Asymptotic Error Probability of the Gallager B Decoder under Timing Errors

Elsa DUPRAZ[1], David DECLERCQ[2], and Bane VASIC[3]

[1] Telecom Bretagne; UMR CNRS 6285 Lab-STICC, Brest, France
[2] ETIS - ENSEA / Univ. Cergy-Pontoise / CNRS UMR-8051, Cergy-Pontoise, France
[3] Dept. of Electrical and Computer Eng., University of Arizona, Tucson, USA

***Abstract*—In a circuit, timing errors occur when a logic gate output does not switch before the clock rising edge. In this paper, we consider Gallager B decoders under timing errors, following the error model derived in [1] from SPICE measurements. For this model, we provide a theoretical analysis of the performance of LDPC decoders. Our study is based on the analysis of the computation trees of the decoder free of logic gate errors and of the decoder with timing errors. As a main result, we show that as the number of iterations goes to infinity, the error probability of the decoder with timing errors converges to the error probability of the logic gate error-free decoder. Monte Carlo simulations confirm this result even for moderate code lengths, which is in accordance with the experimental observations [2].**

***Index Terms*—low-density parity check codes, faulty Gallager B decoders, timing errors, computation trees**

I. INTRODUCTION

Over the past decade, the size of electronic chips has considerably reduced, while the integration factors have dramatically increased. Due to this huge scale change, energy consumption has become a major issue in the design of the next generations of electronic devices. Typical solutions involve decreasing the power supply of electronic chips by several orders of magnitude and/or reducing the clock period [3]. However, both lower power supply and clock period reduction make electronic components much more sensitive to noise.

Due to increased noise sensitivity, errors of different nature may appear in the computation units built on such hardware. Permanent errors make some of the gate outputs stuck at a certain value, while transient errors appear only from time to time at the gate output. As a particular case, reducing the clock period makes computational units much more sensitive to transient timing errors that appear when the logic gate output does not switch before the clock rising edge [4].

In the context of communication and storage, several recent works were devoted to the performance analysis of Low Density Parity Check (LDPC) decoders under hardware errors. Most of these works consider very simple transient data-independent models to represent hardware errors [5]–[10]. The performance of LDPC decoders under data-dependent transient models was investigated in [11] while permanent errors were considered in [12]. Most of these works show that hardware errors degrade the performance of LDPC decoders. Surprisingly, it was also shown that under certain conditions, the noise may improve the performance of Gradient Descent Bit-Flipping decoders [8], [9] and Gallager-B decoders [13].

The objective of this paper is to analyze the effect of timing errors on the performance of LDPC decoders. Timing errors cannot be represented by the transient error models considered in [5]–[9] since they induce memory in the decoder. For example, in the timing errors model considered in [2], the gate output can be either the correct value, or the value from the previous iteration if the gate output did not switch before the clock rising edge. This model was initially proposed in [1] and its accuracy was verified by Monte-Carlo simulations on SPICE (Simulation Program with Integrated Circuit Emphasis). In [2], it was observed that Gallager B decoders with timing errors can actually achieve the same performance as error-free decoders. Although very promising, the performance analysis of [2] was only carried through Monte-Carlo simulations. Hence, in order to understand whether the results of [2] can be generalized to other code and decoder parameters, there is a need for a theoretical analysis of LDPC decoders under timing errors.

In this paper, we consider the error model introduced in [2] for timing errors in a Gallager-B LDPC decoder (see Section III). For this model, we provide a theoretical analysis of the performance of LDPC decoders under timing errors (Section IV). Our analysis relies on the same proof technique as for the analysis of the performance of LDPC decoders under serial scheduling [14]. We first provide bounds on the computation tree of the LDPC decoder under timing errors. From these bounds, we show that as the number of iterations goes to infinity, the error probability of the decoder with timing errors converges to the error probability of the error-free decoder. We confirm through Monte Carlo simulations the accuracy of the proposed analysis and show that the asymptotic performance of LDPC decoders under timing errors is indeed the same as the performance of error-free decoders even for moderate code lengths (Section V).

II. PRELIMINARIES

Consider an LDPC code, with the parity check matrix H of size $m \times n$. The corresponding Tanner graph is composed of n Variable Nodes (VN) $v \in V = \{1, \dots, n\}$, m Check Nodes (CN) $c \in C = \{1, \dots, m\}$, and edges connecting VNs and CNs. The set of CNs (resp. VNs) that are connected to VN v (resp. CN c) is denoted $\mathcal{N}_v$ (resp. $\mathcal{N}_c$).

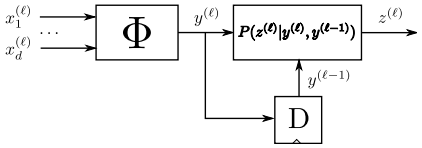

Fig. 1. Error model for timing errors in the decoder

A binary phase-shift keyed codeword is transmitted over a Binary Symmetric Channel (BSC) of parameter α, and decoded by the Gallager-B algorithm for maximum of L iterations. For the v-th symbol of the codeword, denote by y_v the value received from the channel, where $y_v \in \{-1, +1\}$. We assume that the messages in the decoder are updated according to a flooded scheduling. Denote by $\nu_{v,c}^{(\ell)}$ the extrinsic messages from a VN v to a CN c at iteration $\ell \in \{0, \cdots, L\}$. Also let $\mu_{c,v}^{(\ell)}$ be the extrinsic message from a CN c to a VN v at iteration $\ell \in \{1, \cdots, L\}$. Let $\boldsymbol{\mu}^{(\ell)} = \mu_{\mathcal{N}_v \setminus c,v}^{(\ell)}$ denote all incoming messages to the VN v except the message from the CN c. Similarly, $\boldsymbol{\nu}^{(\ell)} = \nu_{\mathcal{N}_c \setminus v,c}^{(\ell)}$ denote all incoming messages to the CN c except from the VN v.

In each iteration $\ell > 0$, for each CN $c \in C$, and for all $v \in \mathcal{N}_c$, $\mu_{c,v}^{(\ell)} = \Psi(\boldsymbol{\nu}^{(\ell-1)}) = \prod \boldsymbol{\nu}^{(\ell-1)}$, where $\prod$ is taken componentwise and denotes the product of the incoming messages. In each iteration $l \geq 0$, for each VN $v \in V$, and for all $c \in \mathcal{N}_v$, $\nu_{v,c}^{(\ell)} = \Phi(y_v, \boldsymbol{\mu}^{(\ell)})$ with

$$\Phi(y_v, \boldsymbol{\mu}^{(\ell)}) = \begin{cases} y_v & \ell = 0 \\ \varphi(y_v, \boldsymbol{\mu}^{(\ell)}) & \ell > 0 \end{cases}$$

where

$$\varphi(y_v, \boldsymbol{\mu}) = \begin{cases} \text{sgn}(\texttt{MAJ}(\boldsymbol{\mu})) & |\texttt{MAJ}(\boldsymbol{\mu})| \geq b \\ y_v & \text{otherwise.} \end{cases} \quad (1)$$

The value b is a parameter of the Gallager B decoder, $|.|$ denotes the absolute value, and sgn is the *signum* function. By convention, we take $\text{sgn}(0) = 0$. The function MAJ is defined as $\texttt{MAJ}(\boldsymbol{\mu}) = \sum \boldsymbol{\mu}$ wherein $\sum$ denotes the sum of its argument's components.

Based on the above description of the error-free Gallager B decoder, we now introduce the error model we consider to represent the timing errors in the decoder.

III. Timing Error Model

We now formulate mathematically the error model that was proposed in [1] and considered in [2] in order to represent timing errors in the decoder.

Denote by Φ a deterministic Boolean function with d inputs and a single output. Φ is computed by a logic gate, and let the inputs at iteration ℓ be $(x_1^{(\ell)}, \ldots, x_d^{(\ell)})$ and denote the output by $y^{(\ell)} = \Phi(x_1^{(\ell)}, \ldots, x_d^{(\ell)})$. The error model is depicted in Figure 1, wherein the output of the noisy gate is $z^{(\ell)}$. In general, $z^{(\ell)}$ is a random variable described by the conditional distribution function $P(z^{(\ell)}|y^{(\ell)}, y^{(\ell-1)})$. It defines an error model with memory, since the output $z^{(\ell)}$ depends on the function output value $y^{(\ell-1)}$ in the previous iteration. In this letter we consider the case

$$z^{(\ell)} = \overline{D}^{(\ell)} \cdot y^{(\ell)} \oplus D^{(\ell)} \cdot y^{(\ell-1)}, \quad (2)$$

where the "delay error" $D^{(\ell)}$, $\ell > 0$ are Bernoulli random variables with parameter ε, $\overline{D} = 1 \oplus D$, and $\cdot$ is the Boolean AND. According to (2), a noisy function outputs the random variable $z^{(\ell)}$ that can take only two values, namely $y^{(\ell)}$ and $y^{(\ell-1)}$ with the following probabilities $P\{z^{(\ell)} = y^{(\ell)}\} = 1 - \varepsilon$ and $P\{z^{(\ell)} = y^{(\ell-1)}\} = \varepsilon$. The model described by (2) captures the probability that the gate output did not switch before the clock rising edge.

In the decoder with timing errors, we assume that the initialization of the messages $\nu_{v,c}^{(0)}$ is error-free. We also assume that the first iteration (labeled as 1) is error-free. This assumption was already considered in [2] so that the decoding of the current codeword does not depend on the decoding of the previous codeword. In practice, this may be done by waiting for several clock cycles for the signal to stabilize before moving to the next iteration.

In the following, we denote by $\nu_{v,c}^{(\ell)}$ and $\mu_{c,v}^{(\ell)}$ the messages that are exchanged in the error-free decoder, and by $\widetilde{\nu}_{v,c}^{(\ell)}$ and $\widetilde{\mu}_{c,v}^{(\ell)}$ the messages that are exchanged in the decoder with timing errors at iteration ℓ. Let $D_{c,v}^{(\ell)}$ and $D_{v,c}^{(\ell)}$ be the realizations of delay errors that occur in computation of messages $\widetilde{\nu}_{v,c}^{(\ell)}$ and $\widetilde{\mu}_{c,v}^{(\ell)}$. The corresponding Bernoulli parameters are ε_{MAJ} and $\varepsilon_{\oplus}$. Then

$$\begin{aligned} \widetilde{\mu}_{c,v}^{(\ell)} &= \overline{D}_{c,v}^{(\ell)} \cdot \Psi(\tilde{\boldsymbol{\nu}}^{(\ell-1)}) \oplus D_{c,v}^{(\ell)} \cdot \Psi(\tilde{\boldsymbol{\nu}}^{(\ell-2)}) \\ \widetilde{\nu}_{v,c}^{(\ell)} &= \overline{D}_{v,c}^{(\ell)} \cdot \Phi(y_v, \tilde{\boldsymbol{\mu}}^{(\ell)}) \oplus D_{v,c}^{(\ell)} \cdot \Phi(y_v, \tilde{\boldsymbol{\mu}}^{(\ell-1)}). \end{aligned} \quad (3)$$

To streamline the exposition of the results, we assume $\varepsilon_{\text{MAJ}} = \varepsilon_{\oplus} = \varepsilon$. Note, however, that our theoretical analysis may be readily generalized to values of ε that are different for CNs and VNs.

IV. Performance Analysis

This section provides a theoretical analysis of the performance of LDPC decoders under timing errors. As for the analysis of the performance of decoders under serial scheduling [14], our analysis is based on the comparison of the computation graphs of the decoder under timing errors and of the error-free decoder. In the following, we will assume, as in [14], [15], that these computation graphs are cycle-free and we will refer to them as computation trees. For a definition of a computation tree, the reader is referred to [16, Chapter 4].

The channel is output symmetric, and the deterministic VN and CN mappings followed by the timing error model (2) are also symmetric, see [7], [17]. Applying [7, Theorem 1] and the results of [17] for symmetric decoders hence gives that the error probability of the decoder with timing errors does not depend on the transmitted codeword. In the analysis, we thus assume without loss of generality that the all-zero codeword was transmitted.

A. Computation Tree Analysis

For the analysis, the first key observation is that the message on the edge $e = (v, c)$ from v to c in iteration ℓ can be ultimately expressed as a function $f_{v,c}^{(\ell)}$ of the channel output values corresponding to the set of all VNs in the computation tree. Denote by $\mathcal{N}_e^{(\ell)}$ the computation tree of edge e at

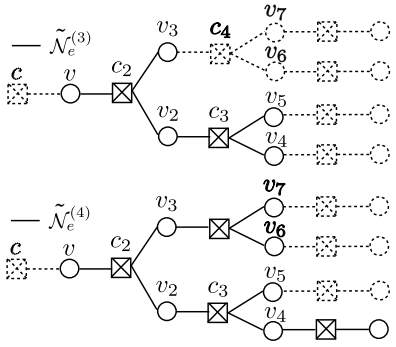

Fig. 2. An illustration of the situation when (i) two timing errors have been introduced in the computation of the messages from v_3 to c_2 and from v_5 to c_3 at iteration $\ell = 2$, and (ii) one timing error has been introduced in the computation of the message from v_3 to c_2 at iteration $\ell = 3$. The solid lines represent the computation trees $\widetilde{\mathcal{N}}_e^{(3)}$ and $\widetilde{\mathcal{N}}_e^{(4)}$ of the decoder with timing errors, while dashed nodes and edges are outside of the computation tree.

iteration ℓ for the error-free decoder, and denote by $\widetilde{\mathcal{N}}_e^{(\ell)}$ the computation tree in the decoder with timing errors. For the error-free decoder, $\mathcal{N}_e^{(\ell)}$ by definition includes all the VNs and CNs at distance strictly less than $2\ell - 1$ of v [15]. The second key observation is that according to the model defined in (2), $\widetilde{\mathcal{N}}_e^{(\ell)}$ is not a complete computation tree but a random graph which depends on the timing errors that occur in the decoder.

In the following, before stating our main result, we first illustrate the relation between computation trees of the perfect decoder and of the decoder with timing errors.

Example. *Fig. 2 shows computation trees for a decoder with timing errors at iterations $\ell = 3$ and $\ell = 4$. These computation trees are obtained from the message exchange in the decoder, assuming that timing errors have been introduced in the computation of the messages from v_3 to c_2 at iterations $\ell = 2$ and $\ell = 3$, and in the computation of the message from v_5 to c_3 at iteration $\ell = 2$. In the computation trees of the decoder with timing errors, a message exchanged from VN v to CN c at iteration ℓ can be expressed as a function $f_{v,c}^{(\ell)}$ of a given set of channel outputs. For example, $\widetilde{\nu}_{v_6,c_4}^{(1)} = f_{v_6,c_4}^{(1)}(y_6)$ and $\widetilde{\mu}_{c_4,v_3}^{(1)} = f_{c_4,v_3}^{(1)}(y_6, y_7)$. The computation trees in Fig. 2 are obtained from the following message exchange in the graph.*

- *At iteration 1, no timing error is introduced, and thus $\widetilde{\nu}_{v_2,c_2}^{(1)} = f_{v_2,c_2}^{(1)}(y_2)$, $\widetilde{\nu}_{v_3,c_2}^{(1)} = f_{v_3,c_2}^{(1)}(y_3)$. Also, $\widetilde{\mu}_{c_2,v}^{(1)} = f(y_2, y_3)$. From this message exchange, we get $\widetilde{\mathcal{N}}_e^{(2)} = \mathcal{N}_e^{(2)} = \{v, c_2, v_2, v_3\}$.*
- *At iteration 2, we get $\widetilde{\nu}_{v_2,c_2}^{(2)} = f_{v_2,c_2}^{(2)}(y_2, y_4, y_5)$. Since one timing error is introduced when computing $\widetilde{\nu}_{v_3,c_2}^{(2)}$, we have $\widetilde{\nu}_{v_3,c_2}^{(2)} = f_{v_3,c_2}^{(2)}(y_3)$ that is the message from iteration 1. As a result, $\widetilde{\mu}_{c_2,v}^{(2)} = f_{c_2,v}^{(2)}(y_2, \cdots, y_5)$. The computation tree $\widetilde{\mathcal{N}}_e^{(3)}$ hence corresponds to the upper part of Fig. 2. Note that the timing error in the computation of the message from v_5 to c_3 will only impact the computation tree at iteration 3.*
- *At iteration 3, since one timing error is introduced when computing $\widetilde{\nu}_{v_3,c_2}^{(3)}$, $\widetilde{\nu}_{v_3,c_2}^{(3)} = f_{v_3,c_2}^{(3)}(y_3, y_6, y_7)$, the correct message from iteration 2. The computation tree $\widetilde{\mathcal{N}}_e^{(4)}$ hence corresponds to the lower part of Fig.2, where the nodes after v_5 are not in the computation tree due to the timing error in the message from v_5 to c_3 at iteration 2.*

In order to perform our analysis, we need the notion of a tree inclusion relation defined as follows. Given two subtrees $\mathcal{N}_1$ and $\mathcal{N}_2$ of the Tanner graph, $\mathcal{N}_1$ is said to be included in $\mathcal{N}_2$, and denoted $\mathcal{N}_1 \subseteq \mathcal{N}_2$, if all the (check and variable) nodes in $\mathcal{N}_1$ are also in $\mathcal{N}_2$. From the message exchange in Example, it can be noticed that $\mathcal{N}_e^{(2)} \subseteq \widetilde{\mathcal{N}}_e^{(3)} \subseteq \mathcal{N}_e^{(3)}$. This relation may be generalized to any error pattern and at any iteration, as stated in the following theorem.

Theorem 1. *For any edge e and at any iteration $\ell > 0$,*

$$\mathcal{N}_e^{(\ell+1)} \subseteq \widetilde{\mathcal{N}}_e^{(3\ell)} \subseteq \mathcal{N}_e^{(3\ell)}. \tag{4}$$

Proof: The proof is done by induction for a given edge e. After the first (error-free) iteration of the decoder, we have $\widetilde{\mathcal{N}}_e^{(2)} = \mathcal{N}_e^{(2)}$. If no timing error is introduced from iteration $\ell + 1 = 2$ to iteration $3\ell = 3$, then $\widetilde{\mathcal{N}}_e^{(3)} = \mathcal{N}_e^{(3)}$. If at iteration 2, timing errors are introduced at every CN and VN computation, then $\widetilde{\mathcal{N}}_e^{(3)} = \mathcal{N}_e^{(2)}$. As a result, since $\widetilde{\mathcal{N}}_e^{(3)} = \mathcal{N}_e^{(3)}$ or $\widetilde{\mathcal{N}}_e^{(3)} = \mathcal{N}_e^{(2)}$, it follows that $\mathcal{N}_e^{(2)} \subseteq \widetilde{\mathcal{N}}_e^{(3)} \subseteq \mathcal{N}_e^{(3)}$.

Now assume that for a given ℓ the relation $\mathcal{N}_e^{(\ell+1)} \subseteq \widetilde{\mathcal{N}}_e^{(3\ell)} \subseteq \mathcal{N}_e^{(3\ell)}$ is satisfied. First consider the most favorable case $\widetilde{\mathcal{N}}_e^{(3\ell)} = \mathcal{N}_e^{(3\ell)}$. If at iterations 3ℓ, $3\ell + 1$ and $3\ell + 2$ no timing error is introduced in the decoder, then $\widetilde{\mathcal{N}}_e^{(3(\ell+1))} = \mathcal{N}_e^{(3(\ell+1))}$. Now consider the least favorable case $\widetilde{\mathcal{N}}_e^{(3\ell)} = \mathcal{N}_e^{(\ell+1)}$. In this case, if at iterations 3ℓ, $3\ell + 1$ and $3\ell + 2$, timing errors are introduced at every VN and CN computation, $\widetilde{\mathcal{N}}_e^{(3\ell+2)} = \widetilde{\mathcal{N}}_e^{(3\ell+1)} = \mathcal{N}_e^{(\ell+1)}$ and $\widetilde{\mathcal{N}}_e^{(3\ell+3)} = \mathcal{N}_e^{(\ell+2)}$. Hence, according to the two above extreme cases, $\mathcal{N}_e^{(\ell+2)} \subseteq \widetilde{\mathcal{N}}_e^{(3(\ell+1))} \subseteq \mathcal{N}_e^{(3(\ell+1))}$, which proves the Theorem by induction. ∎

The above theorem shows that at a given iteration 3ℓ, the computation tree of the decoder with timing errors is smaller than the computation tree of the error-free decoder. The theorem also gives the smallest subtree $\mathcal{N}_e^{(\ell+1)}$ that the decoder with timing errors has explored at iteration 3ℓ. Note that the bounds (4) on computation trees cannot, in general, be rewritten into bounds on error probabilities, unless the Belief Propagation (BP) decoder is considered (see [15, Theorem 7]).

At the end, Theorem 1 permits to analyze the error probability of the decoder under timing errors when the number of iterations goes to infinity.

B. Asymptotic Error Probability

Denote by $P_e^{(\ell)}$ the bit error probability of the error-free decoder at iteration ℓ, and denote by $\widetilde{P}_e^{(\ell)}$ the bit error probability of the decoder with timing errors. The expression of $P_e^{(\ell)}$ can be obtained with density evolution, as described in [15]. The error probability $\widetilde{P}_e^{(\ell)}$ could be expressed with the density evolution technique proposed for decoders with memory [17], but its expression would be very difficult to derive and to evaluate. Hence, in the following, instead of deriving the expression of $\widetilde{P}_e^{(\ell)}$ for any ℓ, we only give the asymptotic error probability $\widetilde{P}_e^{(+\infty)}$.

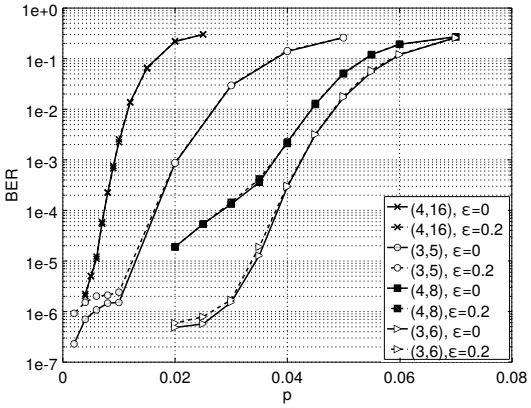

Fig. 3. BER of Gallager B decoder after 100 iterations.

Theorem 2. *If the error probability $P_e^{(\ell)}$ has a limit $P_e^{(+\infty)}$ when ℓ goes to infinity, then $\widetilde{P}_e^{(+\infty)} = P_e^{(+\infty)}$.*

Proof: By taking the limit of (4) when ℓ goes to infinity, we get $\lim_{\ell\to\infty} \widetilde{\mathcal{N}}_e^{(\ell)} = \lim_{\ell\to\infty} \mathcal{N}_e^{(\ell)}$, which gives $\widetilde{P}_e^{(+\infty)} = P_e^{(+\infty)}$, since the same computation tree necessarily gives the same error probability. ∎

Theorem 2 shows that the performance of the decoder with timing errors reaches the performance of the error-free decoder when the number of iterations is sufficiently large. This confirms what was observed experimentally in [2] for the Gallager B decoder for a high number of iterations. We now verify at finite length the accuracy of the asymptotic analysis.

V. Simulation Results

In this section, we evaluate through simulations the Bit Error Rate (BER) performance of the Gallager B decoders under timing errors. We consider a BSC of parameter α and we evaluate the Gallager B performance for four regular codes defined by their degrees (d_v, d_c). The $(3, 5)$-code is of length $n = 1000$, the $(3, 6)$-code is of length $n = 504$, the $(4, 16)$-code and the $(4, 8)$-code are of length $n = 1296$.

Fig. 3 represents the BER of the Gallager B decoder for the two considered codes for $\ell = 100$ iterations and for $\varepsilon = 0$, $\varepsilon = 0.2$. For both codes, the performance of the decoder with timing errors is the same as the performance of the error-free decoder, despite the fairly large value $\varepsilon = 0.2$. This result is in accordance with Theorem 2 that shows that the asymptotic performance is the same with and without timing errors. In Fig. 3, we also see a small difference in the error floor between the curves for the error-free decoder and for the decoder with timing errors. This difference may come from the fact that our analysis assumes that the computation graphs are cycle-free. Future works will be dedicated to predicting the performance of LDPC decoders under timing errors for computation graphs with cycles. The above results confirm what was experimentally observed in [2] for the Gallager B decoder under timing errors on Latin Square (LS) codes [18].

VI. Conclusion

In this paper, we provided an analysis of the asymptotic performance of Gallager B decoders under timing errors. We showed that as the number of iterations goes to infinity, the error probability of the decoder with timing errors converges to the error probability of the error-free decoder. Monte Carlo simulations confirmed this result even for moderate code lengths. The analysis for the Gallager B decoder may be extended to other decoders such as BP, Min-Sum, etc.

Acknowlegement

This work was funded by the Seventh Framework Programme of the European Union, under Grant Agreement number 309129 (i-Risc), and by NSF under grants CCF-1314147 and ECCS-1500170, and supported by Indo-US Science and Technology Forum (IUSSTF) through the Joint Networked Center for Data Storage Research (JC-16-2014-US), and the Fulbright Scholar Program.

References

[1] A. Amaricai, S. Nimara, O. Boncalo, J. Chen, and E. Popovici, "Probabilistic gate level fault modeling for near and sub-threshold CMOS circuits," in *Proc. 17th Euromicro Conf. on Digital Syst. Design (DSD)*, Verona, 2014, pp. 473–479.

[2] S. Brkić, O. Al Rasheed, P. Ivaniš, and B. Vasić, "On fault tolerance of the Gallager B decoder under data-dependent gate failures," *IEEE Communications Letters*, vol. 19, no. 8, pp. 1299–1302, 2015.

[3] V. De, "Near-threshold voltage design in nanoscale CMOS," in *Design, Automation & Test in Europe Conference*, 2013, pp. 612–612.

[4] A. Amaricai, V. Savin, O. Boncalo, N. Cucu-Laurenciu, J. Chen, and S. Cotofana, "Timing error analysis of flooded LDPC decoders," in *IEEE International Conference on Microwaves, Communications, Antennas and Electronic Systems*, 2015, pp. 1–5.

[5] L. Varshney, "Performance of LDPC codes under faulty iterative decoding," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4427–4444, 2011.

[6] 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.

[7] 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.

[8] G. Sundararajan, C. Winstead, and E. Boutillon, "Noisy gradient descent bit-flip decoding for LDPC codes," *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3385–3400, 2014.

[9] 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.

[10] F. Leduc-Primeau and W. Gross, "Faulty Gallager-B decoding with optimal message repetition," in *Proc. 50th Annual Allerton Conference on Communication, Control, and Computing*, Oct. 2012, pp. 549–556.

[11] S. Brkić, P. Ivaniš, and B. Vasić, "Analysis of one-step majority logic decoding under correlated data-dependent gate failures," in *IEEE International Symposium on Information Theory*, 2014, pp. 2599–2603.

[12] 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.

[13] P. Ivaniš and B. Vasić, "Error errore eicitur: A stochastic resonance paradigm for reliable storage of information on unreliable media," *IEEE Transactions on Communications (in press)*, 2016.

[14] E. Sharon, S. Litsyn, and J. Goldberger, "Efficient serial message-passing schedules for LDPC decoding," *IEEE Transactions on Information Theory*, vol. 53, no. 11, pp. 4076–4091, 2007.

[15] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[16] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Univ. Linköping, Sweden, Dept. Elec. Eng., 1996.

[17] E. Janulewicz and A. Banihashemi, "Performance analysis of iterative decoding algorithms with memory over memoryless channels," *IEEE Transactions on Communications*, vol. 60, no. 12, pp. 3556–3566, 2012.

[18] D. Nguyen, S. Chilappagari, M. Marcellin, and B. Vasić, "On the Construction of Structured LDPC Codes Free of Small Trapping Sets," *IEEE Trans. Inform. Theory*, vol. 58, no. 4, pp. 2280–2302, April 2012.