Counter Braids: Asymptotic Optimality of the Message Passing Decoding Algorithm

Yi Lu, Andrea Montanari and Balaji Prabhakar

***Abstract*— A novel counter architecture, called Counter Braids, has recently been proposed for per-flow counting on high-speed links. Counter Braids has a layered structure and compresses the flow sizes as it counts. It has been shown that with a Maximum Likelihood (ML) decoding algorithm, the number of bits needed to store the size of a flow matches the entropy lower bound. As ML decoding is too complex to implement, an efficient message passing decoding algorithm has been proposed for practical purposes.**

The layers of Counter Braids are decoded sequentially, from the most significant to the least significant bits. In each layer, the message passing decoder solves a sparse signal recovery problem. In this paper we analyze the threshold dimensionality reduction rate (d-rate) of the message passing algorithm, and prove that it is correctly predicted by density evolution.

Given a signal in $\mathbb{R}_+^n$ with $n\epsilon$ non-vanishing entries, we prove that one layer of Counter Braids can reduce its dimensionality by a factor $2.08 \cdot \epsilon \log(1/\epsilon) + O(\epsilon)$. This essentially matches the rate for sparse signal recovery via L_1 minimization, while keeping the overall complexity linear in n.

I. Introduction

Per-flow measurement on high-speed links is known to be a technologically challenging problem [1], [2]. Over a period of the order of a few minutes at a 10-Gbps link, millions of flow sizes need to measured. Each flow is divided into packets that interleave with each other. Counter updates are initiated when packets arrive. The inter-arrival time between packets can be as short as 40 nanoseconds for a 10-Gbps link, and the link speed is expected to rise sharply in the future.

The large number of flows and rapid arrival of packets imply that per-flow measurement requires a large array of counters that can be updated at very high speed. However, large memories with high-speed accesses are infeasible with current technology, hence accurate per-flow measurement with small memory space is needed. We refer the reader to [3] for a detailed description of the problem and previous approaches.

A novel counter architecture, called Counter Braids (CB), has recently been proposed [3], [4] to count with only a few bits per flow. CB uses less space as it compresses the flow sizes as it counts. It was shown in [4] that with a Maximum Likelihood (ML) decoding algorithm, the number of bits needed to store the size of a flow matches the entropy lower bound. For implementation, a low-complexity message passing decoding algorithm was proposed in [3].

Counter Braids has a layered structure with the least significant bits of the flow sizes contained in the bottom-most layer, and the most significant bits in the topmost layer. The message passing algorithm decodes each layer sequentially, from the top to the bottom. More details on the architecture of CB, layered decoding, and overall performance can be found in [3], [4].

A. Comparison with Compressed Sensing

Compressed sensing [5], [6] reduces below Nyquist rate the number of samples needed to recover sparse signals. In other words, it *reduces the dimensionality* of signal known to be sparse, using suitable non-adaptive projections. Counter Braids, on the other hand, *compresses* a signal with decreasing digit entropy: it reduces the actual *number of bits* needed to store the signal and achieves the Shannon entropy lower bound.

Interestingly, decoding each layer of CB also solves a *dimensionality reduction* problem. By reducing the number of counters in each layer, and assigning an appropriate number of bits per counter, CB achieves an overall reduction *in bits*. In this way, CB performs compression *via* dimensionality reduction.

In order to differentiate between the compression rate and the dimensionality reduction rate, we introduce the notations:

The dimensionality reduction rate or, for short, *d-rate* β. In our context, it is the number of counters per flow in a single layer of CB.

The compression rate or, for short, *rate* r. In our context, it is the total number of bits per flow in CB.

In this paper, we focus on the decoding of a single-layer using message passing and address the following question:

> *What is the* optimal dimensionality reduction rate (d-rate) *achievable for a given sparsity under the message passing algorithm proposed in [3]?*

The results in this paper are exact counterparts of the "weak threshold" for non-negative signals defined by Donoho and Tanner in [7]. The latter determines (in the large dimension limit) the optimal dimensionality reduction factor that allows for *most* signals of a given sparsity to be recovered exactly, using random gaussian measurement matrices and L_1 minimization recovery algorithm. Let us stress that the present scheme has two advantages with respect to the one in [7]: (i) The measurement matrix is itself sparse, and hence each measurement can be taken in $O(1)$ operations; (ii) The message passing recovery algorithm has complexity $O(n)$. In view of these considerations, it is

Yi Lu is with the Department of Electrical Engineering, Stanford University, yi.lu@stanford.edu. Andrea Montanari is with Departments of Electrical Engineering and Statistics, Stanford University, montanari@stanford.edu. Balaji Prabhakar is with the Department of Electrical Engineering, Stanford University, balaji@stanford.edu.

surprising that the Counter Braids threshold turns out to be extremely close to the one by Donoho and Tanner.

Remark. Threshold of the same order as that by Donoho and Tanner is obtained using expander graph arguments by Berinde and Indyk in [8] and by Xu and Hassibi in [9]. However, tight bounds on constants are difficult to obtain as is characteristic of expansion arguments. In conjunction with the sparse measurement matrix, L_1 minimization recovery algorithm is used in [8] and a recovery algorithm with $O(n)$ complexity is used in [9].

B. Main Results

Given a vector $\mathbf{f} \in \mathbb{R}_+^n$ with $n\epsilon$ non-vanishing entries, we refer to ϵ as the sparsity parameter. The d-rate, sparsity pair (β, ϵ) is *achievable under message passing* (MP-achievable) if there exists a sequence of graphs $\{\mathcal{G}_n\}$ of size $n \to \infty$ such that: (1) The d-rate of $\mathcal{G}_n$ converges to β as $n \to \infty$; (2) The error probability for recovering ϵ-sparse vectors goes to 0 as $n \to \infty$.

Our aim is to determine the optimal tradeoff between rate and sparsity under low complexity message passing decoding. We thus define

$$\beta_*(\epsilon) = \inf \Big\{\beta \ : \ (\beta, \epsilon) \ \text{ is MP achievable}\Big\} . \tag{1}$$

A first step toward the determination of $\beta_*(\epsilon)$ is in the following results. The next theorem characterizes the achievable threshold in the *sparse* regime.

Theorem 1. *Let $\kappa_* = 1/(\log 2)^2 \approx 2.08137$. Then, for any $\epsilon < 1/2$*

$$\beta_*(\epsilon) \le \kappa_* \, \epsilon \log(1/\epsilon) . \tag{2}$$

Notice that, as $\epsilon \to 0$, [7] proves that random gaussian matrices achieve, in conjunction with L_1 minimization, *d*-rate equal to $\beta(\epsilon) \approx 2\epsilon \log(1/\epsilon)$.

In the *dense* regime, a better bound (and better architecture to achieve it) is provided by the following

Theorem 2. *For any $\epsilon \in [0, 1]$ we have*

$$\beta_*(\epsilon) \le \sqrt{\epsilon} . \tag{3}$$

Figure 1 compares the upper bounds on $\beta_*(\epsilon)$ with the threshold by Donoho and Tanner [7]. The latter is reproduced by linearly interpolating sample points listed in [7] since the expression is not explicit, and the curve is a lower bound of the actual threshold.

Let us finally emphasize an important technical point. While the proofs of [3] where written in the case of a flow vector f with iid entries, they indeed hold for most deterministic vectors as well (this corresponds to the weak threshold setting of [7]). Indeed enough randomization is provided by the random bipartite graphs that define Counter Braids, as the graph distribution is symmetric under permutation of the flows.

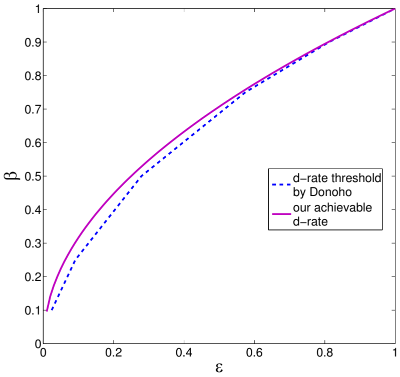

Fig. 1. Comparison between the dimensionality reduction rate achieved via Gaussian matrices and L_1 minimization (dashed [7]) and the one proved in the present paper (continuous).

II. Proof Overview

We start by specifying the single-layer message passing algorithm and some useful notations. There are three parts to the proof: density evolution threshold, exchange of limit, and degree distribution optimization. The corresponding theorems are stated in Section II-C.

A. Single-Layer Message Passing Algorithm

Consider a sparse random bipartite graph with flow nodes on the left and counter nodes on the right, as shown in Figure 2. The vector $\mathbf{f}$ denotes flow sizes and $\mathbf{c}$ denotes counter values. We have

$$c_a = \sum_{i \in \partial a} f_i ,$$

where ∂a denotes all flows that connect to counter a. The decoding problem is to estimate $\mathbf{f}$ given $\mathbf{c}$.

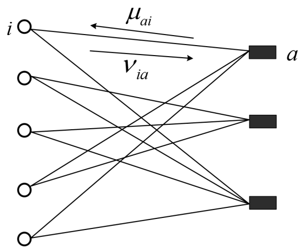

Fig. 2. Message passing on a bipartite graph with flow nodes (circles) and counter nodes (rectangles.)

We consider the following message passing algorithm specified in Exhibit 1. The algorithm is iterative. In the t^{th} iteration, the message from counter node a to flow node i is denoted by $\mu_{ai}(t)$ and the message from flow node i to counter node a is denoted by $\nu_{ia}(t)$. At the end of T iterations, flow estimates, denoted by $\widehat{f_i}(T)$, are computed for each of the flows.

Exhibit 1: The Message Passing Decoding Algorithm

```
Initialize
    $f_{min}$ = minimum flow size;
    $\nu_{ia}(0) = 0 \quad \forall i$ and $\forall a$;
    $c_a = a^{th}$ counter value

Iterations
    for iteration number $t = 1$ to $T$
        $\mu_{ai}(t) = \max\left\{\left(c_a - \sum_{j\neq i}\nu_{ja}(t-1)\right), f_{min}\right\}$;
        $\nu_{ia}(t) = \begin{cases} \min_{b\neq a}\mu_{bi}(t) & \text{if } t \text{ is odd,} \\ \max_{b\neq a}\mu_{bi}(t) & \text{if } t \text{ is even.} \end{cases}$

Final Estimate
    $\widehat{f}_i(T) = \begin{cases} \min_a\{\mu_{ai}(T)\} & \text{if } T \text{ is odd,} \\ \max_a\{\mu_{ai}(T)\} & \text{if } T \text{ is even.} \end{cases}$
```

B. Notations

We define the sparse bipartite graph ensemble. Let the number of flow nodes be n, and the number of counter nodes be m. First, we define degree distributions.

Definition 1. (Degree distribution) *Let Λ_i be the number of flow nodes with degree i, such that $\sum_i \Lambda_i = n$. We call $L_i = \Lambda_i/n$ the node-perspective distribution, and its generating function:*

$$L(x) = \sum_{i=1}^{\infty} L_i x^i.$$

Similarly, let P_i be the number of counter nodes with degree i such that $\sum_i P_i = m$. We call $R_i = P_i/n$ the node-perspective distribution for the counter nodes, and its generating function:

$$R(x) = \sum_{i=1}^{\infty} R_i x^i.$$

Since the number of edges originating from flow nodes are the same as that from counter nodes, we require

$$\Lambda'(1) = P'(1).$$

Define the edge-perspective degree distribution

$$\begin{aligned} \lambda(x) &= \sum_{i=1}^{\infty} \lambda_i x^{i-1} = \frac{L'(x)}{L'(1)} \\ \rho(x) &= \sum_{i=1}^{\infty} \rho_i x^{i-1} = \frac{R'(x)}{R'(1)} \end{aligned}$$

λ_i is the probability that a uniformly random edge is connected to a flow node of degree i (thus connected to $(i-1)$ other edges).

Definition 2. *The random bipartite graph $\mathcal{G}(n, \lambda, \rho)$ is defined as follows. The graph includes n flow nodes and m counter nodes. The flow nodes have degree distribution $L(x)$, with $L_0 = L_1 = 0$, i.e. flow nodes have at least degree 2 and bounded maximum degree. Similarly, counter nodes have degree distribution $R(x)$. A node of degree i has i sockets from which the i edges emanate, hence there are $\Lambda'(1) = P'(1)$ sockets on each side. Label the sockets on each side with the set $[\Lambda'(1)] = \{1, \cdots, \Lambda'(1)\}$. Let σ be a uniformly random permutation on $[\Lambda'(1)]$. Associate σ to a bipartite graph by connecting the i-th socket on the flow side to the $\sigma(i)$-th socket on the counter side.*

Definition 3. (Error Probability) *We denote by $\mathrm{P}_{(\lambda,\rho)}(t; n) = \mathbb{E}_{\mathcal{G}(n,\lambda,\rho)}\mathbb{P}(f_i \neq \widehat{f}_i(t))$ the expected error probability under message passing decoding after t iterations for a random graph from the ensemble $\mathcal{G}(n, \lambda, \rho)$. Whenever clear from the context we shall omit the subscript (λ, ρ).*

Let ϵ be the sparsity parameter defined as the proportion of flows with sizes more than f_{min}, which is the same as the proportion of positive non-vanishing terms in a non-negative signal. The non-minimum flow sizes can have arbitrary magnitudes and no distribution is assumed.

C. Proof of Main Results

For a given pair of degree distribution, we can obtain the threshold for ϵ below which the asymptotic expected error probability, obtained by first letting n go to infinity and *then* letting t go to infinity, is zero and above which it is strictly positive.

Theorem 3. *Given a degree distribution pair (λ, ρ), let*

$$f(x) = \epsilon\lambda(1 - \rho(1 - \lambda(1 - \rho(1 - x)))).$$

The threshold of ϵ associated with the degree distribution λ, call it ϵ^*, is defined as

$$\epsilon^*(\lambda, \rho) \equiv \sup\left\{\epsilon : \lim_{t\to\infty}\lim_{n\to\infty} \mathrm{P}_{(\lambda,\rho)}(t; n) = 0\right\}.$$

We have

$$\begin{aligned} \textit{(i)}\ \epsilon^* &\equiv \sup\{\epsilon \in (0, 1] : x = f(x) \textit{ has no solution}, \\ &\quad x \in (0, 1]\} \\ \textit{(ii)}\ \epsilon^* &\equiv \inf\{\epsilon \in (0, 1] : x = f(x) \textit{ has a solution} \\ &\quad x \in (0, 1]\} \end{aligned}$$

Achievable pairs (β, ϵ), however, are defined by first letting t go to infinity and *then* n go to infinity. Hence we need to show that the density evolution threshold coincides with the achievable threshold.

Theorem 4. *Let $\mathrm{P}(t; n)$ be the expected error probability under message passing decoding, for a random graph in the ensemble $\mathcal{G}(n, \lambda, \rho)$. Then*

$$\lim_{t\to\infty}\lim_{n\to\infty} \mathrm{P}(t; n) = \lim_{n\to\infty}\lim_{t\to\infty} \mathrm{P}(t; n).$$

This implies that for a given degree distribution pair (λ, ρ), for all $\epsilon < \epsilon^*(\lambda, \rho)$,

$$\lim_{n\to\infty}\lim_{t\to\infty} \mathrm{P}(t; n) = 0 \tag{4}$$

and for all $\epsilon > \epsilon^*(\lambda, \rho)$, and the degree distribution pair (λ, ρ) has bounded degrees,

$$\lim_{n\to\infty} \lim_{t\to\infty} \mathrm{P}(t; n) > 0. \tag{5}$$

Hence, we can optimize the density evolution threshold over graph ensembles to upper bound the optimal d-rate $\beta_*(\epsilon)$.

Theorem 5. *Let $\kappa_* = 1/(\log 2)^2 \approx 2.08137$. Then, for any $\epsilon < 1/2$, there exists a sequence of degree distribution pairs (λ_k, ρ_k) with d-rate*

$$\lim_{k\to\infty} \beta_k = \kappa_* \, \epsilon \log(1/\epsilon) \, ,$$

and with density evolution threshold $\lim_{k\to\infty} \epsilon^*(\lambda_k, \rho_k) > \epsilon$.

Theorem 6. *For any $\epsilon \in [0, 1]$, there exists a sequence of degree distribution pairs (λ_k, ρ_k) with d-rate*

$$\lim_{k\to\infty} \beta_k = \sqrt{\epsilon} \, ,$$

and with density evolution threshold $\lim_{k\to\infty} \epsilon^*(\lambda_k, \rho_k) > \epsilon$.

Theorem 5 and 6, together with Theorem 4, complete the proof for Theorem 1 and 2.

The rest of the paper is organized as follows. We prove Theorem 3 in Section III and Theorem 4 in Section IV. Theorem 5 and 6 are shown in Section V. We conclude in Section VI.

III. Density Evolution Threshold

We shall show the following theorem, which yields the density evolution equation for the message passing decoder. Theorem 3 is then obtained in a similar way as in [10].

Theorem 7. *Consider a degree distribution pair (λ, ρ) with the flow node-perspective degree distribution $L(x)$. Let $x_0 = 1$ and for $t \geq 0$ let*

$$x_t \;=\; \begin{cases} \lambda(1 - \rho(t - x_{t-1})) & \textit{if } t \textit{ is odd,} \\ \epsilon\lambda(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is even.} \end{cases} \tag{6}$$

Then for $t \geq 0$,

$$\lim_{n\to\infty} \; \mathrm{P}_{(\lambda,\rho)}(t; n) = \begin{cases} L(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is odd,} \\ \epsilon L(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is even.} \end{cases} \tag{7}$$

A version of Theorem 7 is stated in [3] with the counter degree distribution ρ restricted to the Poisson distribution. We present the proof for the general version in this paper.

In order to prove Theorem 7, we first show that as $n \to \infty$, the error probability after t iterations of the message passing algorithm on $\mathcal{G}(n, \lambda, \rho)$ converges to the error probability at the root of a *finite* tree ensemble $T_t(\lambda, \rho)$. We then write the density evolution equation on the tree ensemble $T_t(\lambda, \rho)$ to compute $\mathrm{P}_{(\lambda,\rho)}(t; n)$ recursively.

Definition 4. (Tree Ensemble) *The tree ensemble $T_t(\lambda, \rho)$ is the distribution over bipartite trees recursively constructed as follows. $T_1(\lambda, \rho)$ is a random tree rooted at flow node i with degree distribution $L(x)$, and each of its children counter nodes having offspring distribution $\rho(x)$. To sample from $T_t(\lambda, \rho)$, $l \geq 2$, first sample an element from $T_{t-1}(\lambda, \rho)$. Next substitute each of its leaf flow node with a flow node with offspring distribution $\lambda(x)$. Finally, substitute each of its leaf counter node with a counter node having degree distribution $\rho(x)$.*

Definition 5. (Error Probability for Tree Ensemble) *For the tree ensemble, assign counter values on one element T of the ensemble as follows. Given an arbitrary flow size vector with sparsity ϵ, set all counter node values to be the sum of its neighboring flow nodes. Let $\mathrm{P}_{T(\lambda,\rho)}(t) = \mathbb{P}(f_i \neq \widehat{f}_i)$ be the expected error probability under the same message passing decoding algorithm at the root of the random tree $T_t(\lambda, \rho)$.*

Standard local convergence arguments (cf. for instance Theorem 3.4.9 in [10]) imply the following.

Lemma 1. (Convergence to Tree Ensemble)*Consider the sequence of graph ensemble $\mathcal{G}(n, \lambda, \rho)$ with increasing n under t iterations of message passing decoding. Then*

$$\lim_{n\to\infty} \mathrm{P}_{(\lambda,\rho)}(t; n) = \mathrm{P}_{T(\lambda,\rho)}(t).$$

Note that all incoming messages to a node in the tree ensemble are independent. Using the anti-monotonicity property (Lemma 2 in [3]), we can write the density evolution equation. We reproduce the lemma below, followed by the lemma on density evolution.

Lemma 2. Anti-monotonicity Property. *If ν and ν' are such that for every i and a, $\nu_{ia}(t-1) \leq \nu'_{ia}(t-1) \leq f_i$, then $\nu_{ia}(t) \geq \nu'_{ia}(t) \geq f_i$. Consequently, since $\widehat{\mathbf{f}}(0) = 0$, $\widehat{\mathbf{f}}(2t) \leq \mathbf{f}$ component-wise and $\widehat{\mathbf{f}}(2t)$ is component-wise non-decreasing. Similarly $\widehat{\mathbf{f}}(2t + 1) \geq \mathbf{f}$ and is component-wise non-increasing.*

Lemma 3. *Consider a degree distribution pair (λ, ρ) with the flow node-perspective degree distribution $L(x)$. Let ϵ be the proportion of flows that are not of the minimum size. Let $x_0 = 1$ and for $t \geq 0$ let*

$$x_t \;=\; \begin{cases} \lambda(1 - \rho(t - x_{t-1})) & \textit{if } t \textit{ is odd,} \\ \epsilon\lambda(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is even.} \end{cases} \tag{8}$$

Then for $t \geq 0$,

$$\mathrm{P}_{T(\lambda,\rho)}(t) = \begin{cases} L(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is odd,} \\ \epsilon L(1 - \rho(1 - x_{t-1})) & \textit{if } t \textit{ is even.} \end{cases} \tag{9}$$

Proof. Let x_t be the probability that the message originating from the root of $T_t(\lambda, \rho)$ is incorrect, and y_t be the probability that the message going from a counter node towards the root of $T_t(\lambda, \rho)$ is incorrect.

Since $\nu_{ia}(0)$ is initialized to 0 for all i and a, $x_0 = 1$.

Let us consider a counter-to-flow message emitting from a counter node of degree k. By definition of the algorithm, the message is in error if at least one of the $(k-1)$ incoming message is in error. By assumption of the tree ensemble, all incoming messages are independent, hence

$$y_t = 1 - (1 - x_{t-1})^{k-1}.$$

With probability $\rho_{\gamma,k}$, the edge connects to a counter node with degree k, it follows that the expected error probability of a counter-to-flow message in the t-th iteration is

$$\begin{aligned} y_t &= \sum_k \rho_{\gamma,k}(1-(1-x_{t-1})^{k-1}) \\ &= 1-\rho(1-x_{t-1}). \end{aligned} \tag{10}$$

Next let us consider a flow-to-counter message emitting from a flow node of degree k. If t is odd, by Lemma 2, $\mu_{ai}(t) \geq f_i$. Since $\nu_{ia}(t) = \min_{b \neq a} \mu_{bi}(t)$, it is in error only if all of the $k-1$ incoming messages are in error. By assumption of the tree ensemble,

$$x_t = y_t^{k-1}.$$

Averaging over the distribution of k, we have

$$x_t = \lambda(y_t) = \lambda(1-\rho(1-x_{t-1})).$$

If t is even, by Lemma 2, $\mu_{ai}(t) \leq f_i$. In addition, by definition of the algorithm, $\mu_{ai}(t) \geq f_{min}$. Since $\nu_{ia}(t) = \max_{b \neq a} \mu_{bi}(t)$, it is in error only if $f_i \neq f_{min}$ *and* all of the $k-1$ incoming messages are in error. Averaging over the distribution of k, we have

$$x_t = \epsilon\lambda(y_t) = \epsilon\lambda(1-\rho(1-x_{t-1})).$$

To compute $P_{T(\lambda,\rho)}(t)$, we only need to replace the edge-perspective distribution $\lambda(x)$ with the node-perspective distribution $L(x)$, and we get (9). ■

Together, Lemma 1 and Lemma 3 yield Theorem 7.

IV. Exchange of Limit

We present the proof for Theorem 4 in two parts, for (4) and (5) respectively.

To show (4) for $\epsilon < \epsilon^*(\lambda,\rho)$, we only need to observe that the algorithm is monotonic, i.e., a correct message stays correct. This follows directly from the anti-monotonicity property (Lemma 2). With the observation, we obtain that for any δ, there exists an N such that for all $n > N$,

$$\lim_{t\to\infty} \mathrm{P}(t;n) \leq \mathrm{P}(t;n) \leq \delta.$$

The first inequality follows from the monotonicity and the second inequality follows from the definition of the density evolution threshold ϵ^*, which yields

$$\lim_{t\to\infty}\lim_{n\to\infty} \mathrm{P}(t;n) = 0.$$

To show (5) for $\epsilon > \epsilon^*(\lambda,\rho)$, we introduce the "peeling decoder" as follows. It is not a practical decoder as it assumes partial knowledge of the indices of flows of size f_{min}. However it is easy to check that it is equivalent to the message passing decoding algorithm.

Peeling Decoder

Consider the following residual graph illustrated in Figure 3. Each edge and flow node in the original decoding graph are duplicated, and connected to the check nodes from two sides. The messages on the left edges are lower bounds, computed at even iterations, and the messages on the right edges are upper bounds, computed at odd iterations. The residual graph consists of edges on which the messages are incorrect (different from true flow sizes).

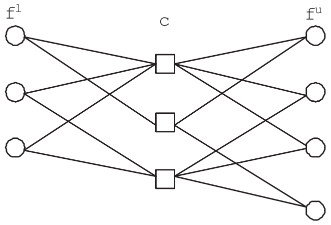

Fig. 3. Residual graph.

Let f^l be the flow nodes on the left, f^u be the flow nodes on the right and c be the counter nodes. Let e^l denote the edges on the left and e^u denote the edges on the right. Each edge is indexed by the flow node and counter node it is attached to. We define the *complement set* of an edge $e^u_{i,a}$ ($e^l_{i,a}$) as edges $e^l_{j,a}$ ($e^u_{j,a}$) where j is any index other than i. Figure 4 shows edge e and its complement set, with other edges in dashed lines. An edge is "removable" if all edges in its complement set have been removed.

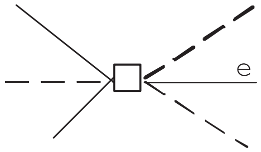

Fig. 4. Complement set of edge e.

The procedure of the peeling decoder is as follows.

1. Initialize the residual graph with the decoding graph duplicated on both sides of check nodes: an edge connects f^l_i to c_a and f^u_i to c_a if f_i and c_a are adjacent on the message passing decoding graph. Next remove all nodes f^l whose true flow size is f_{min}, and all the edges e^l connected to them.
2. Remove **one** "removable" edge from e^u chosen uniformly from all "removable" edges of e^u. Remove the node f^u this edge connects to, together with all other edges connected to f^u.
3. Remove **all** "removable" edges from e^l. Each edge removed peels the flow node f^l it connects to, together with all other edges connected to f^l.
4. Repeat 2 and 3.

The process stops when there is no more "removable" edges in e^u. The decoding is successful if all edges in the graph are removed.

In principle, we can write differential equations that describe the evolution of the residual graph and recover the density evolution equation, as done in [11] for the erasure channel decoding. This would show the validity of the exchange of limits. However, it becomes tedious as we deal

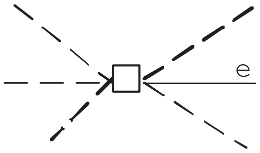

Fig. 5. $d_1 = d_3 = 0$, $d_2 > 0$.

with upper and lower bounds in our algorithm. We use an alternative approach for the proof.

Instead of starting with a complete graph and investigate the asymptotic (in n) error probability after the peeling decoder stops, we start by running the *message passing decoder* for a large, but *finite*, number of iterations. For large enough n, we get a residual graph whose degree distributions are arbitrarily close to those given by density evolution. In particular, the fraction of "removable" nodes is *arbitrarily close to* 0, and the fraction of incorrect messages is large. Next we run the peeling decoder on this residual graph, and show that at each step, the expected fraction of "removable" nodes is decreasing, until it becomes 0. Hence when the peeling decoder stops, the fraction of incorrect messages hardly changes, and we have finished the proof. The details are as follows.

For a given n and its residual graph, let the number of edges from f^l (f^u) to c whose flow node degree is i be F_i^l (F_i^u). A counter node has three degrees associated to it: the degree of edges to f^l, denoted by d_1, the degree of edges to f^u, denoted by d_2, and the degree of edges that are duplicated on both sides, denoted by d_3. Let the number of edges from f^l (f^u) to c with $d_1 = i$, $d_2 = j$ and $d_3 = k$ be $C_{i,j,k}^l$ ($C_{i,j,k}^u$).

For given degree distributions λ and ρ with bounded maximum degree l_{max} and r_{max}, choose an arbitrary $\epsilon > \epsilon^*$. We know that the density evolution curve intersects the diagonal at two points. Let the gradient at the larger intersection be $1-\delta$, $\delta > 0$. Let the degree distributions corresponding to the density evolution limit be $\widehat{F}_i^l$, $\widehat{F}_i^u$, $\widehat{C}_{i,j,k}^l$, $\widehat{C}_{i,j,k}^u$. Run the message passing decoder for a large number of iterations, say T iterations, such that $\max_i |F_i^l - \widehat{F}_i^{l,T}| < \delta_1$, and the same holds for $\widehat{F}_i^{u,T}$, $\widehat{C}_{i,j,k}^{l,T}$ and $\widehat{C}_{i,j,k}^{u,T}$ (we shall write the bounds for F_i^l only, assuming the same holds for the rest). The uniform convergence of F_i^l can be shown using $\sum_i \widehat{F}_i^l < \Lambda'(1)$ and the monotonicity of $\sum_{i=j}^{l_{max}} \widehat{F}_i^l$ in j.

Using density evolution result, we obtain that for large enough n, $\max_i |F_i^l - \widehat{F}_i^{l,T,n}| < \delta_1 + \delta_2$. Next we use the peeling decoder on the residue graph with degrees $\widehat{F}_i^{l,T,n}$, $\widehat{F}_i^{u,T,n}$, $\widehat{C}_{i,j,k}^{l,T,n}$ and $\widehat{C}_{i,j,k}^{u,T,n}$. We omit the superscript T and n to lighten the notations.

We start by specifying the degrees of the removable edges.

Figure 5 and 6 illustrate the two cases when edges become removable :
1) $d_1 = d_3 = 0$, $d_2 > 0$, all d_u edges are removable;
2) $d_3 = d_1 = 1$, only 1 edge is removable.

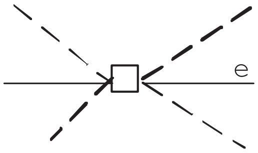

Fig. 6. $d_3 = d_1 = 1$.

Next, we specify the degrees of an edge in e^l upon the removal of which generates *new* removable edges in e^u. There are 3 cases:
1) $d_1 = 2$, $d_2 = i$, $d_3 = 1$ or 2. Suppose we have $d_3 = 1$ after the removal, this generates 1 new removable edge.
2) $d_1 = 1$, $d_2 = i$, $d_3 = 1$. This generates $i-1$ new removable edges.
3) $d_1 = 1$, $d_2 = i$, $d_3 = 0$. This generates i new removable edges.

We compute the degree distribution of the residual graph at the density evolution limit. Let the limit be x, we have from Theorem 3

$$\epsilon\lambda(1-\rho(1-\lambda(1-\rho(1-x)))) = x.$$

Let $y = 1-\rho(1-x)$, $u = \lambda(y)$, $z = 1-\rho(1-u)$.

Let $D = \Lambda'(1)$. The initial numbers of edges of relevant degrees are:

$$\begin{array}{ccl}
F_i^l & = & D\epsilon\lambda_i z^{i-1} \\
F_i^u & = & D\lambda_i y^{i-1} \\
C_{2,i,2}^l + & C_{2,i,1}^l & = \\
D\sum\limits_l & \rho_l & \binom{l-1}{1}\binom{l-1}{i-1} x^2(1-x)^{l-2}u^i(1-u)^{l-i} \\
C_{1,i,1}^l & = & D\sum\limits_l \rho_l \binom{l-1}{i-1} x(1-x)^{l-1}u^i(1-u)^{l-i} \\
C_{1,i,0}^l & = & D\sum\limits_l \rho_l \binom{l-1}{i} x(1-x)^{l-1}u^i(1-u)^{l-i},
\end{array}$$

$C_{i,2,2}^u + C_{i,2,1}^u$, $C_{i,1,1}^u$ and $C_{i,1,1}^u$ are defined analogously with x and u exchanged.

We start with the above degree distribution, and we are interested in the expected change in the number of removable edges after we run step 2 and 3 of the peeling decoder *once*. Let the number of removable edges in e^l be R^l and that in e^u be R^u. Due to symmetry, we only need to consider the change in the number of removable edges in e^u.

For a check node degree distribution C and a flow node degree distribution F, let

$$\varphi(C) = \sum_i (C_{i,2,2} + C_{i,2,1} + (i-1)C_{i,1,1} + iC_{i,1,0}),$$

$$\phi(F) = \frac{\sum_i (i-1)F_i}{(\sum_i F_i)^2},$$

we have

$$\begin{aligned}
&\mathbb{E}[R^u(1) - R^u(0)|\widehat{F}^l(0), \widehat{F}^u(0), \widehat{C}^l(0), \widehat{C}^u(0)] \\
= \;& \left[\varphi(\widehat{C}^u(0))\phi(\widehat{F}^u(0))\right]\left[\varphi(\widehat{C}^l(0))\phi(\widehat{F}^l(0))\right] - 1 \\
= \;& [\varphi(C^u(0))\phi(F^u(0))]\,[\varphi(C^l(0))\phi(F^l(0))] \\
& -1 + O\left(\frac{1}{n}\right) \\
= \;& \left[xu(-\rho'(1-u))\frac{y\lambda'(y)}{u^2}\right]\left[ux(-\rho'(1-x))\frac{\epsilon z\lambda'(z)}{x^2}\right] \\
& -1 + O\left(\frac{1}{n}\right) \qquad (11) \\
\le \;& -\delta + O\left(\frac{1}{n}\right)
\end{aligned}$$

where the first term in (11) is the gradient of the density evolution recursion at the intersection, which is equal to $1-\delta$.

Further, with $t \le n(\delta_1 + \delta_2)$, which is the maximum number of iterations the peeling decoder can run before stopping, with the set R^u decreasing at each iteration,

$$\begin{aligned}
&\mathbb{E}[R^u(t+1) - R^u(t)|\widehat{F}^l(t), \widehat{F}^u(t), \widehat{C}^l(t), \widehat{C}^u(t)] \\
= \;& \left[\varphi(\widehat{C}^u(t))\phi(\widehat{F}^u(t))\right]\left[\varphi(\widehat{C}^l(t))\phi(\widehat{F}^l(t))\right] - 1 \\
\le \;& -\delta + K(\delta_1 + \delta_2) \\
< \;& 0
\end{aligned}$$

since we can make T and n large enough so that $\delta_1 + \delta_2$ is small.

Using Wormald's differential equation approach (Theorem C.28, [10]), we obtain that when the peeling decoder stops,

$$\lim_{n\to\infty} \mathrm{P}_{(\lambda,\rho)}(t;n) > x - \delta_1 - \delta_2 > 0.$$

Since the output of the peeling decoder is equivalent to the message passing decoder with $l \to \infty$, we have shown Theorem 4. ■

V. Ensemble Optimization

A. Proof of Theorem 5

First, we define the ensemble $\mathcal{G}_{\text{P}}(n, \lambda, \gamma)$, where the n flow nodes have degree distribution λ, and the counter nodes have a Poisson degree distribution with mean γ. The number of counter nodes is hence $\Lambda'(1)/\gamma$, and the graph is formed by letting each edge select a counter node uniformly at random.

The desired rate is achieved by considering graphs from the ensemble $\mathcal{G}_{\text{P}}(n, \lambda, \gamma)$ with $\gamma = a/\epsilon$ and $\lambda(x) = x^{l(\epsilon)}$ (i.e. all variable nodes have regular degree $l(\epsilon) + 1$,) with $l(\epsilon) \equiv b\log(1/\epsilon)$. We will choose a and b independent of ϵ such that the density evolution recursion (6) yields $x_t \to 0$ as $t \to \infty$, thus proving the thesis with rate $\beta(\epsilon) = (b/a)\epsilon\log(1/\epsilon)$. The desired proportionality constant is obtained by optimizing the rate over a and b as expressed below

$$\kappa_* \equiv \min\left\{\, b/a \;:\; e^{-a} + e^{-1/b} = 1 \,\right\}. \qquad (12)$$

It is easy to see that the minimum is achieved at $a = (1/b) = \log 2$.

Now we show that a and b satisfying the condition in (12) yield $x_t \to 0$ as $t \to \infty$. It is convenient to rewrite the density evolution recursion in terms of the rescaled variable $y_t \equiv x_t/\epsilon$, and of $L = \log(1/\epsilon)$:

$$y_t \;=\; \begin{cases} (1 - e^{-ay_{t-1}})^{bL} & \text{if } t \text{ is odd,} \\ e^{-L}\,(1 - e^{-ay_{t-1}})^{bL} & \text{if } t \text{ is even.} \end{cases} \qquad (13)$$

with initial condition $y_0 = 1$. The condition $e^{-a} + e^{-1/b} = 1$ implies that $(1 - e^{-a})^{bL} = e^{-L}$, hence $y_1 = e^{-L}$.

For $t \ge 2$ we use the inequality $1 - e^{-x} \le x$ to get

$$y_t \le \begin{cases} (ay_{t-1})^{bL} & \text{if } t \text{ is odd,} \\ e^{-L}\,(ay_{t-1})^{bL} & \text{if } t \text{ is even.} \end{cases} \qquad (14)$$

Hence, for $t \ge 1$, and t odd, defining $\alpha = b\log(1/a)$, we get

$$y_{t+2} \le e^{-\alpha L + (1-\alpha)bL^2}\, y_t^{(bL)^2}. \qquad (15)$$

We get $\lim_{t\to\infty} y_t = 0$ if $(bL)^2 > 1$ and $e^{-\alpha L + (1-\alpha)bL^2}\, y_1^{(bL)^2} < y_1$. Since $y_1 = e^{-L}$, we get the two conditions

$$\begin{cases} bL > 1\,, \\ (bL)^2 - (1-\alpha)(bL) - (1-\alpha) > 0\,. \end{cases} \qquad (16)$$

Obviously both conditions are satisfied for large enough $L = \log(1/\epsilon)$, which proves the thesis for ϵ small enough. With the values a and b chosen above, ($a = 1/b = \log 2$), we get $\alpha = -\log\log(2)/\log(2) \approx 0.52877$. It is then easy to check that the second condition is implied by the first, which is in turn equivalent (for $b = 1/(\log 2)$) to $\epsilon < 1/2$. ■

B. Proof of Theorem 6

We make use of results from the optimization of LDPC code ensembles over the erasure channel [10]. The corresponding density evolution recursion reads

$$x_t = \tilde{\epsilon}\lambda(1 - \rho(1 - x_{t-1})).$$

Notice that this is exactly the same as the density evolution recursion for our message passing algorithm at even iterations. The following is a rephrasing of a result first proved in [12]:

Let β_n be the ratio of the number of check nodes to the number of variable nodes. There exists a sequence of degree distribution pairs (λ_k, ρ_k) with $\lim_{k\to\infty} \beta_k = \tilde{\epsilon}^*$ such that $x_t \to 0$ for all $\tilde{\epsilon} < \tilde{\epsilon}^*$.

In particular, along this sequence of graphs, $\lambda_k(1 - \rho_k(1 - x)) \to x/\tilde{\epsilon}^*$ uniformly for $x \in [0, 1]$ as $k \to \infty$.

Using this sequence of degree distribution pairs with $\tilde{\epsilon}^* = \sqrt{\epsilon}$, we have

$$\begin{aligned}
f_k(x) &= \epsilon\lambda_k(1 - \rho_k(1 - \lambda_k(1 - \rho_k(1 - x)))) \\
&= \sqrt{\epsilon}\lambda_k(1 - \rho_k(1 - x)) + o_k(1) \\
&= x + o_k(1)
\end{aligned}$$

uniformly for $x \in [0, 1]$ as $k \to \infty$. Since $f_k(x)$ is strictly increasing in ϵ, for any $\epsilon' < \epsilon$ we can take k large enough so that $f_k(x) < x$ for all $x \in [0, 1]$. Therefore ϵ is an lower bound of the asymptotic threshold $\epsilon^*(\lambda_k, \rho_k)$ for this sequence where the corresponding d-rate $\beta_k \to \tilde{\epsilon}^* = \sqrt{\epsilon}$. ■

VI. Conclusion

We analyzed the achievable dimensionality reduction rate for a single-layer Counter Braids and found it to be very close to Donoho and Tanner threshold for non-negative signals with the L_1 minimization recovery algorithm. Since the complexity of message-passing algorithm is essentially linear, it is a more efficient solution to non-negative sparse signal recovery than L_1 minimization.

Future work includes determining lower bounds on the dimensionality reduction rate under message passing decoding. More importantly, the analysis of dimensionality reduction rate will be a step towards analyzing the *compression rate* of Counter Braids with the message passing algorithm.

VII. Acknowledgments

Yi Lu is supported by the Cisco Stanford Graduate Fellowship. Andrea Montanari is partially supported by a Terman fellowship and the NSF CAREER grant 0743978.

References

[1] D. Shah, S. Iyer, B. Prabhakar, and N. McKeown, "Analysis of a statistics counter architecture," in *Hot Interconnects 9*, Stanford, August 2001.

[2] C. Estan and G. Varghese, "New directions in traffic measurement and accounting," *Proceedings of ACM SIGCOMM*, 2002.

[3] Y. Lu, A. Montanari, B. Prabhakar, S. Dharmapurikar, and A. Kabbani, "Counter braids: A novel counter architecture for per-flow measurement," in *SIGMETRICS*, 2008.

[4] Y. Lu, A. Montanari, and B. Prabhakar, "Detailed network measurements using sparse graph counters: The theory," Allerton, 2007.

[5] E. Candès and T. Tao, "Near optimal signal recovery from random projections and universal encoding strategies," *IEEE Trans. Inform. Theory*, 2004.

[6] D. L. Donoho, "Compressed Sensing," *IEEE Trans. on Inform. Theory*, vol. 52, pp. 1289– 1306, 2006.

[7] D. L. Donoho and J. Tanner, "Sparse nonnegative solutions of underdetermined linear equations by linear programming," *Proc. Natl. Acad. Sci. USA*, vol. 102, no. 27, pp. 9446–9451, 2005.

[8] R. Berinde and P. Indyk, "Sparse recovery using sparse random matrices," *Harvard Computer Science*, pp. 326–327, 1999.

[9] W. Xu and B. Hassibi, "Efficient compressive sensing with deterministic guarantees using expander graphs," in *ITW*, 2007.

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

[11] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proceedings of the 29th annual ACM Symposium on Theory of Computing*, 1997, pp. 150–159.

[12] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.