

Performance of LDPC Codes Under Noisy Message-Passing Decoding

Lav R. Varshney

Laboratory for Information and Decision Systems, Massachusetts Institute of Technology

Abstract—In traditional communication theory, it is assumed that error correcting codes may be decoded with algorithms that perform perfectly. Noise, however, provides a fundamental limit to computation systems just as it does to communication systems. In this work, we investigate the effect of noise in message-passing decoders for low-density parity-check codes. We observe that the concentration of the performance of the decoder around its average performance continues to hold when noise is introduced into message-passing. Given the concentration result, density evolution equations for a simple noisy message-passing decoder are derived. Analytic computation of thresholds shows that performance degrades smoothly as decoder noise increases. Decoding is robust to noise in the decoder.

I. INTRODUCTION

In the quest for channel capacity, decoder complexity has always been a limiting factor [1]. Low-density parity-check (LDPC) codes have emerged as a class of codes that have performance near the Shannon limit [2], [3] and yet are sufficiently structured as to have instrumentable decoders [4], [5]. The main class of decoders for LDPC codes use iterative message-passing algorithms. When the code is represented as a factor graph, algorithm computations occur at nodes and algorithm communication is carried out over edges. The correspondence between the factor graph and the algorithm is not just a tool for exposition but also the way decoding circuits are implemented [4], [5]. In traditional analysis of decoding performance, it has been assumed that these iterative algorithms work without error. Given the possibility of faulty computation on increasingly unreliable hardware [6], we feel it is disingenuous to assume that there is no noise in the decoder. Therefore, we regard the decoder as a noisy computation system and try to determine whether it still works well.

In this paper, we focus on computation noise that is manifested as message errors, though the asymptotic tools we develop can easily be extended to cases where node computations are also faulty. In belief propagation decoding, messages are continuous-valued and so some quantization or noise must be incurred when representing them [4]. Even for decoding algorithms with finite alphabet messages, the shrinking physical dimensions and desired low power consumption of modern digital computation systems will reduce the signal to noise ratio and perfect message-passing will not hold [5], [7]. There will be transient message errors. Loeliger et al. have observed that decoders are robust to nonidealities and

noise in physical implementations, however they noted that “the quantitative analysis of these effects is a challenging theoretical problem.” In our work we take steps to address this challenge by mathematically characterizing robustness to decoder noise. The results may provide guidelines for power control in decoder circuits.

In other empirical characterizations of decoders, it has been found that the performance of decoding algorithms when messages are quantized at high rate show little effect on probability of error performance [2]. Even algorithms that are low-rate quantized versions of belief propagation show little degradation in performance [3]. In [8], an alternative equivalent algorithm to belief propagation is proposed and shown to have less sensitivity to quantization. Other work has looked at optimizing quantization and analyzing performance change due to quantization [9], [10]. Note, however, that there is a significant difference between suboptimal quantized decoders that are perfect and decoders that make random errors in the message-passing. To the best of our knowledge, the only previous work on message-passing algorithms with random message errors is [11], which deals with problems outside of the decoding context.

We develop tools to study noisy message-passing decoding of LDPC codes by extending the asymptotic characterizations originally developed by Richardson and Urbanke for noiseless message-passing decoders [3]. These tools allow us to reduce the performance analysis problem to the study of a dynamical system described by a density evolution equation. For a particular LDPC code ensemble and a particular message-passing decoding algorithm, we demonstrate that the system is robust to decoder noise. For a given decoder noise level, we compute the channel noise level for which the probability of error can be driven to a small value and show that this value degrades smoothly as a function of the noisiness of the decoder. We close with a discussion of several avenues for further work, including the optimization of codes for noisy decoders and the computation of the reliable information storage capacity of memories designed from unreliable components (originally studied in [12]).

II. TOOLS FOR PERFORMANCE ANALYSIS

Considering the great successes achieved by analyzing the noiseless decoder performance of ensembles of codes [3], [13] rather than of particular codes [2], we would like to do the same for noisy decoders. A main contribution of this work is to extend the methods of analysis promulgated in [3] to the case

This work was supported in part by a National Science Foundation Graduate Research Fellowship and was performed when the author was with l'École Polytechnique Fédérale de Lausanne.

of decoders with random noise that perturb messages. First we show that under certain symmetry conditions, the probability of error does not depend on which codeword is transmitted. Second we show a concentration result that the individual performances of codes in the ensemble is the same as the average performance of the ensemble with high probability. Finally we show that this average behavior converges to the behavior of a code defined on a cycle-free graph. For brevity, we restrict ourselves to regular LDPC codes, however the results can easily be generalized to irregular codes.

A. General Description and Notation

We consider the standard ensemble of $(1, r)$ -regular LDPC codes of length n , $\mathcal{C}^n(1, r)$, defined by a uniform measure on the set of labeled bipartite graphs with variable node degree 1 and check node degree r . We may also consider irregular codes, $\mathcal{C}^n(\lambda, \rho)$ characterized by the degree distribution pair $(\lambda(x), \rho(x))$, where $\lambda(x)$ and $\rho(x)$ are functions of the form $\lambda(x) = \sum_{i=2}^{\infty} \lambda_i x^{i-1}$ and $\rho(x) = \sum_{i=2}^{\infty} \rho_i x^{i-1}$.

Channel input and output letters are denoted X and $Y \in \mathcal{Y}$. At time $\ell = 0$, each variable node has a realization of Y , r_i . A message-passing decoder exchanges messages between nodes along edges. First each variable node sends a message to a neighboring check node through a message channel. Generically, we refer to sent messages as ν , message channel noise as w , and received messages as μ : assume that $\nu, \mu \in \mathcal{M}$. Each check node processes received messages and sends back a message to each neighboring variable node through a message channel. The ν, μ , and w notation is reused. Each variable node now processes the messages it receives and its r_i to produce new messages. This continues iteratively. Only extrinsic information is used in node computations. Denote the variable node computation as $\Psi^{(\ell)} : \mathcal{Y} \times \mathcal{M}^{r-1} \mapsto \mathcal{M}$ and the check node computation as $\Phi^{(\ell)} : \mathcal{M}^{r-1} \mapsto \mathcal{M}$. We consider these functions to be deterministic, but more generally they may be noisy. Message passing induces directed decoding neighborhoods, which involve nodes that have communicated with one another. We use probability of digit error, P_e , as the performance criterion throughout.

B. Restriction to All-One Codeword

If certain symmetry conditions are satisfied by the system, then the probability of error is conditionally independent of the codeword that is sent. We define several symmetry conditions:

Definition 1 (Channel Symmetry): The channel satisfies

$$p(Y_t = r | X_t = 1) = p(Y_t = -r | X_t = -1).$$

Definition 2 (Check Node Symmetry): The check node message map satisfies

$$\Phi^{(\ell)}(b_1 \mu_1, \dots, b_{r-1} \mu_{r-1}) = \Phi^{(\ell)}(\mu_1, \dots, \mu_{r-1}) \left(\prod_{i=1}^{r-1} b_i \right)$$

for any ± 1 sequence (b_1, \dots, b_{r-1}) .

Definition 3 (Variable Node Symmetry): The variable node message map satisfies

$$\Psi^{(0)}(-\mu_0) = -\Psi^{(0)}(\mu_0),$$

and

$$\Psi^{(\ell)}(-\mu_0, -\mu_1, \dots, -\mu_{\ell-1}) = -\Psi^{(\ell)}(\mu_0, \mu_1, \dots, \mu_{\ell-1}),$$

for $\ell \geq 1$.

Definition 4 (Message Channel Symmetry): For messages where the sign indicates bit estimate, the message channel is such that

$$p(\mu = m' | \nu = m) = p(\mu = -m' | \nu = -m),$$

where μ is any message received at a node when the message sent from the opposite node is ν .

Theorem 1 (Conditional Independence of Error): For a given binary linear code and a given noisy message-passing algorithm, let $P_e^{(\ell)}(\mathbf{x})$ denote the conditional probability of error after the ℓ th decoding iteration, assuming that codeword \mathbf{x} was sent. If the channel and the decoder satisfy symmetry conditions given in Definitions 1–4, then $P_e^{(\ell)}(\mathbf{x})$ does not depend on \mathbf{x} .

Proof: The proof is almost identical to the proof of [3, Lemma 1]. The channel, check node, and variable node symmetry requirements are used in the same manner. The requirement of message channel symmetry implies that the message passing errors affect the likelihoods in the same way for messages derived from any \mathbf{x} . ■

In the sequel we assume that the system satisfies the symmetry conditions. Since probability of error is independent of the codeword sent and since all LDPC codes have the all-one codeword as part of the codebook, we can assume without loss of generality that this codeword was sent. This removes the randomness associated with source message selection.

C. Concentration around Ensemble Average

Now we show that the performances of almost all LDPC codes closely match the average performance of the ensemble from which they are drawn. The average is over the instance of the code, the realization of the channel noise, and the realization of the decoder noise. To simplify notation, assume that the number of decoder iterations is fixed at some finite ℓ . Let Z be the number of incorrect values held among all $1n$ variable nodes at the end of the ℓ th iteration (for a particular code, channel noise realization, and decoder noise realization) and let $E[Z]$ be the expected value of Z .

Theorem 2 (Concentration Around Expected Value):

There exists a positive constant $\beta = \beta(1, r, \ell)$ such that for any $\epsilon > 0$,

$$\Pr[|Z - E[Z]| > n1\epsilon/2] \leq 2e^{-\beta\epsilon^2 n}.$$

Proof: Recall that Z denotes the number of incorrect values held at the end of the ℓ th iteration for a particular $(G, R, W) \in \Omega$, where G is a graph in the ensemble $\mathcal{C}^n(1, r)$, R is a particular input to the decoder, W is a particular realization of the decoder noise, and Ω is the probability space. Let $=_i, 0 \leq i \leq (1 + 2\ell + 1)n = m$, be a sequence of equivalence relations on Ω ordered by refinement, such that $(G', R', W') =_i (G'', R'', W'')$ implies $(G', R', W') =_{i-1} (G'', R'', W'')$.

Suppose we sequentially expose variables which are random. First we expose the $1n$ edges of the graph one at a time, at step $i \leq 1n$ exposing the particular check node socket $\pi(i)$ which is connected to the i th variable node socket. Next, in the following n steps, we expose the received values r_i one at a time. Finally in the remaining $2\ell 1n$ steps, we expose the decoder noise values w_i that were encountered for each of the check-to-variable and variable-to-check messages in all iterations up to iteration ℓ . Then we have $(G', R', W') =_i (G'', R'', W'')$ if and only if the information revealed in the first i steps for both pairs is the same.

Now, define Z_0, Z_1, \dots, Z_m by

$$Z_i(G, R, W) = E[Z(G', R', W') | (G', R', W') =_i (G, R, W)],$$

where $Z_0 = E[Z]$ and $Z_m = Z$. By construction Z_0, Z_1, \dots, Z_m is a Doob's martingale.

We would like to use Azuma's inequality [3] to give bounds on

$$\Pr[|Z - E[Z]| > n1\epsilon/2] = \Pr[|Z_m - Z_0| > n1\epsilon/2].$$

To do so, we need to prove a bounded difference condition

$$|Z_{i+1}(G, R, W) - Z_i(G, R, W)| \leq \alpha_i, \quad i = 0, \dots, m-1$$

for suitable constants α_i which may depend on $1, r$, and ℓ .

For the steps where edges are exposed, it was shown by Richardson and Urbanke [3] that

$$|Z_{i+1}(G, R, W) - Z_i(G, R, W)| \leq 8(1r)^\ell, \quad 0 \leq i < n1.$$

It was further shown that for the steps when the channel outputs are revealed that

$$|Z_{i+1}(G, R, W) - Z_i(G, R, W)| \leq 2(1r)^\ell, \quad n1 \leq i < n(1+1).$$

It remains to show that the inequality is also fulfilled for the last $2\ell 1n$ steps when the decoder noise realization is revealed. When an edge noise realization w is revealed, only things whose directed neighborhood includes the edge on which the noise w causes perturbations can be affected. In [3], it is shown that the size of the directed neighborhood of depth 2ℓ of the edge $\vec{e}(w)$ associated with noise w is bounded as $|\mathcal{N}_{\vec{e}(w)}^{2\ell}| \leq 2(1r)^\ell$. Since the maximum depth that can be affected by a noise perturbation is 2ℓ , a weak uniform bound for the remaining exposure steps is

$$|Z_{i+1}(G, R, W) - Z_i(G, R, W)| \leq 2(1r)^\ell, \quad n(1+1)1 \leq i < m.$$

Since we have provided bounded difference constants α_i for all i , the theorem follows from application of Azuma's inequality to the martingale [3]. We can take $\frac{1}{\beta} = (544 + 64\ell)1^{2\ell-1}r^{2\ell}$. ■

Remark 1: Note that the theorem that we presented can be extended to irregular LDPC codes. It can also be extended to noisy message-passing decoders that incur random node computation errors in addition to random message errors. To do so would require more stages of revelation, each of which may have $\alpha_i = 2(1r)^\ell$.

D. Convergence to the Cycle-Free Case

We have seen that the noisy decoding algorithm behaves essentially deterministically. As a further simplification, we show that the ensemble average performance converges to the performance of an associated tree ensemble. This result will allow us to invoke the assumption of independent messages.

For a given edge \vec{e} whose directed neighborhood of depth 2ℓ is tree-like, let p be the expected number of incorrect messages received along this edge (after message noise) at the ℓ th iteration, averaged over all graphs, inputs and decoder noise realizations.

Theorem 3 (Convergence to Cycle-Free Case): There exists a positive constant $\gamma = \gamma(1, r, \ell)$ such that for any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$,

$$|E[Z] - n1p| < n1\epsilon/2.$$

Proof: Proof is identical to the proof of [3, Theorem 2]. ■

The concentration and convergence results directly imply concentration around the average performance of a tree ensemble:

Theorem 4 (Concentration Around Cycle-Free Case):

There exist positive constants $\beta = \beta(1, r, \ell)$ and $\gamma = \gamma(1, r, \ell)$ such that for any $\epsilon > 0$ and $n > \frac{2\gamma}{\epsilon}$,

$$\Pr[|Z - n1p| > n1\epsilon] \leq 2e^{-\beta\epsilon^2 n}.$$

Proof: Follows directly from Theorems 2 and 3. ■

III. DENSITY EVOLUTION FOR NOISY GALLAGER A

In Section II, we showed that concentration results apply to noisy decoders just as they do to noiseless decoders. Thus density evolution equations will tell us about the performance of almost all codes in the large blocklength regime. To show the utility of our result, we now derive the density evolution equation for a simple noisy message-passing decoder, which is a noisy version of Gallager's decoding algorithm A [2], [14]. The algorithm only allows elements of the set $\{\pm 1\}$ as messages, which indicate the estimated sign of a binary digit. Although this simple decoding algorithm cannot match the performance of belief propagation, it is of interest since it is of extremely low complexity and can be analyzed analytically.

We look at decoding of the LDPC coded output of a binary symmetric channel (BSC) with crossover probability ϵ . Each message in the Gallager algorithm A is passed through an independent and identical BSC with crossover probability α . At a check node, the outgoing message along edge \vec{e} is the product of all incoming messages excluding the one incoming on \vec{e} . At a variable node, the outgoing message is the original received message unless all incoming messages give the opposite conclusion. For generality (and eventually concise notation), we develop the density evolution equation for general irregular LDPC ensembles.

The original received message is in error with probability ϵ , thus the

$$P_e^{(0)}(\epsilon, \alpha) = x_0 = \epsilon,$$

where we denote the probability of error at the variable nodes in iteration 0 as $P_e^{(0)}(\epsilon, \alpha)$ and which we define to be x_0 .

The initial variable-to-check message is in error with probability $(1 - \epsilon)\alpha + \epsilon(1 - \alpha)$, since it is passed through a BSC(α). For further iterations, ℓ , we find the probability of error, $P_e^{(\ell)}(\epsilon, \alpha)$, by induction. Assume $P_e^{(i)}(\epsilon, \alpha) = x_i$ for $0 \leq i \leq \ell$. Now consider the error probability of a check-to-variable message in the $(\ell + 1)$ th iteration. A check-to-variable message emitted by a check node of degree r along a particular edge is the product of all the $(r - 1)$ incoming messages along all other edges. By assumption, each such message is in error with probability x_ℓ and all messages are independent. These messages are passed through BSC(α) before being received, so the probability of being received in error is

$$x_\ell(1 - \alpha) + (1 - x_\ell)\alpha = \alpha + x_\ell - 2\alpha x_\ell.$$

The outgoing message will be in error if an odd number of these received messages are in error. The probability of this event, averaged over the degree distribution yields the probability

$$\frac{1 - \rho[1 - 2(\alpha + x_\ell - 2\alpha x_\ell)]}{2}.$$

Now consider $P_e^{(\ell+1)}(\epsilon, \alpha)$, the error probability at the variable node in the $(\ell + 1)$ th iteration. Consider an edge which is connected to a variable node of degree 1. The outgoing variable-to-check message along this edge is in error in the $(\ell + 1)$ th iteration if the original received value is in error and not all incoming messages are received correctly or if the originally received value is correct but all incoming messages are in error. The first event has probability

$$\epsilon \left(1 - \left[1 - (1 - \alpha) \left(\frac{1 - \rho[1 - 2(\alpha + x_\ell - 2\alpha x_\ell)]}{2} \right) - \alpha \left(\frac{1 + \rho[1 - 2(\alpha + x_\ell - 2\alpha x_\ell)]}{2} \right) \right]^{1-1} \right).$$

The second event has probability

$$(1 - \epsilon) \left(\left[(1 - \alpha) \left(\frac{1 - \rho[1 - 2(\alpha + x_\ell - 2\alpha x_\ell)]}{2} \right) + \alpha \left(\frac{1 + \rho[1 - 2(\alpha + x_\ell - 2\alpha x_\ell)]}{2} \right) \right]^{1-1} \right).$$

Averaging over the degree distribution, defining $\omega = (2\alpha - 1)(2x_\ell - 1)$,

$$q_\alpha^+(x) = \lambda \left[\frac{1 + \rho(\omega) - 2\alpha\rho(\omega)}{2} \right],$$

and

$$q_\alpha^-(x) = \lambda \left[\frac{1 - \rho(\omega) + 2\alpha\rho(\omega)}{2} \right],$$

and adding the two terms together yields

$$x_{\ell+1} = \epsilon - \epsilon q_\alpha^+(x_\ell) + (1 - \epsilon)q_\alpha^-(x_\ell). \quad (1)$$

This is our density evolution equation in recursive form.

IV. PERFORMANCE OF DENSITY EVOLUTION

Now that we have the density evolution equation, we can determine the performance of the coding-decoding system with particular values of quality parameters ϵ and α . This is simply an analysis of a deterministic, discrete-time, dynamical system. We take error probability as the state variable and are looking for stable fixed points of the system, the states to which the decoder converges. We would want the probability of error to converge to zero as the number of iterations increases, but we will see that this might not be possible, so we will need a weaker performance criterion than usual [3], [14]. To start, consider partially noiseless cases.

A. Partially Noiseless Systems

For the noiseless decoder case, i.e. $\alpha = 0$, it has been known that there are thresholds on ϵ , below which the probability of error goes to zero as ℓ increases, and above which the probability of error goes to some large value. These can be found analytically for the Gallager A algorithm [14].

For the noisy Gallager A system we are considering, we find that for any $\alpha > 0$, the probability of error does not go to zero as ℓ goes to infinity. This can be seen by considering the case of the perfect original channel, $\epsilon = 0$, and any $\alpha > 0$. The density evolution equation reduces to

$$x_{\ell+1} = q_\alpha^-(x_\ell), \quad (2)$$

with $x_0 = 0$. The recursion does not have a fixed point at zero, and since x is bounded below by zero, it must increase. The derivative is

$$\frac{\partial}{\partial x} q_\alpha^-(x) = \lambda' \left[\frac{1 - \rho(\omega) + 2\alpha\rho(\omega)}{2} \right] \rho'[\omega] (2\alpha - 1)^2,$$

which is greater than zero for $0 \leq x \leq \frac{1}{2}$ and $0 \leq \alpha \leq \frac{1}{2}$; thus the error evolution forms a monotonically increasing sequence. Since the sequence is monotone increasing starting from zero, and there is no fixed point at zero, it follows that this converges to the smallest real solution of $x = q_\alpha^-(x)$ since the fixed point cannot be jumped. The same phenomenon must also happen if the starting $x_0 > 0$, however the value to which the density evolution converges is a non-zero fixed point solution of the original equation (1), not of (2), and is a function of both α and ϵ . Intuitively, for somewhat large initial values of ϵ , the noisy decoder decreases the probability of error in the first few iterations, just like the noiseless one, but when the error probability becomes close to the internal decoder error, the probability of error settles at that level.

B. Noisy Systems

The fact that probability of error cannot asymptotically be driven to zero with the noisy Gallager decoder is expected yet is seemingly displeasing. In a practical scenario, however, the ability to drive P_e to a very small number is also desirable. As such, we define a performance objective of achieving P_e less than η and determine the worst channel (ordered by ϵ) for

which a decoder with noise level α can achieve that objective. The channel parameter

$$\epsilon^*(\eta, \alpha) = \sup\{\epsilon \in [0, \frac{1}{2}] \mid \lim_{\ell \rightarrow \infty} P_e^{(\ell)} < \eta\}$$

is called the threshold. For a large interval of η values, there is a single threshold value below which η -reliable communication is possible and above which it is not. Alternatively, we can determine the probability of error to which a system with particular α and ϵ can be driven, $\eta^*(\alpha, \epsilon) = \lim_{\ell \rightarrow \infty} P_e^{(\ell)}$, and see whether this value is small.

In order to find the threshold in the case of $\alpha > 0$ and $\epsilon > 0$, we need to find the real fixed point solutions of density evolution recursion (1). We find the solutions of the polynomial equation

$$\epsilon - \epsilon q_{\alpha}^{+}(x) + (1 - \epsilon)q_{\alpha}^{-}(x) - x = 0$$

and denote the real solutions as $0 < r_1(\alpha, \epsilon) \leq r_2(\alpha, \epsilon) \leq r_3(\alpha, \epsilon) \leq \dots$. We can also find the solutions of the polynomial equation

$$\frac{x - q_{\alpha}^{-}(x)}{1 - q_{\alpha}^{+}(x) - q_{\alpha}^{-}(x)} - x = 0,$$

whose real solutions we denote as $0 < \tau_1(\alpha) \leq \tau_2(\alpha) \leq \dots$. The number of real solutions can be determined through Descartes' rule of signs or a similar tool.

The threshold ϵ^* as well as the region in the $\alpha - \epsilon$ plane where the decoder improves performance over no decoding is determined by the τ_i . The final probability of error η^* is determined by the r_i . For particular ensembles of LDPC codes, these values can be computed analytically. For these particular ensembles, it can be determined whether the fixed points are stable or unstable. Moreover, various monotonicity results can be established to show that fixed points cannot be jumped.

We evaluate the analytical expressions for the r_i and τ_i for the (3,6) regular LDPC code. Figure 1 shows η^* as a function of ϵ for fixed α . Since r_1 and r_3 are stable fixed points of density evolution, η takes these values for particular ranges of ϵ ; r_1 is the desired small value of η^* whereas r_3 is the undesired large value of η^* . The fixed point r_2 is unstable and forms the boundary between the regions of attraction for the two stable fixed points. The point where $r_1 = \epsilon$ is τ_1 and the point where $r_2 = \epsilon$ is τ_2 . These $\tau(\alpha)$ points determine when it is beneficial to use the decoder, in the sense that $\eta^* < \epsilon$. By varying α (as if in a sequence of plots like Figure 1), an $\alpha - \epsilon$ region where the decoder is beneficial is demarcated; this is shown in Figure 2. It can be noted that $\eta^*(\alpha = 0, \epsilon) = 0$ for $\epsilon \in [0, \epsilon^*]$, where ϵ^* is given in [14], which determine the ordinate intercepts in Figure 2. To provide a better sense of the performance of the noisy Gallager A algorithm, we also provide a table that lists some values of α , ϵ , and η^* (although there are analytical expressions, we provide numerical evaluations). As can be seen from these results, particularly from the τ_2 curve in Figure 2, the error probability performance of the system degrades gracefully as noise is added to the decoder.

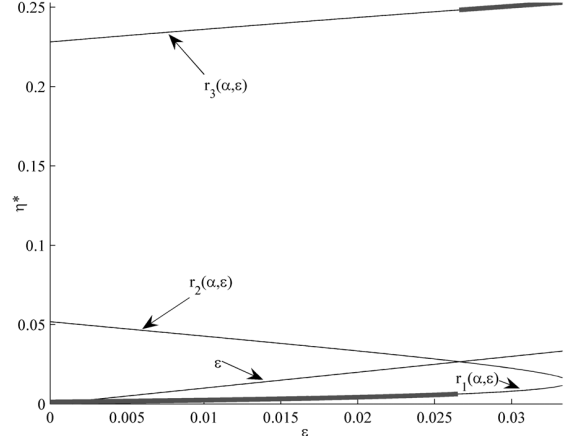


Fig. 1. Thick line shows final error probability, η^* , after decoding a $C^\infty(3,6)$ code with the noisy Gallager A algorithm, $\alpha = 0.005$. This is determined by the fixed points of density evolution, $r_i(\alpha, \epsilon)$, shown with thin lines.

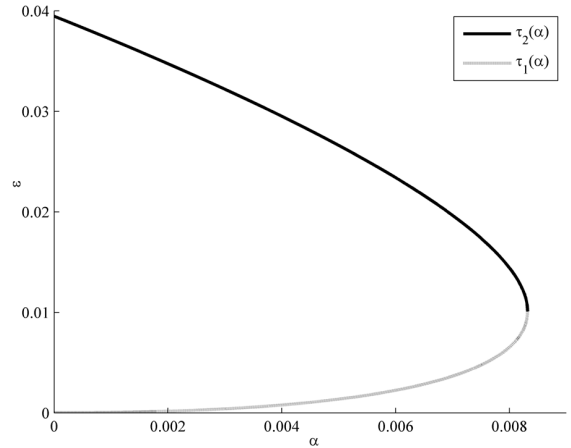


Fig. 2. Decoding a $C^\infty(3,6)$ code with the noisy Gallager A algorithm. Region where it is beneficial to use decoder is between τ_1 and τ_2 .

V. EXTENSIONS AND CONCLUSIONS

In this paper, we have shown two main results. First, due to the concentration theorems that we proved, density evolution is a valid approach to study decoders with stochastic noise. Second, by analyzing the performance of a simple noisy message-passing decoder, error performance is robust to decoder noise.

Since our concentration results apply to general noisy message-passing algorithms, the most obvious path to extending our results is to perform density evolution analysis on a multitude of other LDPC codes and noisy decoders. For example, a noisy version of the Gaussian approximation to belief propagation admits a simple one-dimensional density evolution recursion where the probability of error is the state variable [15], just as we had. In this case as well as in others, we believe that the robustness results found for the noisy Gallager A algorithm generalize.

In this work we have simply measured the error performance

TABLE I
PERFORMANCE OF NOISY GALLAGER A ALGORITHM FOR (3,6) CODE

| α | $\epsilon^*(0.1, \alpha)$ | $\eta^*(\alpha, \epsilon^*)$ | $\eta^*(\alpha, 0.01)$ |
|---------------------|---------------------------|------------------------------|--------------------------|
| 0 | 0.0394636562 | 0 | 0 |
| 1×10^{-10} | 0.0394636560 | 7.8228×10^{-11} | 1.3333×10^{-11} |
| 1×10^{-8} | 0.0394636335 | 7.8228×10^{-9} | 1.3333×10^{-9} |
| 1×10^{-6} | 0.0394613836 | 7.8234×10^{-7} | 1.3338×10^{-7} |
| 1×10^{-4} | 0.0392359948 | 7.8866×10^{-5} | 1.3812×10^{-5} |
| 3×10^{-4} | 0.0387781564 | 2.4050×10^{-4} | 4.4357×10^{-5} |
| 1×10^{-3} | 0.0371477336 | 8.4989×10^{-4} | 1.8392×10^{-4} |
| 3×10^{-3} | 0.0321984070 | 3.0536×10^{-3} | 9.2572×10^{-4} |
| 5×10^{-3} | 0.0266099758 | 6.3032×10^{-3} | 2.4230×10^{-3} |

of a system and have not tried to optimize the code for the decoder parameters. Recent work in distributed detection over noisy channels has shown performance improvements when noise characteristics are included in system design [16]. We conjecture that a similar improvement can be made when LDPC code design takes decoder noise into account. Optimization of LDPC ensembles has been studied in [14] and elsewhere. Since the space of system parameters that must be considered for noisy decoders is much larger than for noiseless decoders, optimization will presumably yield a much larger variety of optimized code ensembles.

As we saw in Table I, $P_e^{(\infty)} = \eta$ depends on $P_e^{(0)} = \epsilon$. If ϵ is smaller, then η can be made smaller. This is an artifact of the Gallager A algorithm and would not occur in other message-passing algorithms. Nevertheless, this phenomenon suggests that it might be worthwhile to have concatenated decoders, where a first decoder cleans up many of the errors and then a second decoder cleans up more; this basic idea was already present in [13].

One can use coding within the decoder to reduce decoder error α . However message-passing will take longer and a message decoder (perhaps a voter for a repetition code) will be needed at each node. Again, the concatenated decoder idea may be useful for dynamically changing the within-decoder code. A strong decoder code may be used to start things out and achieve a slightly higher threshold, a weak decoder code can be used to drive the probability of error to something small and then a strong decoder code can be used to reduce the probability of error even further.

LDPC error correcting was used by Taylor for reliable information storage in memories with noisy computational units and noisy message-passing [12]. He showed the existence of a memory system with non-zero capacity that can achieve arbitrarily small probability of error with linear complexity. The scheme had a separate noisy Gallager decoder copy for each binary digit and only needed to maintain the stored codeword within the distance properties of the code, i.e. it was assumed there would eventually be a noiseless decoder. Combining our density evolution characterization with the distance distribution of LDPC codes [17] would allow computation of the information storage capacity of Taylor's system, not just an existence proof of non-zero capacity.

We had started with the very practical problem of trying to

understand whether LDPC decoding under message-passing decoding still works when the decoder is not perfect. As noted by Pierce in 1965 [18], "The down-to-earth problem of making a computer work, in fact, becomes tangled with this difficult philosophical problem: 'What is possible and what is impossible when unreliable circuits are used to process unreliable information?'" This difficult problem seems not to have been unraveled in the interceding four decades, but it is our hope that this work provides some insight.

ACKNOWLEDGMENT

I thank Rüdiger L. Urbanke for suggesting this study of decoding; İ. Emre Telatar and him for several enlightening discussions and for hosting my visit at EPFL; and Sanjoy K. Mitter for making me aware of [12].

REFERENCES

- [1] D. J. Costello, Jr. and G. D. Forney, Jr., "Channel coding: The road to channel capacity," *Proc. IEEE*, 2007, to appear.
- [2] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge: M.I.T. Press, 1963.
- [3] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.
- [4] H.-A. Loeliger, F. Lustenberger, M. Helfenstein, and F. Tarköy, "Probability propagation and decoding in analog VLSI," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 837–843, Feb. 2001.
- [5] A. J. Blanksby and C. J. Howland, "A 690-mW 1-Gb/s 1024-b, rate-1/2 low-density parity-check code decoder," *IEEE J. Solid-State Circuits*, vol. 37, no. 3, pp. 404–412, Mar. 2002.
- [6] D. K. Pradhan, *Fault-Tolerant Computer System Design*. Upper Saddle River, N.J.: Prentice Hall, 1996.
- [7] R. Ho, K. W. Mai, and M. A. Horowitz, "The future of wires," *Proc. IEEE*, vol. 89, no. 4, pp. 490–504, Apr. 2001.
- [8] L. Ping and W. K. Leung, "Decoding low density parity check codes with finite quantization bits," *IEEE Commun. Lett.*, vol. 4, no. 2, pp. 62–64, Feb. 2000.
- [9] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier, and X.-Y. Hu, "Reduced-complexity decoding of LDPC codes," *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.
- [10] J. Zhao, F. Zarkeshvari, and A. H. Banihashemi, "On implementation of min-sum algorithm and its modifications for decoding low-density parity-check (LDPC) codes," *IEEE Trans. Commun.*, vol. 53, no. 4, pp. 549–554, Apr. 2005.
- [11] A. T. Ihler, J. W. Fisher, III, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," *J. Mach. Learn. Res.*, vol. 6, pp. 905–936, May 2005.
- [12] M. G. Taylor, "Reliable information storage in memories designed from unreliable components," *Bell Syst. Tech. J.*, vol. 47, no. 10, pp. 2299–2337, Dec. 1968.
- [13] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.
- [14] L. Bazzi, T. J. Richardson, and R. L. Urbanke, "Exact thresholds and optimal codes for the binary-symmetric channel and Gallager's decoding algorithm A," *IEEE Trans. Inform. Theory*, vol. 50, no. 9, pp. 2010–2021, Sept. 2004.
- [15] F. Lehmann and G. M. Maggio, "Analysis of the iterative decoding of LDPC and product codes using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 49, no. 11, pp. 2993–3000, Nov. 2003.
- [16] B. Chen, L. Tong, and P. K. Varshney, "Channel-aware distributed detection in wireless sensor networks," *IEEE Signal Processing Mag.*, vol. 23, no. 4, pp. 16–26, July 2006.
- [17] C. Di, T. J. Richardson, and R. L. Urbanke, "Weight distribution of low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4839–4855, Nov. 2006.
- [18] W. H. Pierce, *Failure-Tolerant Computer Design*. New York: Academic Press, 1965.