{l}$ which can only take values in subspace V, $V \in \mathcal{M}_{\text{ini}}$. Similarly, there are $n(1-r)\Phi_{\mathtt{r}s}$ check nodes with type $(\mathtt{r}, s)$, $s \in \mathcal{S}_{\mathtt{r}}$, where $\mathcal{S}_{\mathtt{r}}$ is defined in Eqn(3.4). Here r is the design rate of the initial ensemble $\text{EGL}(\Lambda, \Gamma, m)$ which results in the residual ensemble. From now onwards, we say that r is the *initial rate* of the residual ensemble $\text{REGL}(n, \Omega, \Phi, m)$. The initial rate appears due to the fact that in the residual graph the total number of variable and check nodes is the same as in the original graph. All the bipartite graphs with state assignments and edge labels such that they have degree distribution Ω and Φ are included in the ensemble $\text{REGL}(n, \Omega, \Phi, m)$. There are restriction on the edge labels and graph connections. For example, the sockets from a variable node restricted to subspace V_1 of dimension k can only connect to a check node socket restricted to subspace V_2 and $\mathrm{Dim}\,(V_2) = k$. The edge label on such an edge is restricted to those mappings which map subspace V_1 to V_2.

In order to determine the rate of the residual degree distribution, we need to determine the number of *binary constraints* imposed by a check node of type $(\mathtt{r}, s)$. In the next lemma we prove that the number of binary constraints imposed by a check node of type $(\mathtt{r}, s)$ is $\mathrm{Dim}\,(\sum_{i=1}^{\mathtt{r}} V_i)$, where $s = (V_1, \ldots, V_{\mathtt{r}})$.

Lemma 11. *Consider a check node of degree* $\mathtt{r}$ *which represents the parity-check equation*

$$\sum_{i=1}^{r} x_i = 0, \tag{3.5}$$

where x_i *can only take values in the subspace* $V_i \subseteq \mathrm{GF}_2^m$. *Then the number of binary constraints imposed by such a check node is* $Dim\,(\sum_{i=1}^{r} V_i)$.

Proof. The number of possible words are $2^{\sum_{i=1}^{\mathtt{r}} \mathrm{Dim}(V_i)}$. To compute the total number of codewords, note that a codeword satisfies $x_{\mathtt{r}} = \sum_{i=1}^{\mathtt{r}-1} x_i$. This implies that for a codeword $x = (x_1, \ldots, x_{\mathtt{r}})$ satisfying Eqn(3.5), $x_{\mathtt{r}}$ can only take values in $\left(\sum_{i=1}^{\mathtt{r}-1} V_i\right) \cap V_{\mathtt{r}}$. There are $2^{\sum_{i=1}^{\mathtt{r}-1} \mathrm{Dim}(V_i) - \mathrm{Dim}\left(\sum_{i=1}^{\mathtt{r}-1} V_i\right)}$ number of words $(x_1, \ldots, x_{\mathtt{r}-1})$ which satisfy

$$\sum_{i=1}^{\mathtt{r}-1} x_i = b,$$

where $b \in \left(\sum_{i=1}^{\mathtt{r}-1} V_i\right) \cap V_{\mathtt{r}}$. Then the number of binary constraints is given by,

$$\log_2 \left(\frac{2^{\sum_{i=1}^{\mathtt{r}} \mathrm{Dim}(V_i)}}{2^{\sum_{i=1}^{\mathtt{r}-1} \mathrm{Dim}(V_i) - \mathrm{Dim}\left(\sum_{i=1}^{\mathtt{r}-1} V_i\right)} 2^{\mathrm{Dim}\left(\left(\sum_{i=1}^{\mathtt{r}-1} V_i\right) \cap V_{\mathtt{r}}\right)}} \right) \overset{(I)}{=} \mathrm{Dim} \left(\sum_{i=1}^{\mathtt{r}} V_i \right),$$

where in (I) we used the fact that $\mathrm{Dim}\,(A \cap B) = \mathrm{Dim}\,(A) + \mathrm{Dim}\,(B) - \mathrm{Dim}\,(A + B)$. This proves the lemma. □

From now onwards, we denote the number of constraints imposed by a check node of type $(\mathtt{r}, s)$ by $\mathrm{Dim}(s) = \mathrm{Dim}\,(\sum_{i=1}^{\mathtt{r}} V_i)$, where $s = (V_1, \ldots, V_{\mathtt{r}})$. In the next lemma we compute the design rate of the ensemble REGL (n, Ω, Φ, m).

Lemma 12. *The design rate of the residual ensemble REGL* (n, Ω, Φ, m) *with initial rate* r *is given by*

$$r_{res} = 1 - (1 - r) \frac{\sum_{\mathtt{r}} \sum_{s \in \mathcal{S}_{\mathtt{r}}} \Phi_{\mathtt{r}s} Dim(s)}{\sum_{\mathtt{l}} \sum_{V \in \mathcal{M}_{\mathrm{ini}}} \Omega_{\mathtt{l}V} Dim\,(V)}. \tag{3.6}$$

Proof. Using Lemma 11, for the block length n, the number of binary constraints are

$$n(1-r)\sum_{\mathtt{r}}\sum_{s\in\mathcal{S}_{\mathtt{r}}}\Phi_{\mathtt{r}s}\mathrm{Dim}(s).$$

The number of "unconstrained" bits is given by

$$n\sum_{\mathtt{l}}\sum_{V\in\mathcal{M}_{\mathrm{ini}}}\Omega_{\mathtt{l}V}\mathrm{Dim}\left(V\right).$$

Hence the design rate of the residual ensemble is

$$r_{\mathrm{res}} = 1-(1-r)\frac{\sum_{\mathtt{r}}\sum_{s\in\mathcal{S}_{\mathtt{r}}}\Phi_{\mathtt{r}s}\mathrm{Dim}(s)}{\sum_{\mathtt{l}}\sum_{V\in\mathcal{M}_{\mathrm{ini}}}\Omega_{\mathtt{l}V}\mathrm{Dim}\left(V\right)}.$$

□

The design rate r_{res} of the ensemble REGL (n,Φ,Ω,m) is a lower bound on the rate of every code in the ensemble REGL (n,Φ,Ω,m). This is due to the fact that when we compute the design rate, we assume that all the parity-check equations are independent. However, in a code some parity-check equations might be dependent. A criterion for binary LDPC ensembles was derived in [1] which, when satisfied, guarantees that the design rate is the actual rate. Towards generalizing this criterion for residual ensembles REGL (n,Φ,Ω,m), we compute the expectation of N, the total number of codewords in a randomly chosen code from REGL (n,Φ,Ω,m).

Lemma 13. *Let $N\left(E_1,\ldots,E_m\right)$ be the number of codewords in a randomly chosen element of the ensemble REGL (Φ,Ω,m,ϵ) such that, for each such codeword there are E_k edges which are assigned non-zero values and which are connected to variable nodes which can only take values in the subspace of dimension k, $k\in\{1,\ldots,m\}$. Let N be the total number of codewords in a randomly chosen code. Then,*

$$\mathbb{E}(N) = \sum_{E_1=0}^{n\sum_l l\Omega_{l1}}\cdots\sum_{E_m=0}^{n\sum_l l\Omega_{lm}}\mathbb{E}\left(N\left(E_1,\ldots,E_m\right)\right), \tag{3.7}$$

and

$$\mathbb{E}\left(N\left(E_1,\ldots,E_m\right)\right) = \prod_{k=1}^{m}\frac{coef\left(\prod_{k=1}^{m}\prod_l\left(1+\left(2^k-1\right)u_k^l\right)^{n\Omega_{lk}},\prod_{k=1}^{m}u_k^{E_k}\right)}{g(m,k)^{E_k}\binom{n\sum_l l\Omega_{lk}}{E_k}}$$
$$\times\, coef\left(\prod_r\prod_{s\in\mathcal{S}_r}q_s\left(v_1,\ldots,v_m\right)^{n(1-r)\Phi_{rs}},\prod_{k=1}^{m}v_k^{E_k}\right),$$

where $g(m,k)$ is defined in Eqn(3.1), $s\in\mathcal{S}_r$, $s=\{V_1,\ldots,V_r\}$ and $\mathcal{S}_r$ is defined in Eqn(3.4). The function $q_s(v_1,\ldots,v_m)$ is given by

$$q_s(v_1,\ldots,v_m) = \sum_{(i_1\ldots i_k)\subseteq\{1,\ldots,r\}}q_{i_1\ldots i_k}\prod_{j=1}^{k}v_{Dim\left(V_{i_j}\right)}. \tag{3.8}$$

In Eqn(3.8), $q_{i_1 \dots i_k}$ is the number of permissible edge labels assigned to the edges corresponding to $(V_{i_1}, \dots, V_{i_k})$ which will yield a valid codeword when the remaining edges corresponding to $\{V_1, \dots, V_r\} \backslash \{V_{i_1}, \dots, V_{i_k}\}$ carry the value zero.

Proof. The proof is given in the Appendix 3.A. □

To compute the quantity $q_{i_1 \dots i_k}$ appearing in Eqn(3.8), we will make use of the following lemma.

Lemma 14. *Let x_i be a non-zero element of GF_2^m, $M_i \in \mathrm{GL}_2^m$ and $i \in \{1, \dots, r\}$. Let*

$$\mathcal{Z}_r = \left\{ (M_1, \dots, M_r) : \sum_{i=1}^{r} M_i x_i = 0 \right\},$$

and

$$\mathcal{NZ}_r = \left\{ (M_1, \dots, M_r) : \sum_{i=1}^{r} M_i x_i = y \right\},$$

where y is some fixed non-zero element of GF_2^m. Then,

$$\begin{aligned} F_r = |\mathcal{Z}_r| &= \frac{|\,\mathrm{GL}_2^m\,|^r}{2^m} \left(1 + \frac{(-1)^r}{(2^m-1)^{r-1}} \right), \\ G_r = |\mathcal{NZ}_r| &= \frac{|\,\mathrm{GL}_2^m\,|^r}{2^m} \left(1 - \frac{(-1)^r}{(2^m-1)^{r}} \right). \end{aligned}$$

Proof. Note that the sequence $\{F_\mathtt{r}\}_{\mathtt{r}=1}^{\infty}$ satisfies the following recursive relation:

$$F_{\mathtt{r}+1} = \frac{|\,\mathrm{GL}_2^m\,|}{2^m - 1} \left(|\,\mathrm{GL}_2^m\,|^\mathtt{r} - F_\mathtt{r} \right).$$

This follows from the fact if we look at a particular set of $\mathtt{r}$ terms, they must not sum to zero in order to make sure that $\mathtt{r}+1$ terms sum to zero. Also, there are $\frac{|\,\mathrm{GL}_2^m\,|}{2^m-1}$ elements in GL_2^m which maps a given non-zero element x to another non-zero element y, where $x, y \in \mathrm{GF}_2^m$. By solving the recursion, we get

$$F_\mathtt{r} = \frac{|\,\mathrm{GL}_2^m\,|^\mathtt{r}}{2^m} \left(1 + \frac{(-1)^\mathtt{r}}{(2^m-1)^{\mathtt{r}-1}} \right).$$

Similarly, the sequence $\{G_\mathtt{r}\}$ satisfies the recursive relation

$$G_{\mathtt{r}+1} = \frac{|\,\mathrm{GL}_2^m\,|}{2^m - 1} \left((2^m - 2)\, G_\mathtt{r} + F_\mathtt{r} \right).$$

The solution of this relationship yields

$$G_\mathtt{r} = \frac{|\,\mathrm{GL}_2^m\,|^\mathtt{r}}{2^m} \left(1 - \frac{(-1)^\mathtt{r}}{(2^m-1)^{\mathtt{r}}} \right).$$

□

In the remainder of this section, we show how we can compute the *average residual degree distribution* in the limit of infinite block length. First we show that every element of a residual ensemble has uniform probability conditioned on the event that a random residual graph has the degree distribution of the considered residual ensemble.

Lemma 15. *Let REGL* (n, Ω, Φ, m) *be a residual ensemble. Conditioned on the event that a random residual graph is an element of REGL* (n, Ω, Φ, m)*, it has uniform probability of being any element of the ensemble REGL* (n, Ω, Φ, m)*.*

Proof. The proof is given in the Appendix 3.B. □

We denote the ensemble corresponding to the average residual degree distribution by REGL $(n, \Omega, \Phi, m, \epsilon)$. We assume the following concentration result on the degree distribution of a random residual graph around the average residual degree distribution.

Lemma 16. *Consider transmission over the BEC(*ϵ*). Let REGL* $(n, \Omega, \Phi, m, \epsilon)$ *be the average residual degree distribution of the ensemble EGL* (Λ, Γ, m)*. Let REGL* $(n, \Omega_G, \Phi_G, m, \epsilon)$ *denote the residual degree distribution of a random residual graph* G*. Then, for any* $\eta > 0$*,*

$$\lim_{n\to\infty} P\left\{d\left((\Omega, \Phi), (\Omega_G, \Phi_G)\right) \geq \eta\right\} = 0.$$

The distance $d(.,.)$ *is the* L_1 *distance*

$$d\left((\Omega, \Phi), \left(\tilde{\Omega}, \tilde{\Phi}\right)\right) = \sum_{l,V} |\Omega_{lV} - \tilde{\Omega}_{lV}| + \sum_{r} \sum_{s \in \mathcal{S}_r} |\Phi_{rs} - \tilde{\Phi}_{rs}|, \tag{3.9}$$

where $V \in \mathcal{M}_{\text{ini}}$*.*

Asymptotically, Ω and Φ depend on the probability density of the messages of the BP decoder which can be evaluated by density evolution. We derived the density evolution equations for NBLDPC ensemble in Chapter 2, Lemma 4. The fixed point probability of the event that the message from a variable(check) node of degree $\mathtt{l}$ is of dimension i is denoted by $\mathrm{P}_v(i, \mathtt{l})(\mathrm{P}_c(i, \mathtt{l}))$. Then

$$\Omega_{\mathtt{l}V} = \frac{\Lambda_{\mathtt{l}}\, \mathrm{P}_v\left(k, \mathtt{l}+1\right)}{\binom{m}{k}}, \tag{3.10}$$

where $V \in \mathcal{M}_{\text{ini}}$ and $\dim(V) = k$. Note that we use $\mathtt{l}+1$ as we take into account all the incoming messages to compute the estimate of a symbol. We divide by $\binom{m}{k}$ which is the number of initial subspaces of dimension k in order to compute the probability of a specific subspace of dimension k.

Deriving the residual degree distribution on the check node side is less straight forward than on the variable node side. Towards this end, observe that we can determine the BP estimate of a symbol if we know the incoming and the outgoing message to this symbol on one of its connected edges when the BP decoder has converged. Hence if we know all the incoming messages to a check node we can determine the final estimates of the symbols connected to this check node. Thus from the fixed points of density evolution we can determine the residual degree distribution on the check node side. More precisely, consider a check node of degree $\mathtt{r}$. Let $V_1, \ldots, V_{\mathtt{r}}$ be the incoming messages which are independently distributed according to $\{\mathrm{P}_v(i)\}_{i=0}^{m}$. Then the state of the check node is

$$\left\{ V_1 \cap \left(\sum_{j \neq 1} V_j \right), \ldots, V_{\mathtt{r}} \cap \left(\sum_{j \neq \mathtt{r}} V_j \right) \right\}, \tag{3.11}$$

which is an element of $\mathcal{S}_{\mathtt{r}}$, where $\mathcal{S}_{\mathtt{r}}$ has been defined in Eqn(3.4). Thus determining Φ amounts to determining all the elements of $\mathcal{S}_{\mathtt{r}}$ for all the check node degrees $\mathtt{r}$ and to compute the probability distribution over $\mathcal{S}_{\mathtt{r}}$. Currently we do not have an explicit characterization of these quantities for all values of m. In the following subsection we will exemplify how this can be done for the case $m = 2$.

3.4.1 Calculating the Check Node Distribution for $m = 2$

In this subsection we will assume that $m = 2$. Note that there are five different subspaces of GF_2^2. There is one each of dimension zero (containing the origin only) and of dimension two (GF_2^2). There are three different subspaces of dimension one. We number the three different subspaces of dimension one by $1, 2, 3$ in an arbitrary but fixed way. The sum of two different subspaces of dimension one gives the subspace of dimension two. More precisely, let S_1, S_2, S_3 be the subspaces of dimension one and T be the subspace of dimension two. Then,

$$T = S_i + S_j, \quad i, j \in \{1, 2, 3\}, i \neq j. \tag{3.12}$$

We denote by $n_0(n_2)$ the number of subspaces of dimension $0(2)$ in the state $s \in \mathcal{S}_{\mathtt{r}}$ of a check node. Similarly the number of different subspaces of dimension one is given by $n_1(i)$, $i \in \{1, 2, 3\}$ and let

$$n_1 = \sum_{i=1}^{3} n_1(i).$$

Now we enumerate all the possible states of a check node of degree $\mathtt{r}$ which satisfy the decoding failure criterion and derive their probabilities in terms of the fixed points of density evolution. We recall that when the converged incoming messages to a check node are $\{V_1, \ldots, V_{\mathtt{r}}\}$ then its state is given by Eqn(3.11).

1. The state s for which $n_2 \geq 2$ satisfies the decoding failure criterion. Its probability is given by

$$\Phi_{\mathtt{r}s} = \frac{\mathtt{r}!}{n_0! n_1(1)! n_1(2)! n_1(3)! n_2!} \mathrm{P}_v(2)^{n_2} \left(\frac{\mathrm{P}_v(1)}{3} \right)^{n_1(1)+n_1(2)+n_1(3)} \mathrm{P}_v(0)^{n_0}. \tag{3.13}$$

The Eqn(3.13) can be understood by observing that

$$V_i \cap \left(\sum_{i \neq j} V_j \right) = V_i,$$

if

$$\exists k \in \{1, \ldots, \mathtt{r}\}, k \neq i, \dim (V_k) = 2.$$

Thus the socket i is restricted to its corresponding incoming message V_i.

2. When the state s is such that $n_2 = 1$, then for s to satisfy the decoding failure criterion we need that $n_1(i) > 0$, $n_1(j) > 0$, and $i \neq j$. In this case also the sockets are restricted to their corresponding incoming messages. So, the probability is the same as given by Eqn(3.13).

3. The state for which only one of $n_1(1), n_1(2), n_1(3)$ is positive and greater than one with $n_2 = 0$ satisfies the decoding failure criterion. Lets us assume with out loss of generality (w.l.o.g.) that $n_1(1) > 1$. The probability of such a state s is given by

$$\begin{aligned}\Phi_{\mathtt{r}s} = & \frac{\mathtt{r}!}{1!\,(n_1(1)-1)!\,(\mathtt{r}-n_1(1))!} \mathrm{P}_v(2) \left(\frac{\mathrm{P}_v(1)}{3}\right)^{n_1(1)-1} \mathrm{P}_v(0)^{\mathtt{r}-n_1(1)-1} \\ & + \frac{\mathtt{r}!}{1! n_1(1)!\,(\mathtt{r}-n_1(1)-1)!} 2 \left(\frac{\mathrm{P}_v(1)}{3}\right)^{n_1(1)+1} \mathrm{P}_v(0)^{\mathtt{r}-n_1(1)-1} \mathbb{1}_{n_1(1)<\mathtt{r}} \\ & + \frac{\mathtt{r}!}{n_1(1)!\,(\mathtt{r}-n_1(1))!} \left(\frac{\mathrm{P}_v(1)}{3}\right)^{n_1(1)} \mathrm{P}_v(0)^{\mathtt{r}-n_1(1)}. \end{aligned} \quad (3.14)$$

In order to understand Eqn(3.14) observe that for $n_2 = 0$ to hold, at most one of the incoming messages can be of dimension two. If there are more than one incoming messages of dimension two, then the state of the check node has $n_2 \geq 2$. The first term on the right hand side of Eqn(3.14) corresponds to the case when one of the incoming message is of dimension two, $n_1(1) - 1$ incoming messages which are equal to S_1 and the rest of the messages are equal to the subspace of dimension zero. By Eqn(3.11), we see that such a combination of messages result in the desired check node state. Similarly, we derive the remaining terms of Eqn(3.14).

4. Consider the state s for which two different subspaces of dimension one are present at least twice and the remaining subspace of dimension is absent with $n_2 = 0$. It can be seen that s satisfies decoding failure criterion. Assume w.l.o.g. that $n_1(1) \geq 2, n_1(2) \geq 2$ with $n_1(3) = 0$ and $n_2 = 0$. Its probability can be derived by using arguments similar to the previous state and Eqn(3.12). The probability is given by

$$\Phi_{\mathtt{r}s} = \frac{\mathtt{r}!}{n_1(1)! n_1(2)!\,(\mathtt{r}-n_1(1)-n_1(2))!} \left(\frac{\mathrm{P}_v(1)}{3}\right)^{n_1(1)+n_1(2)} \times \mathrm{P}_v(0)^{\mathtt{r}-n_1(1)-n_1(2)}.$$

5. The state s for which all the subspaces of dimension one are present at least once i.e. $n_1(i) > 0$, $\forall i \in \{1,2,3\}$ and $n_2 = 0$ satisfies the decoding failure criterion. The probability of such a state is given by

$$\Phi_{\mathtt{r}s} = \frac{\mathtt{r}!}{n_1(1)!n_1(2)!n_1(3)!\,(\mathtt{r}-n_1)!}\left(\frac{\mathrm{P}_v(1)}{3}\right)^{n_1}\mathrm{P}_v(0)^{\mathtt{r}-n_1}. \tag{3.15}$$

   Note that in this case also each socket is restricted to its incoming message. This explains Eqn(3.15).

6. The state for which $n_0 = \mathtt{r}$ satisfies the decoding failure criterion. Its probability is given by

$$\Phi_{\mathtt{r}s} = \mathrm{P}_v(0)^{\mathtt{r}} + \mathtt{r}\,\mathrm{P}_v(0)^{\mathtt{r}-1}\left(\mathrm{P}_v(1)+\mathrm{P}_v(2)\right) + 3\frac{\mathtt{r}!}{(\mathtt{r}-2)!}\left(\frac{\mathrm{P}_v(1)}{3}\right)^2\mathrm{P}_v(0)^{\mathtt{r}-2}. \tag{3.16}$$

   The first term on the right hand side of Eqn(3.16) is obvious as it corresponds to the case when all the incoming messages have dimension zero. The second term is for the following combination of messages. One of the message is a subspace of dimension one (two) and the remaining messages are subspaces of dimension zero. As the dimension of the intersection of the subspace of dimension one (two) with a subspace of dimension zero is zero, this combination of incoming messages result in the desired state. The other terms are similarly derived.

In the next section we show that the average conditional entropy of the transmitted codeword is given by the design rate of the average residual ensemble under appropriate technical conditions.

3.5 Equality between Conditional Entropy and Rate

In this section we assume that $m = 2$. In order to derive a sufficient condition which guarantees that asymptotically almost every code in the residual ensemble has its rate equal to design rate, we first find the generating functions $q_s(v_1, v_2)$ defined in Eqn(3.8). Consider a check node of degree $\mathtt{r}$ with the state s such that there are n_0 sockets corresponding to subspaces of dimension zero, $n_1(i)$ sockets corresponding to subspace i of dimension one, $i \in \{1,2,3\}$ and n_2 sockets corresponding to subspace of dimension two. Then the generating function for this type is given by

$$q_s(v_1, v_2) = \sum_{J_1}\left\{\binom{n_2}{i}F_i v_2^i\binom{n_1(1)}{j_1}\binom{n_1(2)}{j_2}\binom{n_1(3)}{j_3}2^{j_1+j_2+j_3}\right.$$
$$\left. v_1^{j_1+j_2+j_3}\right\} + \sum_{J_2}\left\{\binom{n_2}{i}G_i v_2^i\binom{n_1(1)}{j_1}\binom{n_1(2)}{j_2}\binom{n_1(3)}{j_3}2^{j_1+j_2+j_3}v_1^{j_1+j_2+j_3}\right\}, \tag{3.17}$$

where F_i, G_i are defined in Lemma 14, J_1 corresponds to summation over terms such that $j_1(1), j_1(2), j_1(3)$ are all even or all odd. J_2 is the complement of J_1. A simplification of $q_s(v_1, v_2)$ yields

$$\begin{aligned} q_s(v_1, v_2) = {} & f(v_1)\left(\frac{(1+6v_2)^{n_2}+3(1-2v_2)^{n_2}}{4}\right) \\ & + \left((1+2v_1)^{n_1}-f(v_1)\right)\left(\frac{(1+6v_2)^{n_2}-(1-2v_2)^{n_2}}{4}\right), \end{aligned} \tag{3.18}$$

where,

$$f(v_1) = \prod_{i=1}^{3}\frac{(1+2v_1)^{n_1(i)}-(1-2v_1)^{n_1(i)}}{2} + \prod_{i=1}^{3}\frac{(1+2v_1)^{n_1(i)}+(1-2v_1)^{n_1(i)}}{2}.$$

In the following lemma we upper bound the difference between the growth rate of the expectation of total number of codewords and the design rate of the residual ensemble.

Lemma 17. *Let N be the total number of codewords of a randomly chosen code from the ensemble $REGL(n, \Omega, \Phi, 2)$. Then*

$$\lim_{n\to\infty}\frac{\log(\mathbb{E}(N))}{n} - r_{res} \leq \theta(u_1, u_2).$$

The function $\theta(u_1, u_2)$ is defined in the following equation for $u_1 \in [0, \infty)$ and $u_2 \in [0, \infty)$.

$$\begin{aligned} \theta(u_1, u_2) = {} & \sum_{\imath}\Omega_{\imath 1}\log_2\left(\frac{1+u_1^{\imath}}{2}\right) + \sum_{\imath}\Omega_{\imath 2}\log_2\left(\frac{1+3u_2^{\imath}}{4}\right) + \\ & (1-r)\sum_{r}\sum_{s\in\mathcal{S}_r}\Phi_{rs}\log\left(h_s(u_1,u_2)2^{Dim(s)}\right) \\ & - t_{+1}(1)\log_2(t_{+1}(1)) - t_{+2}(1)\log_2(t_{+2}(1)), \end{aligned} \tag{3.19}$$

where,

$$\begin{aligned} h_s(u_1, u_2) = {} & p(u_1)\frac{t_{+2}(u_2)^{n_2}+3t_{-2}(u_2)^{n_2}}{4} + \\ & \left(t_{+1}^{n_1}(u_1) - p(u_1)\right)\frac{t_{+2}(u_2)^{n_2}-t_{-2}(u_2)^{n_2}}{4}, \end{aligned}$$

$$t_{+1}(u_1) = \sum_{\imath}\imath\Omega_{\imath 1}\frac{1+u_1^{\imath-1}}{1+u_1^{\imath}}, \quad t_{-1}(u_1) = \sum_{\imath}\imath\Omega_{\imath 1}\frac{1-u_1^{\imath-1}}{1+u_1^{\imath}},$$

$$t_{+2}(u_2) = \sum_{\imath}\imath\Omega_{\imath 2}\frac{1+3u_2^{\imath-1}}{1+3u_2^{\imath}}, \quad t_{-2}(u_2) = \sum_{\imath}\imath\Omega_{\imath 2}\frac{1-u_2^{\imath-1}}{1+3u_2^{\imath}},$$

$$p(u_1) = \prod_{i=1}^{3}\frac{t_{+1}(u_1)^{n_1(i)}-t_{-1}(u_1)^{n_1(i)}}{2} + \prod_{i=1}^{3}\frac{t_{+1}(u_1)^{n_1(i)}+t_{-1}(u_1)^{n_1(i)}}{2}.$$

Proof. Let $E_1 = ne_1 \sum_{\mathtt{l}} \mathtt{l}\Omega_{11}$, $E_2 = ne_2 \sum_{\mathtt{l}} \mathtt{l}\Omega_{12}$, $e_1 \in [0,1]$ and $e_2 \in [0,1]$. By Eqn(3.7),

$$\lim_{n\to\infty} \frac{\log\left(\mathbb{E}(N)\right)}{n} = \sup_{e_1,e_2} \lim_{n\to\infty} \frac{\log\left(\mathbb{E}(N(e_1,e_2))\right)}{n}.$$

Using Lemma 13 and the Hayman approximation of [26] for the coef term, we get

$$\phi\left(e_1,e_2\right) = \lim_{n\to\infty} \frac{\log_2\left(\mathbb{E}\left[N\left(e_1,e_2\right)\right]\right)}{n} := \inf_{u_1,u_2,v_1,v_2} \phi_1\left(e_1,e_2,v_1,v_2,u_1,u_2\right),$$

where

$$\begin{aligned}\phi_1\left(e_1,e_2,v_1,v_2,u_1,u_2\right) = \sum_{\mathtt{l}} \Omega_{11} \log_2\left(1+u_1^{\mathtt{l}}\right) - e_1 \log_2(u_1) \sum_{\mathtt{l}} \mathtt{l}\Omega_{11} + \\ \sum_{\mathtt{l}} \Omega_{12} \log_2\left(1+3u_2^{\mathtt{l}}\right) - e_2 \log_2(u_2) \sum_{\mathtt{l}} \mathtt{l}\Omega_{12} \\ + (1-r) \sum_{\mathtt{r}} \sum_{s\in\mathcal{S}_{\mathtt{r}}} \Phi_{\mathtt{r}s} \log\left(h_s(v_1,v_2)\right) \\ -e_1\left(\log_2(v_1)+1\right) \sum_{\mathtt{l}} \mathtt{l}\Omega_{11} - e_2\left(\log_2(v_2)+\log_2(6)\right) \sum_{\mathtt{l}} \mathtt{l}\Omega_{12} \\ - \left(\sum_{\mathtt{l}} \mathtt{l}\Omega_{11}\right) h\left(e_1\right) - \left(\sum_{\mathtt{l}} \mathtt{l}\Omega_{12}\right) h\left(e_2\right).\end{aligned}$$

Hence we would like to compute $\sup_{e_1,e_2} \phi\left(e_1,e_2\right)$, where $e_1 \in [0,1], e_2 \in [0,1]$. Thus,

$$\sup_{e_1,e_2} \phi\left(e_1,e_2\right) = \sup_{e_1,e_2} \inf_{v_1,v_2,u_1,u_2} \phi_1\left(e_1,e_2,v_1,v_2,u_1,u_2\right). \tag{3.20}$$

We find an upper bound on $\sup_{e_1,e_2} \phi\left(e_1,e_2\right)$. Towards this end, first take the derivative of $\phi_1\left(e_1,e_2,u_1,u_2,v_1,v_2\right)$ with respect to e_1 and e_2 and equate it to zero. This gives,

$$e_1 = \frac{2u_1v_1}{1+2u_1v_1}, \quad e_2 = \frac{6u_2v_2}{1+6u_2v_2}.$$

We substitute this in the expression for $\phi_1\left(e_1,e_2,u_1,u_2,v_1,v_2\right)$. After that we take the derivative with respect to u_1, u_2 and equate these to zero. Solving for v_1 and v_2 gives

$$v_1(u_1) = \frac{1}{2} \frac{\sum_{\mathtt{l}} \mathtt{l}\Omega_{11} \frac{u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}}{\sum_{\mathtt{l}} \mathtt{l}\Omega_{11} \frac{1}{1+u_1^{\mathtt{l}}}}, \quad v_2(u_2) = \frac{1}{6} \frac{\sum_{\mathtt{l}} \mathtt{l}\Omega_{12} \frac{3u_2^{\mathtt{l}-1}}{1+3u_2^{\mathtt{l}}}}{\sum_{\mathtt{l}} \mathtt{l}\Omega_{12} \frac{1}{1+3u_2^{\mathtt{l}}}}.$$

Substituting the expression for $v_1(u_1)$ and $v_2(u_2)$ in the expression for e_1 and e_2, we get

$$e_1(u_1) = \frac{\sum_{\mathtt{l}} \mathtt{l}\Omega_{11} \frac{u_1^{\mathtt{l}}}{1+u_1^{\mathtt{l}}}}{\sum_{\mathtt{l}} \mathtt{l}\Omega_{11}}, \quad e_2(v) = \frac{\sum_{\mathtt{l}} \mathtt{l}\Omega_{12} \frac{3u_2^{\mathtt{l}}}{1+3u_2^{\mathtt{l}}}}{\sum_{\mathtt{l}} \mathtt{l}\Omega_{12}}.$$

Let $\theta(u_1,u_2) = \phi_1\left(e_1(u_1), e_2(u_2), v_1(u_1), v_2(u_2), u_1, u_2\right) - r_{res}$. By rearranging terms we get the desired expression for $\theta(u_1,u_2)$. □

Now, we give the criterion which, when satisfied, yields that almost every code in the residual ensemble has its rate equal to its design rate.

Lemma 18. *Let G be a code chosen uniformly at random from the residual ensemble REGL$\,(n,\Omega,\Phi,m)$ and let r_G be its rate. If the function $\theta(u_1,u_2)$ defined in Eqn(3.19) satisfies*

$$\theta(u_1,u_2) \leq 0, \quad \forall u_1 \in [0,\infty), u_2 \in [0,\infty),$$

then there exists $B>0$ and a positive integer n_0 such that, for $n>n_0$ and any $\eta>0$,

$$P\left(|r_G - r_{res}| > \eta\right) \leq e^{-Bn\eta}.$$

In addition, there exist $C>0$ such that, for $n>n_0$,

$$\mathbb{E}\left(|r_G - r_{res}|\right) \leq C\frac{\log(n)}{n},$$

where r_{res} is the design rate of the residual ensemble REGL$\,(n,\Omega,\Phi,2)$.

Proof. This follows from the same arguments as that of Lemma 7 of [1]. □

We prove some properties of the function $\theta(u_1,u_2)$ defined in Eqn(3.19) which will help us in showing that the condition entropy of NBLDPC ensemble EGL$\,(\Lambda,\Gamma,2)$ is given by the design rate of the average residual ensemble. We will prove our arguments for left regular ensembles. We believe that they also hold for irregular ensembles. In the rest of this section we assume that $\Lambda(x) = x^{1}$.

By Lemma 18, the maximum of the function $\theta(u_1,u_2)$ (defined in Eqn(3.19) determines if the rate of of the residual ensemble is equal to its design rate. We prove that $\theta(u_1,u_2)$ attains its maximum in the unit square $[0,1]^2$ for the left regular residual ensemble.

Lemma 19. *Consider the ensemble REGL$\,(\Omega,\Phi,m)$ which is left regular with left degree $\mathtt{l}$. Then the function $\theta(u_1,u_2)$ defined in Eqn(3.19) attains its maximum inside the unit square. More precisely,*

$$\theta(u_1,u_2) \leq \begin{cases} \theta\left(\frac{1}{u_1}, u_2\right), & \forall u_1 \in [1,\infty), \quad \forall u_2 \in [0,1], \\ \theta\left(u_1, x(u_2)\right), & \forall u_1 \in [0,1], \quad \forall u_2 \in [1,\infty), \\ \theta\left(\frac{1}{u_1}, x(u_2)\right), & \forall u_1 \in [1,\infty), \quad \forall u_2 \in [1,\infty), \end{cases}$$

where

$$x(u_2) = \left(\frac{u_2^{\mathtt{l}-1}+1}{3u_2^{\mathtt{l}-1}-1}\right)^{\frac{1}{\mathtt{l}-1}}, \quad \forall u_2 \in [1,\infty).$$

Also, the function $\theta(u_1,u_2)$ depends smoothly on its residual degree distribution. More precisely, there exist constants $B_1, B_2, B_3 > 0$ such that, for any two left regular residual degree distributions REGL $(\Omega, \Phi, 2)$ and REGL $\left(\tilde{\Omega}, \tilde{\Phi}, 2\right)$ having the same left degree and the same maximum check node degree, the corresponding functions $\theta\,(u_1,u_2)$ and $\tilde{\theta}\,(u_1,u_2)$ satisfy for $u_1, u_2 \in [0,1]$,

$$\left|\theta\,(u_1,u_2) - \tilde{\theta}\,(u_1,u_2)\right| \leq d\left((\Omega,\Phi),\left(\tilde{\Omega},\tilde{\Phi}\right)\right) \\ \left(B_1(1-u_1)^2 + B_2(1-u_1)(1-u_2) + B_3(1-u_2)^2\right), \quad (3.21)$$

where $d(.,.)$ is the L_1 distance defined in Eqn(3.9).

Proof. We defer the proof to the Appendix 3.C. □

We now show the equality between the design rate of the residual ensemble and the average conditional entropy of the transmitted ensemble.

Lemma 20. *Consider transmission over the BEC(ϵ). Let G be a code chosen uniformly at random from the left regular NBLDPC ensemble EGL (n, Λ, Γ, m) with left degree $\mathtt{l}$ and let $H_G\,(X|Y)$ be its conditional entropy. Denote the average residual ensemble of EGL $(n, \Lambda, \Gamma, 2)$ by REGL $(n, \Omega, \Phi, 2, \epsilon)$. Consider the corresponding function $\theta(u_1,u_2)$ for REGL $(n, \Omega, \Phi, 2, \epsilon)$ defined in Eqn(3.19). Assume that $\theta(u_1,u_2)$ has a unique global maximum at $(u_1,u_2) = (1,1)$ for $u_1 \in [0,\infty), u_2 \in [0,\infty)$, with*

$$\left.\frac{\partial^2\theta\,(u_1,u_2)}{\partial u_1^2}\frac{\partial^2\theta\,(u_1,u_2)}{\partial u_2^2} - \left(\frac{\partial^2\theta\,(u_1,u_2)}{\partial u_1 \partial u_2}\right)^2\right|_{u_1=1,u_2=1} > 0, \quad (3.22)$$

$$\left.\frac{\partial^2\theta\,(u_1,u_2)}{\partial u_1^2}\right|_{u_1=1,u_2=1} < 0. \quad (3.23)$$

Then

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}\left(H_G(X|Y)\right) = r_{res},$$

where r_{res} is the design rate of the average residual ensemble REGL $(n, \Omega, \Phi, 2, \epsilon)$.

Proof. This lemma is a straightforward generalization of [1, Thm. 10]. So the proof is very similar to that of [1, Thm. 10]. The Eqns(3.22, 3.23) guarantee that the point $(u_1,u_2) = 1$ is a maximum and imply that

$$\left.\frac{\partial^2\theta\,(u_1,u_2)}{\partial u_2^2}\right|_{u_1=1,u_2=1} < 0. \quad (3.24)$$

By Eqns(3.23, 3.24), the assumption that the point $(1,1)$ is the unique global maximum of $\theta(u_1,u_2)$ and

$$\theta(1,1) = 0, \quad \left.\frac{\partial\theta(u_1,u_2)}{\partial u_1}\right|_{u_1=1,u_2=1} = 0, \quad \left.\frac{\partial\theta(u_1,u_2)}{\partial u_2}\right|_{u_1=1,u_2=1} = 0,$$

there exist constants $\delta, C_1, C_3 > 0$ and C_2 such that $\forall u_1 \in [0,1], u_2 \in [0,1]$

$$\theta(u_1,u_2) \le -\delta\left(C_1(1-u_1)^2 + C_2(1-u_1)(1-u_2) + C_3(1-u_2)^2\right),$$

and

$$\left(C_1(1-u_1)^2 + C_2(1-u_1)(1-u_2) + C_3(1-u_2)^2\right) > 0, \forall u_1 \in [0,1), u_2 \in [0,1).$$

Using Lemma 19, we observe that there exist $\eta > 0$ such that for every residual degree distribution pair $(\tilde{\Omega}, \tilde{\Phi})$ which satisfies $d\left((\Omega,\Phi),(\tilde{\Omega},\tilde{\Phi})\right) \le \eta$, its corresponding function $\tilde{\theta}(u_1,u_2)$ is upper bounded in the interval $u_1 \in [0,1], u_2 \in [0,1]$ by

$$\tilde{\theta}(u_1,u_2) \le -\frac{\delta}{2}\left(C_1(1-u_1)^2 + C_2(1-u_1)(1-u_2) + C_3(1-u_2)^2\right).$$

Thus Lemma 18 is applicable to REGL $\left(n, \tilde{\Omega}, \tilde{\Phi}, 2\right)$. So, the design rate $\tilde{r}_{res}$ of REGL $\left(n, \tilde{\Omega}, \tilde{\Phi}, 2\right)$ is equal to its average rate $\tilde{r}$. Let $\mathcal{Q}(\eta)$ be the set of residual ensembles whose degree distribution pair $\left(\tilde{\Omega}, \tilde{\Phi}\right)$ satisfy $d\left((\Omega,\Phi),(\tilde{\Omega},\tilde{\Phi})\right) \le \eta$. Let $\mathrm{P}_\epsilon\left(\tilde{\mathcal{R}}\right)$ be the probability that a random residual graph belongs to the residual ensemble $\tilde{\mathcal{R}} = \text{REGL}\left(n, \tilde{\Omega}, \tilde{\Phi}, 2\right)$. Then the conditional entropy of the ensemble REGL $(n, \Lambda, \Gamma, 2)$ is given by

$$\begin{aligned} \frac{1}{n}\mathbb{E}\left[H_G(X|Y)\right] &= \sum_{\tilde{\mathcal{R}}} \mathrm{P}_\epsilon\left(\tilde{\mathcal{R}}\right)\tilde{r}, \\ &= \sum_{\tilde{\mathcal{R}} \in \mathcal{Q}(\eta)} \mathrm{P}_\epsilon\left(\tilde{\mathcal{R}}\right)\tilde{r} + \gamma(n,\eta). \end{aligned}$$

By Lemma 16 and the fact that $\tilde{r} \le 1$, the term $\gamma(n,\eta)$ satisfies

$$\lim_{n\to\infty} \gamma(n,\eta) = 0.$$

Now,

$$\begin{aligned} \left|\frac{1}{n}\mathbb{E}\left[H_G(X|Y)\right] - r_{res}\right| &\le \sum_{\tilde{\mathcal{R}} \in \mathcal{Q}(\eta)} \mathrm{P}_\epsilon\left(\tilde{\mathcal{R}}\right)|\tilde{r} - r_{res}| + \gamma'(n,\eta), \\ &\le \sum_{\tilde{\mathcal{R}} \in \mathcal{Q}(\eta)} \mathrm{P}_\epsilon\left(\tilde{\mathcal{R}}\right)|\tilde{r}_{res} - r_{res}| + \gamma'(n,\eta), \end{aligned}$$

where by Lemma 18

$$\lim_{n\to\infty} \gamma'(n,\eta) = 0.$$

Note that there exists a constant $C > 0$ such that

$$|\tilde{r}_{res} - r_{res}| \le Cd\left(\left(\tilde{\Omega}, \tilde{\Phi}\right), (\Omega,\Phi)\right).$$

This implies that

$$\lim_{n\to\infty}\left|\frac{1}{n}\mathbb{E}\left[H_G(X|Y)\right]-r_{res}\right|\leq C\eta.$$

As we can chose η to be arbitrarily small, the proof is completed. □

3.5.1 Example

Consider the ensemble EGL$(\lambda,\rho,2)$, where $\lambda(x)=x$, $\rho(x)=x^3$. Note that it is left regular with $\mathtt{l}=2$. Its BP threshold is equal to $\epsilon^{\mathrm{BP}}=0.4096$. For $\epsilon\geq 0.41$, the function $\theta\left(u_1,u_2\right)$ corresponding to the average residual degree distribution is non-positive for $u_1,u_2\in[0,1]$. Also, the assumptions of Lemma 20 are satisfied by $\theta\left(u_1,u_2\right)$ for $\epsilon\geq 0.41$. So, by Lemma 18 the rate of the average residual ensemble is equal to its design rate for $\epsilon\geq 0.41$, which in turn, is equal to the normalized average conditional entropy of the ensemble EGL$(\lambda,\rho,2)$ by Lemma 20. This implies that $\epsilon=0.41$ is the MAP threshold ϵ^{MAP} as the design rate becomes zero for this erasure probability which is equal to the normalized average conditional entropy.

Another approach to compute the MAP threshold is by applying the Maxwell construction to the EBP GEXIT function [1]. In order to define the EBP GEXIT function, let us write the check node side density evolution map given in Eqns(2.14, 2.15) as

$$\mathrm{P}_c^{(l)}=G_c\left(\mathrm{P}_v^{(l)}\right),$$

where $\mathrm{P}_c^{(l)}(\mathrm{P}_v^{(l)})$ is the probability distribution of the dimension of the message emanating from the check (variable) node side in the l^{th} iteration. Similarly, we write the density evolution map on the variable node side given in Eqns(2.16, 2.17) as

$$\mathrm{P}_v^{(l+1)}=G_v\left(\epsilon,\mathrm{P}_c^{(l)}\right).$$

Thus the density evolution recursion can be written as

$$\mathrm{P}_v^{(l+1)}=G(\epsilon,\mathrm{P}_v^{(l)})=G_v\left(\epsilon,G_c\left(\mathrm{P}_v^{(l)}\right)\right). \tag{3.25}$$

We also define another density evolution map on the variable node side, where we do not take into account the channel observation of a variable node. But, we take into account all the incoming messages to this variable node. This corresponds to estimating the variable node from extrinsic observations. Let us denote the resulting distribution as $\mathrm{P}_{v,\mathrm{ext}}^{l}$ and the corresponding map by $G_{v,\mathrm{ext}}$. Then,

$$\mathrm{P}_{v,\mathrm{ext}}^{l}=G_{v,\mathrm{ext}}\left(\mathrm{P}_c^{(l)}\right).$$

Now, the EBP GEXIT function is a parametric function of the fixed point pairs $(\epsilon^h,\mathrm{P}_v^h)$ of the density evolution map given in Eqn(3.25), i.e., $\mathrm{P}_v^h=G\left(\epsilon_h,\mathrm{P}_v^h\right)$, where $h \in [0,1]$ is the normalized entropy of P_v^h, i.e.,

$$\frac{1}{2}\sum_{i=0}^{2} i\,\mathrm{P}_v^h(i) = h.$$

For the EBP GEXIT function, the x-coordinate is ϵ_h and the y-coordinate is h_{ext}, where h_{ext} is the BP extrinsic entropy of a bit, i.e.,

$$h_{\mathrm{ext}} = \frac{\mathrm{P}_{v,\mathrm{ext}}^h(1)}{3} + \frac{\epsilon_h\,\mathrm{P}_{v,\mathrm{ext}}^h(1)}{3} + \mathrm{P}_{v,\mathrm{ext}}^h(2), \tag{3.26}$$

where $\mathrm{P}_{v,\mathrm{ext}}^h = G_{v,\mathrm{ext}}\left(G_c(\mathrm{P}_v^h)\right)$. Note that the first term of Eqn(3.26) takes care of the fact that among the three subspaces of dimension one, the value of a given bit is completely erased in one of them. The second term corresponds to the subspace of dimension one in which both the bits take the same value. So, for one to be erased another one should also be erased. The last term takes care of the subspace of dimension two.

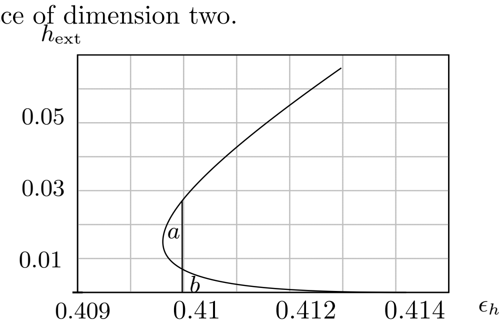


Figure 3.1: EBP curve for EGL $(x, x^3, 2)$. The MAP threshold is 0.41 as computed by the Maxwell construction. The point $\epsilon = 0.41$ corresponds to the erasure probability such that the areas a and b are equal.

When we compute the EBP GEXIT function for the ensemble EGL $(\lambda, \rho, 2)$ (shown in Fig. 3.1) and apply the Maxwell construction of [1], we get the MAP threshold $\epsilon^{\mathrm{MAP}} = 0.41$. The Maxwell construction corresponds to finding the erasure probability such that the areas a and b shown in Fig. 3.1 are equal. The MAP threshold given by the Maxwell construction coincides with the upper bound on the MAP threshold derived by computing the conditional entropy of the average residual ensemble.

3.6 Conclusion

We have defined the stopping constellations for NBLDPC codes. Then we showed that both the peeling and BP decoder get stuck in the largest stopping constellation contained in the channel state assignment. After that we computed the average residual degree distribution for NBLDPC ensembles. We generalized the criterion of [1] which, when satisfied, yields that the actual rate of the residual ensemble is equal to its design rate. We observed that the Maxwell construction of [1] seem to also hold in the setting of NBLDPC codes. Some of our arguments are restricted to the case $m = 2$. It will be of interest to generalize these arguments to any value of m.

3.A Proof of Lemma 13

Proof. The proof of Eqn(3.7) is straight forward. Now,

$$\begin{aligned}\mathbb{E}\left[N\left(E_1,\ldots,E_m\right)\right] &= \sum_{w\in W\left(E_1,\ldots,E_m\right)} \mathrm{P}\left(w \text{ is a codeword}\right)\\ &= \left|W\left(E_1,\ldots,E_m\right)\right|\mathrm{P}\left(w \text{ is a codeword}\right). \quad (3.27)\end{aligned}$$

Here $W\left(E_1,\ldots,E_m\right)$ denotes the set of words which will yield E_k edges which are assigned non-zero values and which are connected to variable nodes which can take values only in subspace of dimension k, $k \in \{1,\ldots,m\}$. Now,

$$\left|W\left(E_1,\ldots,E_m\right)\right| = \prod_{k=1}^{m} \operatorname{coef}\left(\prod_{\mathtt{l}} \left(1+\left(2^k-1\right)u_k^{\mathtt{l}}\right)^{n\Omega_{\mathtt{l}k}}, u_k^{E_k}\right), \qquad (3.28)$$

The factor $2^k - 1$ in the coef term is to take into account that there are $2^k - 1$ non-zero symbols in a subspace of dimension k. We have to compute $\mathrm{P}\left(w \text{ is a codeword}\right)$, where $w \in W\left(E_1,\ldots,E_m\right)$ is some fixed word. Then

$$\mathrm{P}\left(w \text{ is a codeword}\right) = \frac{\left|\mathcal{G}\left(w\right)\right|}{\left|\mathrm{REGL}\left(n,\Omega,\Phi,m\right)\right|}, \qquad (3.29)$$

where $\mathcal{G}\left(w\right) \subseteq \mathrm{REGL}\left(n,\Omega,\Phi,m\right)$ is the set of codes for which w is codeword. Total number of graphs in $\mathrm{REGL}\left(n,\Omega,\Phi,m\right)$ is given by

$$\left|\mathrm{REGL}\left(n,\Omega,\Phi,m\right)\right| = \prod_{k=0}^{m}\left(n\sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}k}\right)! g(m,k)^{n\sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}k}}. \qquad (3.30)$$

The factorial terms correspond to permutations of edges coming from variable nodes of different dimensions. There are $g(m,k)$ mappings which map a given subspace of dimension k to another given subspace of dimension k, where $g(m,k)$ is defined in Eqn(3.1). This explains the power term.

Next we count the number of graphs for which w is a codeword. This number is given by

$$\begin{aligned}|\mathcal{G}(w)| = \operatorname{coef}&\left(\prod_{\mathtt{r}}\prod_{s\in\mathcal{S}_{\mathtt{r}}} q_s(v_1,\ldots,v_m)^{n(1-r)\Phi_{\mathtt{r}s}}, \prod_{k=1}^{m} v_k^{E_k}\right)\left(n\sum\mathtt{l}\Omega_{\mathtt{l}0}\right)!\\ &\left(g(m,0)^{n\sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}0}}\right)\prod_{k=1}^{m}\left(E_k!\left(n\sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}k}-E_k\right)! g(m,k)^{n\sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}k}-E_k}\right).\end{aligned}$$
$$(3.31)$$

The factorial terms in Eqn(3.31) correspond to permuting edges among their class. This means that the edges which are attached to a variable node restricted to a subspace of dimension k and carrying a non-zero values can only be permuted among themselves. The power terms correspond to assigning permissible elements of GL_2^m to the edges which carry the value zero. The generating function $q_s(v_1, \ldots, v_m)$ for a check node with state s is self explanatory by its definition. By combining Eqns(3.27, 3.28, 3.29, 3.30, 3.31), we get the desired result. □

3.B Proof of Lemma 15

Proof. First consider the binary case. We show that for every residual graph with the desired degree distribution, the probability of erasure patterns resulting the graph is the same. Note that the edge connection in the original graph is the same as in the residual graph. Consider the residual graph A. Consider two variable nodes $\mathtt{v}_1$ and $\mathtt{v}_2$ and the edges e_1 (connected to $\mathtt{v}_1$) and e_2 (connected to $\mathtt{v}_2$). We swap the end of these edges on the check node side. Call this new graph A'. Consider the case when $\mathtt{v}_1$ and $\mathtt{v}_2$ are erased in A (their state has dimension equal to 1). Clearly, the erasure pattern which results in A also results in A'. So, this is trivial. Consider the case when $\mathtt{v}_1$ and $\mathtt{v}_2$ are known in A. If initially also $\mathtt{v}_1$ and $\mathtt{v}_2$ are known then this erasure pattern also results in A' (by initially we mean the output of the channel). If the erasure pattern is such that $\mathtt{v}_1$ is known but $\mathtt{v}_2$ is erased, then for this erasure pattern we will construct another erasure pattern of equal probability for A'. We do this by making both $\mathtt{v}_1$ and $\mathtt{v}_2$ known initially but we erase another variable node which was known in the erasure pattern of A. As $\mathtt{v}_2$ is known in A and was erased initially, it was revealed at some stage of the peeling decoder. Before this stage, the peeling decoder is identical for both A and A'. When $\mathtt{v}_2$ is revealed, all the edges connected to the check node are known. So by backtracking the peeling decoder, we can find a node which was known initially. Now, we erase this variable node and still the output of the peeling decoder is the same as in the case when $\mathtt{v}_2$ is known. If initially $\mathtt{v}_1$ and $\mathtt{v}_2$ are erased, then we construct an erasure pattern where both $\mathtt{v}_1$ and $\mathtt{v}_2$ are known but we erase two other variable nodes which were known in A. By similar arguments as in the previous case, it can again be shown that peeling decoder will result in A'. The same sequence of arguments goes through in the non-binary case. □

3.C Proof of Lemma 19

Proof. Consider the case when $u_1 \in [1, \infty)$ and $u_2 \in [0, 1]$. Note that $t_{-2}(u_2)$ is non-negative for $u_2 \in [0, 1]$. Also

$$u_1 t_{+1}(u_1) = t_{+1}\left(\frac{1}{u_1}\right), \quad t_{-1}\left(\frac{1}{u_1}\right) = -u_1 t_{-1}\left(u_1\right).$$

We use these relationships in $p\left(\frac{1}{u_1}\right)$ and obtain

$$p\left(\frac{1}{u_1}\right) = \frac{u_1^{n_1}}{4}\left(t_{+1}\left(u_1\right)^{n_1} + \sum_{i=1}^{3} t_{+1}\left(u_1\right)^{n_1(i)}\left(-t_{-1}\left(u_1\right)\right)^{n_1-n_1(i)}\right).$$

As $t_{-1}\left(u_1\right)$ is non-positive for $u_1 \in [1,\infty)$, we get

$$p\left(\frac{1}{u_1}\right) \geq u_1^{n_1} p\left(u_1\right).$$

Noting that the function $h_s(u_1,u_2)$ can also be written as

$$h_s(u_1,u_2) = p(u_1)t_{-2}(u_2)^{n_2} + \frac{t_{+1}(u_1)^{n_1}}{4}\left(t_{+2}\left(u_2\right)^{n_2} - t_{-2}\left(u_2\right)^{n_2}\right).$$

Using the above relationships, we obtain

$$h_s\left(\frac{1}{u_1}, u_2\right) \geq u_1^{n_1} h_s\left(u_1,u_2\right).$$

Substituting this in the expression for $\theta\left(\frac{1}{u_1}, u_2\right)$ and using the relation that

$$(1-r)\sum_{\mathtt{r}}\sum_{s\in\mathcal{S}_{\mathtt{r}}}\Phi_{\mathtt{r}s}n_1 = \sum_{\mathtt{l}}\mathtt{l}\Omega_{\mathtt{l}1}, \tag{3.32}$$

we obtain

$$\theta\left(\frac{1}{u_1}, u_2\right) \geq \theta\left(u_1,u_2\right),$$

for $u_1 \in [1,\infty)$ and $u_2 \in [0,1]$. Note that till now we did not use the left regularity of the code. From now onwards we will assume that the code is left regular. Consider the case $u_1 \in [0,1]$ and $u_2 \in [1,\infty)$. Consider the difference

$$\theta\left(u_1,x(u_2)\right) - \theta\left(u_1,u_2\right) = \log_2\left(\frac{1+3x(u_2)^{\mathtt{l}}}{1+3u_2^{\mathtt{l}}}\right)^{\Omega_{\mathtt{l}2}} + \\ (1-r)\sum_{\mathtt{r}}\sum_{s\in\mathcal{S}_{\mathtt{r}}}\log\left(\frac{h_s(u_1,x(u_2))}{h_s(u_1,u_2)}\right).$$

We write $h_s(u_1,u_2)$ as

$$h_s(u_1,u_2) = \frac{t_{+1}(u_1)^{n_1}t_{+2}(u_2)^{n_2}}{4} + t_{-2}(u_2)^{n_2}\left(p(u_1) - \frac{t_{+1}(u_1)^{n_1}}{4}\right).$$

Noting that the code is left regular, this simplifies to

$$h_s(u_1,u_2) = \frac{t_{+1}(u_1)^{n_1}}{4}\left(\mathtt{l}\Omega_{\mathtt{l}2}\frac{1+3u_2^{\mathtt{l}-1}}{1+3u_2^{\mathtt{l}}}\right)^{n_2} + \\ \left(\mathtt{l}\Omega_{\mathtt{l}2}\frac{1-u_2^{\mathtt{l}-1}}{1+3u_2^{\mathtt{l}}}\right)^{n_2}\left(p(u_1) - \frac{t_{+1}(u_1)^{n_1}}{4}\right).$$

Note that as $u_1 \in [0,1]$ we can write $h_s(u_1,u_2)$ as

$$h_s(u_1,u_2) = c_1 \left(\frac{1+3u_2^{\mathtt{1}-1}}{1+3u_2^{\mathtt{1}}} \right)^{n_2} + c_2 \left(\frac{1-u_2^{\mathtt{1}-1}}{1+3u_2^{\mathtt{1}}} \right)^{n_2},$$

where c_1, c_2 are positive constants. Using

$$\mathtt{1}\Omega_{12} = (1-r) \sum_{\mathtt{r}} \sum_{s \in \mathcal{S}_{\mathtt{r}}} \Phi_{\mathtt{r}s} n_2, \tag{3.33}$$

we write

$$\Omega_{12} \log_2 \left(1+3u_2^{\mathtt{1}}\right) = (1-r) \sum_{\mathtt{r}} \sum_{s \in \mathcal{S}_{\mathtt{r}}} \Phi_{\mathtt{r}s} \log_2 \left(1+3u_2^{\mathtt{1}}\right)^{\frac{n_2}{\mathtt{1}}}.$$

Then,

$$\theta\left(u_1, x(u_2)\right) - \theta\left(u_1,u_2\right) = (1-r) \sum_{\mathtt{r}} \sum_{s \in \mathcal{S}_{\mathtt{r}}} \Phi_{\mathtt{r}s} \log_2 \underbrace{\left(\frac{h_s(u_1,x(u_2))\left(1+3x(u_2)^{\mathtt{1}}\right)^{\frac{n_2}{\mathtt{1}}}}{h_s(u_1,u_2)\left(1+3u_2^{\mathtt{1}}\right)^{\frac{n_2}{\mathtt{1}}}} \right)}_{A}.$$

We write the term A as

$$\frac{c_1 \left(\frac{\left(1+3x(u_2)^{\mathtt{1}-1}\right)\left(1+3x(u_2)^{\mathtt{1}}\right)^{\frac{1}{\mathtt{1}}}}{1+3x(u_2)^{\mathtt{1}}} \right)^{n_2} + c_2 \left(\frac{\left(1-x(u_2)^{\mathtt{1}-1}\right)\left(1+3x(u_2)^{\mathtt{1}}\right)^{\frac{1}{\mathtt{1}}}}{1+3x(u_2)^{\mathtt{1}}} \right)^{n_2}}{c_1 \left(\frac{\left(1+3u_2^{\mathtt{1}-1}\right)\left(1+3u_2^{\mathtt{1}}\right)^{\frac{1}{\mathtt{1}}}}{1+3u_2^{\mathtt{1}}} \right)^{n_2} + c_2 \left(\frac{\left(1-u_2^{\mathtt{1}-1}\right)\left(1+3u_2^{\mathtt{1}}\right)^{\frac{1}{\mathtt{1}}}}{1+3u_2^{\mathtt{1}}} \right)^{n_2}}.$$

We show that coefficients of c_1 and c_2 in the numerator are greater than in the denominator. We denote the ratio of coefficients of c_1 by $r_1(u_2)$. Substituting the value of $x(u_2)$ in $r_1(u_2)$ gives

$$r_1(u_2) = \frac{2\left(1+3u_2^{\mathtt{1}}\right)^{\frac{\mathtt{1}-1}{\mathtt{1}}}}{\left(3\left(1+u_2^{\mathtt{1}-1}\right)^{\frac{1}{\mathtt{1}-1}} + \left(3u_2^{\mathtt{1}-1}-1\right)^{\frac{1}{\mathtt{1}-1}}\right)^{\frac{\mathtt{1}-1}{\mathtt{1}}}}.$$

The derivative of $r_1(u_2)$ is given by

$$\frac{dr_1(u_2)}{du_2} = g(u_2)\left((3u_2-1)\left(1+u_2^{\mathtt{1}-1}\right)^{\frac{1}{\mathtt{1}-1}} - (1+u_2)\left(3u_2^{\mathtt{1}-1}-1\right)^{\frac{1}{\mathtt{1}-1}} \right),$$

where $g(u_2)$ is a positive function of u_2 for $u_2 \geq 1$. It can be easily shown that $\frac{dr_1(u_2)}{du_2}$ is positive for $u_2 \geq 1$. So $r_1(u_2)$ is an increasing function and as $r_1(1) = 1$, we prove that $r_1(u_2) \geq 1$ for $u_2 \in [1,\infty)$. Now consider the ratio of coefficients of c_2 and denote it by $r_2(u_2)$. Substituting the expression for $x(u_2)$ in $r_2(u_2)$ gives $r_1(u_2)$. Hence we prove that $\theta(u_1,x(u_2)) - \theta(u_1,u_2)$ is positive for $u_1 \in [0,1]$ and $u_2 \in [1,\infty)$. By the same arguments we can show that $\theta\left(\frac{1}{u_1}, x(u_2)\right) \geq \theta(u_1,u_2)$ for $u_1 \in [1,\infty)$ and $u_2 \in [1,\infty)$.

We now prove Eqn(3.21). By using Eqns(3.32, 3.33) and noting that $t_{+1}(1) = \mathtt{l}\Omega_{11}$ and $t_{+2}(1) = \mathtt{l}\Omega_{12}$, we write

$$\theta(u_1,u_2) =$$
$$(1-r)\sum_{\mathtt{r}}\sum_{s\in\mathcal{S}_{\mathtt{r}}}\Phi_{\mathtt{r}s}\underbrace{\log\left(\frac{h_s(u_1,u_2)\,2^{\dim(s)}}{t_{+1}(1)^{n_1}t_{+2}(1)^{n_2}}\left(\frac{1+u_1^{\mathtt{l}}}{2}\right)^{\frac{n_1}{\mathtt{l}}}\left(\frac{1+3u_2^{\mathtt{l}}}{4}\right)^{\frac{n_2}{\mathtt{l}}}\right)}_{T}.$$

If we upper bound the term T for each state s by a function of the form

$$B_1(1-u_1)^2 + B_2(1-u_1)(1-u_2) + B_3(1-u_2)^2,$$

then we prove the desired statement. This can be done in the following manner. We bound the term T corresponding to state s which has $n_2 = 0$ and $n_1 > 0$ by

$$T = \log\left(\left(\frac{1+u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}\right)^{n_1} + \left(\frac{1-u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}\right)^{n_1}\right) + \frac{n_1}{\mathtt{l}}\log\left(\frac{1+u_1^{\mathtt{l}}}{2}\right)$$
$$\leq 2n_1(\mathtt{l}-1)(1-u_1)^2.$$

To prove this consider the function $w(u_1)$

$$w(u_1) = 2n_1(\mathtt{l}-1)(1-u_1)^2 - \log\left(\left(\frac{1+u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}\right)^{n_1} + \left(\frac{1-u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}\right)^{n_1}\right)$$
$$- \frac{n_1}{\mathtt{l}}\log\left(\frac{1+u_1^{\mathtt{l}}}{2}\right).$$

The derivative of $w(u_1)$ is given by

$$\begin{aligned}
\frac{dw(u_1)}{du_1} &= -4n_1(\mathtt{l}-1)(1-u_1) - n_1(\mathtt{l}-1)u_1^{\mathtt{l}-2} \times \\
&\quad \left(\frac{(1+u_1^{\mathtt{l}-1})^{n_1-1} - (1-u_1^{\mathtt{l}-1})^{n_1-1}}{(1+u_1^{\mathtt{l}-1})^{n_1} + (1-u_1^{\mathtt{l}-1})^{n_1}} - \frac{u_1}{1+u_1^{\mathtt{l}}}\right), \\
&\leq -4n_1(\mathtt{l}-1)(1-u_1) + n_1(\mathtt{l}-1)\frac{u_1^{\mathtt{l}-1}}{1+u_1^{\mathtt{l}}}, \\
&\leq -4n_1(\mathtt{l}-1) + 3n_1(\mathtt{l}-1)u_1, \\
&\leq 0,
\end{aligned}$$

for $u_1 \in [0,1]$. As $w(1) = 0$, so $w(u_1) \geq 0$ for $u_1 \in [0,1]$. Similarly, we can bound the term T for the check node state s for which $n_1 = 0$ and $n_2 > 0$ by

$$B_3(1-u_2)^2$$

and for state s for which $n_1, n_2 > 0$ by

$$B_2(1-u_1)(1-u_2).$$

So we prove the desired result. □

4 Weight and Stopping Set Distribution

This chapter is organized in the following way. In Section 4.1 we give a motivation and summary of the results. Some notations and a brief introduction to the second moment method is given in Section 4.2. In Section 4.3 we obtain the average weight distribution and the average *binary* weight distribution of regular NBLDPC ensembles EGL $(n, \mathtt{l}, \mathtt{r}, m)$ and EGF $(n, \mathtt{l}, \mathtt{r}, m)$. In Section 4.4 we use the second moment method to prove the bound stated in Eqn(4.1) on the weight distribution. We apply the second moment method to the stopping set distribution in Section 4.5. A discussion in Section 4.6 concludes the chapter.

4.1 Motivation and Summary of Results

For a code G of block length n, we define $N(G, n\omega)$ as the weight distribution function, denoting the number of codewords with normalized weight ω[1]. The problem of computing the average weight distribution of *binary* LDPC ensembles has been considered by many researchers [27, 28, 29, 30]. We compute the average weight distribution for regular NBLDPC ensembles EGL $(\mathtt{l}, \mathtt{r}, m)$ and EGF $(\mathtt{l}, \mathtt{r}, m)$. We also derive the average binary weight distribution of these ensembles. We show that, as the alphabet size (or m) increases, the growth rate of the average binary weight distribution converges to a straight line up to a critical value of normalized weight. Beyond the critical value, it converges to the growth rate of the weight distribution corresponding to the random parity-check ensemble introduced by Gallager [6].

As we first discussed, the average weight distribution can be computed for suitably defined ensembles. However, it is hard to characterize the weight distribution of a *specific* code. In fact, even the determination of the minimum

[1] Here and in what follows it is understood that ω is such that $n\omega$ is an integer.

distance is NP-complete [31]. A possible approach to study the weight distribution of individual codes is to first compute the ensemble average, where the ensemble contains the desired codes, and then to show that most codes in the ensemble have a weight distribution close to this average.

Motivated by this approach, let us consider the exponent $\frac{1}{n}\ln N(G, n\omega)$ and define:

$$\begin{aligned} W_{\mathrm{sp}}(\omega) &:= \lim_{n\to\infty} \frac{1}{n}\mathbb{E}[\ln N(G, n\omega)], \\ W_{\mathrm{com}}(\omega) &:= \lim_{n\to\infty} \frac{1}{n}\ln \mathbb{E}[N(G, n\omega)], \end{aligned}$$

where sp stands for "statistical physics", since $W_{\mathrm{sp}}(\omega)$ can be computed by statistical physics methods and com stands for "combinatorics", as $W_{\mathrm{com}}(\omega)$ can be computed by combinatorial methods. From Jensen's inequality we know that $W_{\mathrm{sp}}(\omega) \leq W_{\mathrm{com}}(\omega)$. It has been shown in [32, 33] that for regular binary LDPC ensembles $W_{\mathrm{sp}}(\omega) = W_{\mathrm{com}}(\omega)$. However, for irregular binary LDPC ensembles this is not the case [34]. The equality between $W_{\mathrm{com}}(\omega)$ and $W_{\mathrm{sp}}(\omega)$ for regular ensembles suggests that a randomly chosen code should have $N(G, n\omega)$ "close" to $\mathbb{E}[N(G, n\omega)]$ with high probability. In other words $N(G, n\omega)$ should be concentrated around the ensemble average. The question of the concentration of the weight distribution has not been addressed before. We partially answer this question for regular LDPC ensembles.

Let us give a short overview of our main results. Using the second moment method, we obtain an asymptotic lower bound on the probability that a randomly chosen code from binary regular LDPC ensemble has its weight distribution close to the ensemble average. Under some technical conditions (see Lemma 32), we show that for a binary regular LDPC ensemble with left degree $\mathtt{l}$ and right degree $\mathtt{r}$, any $\epsilon > 0$, and for all ω such that $W_{\mathrm{com}}(\omega)$ is positive,

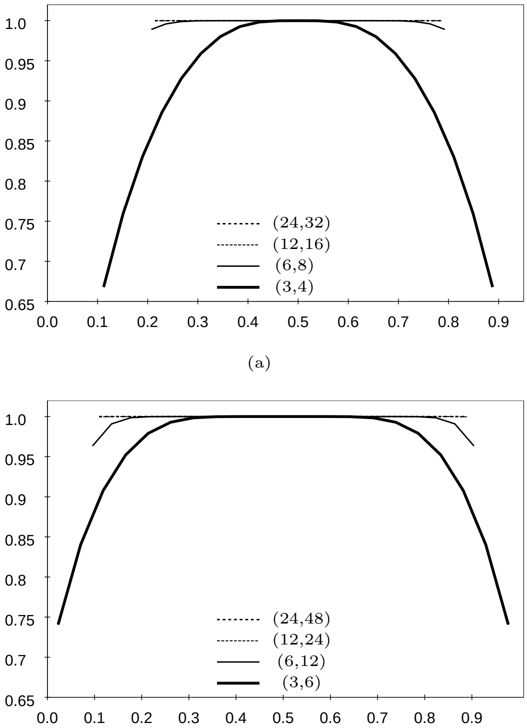
$$\lim_{n\to\infty} \mathrm{P}\left(1-\epsilon \leq \frac{N(G, n\omega)}{\mathbb{E}[N(G, n\omega)]} \leq 1+\epsilon\right) \geq 1 - \frac{\delta(\omega, \mathtt{l}, \mathtt{r})}{\epsilon^2}, \tag{4.1}$$

where $\delta(\omega, \mathtt{l}, \mathtt{r})$ is a function of ω and can be evaluated by solving a polynomial equation. If $W_{\mathrm{com}}(\omega) \leq 0$, then $\mathbb{E}[N(G, n\omega)] = o(1)$ (Lemma 27) and by Markov's inequality we have

$$\lim_{n\to\infty} \mathrm{P}\left(N(G, n\omega) = 0\right) = 1.$$

Clearly, for ω such that $W_{\mathrm{com}}(\omega) \leq 0$, the convergence of $N(G, n\omega)$ to its average follows trivially. Hence our focus will be on the weight ω such that $W_{\mathrm{com}}(\omega) > 0$.

Note that the convergence implied by the bound in Eqn(4.1) is pointwise, for a fixed ω, Eqn(4.1) implies that asymptotically at least a fraction $1 - \frac{\delta(\omega, \mathtt{l}, \mathtt{r})}{\epsilon^2}$ of codes in the ensemble have their weight distribution function in a window of width ϵ around the ensemble average. In Fig. 4.1 we plot the bound in Eqn(4.1) for regular codes with $\frac{\mathtt{l}}{\mathtt{r}} = 0.75$ and 0.5. We observe that if we fix the ratio $\frac{\mathtt{l}}{\mathtt{r}}$ and let $\mathtt{l}, \mathtt{r}$ increase then the bound converges to 1. This implies that for large left and right degrees, almost all the codes in the ensemble have their weight distribution very close to the ensemble average. Note that in this case it is well known that the growth rate of the average weight distribution converges to that of Gallager's random parity check ensemble [35].



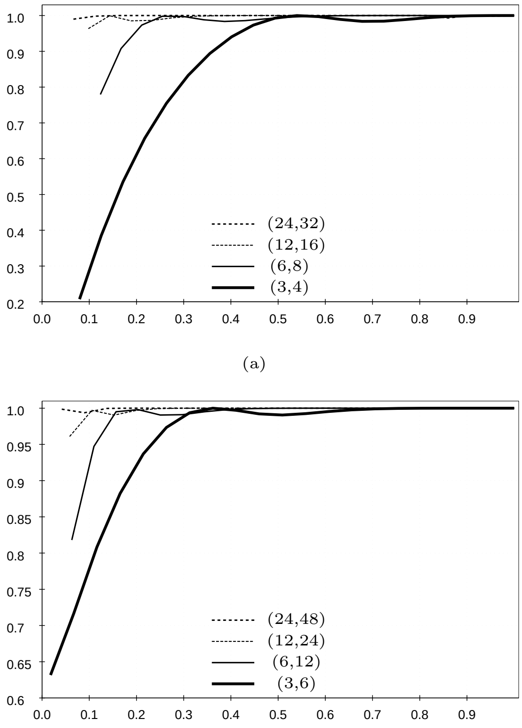
(a)

(b)

Figure 4.1: The x-axis is the relative weight ω such that $W_{\mathsf{com}}(\omega) > 0$ and y-axis is the bound $1 - \frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2}$ with $\epsilon = 0.95$, (a) for ensembles with rate=0.25, (b) for ensembles with rate=0.5.

Another important quantity of binary LDPC codes is the stopping set distribution. Stopping sets determine the performance of LDPC codes under iterative decoding over the BEC. The bound obtained in Eqn(4.1) can be easily extended to the stopping set distribution. This is because the method of determining the moments in both cases is the same. In Fig. 4.2 we plot the bound in Eqn(4.1) for the stopping set distribution. Again, we observe that if we fix the ratio $\frac{\mathtt{l}}{\mathtt{r}}$ and let $\mathtt{l}, \mathtt{r}$ increase then the bound converges to 1.

Independently, authors in [36] have computed the second moment of the weight distribution for binary regular LDPC ensembles. Based on the com-



(a)

(b)

Figure 4.2: The x-axis is the relative stopping set size s and y-axis is the bound $1 - \frac{\delta(s,\mathtt{l},\mathtt{r})}{\epsilon^2}$ with $\epsilon = 0.95$, (a) for ensembles with rate=0.25, (b) for ensembles with rate=0.5.

putations of the second moment of weight and stopping set distribution, the authors in [37, 38] have obtained probabilistic lower and upper bounds on the error exponent of binary regular LDPC ensembles.

4.2 Preliminaries

4.2.1 Definitions and Notations

Consider the NBLDPC ensemble EGL$(n, \mathtt{l}, \mathtt{r}, m)$ defined in Chapter 2. Let G be a code chosen at random from EGL$(n, \mathtt{l}, \mathtt{r}, m)$. Let $N(G, n\omega)$ be the weight distribution function denoting the number of codewords of weight $n\omega$ in G, where $\omega \in (0, 1)$ is the normalized weight. The binary weight distribution $N_b(G, nm\omega)$ of an NBLDPC code G is the number of codewords with binary weight $nm\omega$. Note that we have normalization with respect to nm as this is the binary length of the NBLDPC ensemble EGL$(n, \mathtt{l}, \mathtt{r}, m)$. We use the same notations for the ensemble EGF $(n, \mathtt{l}, \mathtt{r}, m)$. Let $\sigma^2(G, n\omega)$ denote the variance of $N(G, n\omega)$ over the binary ensemble EGL$(n, \mathtt{l}, \mathtt{r}, 1)$, $\sigma^2(G, n\omega) = \mathbb{E}[N(G, n\omega)^2] - \mathbb{E}[N(G, n\omega)]^2$.

The *support set* of a word is the set of its non zero bits. The *overlap* between two words is the intersection of their support sets.

A stopping set of a Tanner graph is a subset of the set of variable nodes in its such that its neighboring check nodes are connected to it at least twice. Consider the binary LDPC ensemble EGL $(n, \mathtt{l}, \mathtt{r}, 1)$. Let $S(G, ns)$ denote the number of stopping sets of size ns and let $\sigma^2(G, ns)$ denote the variance of $S(G, ns)$ over the ensemble EGL$(n, \mathtt{l}, \mathtt{r}, 1)$.

We denote a vector $(x_1, \ldots, x_k)$ by $\underline{x}$, the transpose of $\underline{x}$ by $\underline{x}^T$, the dot product between $\underline{x}$ and $\underline{y}$ is denoted by $\underline{x}.\underline{y}^T$, $\underline{xy}$ denotes the component wise multiplication, i.e., the vector $(x_1y_1, \ldots, x_ky_k)$. We use the notation that a vector to the power of a vector and also a scalar to the power of a vector is a vector i.e. , $\underline{x}^{\underline{i}} := (x_1^{i_1}, \ldots, x_k^{i_k})$ and $e^{\underline{x}} := (e^{x_1}, \ldots, e^{x_k})$. Finally, $x^+ := \max(x, 0)$ and $f'(t)$ denotes the derivative of the function $f(x)$ evaluated at t.

4.2.2 Second Moment Method

Let $\{X_n\}$ be a sequence of random variables indexed by n, $n \in \mathbb{N}$. Let $\sigma_n^2 = \mathbb{E}[(X_n - \mathbb{E}[X_n])^2]$ be the variance of X_n. Then by Chebyshev's inequality we have for any $a \geq 0$,

$$\mathrm{P}(|X_n - \mathbb{E}[X_n]| \geq a) \leq \frac{\sigma_n^2}{a^2}.$$

Let $\lim_{n\to\infty} \frac{\sigma_n^2}{\mathbb{E}[X_n]^2} = \delta$. If we choose $a = \epsilon\mathbb{E}[X_n]$ and ϵ such that $\epsilon^2 > \delta$, then we can draw the conclusion that

$$\lim_{n\to\infty} \mathrm{P}\left(1 - \epsilon \leq \frac{X_n}{\mathbb{E}[X_n]} \leq 1 + \epsilon\right) \geq 1 - \frac{\delta}{\epsilon^2}.$$

In order to apply this bound to $N(G, n\omega)$, we need to compute the ratio

$$\lim_{n\to\infty} \frac{\sigma^2(G, n\omega)}{\mathbb{E}[N(G, n\omega)]^2} = \lim_{n\to\infty} \frac{\mathbb{E}[N^2(G, n\omega)]}{\mathbb{E}[N(G, n\omega)]^2} - 1.$$

In the following section we derive the average weight distribution and the average binary weight distribution of regular NBLDPC ensemble.

4.3 Expectation of Weight Distribution

In the following lemma we derive the average weight distribution of the NBLDPC ensemble EGL $(n, \mathtt{l}, \mathtt{r}, m)$.

Lemma 21 (Average Weight Distribution of EGL $(n, \mathtt{l}, \mathtt{r}, m)$)**.** *Consider the ensemble EGL$(n, \mathtt{l}, \mathtt{r}, m)$. Its average weight distribution is given by*

$$\mathbb{E}[N(G, n\omega)] = \frac{\binom{n}{n\omega}(2^m - 1)^{n\omega}}{\binom{n\mathtt{l}}{n\mathtt{l}\omega}|\,\mathrm{GL}_2^m\,|^{n\mathtt{l}\omega}} coef\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}}, x^{n\mathtt{l}\omega}\right), \tag{4.2}$$

where

$$p_m(x) = \frac{1}{2^m}\left[(1 + |\,\mathrm{GL}_2^m\,| x)^r + \left(1 - \frac{x}{2^m - 1}\right)^r (2^m - 1)\right].$$

Proof. The number of codewords of weight $n\omega$ is given by

$$N(G, n\omega) = \sum_{w \in \mathcal{W}} \mathbb{1}_w(G),$$

where $\mathcal{W}$ is the set of all the words of weight $n\omega$ over GF_2^m and

$$\mathbb{1}_w(G) = \begin{cases} 1, & \text{if } w \text{ is a codeword of } G, \\ 0, & \text{otherwise.} \end{cases}$$

Now the expectation is given by

$$\mathbb{E}\left(N(G, n\omega)\right) = \sum_{w \in \mathcal{W}} \mathbb{E}\left(\mathbb{1}_w(G)\right). \tag{4.3}$$

Because of the symmetry in the permutation of edges and due to uniform probability of all the possible edge labels on every edge, $\mathbb{1}_w(G)$ is independent of the word w and depends only on its weight. Thus we fix w to a word with support set $\{1, \ldots, n\omega\}$ and its non-zero symbols are unity. Then Eqn(4.3) reduces to

$$\mathbb{E}\left(N(G, n\omega)\right) = \binom{n}{n\omega}(2^m - 1)^{n\omega}\,\mathbb{E}\left(\mathbb{1}_w(G)\right). \tag{4.4}$$

Now,

$$\mathbb{E}\left(\mathbb{1}_w(G)\right) = \frac{\text{number of graphs for which } w \text{ is a codeword}}{\text{total number of graphs}}.$$

The total number of graphs is given by $(n\mathtt{l})!|\,\mathrm{GL}_2^m\,|^{n\mathtt{l}}$. The number of graphs for which w is a codeword is given by

$$(n\mathtt{l} - n\mathtt{l}\omega)!(n\mathtt{l}\omega)!|\,\mathrm{GL}_2^m\,|^{n\mathtt{l} - n\mathtt{l}\omega}\mathrm{coef}\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}}, x^{n\mathtt{l}\omega}\right).$$

The factorial terms correspond to permuting the edges carrying non-zero values and zero value. The term $|\,\mathrm{GL}_2^m\,|^{n\mathtt{l} - n\mathtt{l}\omega}$ takes care of the fact that we can put any edge label on the edges carrying the value zero. To describing the polynomial $p_m(x)$, we recall the definition of F_i defined in the Chapter 3, Lemma 14.

$$\begin{aligned} F_{\mathtt{r}} &= \left|\{(M_1 \ldots, M_{\mathtt{r}}) : \sum_{i=1}^{\mathtt{r}} M_i x_i = 0, M_i \in \mathrm{GL}_2^m, x_i \in \mathrm{GF}_2^m\}\right|, \\ &= \frac{|\,\mathrm{GL}_2^m\,|^{\mathtt{r}}}{2^m}\left(1 + \frac{(-1)^{\mathtt{r}}}{(2^m - 1)^{\mathtt{r}-1}}\right). \end{aligned}$$

Then the polynomial $p_m(x)$ is defined by

$$\begin{aligned} p_m(x) &= 1+\sum_{i=1}^{\mathtt{r}}\binom{\mathtt{r}}{i}F_i x^i, \\ &= 1+\sum_{i=1}^{\mathtt{r}}\binom{\mathtt{r}}{i}\frac{|\,\mathrm{GL}_2^m\,|^i}{2^m}\left(1+\frac{(-1)^i}{(2^m-1)^{i-1}}\right)x^i, \\ &= \frac{1}{2^m}\left[(1+|\,\mathrm{GL}_2^m\,|x)^{\mathtt{r}}+\left(1-\frac{x}{2^m-1}\right)^{\mathtt{r}}(2^m-1)\right]. \end{aligned} \tag{4.5}$$

In summary,

$$\mathbb{E}\left(\mathbb{1}_w(G)\right)=\frac{\mathrm{coef}\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}},x^{n\mathtt{l}\omega}\right)}{\binom{n\mathtt{l}}{n\mathtt{l}\omega}|\,\mathrm{GL}_2^m\,|^{n\mathtt{l}\omega}}. \tag{4.6}$$

Substituting Eqn(4.6) in the expression for $\mathbb{E}\left(N(G,n\omega)\right)$ in Eqn(4.4) gives the desired result. □

Using similar arguments we obtain the average weight distribution for the ensemble EGF $(n,\mathtt{l},\mathtt{r},m)$ in the stated lemma.

Lemma 22 (Average Weight Distribution of EGF $(n,\mathtt{l},\mathtt{r},m)$)**.** *Consider the ensemble EGF$(n,\ \mathtt{l},\mathtt{r},m)$. Its average weight distribution is given by*

$$\mathbb{E}[N(G,n\omega)]=\frac{\binom{n}{n\omega}}{\binom{n\mathtt{l}}{n\mathtt{l}\omega}(2^m-1)^{n(\mathtt{l}-1)\omega}}\,\mathit{coef}\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}},x^{n\mathtt{l}\omega}\right), \tag{4.7}$$

where

$$p_m(x)=\frac{1}{2^m}\left[(1+(2^m-1)x)^r+(1-x)^r(2^m-1)\right]. \tag{4.8}$$

Note: The average weight distribution of regular non binary coset LDPC ensemble has been derived in [39]. In [40] authors have obtained average weight distribution of regular LDPC ensemble defined over the cyclic group of integers where group operation is performed modulo an integer.

As we consider transmission over BMS channels, the binary weight distribution is important in upper bounding the MAP block error probability as given in [41]. In the next lemma we derive the average binary weight distribution.

Lemma 23 (Average Binary Weight Distribution of EGL $(n,\mathtt{l},\mathtt{r},m)$)**.** *Consider the ensemble EGL $(n,\ \mathtt{l},\mathtt{r},m)$. Let $N_b\left(G,nm\omega\right)$ denote the binary weight distribution of a randomly chosen code G. Then the average of $N_b\left(G,nm\omega\right)$ is given by*

$$\mathbb{E}\left(N_b\left(G,nm\omega\right)\right)= \\ \sum_{i=n\omega}^{min(n,nm\omega)}\frac{\binom{n}{i}\mathit{coef}\left(\left((1+x)^m-1\right)^i,x^{nm\omega}\right)\mathit{coef}\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}},x^{\mathtt{l}i}\right)}{\binom{n\mathtt{l}}{i\mathtt{l}}|\,\mathrm{GL}_2^m\,|^{\mathtt{l}i}}, \tag{4.9}$$

where $\omega\in(0,1)$.

Proof. The average binary weight distribution is given by,

$$\mathbb{E}\left(N_b\left(G, nm\omega\right)\right) = \sum_{i=n\omega}^{\min(n,nm\omega)} g(i, nm\omega)\, \mathrm{P}\left(\text{A given weight } i \text{ word is codeword of G}\right), \quad (4.10)$$

where $g(i, nm\omega)$ denotes the number of words with weight i whose binary weight is $nm\omega$ and is given by

$$g(i, nm\omega) = \binom{n}{i} \mathrm{coef}\left(\left((1+x)^m - 1\right)^i, x^{nm\omega}\right).$$

Combining Eqn(4.6) and the expression for $g(i, nm\omega)$ gives the desired result. □

Using similar arguments, we obtain the average binary weight distribution of the ensemble EGF $(n, \mathtt{l}, \mathtt{r}, m)$ in the following lemma.

Lemma 24 (Average Binary Weight Distribution of EGF $(n, \mathtt{l}, \mathtt{r}, m)$)**.** *Consider the ensemble EGF $(n, \mathtt{l}, \mathtt{r}, m)$. Let $N_b\left(G, nm\omega\right)$ denotes the binary weight distribution of a randomly chosen code G. Then the average of $N_b\left(G, nm\omega\right)$ is given by*

$$\mathbb{E}\left(N_b\left(G, nm\omega\right)\right) = \sum_{i=n\omega}^{min(n,nm\omega)} \frac{\binom{n}{i} coef\left(\left((1+x)^m - 1\right)^i, x^{nm\omega}\right) coef\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}}, x^{\mathtt{l}i}\right)}{\binom{n\mathtt{l}}{i\mathtt{l}} \left(2^m - 1\right)^{\mathtt{l}i}}, \quad (4.11)$$

where $\omega \in (0,1)$ and $p_m(x)$ is defined in Eqn(4.8).

In the next lemma, we recall the Hayman method for approximating the term coef$(q(y)^n, y^{n\omega})$ for large values of n, where $q(y)$ is a finite degree polynomial which satisfies appropriate technical conditions. The Hayman method is based on the saddle point approximation. More details and its proof can be found in [42, 43].

Lemma 25 (Hayman Method)**.** *Let $q(y) = \sum_i q_i y^i$ be a polynomial with non negative coefficients such that $q_0 \neq 0$ and $q_1 \neq 0$. Define $a_q(y) := y\frac{dq(y)}{dy}\frac{1}{q(y)}$ and $b_q(y) := y\frac{da_q(y)}{dy}$. Then for n tending to infinity so that $n\omega \in \mathbb{N}$*

$$\mathrm{coef}(q(y)^n, y^{n\omega}) = \frac{q(y_\omega)^n}{y_\omega^{n\omega}\sqrt{2\pi n b_q(y_\omega)}}(1 + o(1)), \quad (4.12)$$

where the term $o(1)$ converges to zero and y_ω is the unique positive solution of $a_q(y) = \omega$.

Note that $p_m(x)$ satisfies the technical conditions of the Lemma 25 for $m \geq 2$. We give the estimate of the average weight distribution for $m = 2$ in the following lemma and after that we discuss the case for $m = 1$.

Lemma 26 (Ensemble Average of Weight Distribution for $m \geq 2$)**.** *Consider the regular ensemble EGL* $(n, \mathtt{l}, \mathtt{r}, m)$*, where* $m \geq 2$*. The average weight distribution is given by*

$$\mathbb{E}\,(N(G, n\omega)) = \\ \frac{\sqrt{\mathtt{r}}}{\sqrt{2\pi n b_{p_m}(x_\omega)}} e^{n\left(\omega \ln(2^m-1) - h(\omega)(\mathtt{l}-1) + \frac{\mathtt{l}}{r}\ln(p_m(x_\omega)) - \mathtt{l}\omega\ln(|\,\mathrm{GL}_2^m\,|) - \mathtt{l}\omega\ln(x_\omega)\right)}, \tag{4.13}$$

where x_ω *is the solution of* $a_{p_m}(x) = \mathtt{r}\omega$ *and* $h(\omega) = -(\omega\ln\omega + (1-\omega)\ln(1-\omega))$ *and* $\ln\omega$ *is the natural logarithm of* ω*.*

Proof. Using Stirling's approximation we get:

$$\binom{n}{n\omega} = \frac{e^{nh(\omega)}}{\sqrt{2\pi n\omega(1-\omega)}}(1 + o(1)). \tag{4.14}$$

Using this, Lemma 21 and Lemma 25, we get the desired result. □

For the case $m = 1$ the polynomial $p_1(x)$ contains only even powers of x. To circumvent this we can define $q(y) = p_1(x)$ and $y = x^2$. Then the polynomial $q(y)$ satisfies the technical conditions of the Lemma 25. Since $q(y) = p_1(x)$ and $y = x^2$, we have $a_q(y) = a_{p_1}(x)/2$, where $a_{p_1}(x) = x\frac{dp_1(x)}{dx}\frac{1}{p_1(x)}$. Similarly, $b_q(y) = b_{p_1}(x)/4$, where $b_{p_1}(x) = x\frac{da_{p_1}(x)}{dx}$. Also, $y_\omega = x_\omega^2$, where x_ω is the unique positive solution of $a_{p_1}(x) = r\omega$ which simplifies to,

$$x\frac{(1+x)^{\mathtt{r}-1} - (1-x)^{\mathtt{r}-1}}{(1+x)^{\mathtt{r}} + (1-x)^{\mathtt{r}}} = \omega. \tag{4.15}$$

Thus by substituting these relationships in Lemma 25, we get

$$\operatorname{coef}\left(p(x)^{\frac{n\mathtt{l}}{\mathtt{r}}}, x^{n\mathtt{l}\omega}\right) = \frac{2p(x_\omega)^{\frac{n\mathtt{l}}{\mathtt{r}}}}{(x_\omega)^{n\mathtt{l}\omega}\sqrt{2\pi\frac{n\mathtt{l}}{\mathtt{r}}b_p(x_\omega)}}(1 + o(1)). \tag{4.16}$$

We summarize our results thus far.

Lemma 27 (Ensemble Average of Weight Distribution for m=1)**.** *Consider the regular LDPC ensemble* $\mathbb{G}(n, \mathtt{l}, \mathtt{r})$*. Then for* $\omega \in (0, 1)$ *such that* $\mathtt{l}n\omega \in 2\mathbb{N}$*,*

$$\mathbb{E}[N(G, n\omega)] = \frac{2\sqrt{\mathtt{r}}e^{n(\frac{\mathtt{l}}{r}\ln(p(x_\omega)) - (\mathtt{l}-1)h(\omega) - \mathtt{l}\omega\ln(x_\omega))}}{\sqrt{2\pi n b_p(x_\omega)}}(1 + o(1)), \tag{4.17}$$

where $h(\omega) = -(\omega\ln\omega + (1-\omega)\ln(1-\omega))$, $\ln\omega$ *is the natural logarithm of* ω *and* x_ω *is the unique positive solution of equation (4.15). If* $n\,\mathtt{l}\omega$ *is odd, then* $\mathbb{E}[N(G, n\omega)] = 0$.

It will be of interest to characterize the growth rate of the binary weight distribution of EGL $(n, \mathtt{l}, \mathtt{r}, m)$ as m tends to infinity. In the following lemma we show that the growth rate of the binary weight distribution converges to the growth rate of Gallager's random parity-check ensemble.

Lemma 28 (Convergence of EGL $(\mathtt{l}, \mathtt{r}, m)$)**.** *Consider the ensemble EGL (n, l, r, m). Let $N_b\left(G, nm\omega\right)$ denote the binary weight distribution of a randomly chosen code G. Then*

$$\lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(\mathbb{E}\left(N_b(G, nm\omega)\right)\right) = \begin{cases} -\frac{\omega}{\log(2)}\log(2^{l/r}-1), & 0 \le \omega < 1 - 2^{-l/r}, \\ h(\omega) - \frac{l}{r}\log(2), & 1 - 2^{-l/r} \le \omega \le 1. \end{cases}$$

Proof. By Lemma 23, the binary weight distribution is given by

$$\mathbb{E}\left(N_b\left(G, nm\omega\right)\right) = \sum_{i=n\omega}^{\min(n,nm\omega)} \frac{\binom{n}{i}\mathrm{coef}\left(\left((1+x)^m-1\right)^i, x^{nm\omega}\right)\mathrm{coef}\left(p_m(x)^{\frac{n\mathtt{l}}{\mathtt{r}}}, x^{\mathtt{l}i}\right)}{\binom{n\mathtt{l}}{i\mathtt{l}}|\,\mathrm{GL}_2^m\,|^{\mathtt{l}i}}.$$

Let us denote the summation term corresponding to index i by S_i. We normalize i by n and write $i = \gamma n$.

$$\lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(\mathbb{E}\left(N_b\left(G, nm\omega\right)\right)\right) = \max\nolimits_{\gamma\in[\omega,1]}\lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(S_{\gamma n}\right),$$

where we used the fact that for m large enough $\min\left(n, nm\omega\right) = n$. Using Stirling's approximation given in Eqn(4.14) and Hayman method given in Lemma 25, we obtain

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{n}\log\left(S_{\gamma n}\right) &= -(\mathtt{l}-1)h\left(\gamma\right) - \gamma\mathtt{l}\log\left(\mathrm{GL}_2^m\right) + \frac{\mathtt{l}}{\mathtt{r}}\log\left(p_m\left(y_m\right)\right) \\ &\quad -\mathtt{l}\gamma\log\left(y_m\right) + \gamma\log\left(f\left(x_m\right)\right) - \left(\omega m - \gamma\right)\log\left(x_m\right).\end{aligned} \tag{4.18}$$

In order to satisfy the conditions of Lemma 25 we notice that

$$\mathrm{coef}\left(\left((1+x)^m-1\right)^{\gamma n}, x^{nm\omega}\right) = \mathrm{coef}\left(\left(\frac{(1+x)^m-1}{x}\right)^{\gamma n}, x^{nm\omega-\gamma n}\right).$$

The x_m appearing in Eqn(4.18) is the solution of the equation

$$\frac{x(1+x)^{m-1}}{(1+x)^m-1} = \frac{\omega}{\gamma}.$$

Note that for large values of m, $(1+x)^m - 1 \approx (1+x)^m$. Using this we get that for m large enough

$$x_m = \frac{\omega}{\gamma-\omega}. \tag{4.19}$$

The y_m appearing in Eqn(4.18) is solution of the equation

$$y\frac{\mathtt{r}\left[(1+|\,\mathrm{GL}_2^m\,|y)^{\mathtt{r}-1}\,|\,\mathrm{GL}_2^m\,| - \left(1-\frac{y}{2^m-1}\right)^{\mathtt{r}-1}\right]}{(1+|\,\mathrm{GL}_2^m\,|y)^{\mathtt{r}} + \left(1-\frac{y}{2^m-1}\right)^{\mathtt{r}}(2^m-1)} = \gamma.$$

Note that for large values of m, $1+|\,\mathrm{GL}_2^m\,|y \gg 1-\frac{y}{2^m-1}$. This gives

$$y_m = \frac{\gamma}{1-\gamma}\frac{1}{\mathrm{GL}_2^m}. \tag{4.20}$$

Substituting for x_m, y_m from Eqn(4.19, 4.20) in Eqn(4.18) gives for $\gamma \in [\omega, 1]$

$$f(\gamma) \triangleq \lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(S_{\gamma n}\right) = \gamma\log\left(\frac{\gamma}{\gamma-\omega}\right) - \omega\log\left(\frac{\omega}{\gamma-\omega}\right) - \frac{\mathtt{l}}{\mathtt{r}}\gamma\log(2).$$

We now compute the derivative of $f(\gamma)$ in order to compute its maximum. The derivative is given by

$$\frac{df\left(\gamma\right)}{d\gamma} = \log\left(\frac{\gamma}{\gamma-\omega}\right) - \frac{\mathtt{l}}{\mathtt{r}}\log(2),$$

and becomes zero for

$$\gamma^* = \frac{\omega}{1-2^{-\mathtt{l}/\mathtt{r}}}.$$

The second derivative is given by

$$\frac{d^2f(\gamma)}{d\gamma^2} = \frac{-\omega}{\gamma(\gamma-\omega)\log_e(2)}.$$

So, γ^* is indeed a maximum provided $\gamma^* < 1$. This means that for $\omega \le 1-2^{-\mathtt{l}/\mathtt{r}}$, γ^* is maximum. Otherwise, the maximum is achieved at $\gamma = 1$. Substituting these values, we obtain

$$\lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(\mathbb{E}\left(N_b(G, nm\omega)\right)\right) = \begin{cases} -\frac{\omega}{\log(2)}\log(2^{\mathtt{l}/\mathtt{r}}-1), & 0 \le \omega < 1-2^{-\mathtt{l}/\mathtt{r}}, \\ h(\omega) - \frac{\mathtt{l}}{\mathtt{r}}\log(2), & 1-2^{-\mathtt{l}/\mathtt{r}} \le \omega \le 1. \end{cases}$$

This proves the lemma. □

The result of Lemma 28 holds also for the ensemble EGF $(\mathtt{l}, \mathtt{r}, m)$ with almost the same proof. We state this in the following lemma.

Lemma 29 (Convergence of EGF $(\mathtt{l}, \mathtt{r}, m)$)**.** *Consider the ensemble EGF(n, l, r, m). Let $N_b\left(G, nm\omega\right)$ denote the binary weight distribution of a randomly chosen code G. Then*

$$\lim_{m\to\infty}\lim_{n\to\infty}\frac{1}{nm}\log\left(\mathbb{E}\left(N_b(G, nm\omega)\right)\right) = \begin{cases} -\frac{\omega}{\log(2)}\log(2^{l/r}-1), & 0 \le \omega < 1-2^{-l/r}, \\ h(\omega) - \frac{l}{r}\log(2), & 1-2^{-l/r} \le \omega \le 1. \end{cases}$$

4.4 Second Moment of Weight Distribution

In this section we restrict ourselves to binary regular LDPC ensembles and compute the second moment of their weight distribution function. To compute the second moment, we note that $\mathbb{E}[N^2(G, n\omega)] = \mathbb{E}[\sum_{w,w'} \mathbb{1}_{w,w'}(G, n\omega)]$, where w, w' are both words of length n and weight $n\omega$ and

$$\mathbb{1}_{w,w'}(G, n\omega) = \begin{cases} 1, & \text{if } w, w' \text{ are codewords of } G, \\ 0, & \text{otherwise.} \end{cases}$$

By definition of the ensemble, the expectation $\mathbb{E}[\mathbb{1}_{w,w'}(G, n\omega)]$ does not depend on the specific choice of the pair w, w' but only on the cardinality of the overlap between the support sets of w and w'. In particular we can fix w to be a codeword of weight $n\omega$ with support set $\mathcal{W} = \{1, 2, \ldots, n\omega\}$, so that

$$\mathbb{E}[N^2(G, n\omega)] = \binom{n}{n\omega} \sum_{w'} \mathbb{E}[\mathbb{1}_{w'}(G, n\omega)],$$

where we have dropped the subscript w as w is fixed. We can also fix w' to $w'(i)$ for a given cardinality of overlap i with w. $w'(i)$ has support set $\mathcal{W}' = \{1, 2, \ldots, i, n\omega + 1, \ldots, 2n\omega - i\}$. Then,

$$\mathbb{E}[N^2(G, n\omega)] = \binom{n}{n\omega} \sum_{i=0}^{n\omega} \binom{n\omega}{i} \binom{n - n\omega}{n\omega - i} \mathbb{E}[\mathbb{1}_{w'(i)}(G, n\omega)].$$

The binomials inside the summation correspond to the number of words having cardinality of overlap with w equals to i. To calculate $\mathbb{E}[\mathbb{1}_{w'(i)}(G, n\omega)]$, we note that there are 3 different types of edges taking value 1. These types are: edges connected to $\mathcal{W} \cap \mathcal{W}'$, edges connected to $\mathcal{W} \backslash (\mathcal{W} \cap \mathcal{W}')$ and finally, edges connected to $\mathcal{W}' \backslash (\mathcal{W} \cap \mathcal{W}')$. A placement of edges is *valid* if each check node is connected to an even number of edges from $\mathcal{W}$ as well as from $\mathcal{W}'$, i.e., if the number of edges from each of the 3 different classes are *all* even or *all* odd. A moment's thought shows that the generating function for the number of valid placement is given by $f(x_1, x_2, x_3)^{\frac{n\mathtt{l}}{\mathtt{r}}} = f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}$, where x_1 corresponds to the number of edges connected to $\mathcal{W} \backslash (\mathcal{W} \cap \mathcal{W}')$, x_2 corresponds to the number of edges connected to $\mathcal{W} \cap \mathcal{W}'$ and x_3 corresponds to the number of edges connected to $\mathcal{W}' \backslash (\mathcal{W} \cap \mathcal{W}')$, and where $f(\underline{x})$ is the summation of the terms in the expansion of $(1 + x_1 + x_2 + x_3)^{\mathtt{r}}$ which have powers of x_1, x_2 and x_3 either all even or all odd. Explicitly,

$$\begin{aligned} f(\underline{x}) \quad = \quad & \frac{1}{4}\Big((1 + x_1 + x_2 + x_3)^{\mathtt{r}} + (1 + x_1 - x_2 - x_3)^{\mathtt{r}} \\ & + (1 - x_1 + x_2 - x_3)^{\mathtt{r}} + (1 - x_1 - x_2 + x_3)^{\mathtt{r}}\Big). \end{aligned} \tag{4.21}$$

Since there are $\mathtt{l}(n\omega - i)$ edges connected to $\mathcal{W}\backslash(\mathcal{W} \cap \mathcal{W}')$, $\mathtt{l}i$ edges connected to $\mathcal{W} \cap \mathcal{W}'$ and $\mathtt{l}(n\omega - i)$ edges connected to $\mathcal{W}'\backslash(\mathcal{W} \cap \mathcal{W}')$, we have

$$\begin{aligned}\mathbb{E}[\mathbb{1}_{w'(i)}(G, n\omega)] &= \operatorname{coef}\left(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, x_1^{\mathtt{l}(n\omega-i)} x_2^{\mathtt{l}i} x_3^{\mathtt{l}(n\omega-i)}\right) \\ &\quad \cdot \frac{1}{(n\mathtt{l})!}((\mathtt{l}(n\omega - i))!)^2 (\mathtt{l}i)!(n\mathtt{l} - 2n\mathtt{l}\omega + \mathtt{l}i)!.\end{aligned}$$

As all the edges are labeled, the factor $(n\mathtt{l})!$ corresponds to the total number of graphs in the ensemble $\mathbb{G}(n, \mathtt{l}, \mathtt{r})$. The term $(\mathtt{l}(n\omega - i))!^2$ corresponds to interchanging the positions of edges connected to $\mathcal{W}\backslash(\mathcal{W} \cap \mathcal{W}')$, as well as to $\mathcal{W}'\backslash(\mathcal{W} \cap \mathcal{W}')$, $(\mathtt{l}i)!$ corresponds to interchanging the positions of edges connected to $\mathcal{W} \cap \mathcal{W}'$, and $(\mathtt{l}(n - 2n\omega + i))!$ corresponds to interchanging of the positions of edges taking value 0. Hence,

$$\mathbb{E}[N^2(G, n\omega)] = \sum_{i=0}^{n\omega} \underbrace{\operatorname{coef}\left(f(\underline{x})^{n\frac{\mathtt{l}}{\mathtt{r}}}, x_1^{\mathtt{l}(n\omega-i)} x_2^{\mathtt{l}i} x_3^{\mathtt{l}(n\omega-i)}\right)}_{C_i}$$

$$\underbrace{\cdot \frac{\binom{n}{n\omega}}{(n\mathtt{l})!} \binom{n\omega}{i} \binom{n - n\omega}{n\omega - i} ((\mathtt{l}(n\omega - i))!)^2 (\mathtt{l}i)!(\mathtt{l}(n - 2n\omega + i))!}_{F_i}.$$

Let S_i be the i^{th} summation term in (4.22), so $S_i = F_i C_i$. Note that $S_i = 0$ for $i < (2n\omega - n)^+$ as there can not exist two words of length n and weight $n\omega$ such that the cardinality of their overlap is less than $(2n\omega - n)^+$. A property of the term $S_{n\omega}$ that we will need later is

$$S_{n\omega} = \mathbb{E}[N(G, n\omega)]. \tag{4.22}$$

This simply follows from the fact that for $i = n\omega$, the words w and $w'(i)$ are identical. Now to get a closed form expression for $\mathbb{E}[N^2(G, n\omega)]$, we use Stirling's formula to approximate the factorial terms and to approximate the coef function we use the following multidimensional extension of Lemma 25 as given in Theorem 2 of [26].

Lemma 30. *[Multidimensional Saddle Point Method] Let us denote* $\underline{i} := (\mathtt{l}(n\omega - i), \mathtt{l}i, \mathtt{l}(n\omega - i))$, $\underline{j} := (\mathtt{l}(n\omega - j), \mathtt{l}j, \mathtt{l}(n\omega - j))$ *and* $0 < \lim_{n\to\infty} \frac{i}{n} < \omega$, $f(\underline{x})$ *be as defined in (4.21) and* $\underline{t} = (t_1, t_2, t_3)$ *be a positive solution of* $a_f(\underline{x}) = \frac{\mathtt{r}\underline{i}}{n\mathtt{l}}$, *where* $a_f(\underline{x}) = (\frac{x_i \partial f}{f \partial x_i})_{i=1}^3$. *Then* $\mathit{coef}\left(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, \underline{x}^{\underline{i}}\right)$ *can be approximated using the saddle point method for multivariate polynomials,*

$$\mathit{coef}\left(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, \underline{x}^{\underline{i}}\right) = \frac{4 f(\underline{t})^{\frac{n\mathtt{l}}{\mathtt{r}}}}{(\underline{t})^{\underline{i}} \sqrt{\left(2\pi \frac{n\mathtt{l}}{\mathtt{r}}\right)^3 |B_f(\underline{t})|}}(1 + o(1)),$$

where $B_f(\underline{x})$ *is a* 3×3 *matrix whose elements are given by* $B_{f(i,j)} = x_j \frac{\partial a_i}{\partial x_j} = B_{f(j,i)}$. *Also,* $\mathit{coef}\left(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, \underline{x}^{\underline{j}}\right)$ *can be approximated in terms of* $\mathit{coef}\left(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, \underline{x}^{\underline{i}}\right)$.

This approximation is called the local limit theorem *of* $\underline{j}$ *around* $\underline{i}$*. Explicitly, if* $\underline{u} := \sqrt{\frac{r}{n l}}(\underline{j} - \underline{i})$ *and* $\|\underline{u}\| = O((\ln n)^{\frac{1}{3}})$*, then*

$$\begin{aligned} \mathit{coef}\left(f(\underline{x})^{\frac{nl}{r}}, \underline{x}^{\underline{j}}\right) &= \underline{t}^{\underline{i}-\underline{j}}\mathit{coef}\left(f(\underline{x})^{\frac{nl}{r}}, \underline{x}^{\underline{i}}\right) \\ &\quad .\mathit{exp}\left(-\frac{1}{2}\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T\right)(1 + o(1)). \end{aligned}$$

Proof. The proof of the lemma follows from a modification of the proof of Theorem 2 of [26]. This is rather tedious and is therefore relegated to the Appendix 4.A. □

Note that in Lemma 30 we require that the matrix $B_f(\underline{t})$ be positive definite. In the following lemma we prove that the matrix B is positive definite for any multivariate polynomial with positive coefficients under appropriate technical conditions.

Lemma 31. *Consider a multivariate polynomial* $g(\underline{x}) = \sum_{\underline{i}} G_{\underline{i}}\underline{x}^{\underline{i}}$ *in* k *variables with non negative coefficients, where* $\underline{x} = (x_1, \ldots, x_k)$*. Let* A_g *be the matrix whose columns are equal to all the difference vectors* $\underline{i} - \underline{j}$*, where* $G_{\underline{i}}$ *and* $G_{\underline{j}}$ *are strictly positive. If the rows of matrix* A_g *are independent then the matrix* B_g *is positive definite, where* B_g *is defined in Lemma 30.*

Proof. In the proof for the sake of notational simplification we will drop the subscript g of B and a, where a is again defined in Lemma 30. The entries $B_{st}(\underline{x})$ turn out to be

$$B_{st} = \begin{cases} x_s x_t \dfrac{g(\underline{x})\frac{\partial^2 g(\underline{x})}{\partial x_s \partial x_t} - \frac{\partial g(\underline{x})}{\partial x_s}\frac{\partial g(\underline{x})}{\partial x_t}}{g(\underline{x})^2} & \text{if } s \neq t, \\ x_s\left(\dfrac{\frac{\partial g(\underline{x})}{\partial x_s} + x_s\frac{\partial^2 g(\underline{x})}{\partial^2 x_s}}{g(\underline{x})} - x_s\left(\dfrac{\frac{\partial g(\underline{x})}{\partial x_s}}{g(\underline{x})}\right)^2\right) & \text{if } s = t. \end{cases} \tag{4.23}$$

Assume that the polynomial $g(\underline{x})$ is given by

$$g(\underline{x}) = \sum_{\underline{i}} c_{\underline{i}}\underline{x}^{i},$$

where the coefficients $c_{\underline{i}}$ are non-negative. After algebraic simplifications the entry $B_{s,t}$ is given by

$$B_{st} = \frac{\sum_{\underline{i},\underline{j}}(j_s - i_s)(j_t - i_t)c_{\underline{i}}c_{\underline{j}}\underline{x}^{\underline{i}+\underline{j}}}{2\sum_{\underline{i}\underline{j}} c_{\underline{i}}c_{\underline{j}}\underline{x}^{\underline{i}+\underline{j}}}. \tag{4.24}$$

In order to show that matrix B is positive definite, we need to show that $\underline{y}^T.B.\underline{y}$ is positive for every non-zero $\underline{y} \in \mathbb{R}^k$. Using Eqn(4.24) $\underline{y}^T.B.\underline{y}$ simplifies to,

$$\underline{y}^T.B.\underline{y} = \sum_{\underline{i}\underline{j}}\left(\sum_{s=1}^{k}(j_s - i_s)\,y_s\right)^2 c_{\underline{i}}c_{\underline{j}}\underline{x}^{\underline{i}+\underline{j}}.$$

This implies that the matrix is positive definite by the linear independence of the rows of matrix A_g. □

Now we again concentrate on the polynomial f defined in Eqn(4.21). Note that Lemma 31 holds for f as its difference matrix has full row rank. The system of equations corresponding to $a_f(\underline{x}) = \frac{\mathtt{r}i}{n\mathtt{l}}$ is symmetric in x_1 and x_3. Hence a positive solution $\underline{x}$ of this system of equations satisfies $x_1 = x_3$ and the system reduces to the following equations,

$$x_1 \frac{(1+2x_1+x_2)^{\mathtt{r}-1} - (1-2x_1+x_2)^{\mathtt{r}-1}}{(1+2x_1+x_2)^{\mathtt{r}} + 2(1-x_2)^{\mathtt{r}} + (1-2x_1+x_2)^{\mathtt{r}}} = \omega - \alpha, \tag{4.25}$$

$$x_2 \frac{(1+2x_1+x_2)^{\mathtt{r}-1} - 2(1-x_2)^{\mathtt{r}-1} + (1-2x_1+x_2)^{\mathtt{r}-1}}{(1+2x_1+x_2)^{\mathtt{r}} + 2(1-x_2)^{\mathtt{r}} + (1-2x_1+x_2)^{\mathtt{r}}} = \alpha, \tag{4.26}$$

where $\alpha = \frac{i}{n}$.

In order to evaluate the second moment, we need to find the dominant terms of the summation in (4.22). To find all the dominant terms, let the term corresponding to $i = i_m$ i.e. $S_{i_m} = F_{i_m} C_{i_m}$ be a local maximum of $\{S_i\}_{i=0}^{n\omega}$. We first check if the end terms $S_{(2n\omega-n)^+}$ and $S_{n\omega}$ can be dominant. The assumption 2 of the Lemma 32 eliminates the possibility that $S_{(2n\omega-n)^+}$ is a dominant term. In the proof of Lemma 32 we will see that $\ln\left(S_{n\omega^2}\right)/n = 2W_{\mathrm{com}}(\omega)$. This with Eqn(4.22) implies that $S_{n\omega}$ is not a dominant term. So we consider i_m such that $0 < \lim_{n\to\infty} \frac{i_m}{n} < \omega$. Let $\Delta = i - i_m$ and $\alpha_m = \frac{i_m}{n}$. We expand F_i and C_i for $\Delta \in (-\sqrt{n}(\ln n)^{\frac{1}{3}}, \sqrt{n}(\ln n)^{\frac{1}{3}})$ in terms of F_{i_m} and C_{i_m} using Stirling's approximation and the local limit theorem of Lemma 30 respectively. Then,

$$\begin{aligned}
F_i &= F_{i_m} \exp\left(\Delta(\mathtt{l}-1)\ln\left(\frac{i_m(n-2n\omega+i_m)}{(n\omega-i_m)^2}\right)\right) \\
&\quad .\exp\left(\Delta^2\left(\frac{\mathtt{l}-1}{n\omega-i_m} + \frac{\mathtt{l}-1}{2i_m} + \frac{\mathtt{l}-1}{2(n-2n\omega+i_m)}\right)\right) \\
&\quad .\left(1+O\left(\frac{\Delta^3}{n^2}\right)\right), \\
C_i &= C_{i_m} \exp\left(\Delta\mathtt{l}\ln\left(\frac{t_1^2}{t_2}\right) - \frac{\Delta^2}{2n\sigma_c^2(\alpha_m)}\right)(1+o(1)),
\end{aligned}$$

where

$$\begin{aligned}
F_{i_m} &= \left(\frac{(n\omega-i_m)^{2(n\omega-i_m)} i_m^{i_m} (n-2n\omega+i_m)^{n-2n\omega+i_m}}{n^n}\right)^{\mathtt{l}-1} \\
&\quad .\mathtt{l}\sqrt{\mathtt{l}}(1+o(1)),
\end{aligned}$$

$$\sigma_c^2(\alpha_m) = \frac{1}{\mathtt{lr}\left((-1,1,-1).B_f(\underline{t})^{-1}.(-1,1,-1)^T\right)}. \tag{4.27}$$

Hence,

$$\frac{S_i}{S_{i_m}} = \exp\left(\Delta\left((\mathtt{l}-1)\ln\left(\frac{i_m(n-2n\omega+i_m)}{(n\omega-i_m)^2}\right) + \mathtt{l}\ln\left(\frac{t_1^2}{t_2}\right)\right)\right)$$
$$.\exp\left(\Delta^2\left(\frac{\mathtt{l}-1}{n\omega-i_m} + \frac{\mathtt{l}-1}{2i_m} + \frac{\mathtt{l}-1}{2(n-2n\omega+i_m)} - \frac{1}{2n\sigma_c^2(\alpha_m)}\right)\right)$$
$$\times(1+o(1)). \tag{4.28}$$

We know that there is a local maximum at $\Delta = 0$, hence the coefficient of Δ in (4.28) will vanish. This gives an additional equation governing α_m:

$$\left(\frac{\alpha_m(1-2\omega+\alpha_m)}{(\omega-\alpha_m)^2}\right)^{\mathtt{l}-1} = \left(\frac{t_2}{t_1^2}\right)^{\mathtt{l}}. \tag{4.29}$$

We solve (4.25), (4.26) and (4.29) and find all the solutions such that $0 < \alpha_m < \omega$, $t_1 > 0$, $t_2 > 0$ and the coefficient of Δ^2 in (4.28) is negative (this ensures that S_{i_m} is a local maximum). One of the possible solution to this system of polynomial equations is $\alpha_m = \omega^2$. This is because $\{C_i\}_{i=0}^{n\omega}$ and $\{F_i\}_{i=0}^{n\omega}$ are concave and convex sequences respectively, both achieving their extreme values at $i = n\omega^2$. Hence $\{S_i\}_{i=0}^{n\omega}$ also achieves an extreme value at $i = n\omega^2$. If $\alpha_m = \omega^2$ is a unique global maximum in the solution set of (4.25), (4.26) and (4.29), then we can get a closed form expression for second moment. We summarize this in the following lemma.

Lemma 32. *[Second Moment Method] Consider the binary regular LDPC ensemble $\mathbb{G}(n, \mathtt{l}, \mathtt{r})$. Then for $\omega \in (0,1)$, if $W_{com}(\omega) > 0$ and if the following conditions are satisfied,*

1. *$\alpha_m = \omega^2$ is the only solution of (4.25), (4.26) and (4.29) for which coefficient of Δ^2 in (4.28) is negative.*
2. *$\lim_{n\to\infty} \frac{\ln(S_{n\omega^2})}{n} > \frac{\ln(S_{(2n\omega-n)^+})}{n}$,*

then by the second moment method we have,

$$\lim_{n\to\infty} P\left(1-\epsilon \leq \frac{N(G,n\omega)}{\mathbb{E}[N(G,n\omega)]} \leq 1+\epsilon\right) \geq 1 - \frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2},$$

where

$$\delta(\omega,\mathtt{l},\mathtt{r}) = \frac{b_p(x_\omega)\sqrt{\mathtt{r}}\,\omega(1-\omega)\sigma_c(\omega^2)}{\sqrt{|B_f(x_\omega,x_\omega^2,x_\omega)|\,(\omega^2(1-\omega)^2-(\mathtt{l}-1)\sigma_c^2(\omega^2))}}$$
$$-1,$$
$$\sigma_c^2(\omega^2) = \frac{1}{\mathtt{l}\mathtt{r}\,((-1,1,-1).B_f(x_\omega,x_\omega^2,x_\omega)^{-1}.(-1,1,-1)^T)},$$

and x_ω is the only positive solution of (4.15).

Remark: Note that the conditions of Lemma 32 are hard to verify in general but they are typically easy to verify for any given regular LDPC ensemble.

Proof. We observe that the solution $\underline{t}$ of (4.25), (4.26) for $\alpha = \omega^2$ satisfies $t_2 = t_1^2$ and this system of equations reduces to a single equation which is identical to (4.15), the equation we need to solve to find $\mathbb{E}[N(G, n\omega)]$. Thus $t_1 = x_\omega$. By (4.28) and noting that the terms $S_{n\omega^2+\Delta}$ for $\Delta \notin (-\sqrt{n}(\ln n)^{\frac{1}{3}}, \sqrt{n}(\ln n)^{\frac{1}{3}})$ are much smaller than $S_{n\omega^2}$, we get

$$\begin{aligned}\mathbb{E}[N^2(G, n\omega)] &= S_{n\omega^2} \sum_{\Delta=-\sqrt{n}(\ln n)^{\frac{1}{3}}}^{\sqrt{n}(\ln n)^{\frac{1}{3}}} \exp\left(\frac{-\Delta^2}{2\sigma_s^2}\right)(1+o(1)),\\ &= S_{n\omega^2}\int_{-\infty}^{\infty} \exp\left(\frac{-x^2}{2\sigma_s^2}\right) dx(1+o(1)),\\ &= S_{n\omega^2}\sqrt{2\pi\sigma_s^2}(1+o(1)),\\ \text{where } \frac{1}{\sigma_s^2} &= \frac{1}{n\sigma_c^2(\omega^2)} - \frac{\mathtt{l}-1}{n\omega^2(1-\omega)^2}.\end{aligned}$$

Also $f(x_\omega, x_\omega^2, x_\omega) = p(x_\omega)^2$. To evaluate $S_{n\omega^2}$, we use Lemma 30 and Stirling's approximation for factorial terms. This gives,

$$\begin{aligned}\mathbb{E}[N^2(G, n\omega)] &= \frac{4\sigma_c(\omega^2)\mathtt{r}\sqrt{\mathtt{r}}\omega(1-\omega)}{2\pi n\sqrt{(\omega^2(1-\omega)^2 - (\mathtt{l}-1)\sigma_c^2(\omega^2))}}\\ &\quad \cdot \frac{e^{2n(\frac{\mathtt{l}}{\mathtt{r}}\ln(p(x_\omega)) - (\mathtt{l}-1)h(\omega) - \mathtt{l}\omega\ln(x_\omega))}}{\sqrt{|B_f(x_\omega, x_\omega^2, x_\omega)|}}(1+o(1)).\end{aligned}$$

We need the condition $W_{\text{com}}(\omega) > 0$, as

$$\lim_{n\to\infty} \frac{\ln(S_{n\omega^2})}{n} = 2W_{\text{com}}(\omega)$$

and

$$\lim_{n\to\infty} \frac{\ln(S_{n\omega})}{n} = W_{\text{com}}(\omega).$$

Clearly when $W_{\text{com}}(\omega)$ is negative, $S_{n\omega^2}$ can not be a global maximum. Now using Lemma 27, the second moment method gives us:

$$\lim_{n\to\infty} \mathrm{P}\left(1-\epsilon \le \frac{N(G, n\omega)}{\mathbb{E}[N(G, n\omega)]} \le 1+\epsilon\right) \quad \ge \quad 1 - \frac{\delta(\omega, \mathtt{l}, \mathtt{r})}{\epsilon^2}.$$

This proves the lemma. □

The bound obtained in Lemma 32 can in general only be evaluated numerically except for the cases when (4.15) can be solved analytically. For example for the $(3, 4)$-regular code we get,

$$\delta(\omega, 3, 4) = \frac{8\omega(1-\omega)(3-\Omega)}{\sqrt{(81 - 27\Omega + 16\omega(1-\omega)(8\omega - 8\omega^2 - 18 + 3\Omega))}}$$

$$\times \frac{1}{\sqrt{-21+80\omega(1-\omega)+9\Omega}},$$

where $\Omega = \sqrt{9-32\omega+32\omega^2}$.

4.5 Moment Calculations for Stopping Set Distribution

As shown in [44], the first moment of stopping set distribution is given by

$$\mathbb{E}[S(G,ns)] = \frac{\binom{n}{ns}}{\binom{n\mathtt{l}}{n\mathtt{l}s}} \text{coef}\left((\beta(x))^{\frac{\mathtt{l}n}{\mathtt{r}}}, x^{n\mathtt{l}s}\right),$$

where $\beta(x) = (1+x)^{\mathtt{r}} - \mathtt{r}x$. Applying the Lemma 3.1 and using Stirling's approximation we get

$$\mathbb{E}[S(G,ns)] = \frac{\sqrt{\mathtt{r}}e^{n(\frac{1}{\mathtt{r}}\ln(\beta(x_s))-(\mathtt{l}-1)h(s)-\mathtt{l}s\ln(x_s))}}{\sqrt{2\pi n b_\beta(x_s)}}(1+o(1)),$$

where x_s is the only positive solution of

$$x\frac{(1+x)^{\mathtt{r}-1}-1}{(1+x)^{\mathtt{r}}-\mathtt{r}x} = s. \tag{4.30}$$

The second moment is $\mathbb{E}[S^2(G,ns)] = \mathbb{E}[\sum_{\mathtt{s},\mathtt{s}'} \mathbb{1}_{\mathtt{s},\mathtt{s}'}(G,ns)]$, where $\mathtt{s},\mathtt{s}'$ are both stopping sets of cardinality ns. By definition of the ensemble, the expectation $\mathbb{E}[\mathbb{1}_{\mathtt{s},\mathtt{s}'}(G,ns)]$ depends only on the cardinality of the overlap between $\mathtt{s}$ and $\mathtt{s}'$. Hence like in the previous section for weight distribution we can fix $\mathtt{s}$ to be equal to $\mathtt{s} = \{1,2,\ldots,ns\}$ and for a given cardinality of overlap i we can fix $\mathtt{s}'$ to be equal to $\mathtt{s}'(i) = \{1,2,\ldots,i,ns+1,\ldots,2ns-i\}$. As $\mathtt{s}$ is fixed, we drop the subscript $\mathtt{s}$ in $\mathbb{1}_{\mathtt{s},\mathtt{s}'}(G,ns)$. This gives

$$\mathbb{E}[S^2(G,ns)] = \binom{n}{ns}\sum_{i=0}^{ns}\binom{ns}{i}\binom{n-ns}{ns-i}\mathbb{E}[\mathbb{1}_{\mathtt{s}'(i)}(G,ns)].$$

For $\mathbb{1}_{\mathtt{s}'(i)}(G,ns) = 1$, we need that every check node in G is either not connected to $\mathtt{s}$ or connected to $\mathtt{s}$ by more than one edge. Similarly every check node is either not connected to $\mathtt{s}'(i)$ or connected to $\mathtt{s}'(i)$ by more than one edge. This implies

$$\begin{aligned}\mathbb{E}[\mathbb{1}_{\mathtt{s}'(i)}(G,ns)] &= \text{coef}\left(g(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, x_1^{\mathtt{l}(ns-i)}x_2^{\mathtt{l}i}x_3^{\mathtt{l}(ns-i)}\right)\\ &\quad .\frac{1}{(n\mathtt{l})!}((\mathtt{l}(ns-i))!)^2(\mathtt{l}i)!(n\mathtt{l}-2n\mathtt{l}s+\mathtt{l}i)!,\end{aligned}$$

where

$$\begin{aligned}g(\underline{x}) &= (1+x_1+x_2+x_3)^{\mathtt{r}} - \mathtt{r}(1+x_1)^{\mathtt{r}-1}(x_2+x_3)\\ &\quad -\mathtt{r}x_1\left((1+x_3)^{\mathtt{r}-1}-(\mathtt{r}-1)x_3\right) - \mathtt{r}x_2\left((1+x_3)^{\mathtt{r}-1}-1\right).\end{aligned} \tag{4.31}$$

Thus

$$\begin{aligned}\mathbb{E}[S^2(G,ns)] &= \sum_{i=0}^{ns} \underbrace{\operatorname{coef}\left(g(\underline{x})^{n\frac{\mathtt{l}}{\mathtt{r}}}, x_1^{\mathtt{l}(ns-i)} x_2^{\mathtt{l}i} x_3^{\mathtt{l}(ns-i)}\right)}_{C_i} \\ & \underbrace{\cdot \frac{\binom{n}{ns}}{(n\mathtt{l})!}\binom{ns}{i}\binom{n-ns}{ns-i}((\mathtt{l}(ns-i))!)^2(\mathtt{l}i)!(\mathtt{l}(n-2ns+i))!}_{F_i}\,. \end{aligned} \tag{4.32}$$

To evaluate coef in (4.32) we use Theorem 2 of [26]. The polynomial g has full rank difference matrix. Thus B_g is positive definite by Lemma 31. By again applying the same line of arguments as for the weight distribution and if the conditions of Lemma 32 are true in the setting of stopping set distribution, then we get

$$\begin{aligned}\mathbb{E}[S^2(G,ns)] &= \frac{\sigma_c(s^2)\mathtt{r}\sqrt{\mathtt{r}}s(1-s)}{2\pi n\sqrt{(s^2(1-s)^2-(\mathtt{l}-1)\sigma_c^2(s^2))}} \\ &\cdot\frac{e^{2n(\frac{\mathtt{l}}{\mathtt{r}}\ln(\beta(x_s))-(\mathtt{l}-1)h(s)-\mathtt{l}s\ln(x_s))}}{|B_g(x_s,x_s^2,x_s)|}(1+o(1)).\end{aligned}$$

where $B_g(\underline{x})$ is same as defined in Lemma 30 with respect to $g(\underline{x})$ and x_s is the positive solution of (4.30). Hence by the second moment method we have,

$$\lim_{n\to\infty} \mathrm{P}\left(1-\epsilon \le \frac{S(G,ns)}{\mathbb{E}[N(G,ns)]} \le 1+\epsilon\right) \ge 1-\frac{\delta(s,\mathtt{l},\mathtt{r})}{\epsilon^2},$$

where

$$\delta(s,\mathtt{l},\mathtt{r}) = \frac{b_\beta(x_s)\sqrt{r}s(1-s)\sigma_c(s^2)}{\sqrt{|B_g(x_s,x_s^2,x_s)|\,(s^2(1-s)^2-(\mathtt{l}-1)\sigma_c^2(s^2))}} - 1,$$

$$\sigma_c^2(s^2) = \frac{1}{\mathtt{lr}\,((-1,1,-1).B_g(x_s,x_s^2,x_s)^{-1}.(-1,1,-1)^T)}.$$

4.6 Discussion

Fix the relative weight ω. If $\epsilon \in (0,1)$ then we conclude that asymptotically for at least a fraction $1-\frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2}$ of codes, the number of codewords $N(G,n\omega)$ (for a fixed ω) is at most a constant factor away from the ensemble average.

Also from Fig. 4.1 we see that $1-\frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2}$ is an increasing function of ω for $\omega \in (\omega_{min}, 0.5)$ and is a decreasing function for $\omega > 0.5$. It is equal to 1 for $\omega = 0.5$. This implies that asymptotically in almost all the codes there are $\mathbb{E}[N\left(G,\frac{n}{2}\right)](1\pm\epsilon)$ codewords of weight $\frac{n}{2}$. For ω close to the typical minimum distance ω_{min}, the bound stays nontrivial. In Table 4.1 and Table 4.2,

$(\mathtt{l},\mathtt{r})$-code	ω_{min}	$1-\frac{\delta(\omega_{min},\mathtt{l},\mathtt{r})}{0.95^2}$
$(3,6)$	0.0227334	0.740611
$(6,12)$	0.0956337	0.963306
$(12,24)$	0.109404	0.999617
$(24,48)$	0.110026	~ 1.0

Table 4.1: $\lim_{\omega\to\omega_{min}+} 1-\frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2}$ for rate $=\frac{1}{2}$ and $\epsilon = 0.95$.

$(\mathtt{l},\mathtt{r})$-code	ω_{min}	$1-\frac{\delta(\omega_{min},\mathtt{l},\mathtt{r})}{0.95^2}$
$(3,4)$	0.112159	0.667889
$(6,8)$	0.207437	0.989098
$(12,16)$	0.214428	0.999994
$(24,32)$	0.214502	~ 1.0

Table 4.2: $\lim_{\omega\to\omega_{min}+} 1-\frac{\delta(\omega,\mathtt{l},\mathtt{r})}{\epsilon^2}$ for rate $=\frac{1}{4}$ and $\epsilon = 0.95$.

$\lim_{\omega\to\omega_{min}+} 1-\frac{\delta(\omega_{min},\mathtt{l},\mathtt{r})}{0.95^2}$ is given for regular codes of rate 0.5 and 0.25 respectively. We observe that if we fix the rate and let $\mathtt{l}$ and $\mathtt{r}$ increase then the bound approaches 1 for all ω for which $W_{\text{com}}(\omega)$ is positive. This implies that for regular ensembles with large left and right degree almost all the codes have a weight distribution which is very close to the ensemble average. We observe the same phenomenon for stopping set distribution. We see that the second moment method can capture the concentration property of the weight distribution and stopping set distribution for regular ensembles with large left and right degrees. However for the regular ensembles in general it fails to do so. Potentially by applying more sophisticated methods one could obtain better bounds, e.g., the second moment method with conditioning [45] or other methods given in [46].

4.A Proof of Lemma 30

We modify the proof of Theorem 2 of [26] to prove Lemma 30. Let $\varphi_n(\underline{z}) = f(\underline{z})^{\frac{n\mathtt{l}}{\mathtt{r}}}$, $R = [-\pi,\pi]^3$ and $I = \sqrt{-1}$. We also expand $\varphi_n(\underline{z})$ as $\varphi_n(\underline{z}) = \sum_{\underline{k}} a_n(\underline{k})\underline{z}^{\underline{k}}$. Let $\underline{t}$ be the positive solution of $a_f(\underline{x}) = \frac{\mathtt{r}\underline{i}}{n\mathtt{l}}$. From the inverse Fourier transform, we get

$$\frac{1}{(2\pi)^3}\int_R \frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})} e^{-I\underline{j}\cdot\underline{v}^T}\, d\underline{v} = \frac{a_n(\underline{j})\underline{t}^{\underline{j}}}{\varphi_n(\underline{t})}. \tag{4.33}$$

We recall that the Fourier transform of a Gaussian is again a Gaussian,

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-I\underline{u}\cdot\underline{s}^T - \frac{\underline{s}\cdot B_f(t)\cdot\underline{s}^T}{2}}\, d\underline{s} = \sqrt{\frac{(2\pi)^3}{|B_f(\underline{t})|}}\, e^{-\frac{1}{2}\underline{u}\cdot B_f(\underline{t})^{-1}\cdot\underline{u}^T}. \tag{4.34}$$

Also, from the proof of Theorem 2 of [26], for any function $K(n)$ growing with n, we have

$$\left|\int_{[-K(n),K(n)]^3} e^{-I\underline{u}.\underline{s}^T-\frac{\underline{s}.B_f(t).\underline{s}^T}{2}}d\underline{s}-\int_{(-\infty,\infty)^3} e^{-I\underline{u}.\underline{s}^T-\frac{\underline{s}.B_f(t).\underline{s}^T}{2}}d\underline{s}\right| = O\left(\frac{1}{K(n)}\right). \tag{4.35}$$

We would like to show that for $n \to \infty$

$$\left|\left(\frac{\mathtt{nl}}{\mathtt{r}}\right)^{\frac{3}{2}}\int_R \frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})}e^{-I\underline{j}.\underline{v}^T}d\underline{v}-4\sqrt{\frac{(2\pi)^3}{|B_f(\underline{t})|}}e^{-\frac{1}{2}\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T}\right| = o(1). \tag{4.36}$$

To prove this, we write $\varphi_n(\underline{t}e^{I\underline{v}})$ in exponential-log form and take the Taylor series expansion of the exponent around $\underline{v}=0$,

$$\varphi_n(\underline{t}e^{I\underline{v}}) = e^{\left(\frac{\mathtt{nl}}{\mathtt{r}}(\ln(f(\underline{t}))+Ia_f(\underline{t}).\underline{v}^T-\frac{\underline{v}.B_f(t).\underline{v}^T}{2}+O(\|\underline{v}\|^3))\right)}. \tag{4.37}$$

Note that as $\ln(\varphi_n(\underline{t}))$ is analytic, so all the third order partial derivative of $\varphi_n\left(\underline{t}e^{I\underline{v}}\right)$ are bounded. Now we partition R as $R=\cup_{i=1}^5 R_i$, where $R_1=[-\delta,\delta]^3, R_2=[-\delta,\delta]\times[\pi-\delta,\pi+\delta]^2, R_3=[\pi-\delta,\pi+\delta]\times[-\delta,\delta]\times[\pi-\delta,\pi+\delta], R_4=[\pi-\delta,\pi+\delta]^2\times[-\delta,\delta], R_5=R\backslash(R_1\cup R_2\cup R_3\cup R_4)$. Here δ can be any decaying function of n which satisfies that as $n\to\infty$ then $n\delta^2\to\infty$ and $n\delta^3\to 0$. We choose $\delta=n^{-\frac{2}{5}}$. This ensures that the term $O(\|\underline{v}\|^3)$ is negligible. By the symmetry of $f(\underline{x})$, $\varphi_n(x_1,x_2,x_3)=\varphi_n(x_1,-x_2,-x_3)=\varphi_n(-x_1,x_2,-x_3)=\varphi_n(-x_1,-x_2,x_3)$. This implies,

$$\int_{R_1}\frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})}e^{-I\underline{j}.\underline{v}^T}d\underline{v}=\int_{R_k}\frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})}e^{-I\underline{j}.\underline{v}^T}d\underline{v},$$

where $k\in\{2,3,4\}$. Now,

$$\int_{R_1}\frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})}e^{-I\underline{j}.\underline{v}^T}d\underline{v} \overset{a_f(\underline{x})=\frac{\mathtt{r}\underline{i}}{\mathtt{nl}}}{\overset{(4.37)}{=}} \int_{R_1} e^{I(\underline{i}-\underline{j}).v^T-\frac{\mathtt{nl}}{2\mathtt{r}}\underline{v}.B_f(\underline{t}).\underline{v}^T+O(n\delta^3)}d\underline{v}. \tag{4.38}$$

The change of variable $\underline{y}:=\sqrt{\frac{\mathtt{nl}}{\mathtt{r}}}\underline{v}$ in Eqn(4.38) gives,

$$\int_{R_1}\frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})}e^{-I\underline{j}.\underline{v}^T}d\underline{v} = \left(\frac{\mathtt{r}}{\mathtt{nl}}\right)^{\frac{3}{2}}\int_{R_1'} e^{-I\underline{u}.\underline{y}^T-\frac{\underline{y}.B_f(\underline{t}).\underline{y}^T}{2}}(1+O(n^{-\frac{1}{5}}))d\underline{y}, \tag{4.39}$$

where $R_1' = [-\sqrt{\frac{1}{\mathtt{r}}}n^{\frac{1}{10}}, \sqrt{\frac{1}{\mathtt{r}}}n^{\frac{1}{10}}]^3$. Using Eqn(4.34), Eqn(4.35) in Eqn(4.39) gives,

$$\begin{aligned}\int_{R_1} \frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})} e^{-I\underline{j}.\underline{v}^T} d\underline{v} &= \left(\frac{\mathtt{r}}{n\mathtt{l}}\right)^{\frac{3}{2}} O(n^{-\frac{1}{10}}) \\ &+ \left(\frac{\mathtt{r}}{n\mathtt{l}}\right)^{\frac{3}{2}} \sqrt{\frac{(2\pi)^3}{|B_f(\underline{t})|}} e^{-\frac{\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T}{2}} (1 + O(n^{-\frac{1}{5}})). \qquad (4.40)\end{aligned}$$

Recall that,

$$\int_{R_5} \left| \frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})} e^{-I\underline{j}.\underline{v}^T} \right| d\underline{v} = \int_{R_5} \left| \frac{f(\underline{t}e^{I\underline{v}})}{f(\underline{t})} \right|^{\frac{n\mathtt{l}}{\mathtt{r}}} d\underline{v}.$$

Further $f(\underline{t})$ is a 3-variable polynomial of finite degree. Let $f(\underline{t}) = \sum_{\underline{k}} b(\underline{k})\underline{t}^{\underline{k}}$. Then by some algebraic manipulation we get,

$$\left| \frac{f(\underline{t}e^{I\underline{v}})}{f(\underline{t})} \right|^2 = 1 - \frac{\sum_{\underline{k} \neq \underline{l}} b(\underline{k})b(\underline{l})\underline{t}^{\underline{k}+\underline{l}}(1 - \cos\left((\underline{k} - \underline{l}).v^T\right)}{f(\underline{t})^2}. \qquad (4.41)$$

Also $f(\underline{t})$ has $1, t_1^2, t_2^2, t_3^2$ as its summation terms and in R_5 at least one of the variable v_k satisfies $v_k \notin [-\delta, \delta]$ where $k \in \{1, 2, 3\}$. This with Eqn(4.41) implies that for some positive constants c, c_1,

$$\begin{aligned}\int_{R_4} \left| \frac{f(\underline{t}e^{I\underline{v}})}{f(\underline{t})} \right|^{\frac{n\mathtt{l}}{\mathtt{r}}} d\underline{v} \leq \pi^3(1 - c_1\delta^2)^{\frac{n\mathtt{l}}{2\mathtt{r}}} &= \pi^3(1 - c_1 n^{-\frac{4}{5}})^{\frac{n\mathtt{l}}{2\mathtt{r}}}, \\ &= O(e^{-cn^{\frac{1}{5}}}).\end{aligned}$$

By combining the above steps we get,

$$\begin{aligned}\left| \left(\frac{n\mathtt{l}}{\mathtt{r}}\right)^{\frac{3}{2}} \int_R \frac{\varphi_n(\underline{t}e^{I\underline{v}})}{\varphi_n(\underline{t})} e^{-I\underline{j}.\underline{v}^T} d\underline{v} - 4\sqrt{\frac{(2\pi)^3}{|B_f(\underline{t})|}} e^{-\frac{1}{2}\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T} \right| \\ = O(n^{-\frac{1}{10}}). \qquad (4.42)\end{aligned}$$

By using Eqn(4.33) in Eqn(4.42), we get

$$\begin{aligned}\left| \left(\frac{n\mathtt{l}}{\mathtt{r}}\right)^{\frac{3}{2}} \frac{(2\pi)^3 a_n(\underline{j})\underline{t}^{\underline{j}}}{\varphi_n(\underline{t})} - 4\sqrt{\frac{(2\pi)^3}{|B_f(\underline{t})|}} e^{-\frac{1}{2}\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T} \right| \\ = O(n^{-\frac{1}{10}}). \qquad (4.43)\end{aligned}$$

The approximation of coef$(f(\underline{x})^{\frac{n\mathtt{l}}{\mathtt{r}}}, \underline{x}^{\underline{i}})$ is obtained by substituting $\underline{j} = \underline{i}$ (which implies $\underline{u} = (0, 0, 0)$) in (4.43). Also for the local limit theorem to hold we need in (4.43) that $e^{\frac{1}{2}\underline{u}.B_f(\underline{t})^{-1}.\underline{u}^T} n^{-\frac{1}{10}} = o(1)$. For our application choosing $\|\underline{u}\| = O((\ln n)^{\frac{1}{3}})$ suffices.

Existence Proofs of Some EXIT Like Functions 5

The chapter is organized in the following way. An introduction is given in Section 5.1. Relevant definitions and the statement of Krasnoselskii-Rabinowitz (KR) theorem is given in Section 5.2. Some examples are given in Section 5.3 for which existence of EXIT like function is proved by the KR theorem. We conclude with a discussion in Section 5.4.

5.1 Introduction

Consider transmission over a BMS channel family using a binary LDPC ensemble. Let the decoder be any symmetric message-passing decoder as defined in [10]. The performance of such a combination can be analyzed in the limit of infinite blocklength by the density evolution recursion which has the following form:

$$a_l = G(c, a_{l-1}), \tag{5.1}$$

where $a_{l-1}(x)$ is the density of messages in the $(l-1)^{\text{th}}$ iteration, $a_l(x)$ is the density of messages in the l^{th} iteration and $c(x)$ is the channel log-likelihood density. We say that a (channel, message) density pair (c, a) is a fixed point of the map G if

$$a = G(c, a).$$

The fixed points are of interest to us as for some cases (like the BP decoder) the performance is determined by means of the fixed points of the map G.

An EXIT like curve for a given combination of a code ensemble, BMS channel family and a symmetric decoder is a function of the fixed point pairs (c, a) of the corresponding density evolution map. In general the density a is infinite dimensional. So, to have a convenient parametrization of the fixed point pairs (c, a) we project the density a to one dimension by a functional of

a. Consider the BP decoder. The functional can be the entropy $H(a)$ of the density a, which is given by,

$$H(a) = \int_{-\infty}^{\infty} a(x) \log_2 \left(1 + e^{-x}\right) dx.$$

The entropy functional is one among the many choices e.g., the error probability, the Battacharya parameter etc. Thus in order to compute the EXIT like function for the BP decoder and a BMS channel family, we need to compute all the fixed point density pairs (c, a). This corresponds to $H(a)$ taking value in the range $[0, 1]$.

For the BP decoder, let us give the example of transmission over the BEC using LDPC ensemble with degree distribution (λ, ρ). The density evolution recursion is

$$a_l = \epsilon\lambda\left(1 - \rho(1 - a_{l-1})\right).$$

Note that the message density a_l in this case is a scalar which keeps track of the probability of erasure messages. The fixed point pairs for this recursion can be easily computed and are given by $(\epsilon(x), x)$, where

$$\epsilon(x) = \frac{x}{\lambda(1 - \rho(1 - x))}, \tag{5.2}$$

with $\lambda(x) = \sum \lambda_i x^{i-1}$. The Extended BP (EBP) GEXIT function introduced in [1] for BEC is an example of an EXIT like function. The EBP GEXIT function for BEC is given by

$$\left(\epsilon(x), \Lambda\left(1 - \rho(1 - x)\right)\right), \quad x \in [0, 1],$$

where $\Lambda(x) = \sum \Lambda_i x^i$, $\{\Lambda_i\}$ is the degree distribution of variable nodes from node perspective and $\epsilon(x)$ is defined in Eqn(5.2). Thus the EBP GEXIT function for BEC is a parametric function defined over a *complete set* of fixed points of the density evolution map. By a complete set we mean that the entropy of fixed points takes all the values from zero to one. The EBP GEXIT function for BEC encodes both the behavior of the BP as well as the MAP decoder as was shown in [1]. A generalization of the EBP GEXIT function for general BMS channels was introduced in [2]. It is conjectured that again the EBP GEXIT function completely characterizes the behavior of the BP and the MAP decoder for general BMS channels. The density evolution map for BP decoding over general BMS channel is given by

$$a_l = c \otimes \lambda\left(\rho(a_{l-1})\right),$$

where $\lambda(a) = \sum_i \lambda_i a^{\otimes(i-1)}$ and $\rho(a) = \sum_i \rho_i a^{\boxplus(i-1)}$. The operator $\otimes$ is the convolution operator and $\boxplus$ is the check node side convolution defined in [12]. Note that unlike the case of BEC, where the fixed point pairs can be computed explicitly, no such characterization exist in the general case. However, the fixed points pairs can be computed numerically. Figure 5.1 shows the EBP GEXIT

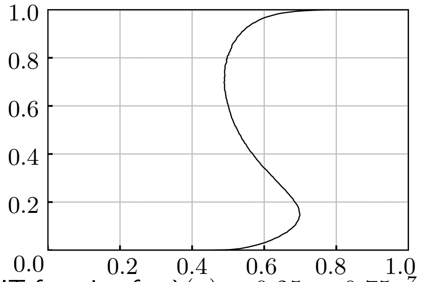


Figure 5.1: EBP GEXIT function for $\lambda(x) = 0.25x + 0.75x^7$, $\rho(x) = x^7$ and BSC.

function for the degree distribution pair ($\lambda(x) = 0.25x + 0.75x^7$, $\rho(x) = x^7$), assuming that transmission takes place over the Binary Symmetric Channel (BSC). Note that this curve smoothly connects the point $(1, 1)$, corresponding to the channel BSC with crossover probability 0.5, with the point $(h^{\text{stab}}, 0)$, where h^{stab} corresponds to that channel parameter at which the coding system changes its stability behavior. More precisely, h^{stab} is the entropy of the channel log-likelihood density $c(x)$ whose Battacharya parameter $\mathfrak{B}(c)$ is equal to

$$\mathfrak{B}(c) = \frac{1}{\lambda'(0)\rho'(1)},$$

where

$$\mathfrak{B}(c) = \int_{-\infty}^{\infty} c(x) e^{-\frac{x}{2}}.$$

The curve in Figure 5.1 was computed using a procedure suggested in [2]. This procedure guarantees in general the existence of a fixed point density for every point on the vertical axis. Unfortunately, it does *not* guarantee that the set of fixed points so computed forms a smooth one-dimensional manifold.

It can be easily shown that the fixed point pairs which are reached by running the density evolution recursion are ordered by physically degradation. However, there are also the fixed points which are not reachable by the density evolution recursion. For example, in Figure 5.2 we plot the EBP GEXIT function of regular $(3, 6)$ code for transmission over the BSC. The branch from point a to b represents fixed points which are not reachable by the density evolution recursion. It is a challenging task to prove that the fixed points not reachable by density evolution recursion are also ordered by physical degradation. Numerically calculation of fixed points suggests this to be true. Such a property, however, is required in order to complete the theory of EBP GEXIT functions. E.g., it is known that if the curve is smooth then the area it encloses is equal to the rate of the code. Combined with the General Area Theorem (first proved for the BEC in [47] and then extended to the general case in [2]) this statement on the area gives rise to bounds on the MAP performance for sparse graph codes. For the BEC it has been shown that in many cases the bound is tight and it is conjectured to be tight not only for the BEC but also in the general case.

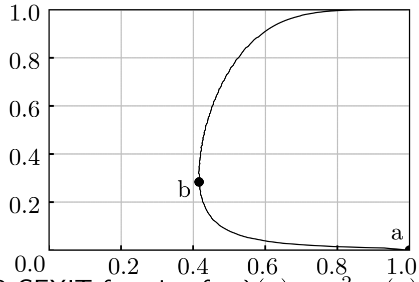


Figure 5.2: EBP GEXIT function for $\lambda(x) = x^2$, $\rho(x) = x^5$ and BSC.

The existence of the EBP GEXIT function is therefore a fundamental question at the heart of the asymptotic theory of sparse graph codes. We introduce tools from non-linear analysis, in particular the Krasnoselskii-Rabinowitz (KR) bifurcation theorem which can be useful to prove the existence of EXIT like curves in some cases.

5.2 Definitions and Theorem Related to the Existence of Fixed Points

As discussed in the last section, it is a difficult task to prove the existence of the EBP GEXIT curve for general channels. I.e., it is difficult to prove that the set of fixed point densities of density evolution forms a differentiable one-dimensional manifold.

Although we currently do not know how to prove the existence for the general case, a fundamental theorem of non-linear analysis, called the Krasnoselskii-Rabinowitz (KR) theorem ([48], [49]), can be helpful in some instances to establish the existence of an unbounded connected component of fixed points. To be more precise: density evolution represents a non-linear map in the space of densities. If we are given a degree distribution pair with a non-zero fraction of degree-two variable nodes and a family of BMS channels, then this map has a *bifurcation* point for that channel parameter which corresponds to the stability condition. In other words, consider the channel parameter for which the linearization of the density evolution map around the density corresponding to perfect decoding has its largest eigenvalue equal to one. Then this channel parameter is a bifurcation point. Under some technical conditions the KR theorem then guarantees that there is a connected set of fixed points which starts at this bifurcation point and which either extends to infinity or which connects back to another bifurcation point. This is not quite as strong a statement as we would wish: we are not guaranteed that this connected set forms a smooth manifold, nor do we know that the curve connects to the fixed point corresponding to the worst density and worst channel. Nevertheless, if the theorem applies, we at least know the existence of the EBP GEXIT curve locally around the stability point. Before we can show some cases where the KR theorem can be applied let us quickly review the main notation and the main statement.

We denote a generic Banach space by X (e.g., $X = \mathbb{R}^N$). We denote elements of X in boldface letters, i.e., $\mathbf{x} \in X$. Before defining a bounded linear operator on X, we define a bounded set. A set $S \subset X$ is bounded if

$$||x|| \leq \delta, \forall x \in X,$$

where $\delta > 0$. A linear operator $T : X \to X$ is *bounded* if, for any bounded subset B of X, the image $T(B)$ is also bounded. Let $\mathbf{N}_n(\lambda)$ denotes the null space of $\left(I - \frac{T}{\lambda}\right)^n$, where $\lambda \in \mathbb{R}$, I is the identity map and n is a positive integer. Then for an eigenvalue λ of T, the *multiplicity* of λ is defined as:

$$\max_{n \geq 1} \left\{ \text{Dimension} \left(N_n(\lambda) \right) \right\}.$$

By [48, Thm 16.1], the multiplicity of an eigenvalue is finite. We denote the space of bounded linear operators from X to X by $L(X)$.

We are interested in maps of the form $G : \mathbb{R} \times X \to X$. The first argument γ of $G(\gamma, \mathbf{x})$ is called the *parameter*. In our setting the parameter will be the *channel parameter* (e.g., the erasure probability of the BEC or the cross-over probability for the BSC). We recall the following definitions:

- Completely Continuous (CC) Map: A map $G : \mathbb{R} \times X \to X$ is CC if it maps every bounded set A of $\mathbb{R} \times X$ to a relatively compact set in X.

- Frechet differentiable: Let $G : \mathbb{R} \times X \to X$ be a map such that $G(\gamma, \mathbf{0}) = \mathbf{0}$. G is Frechet differentiable at $\mathbf{x} = \mathbf{0}$ if there exists $T \in L(X)$ such that, given $\epsilon > 0$ and an interval $[\gamma_0, \gamma_1]$ of $\mathbb{R}$, there exists $\delta > 0$ with the property that $||\mathbf{y}|| < \delta$ implies

$$\frac{||G(\gamma, \mathbf{y}) - \gamma T \mathbf{y}||}{||\mathbf{y}||} < \epsilon$$

for all $\gamma \in [\gamma_0, \gamma_1]$. Note that δ depends on both the choice of interval and the value of ϵ. We say that γT is the Frechet derivative of G at $\mathbf{0}$.

We denote the set of non trivial fixed points of G by $S = \{(\gamma, \mathbf{x}) : G(\gamma, \mathbf{x}) = \mathbf{x}, \mathbf{x} \neq \mathbf{0}\}$ and the closure of S by $\overline{S}$. If a point $(\mu, \mathbf{0}) \in \overline{S}$, then μ is called a *bifurcation point* for the solutions to $G(\gamma, \mathbf{x}) = \mathbf{x}$.

Theorem 1 (KR Theorem)**.** *[48, Theorem 17.8] Let X be a Banach space and let $G : \mathbb{R} \times X \to X$ be a map. Let $S = \{(\gamma, \boldsymbol{x}) : G(\gamma, \boldsymbol{x}) = \boldsymbol{x}, \boldsymbol{x} \neq 0\}$ be the set of non trivial fixed points of G and let $\overline{S}$ denote the closure of S. Assume that the following hypothesis holds.*

1. *$G(\gamma, \boldsymbol{x})$ is a completely continuous map.*

2. *$G(\gamma, \boldsymbol{x})$ is Frechet differentiable at $\boldsymbol{0}$, with Frechet derivative γT.*

3. *Let $\frac{1}{\mu}$ be an eigenvalue of T which is of odd algebraic multiplicity.*

Then there exists a maximal closed connected subset C_μ of $\overline{S}$ which contains $(\mu, \boldsymbol{0})$ and one of the following is true.

1. *C_μ is unbounded in $\mathbb{R} \times X$.*
2. *C_μ contains $(\mu^*, \boldsymbol{0})$ for some other bifurcation point $\mu^* \neq \mu$.*

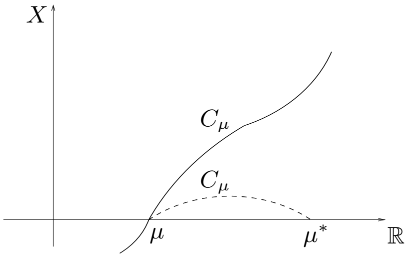


Figure 5.3: The solid curve shows how the component C_μ would look like if the first conclusion of theorem holds and the dotted one shows the how the component C_μ would look like if the second conclusion holds.

A graphical representation of the KR theorem is shown in Figure 5.3.

Our basic plan of attack is the following. In our setting $\mathbf{x}$ will denote a density, and G will be the density evolution map. We want to parametrize the space in such a way that $\mathbf{0}$ denotes the desired fixed point corresponding to perfect decoding. The parameter μ will parametrize the channel. If we can show that the linearization of the density evolution map around $\mathbf{0}$ has eigenvalue $1/\mu$, where μ denotes the channel parameter which corresponds to the stability condition, and if the linearization fulfills the desired technical conditions, then there is a connected component of fixed-points which either extends to infinity or is connected to another bifurcation point. At least locally, we will therefore have proved the existence of a connected component of fixed points.

In the sequel it is also good to know the following fact.

Theorem 2. *[48, Theorem 17.4] Let $G : \mathbb{R} \times X \to X$ be a completely continuous and Frechet differentiable at $\boldsymbol{0}$, with derivative γT. If $\frac{1}{\mu}$ is not an eigenvalue of the compact linear operator T, then there exist $\epsilon, \eta > 0$ such that $G(\gamma, x) \neq x$ for all (γ, x) for which $|\gamma - \mu| < \epsilon$ and $0 < ||x|| < \eta$. In particular, μ is not a bifurcation point for the solutions to $G(\gamma, x) = x$.*

We also use the following terminology in the rest of this chapter. Let $G : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N$ be a map of the form $G = \{G_i\}_{i=1}^N$, where $G_i : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}$ is a multivariate polynomial in the components of $\mathbf{x}$. The map G_i can be written as $G_i(\mu, \mathbf{x}) = G_i^1(\mathbf{x}) + \mu G_i^2(\mathbf{x})$, where $G_i^j : \mathbb{R}^N \to \mathbb{R}$, $j \in \{1, 2\}$. Then we say that G is a vector polynomial map.

5.3 Examples

In principle, we would like to apply the KR theorem directly to the BP or min-sum decoder. But there are some technical conditions that make the direct application difficult. To give an example of these technical difficulties, the bifurcation point for the BP decoder appears when the Battacharyya parameter is equal to $\frac{1}{\lambda'(0)\rho'(1)}$. This suggest that the Battacharyya parameter should play the role of the parameter in the setting of the KR theorem. The theorem requires that the parameter γ takes on values in $\mathbb{R}$ and not only on $[0,1]$. Therefore, we can not just work in the space of symmetric densities (for which the Battacharyya parameter is in the range $[0,1]$) but we are required to extend the space. How this is best done is currently an open question. Because of these technical difficulties, we consider quantized decoders. First we show the application of the KR theorem to the simplest possible case.

Example 1 (BP Decoder for Binary Erasure Channel)**.** *It is instructive (and easy) to analyze the fixed points of the density evolution map for the BEC(ϵ). Consider a degree distribution pair (λ, ρ) with $\lambda'(0)\rho'(1) > 0$. The density evolution recursion reads*

$$x_l = \epsilon\lambda\left(1 - \rho\left(1 - x_{l-1}\right)\right).$$

We take the space X to be $X = \mathbb{R}$ and set $G(\epsilon, x) = \epsilon\lambda\left(1 - \rho\left(1 - x\right)\right)$. Here the erasure probability ϵ plays the role of the parameter. As $G(\epsilon, x)$ is a polynomial map, it is completely continuous by Lemma 33 and Frechet differentiable by Lemma 34. From Lemma 34, the Frechet derivative of $G(\epsilon, x)$ is given by $\epsilon T x = \epsilon\lambda'(0)\rho'(1)x$. Thus the parameter ϵ appears multiplicatively, as required by the KR theorem.

Trivially, $\lambda'(0)\rho'(1)$ is the eigenvalue of the operator T and this eigenvalue has multiplicity one (the space is only one-dimensional), which is odd. Since by assumption $\lambda'(0)\rho'(1) > 0$, this eigenvalue is strictly positive. Thus $1/\left(\lambda'(0)\rho'(1)\right)$ is a bifurcation point. As there can be only one eigenvalue of T, there can be at most one bifurcation point (Theorem 2). Thus the first conclusion of Theorem 1 holds true: the connected component of fixed points containing the bifurcation point $\left(\frac{1}{\lambda'(0)\rho'(1)}, \boldsymbol{0}\right)$ is unbounded.

Of course, for this simple example we even have an explicit characterization of this connected set of fixed points and an application of the powerful KR theorem is not needed. But for only slightly more elaborate examples an explicit characterization is typically no longer available.

Consider now transmission over the BSC with transition probability p and min-sum (MS) decoding. For iteration l, let $M^{(l)}_{m->n}$ be the message sent from check node m to variable node n and $M^{(l)}_{n->m}$ be the message sent from variable node n to check node m. We denote the set of neighbors of a node m by $\mathcal{N}(m)$. If we assume that we represent messages as log-likelihood ratios then the processing rules in each iterations are as follows:

1. Processing rule at check nodes—for each m and each $n \in \mathcal{N}(m)$,

$$M^{(l)}_{m->n} = \prod_{n' \in \mathcal{N}(m)n} \operatorname{sgn}\left(M^{(l)}_{n'->m}\right) \min_{n' \in \mathcal{N}(m)n} \left|M^{(l)}_{n'->m}\right| \tag{5.3}$$

2. Processing rule at variable nodes—for each n and each $m \in \mathcal{N}(n)$,

$$M^{(l)}_{n->m} = L_n + \sum_{m' \in \mathcal{N}(n)m} M^{(l-1)}_{m'->n}, \tag{5.4}$$

   where L_n denotes the initial log-likelihood ratio received by node n.

We claim that there exists a one-to-one mapping between the messages of the min-sum decoder and the set of integers $\mathbb{Z}$. More precisely, the messages of the min-sum decoder are of the form $i \ln \frac{1-p}{p}, i \in \mathbb{Z}$. This can be easily seen by induction. The initial messages from the variable nodes to the check nodes are $\pm \ln \frac{1-p}{p}$. At the check nodes if all the incoming messages are of the form $i \ln \frac{1-p}{p}$, then by inspecting the check node processing rule given in Eqn(5.3) we see that the outgoing message is again of this form. At the variable nodes, all the messages are added up which clearly preserve this property. We can therefore equivalently formulate message-passing under min-sum on the lattice $\mathbb{Z}$ by assuming that the initial messages are from the set $\{\pm 1\}$ and have probabilities $(1-p)$ and p, respectively.

In order to be able to apply the KR theorem, below we consider *bounded* versions of min-sum, i.e., we bound the absolute value of the messages to M, where M is a fixed integer. More precisely, we assume that message alphabet is $\mathcal{M} = \{-M, -(M-1), \cdots, -1, 0, 1, \cdots, M-1, M\}$. As mentioned before, $\mathcal{M}_c = \{-1, 1\}$. The message passing rule for the check node side is the same as given by Eqn(5.3). On the other hand, to enforce the boundedness constraint, we need to slightly modify the message-passing rule for variable nodes. For a node of degree d_v the rule is defined by:

$$\Psi_v(m_0, m_1, \cdots, m_{d_v-1}) = \begin{cases} M & \begin{array}{l}\exists i \text{ s.t. } m_i = M \\ \nexists j \text{ s.t. } m_j \neq -M\end{array} \\ \\ -M & \begin{array}{l}\exists i \text{ s.t. } m_i = -M \\ \nexists j \text{ s.t. } m_j \neq M\end{array} \\ \\ 0 & \begin{array}{l}\exists i, j \text{ s.t.} \\ m_i = -M, m_j = M\end{array} \\ \\ \mathcal{Q}\left(\sum_{i=0}^{d_v-1} m_i\right) & \text{otherwise,} \end{cases} \tag{5.5}$$

where the quantization function $\mathcal{Q}(x) = M$ if $x \geq M$, $\mathcal{Q}(x) = -M$ if $x \leq -M$ and equal to x otherwise. Note that the exact rule for the case when both M and $-M$ are incoming to the variable node is not really important since this should hardly ever happen if M is large enough. This is because if M is large, the quantized decoder will mimic more and more the min-sum decoder.

For future reference, we consider the ensemble with degree distribution $\big(\Lambda(x) = 0.4x^2 + 0.6x^5, \Gamma(x) = x^4\big)$. It has design rate $r = 0.05$. The Shannon threshold for this rate is $p^{\text{Sh}} = 0.369$. Table 5.1 shows the threshold values of this ensemble for increasing values of M as well as the threshold under true min-sum decoding. We see that the thresholds for finite M quickly converge to the unbounded case. Note that this quantizer and the message passing rules satisfy the symmetry conditions of [10]. Thus we can do the density evolution under the all-one codeword assumption. Recall that the alphabet has $2M + 1$ elements. since the probability of the individual elements sums up to one, we write the density evolution recursion G as a function of $2M$ variables. We write the probability of the message with largest value in terms of the probabilities of other messages. Thus, the underlying space is $X = \mathbb{R}^{2M}$ with $\mathbf{x} = \{\mathrm{P}(-M), \ldots, 0, \ldots, \mathrm{P}(M-1)\}$. As can be easily seen, the density evolution map is again a vector polynomial map. Thus such a map is completely continuous by Lemma 33 and Hypothesis 1 of Theorem 1 is satisfied. The first condition for the second hypothesis to hold true is that $G(p, \mathbf{0}) = \mathbf{0}$. Note that $\mathbf{x} = 0$ implies that with probability one, the message is equal to M. Now at the check node side if all the incoming messages are equal to M, then the outgoing is also equal to M. The same holds true for the variable node side by the definition of Ψ_v given in Eqn(5.5). Thus $G(p, \mathbf{0}) = \mathbf{0}$ holds true. Also the channel transition probability p appears only as p and $1 - p$. Thus the Frechet derivative of the map is of the form $pT + T'$, where both T and T' are elements of $\mathbb{R}^{2M \times 2M}$. In order to satisfy Hypothesis 2 of Theorem 1, we need to modify the density evolution map as the parameter p does not appear multiplicatively in the map $pT + T'$. To circumvent this problem, we use Lemma 35 and consider the derived map whose Frechet derivative is $p(I_{2M} - T')^{-1}T$. This satisfies the Hypothesis 2 of Theorem 1.

M	1	2	3	4	5	∞
$\approx p^*$	0.0319	0.0962	0.0974	0.1219	0.1318	0.148

Table 5.1: Thresholds of $\Lambda(x) = 0.4x^2 + 0.6x^5$, $\Gamma(x) = x^4$ under quantized min-sum decoding.

Example 2 (Min-Sum Decoder with $M = 2$)**.** *For our running example consider $M = 2$. The Frechet derivative is of the form $pT + T'$, where T' is not identically zero. Fortunately $(I_4 - T')^{-1}$ exists. As mentioned before, by Lemma 35 we need to study the eigenvalues of the matrix $(I_4 - T')^{-1}T$. The matrix $(I_4 - T')^{-1}T$ has eigenvalues $\frac{1}{\mu_1} = 3.50027$, $\frac{1}{\mu_2} = -2.70249$ and the other two eigenvalues are zero. Both $\frac{1}{\mu_1}$ and $\frac{1}{\mu_2}$ have multiplicity one (i.e., the multiplicities are odd). This implies that the KR theorem is applicable to both the eigenvalues and at least one of the conclusion of the KR theorem must hold true for both of them. In particular $(\mu_1, \mathbf{0})$ and $(\mu_2, \mathbf{0})$ are bifurcation points.*

Let C_{μ_1} and C_{μ_2} be the fixed point component containing μ_1 and μ_2 respectively. Now by the KR theorem either the fixed point connected component C_{μ_1} and C_{μ_2} are unbounded or $C_{\mu_1} = C_{\mu_2}$.

We can compute the fixed points explicitly in this case. The result is shown in Figure 5.4. Since the fixed points are elements of $\mathbb{R}^4$ we need to project them into $\mathbb{R}$ in order to be able to plot them. We choose to apply the error probability operator. As the density evolution is done assuming that the all-one codeword has been transmitted, so the error probability operator sums up the component corresponding to negative indices and adds to this sum half the weight of index zero as it is like an erasure.

$$P_e(\boldsymbol{x}) = \sum_{i=-M}^{-1} x_i + \frac{x_0}{2}. \tag{5.6}$$

As we can see, the second conclusion of Theorem 1 holds, i.e., $C_{\mu_1} = C_{\mu_2} = C_{\mu}$. The fixed point connected component C_μ containing the point a= $(\mu_1, \boldsymbol{0}) = (0.28569, \boldsymbol{0})$ also contains the point d= $(\mu_2, \boldsymbol{0}) = (-0.37003, \boldsymbol{0})$. In the component C_μ, the branch from a to b is stable, b to c is unstable and c to d is stable. The component C' is stable. The threshold is $p^* = 0.0962$. The fixed point of iterative decoder at the threshold is represented by point e of the fixed point component C'. Above the threshold, the fixed points of iterative decoder moves upward along C' as the channel transition probability p increases.

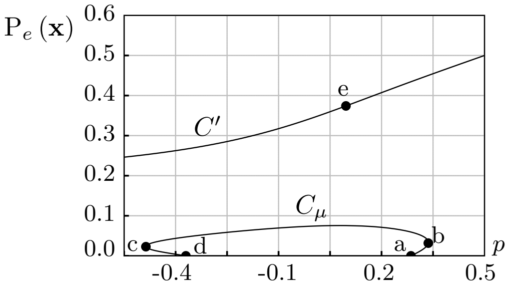


Figure 5.4: Fixed point components for 5 point quantizer.

Example 3 (Min-sum decoder with $M = 3$)**.** *For our running example we consider $M = 3$. The Frechet derivative is again of the form $pT + T'$. In this case also the inverse $(I_6 - T')^{-1}$ exists. By Lemma 35, we need to study the eigenvalues of $(I_6 - T')^{-1}T$. The matrix $(I_6 - T')^{-1}T$ has the only non-zero real eigenvalue as $\frac{1}{\mu} = 2.09804$ and its multiplicity is one. So the KR theorem is applicable in this case. Note that as there is only one non-zero eigenvalue, there can be at most one bifurcation point by Theorem 2. Thus the second conclusion of KR theorem can not be true. This implies that the first conclusion holds: there is an unbounded component C_μ of fixed point containing the bifurcation point μ. In this case also we can compute this component explicitly. As the fixed points are element of $\mathbb{R}^6$, in order to plot them we project them to one* *dimension by the error probability operator given in Eqn(5.6). The plot is shown in Figure 5.5. The bifurcation point is a=*$(\mu, 0.0) = (0.476636, 0.0)$*. As far as the stability of the fixed point in* C_μ *is concerned, the branch a to b is stable. The fixed points in branch b to c is unstable and from point c onwards the fixed points are stable. The point e represents the fixed point at which the iterative decoder get stuck at threshold* $p^* \approx 0.0974$*. Above the threshold, the fixed points of iterative decoder moves upward along* C_μ *as the channel transition probability p increases.*

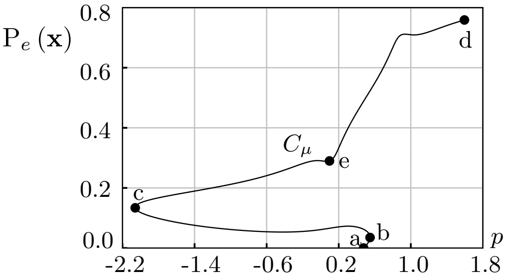


Figure 5.5: Fixed point components for 7 point quantizer.

Example 4 (Decoder with Erasure)**.** *The decoder with erasure was introduced in [10]. The underlying channel is the BSC(p). On the variable node side the message-passing rule for a node of degree* l *reads*

$$\Psi_v(m_0, m_1, \cdots, m_{l-1}) = \operatorname{sgn}\left(m_0 + \sum_{i=1}^{l-1} m_i\right).$$

The rule for a check node of degree r *is*

$$\Psi_c(m_1, \cdots, m_{r-1}) = \prod_{i=1}^{r-1} m_i.$$

Note that for this decoder if there are degree two variable nodes then the threshold is 0*, i.e.,* $\mathbf{0}$ *can not be a fixed point. To see this, suppose that all the incoming messages to variables nodes are equal to one. Then with probability* p*, the outgoing message from a variable node is equal to* 0*. Thus the probability of* 0 *is equal to* $\lambda_2 p$*. Hence we assume that* $\lambda_2 = 0$*. For this example* $M = 1$*, hence the underlying space is* $X = \mathbb{R}^2$*. The Frechet derivative of the density evolution map can again be computed and it turns out that its only eigenvalue is* $2\lambda_3\rho'(1)$*. But now this eigenvalue has even multiplicity. So we can not apply the KR theorem to this case. Using results of [50], it is possible to show that the conclusions of the KR theorem is still applicable to an eigenvalue of even multiplicity. Numerical computation of fixed point suggest that indeed* $\frac{1}{2\lambda_3\rho'(1)}$ *is a bifurcation point. For example, in Figure 5.6 we plot the fixed point component of* $(3, 6)$ *regular ensemble. For this ensemble* $2\lambda_3\rho'(1) = 10$*, so* $p = 0.1$ *is* *a bifurcation point. We can see from Figure 5.6 that point a which corresponds to $p = 0.1$ is indeed a bifurcation point. The threshold for this ensemble is $p^* = 0.0708$.*[1] *The point b represents the fixed point at which the decoder get stuck at the threshold. The branch a to b is unstable. From b onwards the fixed points are stable.*

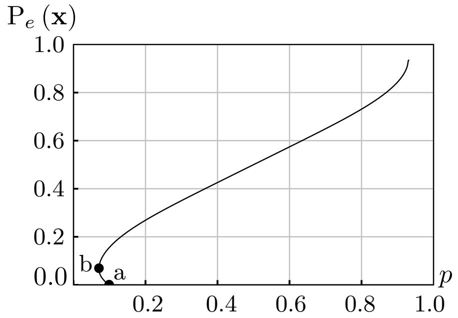


Figure 5.6: Fixed point component for the decoder with erasure for $(3, 6)$ LDPC ensemble.

5.4 Conclusion

We have shown how the tools of non-linear analysis can be used in proving the existence of fixed points. Our ultimate goal is to understand the fixed point structure of the BP and the min-sum decoder.

5.A Some Useful Lemmas

Lemma 33. *Every vector polynomial map $G : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N$ is a completely continuous map.*

Proof. Consider any bounded set S in $\mathbb{R}^N$. As S is bounded, so will be all the components $G_i(S)$. Hence the set $G(S)$ is also bounded. Clearly this would imply that the closure $\overline{G(S)}$ is also bounded. In a finite dimensional vector space a closed and bounded set is a compact. Hence $G(S)$ is relatively compact. Thus the map G is Completely continuous. □

Lemma 34. *Let $G : \mathbb{R}^N \to \mathbb{R}^N$ be a vector polynomial map such that $G(\boldsymbol{0}) = \boldsymbol{0}$. Then G is Frechet differentiable. The Frechet derivative T of G is a matrix whose entries are given by $\{t_{ij}\}$ where $1 \leq i, j \leq N$ and*

$$t_{ij} = \left.\frac{\partial G_i}{\partial x_j}\right|_{\boldsymbol{x}=\boldsymbol{0}}.$$

[1] This assumes that in the first iteration we set the weight of the channel to 2 and in all subsequent iterations to 1.

Proof. Consider $||G(\mathbf{x}) - T\mathbf{x}||$. As $|\mathbf{x}_i| \leq ||\mathbf{x}||$, there is no linear term in $G(\mathbf{x}) - T\mathbf{x}$ and $G(\mathbf{0}) = \mathbf{0}$ implies that $||G(\mathbf{x}) - T\mathbf{x}|| = o\left(||\mathbf{x}||^2\right)$. Hence

$$\frac{||G(\mathbf{x}) - T\mathbf{x}||}{||\mathbf{x}||} = o\left(||\mathbf{x}||\right).$$

This proves the lemma. □

Note that Hypothesis 2 of Theorem 1 implies that the parameter γ must appear multiplicatively in the Frechet derivative. But in many cases we see that the Frechet derivative is of the form $\gamma T + T'$. The following lemma says that in this case also the KR theorem can be applied provided the linear operator $I - T'$ is invertible.

Lemma 35. *Let $G : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ be a vector polynomial map and Frechet differentiable with Frechet derivative $\gamma T + T'$. Let us assume that $\left(I_n - T'\right)^{-1}$ exists. Let $F(\gamma, \boldsymbol{x}) \triangleq \left(I_n - T'\right)^{-1}(G(\gamma, \boldsymbol{x}) - T'\boldsymbol{x})$. Then F is a vector polynomial map and Frechet differentiable with Frechet derivative $\gamma\left(I_n - T'\right)^{-1} T$. Also the set of fixed points of F is same as set of fixed points of G.*

Proof. The fact that F is a vector polynomial map is obvious. For the Frechet differentiability of F we need that $F(\gamma, \mathbf{0}) = \mathbf{0}$. Now,

$$F(\gamma, \mathbf{0}) = \left(I_n - T'\right)^{-1}(G(\gamma, \mathbf{0}) - T'\mathbf{0}) = \mathbf{0},$$

as $G(\gamma, \mathbf{0}) = \mathbf{0}$. Now the Frechet derivative of $(G(\gamma,\ \mathbf{x}) - T'\mathbf{x})$ is given by $\gamma T\mathbf{x}$. This implies that the Frechet derivative of $F(\gamma, \mathbf{x})$ is equal to $\gamma\left(I_n - T'\right)^{-1} T$. To see that F and G have the same set of fixed points, let $\mathbf{x}$ be a fixed point of G. Then $G(\gamma, \mathbf{x}) - T'\mathbf{x} = \mathbf{x} - T'\mathbf{x}$ which implies $\left(I_n - T'\right)^{-1}(G(\gamma, \mathbf{x}) - T'\mathbf{x}) = \mathbf{x}$ i.e. $F(\gamma, \mathbf{x}) = \mathbf{x}$. □

6 Conclusion

In this thesis we have addressed three main problems. First we analyzed the performance of NBLDPC codes over BEC decoded via the BP and the MAP decoder. Then we considered the question of concentration of the weight distribution for regular binary LDPC ensembles. Finally, we address the problem of the existence of the EBP GEXIT function.

Let us now discuss some problems which we consider to be important in these set ups. For the BP decoding of NBLDPC codes over the BEC, an important question is whether is it possible to construct capacity achieving codes for a given degree distribution by suitably choosing the edge labels. An affirmative answer to this question will give an alternate way of achieving capacity. Another interesting problem is to find capacity achieving degree distributions for a fixed alphabet size. The difficulty is that the density evolution recursion is multi-dimensional. So, it is not amenable to analysis. However, finding capacity achieving degree distributions for NBLDPC codes for a given alphabet size yield degree distributions which have faster convergence to capacity compared to the binary capacity achieving degree distributions.

It will be of interest to generalize the analysis of the peeling decoder given in [7] to the setting of non-binary LDPC codes. Such an analysis will be very useful for the generalization of the scaling laws of [51] in the setting of non-binary LDPC codes. A generalization of the finite length optimization method of [52] to NBLDPC codes can give codes which have better error floor and waterfall performance. We observed that the Maxwell construction of [1] seems to hold in the setting of NBLDPC codes. It will be of interest to give a formal proof of the same.

For the problem of the concentration of the weight distribution, we have obtained partial results. It will be of importance to prove the concentration for every regular LDPC ensembles.

One of the most important open problem in iterative coding theory is to construct capacity achieving codes for general channels. It is believed that a better understanding of EBP GEXIT functions is a first step towards constructing capacity achieving codes for general BMS channels. Consequently, the existence of the EBP GEXIT function is of fundamental interest.

Bibliography

[1] C. Méasson, A. Montanari, and R. Urbanke. Maxwell's construction: The hidden bridge between maximum-likelihood and iterative decoding. submitted to IEEE Transactions on Information Theory, 2005.

[2] C. Méasson, A. Montanari, T.Richardson, and R. Urbanke. Maxwell's construction: The hidden bridge between maximum-likelihood and iterative decoding. submitted to IEEE Transactions on Information Theory, 2005.

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

[4] M. C. Davey and D. J. C. MacKay. Low density parity check codes over GF(q). *IEEE Communications Letters*, 2(6):165–167, jun 1998.

[5] X. Hu. *Low-Delay Low-Complexity Error-Correcting Codes on Sparse Graphs*. PhD thesis, EPFL, Lausanne, Switzerland, 2002.

[6] R. G. Gallager. *Information theory and reliable communication*. Wiley, 1968.

[7] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman. Efficient erasure correcting codes. *IEEE Trans. Inform. Theory*, 47(2):569–584, February 2001.

[8] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman. Improved low-density parity-check codes using irregular graphs. *IEEE Trans. Inform. Theory*, 47(2):585–598, 2001.

[9] A. Shokrollahi. New sequences of linear time erasure codes approaching the channel capacity. In *Proceedings of AAECC-13, Lecture Notes in Computer Science 1719*, number 1719 in Lecture Notes in Computer Science, pages 65–76. Springer Verlag, 1999.

[10] T. Richardson and R. Urbanke. The capacity of low-density parity check codes under message-passing decoding. *IEEE Trans. Inform. Theory*, 47(2):599–618, February 2001.

[11] S.-Y. Chung, Jr. 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*, 5(2):58–60, February 2001.

[12] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approaching irregular low-density parity-check codes. *IEEE Trans. Inform. Theory*, 47(2):619–637, February 2001.

[13] I. Sason and R. Urbanke. Parity-check density versus performance of binary linear block codes over memoryless symmetric channels. *IEEE Trans. Inform. Theory*, 49(7):1611–1635, July 2003.

[14] Jr. Forney, G. D. Codes on graphs: Normal realizations. *IEEE Trans. Inform. Theory*, 47(2):520–548, February 2001.

[15] D. Sridhara and T. E. Fuja. Low density parity check codes over groups and rings. In *Proc. of the IEEE Inform. Theory Workshop*, Banglore, India, October 2002. pp. 163-166.

[16] T. Richardson and V. Novichkov. Methods and apparatus for decoding LDPC codes. US Patent Number 6, 633, 856, oct 2003.

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

[18] J. Thorpe, K. Andrews, and S. Dolinar. Methodologies for designing LDPC codes using protographs and circulants. In *Proc. of the IEEE Int. Symposium on Inform. Theory*, Chicago, USA, June 2004. pp. 238.

[19] A. Bennatan and D. Burshtein. Design and analysis of nonbinary ldpc codes for arbitrary discrete-memoryless channels. *IEEE Trans. Inform. Theory*, 52(2):549–583, February 2006.

[20] D. Declercq and M. Fossorier. Decoding algorithms for nonbinary LDPC codes over GF(q). *IEEE Trans. Commun.*, 55(4):633–643, April 2007.

[21] I. Andriyanova and J. P. Tillich. A family of non-binary TLDPC codes: density evolution, convergence and thresholds. In *Proc. of the IEEE Int. Symposium on Inform. Theory*, Nice, France, July 2007.

[22] F. J. MacWilliams and N. J. Sloane. *The theory of error-correcting codes.* North-Holland, 1977.

[23] G. Strang. *Linear Algebra and Its Applications.* Saunders, 3 edition, 1988.

[24] D. J. C. MacKay and M. C. Davey. *Evaluation of Gallager Codes for Short Block Length and High Rate Applications*, volume 123 of *IMA Volumes in Mathematics and its Applications.* Springer, New York, 2000.

[25] T. Richardson and R. Urbanke. Thresholds for turbo codes. In *IEEE International Symposium on Information Theory*, page 317, Sorrento, Italy, June 2000.

[26] E. A. Bender and L. B. Richmond. Central and local limit theorems applied to asymptotic enumeration ii: multivariate generating functions. *J. Combin. Theory*, A 34(3):255–265, 1983.

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

[28] S. L. Litsyn and V. S. Shevelev. On ensembles of low-density parity-check codes: asymptotic distance distributions. *IEEE Trans. Inform. Theory*, IT–48(4):887 –908, April 2002.

[29] S. L. Litsyn and V. S. Shevelev. Distance distribution in ensembles of irregular low-density parity-check codes. *IEEE Trans. Inform. Theory*, IT–49(12):3140 –3159, Dec. 2003.

[30] G. Miller and D. Burshtein. Asymptotic enumeration method for analyzing LDPC codes. *IEEE Trans. Inform. Theory*, 50(6):1115–1131, June 2004.

[31] A. Vardy. The intractability of computing the minimum distance of a code. *IEEE Trans. Inform. Theory*, 43(6):1757–1766, November 1997.

[32] S. Condamin. Study of the weight enumerator function for a Gallager code. Project report, Cavendish Laboratory, University of Cambridge, July 2002.

[33] J. van Mourik, D. Saad, and Y. Kabashima. Critical noise levels for LDPC decoding. *Physical Review E*, 66(026705), 2002.

[34] 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*, page 102, Chicago, USA, June 27–July 2 2004.

[35] A. Montanari. The glassy phase of gallager codes. *Eur. Phys. J. B*, 23:121–136, 2001.

[36] O. Barak and D. Burshtein. Lower bounds on the spectrum and error rate of ldpc code ensembles. In *International Symposium on Information Theory*, Adelaide, Australia, 2002.

[37] O. Barak and D. Burshtein. Lower bounds on the spectrum and error rate of ldpc code ensembles. *IEEE Trans. Inform. Theory*, 53(11):4225–4236, November 2007.

[38] I. Goldenberg and D. Burshtein. Upper bound on error exponent of regular ldpc codes transmitted over the bec. 2008. submitted, arXiv:0803.2460.

[39] A. Bennatan. *The Application of LDPC Codes to New Problems in Communications.* PhD thesis, Tel Aviv University, 2006.

[40] F. Fagnani G. Como. Average spectra and minimum distances of low density parity check codes over cyclic groups. 2007. submitted.

[41] D. Burshtein and G. Miller. Bounds on the maximum-likelihood decoding error probability of low-density parity-check codes. *IEEE Trans. Inform. Theory*, 47(7):2696–2710, November 2001.

[42] P. Flajolet and R. Sedgewick. The average case analysis of algorithms: Saddle point asymptotics. Technical report, RR 2376, oct 1994.

[43] D. Gardy. Some results on the asymptotic behavior of coefficients of large powers of functions. *Discrete Mathematics*, 139:189–217, 1995.

[44] A. Orlitsky, K. Viswanathan, and J. Zhang. Stopping set distribution of ldpc code ensembles. *IEEE Trans. Inform. Theory*, 51(3):929–953, mar 2004. submitted.

[45] S. Janson. *The second moment method, conditioning and approximation*, volume 76 of *IMA Volumes in Mathematics and its Applications.* Springer, New York, 1996.

[46] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 195–248. Springer Verlag, New York, 1998.

[47] A. Ashikhmin, G. Kramer, and S. ten Brink. Extrinsic information transfer functions: model and erasure channel property. *IEEE Trans. Inform. Theory*, 50(11):2657–2673, November 2004.

[48] R. Brown. *A Topological Introduction to Nonlinear Analysis.* Birkhauser, 2003.

[49] S. Kesavan. *Nonlinear Functional Analysis A First Course.* Hindustan Book Agency, 2004.

[50] T. Ma and S. Wang. Bifurcation of nonlinear equations: I. steady state bifurcation. *Methods and Applications of Analysis*, 11:155–178, 2004.

[51] A. Amraoui, A. Montanari, T. Richardson, and R. Urbanke. Finite-length scaling for iteratively decoded ldpc ensembles. submitted to IEEE IT, June 2004.

[52] A. Amraoui, A. Montanari, and R. Urbanke. How to find good finite-length codes: From art towards science. *European Trans. Telecommunications*, 2006.

Curriculum Vitae

Vishwambhar Rathi

Address	**Status**
Ch. de la Cocarde, 21	Born 6^{th}/10/1980
Ecublens, 1024	in Neemuch, India
Lausanne, Switzerland	Citizen of India

EDUCATION

2003-2008 **École Polytechnique Fédérale de Lausanne (EPFL), Switzerland**
Ph.D. thesis: *"Non-Binary LDPC Codes and EXIT Like Functions"*,
under the supervision of Prof. Rüdiger Urbanke.

2002-2003 **Pre-Doctoral School at EPFL, Lausanne, Switzerland**
School of computer and communications sciences,

1998-2002 **Indian Institute of Technology (IIT), Bombay, India**
Bachelor of Technology (B. Tech) in Electrical Engineering.

EMPLOYMENT HISTORY

11/2003–06/2008 **Research and teaching assistant** at the communication theory laboratory, EPFL, Lausanne, Switzerland.
Research area: Analysis of iterative coding systems.

LANGUAGE SKILLS

Hindi, native
English, fluent
French, basic

PUBLICATIONS

Journals

[1] V. Rathi and R. Urbanke, *Density Evolution, Stability Condition, Thresholds for Non-Binary LDPC Codes*, IEE Communication Proceedings, Volume 152, Issue 6, Dec. 2005, May 2006.

[2] V. Rathi, *On the Asymptotic Weight and Stopping Set Distribution of Regular LDPC Ensembles*, IEEE Transactions on Information Theory, 52 (2006),

pp. 4212-4218.

[3] V. Rathi and I. Andriyanova, *MAP decoding performance of non-binary LDPC codes over the BEC*, to be submitted to IEEE Transactions on Communications.

Conferences

[1] V. Rathi and R. Urbanke, *Density Evolution for LDPC Codes for Higher Alphabets*, Winter School on Coding and Information Theory, 2005, Bratislava, Slovakia.

[2] V. Rathi, *On the Asymptotic Weight Distribution of Regulate LDPC Ensembles*, in Proc. IEEE International Symposium on Information Theory, Adelaide, Australia, 2005.

[3] V. Rathi and R. Urbanke, *Existence Proofs of Some EXIT Like Functions*, in Proc. IEEE International Symposium on Information Theory, Nice, France, 2007.

[4] K. Bhattad, V. Rathi and R. Urbanke, *Degree Optimization and Stability Condition for the Min-Sum Decoder*, in Proc. Information Theory Workshop, Lake Tahoe, USA, 2007.

[5] V. Rathi, *Conditional Entropy of Non-Binary LDPC Codes over BEC*, in Proc. International Symposium on Information Theory, Toronto, Canada, 2008.