Graph-based codes for hierarchical recovery

Allison Beemer
Dept. of Mathematics
University of Wisconsin-Eau Claire
Eau Claire, WI, U.S.A.
beemera@uwec.edu

Rutuja Kshirsagar
Dept. of Mathematics
Virginia Tech
Blacksburg, VA, U.S.A.
rutujak@vt.edu

Gretchen L. Matthews*
Dept. of Mathematics
Virginia Tech
Blacksburg, VA, U.S.A.
gmatthews@vt.edu

***Abstract*—In this paper, we consider approaches to designing Tanner codes to protect against symbol loss from multiple erasures. First, we note that Tanner codes inherit locality and availability from their inner codes, allowing one to design longer codes with specified locality and availability. Availability is desirable in that multiple disjoint repair groups increase the likelihood that symbols are available to repair erased ones. Even so, particular patterns of erasures well-distributed across the repair groups may prevent recovery. Hence, we consider an alternative using hierarchical locality which implements tiered recovery, where the tier utilized depends on the number of erasures. Finally, we define hierarchical stopping sets to characterize local message-passing decoder failure at the various repair levels.**

***Index Terms*—locality, hierarchical locality, Tanner codes, message-passing decoder, stopping sets**

I. Introduction

A locally recoverable code (LRC) provides erasure recovery of a symbol by accessing only a few other symbols, so that each coordinate has a specified recovery set. If multiple erasures occur, an LRC may or may not be able to recover the original codeword using these small recovery sets. Assuming the code has minimum distance d, it can recover up to $d-1$ symbols but in doing so utilizes all available symbols. For instance, an LRC with locality r would only require r symbols to recover a missing coordinate but if two symbols were missing then all $n-2$ symbols may need to be involved in the recovery. Hierarchical LRCs (HLRCs), introduced in [1], [2], address this problem by providing tiers of recovery. If a single erasure occurs, then a small number of symbols are involved in recovery; if more, but not too many, erasures occur, then more (but not all available) symbols are utilized.

In this paper, we harness graphical properties and inner codes of Tanner codes to give rise to hierarchical LRCs. Clearly, any existing LRC can be expressed as a graph-based code; in contrast, here we begin with a graph to construct an (H)LRC. In graph-based message-passing decoding, stopping sets characterize decoder failure over erasure channels [3]. As LRCs are designed for the erasure setting, we can express message-passing decoder failure for our HLRCs in each level of the hierarchy as specialized stopping sets within that level. We show that the minimum size of a stopping set increases with the size of the hierarchical repair level.

*The work of G.L. Matthews is supported in part by NSF under Grant DMS-1855136 and in part by the Commonwealth Cyber Initiative.

The remainder of the paper is organized as follows. Section II gives necessary background on Tanner codes and (H)LRCs. We describe how Tanner codes can be viewed as (H)LRCs and present locality parameters for each in Section III. Section IV explores the connection between stopping sets of graph-based message-passing decoders and hierarchical local recovery, and Section V concludes the paper.

II. Preliminaries

Tanner codes are graph-based codes constructed using bipartite graphs and shorter codes called inner codes [4]. In this section, we review prerequisite material before considering the use of Tanner codes in constructing (H)LRCs.

We use standard notation and terminology, letting an $[n,k,d]_q$ code over a finite field $\mathbb{F}_q$ (denoted $[n,k,d]$ when the field is understood) be a linear code of length n, dimension k, and minimum distance d with respect to the Hamming metric. We define $[n] := \{1,\ldots,n\}$. Given $\mathbf{v} \in \mathbb{F}_q^n$ and $i \in [n]$, we denote the i^{th} component of $\mathbf{v}$ by v_i.

Given an $[n,k,d]$ code $\mathcal{C}$ and $I := \{i_1,\ldots,i_s\} \subseteq [n]$, the corresponding *punctured code* is $\mathcal{C}|_I := \{(c_{i_1},\ldots,c_{i_s}) : \mathbf{c} = (c_1,\ldots,c_n) \in \mathcal{C}\}$. Let $\mathcal{C}_1$ be an $[n_1,k_1,d_1]$ code and $\mathcal{C}_2$ an $[n_2,k_2,d_2]$ code. Then the *concatenated code* $\mathcal{C}$ is comprised of codewords obtained by placing the codewords of $\mathcal{C}_1$ and $\mathcal{C}_2$ adjacent to each other in order. Note that the length of the concatenated code $\mathcal{C}$ is n_1+n_2, the dimension of $\mathcal{C}$, denoted dim($\mathcal{C}$), is at most k_1+k_2, and the minimum distance of $\mathcal{C}$, denoted $d_{\min}(\mathcal{C})$, is $\min\{d_1,d_2\}$.

Definition 1 (Tanner Codes)**.** Let $G = (L\dot{\cup}R, E)$ be a bipartite graph where the nodes in L have degrees $\mathfrak{C} := \{\mathfrak{c}_1,\ldots,\mathfrak{c}_n\}$ such that $|L| = n$ and nodes in R have degrees $\mathfrak{D} := \{\mathfrak{d}_1,\ldots,\mathfrak{d}_m\}$ such that $|R| = m$. The *Tanner code* $\mathcal{T}(G,\{\mathcal{C}_1,\ldots,\mathcal{C}_m\})$ over $\mathbb{F}_q$ is defined as follows:

$$\mathcal{T}(G,\{\mathcal{C}_1,\ldots,\mathcal{C}_m\}) = \{\mathbf{c} : \mathbf{c}|_{N(j)} \in \mathcal{C}_j \ \forall j \in [m]\} \subseteq \mathbb{F}_q^n,$$

where the *neighborhood* $N(j)$ is the set of nodes adjacent to the jth node in R.

We assume that $\mathfrak{c}_1 \le \mathfrak{c}_2 \le \ldots \le \mathfrak{c}_n$ and $\mathfrak{d}_1 \le \mathfrak{d}_2 \le \ldots \le \mathfrak{d}_m$. Moreover, $\mathcal{C}_j \subseteq \mathbb{F}_q^{\mathfrak{d}_j}$ for all $j \in [m]$. It is possible that for some $j,k \in [m]$, $j \ne k$, $\mathfrak{d}_j = \mathfrak{d}_k$ but $\mathcal{C}_j \ne \mathcal{C}_k$.

Throughout the paper, we will refer to nodes in L as *variable nodes* and nodes in R as *check nodes*. The properties of $\mathcal{T}(G,\{\mathcal{C}_1,\ldots,\mathcal{C}_m\})$ depend both on the underlying graph and the choice of *inner codes* $\mathcal{C}_1, \ldots, \mathcal{C}_m$. Recall that the *girth* of a graph G, denoted $g(G)$, is the minimum length of a cycle contained within the graph. Because a Tanner code is derived from a bipartite graph, its girth must be even.

The notion of locality was first introduced in [5]. *Locally recoverable codes (LRCs)* allow recovery of any erased codeword symbol (meaning coordinate) by accessing at most r other surviving codeword symbols. More precisely, an (n, k, r) LRC $\mathcal{C}$ over $\mathbb{F}_q$ encodes a message with k symbols to generate a codeword $\mathbf{c}$ of length n such that each symbol c_i of the codeword can be recovered using a set of at most r other codeword symbols. This set is called a *recovery set* of c_i, and r is the *locality* of $\mathcal{C}$. Note that we may assume $r \leq k$. If there are at least a disjoint recovery sets for each $i \in [n]$, each containing at most r symbols, the code is said to have *availability* a and is denoted as an (n, k, r, a) LRC. The idea of locality has been widely explored in the literature, see e.g. [6], [7], [8], [9].

Kamath et al. extended the definition of LRCs to (t, ρ) locality in [10]. This extension allows for the recovery of up to $\rho - 1$ erasures, as described next via the notion of punctured codes. An $[n, k, d]$ code $\mathcal{C}$ is an LRC with locality (t, ρ) and availability τ if:

- there exist punctured codes $\mathcal{C}|_{I_{1,i} \cup \{i\}}, \ldots, \mathcal{C}|_{I_{\tau,i} \cup \{i\}}$ such that $I_{1,i}, \ldots, I_{\tau,i} \subseteq [n] \setminus \{i\}\ \forall i \in [n]$,
- $\mathcal{C}|_{I_{j,i} \cup \{i\}}$ is a $[|I_{j,i}| + 1, \leq t, \geq \rho]$ code for all $j \in [\tau]$,
- the set $\text{supp}(\mathcal{C}|_{I_{j,i} \cup \{i\}}) \setminus \left[\cup_{\ell \in [\tau], \ell \neq j} \text{supp}(\mathcal{C}|_{I_{\ell,i} \cup \{i\}})\right]$ has $\dim(\mathcal{C}|_{I_{j,i} \cup \{i\}})$ linearly independent coordinates.

Note that the values of t and ρ can in general differ for each code $\mathcal{C}|_{I_{j,i} \cup \{i\}}$ for all $j \in [\tau]$. However, throughout this paper we will consider each punctured code to have the same bounding values (t, ρ). Various approaches to correcting multiple erasures using locality have been studied in the literature, see e.g. [11], [12], [13]. One such approach, which we focus on here, was the introduction of *hierarchical local recovery* in [1], [2]. Hierarchical locally recoverable codes (HLRCs) are linear codes that provide a method of multi-tier erasure recovery; they are defined as follows.

Definition 2 (HLRCs, [1])**.** An $[n, k, d]$ code $\mathcal{C}$ is a code with *h-level hierarchical locality having local parameters* $[(t_1, \rho_1), \ldots, (t_h, \rho_h)]$ if $\rho_1 \geq \cdots \geq \rho_h$, and for every $i \in [n]$, there exists a punctured recovery code $\mathcal{C}_i$ such that $i \in \text{supp}(\mathcal{C}_i)$ and the following conditions hold:

- $\dim(\mathcal{C}_i) \leq t_1$,
- $d_{\min}(\mathcal{C}_i) \geq \rho_1$,
- $\mathcal{C}_i$ is a code with $(h-1)$-level hierarchical locality having local parameters $[(t_2, \rho_2), \ldots, (t_h, \rho_h)]$.

For an HLRC as in Definition 2, we refer to the recovery codes $\mathcal{C}_i$ with parameters (t_j, ρ_j) as belonging to the *j^{th} level* of hierarchy. Up to $\rho_j - 1$ erasures can be corrected using the j^{th} level of hierarchy based on the minimum distance. Notice that to fully exploit the hierarchical structure, we should choose the highest index of a level such that correction is possible with a code in that level (thus minimizing the dimension of the recovery code); as a result, level j may be used to correct between ρ_{j+1} and $\rho_j - 1$ erasures. The notion of availability can be extended to HLRCs as follows.

Definition 3 (HLRCs with availability)**.** An $[n, k, d]$ code $\mathcal{C}$ is code with a h-level hierarchical locality having local parameters $[(t_1, \rho_1), \ldots, (t_h, \rho_h)]$ and *availability* $\tau_1, \ldots, \tau_h$ if the following conditions hold.

- $\mathcal{C}$ is an (t_1, ρ_1) LRC with availability τ_1.
- Each of the punctured codes $\mathcal{C}|_{I_{j_1,i} \cup \{i\}}\ \forall j_1 \in [\tau_1]$ is a $(h-1)$-level HLRC having local parameters $[(t_2, \rho_2), \ldots, (t_h, \rho_h)]$ and availability $\tau_2, \ldots, \tau_h$.

In the next section, we explore the (hierarchical) locality and availability properties of Tanner codes when the inner codes are chosen to have locality properties of their own.

III. Tanner codes for (hierarchical) recovery

In this section, we show how Tanner codes can be considered as (H)LRCs when the inner codes are chosen to be locally recoverable codes. In both subsections, we consider a Tanner code $\mathcal{T} := \mathcal{T}(G, \{\mathcal{C}_1, \ldots, \mathcal{C}_m\})$ over the finite field $\mathbb{F}_q$. Let ℓ_1 be the number of distinct elements of $\mathfrak{C}$, and let $x_{ij} \in L$ denote a node of degree $\mathfrak{c}_i$, $i \in [\ell_1]$, so that $j \in [n_i]$, where n_i is the number of nodes of degree $\mathfrak{c}_i$. Denote the neighborhood of x_{ij} by $N(S) = \{y_{ij1}, \ldots, y_{ij\mathfrak{c}_i}\}$, and let the code associated with $y_{ij\ell}$ be denoted $\mathcal{C}_{ij\ell}$ for all $1 \leq \ell \leq \mathfrak{c}_i$. On the other hand, let m_i denote the number of distinct check nodes in R of degree $\mathfrak{d}_i$. Notice that it is possible that $y_{ij\ell} = y_{i'j'\ell'}$ for some $(i, j, \ell) \neq (i', j', \ell')$; it is of course important that in this case, $\mathcal{C}_{ij\ell} = \mathcal{C}_{i'j'\ell'}$.

A. Tanner codes as LRCs

We begin the study of Tanner codes as LRCs with repair sets by considering locality properties with a single erasure.

Theorem 1. *For each $x_{ij} \in L$, let $\mathcal{C}_{ij\ell}$ be an $(d_{ij\ell}, k_{ij\ell}, r_{ij\ell})$ LRC with availability $a_{ij\ell}$ for all $1 \leq \ell \leq \mathfrak{c}_i$.*

1) *If $g(G) = 4$, $\mathcal{T}$ has locality r and availability a given by*

$$r = \max_{(i,j)} \min_{\substack{\ell \ s.t. \\ y_{ij\ell} \in N(x_{ij})}} \{r_{ij\ell}\}$$

$$a = \min_{(i,j)} \max_{\substack{\ell \ s.t. \\ y_{ij\ell} \in N(x_{ij}),\ r_{ij\ell} \leq r}} \{a_{ij\ell}\}.$$

2) *If $g(G) \geq 6$, $\mathcal{T}$ has locality r and availability a given by*

$$r = \max_{(i,j,\ell)} \{r_{ij\ell}\}$$

$$a = \min_{(i,j)} \sum_{\ell=1}^{\mathfrak{c}_i} a_{ij\ell}.$$

Proof. 1) Let $g(G) = 4$ and consider an erased node x_{ij} for some $i \in [\ell_1]$ and $j \in [n_i]$. The most efficient way to recover x_{ij} is to access the neighbor $y_{ij\ell}$ such that the associated inner code $\mathcal{C}_{ij\ell}$ has minimum locality: $\min_{\ell \ s.t.\ y_{ij\ell} \in N(x_{ij})} \{r_{ij\ell}\}$. To ensure that any choice of erased node can be recovered, we maximize over all possible i, j pairs, yielding the result.

The number of disjoint sets of size at most r that can then recover the erased node x_{ij} is $\max_{\ell \text{ s.t. } y_{ij\ell} \in N(x_{ij}),\ r_{ij\ell} \leq r} \{a_{ij\ell}\}$. The availability must apply to the recovery of any erasure, so we minimize over all possible i, j pairs, yielding the result.

2) Now, let $g(G) \geq 6$. Since there are no 4-cycles in G, none of the repair groups of x_{ij} associated with different $y_{ij\ell}$'s intersect. Thus, we may increase availability significantly by taking the locality r to be a maximum over the neighboring $r_{ij\ell}$'s. Consider the erased node x_{ij} and its neighboring set $N(S) = \{y_{ij1}, \ldots, y_{ij\mathfrak{c}_i}\}$. Again due to the lack of 4-cycles, x_{ij} can now be repaired using $\sum_{\ell=1}^{\mathfrak{c}_i} a_{ij\ell}$ disjoint repair groups of size at most r. We again minimize over all i, j pairs, yielding the result. ■

We now extend the above results to the case of multiple erasures, so that the set of erased nodes is given by $S = \{x_{i_1 j_1}, \ldots, x_{i_s j_s}\}$, where $|S| = s \leq \mathfrak{c}_1$ and any pair of erased nodes shares at most one common neighbor in R. If the girth of $\mathcal{T}$ is at least 6, then this last condition is guaranteed; in graphs of girth 4 it remains possible to have such a set S, though not every erasure set of size $s \geq 2$ will have this property. The neighborhood of S is equal to $N(S) = \bigcup_{\ell=1}^{s} N(x_{i_\ell j_\ell})$, a union that is not necessarily disjoint.

Theorem 2. *For each $x_{ij} \in L$, let $\mathcal{C}_{ij\ell}$ be an $(d_{ij\ell}, k_{ij\ell}, r_{ij\ell})$ LRC with availability $a_{ij\ell}$ for all $1 \leq \ell \leq \mathfrak{c}_i$. Then $\mathcal{T}$ has locality equal to*

$$r = s \max_{(i,j,\ell)} \{r_{ij\ell}\}.$$

Furthermore,

1) If $g(G) = 4$, $\mathcal{T}$ has availability given by

$$a = \min_{(i,j,\ell)} \{a_{ij\ell}\}.$$

2) If $g(G) \geq 6$, $\mathcal{T}$ has availability given by

$$a = \min_{(i,j,\ell)} \left\{(\mathfrak{c}_i - s + 1) a_{ij\ell}\right\}.$$

Proof. Consider the set of erased nodes S such that $|S| = s \leq \mathfrak{c}_1$ and the intersection of the neighborhoods of any two elements in S has size at most one. The most efficient way to recover x_{ij} would be to access its neighbor $y_{ij\ell}$ such that the associated inner code $\mathcal{C}_{ij\ell}$ has minimum locality. Unfortunately, there is no guarantee that another node in this repair group does not also belong to S. However, there are at least $\mathfrak{c}_i - (s-1) \geq 1$ neighbors of x_{ij} that have no other elements of S in its neighborhood. Thus, at most $\max_{\ell \text{ s.t. } y_{ij\ell} \in N(x_{ij})} \{r_{ij\ell}\}$ other nodes need to be contacted to repair x_{ij}. To ensure that every erased node can be recovered, we maximize this over all $i \in [\ell_1]$ and $j \in [n_i]$.

1) Let $g(G) = 4$. The number of disjoint sets that can recover a single erased node x_{ij} for some $i \in [\ell_1]$ and $j \in [n_i]$ is at least $\min_\ell \{a_{ij\ell}\}$. The collection of s erasures is then recovered using s (not necessarily disjoint) recovery sets, each of size at most $\max_{(i,j)} \max_{\ell \text{ s.t. } y_{ij\ell} \in N(x_{ij})} \{r_{ij\ell}\}$. By taking unions comprised of one repair group per bit, we will have at least $a = \min_{(i,j,\ell)} \{a_{ij\ell}\}$ repair groups of size at most $r = s \max_{(i,j,\ell)} \{r_{ij\ell}\}$ for the set S.

2) Now, let $g(G) \geq 6$, and consider the erased node x_{ij} for some $i \in [\ell_1]$ and $j \in [n_i]$ and its neighboring set $N(x_{ij}) = \{y_{ij1}, \ldots, y_{ij\mathfrak{c}_i}\}$. Since G has no 4-cycles, none of the repair groups of x_{ij} associated with $y_{ij\ell}$ intersect with the repair groups of $y_{ij\ell'}$, for all $\ell \neq \ell' \in [\mathfrak{c}_i]$. Recall that at least $\mathfrak{c}_i - s + 1$ of x_{ij}'s neighbors are not adjacent to any other element of S. This means that x_{ij} can be repaired using at least $(\mathfrak{c}_i - s + 1) \min_\ell \{a_{ij\ell}\}$ different repair groups of size at most $\max_{(i,j,\ell)} \{r_{ij\ell}\}$. As before, by taking unions comprised of one repair group per bit, we will have at least $a = \min_{(i,j,\ell)} (\mathfrak{c}_i - s + 1) \{a_{ij\ell}\}$ repair groups of size at most $s \max_{(i,j,\ell)} \{r_{ij\ell}\}$ for the set S. ■

From the previous two results, we can see that in the case of s erasures, the best that can be guaranteed by exploiting the inner code LRC structure is that the size of a repair set for the erasures increases a full s-fold from the size of a repair group for a single erasure in a girth ≥ 6 graph. This motivates the study of Tanner codes as HLRCs for the case of multiple erasures.

B. Tanner codes as HLRCs

Next, we show that Tanner codes may be viewed as HLRCs with $h = 2$ or 3 levels by presenting results on the parameters of h particular choices of recovery code level construction (see the proofs of Theorems 3 and 4, respectively).

Theorem 3. *Let the inner code $\mathcal{C}_i := (d_i, k_i, \delta_i)$ of Tanner code $\mathcal{T}$ be an LRC with locality parameters (t_i, ρ_i) and availability τ_i for all $i \in [m]$. Then $\mathcal{T}$ is an HLRC with 3-level hierarchical locality with locality parameters $[(\tilde{t}_1, \tilde{\rho}_1), (\tilde{t}_2, \tilde{\rho}_2), (\tilde{t}_3, \tilde{\rho}_3)]$, where*

$$(\tilde{t}_1, \tilde{\rho}_1) = \left(\max_{(i,j)} \sum_{\ell=1}^{\mathfrak{c}_i} k_{ij\ell}, \min_{u \in [m]} \delta_u \right),$$

$$(\tilde{t}_2, \tilde{\rho}_2) = \left(\max_{u \in [m]} k_u, \min_{u \in [m]} \delta_u \right),$$

$$(\tilde{t}_3, \tilde{\rho}_3) = \left(\max_{u \in [m]} t_u, \min_{u \in [m]} \rho_u \right).$$

Furthermore, the HLRC has availability $\tilde{\tau}_1 = 1, \tilde{\tau}_2 = \min_j \mathfrak{c}_j, \tilde{\tau}_3 = \min_{u \in [m]} \tau_u$ if $g(G) \geq 6$ and availability $\tilde{\tau}_1 = 1, \tilde{\tau}_2 = 1, \tilde{\tau}_3 = \min_{u \in [m]} \tau_u$ if $g(G) = 4$.

Proof. Let x_{ij} denote an erased node. Consider the code comprised of x_{ij} and all variable nodes distance 2 from x_{ij} (i.e. all neighbors of neighbors of x_{ij}); this is a punctured code relative to $\mathcal{T}$. The dimension of this concatenated code is bounded above by $\sum_{\ell=1}^{\mathfrak{c}_i} k_{ij\ell}$ and its minimum distance is equal to $\min_{\ell\in[\mathfrak{c}_i]} \delta_{ij\ell}$. The result for the 1^{st} level of hierarchy follows from maximizing the upper bound and minimizing the lower bound on minimum distance over all elements of L. For the 2^{nd} level of hierarchy, consider the set of inner codes associated with neighbors of erased node x_{ij}. Each of these inner codes is a punctured code obtained from the 1^{st} level concatenated code associated with x_{ij}, the dimension of each is bounded above by $\max_{u\in[m]} k_u$, and the minimum distance of each is bounded below by $\min_{u\in[m]} \delta_u$. The 3^{rd} level of hierarchy is associated with the locality of each of the inner codes. In particular, the recovery codes of each inner code are the punctured codes obtained from the inner code. The dimension of each of these punctured codes is at most $\max_{u\in[m]} t_u$ and the minimum distance is at least $\min_{u\in[m]} \rho_u$. Regardless of $g(G)$, the availability at the 1^{st} level of hierarchy is 1 because a single repair code at this level consists of all (distance 2) variable node neighbors of an erased node. When $g(G) \geq 6$, availability at the 2^{nd} level of hierarchy is $\min_j \mathfrak{c}_j$, since any inner code associated with any neighbor of an erased node can be used for recovery and the neighborhoods of the check node neighbors of an erased node are pairwise disjoint. The availability at the 3^{rd} level of hierarchy in this case is $\tilde{\tau}_3 = \min_{u\in[m]} \tau_u$ because we may only look at repair codes that exist within a particular 2^{nd} level repair code per Definition 3. Thus, the availability is given by the minimum availability of an inner code LRC. Now, suppose $g(G) = 4$. The availability at the 2^{nd} level of hierarchy is 1 because the inner code associated with any neighbor of an erased node can be used for recovery, but we can no longer guarantee that the neighborhoods of these check nodes are disjoint. The availability at the 3^{rd} level of hierarchy is again $\min_{u\in[m]} \tau_u$. ■

We can also consider $\mathcal{T}$ as an HLRC with two levels of repair codes, as described in the next theorem. The two level recovery offers an advantage in terms of availability over the three level recovery above given $g(G) \geq 6$. The availability in the 2^{nd} level (3^{rd} level of three tier recovery) is larger due to the larger punctured codes (the 1^{st} level) containing the 2^{nd} level. This means that we can simultaneously access the locality of any inner code in the neighborhood of the erased symbol. It is always ideal to eliminate small cycles in graph-based codes and hence the improvement in availability for Tanner codes based on graphs with $g(G) \geq 6$ is meaningful.

Theorem 4. *Let the inner code $\mathcal{C}_i := (d_i, k_i, \delta_i)$ of Tanner code $\mathcal{T}$ be an LRC with locality parameters (t_i, ρ_i) and availability τ_i for all $i \in [m]$. Then $\mathcal{T}$ is an HLRC with 2-level hierarchical locality with locality parameters $[(\tilde{t}_1, \tilde{\rho}_1), (\tilde{t}_2, \tilde{\rho}_2)]$, where*

$$(\tilde{t}_1, \tilde{\rho}_1) = \left(\max_{\ell\in[m]} \sum_{\ell=1}^{\mathfrak{c}_i} k_{ij\ell}, \min_{u\in[m]} \delta_u \right)$$

$$(\tilde{t}_2, \tilde{\rho}_2) = \left(\max_{u\in[m]} t_u, \min_{u\in[m]} \rho_u \right)$$

Furthermore, the HLRC has availability $\tilde{\tau}_1 = 1, \tilde{\tau}_2 = \min_{(i,j)} \sum_{\ell=1}^{\mathfrak{c}_i} \tau_{ij\ell}$ if $g(G) \geq 6$ and availability $\tilde{\tau}_1 = 1, \tilde{\tau}_2 = \min_{u\in[m]} \tau_u$ if $g(G) = 4$.

Proof. The proof follows from the proof of Theorem 3 by letting the 1^{st} level of that result be the 1^{st} level of this result, and the 3^{rd} level of that result be the 2^{nd} level of this result. Notice that the availability at the previous 3^{rd} level, now 2^{nd} level, has increased for the case $g(G) \geq 6$. This is because now we may take all inner LRC repair codes within the entire concatenated 1^{st} level code, and are not limited to a single inner code. In the case where $g(G) \geq 6$, the neighborhoods of the check node neighbors of an erased node are pairwise disjoint and each recovery code of each inner code LRC can be considered. ■

IV. Stopping Sets & Local Recovery

A significant advantage of giving graph-based codes an (H)LRC structure is the opportunity to implement graph-based message-passing decoding algorithms, which have low implementation complexity when operating on sparse graphs. Over an erasure channel, a so-called *peeling decoder*, which iteratively "peels off" erasures by contacting neighboring check nodes, is used [14]. *Stopping sets* of Tanner codes where each inner code is a simple parity-check (e.g. in the case of low-density parity-check codes) are patterns that cause iterative decoder failure over an erasure channel [3]. Formally, a stopping set is a subset S of variable nodes such that every check node adjacent to S is adjacent to at least two elements of S; since every word of weight two is a codeword in a parity-check code, stopping sets completely characterize erasure patterns where a peeling decoder will get stuck. In the case where each check node acts as its own (nontrivial) inner code on a subset of variable nodes (i.e. Tanner codes), the definition of stopping sets generalizes. Several previous works take the generalization of "two" to be $d_{\min}(\mathcal{C})$, where $\mathcal{C}$ is an inner code [15]. However, it is not necessarily the case that every word of weight $d_{\min}(\mathcal{C})$ is a codeword, and thus some such erasure patterns remain correctable with this definition. Hence for our application we adopt the following:

Definition 4. Consider a Tanner code with graph $G = (L\dot{\cup}R, E)$. A *generalized stopping set* is a nonempty subset $S \subseteq L$ of variable nodes such that the support of the variable nodes in S adjacent to any check node i contains the support of a codeword of $\mathcal{C}_i$, the inner code at check node i.

Notice that a codeword whose values are erased at a generalized stopping set could be corrected by each adjacent inner code in at least two distinct ways; hence, the peeling decoder will be stuck. Conversely, if the peeling decoder gets stuck, the erased set must form a generalized stopping set. In the case of our 3-level graph-based HLRCs, we define hierarchical stopping sets in order to characterize message-passing decoder failure at each level.

Definition 5. 1) A 3^{rd} *level stopping set* is a nonempty subset S of variable nodes that is contained within a single inner code $\mathcal{C}_i$ such that the restriction to each repair code of the LRC $\mathcal{C}_i$ contains the support of a codeword of that repair code.

2) A 2^{nd} *level stopping set* is a nonempty subset S of variable nodes contained within the neighborhood of a single check node such that S contains the support of a codeword of the associated inner code (i.e. S is a generalized stopping set within a single inner code of the 2^{nd} level of the HLRC).

3) A 1^{st} *level stopping set* is a nonempty subset S of variable nodes contained within a concatenated code associated with the 1^{st} level of the HLRC such that the intersection of S with the neighborhood of each check node of the concatenated code contains the support of a codeword of the associated inner code (i.e. S is a generalized stopping set within a single concatenated code of the 1^{st} level of the HLRC).

One important parameter of a given graphical representation G of a code $\mathcal{C}$ is the minimum size of a stopping set, $s_{\min}(G)$: fewer than $s_{\min}(G)$ erasures are guaranteed to be correctable by the peeling decoder. This value is referred to as the *stopping distance* or *stopping number* of the representation [16]–[18]. Considering that the support of any codeword of $\mathcal{C}$ must form a stopping set in any representation, we observe that $s_{\min}(G) \leq d_{\min}(\mathcal{C})$. Importantly, the inequality may be strict. The fact that a peeling decoder is being used is paramount: stopping sets indicate where iterative decoders fail, not necessarily where any decoder would fail. This is comparable to the failure of local recovery not necessarily implying failure of global recovery in an LRC.

Theorem 5. *Let $s_{\min}(G_j)$ denote the minimum size of a j^{th} level stopping set for $j \in [3]$, and let ρ_i denote the minimum distance of the LRC inner code $\mathcal{C}_i$. Then,*

$$\begin{aligned} \min_i \rho_i \leq s_{\min}(G_3) &\leq s_{\min}(G_2) \\ &= \min_i d_{\min}(\mathcal{C}_i) \leq s_{\min}(G_1) \end{aligned}$$

Proof. Any 2^{nd} level stopping set associated with inner code $\mathcal{C}_i$ has weight lower bounded by $d_{\min}(\mathcal{C}_i)$, and a minimum weight codeword of $\mathcal{C}_i$ is a 2^{nd} level stopping set. In other words, $s_{\min}(G_2) = \min_i d_{\min}(\mathcal{C}_i)$. Consider a minimum 2^{nd} level stopping set S; by the above argument, $|S| = \min_i d_{\min}(\mathcal{C}_i)$. Let j be a minimizer, so that S is contained in the neighborhood of $\mathcal{C}_j$ and $|S| = d_{\min}(\mathcal{C}_j)$. By definition, S forms the support of a codeword of LRC $\mathcal{C}_j$, so the intersection of S with any repair code of C_j must either be empty or must contain at least ρ_j elements by the definition of the minimum distance of the repair code. Thus, S forms a 3^{rd} level stopping set, and $s_{\min}(G_3) \leq |S| = s_{\min}(G_2)$. For the lower bound on $s_{\min}(G_3)$, notice that the intersection of any 3^{rd} level stopping set associated with LRC inner code $\mathcal{C}_j$ with each repair code of $\mathcal{C}_j$ must be a codeword of the repair code. Thus, the minimum distance of a repair code of $\mathcal{C}_j$, ρ_j, gives a lower bound on the stopping set size. We then minimize over all inner codes. Finally, consider a minimum 1^{st} level stopping set S, and the intersection of S with the neighborhood of some check node y_j that is adjacent to a vertex in S. Call this subset S'. By definition of a 1^{st} level stopping set, it must be the case that S' contains the support of a codeword of $\mathcal{C}_j$, where $\mathcal{C}_j$ the inner code associated with y_j; in other words, S' is a 2^{nd} level stopping set. Then, $s_{\min}(G_2) \leq |S'| \leq |S| = s_{\min}(G_1)$. ■

Remark 1. Notice that $d_{\min}(\mathcal{C})$, where $\mathcal{C}$ is the code defined by the entire Tanner graph, gives an upper bound on $s_{\min}(G_1)$ since the support of any codeword of $\mathcal{C}$ restricted to any concatenated code (such that the result is nonempty) must result in codewords at each of the constituent inner codes.

Remark 2. A natural question is when the inequalities of Theorem 5 are met with equality. First observe that the collection of 2^{nd} level stopping sets is contained in the collection of 3^{rd} level stopping sets: any codeword of an inner code $\mathcal{C}_i$ satisfies each of the repair groups of $\mathcal{C}_i$. Strict set inclusion occurs exactly when the repair codes do not completely define $\mathcal{C}_i$ (i.e. when more checks beyond the repair code checks are needed). If the repair codes do define the code, we have $s_{\min}(G_3) = s_{\min}(G_2)$; otherwise the inequality may be strict. Next, consider a 1^{st} level stopping set S of size $s_{\min}(G_1)$, and suppose $s_{\min}(G_1) = \min_i d_{\min}(\mathcal{C}_i)$. Then the intersection of S with the neighborhood of each constituent check node of its associated concatenated code must be equal to S. As long as $\mathfrak{c}_1 > 1$, this can only occur if $\min_i d_{\min}(\mathcal{C}_i) = 1$ or $g(G) = 4$. Otherwise, $s_{\min}(G_2) < s_{\min}(G_1)$.

Recall that we iteratively correct erasures in an HLRC using increasingly larger repair codes within the graph, i.e. by decreasing the level index. Increasing the size of repair code needed for correction is detrimental in terms of locality, but may be necessary if a higher number of erasures must be corrected. Theorem 5 and Remark 2 imply that decreasing the level index of an HLRC from a Tanner code does not decrease the number of correctable erasures, and in fact can give an erasure correction advantage in many cases, even when restricting to a message-passing peeling decoder.

V. Conclusion

In this paper, we study local erasure recovery in Tanner codes. We provide bounds on the locality and availability of Tanner codes when the inner codes are locally recoverable codes. Moreover, we show that in this setting the Tanner codes allow hierarchical local recovery. This analysis results in insights about the behaviour of tiered stopping sets of Tanner code HLRCs.

We are currently comparing our bounds on the locality and availability of Tanner codes where the underlying inner codes are LRCs with bounds on the locality and availability of existing (H)LRC constructions.

References

[1] B. Sasidharan, G. K. Agarwal, and P. V. Kumar, “Codes with hierarchical locality,” vol. abs/1501.06683, 2015. [Online]. Available: http://arxiv.org/abs/1501.06683

[2] ——, “Codes with hierarchical locality,” in *Proc. IEEE Int'l Symp. on Inf. Theory (ISIT)*, 2015, pp. 1257–1261.

[3] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson, and R. L. Urbanke, “Finite-length analysis of low-density parity-check codes on the binary erasure channel,” *IEEE Trans. on Inf. Theory*, vol. 48, no. 6, pp. 1570–1579, 2002.

[4] R. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. on Inf. Theory*, vol. 27, no. 5, pp. 533–547, 1981.

[5] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, “On the locality of codeword symbols,” *IEEE Trans. on Inf. Theory*, vol. 58, no. 11, pp. 6925–6934, 2012.

[6] I. Tamo, D. S. Papailiopoulos, and A. G. Dimakis, “Optimal locally repairable codes and connections to matroid theory,” *IEEE Trans. on Inf. Theory*, vol. 62, no. 12, pp. 6661–6671, 2016.

[7] B. Chen, S.-T. Xia, J. Hao, and F.-W. Fu, “Constructions of optimal cyclic (r, δ) locally repairable codes,” *IEEE Trans. on Inf. Theory*, vol. 64, no. 4, pp. 2499–2511, 2018.

[8] J. Liu, S. Mesnager, and L. Chen, “New constructions of optimal locally recoverable codes via good polynomials,” *IEEE Trans. on Inf. Theory*, vol. 64, no. 2, pp. 889–899, 2018.

[9] I. Tamo and A. Barg, “A family of optimal locally recoverable codes,” in *Proc. IEEE Int'l Symp. on Inf. Theory*, 2014, pp. 686–690.

[10] G. M. Kamath, N. Prakash, V. Lalitha, and P. V. Kumar, “Codes with local regeneration and erasure correction,” *IEEE Trans. on Inf. Theory*, vol. 60, no. 8, pp. 4637–4660, 2014.

[11] P. Huang, E. Yaakobi, and P. H. Siegel, “Multi-erasure locally recoverable codes over small fields: A tensor product approach,” *IEEE Trans. on Inf. Theory*, vol. 66, no. 5, pp. 2609–2624, 2020.

[12] L. Pamies-Juarez, H. D. Hollmann, and F. Oggier, “Locally repairable codes with multiple repair alternatives,” in *Proc. IEEE Int'l Symp. on Inf. Theory*, 2013, pp. 892–896.

[13] N. Prakash, V. Lalitha, S. B. Balaji, and P. Vijay Kumar, “Codes with locality for two erasures,” *IEEE Trans. on Inf. Theory*, vol. 65, no. 12, pp. 7771–7789, 2019.

[14] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, “Efficient erasure correcting codes,” *IEEE Trans. on Inf. Theory*, vol. 47, no. 2, pp. 569–584, 2001.

[15] C. A. Kelley and D. Sridhara, “Pseudocodewords of Tanner graphs,” *IEEE Trans. on Inf. Theory*, vol. 53, no. 11, pp. 4013–4038, 2007.

[16] A. Orlitsky, R. Urbanke, K. Viswanathan, and J. Zhang, “Stopping sets and the girth of Tanner graphs,” in *Proc. IEEE Int'l Symp. on Inf. Theory*, 2002, p. 2.

[17] A. Orlitsky, K. Viswanathan, and J. Zhang, “Stopping set distribution of LDPC code ensembles,” *IEEE Trans. on Inf. Theory*, vol. 51, no. 3, pp. 929–953, 2005.

[18] M. Schwartz and A. Vardy, “On the stopping distance and the stopping redundancy of codes,” *IEEE Trans. on Inf. Theory*, vol. 52, no. 3, pp. 922–932, 2006.