WEIGHT DISTRIBUTIONS OF LDPC CODE ENSEMBLES: COMBINATORICS MEETS STATISTICAL PHYSICS

CHANGYAN DI [*], ANDREA MONTANARI [†], AND RÜDIGER URBANKE[‡]

Abstract. In this paper we compute the exponent of the weight distribution of Low-Density Parity-Check (LDPC) code ensembles through a statistical physics method and a combinatorics method. We show that the two approaches agree for regular LDPC codes. However, for irregular codes this is not necessarily the case.

Key words. weight distribution, low-density parity-check (LDPC) codes, message-passing algorithm, replica method.

1. Introduction.

The weight distribution is an important characterization of a code. In general though it is hard to compute. In fact, even the determination of the minimum distance is NP-complete [13]. In 1989, Sourlas showed that there is a strong connection between error-correcting codes and disordered spin models [10, 11]. This made it possible to apply the powerful methods of statistical physics to problems in coding, especially for Low-Density Parity-Check (LDPC) codes, which correspond to a disordered spin model on a diluted graph [1, 4, 7, 12].

For a code $\mathcal{G}$ of length N, we use $A(\mathcal{G},N\omega)$, called the weight enumerator function, to denote the number of code words with normalized weight ω. Let us consider the exponent $\frac{1}{N}\ln A(\mathcal{G},N\omega)$. We are interested in determing the exponent $W(\omega)$, if it exists, so that

$$\Pr\{|\frac{1}{N}\ln A(\mathcal{G},N\omega)-W(\omega)|>\delta\}\xrightarrow[N\to\infty]{}0$$

for any $\delta>0$, i.e., the exponent of a "typical" code. To this end, we define the two quantities[1]:

$$W_{\rm sp}(\omega):=\lim_{N\to\infty}\frac{1}{N}\mathbb{E}_{\mathcal{C}}[\ln A(\mathcal{G},N\omega)]$$
$$W_{\rm com}(\omega):=\lim_{N\to\infty}\frac{1}{N}\ln\mathbb{E}_{\mathcal{C}}[A(\mathcal{G},N\omega)]$$

If $W(\omega)$ exists, then $W(\omega)=W_{\rm sp}(\omega)$. The difference between $W_{\rm sp}$ and $W_{\rm com}$ is the order of the expectation and the logarithm. From Jensen's inequality, we know

$$W_{\rm sp}\le W_{\rm com},$$

i.e., $W_{\rm com}$ is an upper bound for $W_{\rm sp}$. The quantity $W_{\rm com}$ is easily computed combinatorically whereas the quantity $W_{\rm sp}$ is the object of interest in the replica method. Note that if $A(\mathcal{G},N\omega)$ is strongly concentrated around its mean then $W_{\rm com}=W_{\rm sp}=W(\omega)$. Indeed, for regular LDPC code ensembles, both Condamin [1] and Mourik et. al. [12] have shown, that $W_{\rm sp}=W_{\rm com}$. Here we show that in general (irregular code ensembles) this is not always the case.

The paper is organized in the following way: a brief introduction of LDPC codes is given in Section 2. In Section 3, the results for the exponent of the weight distribution are presented. An comparison is given in Section 4. Discussion and conclusion are given in Section 5.

[*]Swiss Federal Institute of Technology – Lausanne, LTHC-DSC, CH-1015 Lausanne, email:changyan.di@epfl.ch

[†]Laboratoire de Physique Théorique de l'Ecole Normale Superieure, 24, rue Lhomond, 75231 Paris CEDEX 05, France, email:Andrea.Montanari@lpt.ens.fr

[‡]Swiss Federal Institute of Technology – Lausanne, LTHC-DSC, CH-1015 Lausanne, email:rudiger.urbanke@epfl.ch

[1]Note that in statistical physics, $W_{\rm sp}$ is called the *quenched* average whereas $W_{\rm com}$ is called the *annealed* average.

2. LDPC Codes. LDPC code ensembles, originally discovered by Gallager [5], are usually defined in terms of ensembles of *bipartite graphs*. A graph consists of a set of *variable* nodes and a set of *check* nodes, together with edges connecting both sets. It gives rise to a code in the following way: a vector $(x_1,\cdots,x_N) \in \mathrm{GF}(2)^N$ is a code word if and only if for each check node the sum (modulo 2) of the values of its adjacent variable nodes is zero. The coordinates of a code word are indexed by the variable nodes $1,\cdots,N$.

An ensemble of bipartite graphs is defined in terms of a pair of *degree distributions*. A degree distribution $\gamma(x)$ is a real valued polynomial with non-negative coefficients and $\gamma(1)=1$. Associated with the ensemble is a degree sequence pair $(\lambda(x)=\sum_i \lambda_i x^{i-1}, \rho(x)=\sum_j \rho_j x^{j-1})$, shorthand (λ,ρ), where λ_i (ρ_j) gives the probability of an edge connecting to a degree i (j) variable (check) node. The design *rate* is defined as $r(\lambda,\rho) := 1-\frac{\int\rho}{\int\lambda}$. Given a pair (λ,ρ) of degree distributions and the block length N, an *ensemble* of bipartite graphs $\mathcal{C}_{\mathrm{LDPC}}(N,\lambda,\rho)$ is defined by running over all possible permutations of edges connecting variable and check nodes according to λ and ρ, respectively. One can convert (λ,ρ) into *node perspective* $(L(x)=\sum_i L_i x^i, R(x)=\sum_j R_j x^j)$ by defining $L_i := \frac{\lambda_i}{i\int\lambda}$ and $R_i := \frac{\rho_i}{i\int\rho}$. Each graph in $\mathcal{C}_{\mathrm{LDPC}}(N,\lambda,\rho)$ has NL_i variable nodes of degree i and $(1-r(\lambda,\rho))NR_j$ check nodes of degree j.

One can associate an LDPC code ensemble to a set of sparse parity-check matrices. Each column represents a variable node and each row represents a check node. Depending on the number of edges emanating from each node, the number of ones in a row or a column is determined. So a code ensemble $\mathcal{C}_{\mathrm{LDPC}}(N,\lambda,\rho)$ can be defined as a set of $M\times N$ binary matrices as follows:

$$\mathcal{C}_{\mathrm{LDPC}}(N,\lambda,\rho) = \{\mathcal{A}_{M\times N} : \prod_{i=1}^{M}\delta(\sum_{k=1}^{N} a_{ik} - C_i)\prod_{j=1}^{N}\delta(\sum_{l=1}^{M} a_{lj} - V_j) = 1\},$$

where C_i is the number of ones in the i-th row, i.e., the degree of i-th check node, and V_j is the number of ones in the j-th column, i.e., the degree of j-th variable nodes. $M=(1-r(\lambda,\rho))N$ is the number of check nodes. The distribution of C_i and V_i are determined by λ and ρ.

Note that there is a slight difference between the definition of code ensemble via graphs and via matrices. In the graph case, one allows multiple edges connecting the same variable and check node, unlike in the matrix case. Furthermore, in the language of spin models, one usually makes a transfer from $\mathrm{GF}(2)$ to the set $\{-1,+1\}$. So a vector $\mathbf{x}$=$(x_1,\cdots,x_N) \in \mathrm{GF}(2)^N$ is replaced by a spin configuration, $\sigma=(\sigma_1,\cdots,\sigma_N)$ such that $\sigma_i=(-1)^{x_i}$. Note that the modulo 2 operation is transformed into a product.

Therefore, the code word (valid spin configuration) constraint can be written as:

$$\sigma \text{ is a code word of } \mathcal{A}_{M\times N} \;\Leftrightarrow\; \sum_{i=1}^{M}(\prod_{j=1}^{N}\sigma_j^{a_{ij}} - 1) = 0.$$

Note that we use $\alpha(\mathbf{x})$, defined as $\frac{1}{N}\sum_{i=1}^{N} x_i$, to represent a normalized weight of a code word and $\omega(\sigma)$ for its corresponding spin configuration. It is easy to check that $\alpha(\mathbf{x}) = \frac{1-\omega(\sigma)}{2}$. Hereafter, we make no difference between codewords and valid spin configurations.

3. The Exponent of Weight distribution of LDPC Codes.

3.1. Statistical Physics Approach.

In Statistical Physics, the study of disordered materials has motivated the development of methods to analyze random ensembles. One standard (albeit not rigorous) tool is the so called *Replica Method* [6]. The basic idea, in the present context, is that $\ln A(N\omega)$ should concentrate in probability. One computes its expectation using the trick

$$\mathbb{E}[\ln A(N\omega)] = \lim_{n\to 0}\frac{\mathbb{E}[A^n(N\omega)]-1}{n}.$$

Here we omit the lengthy calculation and present the replica symmetric solution as follows:

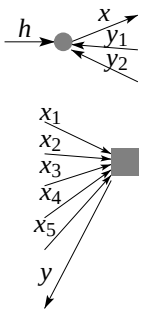
$$W_{\mathrm{sp}} = \sum_i L_i \int_i \cdots \int \ln\left(e^h \prod_{l=1}^{i}(1+y_{i_l}) + e^{-h}\prod_{l=1}^{i}(1-y_{i_l})\right)\prod_{l=1}^{i} dy_{i_l}\hat{\Pi}(y_{i_l}) - \frac{L'(1)}{R'(1)}\ln 2 - h\omega_{\mathrm{sp}}$$
$$+\frac{L'(1)}{R'(1)}\sum_j R_j \int_j \cdots \int \ln(1+\prod_{k=1}^{j} x_{j_k})\prod_{k=1}^{j}\Pi(x_{j_k})dx_{j_k} - L'(1)\iint \Pi(x)\hat{\Pi}(y)\ln(1+xy)dxdy \tag{3.1}$$

$$\omega_{sp} = \sum_i L_i \int_i \cdots \int \tanh\left(h + \sum_{k=1}^{i}\operatorname{atanh}(y_{i_k})\right)\prod_{k=1}^{i} dy_{i_k}\hat{\Pi}(y_{i_k})$$

where Π and $\hat{\Pi}$ are the solutions of

$$\begin{cases} \Pi(x) &= \sum_i \lambda_i \int_{i-1} \cdots \int \delta\left(x - \tanh(h+\sum_{k=1}^{i-1}\operatorname{atanh}(y_{i_k}))\right)\prod_{k=1}^{i-1}\int dy_{i_k}\hat{\Pi}(y_{i_k}) \\ \hat{\Pi}(y) &= \sum_j \rho_j \int_{j-1} \cdots \int \delta\left(y - \prod_{k=1}^{j-1} x_{j_k}\right)\prod_{k=1}^{j-1}\Pi(x_{j_k})dx_{j_k} \end{cases} \tag{3.2}$$

$\Pi(x)$ and $\hat{\Pi}(y)$ can be interpreted as the densities of messages from variable nodes to check nodes and from check nodes to variable nodes respectively. The corresponding Message-Passing Algorithm [8] is defined by the following updating rule as shown in Figure 1.



$$x = \tanh(h + \textstyle\sum_{k=1}^{i-1}\operatorname{atanh}(y_k))$$

$$y = \prod_{k=1}^{j-1} x_k$$

FIG. 1. *Updating rules (we omit the neighbourhood notation for clarity)*

Note that expressions similar to (3.1) and (3.2) have been derived in [4] (in the context of determing the LDPC code performance over symmetric channels) and in [1, 12] for the weight enumerator for *regular* Gallager codes.

3.2. Combinatorics Approach.

We know from [2] that

$$\mathbb{E}_{\mathcal{C}_{\mathrm{LDPC}}(n,\lambda,\rho)}[A(N\alpha)]$$
$$= \sum_k \mathrm{coeff}\left(\prod_i (1+yx^i)^{L_i N}, y^{N\alpha}x^k\right)\frac{\mathrm{coeff}\left(\prod_j\left(\frac{(1+x)^j+(1-x)^j}{2}\right)^{(1-r)NR_j}, x^k\right)}{\binom{\sum_i L_i i N}{k}} \tag{3.3}$$

To get W_{com} from (3.3) we use Hayman's and Stirling's formula. The details can be found in [3]. After a proper substitution of the variables, we can write W_{com} as follows

$$W_{\mathrm{com}} = \sum_i L_i \ln\left(e^h(1+\overline{y})^i + e^{-h}(1-\overline{y})^i\right) - \frac{L'(1)}{R'(1)}\ln 2 - h\omega_{\mathrm{com}}$$
$$+\frac{L'(1)}{R'(1)}\sum_j R_j \ln(1+\overline{x}^j) - L'(1)\ln(1+\overline{y}\,\overline{x}) \tag{3.4}$$
$$\omega_{\mathrm{com}} = \sum_i L_i \tanh(h + i\operatorname{atanh}(\overline{y}))$$

where $\overline{x}$ and $\overline{y}$ are defined by

$$\begin{cases} \overline{x} &= \sum_i \tilde{\lambda}_i \tanh(h+(i-1)\operatorname{atanh}(\overline{y})) \\ \overline{y} &= \sum_j \tilde{\rho}_j \overline{x}^{j-1} \end{cases} \tag{3.5}$$

and

$$\begin{cases} \tilde{\lambda}_i &= \dfrac{\lambda_i \frac{e^h(1+\overline{y})^{i-1}+e^{-h}(1-\overline{y})^{i-1}}{e^h(1+\overline{y})^{i}+e^{-h}(1-\overline{y})^{i}}}{\sum_k \lambda_k \frac{e^h(1+\overline{y})^{k-1}+e^{-h}(1-\overline{y})^{k-1}}{e^h(1+\overline{y})^{k}+e^{-h}(1-\overline{y})^{k}}} \\ \tilde{\rho}_j &= \dfrac{\frac{\rho_j}{1+\overline{x}^j}}{\sum_k \frac{\rho_k}{1+\overline{x}^k}} \end{cases}$$

We interpret (3.4) and (3.5) as follows: h determines the weight, while (3.5) is the result of both the consistence condition for the number of edges on both side of a bipartite graph and the maximization on the number of edges, more precise, the equation for $\overline{x}$ comes from the consistence in the number of edges (k in (3.3)), while the equation for $\overline{y}$ comes from the maximization over k.

Note the similarity of (3.4) and (3.5) with respective to their counterparts (3.1) and (3.2). A detailed discussion will be given in next section.

4. Compare and Contrast. Let's consider the case of regular codes. As shown in [1, 12], the solution for $\Pi(x)$ and $\hat{\Pi}(y)$ of (3.2) have the form of Delta functions as follows.

Fact 1 : For a regular LDPC code ensemble (x^{J-1}, x^{K-1}),

$$\begin{cases} \Pi(x) &= \delta(x-a) \\ \hat{\Pi}(y) &= \delta(y-a^{K-1}) \end{cases}$$

where a is defined by $a = \tanh(h+(J-1)\operatorname{atanh}(a^{K-1}))$.

Computing W_{sp} for this solution, we obtain exactly the same expression for W_{com} with $\overline{x} = a$ and $\overline{y} = a^{K-1}$. So for a regular LDPC code,

$$W_{\text{sp}} = W_{\text{com}}.$$

In Figure 2, we compare W_{sp} and W_{com} for a regular (x^2, x^5) code ensemble. We simply call it $(3,6)$ code ensemble. W_{sp} has been obtained numerically by means of a population dynamics approach to solve Π and $\hat{\Pi}$. W_{com} was found based on the numerical solution of (3.5). Furthermore, in Figure 3, for two different weights, we show the numerically obtained distributions of Π and $\hat{\Pi}$, which are indeed Delta functions.

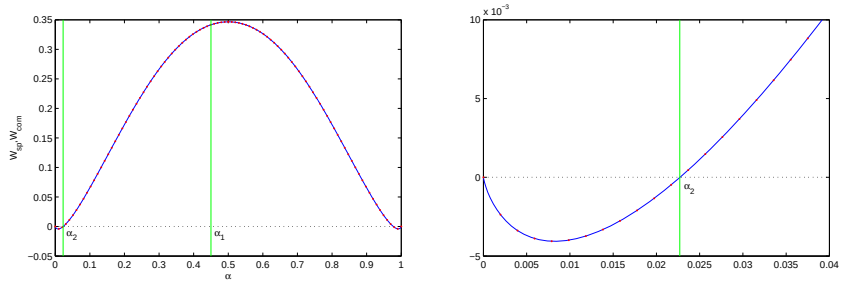


FIG. 2. *Exponent of the weight distribution for* $(3,6)$ *code, including* W_{sp} *(the dotted one),* W_{com} *(the continuous one). The right one is a zoomed-in version for weight close to* 0.

Note that for irregular LDPC codes, it is easy to check that Delta functions are no longer the solution for $\Pi(x)$ and $\hat{\Pi}(y)$. It is difficult to solve $\Pi(x)$ and $\hat{\Pi}(y)$ analytically. Nevertheless, we can obtain them numerically. Let's consider an irregular code ensemble with $(\lambda = 0.1x + 0.2x^2 +$

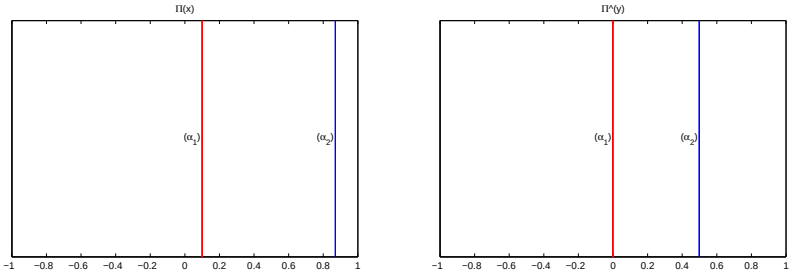


FIG. 3. *Two distribution function of Π and $\hat{\Pi}$ for $(3,6)$ code, one at weight $\alpha_1 = 0.45$, another at weight $\alpha_2 = 0.0227$ close to the minimum distance* 0.02273339.

$0.7x^3, \rho = 0.5x^3 + 0.5x^5)$. In Figure 4, we show the distribution of $\Pi(x)$ and $\hat{\Pi}(y)$ for two different weights $\alpha_1 = 0.45$ and $\alpha_2 = 0.10715$. Note that for the small weight 0.10715, the distributions of $\Pi(x)$ and $\hat{\Pi}(y)$ are quite different from Delta functions. while for the large weight 0.45, they are quite close to (but not equal to!) Delta functions. However, we see in Figure 5 that W_{sp} is not very sensitive to the shape of the distribtutions and therefore the difference between W_{sp} and W_{com} is small and of order 10^{-4} for small weights.

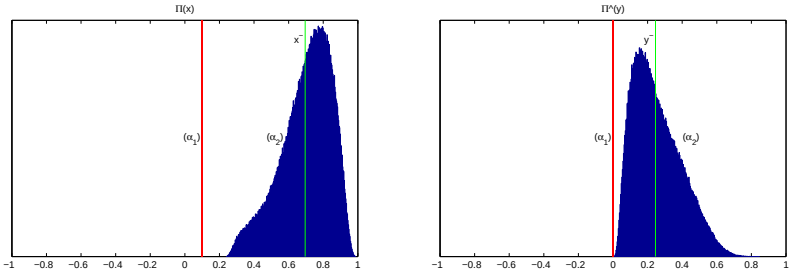


FIG. 4. *Two distribution function of Π and $\hat{\Pi}$ for the code ensemble with $(\lambda = 0.1x + 0.2x^2 + 0.7x^3, \rho = 0.5x^3 + 0.5x^5)$, one at weight $\alpha_1 = 0.45$, another at weight $\alpha_2 = 0.10715$ close to the minimum distance (*0.0103*). The corresponding $\overline{x}$ and $\overline{y}$ for α_2 are also plotted.*

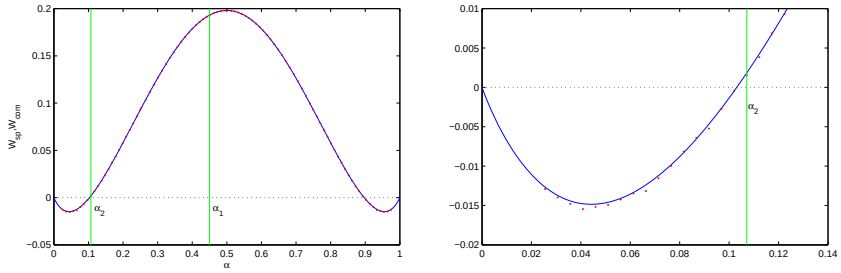


FIG. 5. *Exponent of the weight distribution for the code ensemble with $(\lambda = 0.1x + 0.2x^2 + 0.7x^3, \rho = 0.5x^3 + 0.5x^5)$, including W_{sp} (the dotted one), W_{com} (the continuous one). The right one is a zoomed-in version for weight close to* 0.

5. Discussion. What's the meaning of $\overline{x}$ and $\overline{y}$? Clearly, if $\Pi(x)$ and $\hat{\Pi}(y)$ are Delta functions, $\overline{x}$ and $\overline{y}$ are the averages of the edge messages from variable nodes to check nodes and from check nodes to variable nodes respectively. However, if $\Pi(x)$ and $\hat{\Pi}(y)$ are no longer Delta functions, are they still averages? One knows from the Fourier Transform of Density Evolution [9] that simple formulas can be obtained for two dual spaces in which update rules will be linear. If $\overline{x}$ and $\overline{y}$ are averages, they should be averages in these spaces. However, to connect these two spaces, the duality transform needs to be applied. This induces the modification of λ and ρ to $\tilde{\lambda}$ and $\tilde{\rho}$ depending on $\overline{x}$ and $\overline{y}$.

We conclude that for some cases, both methods give the same or very similar results. However, for other interesting cases, the results can be different. We have identified the origin of this difference in the shape of the message distributions. So far, we have not yet investigated in detail how much the replica symmetric assumption influences the result, instead we have concentrated on the comparison of the two presented methods. Future research will include a more detailed investigation of this issue. Furthermore, new applications of the method to other code properties will be investigated.

REFERENCES

[1] S. CONDAMIN, *Study of the weight enumerator function for a gallager code*, project report, Cavendish Laboratory, University of Cambridge, July 2002.

[2] C. DI, T. RICHARDSON, AND R. URBANKE, *Weight distribution of iterative coding systems: How deviant can you be?*, in International Symposium on Information Theory, Washington, D.C., June 2001, IEEE, p. 50.

[3] ———, *Weight distribution of iterative coding systems*. In Preparation, 2003.

[4] S. FRANZ, M. LEONE, A. MONTANARI, AND F. RICCI-TERSENGHI, *The dynaimc phase transition for decoding algorithms*, Physics Review E, 66 (2002).

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

[6] M. MEZARD, G. PARISI, AND M. A. VIRASORO, *Spin Glass Theory and Beyond*, World Scientific, Singapore, 1987.

[7] A. MONTANARI, *The glassy phase of gallager codes*, European Journal of Physics, 23 (2001).

[8] J. PEARL, *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann Publishers, 1988.

[9] T. RICHARDSON AND R. URBANKE, *The capacity of low-density parity-check codes under message-passing decoding*, IEEE Transactions on Information Theory, 47 (2001), pp. 599–618.

[10] N. SOURLAS, *Spin-glass models as error-correcting codes*, Nature, 339 (1989), pp. 693–695.

[11] ———, *Spin glasses, error-correcting codes and finite-temperature decoding*, Europhysics Letters, 25 (1994), pp. 159–164.

[12] S. VAN MOURIK, D. SAAD, AND Y. KABASHIMA, *Critical noise levels for low-density parity check decoding*, Physics Review E, 66 (2002).

[13] A. VARDY, *The intractability of computing the minimum distance of a code*, IEEE Trans. Inform. Theory, 43 (1997), pp. 1757–1766.