



# Noisy Density Evolution With Asymmetric Deviation Models

Elsa Dupraz, François Leduc-Primeau

## ► To cite this version:

Elsa Dupraz, François Leduc-Primeau. Noisy Density Evolution With Asymmetric Deviation Models. IEEE Transactions on Communications, 2021, Volume: 69 (Issue 3), pp.1403 - 1416. 10.1109/TCOMM.2020.3039856 . hal-03337227

**HAL Id: hal-03337227**

**<https://imt-atlantique.hal.science/hal-03337227>**

Submitted on 7 Sep 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Noisy Density Evolution With Asymmetric Deviation Models

Elsa Dupraz<sup>1</sup> and François Leduc-Primeau<sup>2</sup>

<sup>1</sup> IMT Atlantique, Lab-STICC, UMR CNRS 6285, France.

<sup>2</sup> Department of Electrical Engineering, Polytechnique Montreal, QC, Canada.

## Abstract

This paper considers low-density parity-check (LDPC) decoders affected by deviations introduced by the electronic device on which the decoder is implemented. Noisy density evolution (DE) that allows to theoretically study the performance of these LDPC decoders can only consider symmetric deviation models due to the all-zero codeword assumption. A novel DE method is proposed that admits the use of asymmetric deviation models, thus widening the range of faulty implementations that can be analyzed. DE equations are provided for three noisy decoders: belief propagation, Gallager B, and quantized min-sum (MS). Simulation results confirm that the proposed DE accurately predicts the performance of LDPC decoders with asymmetric deviations. Furthermore, asymmetric versions of the Gallager B and MS decoders are proposed to compensate the effect of asymmetric deviations. The parameters of these decoders are then optimized using the proposed DE, leading to better ensemble thresholds and improved finite-length performance in the presence of asymmetric deviations.

## I. INTRODUCTION

In most applications of error-correction codes, the implementation complexity of the decoder is a primary concern, and energy consumption is an important factor limiting the performance of the codes. Unfortunately, it is becoming increasingly difficult to improve the energy efficiency of integrated circuits while maintaining the abstraction that memory and computation circuits behave deterministically [1]. A growing body of work is therefore being devoted to the study of digital systems built out of unreliable circuit components. For instance, [2] considers fault-tolerant linear computing, [3] addresses logistic regression on unreliable hardware, [4] considers noisy binary recursive estimation, and [5] proposes to train deep neural networks for robustness to hardware faults. Error-correction codes are closely related to the development of such systems,

first because of the interest of developing more energy-efficient decoder implementations, but also because they can be used within a computing system to restore the fully-reliable operation abstraction when it might be required.

There has thus been significant interest in studying the operation of low-density parity-check (LDPC) decoders in contexts where values stored in memory and/or the result of computations can be affected by errors, or *deviations* [6]–[14]. Deviations in memory or computation circuits result from the difficulty or impossibility of predicting the variations in the physical properties of the circuit that occur at the time of fabrication or during operation of the system. Deviations can be prevented by operating the system based on the worst-case conditions, but this can be very costly in terms of energy consumption or performance. Instead, deviations can be modeled and their impact taken into account.

Density evolution (DE) [15], [16] is a powerful tool for the performance analysis of LDPC codes. DE consists of calculating the successive probability density functions of the messages exchanged in the iterative LDPC decoder, which permits to evaluate the decoder error probability under given channel conditions. For a given code degree distribution, DE thus allows to predict the *code ensemble threshold* as the worst channel parameter that allows for a vanishing decoding error probability over the ensemble of codes that follow the considered degree distribution. The threshold is evaluated assuming that the codeword length tends to infinity. In this paper, we are interested in analyzing the performance of decoders affected by deviations that can be asymmetric, in the sense that the probability that a logic 0 deviates to a logic 1 is not necessarily the same as the probability that a logic 1 deviates to a logic 0. When such asymmetric deviations are applied to the messages exchanged in an LDPC decoder, the symmetry of the decoder in the sense of [15] no longer holds. Standard DE [15], [16] requires symmetry assumptions on the channel, the variable node (VN) mapping, and the check node (CN) mapping of the decoder. Under the symmetry assumptions, it can be shown that the decoder error probability does not depend on the transmitted codeword. This allows to calculate the successive message probability density functions under the assumption that the all-zero codeword was transmitted, and greatly simplifies the calculation. The symmetry assumptions were removed in [17], [18], which consider non-symmetric channels, VN, and CN mappings. In [17], [18], the all-zero codeword assumption cannot be considered and the successive message probabilities are conditioned on the codeword

bits.

The literature on faulty LDPC decoders [6]–[14] assumes that deviations are introduced by the hardware in the VN and CN mappings. In this setup, noisy DE was introduced in [6] and latter considered for the analysis of bit-flipping decoders [8], [9], Gallager B decoders [7], [10], Min-Sum decoders [11]–[13], and finite-alphabet iterative decoders (FAIDs) [14]. Most of these works consider symmetric deviation models in addition to symmetric channels, VN, and CN mappings, and their DE analyses consider the all-zero codeword assumption. The only exception is [10], which considers non-symmetric deviation models in the Gallager B decoder. However, the DE expressions obtained in [10] depend on the codeword weight, which makes them difficult to manipulate. The theoretical part of [10] proposes approximations for the DE expressions. However, the proposed approximations only apply to the Gallager B decoder with hard-decision messages.

In this paper, we consider noisy DE with asymmetric deviation models and without the all-zero codeword assumption. The DE expressions we derive however do not depend on the codeword weight as in [10]. Instead, these expressions are conditioned on the codeword bit 0 or 1 at the VN, following the approach of [17] for standard (non-faulty) DE. This approach leads to DE expressions that are much simpler than in [10], and that can be applied to any type of LDPC decoder. We introduce the general noisy DE method under asymmetric deviation models, and then apply this method to three particular LDPC decoders: Belief Propagation (BP), Gallager B, and quantized Min-Sum (MS) decoder. For each considered decoder, we provide the DE equations under specific deviation models. In addition, in order to compensate asymmetric deviations in the decoder, we propose to introduce asymmetric parameters in the Gallager B decoder and in the quantized MS decoder. For the Gallager B decoder, we assume that the decision threshold at VNs is asymmetric and depends on the sign of the channel output. For the quantized MS decoder, we assume that the offset parameter and the scaling parameter are asymmetric, and depend on the sign of the CN messages and on the sign of the channel outputs, respectively. For the two decoders, we show that asymmetric decoder parameters allow to improve the decoders threshold as well as their finite-length performance. Finally, the main contributions of the paper can be summarized as follows:

- 1) We extend the asymmetric DE analysis without deviations of [17] in order to capture the effect of asymmetric deviations onto the LDPC decoders performance. For particular asymmetric deviation models of interest, we provide the DE equations for three decoders: BP, Gallager-B, and MS. The Gallager-B and MS have not been considered in [17], and [10] only proposed approximated DE equations for the Gallager-B decoder.
- 2) We propose to introduce asymmetric parameters onto Gallager-B and MS decoders, while in [10], only symmetric Gallager-B decoders were considered under asymmetric deviation models. We then show how to optimize these asymmetric parameters so as to compensate for asymmetric deviations introduced in the decoder.

The outline of the paper is as follows. Section II gives our assumptions and notations for LDPC decoders under asymmetric deviations. Section III introduces the noisy DE analysis under asymmetric deviations. Section IV provides the noisy DE equations for the three considered decoders. Section V describes asymmetric decoder parameters. To finish, Section VI provides simulation results.

## II. LDPC DECODERS

In this section, we first describe standard LDPC decoders without deviations. We then introduce the generic asymmetric deviation model we consider in this paper. Finally, we discuss practical decoder implementations to convey that the deviation models and the assumptions it contains are realistic.

### *A. Standard LDPC decoder without deviations*

We denote by  $\mathbf{x}^n$  a binary codeword of length  $n$ . The codeword  $\mathbf{x}^n$  is transmitted over a noisy independent and identically distributed (i.i.d.) channel  $\mathbb{P}(y|x)$  which outputs a vector  $\mathbf{y}^n$ . For simplicity, we assume that the considered channel  $\mathbb{P}(y|x)$  satisfies the symmetry conditions of [15] which can be stated as  $\mathbb{P}(y|x = 1) = \mathbb{P}(-y|x = 0)$ . We denote by  $\mathbf{H}$  the parity-check matrix of LDPC code, of size  $m \times n$  and rate  $R = m/n$ . In this paper, for simplicity, most of the DE analysis is described for regular LDPC codes with VN degree  $d_v$  and CN degree  $d_c$ , but we also show how to extend the DE analysis to irregular LDPC codes.

We consider an LDPC decoder which performs  $L$  iterations and which satisfies the extrinsic principle. The extrinsic principle means that when computing a message from VN  $v$  to CN  $c$ , we use all the messages incoming to VN  $v$ , except the one coming from CN  $c$ . In this decoder, the initial message incoming to a given VN is denoted  $\beta_0$ . At iteration  $\ell \in \{1, \dots, L\}$ , input CN messages for a given CN are denoted  $(\alpha_1^{(\ell-1)}, \dots, \alpha_{d_c-1}^{(\ell-1)})$  and output CN message is denoted  $\beta_{d_c}^{(\ell)}$ . The CN mapping is denoted  $\Phi_c$  with

$$\beta_{d_c}^{(\ell)} = \Phi_c(\alpha_1^{(\ell-1)}, \dots, \alpha_{d_c-1}^{(\ell-1)}). \quad (1)$$

In the same way, input VN messages for a given VN are denoted  $(\beta_1^{(\ell)}, \dots, \beta_{d_v-1}^{(\ell)})$  and the output VN message is denoted  $\alpha_{d_v}^{(\ell)}$ . The VN mapping is denoted  $\Phi_v$  with

$$\alpha_{d_v}^{(\ell)} = \Phi_v(\beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v-1}^{(\ell)}). \quad (2)$$

We also consider an *a posteriori* probability (APP) mapping  $\Phi_a$ . In the decoder, the decision on the bit values is taken as

$$\hat{x} = \begin{cases} 0 & \text{if } \Phi_a(\beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v}^{(\ell)}) > 0 \\ 1 & \text{if } \Phi_a(\beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v}^{(\ell)}) < 0 \end{cases}$$

In addition, if  $\Phi_a(\beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v}^{(\ell)}) = 0$ , then  $\hat{x}$  is sampled uniformly at random. The decoder stops after  $L$  iterations or if the stopping condition  $\mathbf{H}\hat{\mathbf{x}}^n = 0$  is satisfied. Although it is often used in LDPC decoder implementations, the stopping condition is not taken into account in the DE analysis.

The above mappings  $\Phi_v$ ,  $\Phi_c$ ,  $\Phi_a$ , give a generic description of a noiseless LDPC decoder which satisfies the extrinsic principle. We choose this description because in this paper, we consider several different decoders. In order to study a specific decoder such as Gallager B or MS, it suffices to replace the mappings  $\Phi_v$ ,  $\Phi_c$ ,  $\Phi_a$ , by the ones of the considered decoder.

### B. Asymmetric deviation model

We now describe our main assumptions for the asymmetric deviation models we consider in this paper. In the following, we denote by  $\tilde{\alpha}_i^{(\ell)}$  ( $i \in \{1, \dots, d_c-1\}$ ), and by  $\tilde{\beta}_j^{(\ell)}$  ( $j \in \{1, \dots, d_v-1\}$ ), the noisy versions of the messages  $\alpha_i^{(\ell)}$  and  $\beta_j^{(\ell)}$  from VN to CN and from CV to VN, respectively. Based on this notation, we first assume that hardware deviations are memoryless in the sense

that deviations introduced at iteration  $\ell$  are statistically independent of deviations introduced at previous iterations. This can be expressed as

$$\mathbb{P}(\tilde{\alpha}_i^{(\ell)} | \alpha_i^{(\ell)}, \tilde{\alpha}_i^{(\ell-1)}, \dots, \tilde{\alpha}_i^{(1)}) = \mathbb{P}(\tilde{\alpha}_i^{(\ell)} | \alpha_i^{(\ell)}), \quad (3)$$

$$\mathbb{P}(\tilde{\beta}_j^{(\ell)} | \beta_j^{(\ell)}, \tilde{\beta}_j^{(\ell-1)}, \dots, \tilde{\beta}_j^{(1)}) = \mathbb{P}(\tilde{\beta}_j^{(\ell)} | \beta_j^{(\ell)}) \quad (4)$$

We also assume that deviations are applied on the noiseless mapping outputs  $\alpha_{d_v}^{(\ell)}$  and  $\beta_{d_c}^{(\ell)}$ . More formally, this corresponds to assuming that

$$\mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \alpha_{d_v}^{(\ell)}, \beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v-1}^{(\ell)}) = \mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \alpha_{d_v}^{(\ell)}), \quad (5)$$

$$\mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \beta_{d_c}^{(\ell)}, \alpha_1^{(\ell)}, \dots, \alpha_{d_c-1}^{(\ell)}) = \mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \beta_{d_c}^{(\ell)}). \quad (6)$$

Expressions (5) and (6) mean that, given the knowledge of the noiseless output, the noisy output is statistically independent of the mapping inputs. As a result, the effect of deviations on the decoder is entirely represented by the conditional probability density function  $\mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \alpha_{d_v}^{(\ell)})$  and  $\mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \beta_{d_c}^{(\ell)})$ . Note that the probability density functions  $\mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \alpha_{d_v}^{(\ell)})$  and  $\mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \beta_{d_c}^{(\ell)})$  may vary from iteration to iteration. The above two assumptions are performed in most works on LDPC decoders with deviations, see [6], [8], [9], [11]–[14], and Section II-C provides additional justification based on hardware implementation considerations.

In addition, most existing works on faulty LDPC decoders [6], [8], [9], [11], [12], [14] assume symmetric error models for which

$$\mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \alpha_{d_v}^{(\ell)}) = \mathbb{P}(-\tilde{\alpha}_{d_v}^{(\ell)} | -\alpha_{d_v}^{(\ell)}) \text{ and } \mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \beta_{d_c}^{(\ell)}) = \mathbb{P}(-\tilde{\beta}_{d_c}^{(\ell)} | -\beta_{d_c}^{(\ell)}). \quad (7)$$

A different symmetry condition is provided in [13]. Assuming that the channel noise is represented by a vector  $\mathbf{z}^n$  such that  $y_i = x_i z_i$  for all  $i \in \{1, \dots, n\}$ , the symmetry condition of [13] is given by

$$\mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} | \mathbf{y}^n) = \mathbb{P}(x \tilde{\alpha}_{d_v}^{(\ell)} | \mathbf{z}^n) \text{ and } \mathbb{P}(\tilde{\beta}_{d_c}^{(\ell)} | \mathbf{y}^n) = \mathbb{P}(-x \tilde{\beta}_{d_c}^{(\ell)} | \mathbf{z}^n), \quad (8)$$

where  $x \in \{-1, 1\}$  is the codeword bit value associated to the VN that sends  $\tilde{\alpha}_{d_v}^{(\ell)}$  or receives  $\tilde{\beta}_{d_c}^{(\ell)}$ . The two models described by (7) and (8) give that the decoder error probability is independent of the transmitted codeword. This allows to consider that the all-zero codeword was transmitted in both the theoretical analysis and in the Monte Carlo simulations of the decoder.

Here, unlike in [6], [8], [9], [11]–[14], we do not perform any assumption on the symmetry of the probability density functions of  $\tilde{\alpha}_{d_v}^{(\ell)}$  and  $\tilde{\beta}_{d_c}^{(\ell)}$ . As a result, we cannot rely on the all-zero codeword assumption. In Section III, we propose a noisy DE analysis that allows to evaluate the performance of LDPC decoders under asymmetric deviation models without considering the all-zero codeword assumption.

### C. Faulty Decoder Implementations

To better place the proposed deviation model in context, let us briefly review some relevant aspects of a decoder implementation.

1) *Asymmetric deviations*: The need to study asymmetric deviations, that is deviations for which the probability of confusing a logic 0 for a 1 is different from the probability of confusing a logic 1 for a 0, arises from the fact that digital systems are commonly engineered from physical mechanisms that are asymmetric in nature. For instance, the vast majority of digital systems are currently built out of CMOS circuits. In CMOS, a logic gate is composed of PMOS devices used for pulling the output up to logic 1, and of NMOS devices used for pulling the output down to logic 0, but PMOS and NMOS devices have different physical properties and must be designed differently. The situation is common in data storage as well. An often used mechanism called *dynamic* memory consists in storing an electric charge to represent a logic 1, but removing the charge to represent a logic 0. This mechanism is also asymmetric in nature since a charge is more likely to leak than to spontaneously appear. For instance, embedded dynamic random-access memories (eDRAMs) are seen as a promising approach to increase storage densities of on-chip memories, but the retention time of a bit stored in an eDRAM cell can be very different depending on whether a 0 or a 1 is stored [19, Fig. 4]. Even *static* memories sometimes use *single-ended* (and thus asymmetric) designs to balance the various conflicting requirements of the design problem [20]. In all these cases, circuit designers usually aim to restore symmetry by careful design, but this comes at a cost. For instance, [21] shows that optimizing the sum of rise and fall times in a CMOS logic gate operated in the energy-efficient *subthreshold* regime is achieved with unequal rise and fall times. As a result, allowing timing violations in such a circuit would result in asymmetric deviations. Perhaps more importantly, as fabrication variations become increasingly problematic when designing energy-efficient circuits, symmetry is one of the engineered features of the system that will tend to break down.

2) *Designing a faulty implementation:* In order to use the proposed analysis to design a faulty decoder implementation, it is necessary to ensure that the assumptions of Section II-B are verified. At first, the memoryless assumptions (3), (4), might seem to be violated by the fact that deviations can be due to fabrication variations, which are permanent. But an important distinction must be made between physical circuits and decoder messages. Indeed, a different computing circuit can be used to perform the computations associated with a particular Tanner graph node at different iterations of the algorithm. Similarly, a message associated with a particular edge of the Tanner graph can be stored at different memory locations at different times.

Assumptions (3), (4) remain adequate in a decoder implementation affected by permanent faults if computations or message storage operations are realized by drawing at random a computation unit or a memory unit, as the case may be, from a large pool of such units. This behavior can be approximated in practice by enforcing in the implementation a *process diversity* policy, that is by ensuring through a combination of system architecture and control assignments that related parts of a computation are performed by different physical units. Fortunately, process diversity can often be provided easily in practical LDPC decoder architectures. For instance, in a parallel decoder architecture targeted at the widely used quasi-cyclic LDPC code family [22], messages are routed between two pools of processing units by applying rotations to vectors of messages. The rotation units could serve the additional purpose of assigning a given node to different computing or memory circuits in different iterations so as to approximate memoryless deviations.

On the other hand, depending on the type of deviation that is studied, there is no guarantee that (5) and (6) are verified. When they are not, they should be seen as a necessary approximation to make the DE tractable, with the redeeming factor that this approximation can be made as precise as desired since a different deviation distribution can be used for every DE iteration.

### III. NOISY DENSITY EVOLUTION WITH ASYMMETRIC DEVIATION MODELS

In this section, we propose a noisy DE analysis of LDPC decoders under asymmetric deviation models. In particular, we introduce the mathematical formalism that allows to perform DE without the all-zero codeword assumption. In our analysis, we consider the cycle-free assumption which is usually used to derive DE equations [15], [16]. This assumption implies that the messages

incoming to a VN or a CN are statistically independent.

#### A. Error probability evaluation

We now describe the DE analysis that allows to evaluate the message error probability of the decoder at successive iterations. We denote by  $P_x^{(\ell)}(\alpha)$  the probability density function of a noiseless message  $\alpha$  at a VN output, conditioned on the codeword bit  $x \in \{0, 1\}$  at this VN. We also denote  $Q_x^{(\ell)}(\beta)$  the probability density function of a noiseless message  $\beta$  at a CN output conditioned on the codeword bit  $x$  at the VN that receives the message  $\beta$ . The notations  $\tilde{P}_x^{(\ell)}(\tilde{\alpha})$  and  $\tilde{Q}_x^{(\ell)}(\tilde{\beta})$  hold for the probability density functions of noisy messages  $\tilde{\alpha}$  and  $\tilde{\beta}$ . Note that we use the term probability density function, although in the paper we consider both continuous-valued and discrete-valued decoder messages. In the latter case, the density function can be defined from the counting measure and reduces to a probability mass function. Here, unlike in [10], we consider conditioning on the bit value  $x$  rather than on the codeword weight  $w$ . This is because [17] shows that the two approaches are equivalent, while the terms  $P_0^{(\ell)}(\alpha)$ ,  $P_1^{(\ell)}(\alpha)$ , etc. are simpler to evaluate when conditioned on  $x$  rather than on  $w$ .

DE consists of calculating the probability density functions  $P_0^{(\ell)}(\alpha)$ ,  $P_1^{(\ell)}(\alpha)$ , etc., at successive iterations  $\ell$ . The commonly considered symmetry assumptions restated in Section II-B lead to the equalities  $P_0^{(\ell)}(\alpha) = 1 - P_1^{(\ell)}(\alpha)$ ,  $Q_0^{(\ell)}(\beta) = 1 - Q_1^{(\ell)}(\beta)$ , and the same holds for  $\tilde{P}_x^{(\ell)}(\tilde{\alpha})$  and  $\tilde{Q}_x^{(\ell)}(\tilde{\beta})$ . In standard DE, these equalities allow to consider the all-zero codeword assumption, and as a result, only the terms  $P_0^{(\ell)}(\alpha)$ ,  $Q_0^{(\ell)}(\beta)$  (for noiseless messages),  $\tilde{P}_0^{(\ell)}(\tilde{\alpha})$ ,  $\tilde{Q}_0^{(\ell)}(\tilde{\beta})$  (for noisy messages), are calculated. On the contrary, here, such equalities do not hold and it is required to calculate all the terms  $P_0^{(\ell)}(\alpha)$ ,  $P_1^{(\ell)}(\alpha)$ ,  $Q_0^{(\ell)}(\beta)$ ,  $Q_1^{(\ell)}(\beta)$ , and their noisy counterparts.

When the all-zero codeword assumption is removed, we can express the message error probability  $\tilde{p}_e^{(\ell)}$  of the decoder as follows. First denote

$$\tilde{P}^{(\ell)}(\tilde{\alpha}) = \frac{1}{2}\tilde{P}_0^{(\ell)}(\tilde{\alpha}) + \frac{1}{2}\tilde{P}_1^{(\ell)}(-\tilde{\alpha}). \quad (9)$$

In this expression, the probability density functions  $\tilde{P}_0^{(\ell)}(\tilde{\alpha})$  and  $\tilde{P}_1^{(\ell)}(-\tilde{\alpha})$  are weighted with a coefficient  $1/2$  because we assume that the transmitted codeword bits are equiprobable, which is the capacity-achieving distribution when the channel is symmetric. We then have that the

message error probability  $\tilde{p}_e^{(\ell)}$  in the noisy decoder at iteration  $\ell$  can be expressed as

$$\tilde{p}_e^{(\ell)} = \int_{-\infty}^{0-} \tilde{P}^{(\ell)}(\tilde{\alpha}) d\tilde{\alpha} + \frac{1}{2} \tilde{P}^{(\ell)}(0), \quad (10)$$

where  $\tilde{P}^{(\ell)}(0)$  is the probability that a noisy message  $\tilde{\alpha}$  takes value 0. In the above expression, for continuous-valued messages, the second term is often equal to 0, unless there is some mass point at 0 (this would be the case, for instance, for the Self-Corrected Min-Sum decoder introduced in [23]). For discrete-valued messages, (10) reduces to  $\tilde{p}_e^{(\ell)} = \sum_{\tilde{\alpha} < 0} \tilde{P}^{(\ell)}(\tilde{\alpha}) + \frac{1}{2} \tilde{P}^{(\ell)}(0)$ . Also note that  $\tilde{p}_e^{(\ell)}$  depends on the probability density function  $\tilde{P}^{(\ell)}(\tilde{\alpha})$  whose expression itself depends on the considered decoder. The expression of  $\tilde{P}^{(\ell)}(\tilde{\alpha})$  will be specified later in the paper for different decoders.

An expression similar to (10) was considered in [17], [24] in the case of an asymmetric communication channel, and asymmetric VN and CN mappings, but without deviations in the decoder. In the expression of  $\tilde{p}_e^{(\ell)}$  in (10), the second term comes from the case where the APP decision gives  $\Phi_a(\beta_0, \beta_1^{(\ell)}, \dots, \beta_{d_v}^{(\ell)}) = 0$ . In addition, considering  $\tilde{P}_1^{(\ell)}(-\tilde{\alpha})$  rather than  $\tilde{P}_1^{(\ell)}(\tilde{\alpha})$  in the definition of  $\tilde{P}^{(\ell)}(\tilde{\alpha})$  in (9) allows to have one single integral for the first term of  $\tilde{p}_e^{(\ell)}$ .

As a result, in order to evaluate the message error probability  $\tilde{p}_e^{(\ell)}$ , we need to express the probability density functions  $\tilde{P}_0^{(\ell)}(\tilde{\alpha})$  and  $\tilde{P}_1^{(\ell)}(\tilde{\alpha})$  for the two possible codeword bit values  $x = 0$  and  $x = 1$ . The expressions of these probability density functions depend on the considered LDPC decoder. In Section IV, we give their expressions for three decoders: BP, Gallager B, and MS.

### B. Threshold definition

The message error probability  $\tilde{p}_e^{(\ell)}$  defined in (10) is calculated under particular channel and deviation models. The code ensemble threshold [15] permits to evaluate the average decoder performance for the ensemble of all codes with regular degrees  $d_v$  and  $d_c$ . For decoders without deviations, the threshold is evaluated as the worst channel parameter that allows to have a vanishing error probability  $p_e^{(\ell)}$  when the codeword length tends to infinity [15].

For decoders with deviations, the standard threshold definition does not apply. Indeed, deviations usually prevent the decoder from reaching a zero error probability. Therefore, several alternative

threshold definitions were introduced for decoders with deviations [6], [14]. In this paper, we consider the definition of [6] which sets up a parameter  $\epsilon$  and defines the threshold as the worst channel parameter for which  $\tilde{p}_e^{(\ell)} < \epsilon$ . Although introduced in the case of symmetric deviation models, this definition still applies to the case of asymmetric deviations.

### C. Finite-length evaluation of the message error probability

The error probability  $\tilde{p}_e^{(\ell)}$  defined in (10) is evaluated under asymptotic conditions, that is assuming that the codeword length  $n$  goes to infinity. Alternatively, [25] describes a method for evaluating the error probability at finite length as follows.

We first describe the method of [25] for a binary symmetric channel (BSC) with parameter  $p_0$ . We denote by  $\tilde{p}_{e,n}^{(\ell)}$  the message error probability for a codeword length  $n$ . For a given value  $p_0$ , the error probability  $\tilde{p}_{e,n}^{(\ell)}(p_0)$  can be evaluated as

$$\tilde{p}_{e,n}^{(\ell)}(p_0) = \int_0^{1/2} \tilde{p}_e^{(\ell)}(z) \Phi_{\mathcal{N}}\left(z; p_0, \frac{p_0(1-p_0)}{n}\right) dz. \quad (11)$$

In this expression,  $\tilde{p}_e^{(\ell)}(z)$  is the asymptotic error probability given in (10), whose expression will be specified in Section IV, depending on the considered decoder. In addition,  $\Phi_{\mathcal{N}}$  gives the probability density function of a Gaussian random variable with mean  $p_0$  and variance  $\frac{p_0(1-p_0)}{n}$ . This expression is obtained by considering that the observed BSC parameter can be modeled as a Gaussian random variable [26, Chapter 8]. We can apply the same method in order to obtain the error probability  $\tilde{p}_{e,n}^{(\ell)}(\sigma^2)$  for an Additive White Gaussian Noise (AWGN) channel of variance  $\sigma^2$ . For this, it suffices to apply (11) with  $p_0 = \frac{1}{2} - \frac{1}{2}\text{erf}\left(\frac{1}{\sqrt{2\sigma^2}}\right)$ , where  $\text{erf}$  is the error function of the Gaussian distribution. In this case, in the integral in (11), the asymptotic error probability  $\tilde{p}_e^{(\ell)}(z)$  is evaluated for a channel variance  $v^2$  that depends on  $z$  and is given by  $v^2 = \frac{1}{2(\text{erf}^{-1}(1-2z))^2}$ .

This method allows to take into account the channel variations at finite-length. However, it still assumes that the code is cycle-free, and hence does not evaluate the effect of cycles onto the code performance. Since cycles are known to degrade the code performance at short length [27], the method presented in this section is well-suited to evaluate the finite-length performance for moderate to long lengths. We will use this method in our experiments in order to verify the

accuracy of the DE equations that we now introduce for three decoders: BP decoder, Gallager-B decoder, and quantized MS decoder.

#### IV. NOISY DENSITY EVOLUTION EQUATIONS

The DE methodology described in the previous section is generic and may be applied to any type of LDPC decoder and deviation model that satisfies the assumptions of Section II. In this section, we study three of the most common LDPC decoders: BP decoder, Gallager B decoder, and quantized MS decoder. For each considered decoder, we give examples of asymmetric deviation models and derive the corresponding noisy DE equations. In this part, we derive the DE equations for regular  $(d_v, d_c)$  LDPC codes for the three decoders. We then show how to extend the analysis to irregular LDPC codes.

##### A. BP decoder

We first consider the standard BP decoder with infinite precision on the messages. This first derivation of DE equations under asymmetric deviation models is mostly of theoretical interest since this decoder cannot be directly implemented in hardware. In the BP decoder, the messages are initialized with the channel log-likelihood ratios (LLRs) as  $\beta_0 = \log \frac{\mathbb{P}(X=0|y)}{\mathbb{P}(X=1|y)}$ . The VN and CN mappings (1) and (2) are as follows:

$$\alpha_{d_v}^{(\ell)} = \beta_0 + \sum_{j=1}^{d_v-1} \beta_j^{(\ell-1)} \quad (12)$$

$$\beta^{(\ell)} = g^{-1} \left( \sum_{i=1}^{d_c-1} g(\alpha_i^{(\ell)}) \right). \quad (13)$$

where, according to [16], the function  $g : \mathbb{R} \rightarrow \mathbb{R}^2$  is defined as  $g(x) = (\text{sgn}(x), -\log \tanh |\frac{x}{2}|)$ , and  $\text{sgn}(\cdot)$  is the sign function.

For the BP decoder, we can consider for instance the additive deviation model described in [6], [28]. In this model, noisy VN output messages  $\tilde{\alpha}_{d_v}^{(\ell)}$  are in the form

$$\tilde{\alpha}_{d_v}^{(\ell)} = \alpha_{d_v}^{(\ell)} + b^{(\ell)}, \quad (14)$$

where  $b^{(\ell)}$  is a continuous random variable that represents the noise, and the noise on CN output messages  $\beta_{d_c}^{(\ell)}$  is also additive, and as such can be combined with additive noise on VN output

messages without loss of generality since the VN mapping (12) is a sum. The random variable  $b^{(\ell)}$  is assumed independent of the noiseless message value  $\alpha_{d_v}^{(\ell)}$ . The probability density function  $B$  of  $b^{(\ell)}$  does not depend on  $\ell$ . For instance, [28] considers a zero-mean Gaussian distribution for  $b^{(\ell)}$ . However, here, unlike in [6], [28], no symmetry assumption is placed on  $B$ , which means that we might have  $B(\tilde{\alpha}) \neq B(-\tilde{\alpha})$  (for instance, a  $\chi^2$ -distribution).

We now give the DE equations for the BP decoder with the above additive deviation model, which generalize the asymmetric DE equations from [17] to handle deviations. First, the probability density functions of the initial messages  $\beta_0$  depends on the channel model. For instance, for an AWGN channel,  $\beta_0 = 2y/\sigma^2$ , where  $y$  is the channel output and  $\sigma^2$  is the channel variance. In this case, the probability density function  $P_x^{(0)}$  of  $\beta_0$  is a Gaussian distribution with mean  $2(1 - 2x)/\sigma^2$ , where  $x \in \{0, 1\}$ , and variance  $4/\sigma^2$ . Then, the probability density function of the noisy VN output messages  $\tilde{\alpha}$  can be expressed as

$$\tilde{P}_x^{(\ell)}(\tilde{\alpha}) = P_x^{(0)} \otimes (Q_x^{(\ell-1)})^{\otimes(d_v-1)} \otimes B(\tilde{\alpha}), \quad (15)$$

where  $\otimes$  represents the convolution product, and  $(\cdot)^{\otimes d}$  represents the power convolution operator. By convention,  $(\cdot)^{\otimes 0} = 1$ . Next, the probability density function of the noiseless CN output messages  $\beta$  is given by

$$Q_0^{(\ell)}(\beta) = \left(\frac{1}{2}\right)^{d_c-2} \sum_{v=0, \text{ even}}^{d_c-1} \binom{d_c-1}{v} \Gamma^{-1} \left( \Gamma \left( \tilde{P}_0^{(\ell)} \right)^{\otimes(d_c-1-v)} \otimes \Gamma \left( \tilde{P}_1^{(\ell)} \right)^{\otimes v} \right) \quad (16)$$

$$Q_1^{(\ell)}(\beta) = \left(\frac{1}{2}\right)^{d_c-2} \sum_{v=1, \text{ odd}}^{d_c-1} \binom{d_c-1}{v} \Gamma^{-1} \left( \Gamma \left( \tilde{P}_0^{(\ell)} \right)^{\otimes(d_c-1-v)} \otimes \Gamma \left( \tilde{P}_1^{(\ell)} \right)^{\otimes v} \right) \quad (17)$$

where  $\Gamma$  is the density transform operator of the function  $g$ , see [18]. The VN output messages probability density function (15) is obtained from the fact that the VN computation (12) is a sum of independent messages. For the derivation of the CN output message probability density function (16), see Appendix A. Finally, note that the density transform operator  $\Gamma$  and its inverse  $\Gamma^{-1}$  do not have known analytical expressions. Therefore, for the BP decoder, DE is usually evaluated either from message quantization [16] or from Monte-Carlo simulations [29].

### B. Gallager B decoder

We now consider the Gallager B decoder with a BSC of parameter  $p_0$  and under an asymmetric deviation model. The Gallager B decoder works with hard-decision messages  $\alpha, \beta \in \{0, 1\}$ . In

this decoder, the initial message  $\beta_0$  is equal to the channel output, that is  $\beta_0 = y$ , with  $y \in \{0, 1\}$ . The VN mapping is given by

$$\alpha_{d_v}^{(\ell)} = \begin{cases} \beta_0 \oplus 1 & \text{if } \left| \{d : \beta_d^{(\ell-1)} = \beta_0 \oplus 1\} \right| \geq b_\ell \\ \beta_0 & \text{otherwise.} \end{cases} \quad (18)$$

where  $b_\ell$  is a decoder parameter that depends on the iteration number  $\ell$  and can be optimized with DE. The CN mapping is given by

$$\beta_{d_c}^{(\ell)} = \bigoplus_{d=1}^{d_c-1} \alpha_d^{(\ell)}, \quad (19)$$

where  $\bigoplus$  denotes the XOR sum of the  $\alpha_d^{(\ell)}$ .

For the Gallager B decoder, we can consider the same deviation model as in [10] and apply it on the output CN messages  $\alpha_d$ . This model is described by two parameters  $\epsilon_{01}$  and  $\epsilon_{10}$  such that  $\epsilon_{01} = \mathbb{P}(\tilde{\beta}_d = 1 | \beta_d = 0)$  and  $\epsilon_{10} = \mathbb{P}(\tilde{\beta}_d = 0 | \beta_d = 1)$ , where we do not assume that  $\epsilon_{01} = \epsilon_{10}$ . Note that since the CN message computation (19) is a XOR sum, it is easy to show that deviations applied onto VN messages could be combined with deviations applied at the output CN messages. Therefore, the above deviation model which only considers some noise on output CN messages can be considered without loss of generality.

With this model, the DE equations of the Gallager B decoder are as follows. The probability density function of initial messages  $\beta_0$  is  $P_0^{(0)}(1) = p_0$  and  $P_1^{(0)}(1) = 1 - p_0$ . Then, the probability density functions of the noiseless CN output messages  $\beta_{d_c} \in \{0, 1\}$  are given by

$$Q_0^{(\ell)}(1) = \left(\frac{1}{2}\right)^{d_c-1} \sum_{v=0, \text{ even}}^{d_c-1} \binom{d_c-1}{v} \left(1 - \left(1 - 2P_0^{(\ell)}(1)\right)^{d_c-1-v} \left(1 - 2P_1^{(\ell)}(1)\right)^v\right) \quad (20)$$

$$Q_1^{(\ell)}(1) = \left(\frac{1}{2}\right)^{d_c-1} \sum_{v=1, \text{ odd}}^{d_c-1} \binom{d_c-1}{v} \left(1 - \left(1 - 2P_0^{(\ell)}(1)\right)^{d_c-1-v} \left(1 - 2P_1^{(\ell)}(1)\right)^v\right) \quad (21)$$

The derivation of these expressions can be done by following the same steps as for the BP decoder, see Appendix A. The probability density functions of the noisy CN output messages  $\tilde{\beta}_{d_v} \in \{0, 1\}$  can be expressed as

$$\tilde{Q}_x^{(\ell)}(0) = \epsilon_{10} Q_x^{(\ell)}(1) + (1 - \epsilon_{01}) Q_x^{(\ell)}(0). \quad (22)$$

From the condition  $\epsilon_{01} \neq \epsilon_{10}$ , we can show that  $\tilde{Q}_0^{(\ell)}(0) \neq 1 - \tilde{Q}_1^{(\ell)}(0)$  in general. The probability density functions of the noiseless VN output messages  $\alpha_d \in \{0, 1\}$  are given by

$$P_x^{(\ell)}(0) = P_x^{(0)}(0) \sum_{v=b_\ell}^{d_v-1} \binom{d_v-1}{v} \left(1 - \tilde{Q}_x^{(\ell)}(1)\right)^{d_v-1-v} \tilde{Q}_x^{(\ell)}(1)^v \\ + P_x^{(0)}(1) \left(1 - \sum_{v=b_\ell}^{d_v-1} \binom{d_v-1}{v} \left(1 - \tilde{Q}_x^{(\ell)}(1)\right)^v \tilde{Q}_x^{(\ell)}(1)^{d_v-1-v}\right), \quad (23)$$

and  $P_x^{(\ell)}(1) = 1 - P_x^{(\ell)}(0)$ . From these expressions, one can determine the code ensemble threshold and optimize the decoder parameter  $b_\ell$ .

Note that here, we considered messages in a binary alphabet  $\{0, 1\}$ . Therefore, the error probability calculation (10) cannot be applied. However, we can easily evaluate the Gallager B decoder error probability as

$$\tilde{p}_e^{(\ell)} = \frac{1}{2}P_0^{(\ell)}(1) + \frac{1}{2}P_1^{(\ell)}(0). \quad (24)$$

### C. Quantized Offset MS decoder

In this section, we consider a quantized offset MS decoder, which is often considered in practical hardware implementations [13]. For message quantization in the decoder, we consider a uniform quantizer on  $q$  bits, which gives  $2^q - 1$  quantization levels. The length of the quantization intervals is  $\mu$  and the message quantization alphabet is  $\mathcal{M} = \{-2^{q-1} + 1, \dots, 0, \dots, 2^{q-1} - 1\}$ . We denote by  $\mathcal{Q}(\cdot)$  the quantization function, with

$$\mathcal{Q}(m) \begin{cases} 2^{q-1} - 1 & \text{if } m > 2^{q-1} - 1 \\ -2^{q-1} + 1 & \text{if } m < -2^{q-1} + 1 \\ \mu \left\lfloor \frac{m}{\mu} + \frac{1}{2} \right\rfloor & \text{otherwise.} \end{cases} \quad (25)$$

For an AWGN channel with variance  $\sigma^2$ , the messages are initialized as  $\beta_0 = \mathcal{Q}(2\gamma y/\sigma^2)$ , where  $y$  is the channel output and  $\gamma$  is a scaling factor. The VN mapping is then given by

$$\alpha_{d_v}^{(\ell)} = \mathcal{Q}\left(\beta_0 + \sum_{d=1}^{d_v-1} \beta_d^{(\ell-1)}\right), \quad (26)$$

where the quantization function  $\mathcal{Q}$  only serves to perform message saturation. The CN mapping is given by

$$\beta_{d_c}^{(\ell)} = \left(\prod_{d=1}^{d_c-1} \text{sgn}(\alpha_d^{(\ell)})\right) \max\left(\min_d |\alpha_d^{(\ell)}| - \lambda, 0\right) \quad (27)$$

where  $\text{sgn}(\cdot)$  is the sign function, and  $\lambda$  is the decoder offset parameter.

Here, we consider a deviation model at the bit level for the quantized messages  $\alpha_{d_v}^{(\ell)}$ . We use  $(b_1, \dots, b_q)$  to represent the  $q$  bit values of a symbol  $m \in \mathcal{M}$ . Then, as for the Gallager B decoder described in Section IV-B, the deviation model is defined by the two parameters  $\epsilon_{01} = \mathbb{P}(\tilde{b}_j = 1 | b_j = 0)$  and  $\epsilon_{10} = \mathbb{P}(\tilde{b}_j = 0 | b_j = 1)$ , where  $\tilde{b}_j$  is the noisy version of  $b_j$ . For simplicity, we assume that the noise parameters  $\epsilon_{01}$  and  $\epsilon_{10}$  are the same for all bits  $b_j$ . In order to represent this deviation model for the quantized decoder, we can construct a probability transition matrix  $\Pi$  of size  $(2^q - 1) \times (2^q - 1)$ , as initially proposed in [14]. However, [14] requires symmetry conditions for  $\Pi$ , which are unnecessary here. With the above deviation model, the components of the matrix  $\Pi$  are such that

$$\Pi_{i,k} = \mathbb{P}(\tilde{\alpha}_{d_v}^{(\ell)} = k | \alpha_{d_v}^{(\ell)} = i),$$

with  $i, k \in \mathcal{M}$ .

For the defined decoder and deviation model, the DE equations are as follows. The initial message probability density functions  $P_x^{(0)}$  can be evaluated by using the erf function, the error function of the Gaussian distribution, and depend on the quantization intervals. The probability density function of the noiseless VN output messages  $\alpha_{d_v}^{(\ell)}$  is then given by

$$P_x^{(\ell)}(\alpha) = \sum_{\beta: \Phi_v(\beta) = \alpha} P_x^{(0)}(\beta_0) \prod_{d=1}^{d_v-1} Q_x^{(\ell)}(\beta_d), \quad (28)$$

where  $\beta = (\beta_0, \beta_1, \dots, \beta_{d_v-1})$ . Then, we use  $\mathbf{P}_x^{(\ell)} = [P_x^{(\ell)}(-2^{q-1} + 1), \dots, P_x^{(\ell)}(2^{q-1} - 1)]$  to denote the probability density function in vectorial form of the noiseless VN output messages  $\alpha_{d_v}^{(\ell)}$ . With this notation, the probability density function in vectorial form  $\tilde{\mathbf{P}}_x^{(\ell)}$  of the noisy VN output messages  $\tilde{\alpha}$  is given by

$$\tilde{\mathbf{P}}_x^{(\ell)} = \Pi \mathbf{P}_x^{(\ell)}. \quad (29)$$

Note that this expression only depends on the transition matrix  $\Pi$  and could therefore be applied to other deviation models. The probability density function of the noiseless CN output messages

$\tilde{\beta}$  is given by

$$Q_0^{(\ell)}(\beta) = \left(\frac{1}{2}\right)^{d_c-2} \sum_{v=0, \text{ even}}^{d_c-1} \binom{d_c-1}{v} \sum_{\alpha: \Phi_c(\alpha)=\beta} \prod_{d=1}^v \tilde{P}_1^{(\ell)}(\alpha_d) \prod_{d=v+1}^{d_c-1} \tilde{P}_0^{(\ell)}(\alpha_d) \quad (30)$$

$$Q_1^{(\ell)}(\beta) = \left(\frac{1}{2}\right)^{d_c-2} \sum_{v=1, \text{ odd}}^{d_c-1} \binom{d_c-1}{v} \sum_{\alpha: \Phi_c(\alpha)=\beta} \prod_{d=1}^v \tilde{P}_1^{(\ell)}(\alpha_d) \prod_{d=v+1}^{d_c-1} \tilde{P}_0^{(\ell)}(\alpha_d), \quad (31)$$

where  $\alpha = (\alpha_1, \dots, \alpha_{d_v-1})$ . By convention, in the above expressions, if  $v = 0$ , we set  $\prod_{d=1}^v \tilde{P}_1^{(\ell)}(\alpha_d) = 1$ , and if  $v = d_c - 1$ , we set  $\prod_{d=v+1}^{d_c-1} \tilde{P}_0^{(\ell)}(\alpha_d) = 1$ . The derivation of these expressions can be done by following the same steps as for the BP decoder, see Appendix A. From these expressions, one can determine the code ensemble threshold and optimize the decoder parameters  $\gamma$  and  $\lambda$ . Finally, the DE equations given in this section could be easily extended to the case where deviations appear at CN outputs, by considering a second transition matrix to be applied to a vector of CN output messages probabilities.

#### D. Extension to irregular LDPC codes

We now describe how to extend the proposed DE analysis to the case of irregular LDPC codes. We consider irregular LDPC codes with edge-perspective VN degree distribution  $\lambda(x)$  and with edge-perspective CN degree distribution  $\rho(x)$  [16]. Following the notation of [16], the VN degree distribution can be expressed as  $\lambda(x) = \sum_{i \geq 2} \lambda_i x^{i-1}$ , where  $\lambda_i$  is the fraction of edges emanating from a VN of degree  $i$ , and the CN degree distribution is given by  $\rho(x) = \sum_{j \geq 2} \rho_j x^{j-1}$ , where  $\rho_j$  is the fraction of edges emanating from a CN of degree  $j$ .

Then, for each of the three considered decoders, we denote by  $\tilde{P}_{x,i}^{(\ell)}(\tilde{\alpha})$  the probability density function of the noisy VN output messages, evaluated for  $d_v = i$ . For instance, for the BP decoder,  $\tilde{P}_{x,i}^{(\ell)}(\alpha)$  is calculated from (15) by setting  $d_v = i$ , but by still using  $Q_x^{(\ell)}$ . We also denote by  $Q_{x,j}^{(\ell)}(\beta)$  the probability density function of the CN output messages, evaluated with  $d_c = j$ , from  $\tilde{P}_x^{(\ell)}(\alpha)$ . Then, at iteration  $\ell$ , the probability density functions of decoder messages have expressions

$$\tilde{P}_x^{(\ell)}(\tilde{\alpha}) = \sum_{i \geq 2} \lambda_i \tilde{P}_{x,i}^{(\ell)}(\alpha) \quad (32)$$

$$Q_x^{(\ell)}(\beta) = \sum_{j \geq 2} \rho_j Q_{x,j}^{(\ell)}(\beta). \quad (33)$$

In the simulation section, we mainly focus on regular LDPC codes, but also evaluate the performance of a few irregular LDPC codes from the proposed DE analysis.

## V. ASYMMETRIC DECODER PARAMETERS

In this section, we want to optimize the decoder parameters under asymmetric error models. We focus on the two practical LDPC decoders that were introduced in Section IV: the Gallager B decoder and the offset MS decoder. We propose variants of these two decoders, in which the parameter  $b_\ell$  of the Gallager B decoder and the parameters  $\gamma, \lambda$  of the MS decoder are asymmetric in the sense that they now depend on the signs of the messages exchanged in the decoder. We show how to optimize these asymmetric parameters in order to improve the decoding performance under asymmetric error models.

### A. Asymmetric parameters in the Gallager B decoder

The standard Gallager B decoder described in Section IV-B depends on one parameter  $b_\ell$  which can vary from iteration to iteration. We now set two different parameters  $b_{0,\ell}$  and  $b_{1,\ell}$  and re-define the Gallager B VN update (18) as

$$\alpha_{d_v}^{(\ell)} = \begin{cases} 1 & \text{if } \beta_0 = 0 \text{ and } \left| \{d : \beta_d^{(\ell)} = 1\} \right| \geq b_{0,\ell}, \\ 0 & \text{if } \beta_0 = 1 \text{ and } \left| \{d : \beta_d^{(\ell)} = 0\} \right| \geq b_{1,\ell}, \\ \beta_0 & \text{otherwise.} \end{cases} \quad (34)$$

In this equation, the parameter  $b_{0,\ell}$  is associated to channel output  $\beta_0 = 0$  while the parameter  $b_{1,\ell}$  is associated to channel output  $\beta_0 = 1$ . The Gallager B CN update given in (19) does not change.

With this new VN update, the VN message probability density functions  $P_x^{(\ell)}(0)$  in (23) now depend on both parameters  $b_{0,\ell}$  and  $b_{1,\ell}$ , and become

$$\begin{aligned} P_x^{(\ell)}(0) = & P_x^{(0)}(0) \sum_{v=b_0}^{d_v-1} \binom{d_v-1}{v} \left(1 - \tilde{Q}_x^{(\ell)}(1)\right)^{d_v-1-v} \tilde{Q}_x^{(\ell)}(1)^v \\ & + P_x^{(0)}(1) \left(1 - \sum_{v=b_1}^{d_v-1} \binom{d_v-1}{v} \left(1 - \tilde{Q}_x^{(\ell)}(1)\right)^v \tilde{Q}_x^{(\ell)}(1)^{d_v-1-v}\right). \end{aligned} \quad (35)$$

The expressions of the other probability density functions  $\tilde{P}_x^{(\ell)}(0)$  and  $Q_x^{(\ell)}(1)$  do not change compared to Section IV-B.

We now propose a method to optimize the two parameters  $b_{0,\ell}$  and  $b_{1,\ell}$ . For this, we first extend the method of [10], [15] that only considers one single parameter  $b_{0,\ell} = b_{1,\ell} = b_\ell$ . Denote  $\mathcal{H}_x(b_{0,\ell}, b_{1,\ell}) = P_x^{(\ell)}(0)$ , where  $P_x^{(\ell)}(0)$  is evaluated with given parameters  $b_{0,\ell}$  and  $b_{1,\ell}$ . The objective is to find the smallest integers  $b_{0,\ell}$  and  $b_{1,\ell}$  such that

$$\mathcal{H}_0(b_{0,\ell} + 1, b_{1,\ell} + 1) \geq \mathcal{H}_0(b_{0,\ell}, b_{1,\ell}), \quad (36)$$

$$\mathcal{H}_1(b_{0,\ell} + 1, b_{1,\ell} + 1) \leq \mathcal{H}_1(b_{0,\ell}, b_{1,\ell}). \quad (37)$$

The first inequality comes from the fact that we want to maximize  $P_0^{(\ell)}(0)$  and the second one comes from the fact that we want to minimize  $P_1^{(\ell)}(0)$ . These two inequalities can be restated as

$$\frac{P_0^{(0)}(1)}{P_0^{(0)}(0)} \geq \frac{\binom{d_v-1}{b_{0,\ell}}}{\binom{d_v-1}{b_{1,\ell}}} \left( \frac{1 - \tilde{Q}_0^{(\ell)}(1)}{\tilde{Q}_0^{(\ell)}(1)} \right)^{d_v-1-b_{0,\ell}-b_{1,\ell}}, \quad (38)$$

$$\frac{P_1^{(0)}(1)}{P_1^{(0)}(0)} \leq \frac{\binom{d_v-1}{b_{0,\ell}}}{\binom{d_v-1}{b_{1,\ell}}} \left( \frac{1 - \tilde{Q}_1^{(\ell)}(1)}{\tilde{Q}_1^{(\ell)}(1)} \right)^{d_v-1-b_{0,\ell}-b_{1,\ell}}. \quad (39)$$

In addition, in order to take into account asymmetric deviations, we introduce a third condition, that is that we would like to find parameters  $b_{0,\ell}, b_{1,\ell}$  that minimize the gap

$$|P_0^{(\ell)}(1) - P_1^{(\ell)}(0)| \quad (40)$$

between  $P_0^{(\ell)}(1)$  and  $P_1^{(\ell)}(0)$ . This third condition aims to reduce the asymmetry that is introduced between  $P_x^{(\ell)}(0)$  and  $P_1^{(\ell)}(0)$  by the deviations, see (22). At the end, at each iteration  $\ell$ , we propose to select the two parameters  $b_{0,\ell}$  and  $b_{1,\ell}$  as follows:

- 1) From parameters  $b_{0,\ell-1}$  and  $b_{1,\ell-1}$  retained at iteration  $\ell - 1$ , and from the corresponding probability density functions  $P_x^{(\ell-1)}$  and  $\tilde{Q}_x^{(\ell-1)}$ , we evaluate the probability density functions  $\tilde{Q}_x^{(\ell)}$  for all possible pairs of values  $(b_{0,\ell}, b_{1,\ell}) \in \{\lceil \frac{d_v}{2} \rceil, \dots, d_v - 1\}^2$ . We then compute the two sides of the two inequalities (38) and (39) for all pair of values  $(b_{0,\ell}, b_{1,\ell})$ .
- 2) We then identify the pairs of parameters  $(b_{0,\ell}, b_{1,\ell})$  that satisfy both (38) and (39).
- 3) We retain the pairs of parameters  $(b_{0,\ell}, b_{1,\ell})$  that minimize the sum  $b_{0,\ell} + b_{1,\ell}$ .
- 4) If there is more than one pair that achieves the minimum, we select the pair that minimizes  $|P_0^{(\ell)}(1) - P_1^{(\ell)}(0)|$ .

In our experiments, we show that considering two parameters with the above optimization method allows to obtain better decoding performance than when considering one single parameter.

### B. Asymmetric offset parameters in the quantized MS decoder

We now consider the quantized MS decoder introduced in Section IV-C, and propose to use two scaling parameters  $\gamma_0, \gamma_1$ , and two offset parameters  $\lambda_0$  and  $\lambda_1$ . The messages are now initialized by using the two scaling parameters as

$$\beta_0 = \begin{cases} \mathcal{Q}(\gamma_0 y) & \text{if } y \geq 0, \\ \mathcal{Q}(\gamma_1 y) & \text{if } y < 0, \\ 0 & \text{otherwise.} \end{cases} \quad (41)$$

The VN mapping is still given by (26), but the CN mapping is now

$$\beta_{d_c}^{(\ell)} = \begin{cases} \left( \prod_{d=1}^{d_c-1} \text{sgn}(\alpha_d^{(\ell)}) \right) \max \left( \min_d |\alpha_d^{(\ell)}| - \lambda_0, 0 \right) & \text{if } \prod_{d=1}^{d_c-1} \text{sgn}(\alpha_d^{(\ell)}) > 0, \\ \left( \prod_{d=1}^{d_c-1} \text{sgn}(\alpha_d^{(\ell)}) \right) \max \left( \min_d |\alpha_d^{(\ell)}| - \lambda_1, 0 \right) & \text{if } \prod_{d=1}^{d_c-1} \text{sgn}(\alpha_d^{(\ell)}) < 0, \\ 0 & \text{otherwise.} \end{cases} \quad (42)$$

As a result, the scaling parameter now depends on the sign of the output channel value, and the offset now depends on the sign of the output message.

In order to take these asymmetric parameters into account, the DE equations given in Section IV-C, are modified as follows. The probability density functions  $P_x^{(0)}$  of the initial messages can be evaluated from the quantization intervals and from the cumulative distribution function provided in Appendix B. The probability density functions  $Q_x^{(\ell)}$  of the CN messages are still evaluated from (30) and (31), except that now the expression of the function  $\Phi_c$  is given by (42) and depends on the two offset values. Finally, the probability density functions  $P_x^{(\ell)}$  and  $\tilde{P}_x^{(\ell)}$  of the VN messages without and with deviations do not change and are still evaluated from (28) and (29).

Finally, in order to optimize both offset and scaling parameters, we perform an exhaustive search in order to select the four values of  $\gamma_0, \gamma_1, \lambda_0$  and  $\lambda_1$  that minimize the decoder error probability  $\tilde{p}_e^{(\ell)}$  evaluated with asymptotic DE for given channel and deviation conditions. Simulation results show the gain in performance obtained with these asymmetric parameters.

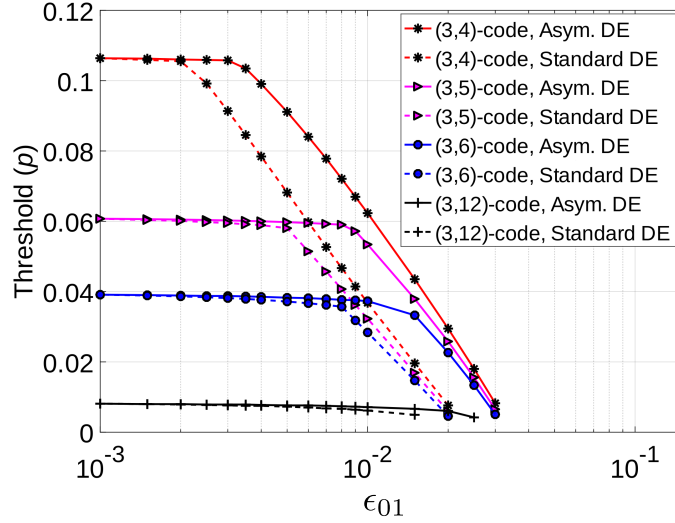


Fig. 1. Regular code ensemble thresholds for  $\epsilon_{10} = 10^{-3}$  with respect to  $\epsilon_{01}$  for the Gallager B decoder, without the all-zero codeword assumption (standard DE) and with the all-zero codeword assumption (Asym. DE).

## VI. NUMERICAL RESULTS

This section provides simulation results for the Gallager B decoder and for the quantized Min-Sum decoder. For both decoders, it gives threshold values obtained from the noisy DE analysis presented in this paper for various channel and asymmetric deviation models. It also compares the decoders finite-length performance predicted from noisy DE by using the method described in Section III-C, with the performance evaluated from Monte Carlo simulations. Finally, it evaluates the effect of asymmetric parameters introduced in Section V.

In the Monte-Carlo simulations performed in this section, we do not consider the all-zero codeword. Instead, we generate information sequences at random and perform the encoding with the generator matrix, see [30].

### A. Gallager B decoder

In this section, we first provide the code ensemble thresholds obtained for a Gallager B decoder under asymmetric deviations. For this decoder, we assume a BSC with crossover probability  $p$ . Following the definition in Section III-B, we measure the threshold as the largest value of  $p$  for which  $\tilde{p}_e^{(\ell)} < 10^{-3}$ , where  $\tilde{p}_e^{(\ell)}$  is expressed in (24) with respect to the probability density

functions  $\tilde{P}_x^{(\ell)}$  obtained by DE, and where  $p$  is found using a binary search. We consider four regular codes with  $d_v = 3$  and  $d_c = 4, 5, 6, 12$ , respectively, and parameters  $L = 200$  iterations,  $b_\ell = 2$ , and  $\epsilon_{10} = 10^{-3}$ . From the DE equations provided in Section IV-B, we calculate the thresholds for the four considered codes. In Figure 1, we show the thresholds for different values of  $\epsilon_{01}$ , obtained with the asymmetric DE presented in this paper, and also the thresholds obtained from standard noisy DE [6] performed with the all-zero codeword assumption. The thresholds for standard noisy DE were evaluated by calculating only the probability density functions  $Q_0^{(\ell)}$ ,  $\tilde{Q}_0^{(\ell)}$ , and  $P_0^{(\ell)}$  in (21), (22), (23), respectively, and  $Q_0^{(\ell)}$  was calculated by considering that  $P_0^{(\ell)} = 1 - P_1^{(\ell)}$ , as in standard DE. As a result, the curves for standard DE were evaluated for the same values  $\epsilon_{01} \neq \epsilon_{10}$  as in asymmetric DE. As expected, in all the considered cases, we observe from Figure 1 that the thresholds decrease with  $\epsilon_{01}$ . In addition, we observe that the thresholds obtained under the all-zero codeword assumption differ from the thresholds obtained without this assumption, which shows the need to remove the all-zero codeword assumption in DE under asymmetric deviations. Nonetheless, for small values of  $\epsilon_{01}$ , the two threshold values are the same. This comes from the fact that for a small amount of deviations, the noisy thresholds become equal to the noiseless threshold.

Then, in order to verify the accuracy of the proposed asymmetric DE, we use the method described in Section III-C that allows to predict the decoder performance at finite-length from DE. We construct one regular  $(3, 4)$ -code, one regular  $(3, 6)$ -code, and one regular  $(3, 12)$ -code, both of length  $N = 10000$ , with a Progressive-Edge-Growth (PEG) algorithm [27]. The  $(3, 4)$  and  $(3, 6)$  code both have girth 10, while the  $(3, 12)$ -code has girth 6. We then fix deviation parameters  $\epsilon_{01} = 10^{-2}$ ,  $\epsilon_{10} = 10^{-4}$ . We evaluate the Bit Error Rate (BER) performance of the two codes from Monte Carlo simulations, and compare the obtained BERs to the ones predicted by the finite-length DE-based method described in Section III-C. We apply the method of Section III-C with and without the all-zero codeword assumption. Figure 2 shows the obtained BERs with respect to the crossover parameter  $p$ . We see that our DE analysis without all-zero codeword assumption accurately predicts the decoder performance of the finite-length codes. On the contrary, we observe a gap between the performance predicted with the all-zero codeword assumption, and the BERs obtained from Monte Carlo simulations. This shows the accuracy and the interest of the method proposed in this paper. Finally, Figure 2 shows that an error floor

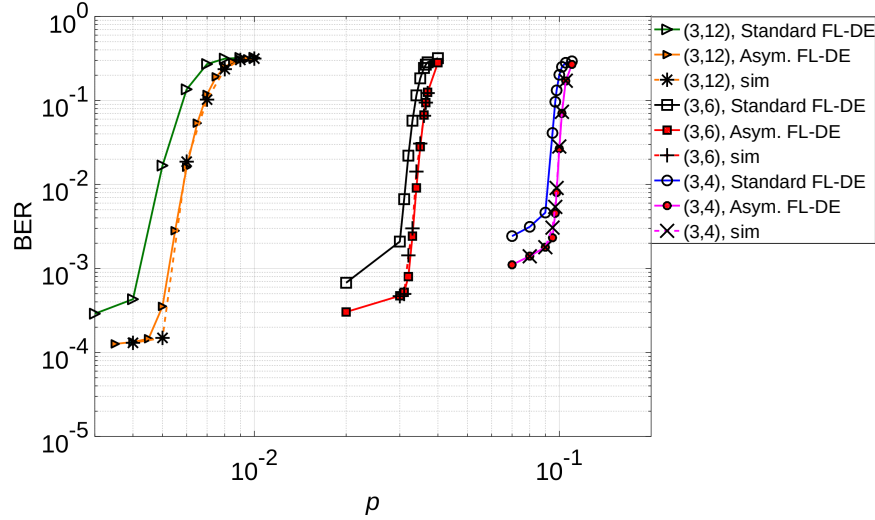


Fig. 2. For the Gallager-B decoder, comparison of BER measured from Monte-Carlo simulations and evaluated with the finite-length DE-based method (FL-DE), with and without the all-zero codeword assumption, for (3,4), (3,6), and (3,12) regular codes of length  $N = 10000$ , with  $\epsilon_{01} = 10^{-2}$ ,  $\epsilon_{10} = 10^{-4}$ . For each considered setup, the curves of Monte Carlo simulations and of FL-DE without the all-zero codeword assumption are superimposed.

appears for both codes for low values of  $p$ . This error floor is due to deviations introduced in the decoder, which DE can capture, and not to short-length effects such as Tanner graph cycles.

Finally, we evaluate the performance of the Gallager B decoder with asymmetric parameters introduced in Section V, both from threshold computation and from finite-length performance analysis. For threshold comparison, as before, we set  $L = 200$  iterations and  $\epsilon_{10} = 10^{-3}$ . We consider three regular (5,6), (9,10), and (5,10) codes. For each code, we consider two Gallager B decoders: the first one with a symmetric parameter  $b_\ell$  (optimized with the method of [15]), and the second one with two asymmetric parameters  $b_{0,\ell}$  and  $b_{1,\ell}$  (optimized with the method of Section V). Figure 3 (a) shows the thresholds obtained for the two methods for various values of  $\epsilon_{01}$ . We observe that asymmetric parameters improve the decoder thresholds compared to symmetric parameters.

We also confirm these results on finite-length simulations. For the (5,6) and the (5,10) code, we compare the BER performance obtained with both symmetric and asymmetric decoder

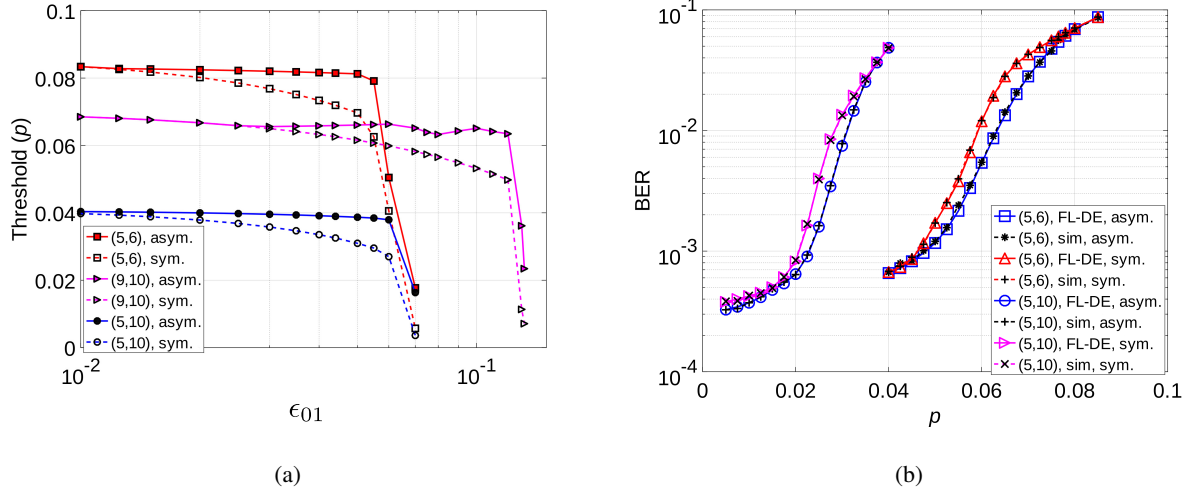


Fig. 3. (a) Threshold optimization with both symmetric and asymmetric Gallager B decoder parameters, for (5,6), (9,10), and (5,10) regular codes, and for  $\epsilon_{10} = 10^{-3}$ . The  $x$ -axis label  $\epsilon_{01}$  is the deviation parameter from bit value 0 to bit value 1. (b) Finite-length performance of (5,6) and (5,10)-codes of length  $N = 10000$  for the Gallager B decoder with symmetric and asymmetric parameters, with  $\epsilon_{01} = 5 \times 10^{-2}$ ,  $\epsilon_{10} = 10^{-4}$ . The finite-length performance is measured from Monte-Carlo simulations and evaluated from the finite-length DE-based method (FL-DE) without the all-zero codeword assumption. For each considered setup, the curves of Monte Carlo simulations and of FL-DE are superimposed.

parameters. In both cases, we predict the BER from the finite-length DE-based method of Section III-C, and evaluate the BER from Monte Carlo simulations on codes of length  $N = 10000$  and girth 8. The results are shown in Figure 3 (b) for  $\epsilon_{01} = 5 \times 10^{-2}$ ,  $\epsilon_{10} = 10^{-4}$ , and  $L = 5$  iterations. We observe that asymmetric parameters clearly improve the BER performance of the Gallager B decoder under asymmetric deviations. We also see that the method of Section III-C accurately predicts the BER performance at finite length, which validates the asymmetric DE analysis introduced in this paper.

### B. Quantized Min-Sum decoder

We now consider the quantized Min-Sum decoder with the asymmetric deviation model described in Section IV-C. We consider an AWGN channel with variance  $\sigma^2$  and normalized Signal-to-Noise Ratio (SNR)  $\text{snr} = 10 \log_{10} \left( \frac{1}{2R\sigma^2} \right)$  in dB. In this case, the code ensemble threshold is defined as the smallest value of  $\text{snr}$  for which  $\tilde{p}_e^{(\ell)} < 10^{-3}$  for a large enough  $\ell$ . For the decoder parameters, we fix  $L = 100$  iterations,  $q = 7$ ,  $\Delta = 0.25$ ,  $\gamma = 1$ ,  $\lambda = 0$ , and  $\epsilon_{10} = 10^{-3}$ . We also assume a sign-magnitude binary representation for the messages exchanged in the decoder.

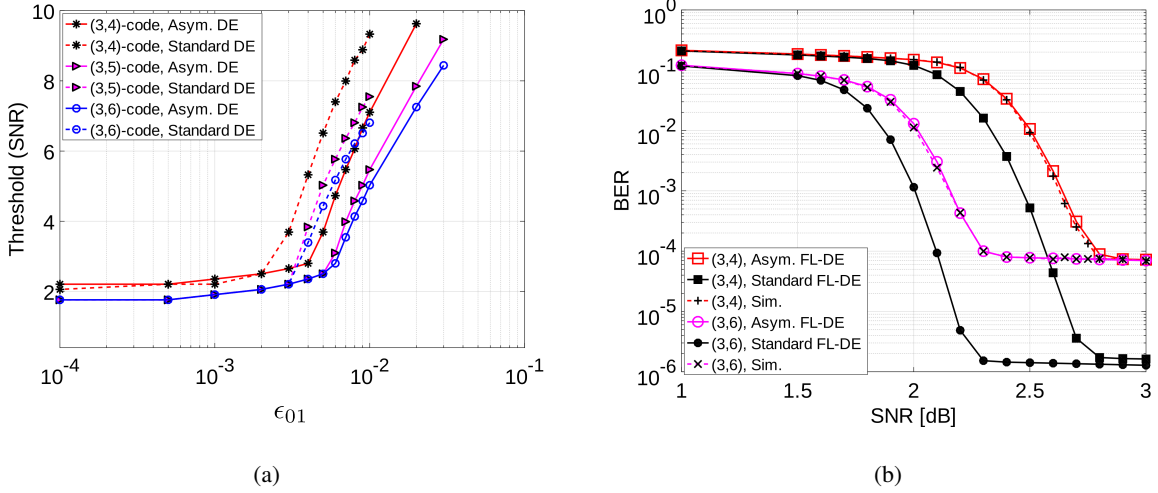


Fig. 4. (a) Regular code ensemble thresholds with respect to  $\epsilon_{01}$  for the quantized Min-Sum decoder, with and without the all-zero codeword assumption, and with  $\epsilon_{10} = 10^{-3}$ . The  $x$ -axis label  $\epsilon_{01}$  is the deviation parameter from bit value 0 to bit value 1. (b) For the quantized Min-Sum decoder, comparison of BER measured from Monte-Carlo simulations and evaluated with the finite-length DE-based method (FL-DE), with and without the all-zero codeword assumption, for (3,4) and (3,6) regular codes of length  $N = 10000$ , with  $\epsilon_{01} = 5 \times 10^{-4}$ ,  $\epsilon_{10} = 10^{-3}$ . For each considered setup, the curves of Monte Carlo simulations and of FL-DE without the all-zero codeword assumption are superimposed.

From the DE equations provided in Section IV-C, we calculate the thresholds with respect to the parameter  $\epsilon_{10}$  for four regular codes with VN degree  $d_v = 3$  and with CN degrees  $d_c = 4, 5, 6, 12$ , respectively. Figure 4 (a) shows the thresholds obtained with respect to  $\epsilon_{01}$  with standard noisy-DE [6] with the all-zero codeword assumption, and with the asymmetric noisy-DE analysis introduced in this paper. The thresholds for standard noisy DE were evaluated by calculating only the probability density functions  $Q_0^{(\ell)}$ ,  $\tilde{Q}_0^{(\ell)}$ , and  $P_0^{(\ell)}$  in (28), (29), (30) and by considering that  $P_0^{(\ell)} = 1 - P_1^{(\ell)}$ . As expected, the code ensemble thresholds increase with  $\epsilon_{01}$ . In addition, the thresholds obtained with standard DE differ from thresholds obtained with our DE analysis, although for small values of  $\epsilon_{01}$ , the two threshold values are the same.

We then evaluate the finite-length performance of the quantized Min-Sum decoder over one (3,4) regular code and one (3,6) regular code. For the decoder, we set parameters  $L = 50$ ,  $q = 7$ ,  $\Delta = 0.25$ ,  $\gamma = 1$ ,  $\lambda = 0$ ,  $\epsilon_{01} = 5 \times 10^{-4}$ ,  $\epsilon_{10} = 10^{-3}$ . For each considered code, we evaluate the BER from Monte-Carlo simulations realized on codes of length  $N = 10000$  and girth 10 constructed from the PEG algorithm. We compare the obtained BER values with the

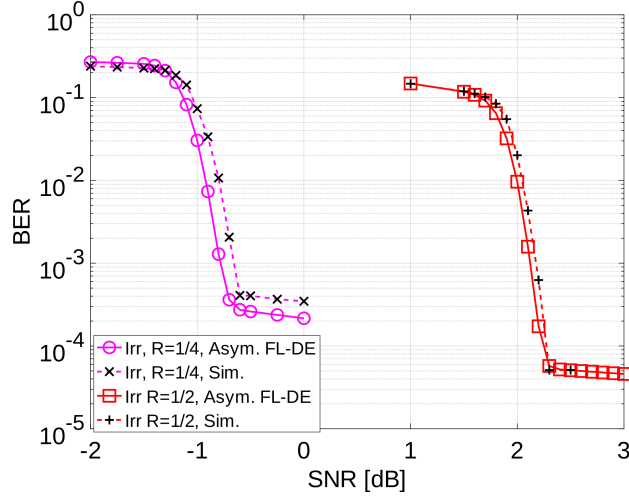


Fig. 5. For the quantized Min-Sum decoder, comparison of BER measured from Monte-Carlo simulations and evaluated with the finite-length DE-based method (FL-DE) with parameters  $\epsilon_{01} = 5 \times 10^{-4}$ ,  $\epsilon_{10} = 10^{-3}$  for two irregular codes: one with rate 1/4 and length  $N = 8000$  and one with rate 1/2 and length  $N = 9972$ .

performance predicted by the finite-length DE-based method of Section III-C, evaluated with and with standard DE (with the all-zero codeword assumption) and with our method (without the all-zero codeword assumption). The results are shown in Figure 4 (b). We observe that our method accurately predicts the decoders performance evaluated with Monte-Carlo simulations, while the curve obtained from standard DE shows a gap with the simulations. These results confirm that the DE method introduced in this paper accurately predicts the performance of quantized Min-Sum decoders with asymmetric deviations, while standard Density Evolution with the all-zero codeword assumption is not accurate. Finally, as for the Gallager B decoder, we observe an error floor at high SNR values, which is due to the deviations in the decoder.

We also evaluate the finite-length performance of the quantized Min-Sum decoder over two irregular LDPC codes, both constructed with the PEG algorithm. The first irregular LDPC code has rate 1/2, codeword length  $N = 9972$ , and degree distributions  $\lambda_1(x) = 0.7857x^2 + 0.2143x^8$ ,  $\rho_1(x) = x^6$  [31]. The second irregular code has rate 1/4, codeword length  $N = 8000$ , and degree distributions  $\lambda_2(x) = 0.1688x + 0.1624x^2 + 0.1313x^8 + 0.2264x^9 + 0.007575x^{10} + 0.2737x^{14} + 0.02986x^{15}$ ,  $\rho_2(x) = 0.19x^5 + 0.81x^6$ . For the decoder, we set parameters  $L = 50$ ,  $q = 5$ ,  $\Delta = 0.5$ ,  $\gamma = 1$ ,  $\lambda = 0$ ,  $\epsilon_{01} = 5 \times 10^{-4}$ ,  $\epsilon_{10} = 10^{-3}$ . For each of the two irregular codes,

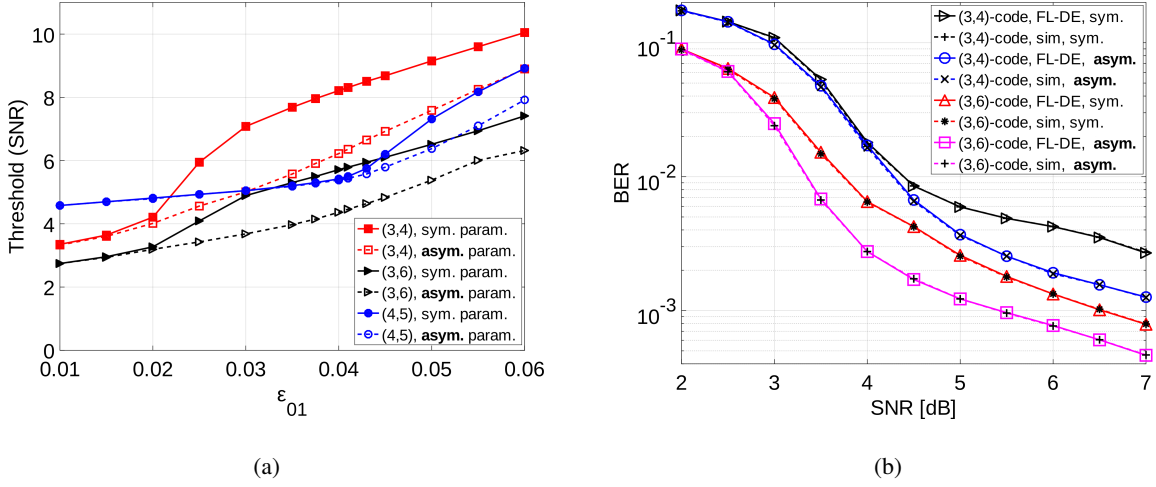


Fig. 6. (a) SNR threshold optimization with symmetric and asymmetric offset and scaling parameters for the quantized Min-Sum decoder, for (3, 4), (3, 6), and (4, 5)-codes, and for  $\epsilon_{10} = 10^{-5}$ . The  $x$ -axis label  $\epsilon_{01}$  is the deviation parameter from bit value 0 to bit value 1. (b) Comparison of BER with respect to SNR for the quantized Min-Sum decoder, for symmetric and asymmetric decoder parameters, for codes of length  $N = 10000$  with  $\epsilon_{10} = 10^{-5}$ , and  $\epsilon_{01} = 5 \times 10^{-2}$ . For each considered setup, plain curves (performance evaluated from FL-DE) and dashed curves (performance evaluated from Monte Carlo simulations) are superimposed.

Figure 5 compares the BER performance evaluated with Monte-Carlo simulations, with the BER performance predicted by the finite length DE method. This Figure confirms that the asymmetric DE method developed in this paper also allows to consider irregular LDPC codes. The gap between the curves obtained from Monte-Carlo simulations and the curves obtained by finite-length DE are slightly increased compared to regular codes considered in Figure 4 (b). This is probably due to the fact that the considered irregular codes are more dense, and hence contain more short cycles, than the regular codes.

We now evaluate the performance of the quantized Min-Sum decoder with asymmetric scaling and offset parameters introduced in Section V-B. We consider regular (3, 4), (3, 6), and (4, 5)-codes, and decoder parameters given by  $L = 10$  iterations,  $q = 4$ ,  $\Delta = 1$ , and  $\epsilon_{10} = 10^{-5}$ . In Figure 6 (a), we show the SNR thresholds with respect to the noise parameter  $\epsilon_{01}$ , obtained by considering symmetric and asymmetric parameters. We observe that asymmetric parameters improve the decoder performance compared to symmetric parameters. In our simulations, we observed that both the scaling parameter and the offset parameter need to be asymmetric.

TABLE I  
OPTIMIZED SYMMETRIC AND ASYMMETRIC PARAMETER VALUES FOR THE QUANTIZED MIN-SUM DECODER, FOR (3, 4),  
(3, 6), AND (4, 5)-CODES, FOR  $\epsilon_{10} = 10^{-5}$ .

		Symmetric parameters			Asymmetric parameters				
code	$\epsilon_{01}$	$\gamma$	$\lambda$	Threshold	$\gamma_0$	$\gamma_1$	$\lambda_0$	$\lambda_1$	Threshold
(3,4)	0.01	0.85	0	3.45 dB	0.90	0.80	0	0	3.30 dB
	0.03	0.55	1	7.08 dB	1.0	0.25	1	0	5.01 dB
(3,6)	0.01	0.70	0	2.75 dB	0.75	0.7	0	0	2.74 dB
	0.03	0.60	1	4.89 dB	1.0	0.35	1	0	3.68 dB
(4,5)	0.01	0.95	0	4.58 dB	0.95	0.90	0	0	4.48 dB
	0.05	1.0	0	7.31 dB	1.0	0.25	1	0	6.38 dB

For instance, symmetric scaling parameter with asymmetric offset do not improve much the performance compared to the fully symmetric setup. In addition, Table I provides the optimized values of the symmetric and asymmetric parameters as well as the corresponding thresholds, for two values of  $\epsilon_{10}$  for the three considered codes. The values in the Table confirm that asymmetric parameters allow to improve the decoder thresholds.

At the end, we confirm the interest of asymmetric quantized Min-Sum parameters on finite-length simulations. We consider decoder parameters given by  $L = 10$  iterations,  $q = 4$ ,  $\Delta = 1$ ,  $\epsilon_{10} = 10^{-5}$ , and  $\epsilon_{01} = 5 \times 10^{-2}$ . For (3, 4) and the (3, 6) regular codes, we compare the BER performance obtained with optimized symmetric and asymmetric decoder parameters. In both cases, we predict the BER from the finite-length DE-based method of Section III-C, and evaluate the BER performance from Monte Carlo simulations on codes of length  $N = 10000$  and girth 8 obtained from a PEG algorithm. The results are shown in Figure 6 (b), and confirm that asymmetric parameters clearly improve the BER performance of the quantized Min-Sum decoder under asymmetric deviations. We also see that the method of Section III-C accurately predicts the BER performance at finite length, which validates the asymmetric DE analysis introduced in this paper.

## VII. CONCLUSION

We considered noisy LDPC decoders under asymmetric deviation models and derived the noisy DE equations without the all-zero codeword assumption for three noisy decoders: the BP decoder, the Gallager B decoder, and the quantized offset MS decoder. We then proposed to compensate the effects of asymmetric deviations by introducing asymmetric parameters in the Gallager B and quantized MS decoders. Numerical simulations confirmed both the accuracy of the proposed DE method, and the performance improvement provided by the use of asymmetric parameters. Future work will target the optimization of irregular degree distributions and protographs under asymmetric deviation models.

## ACKNOWLEDGEMENT

The authors were supported by the grant ANR-17-CE40-0020 of the French National Research Agency ANR (project EF-FEctive).

## APPENDIX A

### CN MESSAGE PROBABILITY DENSITY FUNCTION IN THE BP DECODER

We derive the CN output message probability density function (16) for the BP decoder. At the CN, the output message is denoted  $\beta$  and the input messages are denoted  $\alpha_i$ . We first consider the case where the VN that receives message  $\beta$  has value  $x = 0$ . Denote by  $x_1, \dots, x_{d_c-1}$  the bit values of the VNs that sent messages  $\alpha_1, \dots, \alpha_{d_c-1}$ . Denote by  $V$  the random variable that represents the number of values 1 among  $x_1, \dots, x_{d_c-1}$ . By marginalization with respect to  $V$ , we have

$$Q_0^{(\ell)}(\beta) = \mathbb{P}(\beta|x=0) = \sum_{v=0}^{d_c-1} \mathbb{P}(V=v|x=0) \mathbb{P}(\beta|x=0, V=v). \quad (43)$$

If  $v$  is odd, then  $\mathbb{P}(V=v|X=0) = 0$ , since  $x + \sum_{i=1}^{d_c-1} x_i = 0$ . If  $v$  is even, then

$$\mathbb{P}(V=v|X=0) = \binom{d_c-1}{v} \left(\frac{1}{2}\right)^{d_c-2}.$$

In addition,

$$\mathbb{P}(\beta|x=0, V=v) = \Gamma^{-1} \left( \Gamma \left( \tilde{P}_0^{(\ell)} \right)^{\otimes (d_c-1-v)} \otimes \Gamma \left( \tilde{P}_1^{(\ell)} \right)^{\otimes v} \right),$$

since  $(x_1, \dots, x_{d_c-1})$  contains  $v$  values 1 and  $(d_c-1-v)$  values 0. Combining these expressions gives  $Q_0^{(\ell)}(\beta)$  in (16). The derivation of  $Q_1^{(\ell)}(\beta)$  follows the same process.

## APPENDIX B

### PROBABILITY DENSITY FUNCTION OF INITIAL MESSAGES IN THE QUANTIZED MIN-SUM DECODER WITH ASYMMETRIC PARAMETERS

In this part, we consider an AWGN channel with variance  $\sigma^2$ . In order to compute the probability density functions  $P_x^{(0)}$  of initial messages in the quantized Min-Sum decoder with asymmetric parameters, we need to determine the cumulative distribution function of a random variable  $z = \frac{2\gamma(y)y}{\sigma^2}$ , where  $\gamma(y) = \gamma_0$  if  $y \geq 0$ , and  $\gamma(y) = \gamma_1$  if  $y < 0$ . We denote by  $\mathbb{F}_{z|x}(t) = \mathbb{P}(z \leq t|x)$  the cumulative distribution function of the random variable  $z$  conditioned on  $x \in \{0, 1\}$ .

The cumulative distribution function  $\mathbb{F}_{z|x}(t)$  can be expressed as

$$\mathbb{F}_{z|x}(t) = \mathbb{P}\left(\frac{2\gamma(y)y}{\sigma^2} \leq t|x\right) \quad (44)$$

$$= \mathbb{P}\left(\left(\frac{2\gamma(y)y}{\sigma^2} \leq t\right) \cap (y \geq 0) | x\right) + \mathbb{P}\left(\left(\frac{2\gamma(y)y}{\sigma^2} \leq t\right) \cap (y \leq 0) | x\right) \quad (45)$$

by the law of total probability. We then treat separately the case where  $t < 0$  and the case  $t \geq 0$ , starting with the first case.

Since  $\gamma(y) > 0$ , the condition  $t < 0$  implies that  $y < 0$ . As a result, when  $t < 0$ , the cumulative distribution function  $\mathbb{F}_{z|x}(t)$  is given by  $\mathbb{F}_{z|x}(t) = \mathbb{P}\left(\frac{2\gamma_1 y}{\sigma^2} | x\right)$ . We denote  $\mu_1(x) = \frac{2\gamma_1(1-2x)}{\sigma^2}$  and  $\sigma_1^2 = \frac{4\gamma_1^2}{\sigma^2}$ . Then, the random variable  $\frac{2\gamma_1 y}{\sigma^2}$  follows a Gaussian distribution with mean  $\mu_1(x)$  and variance  $\sigma_1^2$ . As a result,

$$\forall t < 0, \mathbb{F}_{z|x}(t) = \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{t - \mu_1(x)}{\sqrt{2}\sigma_1}\right) \quad (46)$$

where erf is the error function of the Gaussian distribution.

Now, for  $t \geq 0$ , the cumulative distribution function  $\mathbb{F}_{z|x}(t)$  is given by

$$\mathbb{F}_{z|x}(t) = \mathbb{P}\left(\frac{2\gamma(y)y}{\sigma^2} \leq 0|x\right) + \mathbb{P}\left(0 \leq \frac{2\gamma(y)y}{\sigma^2} \leq t|x\right). \quad (47)$$

We denote  $\mu_0(x) = \frac{2\gamma_0(1-2x)}{\sigma^2}$  and  $\sigma_0^2 = \frac{4\gamma_0^2}{\sigma^2}$ . Then, the random variable  $\frac{2\gamma_0 y}{\sigma^2}$  follows a Gaussian distribution with mean  $\mu_0(x)$  and variance  $\sigma_0^2$ . As a result,

$$\forall t \geq 0, \mathbb{F}_{z|x}(t) = \frac{1}{2} + \frac{1}{2}\text{erf}\left(-\frac{\mu_1(x)}{\sqrt{2}\sigma_1}\right) + \frac{1}{2}\text{erf}\left(\frac{\mu_0(x)}{\sqrt{2}\sigma_0}\right) + \frac{1}{2}\text{erf}\left(\frac{t - \mu_0(x)}{\sqrt{2}\sigma_0}\right). \quad (48)$$

## REFERENCES

- [1] P. Gupta, Y. Agarwal, L. Dolecek, N. Dutt, R. K. Gupta, R. Kumar, S. Mitra, A. Nicolau, T. S. Rosing, M. B. Srivastava, S. Swanson, and D. Sylvester, “Underdesigned and opportunistic computing in presence of hardware variability,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 1, pp. 8–23, Jan 2013.
- [2] Y. Yang, P. Grover, and S. Kar, “Computing linear transformations with unreliable components,” *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 3729–3756, 2017.
- [3] —, “Fault-tolerant distributed logistic regression using unreliable components,” in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2016, pp. 940–947.
- [4] E. Dupraz and L. R. Varshney, “Binary recursive estimation on noisy hardware,” in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 877–881.
- [5] G. B. Hacene, F. Leduc-Primeau, A. B. Soussia, V. Gripon, and F. Gagnon, “Training modern deep neural networks for memory-fault robustness,” in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2019, pp. 1–5.
- [6] L. Varshney, “Performance of LDPC codes under faulty iterative decoding,” *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4427–4444, 2011.
- [7] F. Leduc-Primeau and W. J. Gross, “Faulty Gallager-B decoding with optimal message repetition,” in *Proc. 50th Allerton Conf. on Communication, Control, and Computing*, Oct. 2012.
- [8] O. Al Rasheed, P. Ivaniš, and B. Vasić, “Fault-tolerant probabilistic gradient-descent bit flipping decoder,” *IEEE Communications Letters*, vol. 18, no. 9, pp. 1487–1490, 2014.
- [9] G. Sundararajan, C. Winstead, and E. Boutillon, “Noisy gradient descent bit-flip decoding for LDPC codes,” *IEEE Transactions on Communications*, vol. 62, no. 10, p. 16, 2014.
- [10] C.-H. Huang, Y. Li, and L. Dolecek, “Gallager B LDPC decoder with transient and permanent errors,” *IEEE Transactions on Communications*, vol. 62, no. 1, pp. 15–28, 2014.
- [11] C. K. Ngassa, V. Savin, E. Dupraz, and D. Declercq, “Density evolution and functional threshold for the noisy min-sum decoder,” *IEEE Transactions on Communications*, vol. 63, no. 5, pp. 1497–1509, 2015.
- [12] A. Balatsoukas-Stimming and A. Burg, “Density evolution for min-sum decoding of LDPC codes under unreliable message storage,” *IEEE Communications Letters*, vol. 18, no. 5, pp. 849–852, 2014.
- [13] F. Leduc-Primeau, F. R. Kschischang, and W. J. Gross, “Modeling and energy optimization of LDPC decoder circuits with timing violations,” *IEEE Transactions on Communications*, vol. 66, no. 3, pp. 932–946, 2018.
- [14] E. Dupraz, D. Declercq, B. Vasić, and V. Savin, “Analysis and design of finite alphabet iterative decoders robust to faulty hardware,” *IEEE Transactions on Communications*, vol. 63, no. 8, pp. 2797–2809, 2015.

- [15] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, 2001.
- [16] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.
- [17] C.-C. Wang, S. R. Kulkarni, and H. V. Poor, "Density evolution for asymmetric memoryless channels," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4216–4236, 2005.
- [18] J. Chen, D.-k. He, and A. Jagmohan, "The equivalence between Slepian-Wolf coding and channel coding under density evolution," *IEEE Transactions on Communications*, vol. 57, no. 9, 2009.
- [19] E. V. Bravo, A. Bonetti, and A. Burg, "Data-retention-time characterization of gain-cell eDRAMs across the design and variations space," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2019.
- [20] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8t subthreshold sram employing sense-amplifier redundancy," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, 2008.
- [21] M. Nabavi, F. Ramezankhani, and M. Shams, "Optimum pmos-to-nmos width ratio for efficient subthreshold cmos circuits," *IEEE Transactions on Electron Devices*, vol. 63, no. 3, pp. 916–924, 2016.
- [22] J. Nadal, M. Fiorentino, E. Dupraz, and F. Leduc-Primeau, "A deeply pipelined, highly parallel and flexible ldpc decoder," in *2020 18th IEEE International New Circuits and Systems Conference (NEWCAS)*, 2020, pp. 263–266.
- [23] V. Savin, "Self-corrected min-sum decoding of LDPC codes," in *2008 IEEE International Symposium on Information Theory*. IEEE, 2008, pp. 146–150.
- [24] E. Dupraz, V. Savin, and M. Kieffer, "Density evolution for the design of non-binary low density parity check codes for Slepian-Wolf coding," *IEEE Transactions on Communications*, vol. 63, no. 1, pp. 25–36, 2014.
- [25] F. Leduc-Primeau and W. J. Gross, "Finite-length quasi-synchronous LDPC decoders," in *9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*. IEEE, 2016, pp. 325–329.
- [26] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.
- [27] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 386–398, 2005.
- [28] A. Tarighati, H. Farhadi, and F. Lahouti, "Design of LDPC codes robust to noisy message-passing decoding," *arXiv preprint arXiv:1501.02483*, 2015.
- [29] M. Gorgoglione, V. Savin, and D. Declercq, "Optimized puncturing distributions for irregular non-binary ldpc codes," in *2010 International Symposium On Information Theory & Its Applications*. IEEE, 2010, pp. 400–405.

- [30] T. J. Richardson and R. L. Urbanke, “Efficient encoding of low-density parity-check codes,” *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 638–656, 2001.
- [31] D. MacKay, “David mackay’s gallager code resources,” <http://www.inference.phy.cam.ac.uk/mackay/CodesFiles.html>, 2009.