Density Evolution on a Class of Smeared Random Graphs

Kabir Chandrasekher, Orhan Ocal, and Kannan Ramchandran
Department of Electrical Engineering and Computer Sciences, University of California, Berkeley
{kabirc, ocal, kannanr}@berkeley.edu

***Abstract*—We introduce a new ensemble of random bipartite graphs, which we term the 'smearing ensemble', where each left node is connected to some number of consecutive right nodes. Such graphs arise naturally in recovering sparse wavelet coefficients when signal acquisition is in the Fourier domain, such as in magnetic resonance imaging (MRI). Graphs from this ensemble exhibit small, structured cycles with high probability, rendering current techniques for determining iterative decoding thresholds inapplicable. In this paper, we develop a theoretical platform to analyze and evaluate the power of smearing-based structure. Despite the existence of these small cycles, we derive exact density evolution recurrences for iterative decoding on graphs with smear-length two. Furthermore, we give lower bounds on the performance of a much larger class from the smearing ensemble, and provide numerical experiments showing tight agreement between empirical thresholds and those determined by our bounds. We additionally detail a system architecture to recover sparse wavelet representations in the MRI setting, giving explicit oversampling thresholds for the 1-stage Haar wavelet.**

I. Introduction

Consider a specific balls-and-bins game where you have n distinct colors, d balls of each color, and M bins. Only $K(\ll n)$ of the colors are 'active'. You need to throw the Kd active balls into the M bins according to the following rules:

R1) Each of the d balls of an 'active' color must go to a different bin[1].

R2) If a bin contains a single ball, then all the other $d-1$ balls of the same color can be removed.

R3) The process continues iteratively until either (a) all balls have been removed or (b) no further single-occupied bins exist.

The goal of the game is to remove all active balls using the minimum number of bins. We focus on the regime in which $(n, K, M) \to \infty$, $d = \mathcal{O}(1)$ and ask the following questions:

1) What is the optimal design policy of where to dispatch the d balls? (That is, what is the optimal bipartite graph which must be designed without knowledge of which K colors are active)
2) Given (n, K, d), what is the minimum number of bins (M) necessary[2]?

While this is an intriguing game in its own right, more importantly, it has connections to the design of Low Density Parity Check (LDPC) codes and peeling decoding. Surprisingly, and more relevant here, it is also intimately related to the recovery of sparse wavelet representations from Fourier domain samples (see Section III).

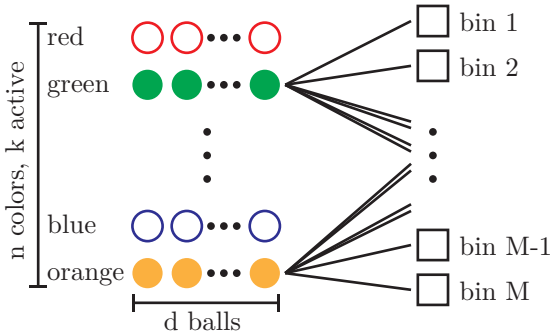

Fig. 1: We have n colors and d balls of each color. The balls are thrown into M bins, and k out of n colors become active (shown as filled circles). Using the rules R1-R3, we want to find the minimum number of bins necessary to recover all the active colors.

To illustrate, say that $d = 3$, then the best known strategy is to throw each ball at a bin selected uniformly at random. It has been demonstrated that we need asymptotically $M \simeq 1.222K$ bins as K grows. This can be shown through density evolution methods, introduced by Richardson and Urbanke in [1], which have proven powerful in analyzing the performance of LDPC codes. Now suppose that $d = 6$. The natural strategy is to again throw each ball at a bin selected uniformly at random. Surprisingly, this strategy is not optimal. To see this, we give a brief introduction to the smearing ensemble. Consider an i dimensional vector $s = (s_1, s_2, \ldots, s_i)$ where $\sum_{j=1}^{i} s_i = d$. This ensemble is such that i bins are selected at random, and for the jth bin, the immediately following $s_j - 1$ bins are deterministically selected[3]; this is what we term the smearing ensemble. We give a formal definition in section II. From this simple ensemble, one can see in Table 1, that many smearing strategies outperform the non-smeared strategy. In this paper, we do not claim to design an optimal strategy for this game; rather, we provide a theoretical platform to analyze and evaluate such policies exploiting structure.

Our balls-and-bins game is motivated by an extension of the recently proposed FFAST (Fast Fourier Aliasing-based Sparse Transform) algorithm [2] to the case where sparsity

[1]That is, each bin may contain only one ball of each color.

[2]Note the importance of parameter n; if the set of K colors were known a priori, it is possible to use a deterministic strategy with $K + d$ bins.

[3]For example, if $s_j = 2$ (we henceforth refer to this as the smear-length), then bins b_j and $b_j + 1$ are selected, where b_j is uniformly selected on $\{1, 2, \ldots, M\}$. See Fig. 1 for an illustration.

is with respect to some wavelet basis. In this setting, we consider the balls to be wavelet coefficients and the balls to be samples. Wavelets are universally recognized to be an extremely efficient sparse representation for the class of piecewise smooth signals having a relatively small number of discontinuities, a very good model for many real-world signals such as natural images [3]. In particular, we note that in MRI, images are observed to be sparse with respect to a wavelet basis, and acquisition is in the Fourier domain [4]. The computational bottleneck in recovering these images has been observed to be the multiple FFT's taken. Our extension of the FFAST algorithm attacks this problem head on, see Sec. III for details.

Table 1: Thresholds (M/K) for $d = 6$. Note that this table contains a strict subset of all possible strategies.

Regime	1,1,1,1,1,1	1,1,1,1,2	1,1,1,3	1,1,4	1,1,2,2	1,2,3	2,2,2
M/K	1.570	1.533	1.489	1.518	1.533	1.542	1.547

A. Related Works

Density evolution methods have proven powerful in analyzing the performance of LDPC codes and their extensions [1], [5]. Unfortunately, the well-known density evolution equations apply only for sparse random graphs that are locally tree-like. This is not the case for all ball-throwing strategies in the game we outlined above, e.g. the $[2, 2, 2]$ scheme. Recently Donoho et. al. have introduced approximate message passing (AMP) techniques to extend the message passing paradigm to the case when the underlying factor graph is dense [6]. These techniques were rigorously analyzed by Bayati and Montanari in [7]. Although AMP has been successfully applied to many problem domains, e.g., [8], [9], it imposes a dense structure on the factor graph. Additionally, Kudekar et. al have been able to show the benefit of structure in convolutional LDPC codes through the spatial coupling effect [10], [11]. However, if the bipartite graph is sparse, but contains small, structured cycles, it may not be necessary to invoke such methods.

B. Main Results and Organization of the Paper

The main results of this paper are the derivation of exact thresholds for random graphs with smear-length 2, and bounds for graphs with smear-length 2 [4], and bounds for higher smear-length which are empirically shown to be very tight. We additionally detail an application to fast recovery of sparse wavelet representations when acquisition is in the Fourier domain. In particular, given a signal with ambient dimension n which is K-sparse with respect to the 1-stage Haar wavelet, our analysis shows that $2.63K$ Fourier domain samples are needed to recover the signal in $\mathcal{O}(K \log K)$ time.

The organization of this paper is as follows. In Section II-A, we derive sharp thresholds for the specific case of 2-smearing. In Section II-B, we derive lower bounds for the probability of recovery in the case of arbitrary smearing. We outline the connection between the ball coloring game and the recovery of sparse wavelet representations in Section III. We conclude with Section IV by summarizing some interesting open problems and conjectures that have resulted from this work in.

[4]The derivations of smear-length 3 are given in the extended version [12].

II. Main Results

In this section, we introduce a new random graph ensemble, termed the 'smearing ensemble', and show how to derive density evolution recursions for graphs from this ensemble. We detail the derivation of density evolution for smearing with smear-length 2, and we give lower bounds for smear-length L.

We now formally define the smearing ensemble.

Definition 1. Let $\mathcal{G}(K, M, s)$ denote the 'smearing graph ensemble' with K left nodes and M right nodes. The vector $s = [s_1, s_2, \ldots, s_i]$ characterizes the connections between left and right nodes according to the following the algorithm:

```
for each left node:
    for j in [1, ..., i]:
        connect left node to a right node
        chosen uniformly at random and its
        s_j-1 following neighbors
```

Denote the ratio $\lambda := K/M$. Now, note that if K balls are thrown into the M bins randomly, the degree of each bin will be distributed like Poisson(λ). It is clear that under smearing, bins in stage i will have degree distributed like Poisson$(s_i\lambda)$, as the degree of each bin is the sum of s_i independent 'streams' of balls.

We begin by carefully deriving the density evolution recurrence for smearing ensembles with the maximum smearing with smear-length 2.[5] Finally, we give lower bounds for the general smearing case.

A. Density Evolution on 2-smear Graphs ($s = [2, 2, 2]$)

In our ball-coloring game, we noted that threshold for the setting $[2, 2, 2]$ outperforms that for the setting $[1, 1, 1, 1, 1, 1]$. In this section, we give exact analysis for these thresholds[6]. First, we define our notation in Table 2.

Table 2: Notation for density evolution with smear-length 2

x_t :	Probability that a random node is *not* removed at iteration t
q_t :	Probability that neither bin in a smeared pair is removed at iteration t
d_t :	Probability that *all* nodes in the same stream as the node of reference are removed at time t
s_t :	Probability that *all* nodes in a stream which intersects, but does not fully overlap, with the reference stream are removed at time t

Unlike conventional density evolution, note that we are taking a node perspective. Our goal is to derive recurrences

[5] We note that the full version of this work, explicit derivation of thresholds for smearing ensembles with smear-length 3 is given, but for the sake of brevity it is omitted here.

[6]Although thresholds can be derived for the other elements in the table, we give the $[2, 2, 2]$ derivation for clarity.

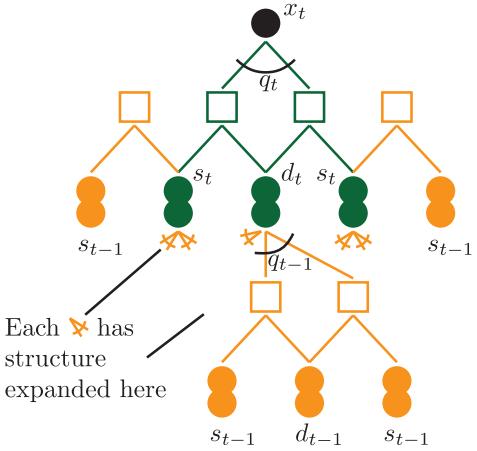

Fig. 2: Depth 2 neighborhood of a ball in a graph with $[2,2,2]$ smearing (see Definition 1). The bins and balls are grouped by their colors with respect to distance from the root.

for each of these quantities. In this case, the equations for x_t and q_t are clear:

$$x_t = q_t^3, \tag{1}$$
$$q_t = 1 - d_t(1-(1-s_t)^2). \tag{2}$$

In order to see this, note that for a node to be removed, at least one of the bins to which it is connected must be removed. Thus, x_t only occurs when none of the pairs of smeared bins to which it is connected are removed. On the other hand, for a bin to be removed, *all* of its connected nodes must be removed. Thus, all of the balls in the same stream as the reference ball must be removed, which happens with probability d_t. Additionally, at least one of the two streams that do not fully overlap with the reference ball must be removed, which happens with probability $(1-(1-s_t)^2)$. Now, note that in Fig. 2, the recurring structures are those with probabilities s_t, d_t, q_t. We first calculate d_t. Each of the balls in this stream act independently of each other, and each of the balls in this stream must be removed in another stage. Letting D_1 denote the number of balls in this stream, we find:

$$\begin{aligned} d_t &= \sum_{i=0}^{\infty} P(D_1 = i)(1-q_{t-1}^2)^i \\ &= \sum_{i=0}^{\infty} e^{-\lambda}\frac{\lambda^i}{i!}(1-q_{t-1}^2)^i = e^{-\lambda q_{t-1}^2}. \end{aligned} \tag{3}$$

Now, we tackle s_t. Note that intuitively, $s_t \geq d_t$. This is because each ball in a stream tracked by s_t gets the same independent help from other stages. However, there is additionally a shared bin between all these balls. This shared bin is able to aid in the removal of the stream tracked by s_t when exactly one ball is left in the stream, and the bin has no contributions from elsewhere. Letting D_2 be the number of balls in this stream, we precisely characterize this as follows:

$$\begin{aligned} s_t &= \sum_{i=0}^{\infty} P(D_2 = i)\left[(1-q_{t-1}^2)^i + iq_{t-1}^2(1-q_{t-2}^2)^{i-1}\right] \\ &= e^{-\lambda q_{t-1}^2} + \lambda s_{t-1} q_{t-1}^2 e^{-\lambda q_{t-2}^2}. \end{aligned} \tag{4}$$

Note that the first term in this recursion is exactly d_t. The second term describes the help received from the shared bin. In order to properly characterize this term, it is necessary to introduce the notion of memory: the shared bin can help if it has all contributions removed except for one by time t. Unlike when the neighborhood is tree-like and contains no cycles, branches of the tree become independent [13], when cycles are introduced, dependence between branches is introduced. The introduced memory captures exactly this dependence. This style of thinking leads to the principles of generalization listed in below.

Generalizing to L-smearing

1)	In a stage with L smearing, there will be $L-1$ steps of memory necessary in order to properly capture the smearing structure
2)	In a stage with L smearing, the number of recurring structures will be L
3)	Shared bins can be used through the introduction of memory in the recursion

In the following section, we will use these principles to derive simple, but effective, lower bounds for L-smearing[7].

We summarize the results of this section with the following lemma:

Lemma 1. *Consider a random graph from the ensemble $G(K, M, [2,2,2])$. Then, for $M \geq 1.547$, recovery using the peeling decoder will succeed with high probability.*

Proof. Follows from the density evolution derived above. □

In the extended version of this work [12], we derive explicit recursions for the ensembles with $s = [1,1,3]$ (smearing of order 3), but omit it here for the sake of clarity. Following the above principles of generalization, we now derive lower bounds on the density evolution for the general smearing ensemble.

B. Lower Bound on L-smearing

For clarity, we will consider the ensembles with $s = [L,L,L]$. Along with x_t, q_t, d_t as described in Table 2, we define quantity $s_t^{(i)}$ in Table 3.

Table 3: Notation for density evolution with smear-length L

$s_t^{(i)}$:	Probability that *all* nodes in the streams which *does not* intersect with the reference stream in j bins where $1 \leq j \leq i$ bins are removed at time t

As described in the principles of generalization, there will be L recursions, and up to $L-1$ memory. The probability d_t is unaffected as it depends only on the other stages. All of the $s_t^{(i)}$, however, can have up to i memory (corresponding to the number of bins that do not overlap with the reference ball). Thus, we take an approach much like a first-order approximation of a Taylor series, and allow each $s_t^{(i)}$ to use only one step of memory. The following lemma characterizes the critical quantity q_t in terms of its component streams.

[7]Although it is possible to derive exact recurrences in such a case, we pursue lower bounds for intellectual clarity.

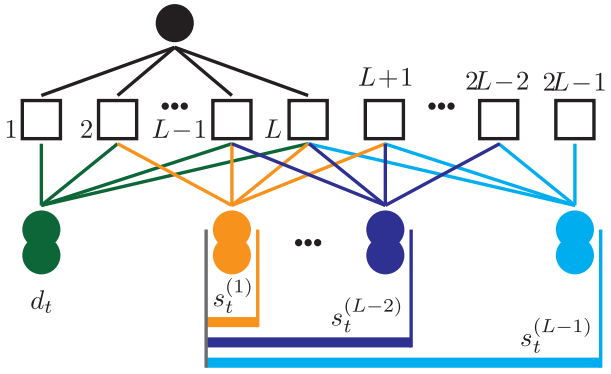

Fig. 3: Depth 1 neighborhood of a ball under L-smearing. The quantities d_t and $\{s_t^{(i)}\}_{i=1}^{L-1}$ are the recurrent structures, where $s_t^{(i)}$ tracks the joint probability that all the streams which intersect with the reference stream in $\{j\}_{j=L-i}^{L-1}$ bins are removed at time t.

Lemma 2. *In a stage with L-smearing,*

$$1-q_t = d_t\left[2s_t^{(L-1)} + \sum_{i=2}^{L-1} s_t^{(i-1)}s_t^{(L-i)} - \sum_{i=1}^{L-1} s_t^{(i)}s_t^{(L-i)}\right].$$

Proof. See the appendix in the extended version [12]. □

The following lemma then establishes monotonicity of $1-q_t$ with respect to $d_t, s_t^{(i)}$, which we may use to complete the bound.

Lemma 3. *Let $f(d_t, s_t^{(1)}, \dots s_t^{(L-1)}) = 1 - q_t$, then $f(d_t, s_t^{(1)}, \dots s_t^{(L-1)})$ is non-decreasing in $(d_t, s_t^{(1)}, \dots, s_t^{(L-1)})$.*

Proof. See the appendix in the extended version [12]. □

Given the above characterization of q_t, it now remains to give lower bounds for $d_t, s_t^{(1)}, \dots, s_t^{(L-1)}$. We can bound these as follows:

$$d_t = e^{-\lambda q_{t-1}^2}, \tag{5}$$

$$s_t^{(i)} \geq e^{i\lambda q_{t-1}^2} + \lambda q_{t-1}^2 e^{-\lambda q_{t-2}^2} r_{t-1}^{(i)}, \tag{6}$$

for $i \in \{1, \dots, L-1\}$, where

$$r_t^{(i)} \stackrel{\text{def}}{=} \sum_{j=1}^{i}\sum_{k=1}^{j} s_{t-1}^{(k)} e^{-(L-k-1)\lambda q_{t-2}^2} e^{-(k-1)\lambda q_{t-1}^2}. \tag{7}$$

There is a simple way to think about the problem so that these bounds appear. Consider the bound on $s_t^{(i)}$. This tracks the joint probability that all the streams which intersect the reference stream in $L-j$ bins where $1 \leq j \leq i$ are removed at time t. In order for all of these streams to be removed, there are two cases:

1) Each stream was removed from another stage. This probability is tracked by the first term: $e^{i\lambda q_{t-1}^2}$.
2) Exactly one ball remains among all the streams. This probability is tracked by the second term.

We focus on the second case. Suppose that the remaining ball is in the stream which intersects the reference stream in j bins. This implies that it is also contained in $L-j$ shared bins that do not intersect the reference stream. It can be removed by any of these bins, as long as it is the only contribution. This help is tracked by the summation in the second term.

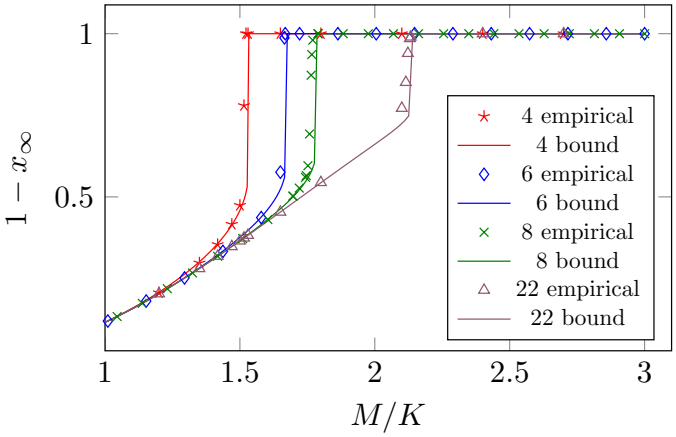

Fig. 4: We plot our lower bounds against empirical simulations with settings $s = [1, 1, L]$ for various L. The y-axis here denotes the probability that a random ball is removed when peeling stops (i.e., $1 - x_t$ as $t \to \infty$). We note that this ensemble is used as it is of the most relevant practical interest to recovery of sparse wavelet representations (see Section III).

Corollary 1. *The lower bounds given in equations* (5) *and* (6) *imply a lower bound on x_t, the probability that a random node is removed at time t.*

Proof. This follows directly from Lemma 2 and Lemma 3. □

We now give numerical experiments corroborating that the bounds in Corollary 1 capture the actual thresholds well. We experimentally find the thresholds for full recovery by sweeping λ, and compare them to the thresholds implied by the bounds. Fig. 4 shows these for filters with different lengths.

III. Connections to Recovery of Sparse Wavelet Representations

We now present how the game of balls and bins described in the introduction relates to MRI. In MRI, one acquires samples from the Fourier transform of an input signal, which is then inverted to recover the original signal. Mathematically, let x be an n length signal, and $X \stackrel{\text{def}}{=} F_n x$ be its Fourier transform, where F_n is the Fourier matrix of size $n \times n$. In MRI, the problem is to recover x from $\{X_i : i \in \mathcal{I}\}$ where the set $\mathcal{I}$ denotes the sampling locations, and this set is a design parameter. If the signal x is K-sparse, one can use the FFAST algorithm to recover x from $O(k)$ samples with $O(k \log k)$ computations [2], [14]. However, the images of interest in MRI are generally not sparse, but they do have sparse wavelet representations [4]. That is, we can express $x = W_n^{-1}\alpha$, where W_n^{-1} is an appropriate wavelet, and α is sparse. Under this signal model, the problem of recovering α can be transformed into the problem decoding on an erasure channel a sparse-graph code. In particular, the graph for the code is drawn

from a smearing ensemble with smearing length L a function of the length of the underlying wavelet filter.

Furthermore, assume α is K-sparse and the length of x is of the form $n = f_1 f_2 f_3$ where f_1, f_2 and f_3 are co-prime. For $m \in \mathbb{Z}$ that divides n, let $D_{m,n} \in \{0,1\}^{m \times n}$ be the regular downsampling matrix from length n to m, that is, $D_{i,j}$ is 1 if $j = ni/m$ or 0 otherwise. Let y_{f_ℓ} for $\ell \in \{1,2,3\}$ be the inverse Fourier transform of the downsampled X, i.e.,

$$y_{f_\ell} \stackrel{\text{def}}{=} F_{f_\ell}^{-1} D_{f_\ell,n} X.$$

Using the properties of Fourier Transform, it follows that

$$y_{f_\ell} = A_{f_\ell,n} x = A_{f_\ell,n} W_n^{-1} \alpha,$$

where $A_{m,n}$ is the aliasing matrix, i.e., $A_{m,n} = [I_m \cdots I_m]$ with I_m is repeated n/m times,

Now, for simplicity, assume that W is a block transform with block size b (eg., for 1 stage Haar wavelets $b = 2$), and the non-zero coefficients of α are distributed uniformly at random. Using the relations between y_{f_1}, y_{f_2} and y_{f_3} and α, recovering α is equivalent to decoding on a random graph from $G(K, M = f_1 + f_2 + f_2, s = [L, L, L])$.

We can actually 'improve' the induced graph if a factor of the signal length has b as a factor. In particular, say that b divides f_1. In this case, it follows that $A_{f_1,n} W_n^{-1} = [W_{f_1}^{-1} \cdots W_{f_1}^{-1}]$, where $W_{f_1}^{-1}$ is repeated n/f_1 times. It can be checked that $y'_{f_1} \stackrel{\text{def}}{=} W_{f_1} y_{f_1} = A_{f_1,n} a$, hence it only aliases the wavelet coefficients. The relation between y'_{f_1}, y_{f_2} and y_{f_3} and α then induces a graph from $G(K, M = f_1 + f_2 + f_2, s = [1, L, L])$, which gives raise to a better threshold.

To complete the equivalence to decoding a sparse-graph code on an erasure channel, we need to be able to check if there is a single component in a bin (a single color in a bin). This can be implemented by processing a shifted version of x (incurring an additional factor of 2 of oversampling). Due to space limitations, we defer the detailed explanation to the extended version [12]. We end this section with the following lemma:

Lemma 4. *Consider a signal α with ambient dimension n and sparsity K, and access to samples from $F_n W_n^{-1} \alpha$. Then, the subsampling scheme described above along with the peeling decoder is able to exactly recover the sparse signal α using $2.63K$ samples and time complexity $\mathcal{O}(K \log K)$.*

Proof. The threshold $2.63K$ follows from the density evolution derived in Section II. We refer the reader to the extended version [12] for complexity proof. □

IV. Conclusions and Future Work

We have introduced the new random graph ensemble, termed the 'smearing ensemble' and derived density evolution recurrences for certain cases from this ensemble. Looking back to our ball coloring game, these recurrences allow us to explicitly analyze strategies involving smearing bounded by smear-length 3. Intriguingly, these results show that some amount of smearing helps to achieve an optimal ball-throwing strategy. A fascinating open question arises here: what is the optimal ball-throwing strategy and what are the density evolution recurrences for such a strategy? In this paper, we have given the first steps in analyzing this problem rigorously and have shown bounds on the smearing ensemble. To do this, we have leveraged the existence of small, structured cycles and introduced the notion of memory into our density evolution. We believe there to be a deep connection between the introduction of memory in our recurrences and the introduction of the 'Onsager' term in the update equations of AMP [15]. We additionally believe the gains seen in spatially coupled ensembles [11] are intimately related to the structural gains of the smearing ensemble. An extremely interesting open problem is to determine the nature of these connections. We have additionally shown the practical connection between the smearing ensemble and the recovery of a sparse wavelet representation of a signal whose samples are taken in the Fourier domain.

References

[1] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[2] S. Pawar and K. Ramchandran, "Computing a k-sparse n-length discrete fourier transform using at most 4k samples and O(klogk) complexity," in *IEEE Int. Symp. on Inf. Theory*. IEEE, Jul. 2013, pp. 464–468.

[3] M. Vetterli and J. Kovacevic, "Wavelets and subband coding," 1995.

[4] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 72–82, Mar. 2008.

[5] T. Richardson and R. Urbanke, *Modern coding theory*. Cambridge University Press, 2008.

[6] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Natl. Acad. Sci. USA*, vol. 106, no. 45, pp. 18 914–18 919, Sep. 2009.

[7] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Jan. 2011.

[8] J. Tan, Y. Ma, and D. Baron, "Compressive imaging via approximate message passing with image denoising," *IEEE Trans. Sig. Process.*, vol. 63, no. 8, pp. 2085–2092, Mar. 2015.

[9] A. Maleki, L. Anitori, Z. Yang, and R. G. Baraniuk, "Asymptotic analysis of complex LASSO via complex approximate message passing (CAMP)," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4290–4308, Mar. 2013.

[10] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 803–834, Jan. 2011.

[11] S. Kudekar, T. Richardson, and R. L. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 7761–7813, Jan. 2013.

[12] K. Chandrasekher, O. Ocal, and K. Ramchandran, "Density evolution on a class of random graphs with cycles (extended)," Jan. 2017. [Online]. Available: http://kabirc.github.io/files/DensityEvolutionSmearing.pdf

[13] C. Häger, H. D. Pfister, F. Brännström *et al.*, "Density evolution for deterministic generalized product codes on the binary erasure channel," *arXiv preprint arXiv:1512.00433*, Dec. 2015.

[14] F. Ong, S. Pawar, and K. Ramchandran, "Fast and efficient sparse 2D Discrete Fourier Transform using sparse-graph codes," *arXiv preprint arXiv:1509.05849*, Sep. 2015.

[15] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing: I. motivation and construction," in *IEEE Inf. Theory Workshop on Inf. Theory*. IEEE, Jan. 2010, pp. 1–5.

APPENDIX

Here, we give rigorous proofs for the bounds on k-smearing.

A. Proof of Lemma 2

Proof. Consider the depth 1 neighborhood of a random node, as shown in Fig. 3. We define the following events: let A_i be the event that node B_i is recovered by time t. Since we are not recursing here, we drop the references to t. Thus, what we are interested in is:

$$\begin{aligned} q_t &= P\left(\bigcup_{i=1}^{k} P(A_i)\right) \\ &= \sum_{i=1}^{L}(-1)^{i-1}\left(\sum_{1\le j_1\le j_2\le\cdots\le j_i\le n} P(A_{j_1}, A_{j_2}, \ldots, A_{j_i})\right) \end{aligned}$$

where the second equality follows from the principle of inclusion-exclusion. We make the following crucial observation: whenever A_i and A_j occur, then all the events $A_\ell, i \le \ell \le j$ also occur. Thus, we can see that $P(A_{j_1}, A_{j_2}, \ldots, A_{j_i}) = P(A_{j_1}, A_{j_i})$. Using this observation, we see:

$$\begin{aligned} P\left(\bigcup_{i=1}^{L} A_i\right) &= \sum_{i=1}^{L} P(A_i) + \sum_{r=2}^{L}(-1)^{r-1} \\ &\quad \cdot \left(\sum_{i,j:j-i-1\ge r-2} \binom{j-i-1}{r-2} P(A_i, A_j)\right) \\ &= \sum_{i=1}^{L} P(A_i) - \sum_{i=1}^{L-1} P(A_i, A_{i+1}) + \\ &\quad \sum_{i,j:j\ge i+2} \sum_{\substack{r: \\ 0\le r-2\le j-i-1}} \\ &\quad (-1)^{r-1}\binom{j-i-1}{r-2} P(A_i, A_j) \\ &= \sum_{i=1}^{L} P(A_i) - \sum_{i=1}^{L-1} P(A_i, A_{i+1}) + \\ &\quad \sum_{i,j:j>i+2} (-1) P(A_i, A_j) \sum_{\ell=0}^{j-i-1} (-1)^{\ell}\binom{j-i-1}{\ell} \\ &= \sum_{i=1}^{L} P(A_i) - \sum_{i=1}^{L-1} P(A_i, A_{i+1}) \end{aligned} \tag{8}$$

Where the last equality follows from the binomial theorem. Now, we note that $P(A_1) = P(A_k) = d_t s_t^{(L-1)}$ and $P(A_i) = d_t s + t^{(L-i)} s_t^{(i-1)}$ where $1 < i < L$. Additionally, we can see that $P(A_i, A_{i+1}) = d_t s_t^{(i)} s_t^{(L-i)}$. Plugging these into equation 8 gives the result. □

B. Proof of Lemma 3

Proof. First, we note that

$$d_t \ge s_t^{(j)} \ge s_t^{(i)} \tag{9}$$

if $i \ge j \ge 1$ by definition. Additionally, we can see that $f(0, 0, \ldots, 0) = 0$ and $f(1, 1, \ldots, 1) = 1$. Now, we have:

$$\begin{aligned} \frac{\partial f}{\partial d_t} &= 2s_t^{(L-1)} + \sum_{i=2}^{L-1} s_t^{(i-1)} s_t^{(L-i)} - \sum_{i=1}^{L-1} s_t^{(i)} s_t^{(L-i)} \\ &= s_t^{(L-1)} + \sum_{i=1}^{L-2} s_t^{(1)}(s_t^{(L-i-1)} - s_t^{(L-i)}) + s_t^{(L-1)} - s_t^{(L-1)} s_t^{(1)} \\ &\ge s_t^{(L-1)} + s_t^{(L-1)}(1 - s_t^{(1)}) \\ &\ge 0 \end{aligned}$$

where the first inequality follows by Equation 9 and the last inequality follows since $s_t^{(1)} \le 1$ and $s_t^{(L-1)} \ge 0$. Additionally, we can see that for $1 \le i < L-1$:

$$\frac{\partial f}{\partial s_t^{(i)}} = 2d_t(s_t^{(L-i-1)} - s_t^{(L-i)}) \ge 0$$

where the inequality follows from Equation 9. The argument follows similarly for $\frac{\partial f}{\partial s_t^{(L-1)}}$. Thus, since the partial derivatives are all non-negative, the result follows. □

C. 3-smearing ($S[1, 1, 3]$)

In this section, we provide exact thresholds for the ensemble drawn from $S[1, 1, 3]$, illustrate why it is difficult to generalize, and draw out the structure in the smearing patterns.

We introduce the notation p_t to denote the probability of an edge from a node to a bin in a stage with 1 smearing is *not* removed at time t. Also note that there are streams missing from the diagram, the symmetric picture for nodes "hanging off the edge" are not shown. We can see that (from Lemma 2):

$$q_t = P(A_1 \cup A_2 \cup A_3) \tag{10}$$

$$\begin{aligned} &= P(A_1) + P(A_2) + P(A_3) - P(A_1 \cap A_2) \\ &\quad - P(A_2 \cap A_3) - P(A_1 \cap A_3) + P(A_1 \cap A_2 \cap A_3) \end{aligned} \tag{11}$$

$$= d_t(2s_t^{(2)} + (s_t^{(1)})^2 - 2s_t^{(2)} s_t^{(1)}) \tag{12}$$

Now, exactly as in section II-A, we can see that:

$$d_t = e^{-\lambda p_{t-1}^2} \tag{13}$$

Now we analyze $s_t^{(1)}$. We omit the detail before the Taylor series approximation for readability. Now, note that the nodes in the stream can either all be cleared from the other stages, or they can be cleared by the bin shared by all streams tracked by $s_t^{(2)}$, call this bin B_1. Additionally call the bin shared by the stream thrown to the outermost bin as B_2. The probability that the stream is cleared from the other stage is $e^{-p_{t-1}^2}$. The only way the stream could have been cleared by B_1 is if there were exactly 1 node remaining at time $t-1$ and B_1 knows it is a singleton at time $t-1$. Note that B_1 is contaminated by the stream of balls thrown to the outermost bin and this stream must have been cleared by time $t-2$. This can happen in 3 not necessarily disjoint ways. First, B_1 is "clear from below" and all the balls in the stream thrown to the outermost bin were peeled by $t-2$. This happens with probability

$$s_{t-1}^{(1)} e^{-\lambda p_{t-2}^2} \tag{14}$$

Second, exactly one node is missing from the stream of balls thrown to B_1, the stream of nodes thrown to the outermost bin was empty at time $t-2$, and B_2 knows it is a singleton at time $t-2$. This happens with probability:

$$\left(e^{-\lambda p_{t-2}^2} + s_{t-1}^{(2)} \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2}\right) \cdot \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2} \tag{15}$$

Finally, one ball can be missing from the stream of balls thrown to the outermost bin. Then, B_2 must be a singleton at time $t-2$, so we have the probability:

$$s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2} \tag{16}$$

Finally, we add a term to correct for overcounting. The overlap between the first and last terms is:

$$s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-2\lambda p_{t-3}^2} \tag{17}$$

Note that if at time $t-2$, there were exactly 1 node in the stream of nodes thrown to B_1, and both B_2 and the bin below were singletons, then the first and last terms overlap. Also note that these two streams are exactly $s_{t-2}^{(2)}$ from the perspective of B_2 and the bin below, and the term follows.

Thus, we can see that:

$$\begin{aligned} s_t^{(1)} = e^{-\lambda p_{t-1}^2} + \lambda p_{t-1}^2 e^{-\lambda p_{t-2}^2} \cdot \Bigg[& s_{t-1}^{(1)} e^{-\lambda p_{t-2}^2} + \\ & \left(e^{-\lambda p_{t-2}^2} + s_{t-1}^{(2)} \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2}\right) \\ & \cdot \lambda p_{t-2}^2 e^{-2\lambda p_{t-3}^2} \\ & + s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2} - s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-2\lambda p_{t-3}^2} \Bigg] \end{aligned} \tag{18}$$

Finally, we give:

$$\begin{aligned} s_t^{(2)} = e^{-2\lambda p_{t-1}^2} + \lambda p_{t-1}^2 e^{-\lambda p_{t-2}^2} \cdot & \\ \Bigg[2 s_{t-1}^{(1)} e^{-\lambda p_{t-2}^2} + & \left(e^{-\lambda p_{t-2}^2} + s_{t-1}^{(2)} \lambda p_{t-2}^2 \right. \\ & \left. \cdot e^{-\lambda p_{t-3}^2}\right) \lambda p_{t-2}^2 e^{-2\lambda p_{t-3}^2} \\ & + s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-\lambda p_{t-3}^2} + s_{t-1}^{(2)} e^{-\lambda p_{t-1}^2} \\ & - s_{t-1}^{(2)} e^{-\lambda p_{t-1}^2} - s_{t-2}^{(2)} \lambda p_{t-2}^2 e^{-2\lambda p_{t-3}^2} \Bigg] \end{aligned} \tag{19}$$

We again note that these recursions show tight agreement with simulations. Note that while difficult to digest, there is significant structure in the recursions. Analyzing the structure from whether or not a bin is cleared leads us to the bounds of the next section. Additionally, one can see that whereas 2 smearing involved 1 step of memory, 3 smearing involves 2 steps of memory, and it becomes clear that in general L smearing will involve $L-1$ steps of memory. The amount of memory in addition to the smearing length causes the complexity to blow up. A simple lower bound that is easy to generalize is given in the next section and based on these observations.

D. Proof of Lemma 4

In order to describe the computational complexity, we first give pseudocode for the decoding algorithm:

Algorithm 1 Basis-aware Peeling

Input : Coupled bipartite graph G
Output: X
while *singletons remain* **do**
  for B *in bins* **do**
    if B *singleton, good stage* **then**
      Peel(B)
    end
    if B *singleton, bad stage* **then**
      AddToHypothesisList(B)
    end
  end
end

Algorithm 2 AddToHypothesisList

Input: Bin B
$L \leftarrow$ Location(B) $V \leftarrow$ Value(B) **for** *Bin* B_1 *connected to variable node* (L, V) **do**
  for *Basis* b *in set of filters* **do**
    $B_1 \leftarrow$ Hypothesis (b, L, V) **if** *Singleton created* **then**
      Peel(B);
    end
  end
end

Now, note that the creation of the bipartite graph is done in $\mathcal{O}(K \log K)$ time and is disjoint from the decoding process. Decoding uses an iterative decoder with a constant number of iterations [5]. Additionally, the number of bins iterated over is linear in K. We now note the size of the hypothesis list in each bin is $\mathcal{O}(K)$. Thus, if the list is stored using a data structure with $\mathcal{O}(\log K)$ insertion and deletion, such as a red-black tree, the iterative decoding complexity is $\mathcal{O}(K \log K)$.