Density Evolution for Min-Sum Decoding of LDPC Codes Under Unreliable Message Storage

Alexios Balatsoukas-Stimming, *Student Member, IEEE* and Andreas Burg, *Member, IEEE*

Abstract—We analyze the performance of quantized min-sum decoding of low-density parity-check codes under unreliable message storage. To this end, we introduce a simple bit-level error model and show that decoder symmetry is preserved under this model. Subsequently, we formulate the corresponding density evolution equations to predict the average bit error probability in the limit of infinite blocklength. We present numerical threshold results and we show that using more quantization bits is not always beneficial in the context of faulty decoders.

I. Introduction

RECENTLY, there has been strong interest in studying the performance of low-density parity-check (LDPC) codes under faulty decoding, which is motivated by the fact that VLSI integration has reached a level where it is difficult to guarantee fully reliable operation [1]. Specifically, in [2] the Gallager A and the sum-product algorithms are analyzed under faulty decoding and an important concentration result is proved that makes the density evolution [3] analysis meaningful. Similar analyses are provided in [4], [5] for the Gallager B algorithm, while more general finite-alphabet decoders were considered in [6]. A study of min-sum (MS) decoding [7], which is widely used in practice, under faulty computations due to unreliable logic is presented in [8].

In this work, we study MS decoding that is faulty due to unreliable memory. The reason for this approach is twofold. First, memory elements and flip-flops have been shown to be the first point of failure in digital circuits [9]. Secondly, memory is the largest part, in terms of area, of most hardware LDPC decoders (see, e.g., [10]). Thus, it is reasonable to assume that the remaining part of the decoder, which consists of logic, can be protected by standard fault-tolerance methods with small overhead. To this end, we introduce a bit-level fault model for unreliable memory reads and we prove that decoder symmetry is preserved under the aforementioned fault model. This fault model allows us to move away from the unrealistic assumption of uniformly distributed errors, which is found in all previous work.

II. Quantized Min-Sum Decoding

A. LDPC Codes and Channel Model

An LDPC code $\mathcal{C}$ of blocklength N can be defined as $\mathcal{C} \triangleq \left\{ \mathbf{c} \in \{0,1\}^N : \mathbf{Hc} = \mathbf{0} \right\}$, where $\mathbf{H}$ is a sparse matrix with $H_{ij} \in \{0,1\}$ and operations are performed over GF(2). $\mathbf{H}$ acts as an adjacency matrix for a bipartite graph which contains *variable nodes* and *check nodes*. Variable node i is connected to check node j iff $H_{ji} = 1$. If all variable nodes have degree d_v and all check nodes have degree d_c, then we say that the code is a (d_v, d_c)-regular code and that it belongs to the (d_v, d_c)-regular code *ensemble* [3].

Transmission of $\mathbf{c} \in \mathcal{C}$ over an additive white Gaussian noise (AWGN) channel using binary phase-shift keying (BPSK) modulation is modeled as

$$y_i = x_i + w_i, \ w_i \sim \mathcal{N}\left(0, \sigma^2\right), \ i = 1, \ldots, N, \quad (1)$$

where $x_i = 1 - 2c_i$. Let $L(y_i) \triangleq \ln \frac{p(y_i|x_i=+1)}{p(y_i|x_i=-1)}$ denote the channel log-likelihood ratio (LLR). The AWGN channel is output-symmetric in the sense that

$$L(-y_i) = -L(y_i). \quad (2)$$

B. Min-Sum Decoding

In MS decoding, the message sent from variable node i to check node j is calculated as

$$m_{i \to j}^{vc} = m_i^0 + \sum_{j' \in \partial i / j} m_{j' \to i}^{cv}, \quad (3)$$

where $m_i^0 = L(y_i) = y_i/2\sigma^2$ and $\partial i/j$ denotes the set of all neighboring nodes of node i except node j. The message sent from check node j back to variable node i is given by

$$m_{j \to i}^{cv} = \left(\prod_{i' \in \partial j / i} \text{sign}\left(m_{i' \to j}^{vc}\right) \right) \min_{i' \in \partial j / i} \left| m_{i' \to j}^{vc} \right|. \quad (4)$$

Decisions for codeword symbol i are taken according to

$$\hat{x}_i = \text{sign}\left(m_i^0 + \sum_{j' \in \partial i} m_{j' \to i}^{cv} \right). \quad (5)$$

Let $\Phi(m_0, m_1, \ldots, m_{d_v-1})$ and $\Psi(m_1, \ldots, m_{d_c-1})$ denote the update rules in (3) and (4), respectively. Let $b_i \in \{\pm 1\}$, $i = 1, \ldots, d_c - 1$. The following symmetries hold

$$\Phi(-m_0, -m_1, \ldots) = -\Phi(m_0, m_1, \ldots), \quad (6)$$

$$\Psi(b_1 m_1, b_2 m_2 \ldots) = \prod_{i=1}^{d_c-1} b_i \Psi(m_1, m_2 \ldots). \quad (7)$$

C. Message Quantization

We assume uniform b-bit symmetric message quantization. In LDPC decoder implementations, $4 \leq b \leq 7$ are common values (see, e.g., [10] and references therein). The quantizer is defined by

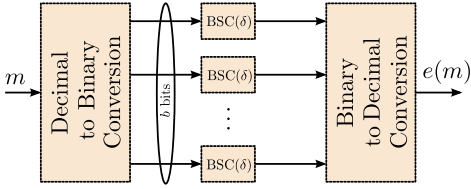$$q(x) = \text{sign}(x)\Delta \left\lfloor \frac{|x|}{\Delta} + \frac{1}{2} \right\rfloor, \quad (8)$$

Fig. 1. Message fault model: an incoming b-bit noiseless message of value m is passed through b independent BSC(δ) channels, resulting in the faulty message $e(m)$.
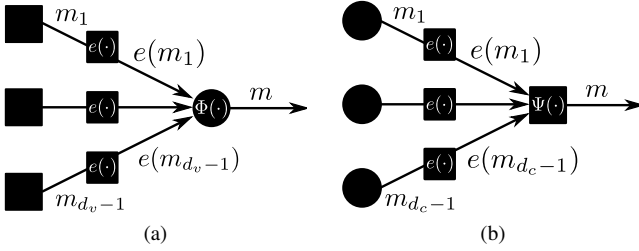

Fig. 2. Application of fault model to (a) variable node and (b) check node update rules. The fault model is a specific instance of the generic *message wires* defined in [2].

where Δ denotes the quantization step. Let the set of all quantization levels be denoted by $\mathcal{L}$, so that $q(x) \in \mathcal{L}, \ \forall x \in \mathbb{R}$. The corresponding quantization intervals are

$$t_i = \left(\frac{l_{i-1} + l_i}{2}, \frac{l_i + l_{i+1}}{2} \right], \ i = 0, \ldots, 2^b - 2, \quad (9)$$

where $l_{-1} = -\infty$ and $l_{2^b - 1} = +\infty$. Results of (3)–(5) that are smaller than l_0 or larger than $l_{2^b - 2}$ are saturated to l_0 and $l_{2^b - 2}$, respectively.

III. FAULT MODEL AND RESTRICTION TO THE ALL-ONE BPSK CODEWORD

A. Fault Model

We assume sign-magnitude binary representation for all message values $m \in \mathcal{L}$. The memory read errors are modeled as independent and identically distributed (i.i.d.) random bit-flips. We assume that the memory errors are caused by single-event upsets, to which memory that is produced at very small lithographic nodes is particularly sensitive. Thus, all faults are transient, as in [2], and independent of the stored message. More precisely, at each iteration, each bit of the binary representation of the messages used to compute (3)–(5) is passed through a binary symmetric channel (BSC) with crossover probability δ, denoted by BSC(δ). We denote the set of all possible binary error patterns by $\mathcal{E}$ and the resulting faulty message after applying $e \in \mathcal{E}$ to a message of value m by $e(m)$. The distribution of the error patterns is

$$\mathbb{P}(e) = \delta^{\mathrm{w_H}(e)} (1 - \delta)^{b - \mathrm{w_H}(e)}, \quad e \in \mathcal{E}, \quad (10)$$

where $\mathrm{w_H}(e)$ denotes the Hamming weight of e. The fault model and its application to MS decoding are illustrated in Fig. 1 and Fig. 2, respectively.

B. Restriction to the All-One BPSK Codeword

Under the update rule symmetry defined in (6) and (7) and under channel symmetry, as defined in (2), the probability of bit error is independent of the transmitted codeword [3]. Thus, the asymptotic analysis of MS decoding can be restricted to the all-one BPSK codeword. The following proposition ensures that the same simplification can be applied to faulty quantized MS decoding with our error model.

Proposition 1. *When messages are represented in sign-magnitude form, MS decoder symmetry is preserved under faulty decoding with read errors modeled as i.i.d. bit-flips.*

Proof: Due to quantizer symmetry, we have $q(L(-y_i)) = q(-L(y_i)) = -q(L(y_i))$, so channel symmetry holds. Moreover, when using sign-magnitude representation where "$+0$" and "-0" exist as distinct values, it holds that

$$e(-m) = -e(m), \quad \forall e \in \mathcal{E}, \ m \in \mathcal{L}. \quad (11)$$

Thus, for the variable node update rule, we have

$$\Phi(e(-m_0), e(-m_1), \ldots) = \Phi(-e(m_0), -e(m_1), \ldots) \quad (12)$$
$$= -\Phi(e(m_0), e(m_1), \ldots). \quad (13)$$

Similarly, for $b_i \in \{\pm 1\}, \ i = 1, \ldots, d_c - 1$, we have

$$\Psi(e(b_1 m_1), e(b_2 m_2), \ldots) = \Psi(b_1 e(m_1), b_2 e(m_2), \ldots) \quad (14)$$
$$= \prod_{i=1}^{d_c - 1} b_i \Psi(e(m_1), e(m_2), \ldots), \quad (15)$$

meaning that update rule symmetry holds for both variable nodes and check nodes. Moreover, we assume that, whenever $m = 0$ appears, a uniform random choice between "$+0$" and "-0" is made, so that the bit error rate when $m = 0$ is always $1/2$ independently of the codeword bit value.

IV. DENSITY EVOLUTION

Density evolution (DE) tracks the average probability density functions of the messages exchanged between the variable and check nodes at each decoding iteration in the limit of infinite blocklength [3]. DE operates under the assumption that all messages are independent because it can be shown that the decoding graph is asymptotically cycle-free. A concentration result guarantees that the performance of individual codes chosen from an ensemble is close to the ensemble average performance with high probability [3].

The existence of transient errors using the error model introduced in Section III does not affect the asymptotic cycle-free property of the decoding graph. However, care still needs to be taken to ensure that all messages involved in decoding are independent. With our error model the faulty messages are independent, because the corresponding non-faulty messages from which they are derived are independent and the errors affecting a specific message are independent of the message value. A concentration result that is similar to the concentration result for noiseless decoders was proved in [2].

$$\Phi_+^\ell(m) = \sum_{\substack{k=0, \\ k \text{ even}}}^{d_c-1} \binom{d_c-1}{k} \left(A_+^\ell(m)\right)^k \left(A_-^\ell(m)\right)^{d_c-k-1}, \qquad \Phi_-^\ell(m) = \sum_{\substack{k=0, \\ k \text{ odd}}}^{d_c-1} \binom{d_c-1}{k} \left(A_+^\ell(m)\right)^k \left(A_-^\ell(m)\right)^{d_c-k-1} \quad (16)$$

A. Density Evolution for Quantized MS Decoding

Let $p^\ell(m)$ and $q^\ell(m)$ denote the probability mass functions (PMFs) of the variable-to-check and the check-to-variable messages at iteration $\ell \geq 1$, respectively, and let $p^0(m)$ denote the PMF of the channel LLR messages assuming that the all-one BPSK codeword was transmitted. We have

$$p^0(l_i) = \frac{1}{\sqrt{8\pi\sigma^{-2}}} \int_{t_i} e^{-\left(x-\frac{2}{\sigma^2}\right)^2 \cdot \frac{\sigma^2}{4}} dx, \quad \forall l_i \in \mathcal{L}. \quad (17)$$

The check-to-variable message density is given by

$$q^\ell(m) = \begin{cases} \Phi_-^\ell(m) - \Phi_-^\ell(m-1), & m < 0, \\ 1 - \left(1-p^\ell(0)\right)^{d_c-1}, & m = 0, \\ \Phi_+^\ell(m+1) - \Phi_+^\ell(m), & m > 0, \end{cases} \quad (18)$$

where $\Phi_-^\ell(m)$ and $\Phi_+^\ell(m)$ are defined in (16) and

$$A_+^\ell(m) = \sum_{x=m}^{l_{2^b}-2} p^\ell(x), \quad m > 0, \quad (19)$$

$$A_-^\ell(m) = \sum_{x=l_0}^{m} p^\ell(x), \quad m < 0. \quad (20)$$

The variable-to-check message density is given by

$$p^\ell(m) = p^0(m) \otimes \left(q^{(\ell-1)}(m)\right)^{\otimes(d_v-1)}, \quad (21)$$

where $\otimes$ denotes the convolution and $q^0(m) = \delta[m]$, where $\delta[m]$ is the Kronecker delta function. The density of the quantity used for bit decisions is given by

$$d^\ell(m) = p^0(m) \otimes \left(q^{(\ell-1)}(m)\right)^{\otimes d_v}. \quad (22)$$

When applying (21) and (22), any probability mass that corresponds to values smaller than l_0 or larger than $l_{2^b}-2$ is added to the mass corresponding to l_0 or $l_{2^b}-2$, respectively.

B. Density Evolution for Faulty Quantized MS Decoding

Let $f_\delta(P)(m)$ denote the probability of a faulty message m, $m \in \mathcal{L}$, where P is the distribution of the non-faulty messages m', i.e., P can be p^ℓ or q^ℓ. We have

$$f_\delta(P)(m) = \sum_{\substack{e \in \mathcal{E}, m' \in \mathcal{L}: \\ e(m')=m}} P(m')\mathbb{P}(e). \quad (23)$$

For each value m, there are 2^b pairs (e, m') such that $e(m') = m$. Since there are $2^b - 1$ values for m,[1] evaluating $f_\delta(P)$ requires the calculation of approximately 2^{b+1} terms.

Unreliable memory reads cause errors in the *input* messages of (3)–(5). Thus, DE for faulty MS decoding with transient memory read errors can be formulated by replacing the p^ℓ and q^ℓ distributions that appear on the right-hand side of (18)–(22) with $f_\delta\left(p^\ell\right)$ and $f_\delta\left(q^\ell\right)$, respectively.

---

[1] Recall that the decimal value 0 corresponds to two binary patterns (i.e., "+0" and "−0").

C. Bit Error Probability and Threshold

Let $P_e^\ell(\sigma^2)$ denote the bit error probability at iteration ℓ when transmission takes place over an AWGN channel with noise variance σ^2, defined as

$$P_e^\ell(\sigma^2) \triangleq \frac{1}{2}d^\ell(0) + \sum_{m=l_0}^{l_{2^b-1}-2} d^\ell(m). \quad (24)$$

The threshold for faulty decoding is defined as [2]

$$\sigma_*^2(\eta) \triangleq \sup\left\{\sigma^2 \geq 0 : \lim_{\ell \to \infty} P_e^\ell(\sigma^2) \leq \eta\right\}. \quad (25)$$

For all examined (d_v, d_c)-regular ensembles and σ^2, our numerical results show that $P_e^\ell\left(\sigma^2\right) \geq \delta$. This observation can help us in choosing a meaningful value for η. Specifically, we choose $\eta = \alpha\delta$, for some $\alpha > 1$. If α is chosen so that $\alpha\delta$ lies within the waterfall region of the code, then the value of α does not have a significant effect on the computed threshold. To make the dependence on α and δ explicit, we denote the threshold by $\sigma_*^2(\alpha, \delta)$.

V. NUMERICAL RESULTS

A. Decoding Threshold

We computed $\sigma_*^2(\alpha, \delta)$ under MS and faulty MS decoding for various (d_v, d_c)-regular ensembles of rate 0.5 and for various values of δ, with $\alpha = 10$ and $b = 5$ bits. The maximum number of iterations is set to $\ell_{\max} = 200$. Quantization is performed with $\Delta = 1$. For fair comparison, the definition in (25) was used for both MS and faulty MS decoding.

The results are summarized in Table I. The evolution of $P_e^\ell(\sigma^2)$ as a function of ℓ for the $(3,6)$-regular ensemble and for two indicative cases of $\delta = 10^{-5}$ and $\delta = 10^{-6}$ under MS and faulty MS decoding is presented in Fig. 3 and Fig. 4, respectively. The error floor for faulty MS decoding is very apparent. This visualization also enables us to calculate the overhead, in terms of additional iterations, introduced by faulty decoding. In Fig. 3, the faulty MS decoder for $\delta = 10^{-5}$ requires 64 more iterations than the MS decoder to achieve the same bit error probability when operating at the faulty MS decoder's threshold. In Fig. 4, we see that for a smaller δ, the difference in iterations is smaller, as intuitively expected.

It is interesting to note that the threshold generally decreases when d_v and d_c are increased, but the resulting code ensembles seem to be more resilient to errors. The loss in $\sigma_*^2(\alpha, \delta)$ as δ is increased is smaller for larger (d_v, d_c) pairs. This interesting trade-off between threshold and error resilience motivates the design of irregular LDPC codes for faulty MS decoding.

B. Are More Bits Always Better?

In faulty decoding it can not be claimed in advance that increasing the number of quantization bits b will result in better
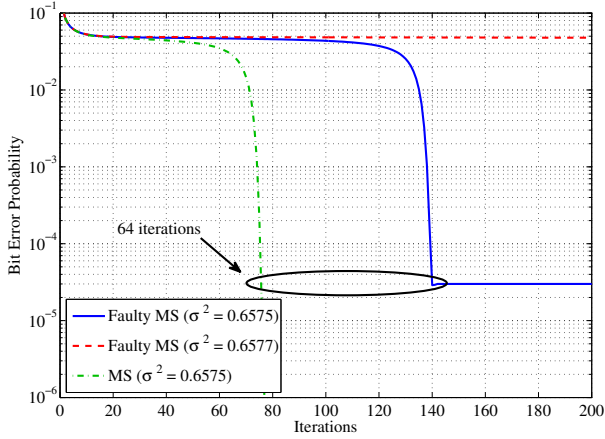
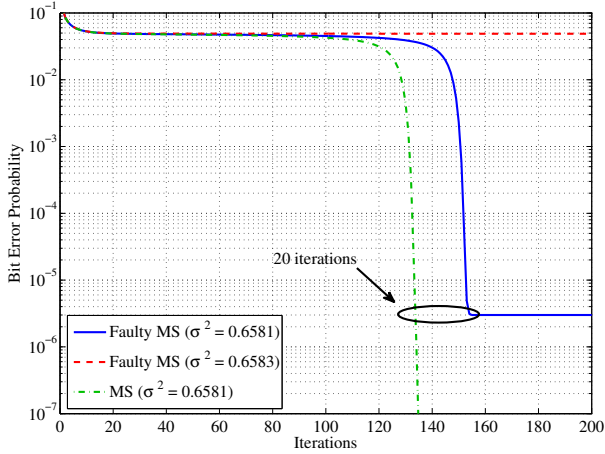Fig. 3. Error probability for a $(3,6)$-regular LDPC code under faulty MS and MS decoding for $\delta = 10^{-5}$.

Fig. 4. Error probability for a $(3,6)$-regular LDPC code under faulty MS and MS decoding for $\delta = 10^{-6}$.

TABLE I
THRESHOLDS OF VARIOUS (d_v, d_c)-REGULAR CODES UNDER MS AND
FAULTY MS DECODING FOR $\alpha = 10$ AND $b = 5$ BITS.

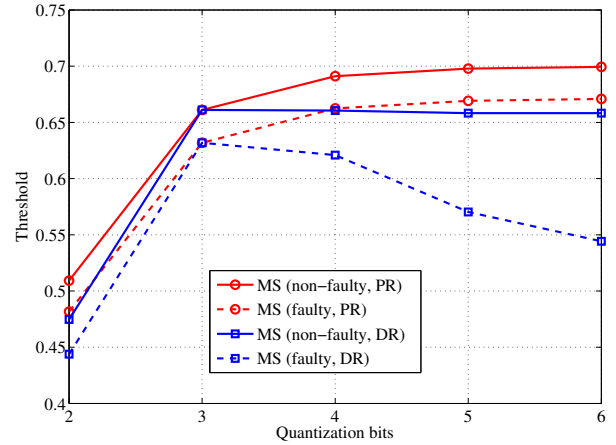| (d_v, d_c)		δ	10^{-3}	10^{-4}	10^{-5}	10^{-6}
(3,6)	MS	$\sigma_*^2(\alpha, \delta)$	0.6579	0.6579	0.6579	0.6582
	F-MS	$\sigma_*^2(\alpha, \delta)$	0.5703	0.6518	0.6576	0.6582
(4,8)	MS	$\sigma_*^2(\alpha, \delta)$	0.5486	0.5486	0.5486	0.5486
	F-MS	$\sigma_*^2(\alpha, \delta)$	0.5077	0.5446	0.5482	0.5486
(5,10)	MS	$\sigma_*^2(\alpha, \delta)$	0.4793	0.4793	0.4793	0.4793
	F-MS	$\sigma_*^2(\alpha, \delta)$	0.4473	0.4761	0.4790	0.4792
(6,12)	MS	$\sigma_*^2(\alpha, \delta)$	0.4320	0.4320	0.4320	0.4320
	F-MS	$\sigma_*^2(\alpha, \delta)$	0.4041	0.4292	0.4317	0.4320

Fig. 5. Error probability for a $(3,6)$-regular LDPC code under faulty MS and MS decoding for $\delta = 10^{-3}$ for different numbers of quantization bits.

performance of faulty MS decoding can actually degrade with increasing b. Thus, we conclude that more bits are, perhaps counter-intuitively, not always better in the context of faulty MS decoding.

performance, because by increasing b we also increase the number of faults in the decoder. The additional bits can be used either to increase the dynamic range (DR) or to increase the precision (PR) of the messages. The DR case corresponds to quantization with a fixed step size, while in the PR case the quantization step is a function of b.

In Fig. 5 we present indicative threshold results for the DR case with $\Delta_{\mathrm{DR}} = 1$, as used in Section V-A, and for the PR case with $\Delta_{\mathrm{PR}} = 2^{3-b}$, which we empirically found to provide good performance with fault-free decoding in both cases, and $\delta = 10^{-3}$ and $\alpha = 10$. We observe that increasing the dynamic range does not offer any benefits after $b = 3$ for fault-free decoding in the examined scenario, and that PR quantization provides better performance than DR quantization. More importantly, however, in the DR case the performance actually degrades for $b \geq 3$. This behavior can be explained intuitively as follows. In the DR case, bit-flips in the additional bits cause increasingly larger errors in the message values, whereas in the PR case these errors become smaller when b is increased.

We have demonstrated that, if one is not careful, the

REFERENCES

[1] S. Ghosh and K. Roy, "Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1718–1751, Oct. 2010.

[2] L. Varshney, "Performance of LDPC codes under faulty iterative decoding," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4427–4444, Jul. 2011.

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

[4] S. M. Sadegh Tabatabaei Yazdi, H. Cho, and L. Dolecek, "Gallager B decoder on noisy hardware," *IEEE Trans. Commun.*, vol. 61, no. 5, pp. 1660–1672, May 2013.

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

[6] C.-H. Huang and L. Dolecek, "Analysis of finite-alphabet iterative decoders under processing errors," in *IEEE Int. Conf. Acoustics, Speech, Sig. Proc.*, May 2013, pp. 5085–5089.

[7] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping University, Linköping, Sweden, 1996.

[8] C. Kameni Ngassa, V. Savin, and D. Declercq, "Min-sum-based decoders running on noisy hardware," in *IEEE GLOBECOM*, Dec. 2013.

[9] A. Chandrakasan, D. Daly, D. Finchelstein, J. Kwong, Y. Ramadass, M. Sinangil, V. Sze, and N. Verma, "Technologies for ultradynamic voltage scaling," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 191–214, Feb. 2010.

[10] C. Studer, N. Preyss, C. Roth, and A. Burg, "Configurable high-throughput decoder architecture for quasi-cyclic LDPC codes," in *Asilomar Conf. on Signals, Systems and Computers*, Oct. 2008, pp. 1137–1142.