Exchange of Limits: Why Iterative Decoding Works

Satish Babu Korada and Rüdiger Urbanke

***Abstract*— We consider communication over binary-input memoryless output-symmetric channels using low-density parity-check codes and message-passing decoding. The asymptotic (in the length) performance of such a combination for a fixed number of iterations is given by density evolution. Letting the number of iterations tend to infinity we get the density evolution *threshold*, the largest channel parameter so that the bit error probability tends to zero as a function of the iterations.**

In practice we often work with short codes and perform a large number of iterations. It is therefore interesting to consider what happens if in the standard analysis we exchange the order in which the blocklength and the number of iterations diverge to infinity. In particular, we can ask whether both limits give the same threshold.

Although empirical observations strongly suggest that the exchange of limits is valid for *all* channel parameters, we limit our discussion to channel parameters below the density evolution threshold. Specifically, we show that under some suitable technical conditions the bit error probability vanishes below the density evolution threshold regardless of how the limit is taken.

***Index Terms*— LDPC, sparse graph code, density evolution**

I. Introduction

A. Motivation

Consider transmission over a binary-input memoryless output-symmetric (BMS) channel using a low-density parity-check (LDPC) code and decoding via a message-passing (MP) algorithm. We refer the reader to [1] for an introduction to the standard notation and an overview of the known results. It is well known that, for good choices of the degree distribution and the MP decoder, one can achieve rates close to the capacity of the channel with low decoding complexity [2].

The standard analysis of iterative decoding systems assumes that the blocklength is large (tending to infinity) and that a fixed number of iterations is performed. As a consequence, when decoding a given bit, the output of the decoder only depends on a fixed-sized local neighborhood of this bit and this local neighborhood is tree-like. This local tree property implies that the messages arriving at nodes are conditionally independent, significantly simplifying the analysis. To determine the performance in this setting, we track the evolution of the message densities as a function of the iteration. This process is called *density evolution* (DE). Denote the bit probability of error of a code $\mathtt{G}$ after ℓ iterations by $P_b(\mathtt{G}, \epsilon, \ell)$, where ϵ is the channel parameter. Then DE computes

$$\lim_{n\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]. \tag{1}$$

If we now perform more and more iterations then we get a limiting performance corresponding to

$$\lim_{\ell\to\infty} \lim_{n\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]. \tag{2}$$

EPFL, School of Computer, & Communication Sciences, Lausanne, CH-1015, Switzerland, {satish.korada, ruediger.urbanke}@epfl.ch.

In order for the computation graphs of depth ℓ to form a tree, the number of iterations can not exceed $c\log(n)$, where c is a constant that only depends on the degree distribution. (For a $(\mathtt{l}, \mathtt{r})$-regular degree distribution pair a valid choice of c is $c(\mathtt{l}, \mathtt{r}) = \frac{2}{\log(\mathtt{l}-1)(\mathtt{r}-1)}$, [3].) In practice, this condition is rarely fulfilled: standard blocklengths measure only in the hundreds or thousands but the number of iterations that have been observed to be useful in practice can easily exceed one hundred.

Consider therefore the situation where we fix the blocklength but let the number of iterations tend to infinity. This means, we consider the limit

$$\lim_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]. \tag{3}$$

Now take the blocklength to infinity, i.e., consider

$$\lim_{n\to\infty} \lim_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]. \tag{4}$$

What can we say about (4) and its relationship to (2)?

Consider the belief propagation (BP) algorithm. It was shown by McEliece, Rodemich, and Cheng [4] that one can construct specific graphs and noise realizations so that the messages on a specific edge either show a chaotic behavior (as a function of iteration) or converge to limit cycles. In particular, this means that the messages do not converge as a function of the iteration. For a fixed length and a discrete channel, the number of graphs and noise realizations is finite. Therefore, if for single graph and noise realization the messages do not converge as a function of ℓ, then it is likely that also $\lim_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]$ does not converge as a function of n (unless by some miracle the various non-converging parts cancel). Let us therefore consider $\limsup_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]$ and $\liminf_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]$. What happens if we increase the blocklength and consider $\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]$ and $\lim_{n\to\infty} \liminf_{\ell\to\infty} \mathbb{E}[P_b(\mathtt{G}, \epsilon, \ell)]$?

We restrict our present study to the exchange of limits *below the density threshold*. I.e., suppose that the given combination (of the channel family and the MP decoder) has a threshold in the following sense: for the given channel family characterized by the real valued parameter ϵ there exists a threshold ϵ^{MP} so that for all $0 \le \epsilon < \epsilon^{\text{MP}}$ the DE limit (2) is 0, whereas for all $\epsilon > \epsilon^{\text{MP}}$ it is strictly positive. We will show that under suitable technical conditions the bit error probability also tends to zero if we exchange the limits. This implies that the DE threshold is a meaningful and robust design parameter.

B. Summary of Main Result

Consider transmission over a BMS channel parametrized by ϵ, using an LDPC$(n, \mathtt{l}, \mathtt{r})$ ensemble and decoding via an MP algorithm. Assume that the algorithm is symmetric in the sense of [1][Definition 4.81, p. 209]. Moreover, assume that this

combination has a threshold and let ϵ^{MP} denote this threshold. If $\epsilon < \epsilon^{\text{MP}}$ then under the conditions stated in Sections II and III,

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}[P_b^{\text{MP}}(\mathtt{G}, \epsilon, \ell)] = 0.$$

Instead of considering just an exchange of limits one can consider joint limits where the iteration is an arbitrary but increasing function of the blocklength, i.e., one can consider $\lim_{n\to\infty} \mathbb{E}[P_b^{\text{MP}}(\mathtt{G}, \epsilon, \ell(n))]$. Our arguments extend to this case and one can show that

$$\limsup_{n\to\infty} \mathbb{E}[P_b^{\text{MP}}(\mathtt{G}, \epsilon, \ell(n))] = 0.$$

But for the sake of simplicity we restrict ourselves to the standard exchange of limits discussed above. In the same spirit, although some of the techniques and statements we discuss extend directly to the irregular case, in order to keep the exposition simple we restrict our discussion to the standard regular ensemble LDPC$(n, \mathtt{l}, \mathtt{r})$.

C. Outline

We introduce two techniques that are useful in our context. First, we consider expanders. More precisely, in Section II we show that for codes with sufficient expansion the exchange of limits is valid below the DE threshold. The advantage of using expansion is that the argument applies to a wide variety of decoders. On the negative side, the argument can only be applied to ensembles with large variable-node degrees.

Why does expansion help in proving the desired result and why do we need large variable-node degrees? Assume that a sufficient number of iterations has been performed so that the number of still erroneous messages is relatively small. Consider further iterations. There are two reasons why a message emitted by a variable node can be bad. This can be due to the received value, or it can be due to a large number of bad incoming messages. If the degree of the variable node is large then the received value becomes less and less important (think of a node of degree 1000 and a decoder with a finite number of messages; in this case the received value has only a limited influence on the outgoing message and this message is mostly determined by the 999 incoming messages). If we ignore therefore the received message then we see that expansion helps since it can guarantee that only few nodes have many bad incoming messages; otherwise the set of nodes that has bad outgoing messages has too few neighbors in order for the graph to be an expander.

If the variable nodes have small degree, then the received values play a significant role and can no longer be ignored. Therefore, for small degrees expansion arguments do not suffice by themselves. In Section III we concentrate on the case $\mathtt{l} = 3$. This is the smallest degree that is meaningful for all the decoders that we consider and so one can think of it as the most difficult general case. Except for the BEC, this case is not covered by a simple expansion argument and the techniques are more involved.

II. Sufficient Conditions Based on Expansion Arguments

Burshtein and Miller were the first to realize that expansion arguments can be applied not only to the flipping algorithm but also to show that certain MP algorithms have a fixed error correcting radius [5]. Although their results can be applied directly to our problem, we get stronger statements by using the expansion in a slightly different manner.

A. Definitions and Review

Definition 1 (Expansion): Let $\mathtt{G}$ be an element from LDPC$(n, \mathtt{l}, \mathtt{r})$.
1) Left Expander: The graph $\mathtt{G}$ is an $(\mathtt{l}, \mathtt{r}, \alpha, \gamma)$ *left expander* if for every subset $\mathcal{V}$ of at most αn variable nodes, the set of check nodes that are connected to $\mathcal{V}$ is at least $\gamma|\mathcal{V}|\mathtt{l}$.
2) Right Expander: Let $m = n\frac{\mathtt{l}}{\mathtt{r}}$. The graph $\mathtt{G}$ is an $(\mathtt{l}, \mathtt{r}, \alpha, \gamma)$ *right expander* if for every subset $\mathcal{C}$ of at most αm check nodes, the set of variable nodes that are connected to $\mathcal{C}$ is at least $\gamma|\mathcal{C}|\mathtt{r}$. $\lozenge$

Why are we using expansion arguments in the context of standard LDPC ensembles? It is well known that such codes are good expanders with high probability [5].

Theorem 2 (Expansion of Random Graphs [5]): Let $\mathtt{G}$ be chosen uniformly at random from LDPC$(n, \mathtt{l}, \mathtt{r})$. Let $\alpha_{\max}$ be the positive solution of the equation

$$\frac{\mathtt{l}-1}{\mathtt{l}} h_2(\alpha) - \frac{\mathtt{l}}{\mathtt{r}} h_2(\alpha\gamma\mathtt{r}) - \alpha\gamma\mathtt{r} h_2(1/\gamma\mathtt{r}) = 0.$$

Let $\mathcal{X}(\mathtt{l}, \mathtt{r}, \alpha, \gamma)$ denote the set of graphs

$$\{\mathtt{G} \in \text{LDPC}(n, \mathtt{l}, \mathtt{r}) : \ \mathtt{G} \in (\mathtt{l}, \mathtt{r}, \alpha, \gamma) \text{ left expander}\}.$$

If $\gamma < 1 - \frac{1}{\mathtt{l}}$ then $\alpha_{\max}$ is strictly positive and for $\alpha < \alpha_{\max}$

$$\mathbb{P}\{\mathtt{G} \in \mathcal{X}(\mathtt{l}, \mathtt{r}, \alpha, \gamma)\} \geq 1 - O(n^{-(\mathtt{l}(1-\gamma)-1)}). \tag{5}$$

Let $m = n\frac{\mathtt{l}}{\mathtt{r}}$. We get the equivalent result for right expanders by exchanging the roles of $\mathtt{l}$ and $\mathtt{r}$ as well as n and m.

As explained before, the idea is to show that the error probability goes to zero once the number of bad messages becomes smaller than a certain threshold. To make this more concrete we need a proper definition of "good" message subsets.

Definition 3 (Good Message Subsets): For a fixed $(\mathtt{l}, \mathtt{r})$-regular ensemble and a fixed MP decoder with message alphabet $\mathcal{M}$, let β, $0 < \beta \leq 1$, be such that $\beta(\mathtt{l}-1) \in \mathbb{N}$. A "good" pair of subsets of $\mathcal{M}$ of "strength" β is a pair of subsets $(G_{\mathtt{v}}, G_{\mathtt{c}})$ so that

- if at least $\beta(l-1)$ of the $(\mathtt{l}-1)$ incoming messages at a variable node belong to $G_{\mathtt{v}}$ then the outgoing message on the remaining *edge* is in $G_{\mathtt{c}}$
- if all the $(\mathtt{r}-1)$ incoming messages at a check node belong to $G_{\mathtt{c}}$ then the outgoing message on the remaining *edge* is in $G_{\mathtt{v}}$
- if at least $\beta(\mathtt{l}-1)+1$ of all $\mathtt{l}$ incoming messages belong to $G_{\mathtt{v}}$, then the *variable* is decoded correctly

We denote the probability of the bad message set $\mathcal{M}\backslash G_{\mathtt{v}}$ after ℓ iterations of DE by $p_{\text{bad}}^{(\ell)}$. $\lozenge$

As we will see shortly, for many MP decoders of interest the sets $G_{\mathtt{v}}$ and $G_{\mathtt{c}}$ can be chosen to be equal. This is true for all those MP decoders where the outgoing reliability at a check node is equal to the least reliability of all the incoming messages (we call them min-sum-type decoders). Therefore, if all incoming messages are good (meaning they are correct and have sufficiently large reliability) then the outgoing message is correct and also has sufficiently large reliability. The BP decoder is an interesting case where $G_{\mathtt{v}} \neq G_{\mathtt{c}}$. For this decoder the reliability of the outgoing message at a check node is *strictly* smaller than the smallest reliability of all incoming messages. Therefore, we need to define the set $G_{\mathtt{c}}$ to consist of messages of strictly higher reliability than the set of messages in $G_{\mathtt{v}}$.

Definition 4 (Good Nodes): We call a variable or check node "good" if all of its outgoing messages are good. All other nodes are called "bad." ♢

Example 5 (BEC and BP): If at least 1 of the $(\mathtt{l}-1)$ messages entering a variable node is known then the outgoing message is known and if at least 1 of the $\mathtt{l}$ messages entering a variable node is known then the variable itself is known. Further, if all of the $(\mathtt{r}-1)$ incoming messages entering a check node are known then the outgoing message is known. We conclude that *good* is equivalent to *known* and that $\beta = \frac{1}{\mathtt{l}-1}$. ♢

As a second standard example we consider transmission over the BSC(ϵ) and decoding via the so-called *Gallager Algorithm B* (GalB).

Definition 6 (Gallager Algorithm B): Messages are elements of $\{\pm 1\}$. The initial messages from the variable nodes to the check nodes are the values received via the channel. The decoding process proceeds in iterations with the following processing rules:

> Check-Node Processing: At a check node the outgoing message along a particular edge is the product of the incoming messages along all the remaining edges.
> Variable-Node Processing: At a variable node the outgoing message along a particular edge is equal to the majority vote on the set of other incoming messages and the received value. Ties are resolved randomly.

♢

Example 7 (BSC and GalB): Assume that the received value (via the channel) is incorrect. In this case at least $\lceil (\mathtt{l}-1)/2 \rceil + 1$ of the $(\mathtt{l}-1)$ incoming messages should be correct to ensure that the outgoing message is correct. If at least $\lceil (\mathtt{l}-1)/2 \rceil + 2$ of the $\mathtt{l}$ incoming messages are correct then the variable is decoded correctly. (In fact, it is sufficient to have $\lfloor (\mathtt{l}-1)/2 \rfloor + 2$ correct incoming messages to be able to decode correctly.) Therefore, *good* is equivalent to *correct* and $\beta = \frac{\lceil (\mathtt{l}-1)/2 \rceil + 1}{\mathtt{l}-1}$. ♢

B. Expansion and Bit Error Probability

Theorem 8 (Expansion and Bit Error Probability): Consider an LDPC$(n, \mathtt{l}, \mathtt{r})$ ensemble, transmission over a BMS(ϵ) channel, and a symmetric MP decoder. Let β be the strength of the good message subset. If $\beta < 1$ and if for some ϵ, $p_{\text{bad}}^{(\infty)} = 0$ then

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}_{\text{LDPC}(n,\mathtt{l},\mathtt{r})}[P_b^{\text{MP}}(\mathtt{G}, \epsilon, \ell)] = 0. \tag{6}$$

Proof: Here is the idea of the proof: we first run the MP algorithm for a fixed number of iterations such that the bit error probability is sufficiently small, say p. If the length n is sufficiently large then we can use DE to gage the number of required iterations. Then, using the expansion properties of the graph, we show that the probability of error stays close to p for any number of further iterations. In particular, we show that the error probability never exceeds cp, where c is a constant, which only depends on the degree distribution and β. Since p can be chosen arbitrarily small, the claim follows.

Here is the fine print. Define

$$\gamma = \Big(1 - \frac{1}{\mathtt{l}}\Big)\frac{1+\beta}{2} \overset{\beta<1}{<} \Big(1 - \frac{1}{\mathtt{l}}\Big). \tag{7}$$

Let $0 < \alpha < \alpha_{\max}(\gamma)$, where $\alpha_{\max}(\gamma)$ is the function defined in Theorem 2. Let $p = \frac{\alpha(1-\beta)(\mathtt{l}-1)}{4}$ and let $\ell(p)$ be the number of iterations such that $p_{\text{bad}}^{(\ell)} \leq p$. Since $p_{\text{bad}}^{(\infty)} = 0$ and $p > 0$ this is possible. Let $P_{\mathtt{e}}(\mathtt{G}, \mathtt{E}, \ell)$ denote the fraction of messages belonging to the bad set after ℓ iterations. Let Ω denote the space of code and noise realizations. Let $A \subseteq \Omega$ denote the subset

$$A = \{(\mathtt{G}, \mathtt{E}) \subseteq \Omega \,|\, P_{\mathtt{e}}(\mathtt{G}, \mathtt{E}, \ell(p)) \leq 2p\}. \tag{8}$$

From (the Concentration) Theorem 39 we know that

$$\mathbb{P}\{(\mathtt{G}, \mathtt{E}) \notin A\} \leq 2e^{-Knp^2} \tag{9}$$

for some strictly positive constant $K = K(\mathtt{l}, \mathtt{r}, p)$. In words, for most (sufficiently large) graphs and noise realizations the error probability after a fixed number of iterations behaves close to the asymptotic ensemble. We now show that once the error probability is sufficiently small it never increases substantially thereafter if the graph is an expander, regardless of how many iterations we still perform.

Let $V_0 \subseteq [n]$ be the *initial* set of bad variable nodes. More precisely, V_0 is the set of all variable nodes that are bad in the $\ell(p)$-th iteration. We claim that $|V_0| \leq \frac{2p}{1-\beta(\mathtt{l}-1)} n$. (This is because for a variable to send a bad message it must have at least $\mathtt{l} - \beta(\mathtt{l}-1)$ incoming bad messages.) As we just discussed, for most graphs and noise realizations this is the case. As a worst case we assume that all its outgoing edges are bad. Let the set of check nodes connected to V_0 be C_0. These are the only check nodes that potentially can send bad messages in the next iteration. Therefore, we call C_0 the initial set of *bad* check nodes. Clearly,

$$|C_0| \leq \mathtt{l}|V_0|. \tag{10}$$

Consider a variable node and a fixed edge $\mathtt{e}$ connected to it: the outgoing message along $\mathtt{e}$ is determined by the received value as well as by the $(\mathtt{l}-1)$ incoming messages along the other $(\mathtt{l}-1)$ edges. Recall that if $\beta(\mathtt{l}-1)$ of those messages are good then the outgoing message along edge $\mathtt{e}$ is good. Therefore, if a variable node has $\beta(\mathtt{l}-1)+1$ good incoming messages, then *all* outgoing messages are good. We conclude that for a variable node to be bad at least $\mathtt{l} - \beta(\mathtt{l}-1)$ incoming messages must be bad. Therefore, it should connect to at least

$\mathtt{l}-\beta(\mathtt{l}-1)$ bad check nodes. This leaves at most $\beta(\mathtt{l}-1)$ edges that are connected to *new* check nodes.

We want to count the number of bad variables that are created in any of the future iterations. For convenience, once a variable becomes bad we will consider it to be bad for all future iterations. This implies that the set of bad variables is non-decreasing.

Let us now bound the number of bad variable nodes by the following process. The process proceeds in discrete steps. At each step t, consider the set of variables that are not contained in V_t but that are connected to at least $\mathtt{l}-\beta(\mathtt{l}-1)$ check nodes in C_t (the set of "bad" check nodes). If at time t no such variable exists stop the process. Otherwise, choose one such variable at random and add it to V_t. This gives us the set V_{t+1}. We also add all neighbors of this variable to C_t. This gives us the set C_{t+1}. By this we are adding the variable nodes that can potentially become bad and the check nodes that can potentially send bad messages to V_t and C_t respectively. As discussed above, for a good variable to become bad it must be connected to at least $\mathtt{l}-\beta(\mathtt{l}-1)$ check nodes that are connected to bad variable nodes. Therefore, at most $\beta(\mathtt{l}-1)$ new check nodes are added in each step. Hence, if the process continues then

$$|V_{t+1}| = |V_t| + 1, \tag{11}$$
$$|C_{t+1}| \leq |C_t| + \beta(\mathtt{l}-1). \tag{12}$$

By assumption, the graph is an element of $\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)$. Initially we have $|V_0| \leq \frac{2p}{1-\beta(1-1)}n = \frac{\alpha(1-1)(1-\beta)}{2(1-\beta(1-1))}n \leq \alpha n$. Therefore, as long as $|V_t| \leq \alpha n$,

$$\gamma\mathtt{l}|V_t| \leq |C_t|, \tag{13}$$

since C_t contains all neighbors of V_t. Let T denote the stopping time of the process, i.e., the smallest time at which no new variable can be added to V_t. We will now show that the stopping time is finite. We have

$$\begin{aligned}\gamma\mathtt{l}(|V_0|+t) &\overset{(11)}{=} \gamma\mathtt{l}|V_t| \overset{(13)}{\leq} |C_t| \overset{(12)}{\leq} |C_0| + t\beta(\mathtt{l}-1)\\ &\overset{(10)}{\leq} \mathtt{l}|V_0| + t\beta(\mathtt{l}-1).\end{aligned}$$

Solving for t this gives us

$$T \leq \frac{|V_0|\mathtt{l}(1-\gamma)}{\gamma\mathtt{l}-\beta(\mathtt{l}-1)}.$$

Therefore,

$$|V_T| \leq \frac{|V_0|\mathtt{l}(1-\gamma)}{\gamma\mathtt{l}-\beta(\mathtt{l}-1)} + |V_0| \leq \frac{2p}{\gamma\mathtt{l}-\beta(\mathtt{l}-1)}n = \alpha n, \tag{14}$$

where in the one before last step we used the fact that $|V_0| \leq \frac{2p}{1-\beta(1-1)}n$. The whole derivation so far was based on the assumption that $|V_t| \leq \alpha n$ for $0 \leq t \leq T$. But as we can see from the above equation, this condition is indeed verified ($|V_t|$ is non-decreasing and $|V_T| \leq \alpha n$).

Putting all these things together, we get

$$\begin{aligned}\mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell)] =& \mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\mathtt{E},\ell)(\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}} + \mathbb{1}_{\{(\mathtt{G},\mathtt{E})\notin A\}})]\\ \leq& \mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\mathtt{E},\ell)\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}}] + \mathbb{P}\{(\mathtt{G},\mathtt{E})\notin A\}\\ \leq& \mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\mathtt{E},\ell)\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}}\mathbb{1}_{\{\mathtt{G}\in\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)\}}]+\\ &\mathbb{P}\{\mathtt{G}\notin\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)\} + \mathbb{P}\{(\mathtt{G},\mathtt{E})\notin A\}.\end{aligned}$$

Apply $\limsup_{\ell\to\infty}$ on both sides of the inequality. According to (14) the first term is bounded by α. For the second term, since $\gamma < 1-\frac{1}{\mathtt{l}}$, we know from Theorem 2 that it is upper bounded by $O(n^{-(\mathtt{l}(1-\gamma)-1)})$. For the third term we know from (9) that it is bounded by $2e^{-Knp^2}$ for some strictly positive constant $K = K(\mathtt{l},\mathtt{r},p)$. Therefore, if we subsequently apply the limit $\lim_{n\to\infty}$ then we get

$$\lim_{n\to\infty}\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell)] \leq \alpha.$$

Since this conclusion is valid for any $0 < \alpha \leq \alpha_{\max}$ it follows that

$$\lim_{n\to\infty}\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell)] = 0.$$

■

Example 9 (BEC and BP): We know from Example 5 that $\beta(\mathtt{l}-1) = 1$. If we apply the conditions of Theorem 8, we see that we require $1/(\mathtt{l}-1) < 1$. Hence, the exchange of the limits is valid for $\mathtt{l} \geq 3$. Of course, for the BEC the exchange of limits in this regime follows directly by the monotonicity of the algorithm. ◊

Example 10 (BSC and GalB): We know from Example 7 that $\beta(\mathtt{l}-1) = \lceil(\mathtt{l}-1)/2\rceil + 1$. From Theorem 8 if $\epsilon < \epsilon^{\text{GalB}}$, the limits can be exchanged if $\mathtt{l}-1 > 1+\lceil(\mathtt{l}-1)/2\rceil$, i.e., for $\mathtt{l} \geq 5$. ◊

The key to applying expansion arguments to decoders with a continuous alphabet is to ensure that the received values are no longer dominant once DE has reached small error probabilities. This can be achieved by ensuring that the input alphabet is smaller than the message alphabet.

Definition 11 (Bounded MP Decoders): Given a MP decoder whose message passing alphabet is unbounded, i.e., it is equal to $\mathbb{R}$, we associate to it a *bounded* version. The *bounded* MP decoder with parameter $M \in \mathbb{R}^+$, denote it by MP(M), is identical to the standard MP decoder except that the reliability of the messages emitted by the check nodes is bounded to M before the messages are forwarded to the variable nodes. ◊

Note that the outgoing messages from the check nodes lie in $[-M, M]$ while the outgoing messages from the variable nodes can lie outside this range.

Example 12 (MS(M), BP(M) Decoders): The MS(M) decoder and the BP(M) decoder are identical to the standard min-sum (MS) and belief propagation (BP) decoder, except that the reliability of the messages emitted by the check nodes is bounded to M before the messages are forwarded to the variable nodes. ◊

Example 13 (MS(5) Decoder): Consider an $(\mathtt{l}\geq 5,\mathtt{r})$ ensemble and fix $M = 5$. Let the channel log-likelihoods belong to $[-1, 1]$. It is easy to check that in this case we can choose $G_{\mathtt{v}} = G_{\mathtt{c}} = [4, 5]$ and that it has strength $\beta \leq \frac{3}{4}$. Therefore, if the probability of outgoing messages from check nodes being in $[4, 5]$ goes to 1 under DE, then according to Theorem 8 the limits can be exchanged.

For example, consider BSC(ϵ) and LDPC$(5, 6)$ ensemble. It is known for this channel and MS decoder the messages are

of the form $k \log \frac{1-\epsilon}{\epsilon}$, for $k \in \mathbb{Z}$. Therefore we can restrict the message space to $\mathbb{Z}$ with the channel values mapped to $\{\pm 1\}$. Now, if we consider MS(5) decoder, the messages belong to $\{-5, \ldots, 5\}$. For this decoder, we can show that the limits can be exchanged till the DE threshold of 0.067. $\Diamond$

Example 14 (BP(10) Decoder): Let $\mathtt{l} = 5$ and $\mathtt{r} = 6$ and fix $M = 10$. Let the channel log-likelihoods belong to $[-3, 3]$. We claim that in this case the message subset pair $G_{\mathtt{v}} = [9, 10], G_{\mathtt{c}} = [14, 43]$ is good with strength $\beta = \frac{3}{4}$. This can be seen as follows: If all the incoming messages to a check node belong to $G_{\mathtt{c}}$, then the outgoing message is at least 12.39, which is mapped down to 10. Suppose that at a variable node at least $3 (= \beta(\mathtt{l}-1))$ out of the 4 incoming messages belong to $G_{\mathtt{v}}$. In this case the reliability of the outgoing message is at least $14 = 3 \times 9 - 10 - 3$. The maximum reliability is 43. Moreover, if all the incoming messages belong to $G_{\mathtt{v}}$ then the variable is decoded correctly. Therefore if the probability of outgoing messages from check nodes being in $[9, 10]$ goes to 1 in the DE limit then from Theorem 8, the limits can be exchanged.

For example, consider BSC(ϵ) with channel log-likelihoods restricted between $[-3, 3]$. For $\epsilon < \frac{1}{1+e^3}$, the log-likelihoods lie outside $[-3, 3]$ and hence they are mapped to $\{\pm 3\}$. In this case the limits can be exchanged till the DE threshold of 0.136. Note that this is what is done practice, since one has to work with bounded likelihoods. $\Diamond$

C. Expansion and Block Error Probability

In the previous section we considered the bit error probability. We will now derive sufficient conditions for the block error probability. Again we use expansion arguments but we proceed in a slightly different way.

Theorem 15 (Expansion and Block Error Probability): Consider an LDPC$(n, \mathtt{l}, \mathtt{r})$ ensemble, transmission over a BMS(ϵ) channel, and a symmetric MP decoder. Let β be the strength of the good message subset. If $\beta < \frac{\mathtt{l}-2}{\mathtt{l}-1}$ and if for some ϵ, $p_{\text{bad}}^{(\infty)} = 0$ then

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}_{\text{LDPC}(n,\mathtt{l},\mathtt{r})}[P_B^{\text{MP}}(\mathtt{G}, \epsilon, \ell)] = 0. \tag{15}$$

Proof: As in Theorem 8 we first perform a fixed number of iterations to bring down the bit error probability below a desired level. We then use Theorem 36 to show that for a graph with sufficient expansion the MP algorithm decodes the whole block correctly once the bit error probability is sufficiently small. This is very much in the spirit of Burshtein and Miller [5].

Define

$$\gamma = \left(1 - \frac{1}{\mathtt{l}}\right)\left(\frac{3+\beta}{4}\right).$$

Let $0 < \alpha < \alpha_{\max}(\gamma)$, where $\alpha_{\max}(\gamma)$ is the function defined in Theorem 2. Let $p = \frac{\alpha(\mathtt{l}-\beta(\mathtt{l}-1))}{2\mathtt{l}\mathtt{r}}$ and let $\ell(p)$ be the number of iterations such that $p_{\text{bad}}^{(\ell)} \le p$. Let Ω denote the space of code and noise realizations. Let $P_{\mathtt{e}}(\mathtt{G}, \mathtt{E}, \ell)$ denote the fraction of messages belonging to the bad set after ℓ iterations. Let $A \subseteq \Omega$ denote the subset

$$A = \{(\mathtt{G}, \mathtt{E}) \subseteq \Omega \,|\, P_{\mathtt{e}}(\mathtt{G}, \mathtt{E}, \ell(p)) \le 2p\}.$$

From (the Concentration) Theorem 39 we know that

$$\mathbb{P}\{(\mathtt{G}, \mathtt{E}) \notin A\} \le 2e^{-Knp^2} \tag{16}$$

for some strictly positive constant $K = K(\mathtt{l}, \mathtt{r}, p)$.

Since $\beta\frac{\mathtt{l}-1}{\mathtt{l}} \le 2\gamma - 1$ we can apply Theorem 36: if $\mathtt{G} \in \mathcal{X}(\mathtt{l}, \mathtt{r}, \alpha, \gamma)$ and if the initial number of bad messages is less than $\frac{\alpha}{\mathtt{l}\mathtt{r}}$ then all the messages will become good after a sufficient number of iterations.

Putting all these things together, we get

$$\begin{aligned}
\mathbb{E}[P_B^{\text{MP}}(\mathtt{G}, \epsilon, \ell)] =& \mathbb{E}[P_B^{\text{MP}}(\mathtt{G}, \mathtt{E}, \ell)(\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}} + \mathbb{1}_{\{(\mathtt{G},\mathtt{E})\notin A\}})] \\
\le& \mathbb{E}[P_B^{\text{MP}}(\mathtt{G}, \mathtt{E}, \ell)\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}}] + \mathbb{P}\{(\mathtt{G}, \mathtt{E}) \notin A\} \\
\le& \mathbb{E}[P_B^{\text{MP}}(\mathtt{G}, \mathtt{E}, \ell)\mathbb{1}_{\{(\mathtt{G},\mathtt{E})\in A\}}\mathbb{1}_{\{\mathtt{G}\in\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)\}}] + \\
& \mathbb{P}\{\mathtt{G} \notin \mathcal{X}(\mathtt{l}, \mathtt{r}, \alpha, \gamma)\} + \mathbb{P}\{(\mathtt{G}, \mathtt{E}) \notin A\}.
\end{aligned}$$

Apply $\limsup_{\ell\to\infty}$ on both sides of the inequality. According to Theorem 36 the first term is 0. For the second term, since $\gamma < 1 - \frac{1}{\mathtt{l}}$, we know from Theorem 2 that it is upper bounded by $O(n^{-(\mathtt{l}(1-\gamma)-1)})$. For the third term we know from (16) that it is bounded by $2e^{-Knp^2}$ for some strictly positive constant $K = K(\mathtt{l}, \mathtt{r}, p)$. Therefore, if we subsequently apply the limit $\lim_{n\to\infty}$ then we get

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}[P_B^{\text{MP}}(\mathtt{G}, \epsilon, \ell)] = 0.$$

■

Example 16 (BEC and BP): According to Theorem 8 we require $\mathtt{l} \ge 4$. Hence, if $\mathtt{l} \ge 4$ then the block error probability tends to zero below the BP threshold. $\Diamond$

Example 17 (BSC and GalB): As explained in Example 7 for the Gallager B algorithm over BSC, $\beta(\mathtt{l}-1) = 1 + \lceil(\mathtt{l}-1)/2\rceil$. The above condition implies if $\mathtt{l} - 2 > 1 + \lceil(\mathtt{l}-1)/2\rceil$, i.e., for $\mathtt{l} \ge 7$ the block error probability goes to zero below ϵ^{GalB}. $\Diamond$

Example 18 (MS(5) Decoder): Consider an $(\mathtt{l} \ge 7, \mathtt{r})$ ensemble and fix $M = 5$. Let the channel log-likelihoods belong to $[-1, 1]$. It is easy to check that in this case we can choose $G_{\mathtt{v}} = G_{\mathtt{c}} = [4, 5]$ and that it has strength $\beta \le \frac{2}{3}$. Therefore, if the probability of outgoing messages from check nodes being in $[4, 5]$ goes to 1 under DE then according to Theorem 15 the block error probability tends to 0. $\Diamond$

Example 19 (BP(10) Decoder): Let $\mathtt{l} = 7$ and $\mathtt{r} = 8$ and fix $M = 10$. Let the channel log-likelihoods belong to $[-1, 1]$. We claim that in this case the message subset pair $G_{\mathtt{v}} = [9, 10], G_{\mathtt{c}} = [15, 59]$ is good with strength $\beta = \frac{2}{3}$. Therefore if the probability of outgoing messages from check nodes being in $[9, 10]$ goes to 1 in the DE limit then from Theorem 15, the block error probability goes to zero. $\Diamond$

Theorem 8 has a stronger implication than Theorem 15 since it concerns the *block* error probability. Unfortunately, the required conditions are considerably more restrictive. We conjecture that in fact the conditions of Theorem 15 can be weakened by considering several stages of the algorithm jointly and that the required conditions are identical to the ones in Theorem 15.

Conjecture 20 (Expansion and Block Error Probability): Consider an LDPC$(n, \mathtt{l}, \mathtt{r})$ ensemble, transmission over a BMS(ϵ) channel, and a symmetric MP decoder. Let β be

the strength of the good message subset. If $\beta < 1$ and if for some ϵ, $p_{\text{bad}}^{(\infty)} = 0$ then

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}_{\text{LDPC}(n,\mathtt{l},\mathtt{r})}[P_B^{\text{MP}}(\mathtt{G},\epsilon,\ell)] = 0. \tag{17}$$

III. Sufficient Condition Based on Birth-Death Process

In the previous section we relied solely on the expansion of the graph to prove the validity of the limit exchange. As can be seen from the examples, for the decoders of interest the theorems are only valid for higher degrees, lets say $\mathtt{l} \geq 5$. Practical codes however typically have small degrees. In these cases expansion itself is not sufficient.

In more detail, the proofs in the previous section have two phases. In the first phase we run the MP algorithm for some fixed number of iterations to get the error probability down to a small constant. In the second phase we prove that the error probability stays close to 0 regardless of how many further iterations we perform and assuming pessimistically that all variables nodes have bad received values. This is too pessimistic an assumption for small degrees, where the received value plays an important role. In this section, we develop a method which takes the actual channel realization into account.

Consider a MP decoder operating on a message alphabet $\mathcal{M} \subseteq \mathbb{R}$. Further, for $\mu \in \mathcal{M}$, define $|\mu|$ to be the *reliability* of the message. This means that we define the reliability of $-\mu$ to be the same as the reliability of μ.

Most of the MP algorithms used in practice like GalB, BP, and MS, fall in the following category of *monotone* decoders.

Definition 21 (Monotone MP Decoders): We say that a symmetric MP decoder is monotone if the following conditions are fulfilled. At variable nodes the processing rules are monotone with respect to the natural order on $\mathcal{M}$; for a fixed received value, the outgoing message is a non-decreasing function of the incoming messages.

At check nodes the processing rules are monotone with respect to the natural order on the reliabilities; the reliability of the outgoing message is a non-decreasing function of the reliabilities of the incoming messages. $\diamondsuit$

Monotonicity is a useful property and it is also quite natural. A remaining difficulty in analyzing these decoders is that at check nodes the monotonicity is with respect to the reliability and not the message itself. We will see shortly how to get around this problem.

In what follows we mainly discuss the case of the GalB algorithm and $\mathtt{l} = 3$. The generalization to degree $\mathtt{l} \geq 4$ is straightforward and it is discussed in Section III-H. In this section we further give some examples of other monotone decoders to which the method can be extended.

A. Main Result and Outline

Lemma 22 (Exchange of Limits): Consider transmission over the BSC(ϵ) using random elements from the $(\mathtt{l},\mathtt{r})$-regular ensemble and decoding by the GalB algorithm. If $\epsilon < \epsilon^{\text{LGalB}}$ then

$$\lim_{n\to\infty} \limsup_{\ell\to\infty} \mathbb{E}[P_b^{\text{GalB}}(\mathtt{G},\epsilon,\ell)] = 0,$$

where ϵ^{LGalB} is the smallest parameter ϵ for which a solution to the following fixed point equation exists in $(0,\epsilon]$.

$$\begin{aligned} x = {} & \epsilon \sum_{k=0}^{\lfloor \frac{\mathtt{l}-1}{2} \rfloor} \binom{\mathtt{l}-1}{k} y^k (1-y)^{\mathtt{l}-1-k} \\ & + \bar{\epsilon} \sum_{k=\lfloor \frac{\mathtt{l}}{2} \rfloor + 1}^{\mathtt{l}-1} \binom{\mathtt{l}-1}{k} (1-y)^k y^{\mathtt{l}-1-k} \\ & + \frac{\mathbb{1}_{\{\frac{\mathtt{l}}{2} \in \mathbb{N}\}}}{2} \binom{\mathtt{l}-1}{\frac{\mathtt{l}}{2}} \Big(\epsilon y^{\frac{\mathtt{l}}{2}} (1-y)^{\frac{\mathtt{l}}{2}-1} + \bar{\epsilon} (1-y)^{\frac{\mathtt{l}}{2}} (y)^{\frac{\mathtt{l}}{2}-1} \Big), \end{aligned} \tag{18}$$

where $y = (1-x)^{\mathtt{r}-1}$. For the case of $(\mathtt{l}=3,\mathtt{r})$-regular ensemble this equation simplifies to

$$x = \bar{\epsilon}(1-(1-x)^{\mathtt{r}-1})^2 + \epsilon(1-(1-x)^{2(\mathtt{r}-1)}).$$

Discussion: Note that the threshold ϵ^{LGalB} introduced in the preceding lemma is in general slightly smaller than the DE threshold ϵ^{GalB}. We pose the extension of the result to channel values up to the DE threshold as an interesting open problem. It is likely to be difficult.

$\mathtt{r}$	rate	ϵ^{Sha}	ϵ^{GalB}	ϵ^{LGalB}
3	0.0	≈ 0.5	≈ 0.222	≈ 0.1705
4	0.25	≈ 0.2145	≈ 0.1068	≈ 0.0847
5	0.4	≈ 0.1461	≈ 0.06119	≈ 0.0506
6	0.5	≈ 0.11002	≈ 0.0394	≈ 0.0336
7	0.5714	≈ 0.08766	≈ 0.02751	≈ 0.02398
8	0.625	≈ 0.07245	≈ 0.02027	≈ 0.01795
9	0.667	≈ 0.06141	≈ 0.01554	≈ 0.01395
10	0.7	≈ 0.05324	≈ 0.01229	≈ 0.01115

TABLE I

Threshold values for some degree distributions with $\mathtt{l} = 3$.

$\mathtt{r}$	rate	ϵ^{Sha}	ϵ^{GalB}	ϵ^{LGalB}
4	0.0	≈ 0.5	≈ 0.0840	≈ 0.0697
5	0.2	≈ 0.1461	≈ 0.0464	≈ 0.0399
6	0.333	≈ 0.11002	≈ 0.0292	≈ 0.0258
7	0.4286	≈ 0.08766	≈ 0.0200	≈ 0.018
8	0.5	≈ 0.07245	≈ 0.0146	≈ 0.0133
9	0.556	≈ 0.06141	≈ 0.0111	≈ 0.0102
10	0.6	≈ 0.05324	≈ 0.0087	≈ 0.0081

TABLE II

Threshold values for some degree distributions with $\mathtt{l} = 4$.

Example 23: Table I shows thresholds for $\mathtt{l} = 3$, $\mathtt{r} = 3,\cdots,10$. For the $(\mathtt{l}=3,\mathtt{r}=6)$ degree distribution we have $\epsilon^{\text{LGalB}} \approx 0.0336$. This is slightly smaller than, but comparable to, $\epsilon^{\text{GalB}} \approx 0.0394$. $\diamondsuit$

We proceed by a sequence of simplifications, ensuring in each step that the modified algorithm is an upper bound on the original process. In Section III-B we simplify the decoder by "linearizing" the processing rules at the check nodes. In Section III-C we further upper bound the process by considering the marking process associated with the decoding algorithm. In Section III-D we construct a witness for the marking process and derive bounds on the size of such a witness. In Section III-E we then show that, conditioned on the witness, we can consider the channel realizations outside

the witness to be random and independent of the witness. In Section III-F we use an expansion argument to bound the stopping time of the birth and death process associated with the marking process. Finally, in Section III-G we combine all previous statements to derive at our conclusion.

B. Linearized Gallager Algorithm B

We proceed as in Section II: Fix $0 \leq \epsilon < \epsilon^{\text{LGalB}}$. We prove that for every $\alpha > 0$ there exists an $n(\alpha, \epsilon)$ so that $\limsup_{\ell \to \infty} \mathbb{E}[P_b^{\text{GalB}}(\mathtt{G}, \epsilon, \ell)] < \alpha$ for $n \geq n(\alpha, \epsilon)$.

Without loss of generality we can assume that the all-one codeword was sent. We will make this assumption throughout the remainder of this section. Therefore, the message 1 signifies in the sequel a *correct* message, whereas -1 implies that the message is *incorrect*.

For this setting, we define the following *linearized* version of the decoder.

Definition 24 (Linearized GalB): The *linearized* GalB decoder, denoted by LGalB, is defined as follows: at the variable node the computation rule is same as that of the GalB decoder. At the check node the outgoing message is the minimum of the incoming messages.

Discussion: The LGalB is not a practical decoding algorithm but rather a convenient device for analysis; it is understood that we assume that the all-one codeword was transmitted and that quantities like the error probability refer to the variables decoded as -1. By some abuse of notation, we nevertheless refer to it as a decoder.

The LGalB decoder is monotone also with respect to the incoming messages at check nodes. Moreover, it satisfies the following property.

Lemma 25 (LGalB is Upper Bound on GalB): For any graph $\mathtt{G}$, any noise realization $\mathtt{E}$, any starting set of "bad" edges, and any ℓ, we have $P_{\mathtt{e}}^{\text{GalB}}(\mathtt{G}, \mathtt{E}, \ell) \leq P_{\mathtt{e}}^{\text{LGalB}}(\mathtt{G}, \mathtt{E}, \ell)$, where $P_{\mathtt{e}}(\mathtt{G}, \mathtt{E}, \ell)$ denotes the fraction of erroneous messages after ℓ iterations of decoding.

Proof: Consider one iteration, i.e., a check-node step followed by a variable-node step. Let $\mathcal{B}_\ell^{\text{GalB}/\text{LGalB}}$ denote the set of bad edges (edges with message -1) after the ℓ-th iteration of GalB and LGalB, respectively. Let $\psi_{\mathtt{E}}^{\text{GalB}/\text{LGalB}}(\mathcal{B})$ denote the set of bad edges after one iteration assuming that the initial such set is $\mathcal{B}$.

We use the following two facts: (i) The outgoing messages for the LGalB decoder at variable/check nodes are monotone; if we decrease (with respect to the natural order on $\mathcal{M}$) the input at a variable/check node then the output is either decreased or stays the same. I.e., if $\mathcal{B} \subseteq \mathcal{B}'$, meaning that the messages in $\mathcal{B}'$ can be obtained by decreasing some of the $+1$ messages in $\mathcal{B}$ to -1, then $\psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B}) \subseteq \psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B}')$. (ii) For any set of input messages, the outgoing message of LGalB is less than or equal to the message of the GalB decoder, i.e., $\psi_{\mathtt{E}}^{\text{GalB}}(\mathcal{B}) \subseteq \psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B})$.

For the proof, we proceed by induction. Let $\mathcal{B}_0$ be the initial set of bad edges. After the first iteration, from (ii) we get $\mathcal{B}_1^{\text{GalB}} = \psi_{\mathtt{E}}^{\text{GalB}}(\mathcal{B}_0) \subseteq \psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B}_0) = \mathcal{B}_1^{\text{LGalB}}$. To complete the proof it is sufficient to show that $\mathcal{B}_\ell^{\text{GalB}} \subseteq \mathcal{B}_\ell^{\text{LGalB}}$ implies $\mathcal{B}_{\ell+1}^{\text{GalB}} \subseteq \mathcal{B}_{\ell+1}^{\text{LGalB}}$. Using (i) and (ii) we have $\mathcal{B}_{\ell+1}^{\text{LGalB}} = \psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B}_\ell^{\text{LGalB}}) \supseteq \psi_{\mathtt{E}}^{\text{LGalB}}(\mathcal{B}_\ell^{\text{GalB}}) \supseteq \psi_{\mathtt{E}}^{\text{GalB}}(\mathcal{B}_\ell^{\text{GalB}}) = \mathcal{B}_{\ell+1}^{\text{GalB}}$ and hence the lemma. ■

From the above lemma it suffices to prove the exchange of limits for the linearized algorithm. Note that ϵ^{LGalB} as defined in Lemma 22 is the threshold of the LGalB algorithm. We will prove that for every $0 \leq \epsilon < \epsilon^{\text{LGalB}}$ and every $\alpha > 0$ there exists an $n(\alpha, \epsilon)$ so that $\limsup_{\ell \to \infty} \mathbb{E}[P_b^{\text{LGalB}}(\mathtt{G}, \epsilon, \ell)] < \alpha$ for $n \geq n(\alpha, \epsilon)$. As we will see later, the monotonicity property of LGalB considerably simplifies the analysis. But the price paid for the simplification is that the technique works only for $\epsilon < \epsilon^{\text{LGalB}}$, which is slightly smaller than the DE threshold.

C. Marking Process

Rather than analyzing the LGalB algorithm directly, we analyze the associated *marking process*. This process is monotone as a function of the iterations.

More precisely, we split the process into two phases: we start with LGalB for $\ell(p)$ iterations to get the error probability below p; we then continue the marking process associated with an infinite number of further iterations of LGalB. This means that we mark any variable that is bad in at least one iteration $\ell \geq \ell(p)$. Clearly, the union of all variables that are bad at at least one point in time $\ell \geq \ell(p)$ is an upper bound on the maximum number of variables that are bad at any specific instance in time.

The standard *schedule* of the LGalB is parallel, i.e., all incoming messages (at either variable or check nodes) are processed at the same time. This is the natural schedule for an actual implementation. For the purpose of analysis it is convenient to consider an *asynchronous* schedule.

Here is how the general asynchronous marking process proceeds. We are given a graph $\mathtt{G}$ and a noise realization $\mathtt{E}$. We are also given a set of *marked* edges. These marked edges are directed, from variable node to check node. At the start of the process mark the variable nodes that are connected to the marked edges. Declare all other variables and edges as *unmarked*. Unmarked edges do not have a direction. The process proceeds in discrete steps. At each step we pick a marked edge and we perform the processing described below. We continue until no more marked edges are left. Here are the processing rules:

If the marked edge $\mathtt{e}$ goes from variable to check:

- Let $\mathtt{c}$ be the check node connected to $\mathtt{e}$. Declare $\mathtt{e}$ to be *unmarked* but *mark* all other edges connected to $\mathtt{c}$; orient these marked edges from check to variable;

If the marked edge $\mathtt{e}$ goes from check to variable:

- Let $\mathtt{v}$ be the connected variable node. If $\mathtt{v}$ has a *good* associated channel realization and $\mathtt{v}$ is unmarked then mark $\mathtt{v}$ and declare $\mathtt{e}$ to be unmarked.
- Let $\mathtt{v}$ be the connected variable node. If $\mathtt{v}$ has an associated *bad* channel realization or if $\mathtt{v}$ has an associated *good* channel realization but is *marked*: (i) mark $\mathtt{v}$ and all its outgoing edges; (ii) orient the edges from variable to check; (iii) unmark $\mathtt{e}$.

Let $\mathcal{M}(\mathtt{G}, \mathtt{E}, \mathcal{S})$ denote the set of marked variables assuming that we start with the set of marked edges $\mathcal{S}$ and that we run the asynchronous marking process. Let $M(\mathtt{G}, \mathtt{E}, \mathcal{S}) = |\mathcal{M}(\mathtt{G}, \mathtt{E}, \mathcal{S})|$. As a special case, let $\mathcal{M}(\mathtt{G}, \mathtt{E}, \ell)$ denote the set of marked variables at the end of the process assuming that

the initial set of marked edges is the set of bad edges after ℓ rounds of LGalB. As before, $M(\mathtt{G},\mathtt{E},\ell)=|\mathcal{M}(\mathtt{G},\mathtt{E},\ell)|$.

It is not hard to see that for any $\ell'\geq\ell$, $P_b^{\text{LGalB}}(\mathtt{G},\epsilon,\ell')\leq M(\mathtt{G},\mathtt{E},\ell)/n$: for $\ell'=\ell$ both processes start with the same set of bad edges and both are operating on the same graph and noise realization. At the check-node side the processing rules are identical. At the variable-node side both processes also behave in the same way if they encounter a variable node with a bad channel realization. The difference lies in the behavior when they encounter a variable node with a good channel realization. In such a case the outgoing message for the LGalB is bad only if there are two bad messages entering *at the same time instance*. The asynchronous marking process algorithm declares the outgoing message to be bad if there are two incoming bad messages, even if the two messages might correspond to different time instances as measured by the parallel schedule. We conclude that for $\ell'\in\mathbb{N}$

$$\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{LGalB}}(\mathtt{G},\epsilon,\ell)]\leq\frac{1}{n}\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell')]. \tag{19}$$

D. Witness

It remains to bound $\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]$. Assume at first that we take a random graph $\mathtt{G}$ and a random noise realization $\mathtt{E}$ and that we start the marking process with a sufficiently small *random* set of marked edges (and not the set of bad edges after ℓ iterations of LGalB). In this case one can show that the number of marked nodes at the end of the process is with high probability not more than a constant multiple of the size of the starting set. To prove this statement, we use the fact that the graph, the noise, and the starting set of edges are all independent. Therefore, the marking process behaves essentially like a birth and death process: we pick an edge and we explore its neighborhood; with a certain probability the edge dies (if it enters a variable node with a correctly received value) and with a certain probability the edge spawns some children. As long as the expected number of new children is less than 1 the process eventually dies with probability 1.

Unfortunately our situation is more involved. After ℓ iterations the starting set of marked edges is correlated, both with the graph as well as with the noise realization. Our aim therefore is to reduce this correlated case to the uncorrelated case by a sequence of transformations. As a first step we show how to get rid of the correlation with respect to the noise realization.

Consider a fixed graph $\mathtt{G}$. Assume that we have performed ℓ iterations of LGalB. For each edge $\mathtt{e}$ that is bad in the ℓ-th iteration we construct a "witness." A witness for $\mathtt{e}$ is a subset of the computation tree of height ℓ (where height is counted as the number of variable node levels) for $\mathtt{e}$ consisting of paths that carried bad messages in the past iterations. We construct the witness recursively starting with $\mathtt{e}$. Orient $\mathtt{e}$ from check node to variable node. At any point in time while constructing the witness associated with $\mathtt{e}$ we have a partial witness that is a tree with oriented edges. The initial such partial witness is $\mathtt{e}$. One step in the construction consists of taking a leaf edge of the partial witness and to "grow it out" according to the following rules.

If an edge enters a variable node that has an incorrect received value then add the *smallest* (according to some fixed but arbitrary order on the set of edges) edge that carries an incorrect incoming message to the witness and continue the process along this edge. The added edge is directed from variable node to check node. If an edge enters a variable node that has a correct received value then add both incoming edges to the witness and follow the process along both edges. (Note that in this case both of these edges must have carried bad messages.) Again, both of these edges are directed from variable to check node. If an edge enters a check node then choose the smallest incoming edge that carries an incorrect message and add it to the witness. Continue the process along this edge. The added edge is directed from check to variable node. Continue the process until depth ℓ. Fig. 1 shows an example for $\mathtt{l}=3$, $\mathtt{r}=4$, and $\ell=3$.

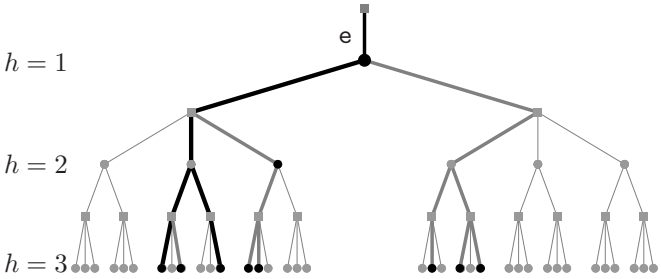

Fig. 1. Construction of the witness for a bad edge $\mathtt{e}$. The *dark* variables represent channel errors. The part of the tree with *dark* edges represent the witness, the *thick* edges, including both dark and gray, represent the bad messages in the past iterations. The number h in the left indicates the height of the tree.

Denote the union of all witnesses for all edges that are bad in the ℓ-th iteration by $\mathcal{W}(\mathtt{G},\mathtt{E},\ell)$. We simply call it *the witness*. The witness is a part of the graph that on its own explains why the set of bad edges after ℓ iterations is bad.

How large is $\mathcal{W}$? The larger ℓ, the fewer bad edges we expect to see in iteration ℓ. On the other hand, the size of the witness for each bad edge grows as a function of ℓ. The next lemma, whose proof can be found in Appendix B, asserts that the first effect dominates and that the expected size of $\mathcal{W}$ converges to zero as the number of iterations increases.

Lemma 26 (Size of Witness): Consider the $(3,\mathtt{r})$-regular ensemble. For $0\leq\epsilon<\epsilon^{\text{LGalB}}$,

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|]=o_\ell(1).$$

Why do we construct a witness? It is intuitive that if we keep the witness fixed but randomize the structure as well as the received values on the remainder of the graph then the situation should only get worse: already the witness itself explains all the bad messages and hence any further bad channel values can only create more bad messages. In the next two sections we show that under some suitable technical conditions this intuition is indeed correct.

E. Randomization

A witness $\mathcal{W}$ consists of two parts, (i) the graph structure of $\mathcal{W}$ and (ii) the channel realizations of the variables in $\mathcal{W}$. We will often need to refer to either of these parts on their own. By some abuse of notation we write $\mathcal{W}$ also if we refer only to the graph structure or only to the channel realizations. The usage should be clear from the context. As an example, we write $\mathcal{W} \subseteq \mathtt{G}$ to indicate that $\mathtt{G}$ contains $\mathcal{W}$ as a subgraph and we write $\mathcal{W} \subseteq \mathtt{E}$ to indicate that the received values of all variables in $\mathcal{W}$ agree with the values that these variables take on in $\mathtt{E}$.

Fix a graph $\mathtt{G}$ and a witness $\mathcal{W}$, $\mathcal{W} \subseteq \mathtt{G}$. Let $\mathcal{E}_{\mathtt{G},\mathcal{W}}$ denote the set of all error realizations $\mathtt{E}$ that give rise to $\mathcal{W}$, i.e., $\mathcal{W}(\mathtt{G}, \mathtt{E}, \ell) = \mathcal{W}$. Clearly, for all $\mathtt{E} \in \mathcal{E}_{\mathtt{G},\mathcal{W}}$ we must have $\mathcal{W} \subseteq \mathtt{E}$. In words, on the set of variables fixed by the witness the errors are fixed by the witness itself. Therefore, the various $\mathtt{E}$ that create this witness differ only on $\mathtt{G}\backslash\mathcal{W}$. As a convention, we define $\mathcal{E}_{\mathtt{G},\mathcal{W}} = \emptyset$ if $\mathcal{W} \not\subseteq \mathtt{G}$.

Let $\mathcal{E}'_{\mathtt{G},\mathcal{W}}$ denote the set of projections of $\mathcal{E}_{\mathtt{G},\mathcal{W}}$ onto the variables in $\mathtt{G}\backslash\mathcal{W}$. Let $\mathtt{E}' \in \mathcal{E}'_{\mathtt{G},\mathcal{W}}$. Think of $\mathtt{E}'$ as an element of $\{0,1\}^{|\mathtt{G}\backslash\mathcal{W}|}$, where 0 denotes a correct received value and 1 denotes an incorrect received value. In this way, $\mathcal{E}'_{\mathtt{G},\mathcal{W}}$ is a subset of $\{0,1\}^{|\mathtt{G}\backslash\mathcal{W}|}$.

This is important: $\mathcal{E}'_{\mathtt{G},\mathcal{W}}$ has structure. We claim that, if $\mathtt{E}' \in \mathcal{E}'_{\mathtt{G},\mathcal{W}}$ then $\mathcal{E}'_{\mathtt{G},\mathcal{W}}$ also contains $\mathtt{E}'_{\leq}$ (as defined in Appendix D). More precisely, if the noise realization $\mathtt{E}' \in \mathcal{E}'_{\mathtt{G},\mathcal{W}}$ gives rise to the witness $\mathcal{W}$ then converting any incorrect received value in $\mathtt{E}'$ to a correct one will also give rise to $\mathcal{W}$. This is true since the LGalB algorithm is monotone, so that taking away some incorrectly received values can not increase the size of bad edges observed in the ℓ-th iteration. But on the other hand, $\mathcal{W}$ itself ensures that the set of bad edges after ℓ iterations includes all the bad edges we saw originally. The proof of the following lemma relies heavily on this property.

Lemma 27 (Channel Randomization): Fix $\mathtt{G}$ and let $\mathcal{W} \subseteq \mathtt{G}$. Let $\mathbb{E}_{\mathtt{E}'}[\cdot]$ denote the expectation with respect to the channel realizations $\mathtt{E}'$ in $\mathtt{G}\backslash\mathcal{W}$. Then

$$\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}] \leq \mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]\mathbb{E}_{\mathtt{E}'}[\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}]. \quad (20)$$

Discussion: Lemma 27 has the following important operational significance. If we divide both sides by $\mathbb{E}_{\mathtt{E}'}[\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}]$, the left-hand side is the expectation of marked variables, where the expectation is computed over all those channel realizations that give rise to the given witness $\mathcal{W}$, whereas the right-hand side gives the expectation over all channel realizations (outside the witness) regardless whether they give rise to $\mathcal{W}$ or not. Clearly, the right-hand side is much easier to compute, since the channel is now independent of $\mathcal{W}$. The lemma states that, if we assume that the channel outside $\mathcal{W}$ is independently chosen then we get an upper bound on the size of the marked variables.

Proof: Let $n' = |\mathtt{G}\backslash\mathcal{W}|$. Let $P\{\cdot\}$ be the probability measure associated with $\mathbb{E}_{\mathtt{E}'}[\cdot]$, i.e., $P\{\mathtt{E}'\} = \epsilon^{n_1}\bar{\epsilon}^{n'-n_1}$, where n_1 denotes the number of ones in $\mathtt{E}'$. Let $f(\mathtt{E}')$ denote the function $\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}$, and let $g(\mathtt{E}')$ denote the function $M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})$. Note that f is a decreasing function on $\{0,1\}^{n'}$ because if $f(\mathtt{E}')=1$ then for all $\mathtt{E}'' \leq \mathtt{E}'$, $f(\mathtt{E}'')=1$. Further, g is an increasing in $\{0,1\}^{n'}$ since LGalB is monotone in the number of channel errors. Since $g(\mathtt{E}') \leq n$, $n-g$ is non-negative and it is a decreasing function. For $s,t \in \{0,1\}^{n'}$, let $|s|$ denote the number of 1s in s and $s\vee t$ and $s\wedge t$ be as defined in Appendix D. Then,

$$\begin{aligned} P\{s\}P\{t\} &= \epsilon^{|s|+|t|}(\bar{\epsilon})^{n'-|s|-|t|}, \\ P\{s\vee t\} &= \epsilon^{|s|+|t|-|s\wedge t|}(\bar{\epsilon})^{n'-(|s|+|t|-|s\wedge t|)}, \\ P\{s\wedge t\} &= \epsilon^{|s\wedge t|}(\bar{\epsilon})^{n'-|s\wedge t|}. \end{aligned}$$

Therefore, $P\{s\}P\{t\} = P\{s\vee t\}P\{s\wedge t\}$. Applying the FKG inequality in the form of Lemma 37 to f and $n-g$, we get

$$\mathbb{E}[f(n-g)] \geq \mathbb{E}[f]\mathbb{E}[n-g].$$

This implies $\mathbb{E}[fg] \leq \mathbb{E}[f]\mathbb{E}[g]$. ∎

We can now upper bound the right-hand side of (19). The proof of the next lemma can be found in Appendix C.

Lemma 28 (Markov Inequality): Consider the $(\mathtt{l}=3,\mathtt{r})$-regular ensemble and transmission over the BSC(ϵ). Let $(\mathtt{G},\mathtt{E})$ be chosen uniformly at random. Let $\ell \in \mathbb{N}$ and $\theta > 0$ so that $\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|] \leq \theta^2 n$. Then

$$\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)] \leq \sum_{\mathcal{W}:|\mathcal{W}|\leq\theta n}\sum_{\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})] + \theta n.$$

F. Back to Expansion

In the previous section we have shown that for a fixed graph $\mathtt{G}$, and a given witness $\mathcal{W}$, we can ignore the correlations between the witness and the *channel values* in $\mathtt{G}\backslash\mathcal{W}$ and consider those channel values to be chosen independently. But the *graph structure* of $\mathtt{G}\backslash\mathcal{W}$ is still correlated with $\mathcal{W}$. Let us now deal with this correlation and get a bound on the marking process for those $\mathtt{G}$ that have an expansion close to the typical one of the ensemble.

Consider the following random process, which we call the R-process. The process proceeds in discrete steps and has *state* (C_t, S_t, B_t, I_t) at *time* t, where each component is an integer. We initialize the process with $(C_0,S_0,B_0,I_0) = (0,S_0,0,0)$, where $S_0 \in \mathbb{N}$.

At each step we have two choices. We can either perform a *regular* step or a *boundary* step. The effect of each step type on the state (C_t,S_t,B_t,I_t) is shown in Table III. If we choose a regular step then, with probability ϵ, an *extension* step is executed and, with probability $\bar{\epsilon}$, a *pruning* step is performed. The choices of extension step versus pruning step are iid.

In our choice of step type we are restricted by the following: at any time during the process the state has to satisfy

$$\gamma\mathtt{r}C_t \leq S_t + B_t + I_t, \quad (21)$$

where $\gamma = 1 - \frac{1+\delta}{\mathtt{r}}$ for some strictly positive number δ. Let T be the smallest time t so that $S_t = 0$. It is convenient to formally define the process for all t by setting $U_t = U_T$ for $t \geq T$.

Discussion: Here is the interpretation of the above process. We are given a fixed graph $\mathtt{G}$ and a witness $\mathcal{W}$. The channel

	C_t	S_t	B_t	I_t
regular extend	$\mathbf{2}$	$\mathbf{2r-3}$	$\mathbf{0}$	$\mathbf{1}$
	1	$\mathtt{r}-3$	0	1
	0	-3	0	1
regular prune	$\mathbf{0}$	$\mathbf{-1}$	$\mathbf{1}$	$\mathbf{0}$
boundary	$\mathbf{1}$	$\mathbf{\mathtt{r}-2}$	$\mathbf{-1}$	$\mathbf{1}$
	0	-2	-1	1

TABLE III

POSSIBLE STATE TRANSITIONS. NOTE THAT THERE ARE SEVERAL POSSIBLE TRANSITIONS CORRESPONDING TO A "REGULAR EXTEND" STEP AS WELL AS A "BOUNDARY" STEP. AS EXPLAINED BELOW, THE TRANSITIONS INDICATED IN BOLD LETTERS DOMINATE THE OTHER TRANSITIONS IN THE SENSE OF DEFINITION 29.

realizations in $\mathtt{G}\backslash\mathcal{W}$ are generated independently with probability of error ϵ. We are interested in computing the expected number of marked variables $\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]$.

The components of the state vector have the following interpretation. By some further abuse of notation, let $\mathcal{W}$ refer now also to the variables contained in $\mathcal{W}$. Let $\mathcal{N}(\mathcal{W})$ denote all the check nodes that neighbor $\mathcal{W}$. We start our process with those edges connected to $\mathcal{N}(\mathcal{W})$ that do not connect to $\mathcal{W}$. The cardinality of this set is denoted by S_0 (where the "s" stands for *surviving*). In each step we take a single edge from this set of surviving edges and "grow it out."

Let us discuss this process in more detail. When we "grow out" an edge we first visit the connected variable node. Suppose that this is the first time that the process visits this variable node. We call this a *regular* step.

If the received value of this variable node is good then we stop the process along this edge. We add the variable to the *boundary* set to make a mental note that we have seen this node exactly once. The boundary set has cardinality B_t. We further subtract 1 from S_t to take into account that we finished processing one of the "surviving" edges.

If the received value is bad then we add this variable node to the *internal* variable nodes. The cardinality of this set is I_t. This means that in this step we increase I_t by 1. Further, we expand the graph along the two outgoing edges, add the (at most) two connected check nodes to the set of internal check nodes (whose cardinality is denoted by C_t) and add all the remaining edges that emanate from these check nodes to the set of surviving edges. This adds (at most) $2(\mathtt{r}-1)$ new survivors, but we have to subtract the edge we started from. Therefore, S_t is increased by at most $2\mathtt{r}-3$.

So far we have assumed that we have not seen the variable node (that is connected to the edge which we grow out) before. Suppose now that, to the contrary, the variable is an element of the boundary. We know that in this case the received value is good, but we also know that the variable received another bad incoming message. Therefore, the variable will send a bad outgoing message along its remaining edge. Hence, we move this variable node from the boundary to the internal set (this decreases B_t by 1 and increases I_t by 1). Further, we grow out the graph along the only remaining outgoing edge. This adds at most one new check node and at most $\mathtt{r}-1$ outgoing edges to the set of surviving edges. Discounting again the edge we started with, we add in total at most $\mathtt{r}-2$ to S_t.

Suppose that the graph $\mathtt{G}$ is a right expander; i.e., $\mathtt{G}\in\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)$, where $\gamma\geq 1-\frac{1+\delta}{\mathtt{r}}$ for some strictly positive δ. This means that every collection $\mathcal{C}$ of check nodes of size at most αm has at least $\gamma|\mathcal{C}|\mathtt{r}$ connected variable nodes. Consider the state of the system at some time t. At this point in time we have C_t check nodes. All these check nodes are "internal," i.e., all their neighboring variable nodes are either counted in V_t or I_t, or they are yet to be encountered by the process which cannot be more than the survivors set S_t. We know that $\mathtt{G}$ is an expander and suppose for now that $C_t\leq\alpha m$. Then we know that the number of connected variable neighbors must be at least $\gamma\mathtt{r}C_t$, i.e., at any time during the process the state should satisfy

$$\gamma\mathtt{r}C_t\leq S_t+B_t+I_t. \tag{22}$$

We claim that

$$\gamma\mathtt{r}C_t\leq S_t+B_t+I_t-(1-\delta) \tag{23}$$

is a necessary condition to be able to perform a boundary step at time t. To see this, suppose we take a boundary step. If you look at Table III you will see that there are two possible transitions. One can check that the transition stated in bold letters gives the less restrictive condition. Let us therefore only focus on this case. The state after applying the boundary state must still fulfill (22). This means that we must have

$$\gamma\mathtt{r}(C_t+1)\leq(S_t+\mathtt{r}-2)+(B_t-1)+(I_t+1).$$

The claim is proved by rewriting this inequality.

From the above discussion we claim that for a given $\mathcal{W}$ and $\mathtt{G}$, where $\mathtt{G}\in\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)$, as long as $C_t\leq\alpha m$ then the marking process can be modeled as the R-process. The random variable I_∞ is equal to the random variable $M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})-|\mathcal{W}|$ of the marking process (we subtract the size of witness because we do not include it in the internal variables). For the actual marking process the decision of whether a regular step or a boundary step is taken is forced by the structure of the graph and our choice of which edge to grow out. For the R-process the role of graph is taken by a *strategy*. A strategy is any (randomized) decision function F that, based on the initial state and past decisions and outcomes, decides whether a regular step or a boundary step is taken at any point in time.

Here is the connection between the actual physical process and the R-process in more detail. Assume we are given a graph $\mathtt{G}$ and a witness $\mathcal{W}$. We know the graph and therefore we also know which edges of the graph are elements of the surviving set. Therefore, when we pick a survivor, we know in advance whether the step is a regular step or a boundary step. The noise realization, which is not known to us a priori, determines whether a regular step is a regular extend or prune step. We see that each graph gives rise to a strategy. As long as the size of all revealed nodes is sufficiently small this strategy will be admissible since the expansion will be valid up to this point.

Since we are only interested in an upper bound on the number of marked variables, we allow the R-process to use an arbitrary *strategy*, only limited by the condition (22). We

call a strategy which obeys (22) an *admissible* strategy. Since the actual physical process is also limited by (22) (under the condition that the graph is an expander and the process has not grown beyond the size where the expansion is valid), it suffices to derive upper bounds on $\mathbb{E}[I_\infty]$ that is valid for all choices of the strategy.

We relax one further restriction imposed by the actual physical process in order to simplify our task. Again, this only increases $\mathbb{E}[I_\infty]$. In the marking process, we can only perform a boundary step if the boundary set is strictly positive. In other words, we require $B_t > 0$ for a boundary step to be performed. We lift this restriction for the R-process.

Definition 29 (Ordering of States): The state $U \equiv (C, S, B, I)$ dominates the state $U' \equiv (C', S', B', I')$, denoted by $U \geq U'$ if

(i) $S \geq S'$,
(ii) $I \geq I'$,
(iii) $S + B + I - \gamma \mathtt{r} C \geq S' + B' + I' - \gamma \mathtt{r} C'$.

$\Diamond$

Lemma 30 (Monotonicity of I_∞ with State): Consider the R-process with admissible strategy F' and initial state $U' \equiv (C', S', B', I')$. Let $U \equiv (C, S, B, I)$ be an initial state which dominates U', i.e., $U \geq U'$. Then there exists an admissible strategy F so that $\mathbb{E}[I_\infty(U, F)] \geq \mathbb{E}[I_\infty(U', F')]$, where $I_\infty(U, F)$ denotes I_∞ assuming that the R-process is initialized with U and that the process uses the strategy F.

Proof: Given U' and the admissible strategy F' we construct the admissible strategy F in the following way. The process with initial state U uses strategy F' but applies it to the *pseudo* state U'. Further, it updates its pseudo state according to the realization of the process and bases its future decisions on strategy F' applied to this evolving pseudo state. Call the phase of the process until the pseudo state has reached $S' = 0$ the "initial" phase of the process. At that point the (U, F) process switches to any admissible strategy based on its real state. To be concrete, assume that it uses a *greedy strategy* at this point. This means that the process performs a boundary step any time it is admissible.

In order to show the desired inequality on the expected values we couple the processes (U', F') and (U, F). We imagine that we run both processes in parallel and that they experience exactly the same randomness (this refers to the randomness contained in the choice of the transitions as well as any randomness which might be used by the strategy). Assume for the moment that strategy F is admissible.

In the initial phase of the algorithm (until the (U', F') process stops because $S'_t = 0$) the (U, F) process proceeds in lock-step with the (U', S') process. Since $S_0 \geq S'_0$ and since $S_t - S_0 = S'_t - S'_0$ it follows that $S_t \geq S'_t$ in this initial phase. This means that the process (U, F) never stops before the process (U', F'). Further, $I_0 \geq I'_0$, $I_t - I_0 = I'_t - I'_0$, and I_t is a non-decreasing function. It follows that for every realization $I_\infty(U, F) \geq I_\infty(U', F')$. This implies, *a fortiori*, the claimed inequality on the expected values.

Let us now show that the protocol F is admissible. We claim that for all $t \in \mathbb{N}$

$$S_t + B_t + I_t - \gamma \mathtt{r} C_t \geq S'_t + B'_t + I'_t - \gamma \mathtt{r} C'_t. \tag{24}$$

By definition this is true for $t = 0$. But by construction of the coupling, $S_t - S_0 = S'_t - S'_0$, $I_t - I_0 = I'_t - I'_0$, $B_t - B_0 = B'_t - B'_0$, and $C_t - C_0 = C'_t - C'_0$. It follows that the left-hand side in (24) is always at least as large as the right-hand side. Therefore, if F' is admissible then so is F. ■

From Table III we see that for regular extend and boundary steps there are several possible outcomes. For each of these two steps, there is a single outcome (highlighted in the table) whose resulting state dominates those of the other outcomes. Since we are interested in an upper bound on I_∞, thanks to the above lemma, we can restrict our attention to these dominating steps.

Consider the *greedy* strategy, call if F^g. For this greedy strategy, whenever (23) is true we perform a boundary step.

Lemma 31 (Domination of the Greedy Process): For a given initial state $U = (C_0, S_0, B_0, I_0)$ and any admissible strategy F, we have

$$\mathbb{E}[I_\infty(U, F^g)] \geq \mathbb{E}[I_\infty(U, F)].$$

Proof: Again we construct a coupling between the processes (U, F) and (U, F^g). As remarked above, for both processes we can assume that the state transitions are the ones indicated in bold in Table III. The only randomness therefore resides in whether for a regular step the process *extends* or *prunes* and, possibly, in the randomness used for the strategy F. There is no randomness involved in any boundary steps. The coupling consists in coupling for each regular step i, $i \in \mathbb{N}$, the outcomes of these regular steps. In more detail, if for the process (U, F) the i-th regular step results in a pruning then the same occurs for the i-th regular step for the process (U, F^g). By construction, for all regular steps the change of S, I, B, and C is the same for both processes. Assume we measure "time" not in the absolute number of steps taken but by the number of regular steps taken. Consider a process (U, F) and assume that this process is still "alive" at 'time t. Then its state U_t only depends on the realization of the random variables during the regular steps and on the total number of boundary steps taken, but it does not depend on the order of the steps taken.

Since the process (U, F^g) has by definition done at least as many boundary steps as the process (U, F) it further follows that if we compare the two processes at "time" i corresponding to i regular steps then the number of survivors (and also the number of internal nodes) for (U, F^g) is at least as large as the number of survivors for (U, F). Therefore, if at this time the process (U, F) is still alive then so is the process (U, F^g) and the latter has at least as many accumulated internal variable nodes as the former. This proves our claim. ■

Since we are interested in *upper* bounding $\mathbb{E}[I_\infty]$, it is sufficient to bound $\mathbb{E}[I_\infty(U, F^g)]$, which is done in the next lemma. We use large deviation properties of the sub-critical Galton-Watson process. For the convenience of the reader we provide this estimate in Appendix E.

Lemma 32 (Birth Death Process): Let the initial state be $U = (0, S_0, 0, 0)$. Fix a strictly positive δ, $0 < \delta < \frac{1}{2(\mathtt{r}-1)}$, so that $\frac{1-\delta}{2\delta} \in \mathbb{N}$ and let $\gamma = 1 - \frac{1+\delta}{\mathtt{r}}$. For all $\epsilon < \frac{1}{2(\mathtt{r}-1)}$ there exist constants $c = c(\mathtt{l}, \mathtt{r}, \epsilon, \delta)$, $c > 1$, and $c' = c'(\mathtt{l}, \mathtt{r}, \epsilon, \delta) >$

0 so that

$$\mathbb{P}\{I_\infty(U,F^g)\geq cS_0\}\leq e^{-c'S_0}.$$

Proof: Since condition (23) is satisfied in the beginning, the greedy R-process starts with some boundary steps. We claim that after exactly $\lfloor\frac{S_0}{1-\delta}\rfloor$ such boundary steps the condition (23) is for the first time no longer fulfilled. To see this, ignore the integer constraint for a moment. At the beginning of the process the condition (23) reads $0\leq S_0-(1-\delta)$. After $\frac{S_0}{1-\delta}$ boundary steps this condition is transformed to

$$\gamma\mathtt{r}\frac{S_0}{1-\delta}\leq S_0+\frac{S_0}{1-\delta}(\mathtt{r}-2)-(1-\delta),$$

which is equivalent to $0\leq-(1-\delta)$. We see that the inequality is no longer fulfilled and it is easy to check that this is the first time that it is no longer fulfilled.

After the initial boundary steps, the greedy strategy performs regular steps until exactly $\frac{1-\delta}{2\delta}$ regular extend steps are performed and then follows it by exactly one boundary step. This sequence is then repeated. (Note that by our assumption $\frac{1-\delta}{2\delta}\in\mathbb{N}$.)

To see this, note that each regular extend step increases the right-hand side of (22) by $2(\mathtt{r}-1)$ and the left-hand side by $2(\mathtt{r}-1-\delta)$. Further, each boundary step increases the left-hand side by $\mathtt{r}-1-\delta$ and the right-hand side by $\mathtt{r}-2$. Since $\frac{1-\delta}{2\delta}2(\mathtt{r}-1-\delta)+(\mathtt{r}-1-\delta)=\frac{1-\delta}{2\delta}2(\mathtt{r}-1)+(\mathtt{r}-2)$, we see that after one such sequence of first $\frac{1-\delta}{2\delta}$ regular extends steps followed by a boundary step the inequality is unchanged (up to an added constant). (A regular prune step does not change the condition (22).)

Since the randomness is contained only in the regular steps, we can model the process as consisting of only regular steps. To include the effect of boundary steps, we alter the outcome of the regular extend step as follows. From Table III note that for each regular extend step we increase S by $2\mathtt{r}-3$ and I by 1. We include the effect of boundary step by changing this to an increment of $2\mathtt{r}-3+(\mathtt{r}-2)\frac{2\delta}{1-\delta}$ for S and $1+\frac{2\delta}{1-\delta}$ for I, respectively.

Now this process is a standard birth and death process. Recall that we have $\epsilon<\frac{1}{2(\mathtt{r}-1)}$ and $\delta<\frac{1}{2(\mathtt{r}-1)}$. Hence, the expected increase in S at each step is $\epsilon(2\mathtt{r}-3+(\mathtt{r}-2)\frac{2\delta}{1-\delta})$. This is strictly less than 1. As discussed in more detail in Appendix E, this shows that, except for an exponentially small probability, this process stops for $t\leq cS_0$ for some appropriate constant $c>1$. This proves our lemma since in each step we create at most $1+\frac{2\delta}{1-\delta}$ internal variables. ∎

Using Lemma 32 we bound the number of variables marked by the marking process as follows.

Lemma 33 (Upper Bound): Let $\gamma=1-\frac{1+\delta}{\mathtt{r}}$ for some $0<\delta<\frac{1}{2(\mathtt{r}-1)}$. Fix $\mathtt{G}$ and $\mathcal{W}$ such that $\mathcal{W}\subseteq\mathtt{G}$ and $\mathtt{G}\in\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)$. Let $c=c(\mathtt{l},\mathtt{r},\epsilon,\delta)$ be the constant appearing in Lemma 32. If $|\mathcal{W}|\leq\frac{1}{6c(\mathtt{r}-1)\mathtt{r}}\alpha n$ then

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]\leq\alpha\frac{\mathtt{l}}{\mathtt{r}}.$$

Proof: Let $m=\frac{1}{\mathtt{r}}n$. The maximum number of surviving edges coming out of the witness $\mathcal{W}$ is $3(\mathtt{r}-1)|\mathcal{W}|$. Let this be S_0. Consider the R-process with initial state $U=(0,S_0,0,0)$ and the greedy strategy F^g. From Lemma 32 there exists a strictly positive constant c' such that

$$\mathbb{P}\{I_\infty(U,F^g)\geq cS_0\}\leq e^{-c'S_0}.$$

The bound on $|\mathcal{W}|$ in the hypothesis implies that $cS_0=c3(\mathtt{r}-1)|\mathcal{W}|\leq\frac{\alpha}{2}m$. From Table III we see that any time the number of internal variable nodes is increased by 1 the number of check nodes increases by at most 2. Therefore, $I_\infty(U,F^g)\leq cS_0$ implies that $C_\infty\leq 2cS_0\leq\alpha m$. This shows that the expansion property is satisfied for the whole duration of the process. Hence, $I_\infty(U,F^g)$ is a valid upper bound for $M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})$.

Let $M(\mathtt{E}')$ denote $M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})$. Since $M(\mathtt{E}')$ counts the initial $|\mathcal{W}|<\frac{\alpha}{2}m$ variables present in $\mathcal{W}$ along with the internal variables created,

$$\mathbb{P}\{M(\mathtt{E}')\geq\alpha m\}\leq\mathbb{P}\{I_\infty(U,F^g)\geq\frac{\alpha}{2}m\}\leq e^{-c'S_0}.$$

Therefore,

$$\begin{aligned}&\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]\\&\quad\leq\mathbb{P}\{M(\mathtt{E}')\leq\alpha m\}\alpha m+\mathbb{P}\{M(\mathtt{E}')\geq\alpha m\}n\\&\quad\leq\alpha\frac{\mathtt{l}}{\mathtt{r}}n+(1-e^{-\frac{c'\alpha 1n}{2\mathtt{r}}})n.\end{aligned}$$

The lemma is proved by taking the limit $n\to\infty$. ∎

G. Putting It All Together

In this section we prove Lemma 22 using the results developed in the previous sections.

Proof of Lemma 22. Recall that we consider an $(\mathtt{l}=3,\mathtt{r})$-regular ensemble and that $0\leq\epsilon<\epsilon^{\text{LGalB}}$.

Fix $0<\delta<\frac{1}{2(\mathtt{r}-1)}$ and define $\gamma=1-\frac{1+\delta}{\mathtt{r}}$. Let $\alpha_{\text{max}}(\gamma)$ be the constant defined in Theorem 2. Note that $\alpha_{\text{max}}(\gamma)$ is strictly positive since δ is strictly positive.

Choose $0<\alpha<\alpha_{\text{max}}(\gamma)$. Let $\mathcal{X}(\mathtt{l},\mathtt{r},\alpha,\gamma)$ denote the set of graphs $\{\mathtt{G}\in\text{LDPC}(n,\mathtt{l},\mathtt{r}):\mathtt{G}\in(\mathtt{l},\mathtt{r},\alpha,\gamma)\text{ right expander}\}$. From Theorem 2 we know that

$$\mathbb{P}\{\mathtt{G}\notin\mathcal{X}\}=o_n(1).\tag{25}$$

Let $c=c(\mathtt{l},\mathtt{r},\epsilon,\delta)$ be the coefficient appearing in Lemma 32 and define $\theta=\frac{1}{6c(\mathtt{r}-1)\mathtt{r}}\alpha$. From Lemma 26 we know that there exists an iteration ℓ such that

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|]\leq\frac{1}{2}\theta^2.\tag{26}$$

Let $n(\theta)$ be such that for $n\geq n(\theta)$, $\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|]\leq\theta^2 n$.

Using Lemma 28, and splitting the expectation over $\mathcal{X}$ and its complement, we get

$$\begin{aligned}&\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]\\&\leq\sum_{\mathcal{W}:|\mathcal{W}|\leq\theta n}\sum_{\mathtt{G}:\mathtt{G}\in\mathcal{X}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]+\\&\quad\sum_{\mathcal{W}:|\mathcal{W}|\leq\theta n}\sum_{\mathtt{G}:\mathtt{G}\notin\mathcal{X}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]+\\&\quad\theta n.\end{aligned}$$

Consider the first term. From Lemma 33 we know that

$$\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})] \le \alpha\frac{\mathtt{l}}{\mathtt{r}}n + o(n). \tag{27}$$

Consider the second term. Bound the expectation by n and remove the restriction on the size of the witness. This gives the bound

$$\sum_{\mathcal{W}}\sum_{\mathtt{G}:\mathtt{G}\notin\mathcal{X}} \mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}n.$$

Switch the two summations and use the fact that, for a given $\mathtt{G}$, each $\mathtt{E}$ realization maps to only one $\mathcal{W}$. We get

$$\begin{aligned}\sum_{\mathtt{G}:\mathtt{G}\notin\mathcal{X}} \mathbb{P}\{\mathtt{G}\}\sum_{\mathcal{W}:\mathcal{W}\subseteq\mathtt{G}} \mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\} &= \sum_{\mathtt{G}:\mathtt{G}\notin\mathcal{X}} \mathbb{P}\{\mathtt{G}\} = \mathbb{P}\{\mathtt{G}\notin\mathcal{X}\}\\ &\overset{(25)}{=} o_n(1).\end{aligned} \tag{28}$$

From (27) and (28) we conclude that for $n \ge n(\theta)$,

$$\begin{aligned}&\frac{1}{n}\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]\\ &\le \sum_{\mathcal{W}:|\mathcal{W}|\le\theta n}\sum_{\mathtt{G}:\mathtt{G}\in\mathcal{X}} \mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\Big(\alpha\frac{\mathtt{l}}{\mathtt{r}} + o_n(1)\Big)\\ &\quad + \frac{\mathtt{l}}{6c(\mathtt{r}-1)\mathtt{r}}\alpha\\ &\le \left(\frac{\mathtt{l}}{\mathtt{r}} + \frac{\mathtt{l}}{6c(\mathtt{r}-1)\mathtt{r}}\right)\alpha + o_n(1).\end{aligned}$$

If we now let n tend to infinity then we get

$$\begin{aligned}\lim_{n\to\infty}\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{LGalB}}(\mathtt{G},\epsilon,\ell)] &\le \lim_{n\to\infty}\frac{1}{n}\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]\\ &\le \left(\frac{\mathtt{l}}{\mathtt{r}} + \frac{\mathtt{l}}{6c(\mathtt{r}-1)\mathtt{r}}\right)\alpha.\end{aligned}$$

Since this conclusion is valid for any $0 < \alpha \le \alpha_{\text{max}}(\gamma)$ it follows that

$$\lim_{n\to\infty}\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{LGalB}}(\mathtt{G},\epsilon,\ell)] = 0. \qquad \blacksquare$$

H. Extensions

1) GalB and $\mathtt{l} \ge 4$: Note that for $\mathtt{l} \ge 5$ the result is already implied by Theorem **??**. For $\mathtt{l} = 4$ the proof is easily adapted from the one for $\mathtt{l} = 3$. The only difference lies in the way the size of the witness is computed (Section III-D) and the analysis of the birth-death process (Section III-F).

2) MS and BSC: The proofs can also be extended to other decoders. For a given MP decoder, the idea is to define an appropriate *linearized* version of the decoder (LMP) and go through the whole machinery as done for GalB.

For example, consider the MS(M) decoder and transmission over BSC(ϵ). The channel realizations are mapped to $\{\pm 1\}$. Let $M \in \mathbb{N}$, the message alphabet is $\mathcal{M} = \{-M,\dots,M\}$. For transmission of the all-one codeword, the *linearized* version of the decoder (LMS(M)) is defined as in Definition 24: i.e., at the check node the outgoing message is the minimum of the incoming messages and the variable node rule is unchanged.

One can check that the LMS algorithm defined above is monotonic with respect to the input log-likelihoods at both the variable and check nodes and the number of errors in the MS decoder can be upper bounded by the errors of the LMS decoder.

Lemma 34 (MS(M) Decoder, BSC and $\mathtt{l} \ge 3$): Consider $(\mathtt{l},\mathtt{r})$ ensemble and transmission over BSC(ϵ). Let ϵ^{LMS} be the channel parameter below which $p^{(\infty)}_{\{M\}} = 1$. If $\epsilon < \epsilon^{\text{LMS}}$, then

$$\lim_{n\to\infty}\limsup_{\ell\to\infty}\mathbb{E}[P_b^{\text{MS}}(\mathtt{G},\epsilon,\ell)] = 0$$

Example 35 (LMS(2) and BSC): Consider communication using LDPC$(3,6)$ code over BSC(ϵ) and decoding using MS(2) algorithm. For this setup, the DE threshold is 0.063. The linearized decoder of this algorithm has $p^{(\infty)}_{\{2\}} = 1$ for $\epsilon < 0.031$. Therefore from the Lemma 34 the limits can be exchanged for this ϵ.

The proof follows by showing results similar to Lemma 26 and 33. Here we give a brief explanation for adapting the proof to the case of $M = 2$ and $\mathtt{l} = 3$. For a given $p > 0$, we first perform $\ell(p)$ iterations such that $p^{(\ell)}_{\{-M,\dots,M-1\}} \le p$. We start the marking process from all the edges with messages in $\{-M,\dots,M-1\}$ and their witness. In this case the witness consists of edges which send messages $\{-M,\dots,M-1\}$.

To show that the size of the witness is going to zero, consider the DE equations similar to those in Appendix B. Let $p^{\mu}_{\ell}(x)$ denote a polynomial with non-negative coefficients where the coefficient in front of x^i denotes the probability that the message emitted by a variable node at iteration ℓ is μ and that the witness (of depth ℓ) for this edge has size i. Let $q^{\mu}_{\ell}(x)$ denote the equivalent quantity for messages emitted at check nodes. Then the DE equations for this augmented system are given by:

$$\begin{aligned}
p_1^{-1}(x) &= \epsilon x, \quad p_1^{+1}(x) = \bar{\epsilon}x,\\
p_\ell^{+1}(x) &= \epsilon x((q_{\ell-1}^{+1}(x))^2 + 2q_{\ell-1}^{+2}(1)q_{\ell-1}^{0}(x)) +\\
&\qquad \bar{\epsilon}x(2q_{\ell-1}^{+2}(1)q_{\ell-1}^{-2}(x) + 2q_{\ell-1}^{+1}(x)q_{\ell-1}^{-1}(x) + (q_{\ell-1}^{0}(x))^2),\\
p_\ell^{0}(x) &= \epsilon x(2q_{\ell-1}^{+2}(1)q_{\ell-1}^{-1}(x) + 2q_{\ell-1}^{+1}(x)q_{\ell-1}^{0}) +\\
&\qquad \bar{\epsilon}x(2q_{\ell-1}^{+1}(x)q_{\ell-1}^{-2}(x) + 2q_{\ell-1}^{0}(x)q_{\ell-1}^{-1}(x)),\\
p_\ell^{-1}(x) &= \bar{\epsilon}x((q_{\ell-1}^{-1}(x))^2 + 2q_{\ell-1}^{-2}(x)q_{\ell-1}^{0}(x)) +\\
&\qquad \epsilon x(2q_{\ell-1}^{+2}(1)q_{\ell-1}^{-2}(x) + 2q_{\ell-1}^{+1}(x)q_{\ell-1}^{-1}(x) + (q_{\ell-1}^{0}(x))^2),\\
p_\ell^{-2}(x) &= \epsilon x 2(q_{\ell-1}^{-2}(x)(q_{\ell-1}^{+1}(x) + q_{\ell-1}^{0}(x) + q_{\ell-1}^{-1}(x)))\\
&\qquad + \epsilon x(2q_{\ell-1}^{0}(x)q_{\ell-1}^{-1}(x) + (q_{\ell-1}^{-1}(x))^2(q_{\ell-1}^{-2}(x))^2)\\
&\qquad + \bar{\epsilon}x(2q_{\ell-1}^{-1}(x)q_{\ell-1}^{-2}(x) + (q_{\ell-1}^{-2}(x))^2),\\
q_\ell^{\mu}(x) &= \frac{p_{\ell-1}^{\mu}(x)}{p_{\ell-1}^{\mu}(1)}((1 - \sum_{i=-M}^{\mu-1} p_{\ell-1}^{i})^{\mathtt{r}-1} - (1 - \sum_{i=-M}^{\mu} p_{\ell-1}^{i})^{\mathtt{r}-1})
\end{aligned}$$

Using the hypothesis $p^{(\infty)}_{\{M\}} = 1$ and doing a similar analysis as in Appendix B we can show that the size of the witness behaves as $o_\ell(1)$. In the corresponding birth-death process we have to keep track of the size of the set of edges with messages in $\{-M,\dots,M-1\}$.

Similar results can be obtained for BP(M) decoder, and channels with continuous outputs. But the analysis of these decoders is more complicated because we have to deal with densities of messages.

3) MS(M) and continuous channel: Consider transmission through BMS channels with bounded output log-likelihoods and decoding using MS(M) decoder. For this setup it is

tempting to conjecture that the proofs can be extended using FKG inequalities for continuous lattices [6].

IV. Conclusion

We have shown two approaches for solving the problem of limit exchange below the DE threshold. The first one, based solely on the expansion property of the graph, helps in proving the result for a large class of MP decoders but only if the degree is relatively large. To prove the result for smaller degrees one has to include the role of channel realizations. The second approach accomplishes this in some cases. In this paper we only considered channel parameters below the DE threshold. But the regime above this threshold is equally interesting. One important application of proving the exchange of limits in this regime is the finite-length analysis via a scaling approach [7] since the computation of the scaling parameters heavily depends on the fact that this exchange is permissible.

Acknowledgment

We would like to thank A. Montanari for suggesting to directly apply the FKG inequalities in the proof of Lemma 27 instead of the original more elaborate construction. The work presented in this paper is partially supported by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 5005-67322.

Appendix

A. Expansion Argument For Block Error Probability

The following theorem is a modified version of a theorem by Burshtein and Miller [5].

Theorem 36 (Expansion): Consider an $(\mathtt{l},\mathtt{r},\alpha,\gamma)$ left expander. Assume that $0 \leq \beta \leq 1$ such that $\beta(l-1) \in \mathbb{N}$ and that $\beta\frac{\mathtt{l}-1}{\mathtt{l}} \leq 2\gamma - 1$. If at some iteration ℓ the number of bad variable nodes is less than $\frac{\alpha}{\mathtt{lr}}n$ then the MP algorithm will decode successfully.

Proof: Let $\mathcal{B}_\ell$ denote the bad set in iteration ℓ. We claim that

$$\begin{aligned}\gamma l|\mathcal{B}_\ell \cup \mathcal{B}_{\ell+1}| &\overset{(i)}{\leq} |\mathcal{N}(\mathcal{B}_\ell \cup \mathcal{B}_{\ell+1})| \\ &\overset{(ii)}{\leq} |\mathcal{N}(\mathcal{B}_\ell)| + \beta(l-1)|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell|. \qquad (29)\end{aligned}$$

Step (ii) follows from the fact that each variable in $\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell$ must be connected to at least $l-\beta(l-1)$ checks in the set $\mathcal{N}(\mathcal{B}_\ell)$ since otherwise this variable will be good and wont be in $\mathcal{B}_{\ell+1}$. Therefore the number of edges coming out of $\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell$ that are not connecting to $\mathcal{N}(\mathcal{B}_\ell)$ is at most $\beta(l-1)|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell|$. Thus the number of neighbors of $\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell$ that are not already neighbors of $\mathcal{B}_\ell$ is at most $\beta(l-1)|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell|$.

Consider now step (i). This step follows in a straightforward fashion from the expansion property since by assumption $|\mathcal{B}_\ell| \leq \frac{\alpha}{\mathtt{lr}}n$ so that $|\mathcal{B}_\ell \cup \mathcal{B}_{\ell+1}| < \alpha n$.

Let T be the set of check nodes that are connected to $\mathcal{B}_\ell \cap \mathcal{B}_{\ell+1}$ but not connected to $\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}$. Suppose an edge from a check node in T is carrying a bad message. Then this check must be connected to one more variable in $\mathcal{B}_\ell \cap \mathcal{B}_{\ell+1}$ because it is not connected to $\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}$ and thus cannot get a bad message from $\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}$. For each variable in $\mathcal{B}_\ell \cap \mathcal{B}_{\ell+1}$, at least $l-\beta(l-1)$ edges must be bad messages and hence it can connect to at most $(l-\beta(l-1))/2+\beta(l-1) = l/2+\beta(l-1)/2$ check nodes. Therefore we have,

$$\begin{aligned}|\mathcal{N}(\mathcal{B}_\ell)| &\leq l|\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| + |T|, \\ |\mathcal{N}(\mathcal{B}_\ell)| &\leq l|\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| + \frac{1+\beta\frac{l-1}{l}}{2}l|\mathcal{B}_\ell \cap \mathcal{B}_{\ell+1}|. \qquad (30)\end{aligned}$$

Using equations (29) and (30), we get

$$\begin{aligned}\gamma l|\mathcal{B}_{\ell+1} \cup \mathcal{B}_\ell| \leq& l|\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| + \frac{1+\beta\frac{l-1}{l}}{2}l|\mathcal{B}_{\ell+1} \cap \mathcal{B}_\ell| \\ &+ \beta(l-1)|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell| \\ \gamma|\mathcal{B}_{\ell+1} \cap \mathcal{B}_\ell| + \gamma|\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| &+ \gamma|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell| \\ \leq& |\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| + \frac{1+\beta\frac{l-1}{l}}{2}|\mathcal{B}_{\ell+1} \cap \mathcal{B}_\ell| + \\ &\beta\frac{l-1}{l}|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell| \\ |\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell| \leq& \frac{(1-\gamma)}{\gamma-\beta\frac{l-1}{l}}|\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}| + \\ &\frac{1+\beta\frac{l-1}{l}-2\gamma}{2(\gamma-\beta\frac{l-1}{l})}|\mathcal{B}_\ell \cap \mathcal{B}_{\ell+1}|\end{aligned}$$

The coefficient of the first term in RHS is less than 1 and the coefficient of the second term is negative and hence $|\mathcal{B}_{\ell+1}\backslash\mathcal{B}_\ell| < |\mathcal{B}_\ell\backslash\mathcal{B}_{\ell+1}|$ ∎

B. Size of Witness

Proof of Lemma 26. Let $\mathtt{G}$ be a graph and let $\mathtt{E}$ be the noise realization. Assume that we perform ℓ iterations. Let $\mathcal{W}_{\mathtt{e}}(\mathtt{G},\mathtt{E},\ell)$ denote the witness of edge $\mathtt{e}$. Then

$$\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|] \leq \sum_{i=1}^{\mathtt{l}n}\mathbb{E}[|\mathcal{W}_{\mathtt{e}_i}(\mathtt{G},\mathtt{E},\ell)|] = n\mathtt{l}\mathbb{E}[|\mathcal{W}_{\mathtt{e}_1}(\mathtt{G},\mathtt{E},\ell)|].$$

It remains to compute the expected size of the witness for the limit of n tending to infinity and a fixed ℓ. This can be accomplished by DE.

Let x_ℓ denote the probability of an edge being in error according to DE. Let $p_\ell(x)$ denote a polynomial with non-negative coefficients where the coefficient in front of x^i denotes the probability that the message emitted by a variable node at iteration ℓ is bad and that the witness (of depth ℓ) for this edge has size i (i variable nodes). Let $q_\ell(x)$ denote the equivalent quantity for messages emitted at check nodes. The DE equations for this augmented system are:

$$\begin{aligned}p_1(x) &= \epsilon x, \\ p_\ell(x) &= \epsilon(2-q_\ell(1))q_\ell(x)x + \bar{\epsilon}q_\ell(x)^2 x, \\ q_\ell(x) &= \frac{p_{\ell-1}(x)}{p_{\ell-1}(1)}(1-(1-p_{\ell-1}(1))^{\mathtt{r}-1}).\end{aligned}$$

The initialization $p_1(x) = \epsilon x$ reflects the fact that with probability ϵ a variable-to-check message is in error in iteration 1 and that its associated witness of depth 1 consists only of the attached variable (hence the x).

The recursion for $q_\ell(x)$ is also straightforward. With probability $1-(1-p_{\ell-1}(1))^{\mathtt{r}-1}$ at least one of the $\mathtt{r}-1$ incoming messages at a check node is bad, and in this case the distribution of the size of the attached witness is $\frac{p_{\ell-1}(x)}{p_{\ell-1}(1)}$.

Let us now look at the recursion for $p_\ell(x)$. There are three contributions: (i) Suppose that the variable has a bad received value and that exactly one of the incoming edges is bad; this happens with probability $\epsilon 2(1-q_\ell(1))q_\ell(1)$ and in this case the distribution of the size of the witness attached to this edge is $\frac{q_\ell(x)x}{q_\ell(1)}$, where the extra x accounts for the attached variable node. (ii) Suppose that the variable has a bad received value and that both incoming edges are bad; this happens with probability $\epsilon q_\ell(1)^2$, and in this case the distribution of the size of the witness attached to this edge is $\frac{q_\ell(x)x}{q_\ell(1)}$. (iii) Finally, suppose that the variable has a good received value and that both the incoming edges are bad; this happens with probability $\bar{\epsilon}q_\ell(1)^2$ and in this case the distribution of the size of the witness attached to this edge is $\frac{q_\ell(x)^2x}{q_\ell(1)^2}$.

Note that we get standard DE by setting $x=1$, i.e., we have $x_\ell = p_\ell(1)$. We want to show that $p'_\ell(1)$ (this is the expected size of the witness in the limit of infinite blocklengths) converges to zero as a function of ℓ.

The augmented DE equation is difficult to handle. So let us first write down a scalar version that tracks the expected value. Define $\beta_\ell = \frac{(1-(1-p_\ell(1))^{\mathtt{r}-1})}{p_\ell(1)}$. Then we get

$$p_\ell(x) = \epsilon(2-q_\ell(1))\beta_{\ell-1}p_{\ell-1}(x)x + \bar{\epsilon}\beta_{\ell-1}^2 p_{\ell-1}(x)^2 x.$$

Differentiate both sides with respect to x. This gives

$$\begin{aligned}p'_\ell(x) =&\epsilon\beta_{\ell-1}(2-q_\ell(1))(p'_{\ell-1}(x)x + p_{\ell-1}(x))\\ &+\bar{\epsilon}\beta_{\ell-1}^2(p_{\ell-1}(x))^2 + \bar{\epsilon}\beta_{\ell-1}^2 2p_{\ell-1}(x)p'_{\ell-1}(x)x.\end{aligned}$$

Now substitute $x=1$. Recall that $x_\ell = p_\ell(1)$ and define $p_\ell = p'_\ell(1)$. Further, bound $2-q_\ell(1)$ by 2 and β_ℓ by $(\mathtt{r}-1)$. This gives the inequality

$$\begin{aligned}p_\ell \le& 2\epsilon(\mathtt{r}-1)p_{\ell-1} + 2\epsilon(\mathtt{r}-1)x_{\ell-1}\\ &+\bar{\epsilon}(\mathtt{r}-1)^2x_{\ell-1}^2 + 2\bar{\epsilon}(\mathtt{r}-1)^2x_{\ell-1}p_{\ell-1}.\end{aligned}$$

We claim that $\ell x_\ell \le p_\ell$. This is true since x_ℓ is the probability of a bad message, whereas p_ℓ is the expected size of the witness and the witness size is always at least ℓ if the message is bad. Therefore,

$$\begin{aligned}\frac{p_\ell}{p_{\ell-1}} \le& 2\epsilon(\mathtt{r}-1) + 2\epsilon(\mathtt{r}-1)\frac{x_{\ell-1}}{p_{\ell-1}}\\ &+\bar{\epsilon}(\mathtt{r}-1)^2\frac{x_{\ell-1}^2}{p_{\ell-1}} + 2\bar{\epsilon}(\mathtt{r}-1)^2x_{\ell-1}\\ \le& 2\epsilon(\mathtt{r}-1) + 2\epsilon\frac{(\mathtt{r}-1)}{\ell} + 3\bar{\epsilon}(\mathtt{r}-1)^2x_{\ell-1}.\end{aligned}$$

Now note that x_ℓ tends to zero since $\epsilon < \epsilon^{\text{LGalB}}$. Therefore, if $2\epsilon(\mathtt{r}-1)<1$ then $p_\ell/p_{\ell-1}<1$ for ℓ sufficiently large. The stability condition implies $\epsilon^{\text{LGalB}} < \frac{1}{2(\mathtt{r}-1)}$. Therefore, for $\epsilon < \epsilon^{\text{LGalB}}$, p_ℓ tends to zero exponentially fast for increasing ℓ. ■

C. Randomization

Proof of Lemma 28. We have

$$\begin{aligned}&\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]\\ &=\sum_{\mathcal{W}}\mathbb{E}[M(\mathtt{G},\mathtt{E},\mathcal{W})\mathbb{1}_{\{\mathcal{W}(\mathtt{G},\mathtt{E},\ell)=\mathcal{W}\}}]\\ &=\sum_{\mathcal{W},\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{E}_{\mathtt{E}}[M(\mathtt{G},\mathtt{E},\mathcal{W})\mathbb{1}_{\{\mathcal{W}(\mathtt{G},\mathtt{E},\ell)=\mathcal{W}\}}]\\ &=\sum_{\mathcal{W},\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{E}_{\mathtt{E}}[M(\mathtt{G},\mathtt{E},\mathcal{W})\mathbb{1}_{\{\mathtt{E}\in\mathcal{E}_{\mathtt{G},\mathcal{W}}\}}].\end{aligned}$$

For all $\mathtt{E}\in\mathcal{E}_{\mathtt{G},\mathcal{W}}$, the channel values on $\mathcal{W}$ are fixed to those appearing in the witness which is also denoted by $\mathcal{W}$. Recall that $\mathcal{E}'_{\mathtt{G},\mathcal{W}}$ is the projection of $\mathcal{E}_{\mathtt{G},\mathcal{W}}$ on $\mathtt{G}\backslash\mathcal{W}$ and $\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}$. The above expectation is equivalent to

$$\begin{aligned}&\mathbb{E}_{\mathtt{E}}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})\mathbb{1}_{\{(\mathcal{W},\mathtt{E}')\in\mathcal{E}_{\mathtt{G},\mathcal{W}}\}}] =\\ &\quad\mathbb{P}(\mathcal{W})\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}],\end{aligned}$$

where $\mathbb{P}(\mathcal{W})$ is the probability of the channel values on $\mathcal{W}$. This implies $\mathbb{P}(\mathcal{W})\mathbb{P}(\mathcal{E}'_{\mathtt{G},\mathcal{W}}) = \mathbb{P}(\mathcal{E}_{\mathtt{G},\mathcal{W}})$. Using (20) we bound

$$\begin{aligned}&\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})\mathbb{1}_{\{\mathtt{E}'\in\mathcal{E}'_{\mathtt{G},\mathcal{W}}\}}]\\ &\quad\le\mathbb{P}(\mathcal{E}'_{\mathtt{G},\mathcal{W}})\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})].\end{aligned}$$

Therefore,

$$\begin{aligned}&\mathbb{E}[M(\mathtt{G},\mathtt{E},\ell)]\\ &\le\sum_{\mathcal{W},\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]\\ &\le\sum_{\mathcal{W}:|\mathcal{W}|\le\theta n,\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})]+\\ &\quad\sum_{\mathcal{W}:|\mathcal{W}|\ge\theta n,\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}\mathbb{E}_{\mathtt{E}'}[M(\mathtt{G},(\mathcal{W},\mathtt{E}'),\mathcal{W})].\end{aligned}$$

Consider the second term in the last line. Bound the expectation by n. This yields

$$\sum_{\mathcal{W}:|\mathcal{W}|\ge\theta n,\mathtt{G}}\mathbb{P}\{\mathtt{G}\}\mathbb{P}\{\mathcal{E}_{\mathtt{G},\mathcal{W}}\}n.$$

If $\mathcal{W}\not\subseteq\mathtt{G}$, then $\mathcal{E}_{\mathtt{G},\mathcal{W}}$ is empty. Therefore the above bound is equivalent to

$$\begin{aligned}&n\sum_{\mathcal{W}:|\mathcal{W}|\ge\theta n}\mathbb{E}[\mathbb{1}_{\{\mathcal{W}\subseteq\mathtt{G}\}}\mathbb{1}_{\{\mathtt{E}\in\mathcal{E}_{\mathtt{G},\mathcal{W}}\}}]\\ &=n\sum_{\mathcal{W}:|\mathcal{W}|\ge\theta n}\mathbb{E}[\mathbb{1}_{\{\mathcal{W}(\mathtt{G},\mathtt{E},\ell)=\mathcal{W}\}}]\\ &=n\mathbb{P}\{|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|\ge\theta n\}.\end{aligned}$$

By assumption, $\mathbb{E}[|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|]\le\theta^2 n$. The Markov inequality therefore shows that

$$\mathbb{P}\{|\mathcal{W}(\mathtt{G},\mathtt{E},\ell)|\ge\theta n\}\le\theta.$$

■

D. FKG Inequality

Consider the Hamming space $\{0,1\}^n$. For $x, y \in \{0,1\}^n$ define the following partial order: $x \le y$ iff $x_i \le y_i$ for all i. Define $x_{\le}$ as

$$x_{\le} = \{y : y \in \{0,1\}^n,\ y \le x\}, \tag{31}$$

and $x \vee y$ and $x \wedge y$ as

$$(x \vee y)_i = \begin{cases} 0 & \text{if } x_i = y_i = 0, \\ 1 & \text{else,} \end{cases} \tag{32}$$

$$(x \wedge y)_i = \begin{cases} 1 & \text{if } x_i = y_i = 1, \\ 0 & \text{else.} \end{cases} \tag{33}$$

We say that a function $f : \{0,1\}^n \to \mathbb{R}$ is monotonically increasing (decreasing) if $f(x) \ge f(y)$ whenever $x \ge y$ ($x \le y$).

Lemma 37 (FKG Inequality – [8]): Let $P\{\cdot\}$ be a probability measure on $\{0,1\}^n$ such that

$$P\{x\}P\{y\} \le P\{x \vee y\}P\{x \wedge y\}.$$

Let f and g be real-valued non-negative functions on $\{0,1\}^n$. If f and g are either both monotonically increasing or both decreasing then

$$\mathbb{E}[f(x)g(y)] \ge \mathbb{E}[f(x)]\mathbb{E}[g(y)].$$

E. Birth and Death Process

Consider the following birth and death process. We start with $X_0 = a > 0$. At step t, $t \in \mathbb{N}$, if $X_{t-1} < 1$ then we stop the process and define $X_{t'} = X_{t'-1}$ for $t' > t$. Otherwise we decrease X_{t-1} by 1 and add Y_t, where the sequence $\{Y_t\}_{t \ge 1}$ is iid. In this way, as long as $X_{t-1} \ge 1$,

$$X_t = X_{t-1} - 1 + Y_t.$$

This process is equivalent to the standard birth and death process if Y_t takes non-negative integer values. In this case, the step described above corresponds to choosing a member of the population which then creates Y_t off-springs and dies.

Let T denote the stopping time, i.e., $T = \min\{t : X_t < 1\}$.

Lemma 38 (Birth-Death): Fix $p \in (0,1]$ and $0 < \mu < 1$. Consider a birth and death process with $X_0 = a \in \mathbb{N}$ and

$$Y_i = \begin{cases} \frac{\mu}{p}, & \text{with probability } p, \\ 0, & \text{with probability } 1-p, \end{cases}$$

so that $\mathbb{E}[Y_i] = \mu$. Then, for $\beta a \in \mathbb{N}$,

$$\mathbb{P}\{T > \beta a\} \le e^{-ac(p,\mu,\beta)}$$

where $c(p,\mu,\beta) > 0$ for $\beta > \frac{1}{1-\mu}$.

Proof: Let $b = \beta a$. Note that

$$\mathbb{P}\{T > b\} \le \mathbb{P}\{X_b \ge 1\} \le \mathbb{P}\{X_b \ge 0\}.$$

Let $\tilde{Y}_t = Y_t - 1$. We have

$$X_b = X_{b-1} + \tilde{Y}_b = X_{b-2} + \tilde{Y}_{b-1} + \tilde{Y}_b = a + \sum_{i=1}^{b} \tilde{Y}_i.$$

Therefore,

$$\mathbb{P}\{T > b\} \le \mathbb{P}\Big\{\sum_{i=1}^{b} \tilde{Y}_i \ge -a\Big\} \overset{s \ge 0}{=} \mathbb{P}\big\{e^{s\sum_{i=1}^{b}\tilde{Y}_i} \ge e^{-as}\big\}$$
$$\overset{\text{Markov}}{\le} e^{as}\mathbb{E}[e^{s\tilde{Y}}]^b = e^{as}\Big((1-p)e^{-s} + pe^{(\frac{\mu}{p}-1)s}\Big)^b.$$

First consider the case $\mu \ge p$. Set $s = \frac{p}{\mu}\ln\frac{(\beta-1)(1-p)}{p+\beta(\mu-p)}$, which is strictly positive since $\mu \ge p$ and $\beta > \frac{1}{1-\mu}$. Set $\beta = \frac{1}{1-\mu-\xi}$, where $\xi > 0$. With this choice we get

$$\mathbb{P}\{T > b\} \le \Big[\frac{\mu(1-p)}{\mu(1-p)-\xi p}\Big(\frac{\mu(1-p)-\xi p}{\mu(1-p)+\xi(1-p)}\Big)^{\frac{p(\mu+\xi)}{\mu}}\Big]^b.$$

For $\xi = 0$ the terms inside the square brackets is 1. If we take the derivative of the expression inside the square brackets wrt to ξ we get

$$\frac{-p}{\mu+\xi}\Big(\frac{(\mu+\xi)(1-p)}{\mu(1-p)-p\xi}\Big)^{1-\frac{p(\mu+\xi)}{\mu}}\log\frac{(\mu+\xi)(1-p)}{\mu(1-p)-\xi p}.$$

For $\xi > 0$ and $\mu > p$ this is strictly negative which proves our claim.

Now consider the case $\mu < p$. For $\frac{1}{1-\mu} < \beta < \frac{p}{p-\mu}$ the above still applies. For $\beta \ge \frac{p}{p-\mu}$, the probability is 0. This is because in each step we can add at most $\frac{\mu}{p} - 1$. Therefore, for $t \ge \frac{p}{p-\mu}a + 1$, $X_t \le a + (\frac{p}{p-\mu}a + 1)(\frac{\mu}{p} - 1) < 0$. ∎

F. Concentration

Theorem 39 (Concentration Theorem [1][p. 222]): Let $\mathtt{G}$, chosen uniformly at random from LDPC(n,λ,ρ), be used for transmission over a BMS(ϵ) channel. Assume that the decoder performs ℓ rounds of message-passing decoding and let $P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell)$ denote the resulting bit error probability. Then, for any given $\delta > 0$, there exists an $\alpha > 0$, $\alpha = \alpha(\lambda,\rho,\delta)$, such that

$$\mathbb{P}\{|P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell) - \mathbb{E}_{\text{LDPC}(n,\lambda,\rho)}\left[P_b^{\text{MP}}(\mathtt{G},\epsilon,\ell)\right]| > \delta\} \le e^{-\alpha n}.$$

REFERENCES

[1] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[2] S.-Y. Chung, G. D. Forney, Jr., T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, Feb. 2001.

[3] R. G. Gallager, "Low-density parity-check codes," *IRE Transactions on Inform. Theory*, vol. 8, pp. 21–28, jan 1962.

[4] R. J. McEliece, E. Rodemich, and J.-F. Cheng, "The turbo decision algorithm," in *Proc. of the Allerton Conf. on Commun., Control, and Computing*, Monticello, IL, USA, 1995.

[5] D. Burshtein and G. Miller, "Expander graph arguments for message-passing algorithms," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 782–790, Feb. 2001.

[6] C. J. Preston, "A generalization of the FKG inequalities," *Commun. math. Phys.*, vol. 36, pp. 233–241, 1974.

[7] A. Amraoui, A. Montanari, T. Richardson, and R. Urbanke, "Finite-length scaling for iteratively decoded LDPC ensembles," in *Proc. of the Allerton Conf. on Commun., Control, and Computing*, Monticello, IL, USA, Oct. 2003.

[8] C. M. Fortuin, P. W. Kasteleyn, and J. Ginibre, "Correlation inequalities on some partially ordered sets," *Commun. math. Phys.*, vol. 22, pp. 89–103, 1971.