Error Floors in LDPC Codes: Fast Simulation, Bounds and Hardware Emulation

Pamela Lee[†], Lara Dolecek[‡], Zhengya Zhang[†], Venkat Anantharam[†], Borivoje Nikolic[†], and Martin J. Wainwright[†]
[†]EECS Department University of California, Berkeley, CA, 94720, USA
[‡]EECS Department, Massachusetts Institute of Technology, Cambridge, MA, 02139, USA
pam_lee@berkeley.edu, dolecek@mit.edu, {zyzhang,ananth,bora,wainwrig}@eecs.berkeley.edu

***Abstract*— The error-correcting performance of low-density parity check (LDPC) codes, when decoded using practical iterative decoding algorithms, is known to be very close to Shannon limits in the asymptotic limit of large blocklengths. A substantial limitation to the use of finite-length LDPC codes is the presence of an error floor in the low frame error rate (FER) region. This paper develops two methods, a stochastic one based on importance sampling and a deterministic one based on high SNR asymptotics, as applied to suitably defined absorbing structures within the LDPC code, to predict error floors. Our results are in very close agreement with hardware-based experimental results, and moreover extend the prediction of the error probability to as low as 10^{-30}. Our deterministic estimates are guaranteed to be a lower bound to the error probability in the high SNR regime.**

I. Introduction

The class of low-density parity check (LDPC) codes was first introduced by Gallager [8], and has recently been the focus of intensive study [12]. These codes are attractive because they yield excellent error-correction, getting close to the Shannon limit for large blocklengths, even when decoded using suboptimal but practical message-passing algorithms.

However, for intermediate block-lengths and low bit error rates that are of primary interest in practical systems, many LDPC codes exhibit an *error floor*, which corresponds to a decrease in the slope in the plot of bit error rate (BER) versus signal-to-noise ratio (SNR). The error floor is commonly attributed to the suboptimality of the iterative decoding algorithms on graphs with cycles, and past work has studied concepts such as near-codewords [10], trapping sets [11], pseudocodewords [7], and elementary trapping sets [9]. In our own previous work, we have introduced the notion of *absorbing sets* as the main cause of the error floor of structured LDPC codes. These absorbing sets are a specific type of near-codewords [10] or trapping sets [11] that are stable under bit-flipping operations. It can be shown [3] that the factor graphs associated with certain structured LDPC codes contain absorbing sets which have strictly fewer bits than the minimum codeword weight. The performance of an iterative decoding algorithm in the low BER region is predominantly dictated by the number and the structure of the smallest absorbing sets, in contrast to the performance of a maximum-likelihood decoder governed by the minimum distance codewords.

In early work on error floors, Richardson [11] developed a fast-simulation method, based on using simulation traces of a hardware emulator to extract trapping set candidates, and then using an approximate integration technique to estimate the associated error probability. This method involves a simulation over a sequence of channel noise realizations, suitably biased towards a candidate trapping set. Another approach to predict the performance utilizes a variant of importance sampling, where typically the simulations are performed under a single but carefully chosen biasing density. This approach also obviates the need for analytical approximations.

Since the pioneering work by Richardson [11], the problem of efficiently predicting the error-floor behavior of graph-based codes has spurred significant research activity. Other past work has proposed various types of importance sampling schemes based on trapping sets [2], [5], [15], [16]. We work directly with combinatorial structures of factor graphs referred to as absorbing sets. An absorbing set is defined independently of the particular decoding scheme or channel noise model, and as such can be studied analytically [3]. Our results also demonstrate a very close agreement with the experimental results obtained on a hardware emulator [17] for low probabilities of error. Our past work [4] introduced a method based on exact enumeration of absorbing sets and importance sampling. In this work, we both refine this importance-sampling-based approach, and develop an analytic approach for lower bounding the error probability. We illustrate its behavior for various codes, and quantized forms of sum-product decoding, showing good agreement with hardware-based experimental results, as well as theoretical predictions that extend to probabilities of error as low as 10^{-30}.

II. Background

A low-density parity check code of blocklength n can be represented as the null space of a sparse parity check matrix $H \in \{0,1\}^{m\times n}$. It is convenient to view the code in terms of its Tanner graph [14]: a bipartite graph $G = (V, F, E)$ in which $V = \{1, \ldots, n\}$ index the code bits and $F = \{1, \ldots, m\}$ index the code checks, and $E = \{e(i,j) | H(j,i) = 1\}$.

To keep the paper self-contained, we briefly define the notion of an absorbing set; see the papers [3], [4] for more details. For a subset D of V, let $\mathcal{E}(D)$ (resp. $\mathcal{O}(D)$) be the set of neighboring vertices of D in F in the graph G with even (resp. odd) degree with respect to D. Given an integer pair (a, b), an (a, b) fully absorbing set is a subset D of V of size a, with $\mathcal{O}(D)$ of size b and with the property that each

978-1-4244-2571-6/08/$25.00 ©2008 IEEE

element of V has strictly fewer neighbors in $\mathcal{O}(D)$ than in $F \backslash \mathcal{O}(D)$.

Array-based LDPC code constructions [6] are a representative of a class of high-performing structured LDPC codes. It is known [3] that these codes have absorbing sets that are strictly smaller than the minimum distance of the code; moreover, results from hardware emulation show that their low BER performance and the error floor are indeed dominated by these absorbing sets [17]. These codes will be used in the remainder of the paper to conveniently illustrate the methodology developed here. We consider the sum-product decoding algorithm, and BPSK signalling (with mapping $0 \to 1$, $1 \to -1$) over the additive white Gaussian noise (AWGN) channel. The AWGN channel model assumes that the channel input $x_i \in \{-1, 1\}$, corresponding to the ith bit in the transmitted codeword, is received as $y_i = x_i + w_i$, where w_i is a zero-mean Gaussian signal with variance σ^2, itself independent of the channel input and Gaussian signals associated with other x_j for $i \neq j$. The exact equations describing message updates of this algorithm are by now standard and well-known [12]. The sum-product algorithm is typically allowed to run for a fixed number of iterations, both because convergence is not guaranteed when cycles are present, and due to practical (delay) constraints. For practical hardware implementations, the real-valued messages in the sum-product decoding algorithm are necessarily quantized, and we present results for various quantization schemes.

III. Importance sampling and hardware emulation

In this section, we describe an importance sampling method for predicting error probabilities, based on mean-shifting the original Gaussian density towards an absorbing set of interest. We also derive approximate confidence intervals for these stochastic estimates, and show that hardware-based experimental results fall well within 95% confidence intervals.

Importance sampling (IS) is a particular type of Monte Carlo method which uses statistical sampling to approximate analytic expressions of probabilities. The basic idea is to perform simulation under a tilted distribution so as to make the event of interest more likely; the averages are then reweighted to compensate for the tilting. Supposing without loss of generality that the all-zeros word is transmitted, let $Y^{(1)}, \ldots, Y^{(M)}$ be a set of M trials, each $Y^{(i)} \in \mathbb{R}^n$ sampled in an i.i.d. manner from a biased distribution f_{bias}. The associated IS estimate of a particular absorbing probability is given by

$$\hat{q}_{IS} = \frac{1}{M} \sum_{i=1}^{M} \mathbb{I}_{\text{error}}(Y^{(i)}) w(Y^{(i)}),$$

where $\mathbb{I}_{\text{error}}$ is a $0-1$-valued indicator function for whether the decoder converges to the given absorbing set on trial $Y^{(i)}$, and $w(Y^{(i)}) = \frac{f(Y^{(i)})}{f_{\text{bias}}(Y^{(i)})}$ is the appropriate weighting function to produce an unbiased estimate.

In the case of the all-zeros codeword being transmitted in a BPSK - modulated Gaussian channel, the original density f is an n-variate Gaussian $N(\vec{1}_n, \sigma^2 I_{n \times n})$. A suitable choice [4] of f_{bias} is the mean-shifted Gaussian $N(\nu(\mu), \sigma^2 I)$, where $\nu(\mu)_i = 1 - \mu$ for elements inside the absorbing set, and $\nu(\mu)_i = 1$ otherwise. With this choice, the IS weight w is given by

$$w(Y^{(i)}|\mu, \sigma^2) = \frac{e^{-\frac{1}{2\sigma^2}[\sum_{k=1}^{a}(Y_k^{(i)}-1)^2]}}{e^{-\frac{1}{2\sigma^2}[\sum_{k=1}^{a}(Y_k^{(i)}-(1-\mu))^2]}},$$

where $Y_k^{(i)}$ are the values of the nodes in the absorbing set.

Due to symmetry of the code and channel, the probability of error—that is, the *absorbing probability*—of any fixed (a, b) absorbing set is equal to that of any other exemplar having the same absorbing set structure. Since the associated events are disjoint, the error probability p associated with all (a, b) absorbing sets is equal to $p = N_{\text{sets}}\, q$, where N_{sets} is the total number of (a, b) absorbing sets of the same structure. The associated IS estimate of p is given by $p \approx \hat{p}_{\text{IS}} = N_{\text{sets}}\, \hat{q}_{IS}$. Note that since we are using $\hat{q}_{IS}$ as an approximation for q, where q denotes the error probability due to a fixed representative of the class of isomorphic (a, b) absorbing sets, the quantity $\hat{p}_{\text{IS}}$ is an approximation for p, and not guaranteed to either upper or lower bound the true error probability. The total number of (a, b) absorbing sets in a given array-based LDPC code can be found using the technique discussed in [3]. When the error floor is dominated by a particular isomorphic sub-class of (a, b) absorbing sets, the associated absorbing probability can be used to estimate the error floor.

In order to evaluate the accuracy of the importance sampling error curves, we calculate 95% confidence intervals for points on the curves. The standard scale for comparing FER curves is $\log_{10}(\hat{p}_{\text{IS}})$ and $\log_{10}(p)$ versus SNR; accordingly, we first find the variance of $\log_{10}(\hat{p}_{\text{IS}})$. The variance of the estimator $\hat{p}_{\text{IS}}$ is given by $\text{var}(\hat{p}_{\text{IS}}) = \mathbb{E}[(\hat{p}_{\text{IS}})^2] - p^2$.

Letting $\xrightarrow{d}$ denote convergence in distribution, the delta-method [13] states that if some sequence of random variables $\{X_M\}$ satisfies $\sqrt{M}(\bar{X}_M - \mu) \xrightarrow{d} \mathbb{N}(0, \sigma^2)$, then for any twice-differentiable function f, we have

$$\sqrt{M}(f(\bar{X}_M) - f(\mu)) \xrightarrow{d} \mathbb{N}(0, [f'(\mu)]^2 \sigma^2).$$

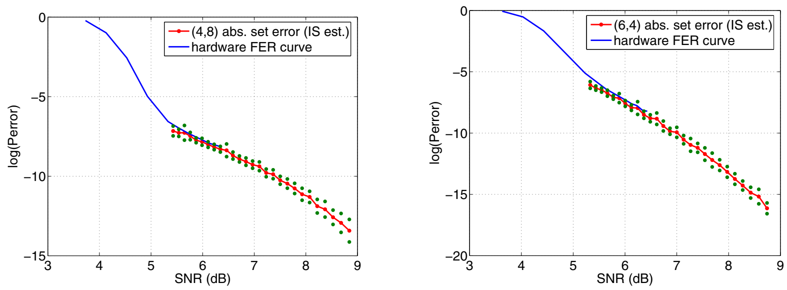
Note that by the central limit theorem, we have $\sqrt{M}(\hat{p}_{\text{IS}} - p) \xrightarrow{d} \mathbb{N}(0, \beta^2)$ where $\beta^2 = M\, \text{var}(\hat{p}_{\text{IS}})$. By applying the delta-method to $f(x) = \log_{10}(x)$, we obtain

$$\sqrt{M}(\log(\hat{p}_{IS}) - \log(p)) \xrightarrow{d} \mathbb{N}\Big(0, \frac{\beta^2}{p^2 (\ln 10)^2}\Big).$$

In practice, we cannot make direct use of this asymptotic result, since we know neither $\beta^2 = M\, \text{var}(\hat{p}_{\text{IS}})$ nor p. However, we can estimate these quantities by their sample versions—namely, $p \approx \hat{p}_{\text{IS}}$, and

$$\beta^2 \approx N_{\text{sets}}^2 \left(\frac{1}{M} \sum_{i=1}^{M} \mathbb{I}_{\text{error}}(Y^{(i)}) w^2(Y^{(i)}) - \hat{q}_{IS}^2 \right).$$

thereby obtaining the approximation $\widehat{\text{var}}(\log(\hat{p}_{IS})) \approx \frac{\hat{\beta}^2}{\hat{p}_{\text{IS}}^2 M\, (\ln 10)^2}$. Finally, using this approximation, we can compute an approximate 95% confidence interval for $\log(\hat{p}_{IS})$ by



(a) (b)

Fig. 1. Comparisons of importance sampling estimates of error probabilities with hardware-based emulation results. (a) $(4, 8)$ absorbing set of the $(2209, 1978)$ array-based LDPC code (b) $(6, 4)$ absorbing set of the $(2209, 2024)$ array-based LDPC code. Dots below and above correspond to approximate 95% confidence intervals; see text for details.

applying Chebyshev's inequality:

$$\mathbb{P}(|\log(\hat{p}_{\mathrm{IS}}) - \log(p)| \geq \epsilon) \quad \leq \quad \frac{\mathrm{var}(\log(\hat{p}_{\mathrm{IS}}))}{\epsilon^2} \approx \frac{\widehat{\mathrm{var}}(\log(\hat{p}_{\mathrm{IS}}))}{\epsilon^2}.$$

Thus, the approximate 95% confidence interval for $\log(\hat{p}_{\mathrm{IS}})$ is $[\log(\hat{p}_{\mathrm{IS}}) \pm \epsilon]$, where $\epsilon = \sqrt{\widehat{\mathrm{var}}(\log(\hat{p}_{\mathrm{IS}}))/.05}$.

Figure 1 provides a comparison of the importance-sampling (IS) predictions to results from hardware emulation [17] for two different codes, having different absorbing sets. The dots surrounding the IS curves correspond to the 95% confidence interval; note how the IS prediction shows very good agreement with the hardware emulation.

IV. Deterministic lower bounds and channel invariance

Applying the IS method requires substantially less computation than direct simulation, but the computation must be re-performed each time that the channel parameters are changed. In this section, we describe a procedure for generating analytical expressions, guaranteed to lower bound the error probability in the high SNR regime, and that require substantially less computation. At a high level, our procedure consists of the following three steps.

(a) First, we compute an inner bound to the *absorbing region* associated with an absorbing set from a particular class. This step is a purely deterministic computation, and is independent of the underlying channel model.
(b) Second, using a generalized Chernoff approximation [1], we compute an analytical approximation to the *absorbing probability*: i.e., the probability under the specified channel model that the log-likelihood vector falls into this given absorbing region. This approximation is guaranteed to lower bound the true absorbing probability in the high SNR regime.
(c) Third, we combine the absorbing probability with a count (or lower bound) on the cardinality of the class of absorbing sets. The combination of these three ingredients yields deterministic approximations to the error probability, guaranteed to be lower bounds in the high SNR regime.

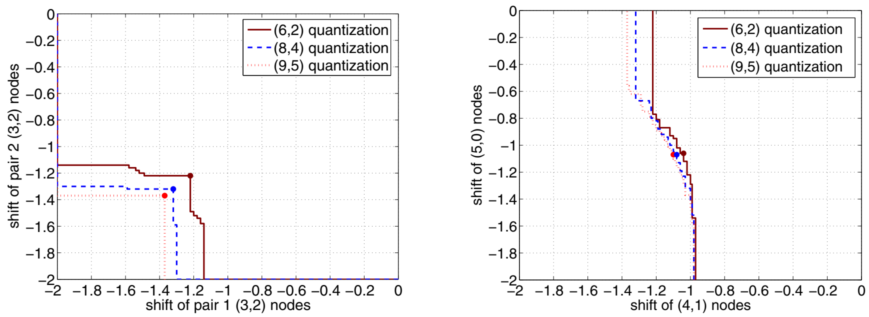
We note that this procedure can be carried out for *any channel*. Indeed, once the absorbing regions and counts in steps one and three have been calculated once, this computation need not be repeated for additional channels, since the only channel-dependent quantity is the bound calculation in the second step.

A. Absorbing regions for decoders

Consider some fixed decoder (e.g., floating-point sum-product, quantized sum-product, or a bit-flipping decoder) that operates on an LDPC code of blocklength n. On any given trial, the decoder is initialized with a log-likelihood ratio (LLR) vector $z \in \mathbb{R}^n$ of observations at each bit node. After some fixed number of iterations, the estimated LLRs of the decoder are thresholded, yielding a $\{0, 1\}^n$ sequence that is an estimate of the transmitted codeword. The absorbing region $\mathcal{R}$ of a given absorbing set is the set of LLR vectors $z \in \mathbb{R}^n$ for which the decoder outputs the indicator vector of the absorbing set as its estimate.

Two properties of this absorbing region are important. First, it is a *channel-independent* quantity, since it is only a function of the initializing LLR vector $z \in \mathbb{R}^n$. (Although z is a random vector, with distribution dependent on the channel, conditioned on a particular initialization, the decoder's behavior is purely deterministic, and hence channel-independent.) Second, it varies as features of the decoder (e.g., quantization scheme etc.) are changed; indeed, the relative size of the absorbing region is a measure of its impact on a particular decoder.

Exact computation of the absorbing region is prohibitively expensive, since it involves testing the decoder over an n-dimensional space. (For instance, discretizing each dimension to ℓ locations, yields the complexity $\mathcal{O}(\ell^n)$.) In practice, we are forced to seek approximations to the absorbing region, based on examining particular low-dimensional projections $\widetilde{\mathcal{R}}$ of the absorbing region. To approximate the exact absorbing region, we first divide the bit nodes of the absorbing set into groups of nodes with the same number of satisfied and unsatisfied check nodes. For example, the $(8, 6)$ absorbing set in the $(2209, 1978)$ array code is made up of six variable nodes of type $(4, 1)$ and two variable nodes of type $(5, 0)$, where any $(4, 1)$ node is connected to four satisfied check nodes and one unsatisfied check node and any $(5, 0)$ node is connected to five satisfied checks and zero unsatisfied checks. Each of these groups of nodes is associated with an axis. All bit nodes are initially assigned value '1' (the all-zeroes codeword under BPSK). In the two-dimensional case, the region is found by



(a) (b)

Fig. 2. Subsets of the signal space regions for which the decoder converges to the absorbing set. These plots show particular projections of these absorbing regions. (a) Absorbing region for a $(4, 8)$ absorbing set in the $(2209, 1978)$ array code. The plot shows three different levels of quantization: $(6, 2)$, $(8, 4)$, and $(9, 5)$ fixed-point quantization [17]. Note how the absorbing region contracts as the quantization scheme improves. (b) Absorbing region for a $(8, 6)$ absorbing set in the $(2209, 1978)$ array code.

trying all combinations of shifts for the two groups of nodes, where each axis ranges from the absorbing set (centered at a shift of -2) to the all-zeros codeword (centered at a shift of 0), separated by increments of size 0.01. The decoder is run for 50 iterations, and if the hard decision at the end of these iterations is the absorbing set, then we include this combination of shifts of the absorbing set nodes in a set S. The approximation to the absorbing region is this set S of points that decode to the absorbing set.

Figure 2 illustrates some approximations of absorbing regions, for different decoders and different absorbing sets. Each panel is a two-dimensional plot, with the upper right $(0, 0)$ point corresponding to receiving the all-zeros codeword (without any noise), and the lower left $(-2, -2)$ centered on the absorbing set. The marked contours correspond to the boundary between not decoding to and decoding to the appropriate absorbing set (towards lower left). Panel (a) shows regions for the $(4, 8)$ absorbing set, taken from the Tanner graph of the $(2209, 1978)$ array-based LDPC code, for three different quantized forms of sum-product (details of the quantization choices are in [17]). The four variable nodes in the $(4, 8)$ absorbing set are all $(3, 2)$ nodes, so the four nodes are randomly divided into two pairs in order to show a two-dimensional plot, which highlights the symmetry of variable nodes of the same type and thus supports the method of grouping nodes of one type into one axis. Note how the size of the absorbing region shrinks as the quantization is improved.

The effect of better quantization on absorbing regions is also seen in panel (b), which shows absorbing regions for the $(8, 6)$ absorbing set of the same code under three different quantization schemes. The effect of a finer quantization scheme is more pronounced when the bits in the absorbing set have only marginally more neighboring satisfied versus unsatisfied checks since the additional bits used to represent the messages can more easily help favorable messages overpower the unfavorable messages, as is the case for the $(4, 8)$ absorbing sets.

B. Asymptotics for absorbing probability

Whereas the absorbing region itself is channel-independent, its effect on decoding depends heavily on the channel. In particular, if we view the decoder's input vector $Z \in \mathbb{R}^n$ as random, drawn from a distribution specified by the channel model, then the absorbing set's influence corresponds to the probability mass assigned to the absorbing region. Let γ denote the signal-to-noise ratio of the channel, and for a given absorbing region $\mathcal{R}$, consider the absorbing probability $\mathbb{P}_\gamma[Z \in \mathcal{R}] = \int_{\mathcal{R}} f_Z(z; \gamma) dz$, where $\mathbb{P}_\gamma$ denotes the fact that the channel probability distribution varies as a function of the SNR, and $f_Z(z; \gamma)$ is the density of the channel under consideration. Assuming that Z has a moment generating function $\mathbb{E}[\exp(\lambda^T Z)]$ for some neighborhood of zero, we may use the (generalized) Chernoff approximation [1]:

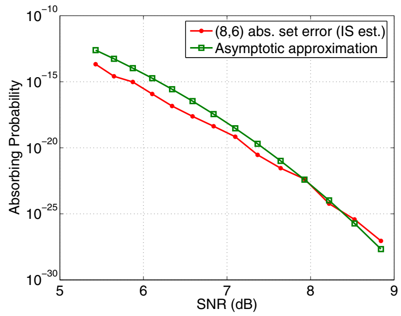
$$\log \mathbb{P}_\gamma[Z \in \mathcal{R}] \approx \inf_{\lambda > 0} \left\{ S_{\mathcal{R}}(-\lambda) + \log \mathbb{E}[\exp(\lambda^T Z)] \right\} \quad (1)$$

where $S_{\mathcal{R}}(-\lambda) := \sup_{u \in \mathcal{R}}(-\lambda^T u)$ is the support function of the set $\mathcal{R}$. As an example, when Z is multivariate Gaussian with mean zero and signal-to-noise parameter $\gamma = 1/\sigma^2$, from (1) we obtain the approximation $\log \mathbb{P}_\gamma[Z \in \mathcal{R}] \approx -\frac{1}{2\sigma^2} d^2(\vec{0}; \mathcal{R})$, where $d(\vec{0}; \mathcal{R}) = \min_{u \in \mathcal{R}} \|u\|_2$ denotes the minimum Euclidean distance from $\vec{0}$ to $\mathcal{R}$. This approximation becomes exact as the signal-to-noise $\gamma = 1/\sigma^2$ tends to infinity.

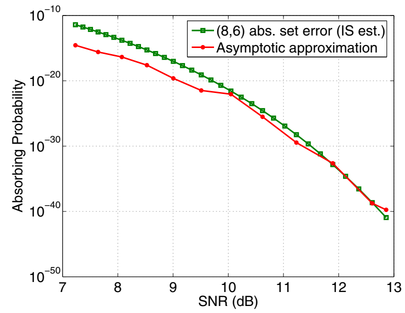
Moreover, this same approach can also be applied to compute analytical approximations for other channels, which could be used to model time-varying behavior or model uncertainty. Consider a mixture channel, in which the noise on per-symbol basis is either $N(0, \sigma^2)$ with probability $1/2$ ("good" channel state), or $N(0, 4\sigma^2)$ with probability $1/2$ ("bad" channel state). In this case the Chernoff approximation (1) no longer has an explicit solution, but can be computed easily.

C. Computing lower bounds on error floors

We now put together the pieces to compute approximations to error floors. We also prove that these approximations are guaranteed to be lower bounds in the high SNR regime. Overall, given that the all-zeros codeword was sent, the error



(a)



(b)

Fig. 3. IS result and the deterministic estimate of abs. probability for the $(8, 6)$ abs. sets of the $(2209, 1978)$ array-based LDPC code. (a) Standard AWGN channel. (b) Mixture channel: noise given by $N(0, \sigma^2)$ with probability $\frac{1}{2}$, and $N(0, 4\sigma^2)$ with probability $\frac{1}{2}$.

probability can be decomposed into a disjoint union of error events, including the error probability associated with some class $\mathcal{A}$ of isomorphic absorbing sets, so that $p_{\text{err}} \geq \mathbb{P}[\mathcal{A}]$. The error event associated with the class $\mathcal{A}$ can be decomposed into a disjoint union of error events associated with individual absorbing sets.

Since each absorbing set belonging to the same class has the same unlabeled structure, they all look the same from the point of view of the transmitted codeword, and thus each of these individual error probabilities is identical, so that

$$p_{\text{err}} \quad \geq \quad \text{card}(\mathcal{A})\, \mathbb{P}[Z \in \mathcal{R}], \tag{2}$$

where $\text{card}(\mathcal{A})$ denotes the number of absorbing sets in the given class. In past work [3], we have described analytical methods for computing $\text{card}(\mathcal{A})$ for structured LDPC codes, so let us focus on the absorbing probability $\mathbb{P}[Z \in \mathcal{R}]$. From equation (1), the absorbing probability for the AWGN channel decays at a rate specified by $d^2(0; \mathcal{R})$, the minimum Euclidean distance from the all-zeros vector to the absorbing region $\mathcal{R}$. In practice, we cannot compute this minimum distance exactly, since the true absorbing region $\mathcal{R}$ is a complicated subset of $\mathbb{R}^n$. However, consider the projections $\widetilde{\mathcal{R}} \subset \mathbb{R}^d$ computed by the procedure previously described. Note that by construction, any vector inside $\widetilde{\mathcal{R}} \subset \mathbb{R}^d$, if padded with $(n - d)$ zeros to extend it to an n-vector, is guaranteed to lie within the true absorbing region. Therefore, the probability $\mathbb{P}[Z \in$ inverse image of $\widetilde{\mathcal{R}}$ under projection$]$ is guaranteed to be a lower bound to the true absorbing probability $\mathbb{P}[Z \in \mathcal{R}]$. We illustrate the bound obtained by this procedure for $(8, 6)$ absorbing sets of the $(2209, 1978)$ array-based LDPC code in Fig. 3. The bound shows a very close agreement with the IS curve and, as predicted theoretically, becomes a lower bound as the SNR increases.

V. Conclusion

We presented two methods for evaluating the probability of error of iteratively decoded LDPC codes. The first method, building on our previous work, uses importance sampling with appropriate mean shifts with respect to absorbing sets, and produces stochastic error probabilities, showing very close agreement with experimental results obtained from a hardware emulator. The second method produces deterministic estimates, based on computing projections of absorbing regions and using asymptotics to estimate associated error probabilities. These deterministic estimates are guaranteed to lower bound the true error probability in the high SNR regime. An interesting future direction is whether similar techniques can yield matching upper bounds on error probabilities.

Acknowledgements: Work supported by NSF grant CCF-0635372 and a grant from Marvell Corp. through the UC Micro program. Computing infrastructure support provided by NSF grant CNS-0403427-PROJECT.

References

[1] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, UK, 2004.

[2] E. Cavus, C. Haymes, and B. Daneshrad. A highly efficient importance sampling method for performance evaluation of LDPC codes at very low bit errors rates. *Submitted IEEE Trans. on Comm.*

[3] L. Dolecek, Z. Zhang, V. Anantharam, and M. Wainwright B. Nikolic. Analysis of absorbing sets of array-based LDPC codes. In *IEEE Int. Conf. on Comm.*, June 2007.

[4] L. Dolecek, Z. Zhang, M. Wainwright, V. Anantharam, and B. Nikolic. Evaluation of the low frame error rate performance of LDPC codes using importance sampling. In *IEEE Info. Theory Workshop*, Sept. 2007.

[5] C. A. Cole et al. A general method for finding low error rates of LDPC codes. *Submitted to IEEE Trans. Info. Theory.*

[6] J. Fan. Array-codes as low-density parity-check codes. In *2nd Int. Symp. on Turbo Codes*, Sept. 2000.

[7] B. J. Frey, R. Koetter, and A. Vardy. Skewness and pseudocodewords in iterative decoding. In *ISIT*, 1998.

[8] R. G. Gallager. *Low-density parity check codes*. MIT Press, Cambridge, MA, 1963.

[9] S. Laendner and O. Milenkovic. Algorithmic and combinatorial analysis of trapping sets in structured LDPC codes. In *WirelessComm*, June 2005.

[10] D. MacKay and M. Postol. Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes. *Elec. Notes in Theo. Comp. Science*, 74, 2003.

[11] T. Richardson. Error floors of LDPC codes. In *Allerton*, Oct. 2003.

[12] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approaching irregular low-density parity check codes. *IEEE Trans. Info. Theory*, 47:619–637, February 2001.

[13] R. J. Serfling. *Approximation Theorems of Mathematical Statistics*. Wiley Series in Probability and Statistics. Wiley, 1980.

[14] R. M. Tanner. A recursive approach to low complexity codes. *IEEE Trans. Info. Theory*, 81(5):533–547, September 1981.

[15] B. Xia and W. E. Ryan. Estimating LDPC codeword error rates via importance sampling. In *ISIT*, July 2004.

[16] B. Xia and W. E. Ryan. Importance sampling for tanner trees. *IEEE Trans. on Info. Theory*, 51:2183 – 2189, June 2005.

[17] Z. Zhang, L. Dolecek, V. Anantharam, and M. Wainwright B. Nikolic. Quantization effects of low-density parity-check decoders. In *IEEE Int. Conf. on Comm.*, June 2007.