Cooperative Local Repair in Distributed Storage

Ankit Singh Rawat, Arya Mazumdar, and Sriram Vishwanath

Abstract—Erasure-correcting codes, that support local repair of codeword symbols, have attracted substantial attention recently for their application in distributed storage systems. This paper investigates a generalization of the usual locally recoverable codes. In particular, this paper studies a class of codes with the following property: any small set of codeword symbols is recoverable from a small number of other symbols. This is referred to as cooperative local repair. The main contribution of this paper is bounds on the trade-off of minimum distance and the dimension of such codes, as well as explicit constructions of families of codes that enable cooperative local repair. Some other results regarding cooperative local repair are also presented, including an analysis for the well-known Hadamard codes.

Index Terms—Coding for distributed storage, locally repairable codes, codes on graphs, cooperative repair.

I. INTRODUCTION

In this paper we explore a new class of codes that enable efficient recovery from the failure of multiple code symbols. In particular, we study codes with \((r, \ell)\)-locality which allow for any \(\ell\) failed codes symbols to be recovered by contacting at most \(r\) other intact code symbols. Our study of such codes is motivated by their application in distributed storage systems a.k.a. cloud storage, where information is stored over a network of storage nodes (disks). In order to protect the stored information against inevitable node (disk) failures, a distributed storage system encodes the information using an erasure-correcting code. The code symbols from the obtained codeword are then stored on the nodes in the system.

The task of recovering the code symbols stored on failed nodes with the help of the code symbols stored on intact nodes is refereed to as code repair or node repair [1]. An erasure-correcting code with an efficient code repair process helps quickly restore the state after node failures. This consequently enables seamless operation of the system for a long time period. Recently, multiple classes of erasure-correcting codes have been proposed that optimize the code repair process with respect to various performance metrics. In particular, the codes that minimize repair-bandwidth, i.e., the number of bits communicated during repair of a single node, are studied in [1]–[4] and references therein. The codes that enable small disk-I/O during the repair process are studied in [3], [5]. Another family of erasure codes that focus on small locality, i.e., enabling repair of a single failed code symbol by contacting a small number of other code symbols, are presented in [6]–[11].

A code is said to have all-symbol locality \(r\) if every code symbol is a function of at most \(r\) other code symbols. This ensures local repair of each code symbol by contacting at most \(r\) other code symbols. In this paper we generalize the notion of codes with all-symbol locality to codes with \((r, \ell)\)-locality: any set of \(\ell\) code symbols are functions of at most \(r\) other code symbols. This allows for cooperative local repair of code symbols, where any group of \(\ell\) failed code symbols is repaired by contacting at most \(r\) other code symbols.

The ability to perform code repairs involving more than one failure is a desirable feature in most of the distributed storage systems that can experience multiple simultaneous failures [12]. Moreover, this property also allows for deliberately delaying code repairs when system resources need to be freed to support other system objectives, e.g., queries (accesses) to the stored information by clients. Here, we note that the approach of cooperative code repair has been previously explored in the context of repair-bandwidth efficient codes in [13], [14] and references therein.

In this paper we address two important issues regarding codes with \((r, \ell)\)-locality: 1) obtaining trade-offs among minimum distance, dimension (rate), and locality parameters \((r, \ell)\) for such code; and 2) presenting explicit constructions for codes with \((r, \ell)\)-locality that are close to the obtained trade-offs. Towards designing codes with \((r, \ell)\)-locality, we mainly focus on codes with maximum possible rate. We construct a code with \((r, \ell)\)-locality that has rate at least \(\ell + 1\). In the light of an upper bound \(\frac{\ell + 1}{r + 1}\) on the rate of a code with \((r, \ell)\)-locality that we show later, this construction provides codes that are very close to being optimal. Here, we also note that there are explicit constructions for the regular bipartite graphs with large girth [15]. Thus, one can obtain high (almost optimal) rate codes with \((r, \ell)\)-locality for distributed storage systems. Note that, a minimum distance is not guaranteed in this construction. We also show that the codes based on expander graphs enable cooperative local repairs while maintaining both high rate and good minimum distance.

Given a large number of parity constraints with low weights, expander graph based codes are natural candidates for codes to enable locality. However, these codes are overkill when one is interested in code repair of single failed symbol and codes with significantly better rate vs. distance trade-off can be obtained [6], [9], [11], [16]. But as we aim to recover from multiple failures in a local manner, these codes become an attractive option.
A. Contributions and organization

We first present a formal definition of codes with \((r, \ell)\)-locality in Section II. In the same section, we also highlight the connections between the notion of cooperative locality as defined in this paper and various other contemporary notions from distributed storage literature [17], [20] that aim to generalize locally repairable codes (LRCs) [6], [7].

In Section III, we obtain an upper bound on the minimum distance of a code with \((r, \ell)\)-locality which encodes \(k\) information symbols to \(n\) symbols long codewords. As a corollary of this upper bound, we then bound the best possible rate for a code with \((r, \ell)\)-locality. We address the issue of providing explicit constructions for codes with \((r, \ell)\)-locality in Section IV, VI-A and VI-B.

In Section IV, we present two simple constructions for the codes that have \((r, \ell)\)-locality and comment on their rates with respect to the bound obtained in Section III. In Section VI-A, we consider the codes based on regular bipartite graphs with large girth (girth = length of the smallest cycle). In particular, we show that a code based on regular bipartite graph with girth \(g\) allow for cooperative local repair of \(g - 1\) failed code symbols. We further study cooperative locality of the codes based on expander graphs in Section VI-B. We comment on the conditions in terms of expansion ratio or second eigenvalue that the underlying expander graph needs to satisfy for the code to enable cooperative repair of a certain number of erasures. Table I summarizes the rates and distances obtained by various code constructions considered in this paper.

Certain families of classical algebraic codes may posses local repair property. In Section VII, we study punctured Hadamard codes (a.k.a. Simplex codes) in the context of cooperative local repair. We show that a punctured Hadamard code with \(n\) symbols long codewords has \((r = \ell + 1, \ell)\)-locality for any \(\ell \leq \frac{n-1}{2}\). We conclude this paper in Section VIII with some directions for future work.

A short note on notation: we use bold lower case letters to denote vectors. For an integer \(n \geq 1\), \([n]\) denotes the set \(\{1, 2, \ldots, n\}\). For a code \(\mathcal{C}\), we use \(\text{rate}(\mathcal{C})\) and \(d_{\min}(\mathcal{C})\) to denote its rate and minimum distance, respectively.\}

B. Related work

The concept of codes with small locality for distributed storage system is introduced in [6], [8], [21]. In [6], Gopalan et al. study the rate vs. distance trade-off for linear codes with small locality or locally repairable codes.\(^1\) The similar trade-offs under more general definitions of locally repairable codes and constructions of the codes attaining these trade-offs are studied in [7], [9]–[11], [16], [22], [23] and references therein.

In [24], Prakash et al. consider codes that allow for local repair of multiple code symbols. In particular, they focus on codes that can correct two erasures by utilizing two parity checks of weights at most \(r + 1\). Prakash et al. derive the rate vs. distance trade-off for such code and (for large enough field size) show the existence of the codes that attain the trade-off. We note that the definition of cooperative locality considered in this paper is more general than that studied in [24]. Moreover, we do not restrict ourselves to only two erasures. In Section VI-A, we show that the codes based on regular bipartite graphs with high girth are rate-wise (almost) optimal under the natural generalization of the trade-off to more than two erasures.

Recently, the codes that enables multiple ways to locally repair a code symbols have received attention. In [11], [17], [18], the codes that enable multiple disjoint repair groups for every code symbol are considered. The codes that provide multiple disjoint repair group for only information symbols are studied in [19], [20]. In Section II-A, we comment on the implication of this line of work for the issue of cooperative locality.

II. Codes with \((r, \ell)\)-Locality

Definition 1. A code \(\mathcal{C}\) with length \(n\) and dimension \(k\) is called an \((n, k)\) code. We define \(\mathcal{C}\) to be an \((n, k)\) code with a single failed code symbol.

\(^1\)Throughout this paper, we use both “codes with small locality” and “locally repairable codes” to refer to the codes that enable local repair of a single failed code symbol.

| Construction                      | Locality \((r, \ell)\) | Rate \(\text{rate}(\mathcal{C})\) | Minimum Distance \(d_{\min}(\mathcal{C})\) |
|----------------------------------|------------------------|----------------------------------|----------------------------------|
| Partition code (Sec. IV-A)       | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Product code (Sec. IV-B)         | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Concatenated code (Sec. V)       | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Regular bipartite graph based codes (Sec. VI-A) | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Unbalanced bipartite expander graph (Sec. VI-B1) | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Double cover of regular expander graph (Sec. VI-B2) | \((r, \ell)\)          | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |
| Hadamard codes (Sec. VII)        | \((\ell + 1, \ell)\)   | \(\ell + 1\)                    | \(n + 1 - \ell(\frac{1}{r} - \frac{1}{\ell})\) |

Table I: Summary of the constructions of codes with \((r, \ell)\)-locality considered in this paper. For the codes based on unbalanced bipartite expander graphs (Sec. VI-B1), we assume that the underlying bipartite graphs is bi-regular with \(h\) and \(d\) representing its left and right degrees, respectively. Moreover, the graph exhibits expansions from left to right of any set of at most \(\alpha n\) left nodes with expansion ratio \(h(1 - \epsilon)\). Here the constituent local codes have distance at least \(t + 1\). For codes based on double cover of regular expander graphs (Sec. VI-B2), \(d\) and \(\lambda\) denote the degree and the second largest absolute eigenvalue of the underlying graph. This construction utilizes smaller code of minimum distance at least \(\delta d\) to define local constraints at the vertices of the double cover.
(r, ℓ)-locality if for each \( S \subset [n] \) with \(|S| = \ell\), we have a set \( \Gamma_S \subseteq [n] \setminus S \) such that
1. \(|\Gamma_S| \leq r\),
2. For any codeword \( c = (c_1, c_2, \ldots, c_n) \in \mathcal{C} \), the \( r \) code symbols \( c_S := \{c_i : i \in S\} \) are functions of the code symbols \( c_{\Gamma} := \{c_i : i \in \Gamma_S\} \).

Note that Definition 1 ensures that any \( r \) code symbols can be cooperatively repaired from at most \( r \) other code symbols. This generalizes the notion of codes with all-symbol locality \( r \) [6]–[8], [16], where locality is defined with respect to one code symbol, i.e., \( \ell = 1 \).

**Remark 1.** For a code \( \mathcal{C} \) with all-symbol locality \( r \), we have the following bound on its minimum distance [7], [7].

\[
d_{\min}(\mathcal{C}) \leq n - k - \left\lceil \frac{k}{r} \right\rceil + 2. 
\] (1)

The code attaining the bound in (1) are presented in [7], [9]–[11] and references therein.

**A. Cooperative locality from codes with multiple disjoint local repair groups for code symbols**

In [11], [17], [18], codes with multiple disjoint local repair groups for all code symbols are studied. These codes allow for multiple ways to recover a particular code symbol by contacting disjoint sets of small number of other code symbols. In particular, the work in [11], [17], [18] study codes with at least \( t \) disjoint local repair groups, each comprising of at most \( \tilde{r} \) other code symbols. We claim, according to our definition, these codes also have \((\tilde{r}, \ell = t)\)-locality for each \( \tilde{r} \in [t] \).

Consider a set of \( t \) code symbols in failure. For any of these \( t \) failed code symbols, each symbol can have at least one failed code symbol in at most \( t - 1 \) of its \( t \) disjoint repair groups. This implies that the code symbol under cosideration has at least one of its local repair groups free of any failures. Thus, the code symbol can be repaired with the help of one of its intact local repair groups. This leaves us with \( t - 1 \) code symbols in failure (erasure). Now, for another code symbol in failure, we can have at most \( t - 2 \) of its disjoint local repair groups with at least one failed code symbol. This leaves at least 2 of its disjoint local repair groups intact; therefore, this code symbol can be repaired with the help of one of its intact local repair groups. Following the similar argument, we can see that all of the \( t \) failed code symbols can be repaired in a code with \( t \) disjoint repair groups for all code symbols. In the worst case, we contact at most \( \tilde{r}t \) code symbols to repair all of the \( t \) failures. This establishes the \((\tilde{r}t, \ell = t)\)-locality for the codes under consideration.

Similarly the codes with availability [19], [20], which enable multiple disjoint repair groups only for information (systematic) symbols in a codeword, can allow for cooperative local repair for certain ranges of system parameters. In particular, [20] Construction I] can give codes with \((\tilde{r}t, \ell)\)-locality and rate \( \frac{\tilde{r}}{r+1} \).

**Remark 2.** Here, we would like to note that the definition of the codes with \((r, \ell)\)-locality (Definition 1) is more general. In particular, we show in Section VII-A, Section VII and Section VII-A that it is possible to have codes with \((r, \ell)\)-locality that do not have at least \( t = \ell \) disjoint local repair groups for all code symbols (or information symbols).

**III. RATE VS. DISTANCE TRADE-OFF FOR CODES WITH \((r, \ell)\)-LOCALITY**

In this section, for given \( r \) and \( \ell \), we present a trade-off between the rate and the minimum distance of a code with \((r, \ell)\)-locality (cf. Definition 1). We employ the general proof technique introduced in [6], [22], [23] to obtain the following result.

**Theorem 1.** Let \( \mathcal{C} \) be an \((n, k)\) code (linear, or non-linear) with \((r, \ell)\)-locality. Then, the minimum distance of \( \mathcal{C} \) satisfies

\[
d_{\min}(\mathcal{C}) \leq n - k + 1 - \ell \left(\left\lceil \frac{k}{r} \right\rceil - 1\right). 
\] (2)

**Algorithm:** Construction of sub-code \( \mathcal{C}' \subset \mathcal{C} \).

**Input:** \((n, k)\) code \( \mathcal{C} \subseteq \mathbb{F}_q^n \) with \((r, \ell)\)-locality.
1. \( \mathcal{C}_0 = \mathcal{C} \)
2. \( j = 0 \)
3. while \( |\mathcal{C}_j| > q^\ell \) do
4. \( j = j + 1 \)
5. Choose \( i_1^j, i_2^j, \ldots, i_m^j \in [n] \) such that, for every \( m \in [\ell] \), there exist at least two codewords in \( \mathcal{C}_{j-1} \) that differ at \( i_m^j \)-th coordinate.
6. Let \( \mathcal{R}_j = \Gamma_{\{i_1^j, \ldots, i_m^j\}} \) be the index of at most \( r \) code symbols that cooperatively repair \( \ell \) code symbols indexed by \( \{i_1^j, \ldots, i_m^j\} \).
7. Let \( y \in \mathbb{F}_q^{[\mathcal{R}_j]} \) be the most frequent element in the multi-set \( \{x_{\mathcal{R}_j} : x \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^m\} \).
8. Define \( \mathcal{C}_j := \{x : x \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^m\} \) and \( x_{\mathcal{R}_j} = y \).
9. if \( 1 < |\mathcal{C}_j| \leq q^\ell \) then
10. \( \mathcal{C}' = \mathcal{C}_j \).
11. end while
12. else if \( |\mathcal{C}_j| = 1 \) then
13. Pick a maximal subset \( \tilde{\mathcal{R}}_j \subseteq \mathcal{R}_j \) such that \( |\tilde{\mathcal{C}}_j| > 1 \), where \( \tilde{\mathcal{C}}_j := \{x : x \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^m, x_{\tilde{\mathcal{R}}_j} = \tilde{y}_j\} \) and \( \tilde{y}_j \in \mathbb{F}_q^{[\tilde{\mathcal{R}}_j]} \) be the most frequent element in the multi-set \( \{x_{\tilde{\mathcal{R}}_j} : x \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^m\} \).
14. \( \mathcal{C}' = \tilde{\mathcal{C}}_j \).
15. end while.
16. end if
17. end while
**Output:** \( \mathcal{C}' \).

**Fig. 1:** Construction of sub-code \( \mathcal{C}' \subset \mathcal{C} \).

**Proof:** The proof involves construction of a subcode \( \mathcal{C}' \subset \mathcal{C} \) such that all but a small number of coordinates in every
In this section we address the issue of constructing high rate codes that have \((r, \ell)\)-locality. In particular, we describe two simple constructions that ensure cooperative local repair for the failure of any \(\ell\) code symbols: 1) Partition code and 2) Product code. In Partition code, we partition the information symbols in groups of \(\frac{k}{r}\) symbol and encode each group with an \((\frac{k}{r} + \ell, \frac{k}{r})\)-MDS code (cf. Section IV-A). On the other hand, a product code is obtained by arranging \(k = \left(\frac{k}{r}\right)^\ell\) information symbols in an \(\ell\)-dimensional array and then introducing parity symbols along different dimensions of the array (cf. Section IV-B).

### A. Partition Code

For the ease of exposition, we assume that \(\ell|r\) and \(\left(\frac{k}{r}\right)|k\). Given \(k\) information symbol over \(F_q\), a Partition code encodes these symbols into \(n = k + \ell\) symbols long codewords as follows (see Fig. 2):

1. Partition \(k\) information symbols into \(p = \frac{k\ell}{r}\) groups of size \(\frac{k}{r}\) each.
2. Encode the symbols in each of the \(p\) groups using an \((\frac{k}{r} + \ell, \frac{k}{r})\)-MDS code over \(F_q\). We refer to the \(\frac{k}{r} + \ell\) code symbols obtained by encoding \(\frac{k}{r}\) information symbols in the \(i\)-th group as \(i\)-th local group.

As it is clear from the construction, Partition code has rate \(\frac{k}{n} = \frac{k}{r + \ell}\). Moreover, a code symbol can be recovered from any \(\frac{k}{r}\) other code symbol from its local group. In the worst case, when \(\ell\) failed code symbols belong to different local groups, we can recover all \(\ell\) symbols from \(\ell\frac{k}{r}\) code symbols, downloading \(\frac{k}{r}\) symbols from each of the \(\ell\) local groups containing one failed code symbol.

**Remark 3.** Note that the Partition codes presented here are special cases of the LRCs with locality \(\frac{k}{r}\) and multiple parity symbols \(\ell\) per repair group studied in [9], [16]. The partition codes as described above only aim at maximizing the rate of the code. If we are also interested in achieving large minimum distance, then we can take \(n\) strictly greater than \(k + \ell\) and attain the following relationship between the minimum distance \(d_{\text{min}}\) and the code dimension \(k\) [9]

\[
\begin{align*}
d_{\text{min}}(C) & = n - k + 1 - \ell\left(\frac{k\ell}{r} - 1\right). \tag{7}
\end{align*}
\]

In the above construction of the Partition codes we use an \((\frac{k}{r} + \ell, \frac{k}{r})\) MDS code to encode disjoint groups of \(\frac{k}{r}\) message symbols. Note that the rate of this MDS code governs the rate of the overall code. One can potentially use some other code \(C_{\text{local}}\) of minimum distance at least \(\ell + 1\) to encode disjoint groups of \(\frac{k}{r}\) message symbols. Now, we use \(r(x), x \in \mathbb{F}_r\) to denote the number of symbols that needs to be contacted to repair \(x\) erasure in one local group. For the case when an \((\frac{k}{r} + \ell, \frac{k}{r})\) MDS code is used, we have \(r(x) = \frac{k}{r}\) for \(x \in\)
Let $r^*(x)$ denote the upper concave envelope of $r(x)$ on the interval $[1, \ell] \in \mathbb{R}$. Assume that we have $p$ disjoint local groups, then a pattern of $\ell$ erasures can be represented by a vector $(l_1, l_2, \ldots, l_p)$. Here, $l_i$ denotes the number of erasures within the $i$-th local group. Note that we have $\sum_{i=1}^{p} l_i = \ell$.

For a given local code $C_{\text{local}}$, one needs to access $\sum_{i=1}^{p} r(l_i)$ number of intact code symbols to repair the erasure pattern $(l_1, l_2, \ldots, l_p)$. Now, we use concavity of $r^*(\cdot)$, the fact that $r^*(x) \geq r(x)$ for $x \in [\ell]$, and Jensen’s inequality to obtain the following.

$$\sum_{i=1}^{p} r(l_i) \leq \sum_{i=1}^{p} r^*(l_i) \leq pr^*(\frac{\sum_{i=1}^{p} l_i}{p}) = pr^*(\frac{\ell}{p}) \quad (8)$$

Since the rate of the Partition code is agnostic to the number of local groups, we can use the value of $p$ which can support $k$ message symbols and minimizes the R.H.S. of (8). This approach optimizes the value of $r$ for a given choice of $\ell$ and $C_{\text{local}}$.

### B. Product Code

Product codes are a well known construction of codes in the coding theory literature. Given $k = \left(\frac{\ell}{2}\right)$ information symbols and $\ell r$, we first arrange $k = \left(\frac{\ell}{2}\right)$ information symbols in an $\ell$-dimensional array where each dimension of the array ranging in the set $\left\{\frac{\ell}{2}\right\}$. These information symbols are then encoded to obtain an $n = \left(\frac{\ell}{2} + 1\right)^{\ell}$ symbols long code word. In the following we describe the encoding process for $\ell = 2$-dimensional array (see Fig. 3). The generalization of the encoding process for higher dimensions is straightforward.

1) Arrange $k = \left(\frac{\ell}{2}\right)^2$ information symbols in an $\frac{\ell}{2} \times \frac{\ell}{2}$ array.
2) For each row of the array, add a parity symbol by summing all $\frac{\ell}{2}$ symbols in the row and append these symbols to their respective rows.
3) For each of the $\frac{\ell}{2} + 1$ columns of the updated array, add a parity by summing all $\frac{\ell}{2}$ symbols in the column.

**Remark 4.** An $\ell$-dimensional product code enables $\ell$ disjoint repair groups for all code symbols. For example, every code symbol in Fig. 2 has two disjoint repair groups, associated with its row and column, respectively. Therefore, cooperative locality of product codes follows from the discussion in Section II-A. We note that product codes along with their minimum distance have been previously considered in [17], [19] in the context of small locality.

We now compare the rate of Product code and Product code with the bound in (5). For any $\ell \geq 1$, we have

$$\left(\frac{r}{r + \ell}\right)^{\ell} \leq \frac{r}{r + \ell^2} \quad (9)$$

Note that (9) follows from the fact that

$$r^{\ell}(r + \ell^2) \leq r(r^\ell + \ell r^{\ell-1} \ell) + r(\sum_{i=2}^{\ell} \ell \prod_{i=1}^{\ell-1} \ell) = r(r + \ell)^{\ell}.$$  

![Fig. 3: Description of Product code construction with $\ell = 2$.](image)

Fig. 3: Description of Product code construction with $\ell = 2$. \{p_1^\ell, p_2^\ell, \ldots, p_{\frac{\ell}{2}}^\ell\} denote the parity symbols associated with each row of the two dimensional array of information symbols. \{p_1^\ell, p_2^\ell, \ldots, p_{\frac{\ell}{2}}^\ell\} represent the parity symbols for the columns of the updated array. We use $\gamma$ to denote $\left(\frac{\ell}{2} - 1\right)^{\frac{\ell}{2}}$.

Therefore, Partition code approach provides $(r, \ell)$-locality with a better rate. However, for all system parameters, the rate of Partition code is smaller than the known bound (5), i.e.,

$$\frac{r}{r + \ell^2} \leq \frac{r}{r + \ell}. $$

Here, we would like to note that the difference between the rate achieved by the Partition code and the bound in (5) gets smaller as the parameter $r$ becomes large as compared to the parameter $\ell$. It is an interesting problem to either tighten the bound in (9) or present a construction for codes with $(r, \ell)$-locality which have higher rate than that of Partition code. In the next two sections we present two approaches to achieve this goal.

### V. CONCATENATED CODES WITH $(r, \ell)$-LOCALITY

Here, we describe a family of concatenated codes with $(r, \ell)$-locality. This construction employs an MDS code and a code with small locality as inner and outer codes, respectively. In particular, we employ an $[\frac{\ell}{2} + x, \frac{\ell}{2}, x + 1]$ MDS code over $\mathbb{F}_q$ and an $[n_{\text{out}}, k_{\text{out}}]$ code with $(r_{\text{out}}, \ell_{\text{out}})$-locality over $\mathbb{F}_{q^r}$ as inner and outer codes, respectively. Let $C$ be the concatenated code. We know that

$$R = \text{rate}(C) = \frac{r}{r + x\ell} \cdot \frac{k_{\text{out}}}{n_{\text{out}}} \quad (10)$$

Before we describe the concatenated codes with $(r, \ell)$-locality for general $\ell$, let’s consider a few examples for small values of $\ell$.

1) $\ell = 3$: Let us take an $[\frac{3}{2} + 1, \frac{3}{2}, 2]$ MDS code over $\mathbb{F}_q$ as the inner code. This code can repair any one failed code symbol by contacting the remaining $\frac{3}{2}$ code symbols. For outer code, we employ an code with $(r_{\text{out}}, 1)$-locality over $\mathbb{F}_{q^{3r}}$. This can repair any one super symbol (which consists of $\frac{3}{2}$ symbols of $F_q$) by contacting $r_{\text{out}}$ symbols over $\mathbb{F}_{q^{3r}}$, i.e., $r_{\text{out}} \frac{3}{2}$ symbols over $F_q$. (Note that in order to repair a super symbol, we can obtain the value of $r_{\text{out}}$ required super symbol by contacting $\frac{3}{2}$ symbols over $F_q$ from each of their corresponding codewords of the inner code.)

If $\ell = 3$ erasures lie in the inner codewords of 3 different super symbols, then we can repair each of these erasures by contacting $\frac{3}{2}$ other code symbol. This amounts to using $3\frac{3}{2}$ symbols over $F_q$. If at least 2 erasures belong to the inner codeword of a super symbol, we can employ $(r_{\text{out}}, 1)$-locality
of the outer code to repair the corresponding super symbol. In the worst case, we contact \( r_{\text{out}} \frac{r}{3} + \frac{r}{3} \) symbols over \( \mathbb{F}_q \), when 2 erasures belong to one super symbol and the third erasure belongs to another super symbol. Since we want

\[
r_{\text{out}} \frac{r}{3} + \frac{r}{3} \leq r,
\]

we have \( r_{\text{out}} \leq 2 \). Taking \( r_{\text{out}} = 2 \), we can get the concatenated code with rate

\[
R = \frac{r}{r + 3} \cdot \frac{r_{\text{out}}}{r_{\text{out}} + 1} = \frac{2r}{3(r + 3)}.
\]

Moreover, this code has minimum distance at least 4. Now, we compare the rate of the obtained concatenated code with that of the Partition code described in Section [IV-A] which has rate \( \frac{r}{r + 9} \).

\[
\frac{2r}{3(r + 3)} > \frac{r}{r + 9}
\]

\[
\Rightarrow r < 9.
\]

Hence, for all \( r < 9 \), the concatenated codes have a higher rate than the Partition codes.

2) \( \ell = 4 \): Here, we focus on obtaining the codes with \((r,4)\)-locality. We use an \([\frac{r}{4} + 2, \frac{r}{4}, 3]\) MDS code over \( \mathbb{F}_q \) as the inner code. This code can correct 2 erasures within an inner codeword associated with a super symbol. In order to repair a super symbol, we employ a code with \((r_{\text{out}}, 1)\)-locality over \( \mathbb{F}_q^{r_{\text{out}}/4} \) as an outer code. It can be easily verified that (for suitable value of \( r_{\text{out}} \)) the concatenated code obtained by this approach allows for the recovery of 4 erasures by contacting at most \( r \) symbols over \( \mathbb{F}_q \). In particular, when the inner codeword associated with one super symbol encounter 3 erasures and the inner codeword associated with another super symbol encounter 1 erasure, we contact at most \( r_{\text{out}} \frac{r}{4} + \frac{r}{4} \) symbols over \( \mathbb{F}_q \). Since we need to satisfy

\[
r_{\text{out}} \frac{r}{4} + \frac{r}{4} \leq r,
\]

we have \( r_{\text{out}} \leq 3 \). Working with \( r_{\text{out}} = 3 \), one can obtain a code with \((r,4)\)-locality and rate

\[
R = \frac{r}{r + 4} \cdot \frac{r_{\text{out}}}{r_{\text{out}} + 1} = \frac{3r}{4(r + 8)}.
\]

Moreover, the concatenated codes obtained in this manner have minimum distance at least \( 2 \times 3 = 6 \). We obtain better rate as compared to that of the Partition codes (cf. Section [IV-A]), iff

\[
\frac{3r}{4(r + 8)} > \frac{r}{r + 16}
\]

\[
\Rightarrow r < 16.
\]

3) For general value of \( \ell \): Here, in addition to \( \ell | r \), we assume that \( \ell \) is even. For \( 1 \leq x \leq \ell - 1 \), we now take an \([\frac{r}{x} + x, \frac{r}{x}, x + 1]\) MDS code over \( \mathbb{F}_q \) as the inner code. For outer code, we employ an code over \( \mathbb{F}_q^{r/\ell} \) that can locally recover \( \left\lfloor \frac{\ell}{x+1} \right\rfloor \) failed (erased) super symbols. Note that there exist such codes with rate (cf. Section [IV-A] and Section [II-A])

\[
\hat{r} + \left\lfloor \frac{\ell}{x+1} \right\rfloor,
\]

where \( \hat{r} \) denotes the number of super symbols needed for the local repair of 1 super symbol. In our definition, these codes have \((i\hat{r}, i)\)-locality for all \( i \in \{1, 2, \ldots, \left\lfloor \frac{\ell}{x+1} \right\rfloor \} \).

Now, consider the case where all \( \ell \) erasures lie in the inner codewords corresponding to different super symbols, we can repair all \( \ell \) erasures by contacting \( \ell \times \frac{r}{x} = r \) code symbols over \( \mathbb{F}_q \). For the case where \( 1 \leq y \leq \left\lfloor \frac{\ell}{x+1} \right\rfloor \) super symbols are in erasure, in the worst case, we have \((x + 1)\) erasures in the inner codewords corresponding to \( y \) distinct super symbols and 1 erasure in the inner codewords associated with \( \ell - y(x + 1) \) different super symbols. In order to repair these erasures, we contact

\[
y\left( \frac{r}{\ell} \right) + (\ell - y(x + 1)) \left( \frac{r}{\ell} \right)
\]

symbols over \( \mathbb{F}_q \). Since we need to satisfy,

\[
y\left( \frac{r}{\ell} \right) + (\ell - y(x + 1)) \left( \frac{r}{\ell} \right) \leq r,
\]

we get \( \hat{r} \leq x + 1 \). Therefore, the rate of the concatenated code we get is

\[
R(x) = \frac{x + 1}{x + 1 + \left\lfloor \frac{\ell}{x+1} \right\rfloor} \cdot \frac{r}{\frac{r}{\ell} + x} = \frac{x + 1}{x + 1 + \left\lfloor \frac{\ell}{x+1} \right\rfloor} \cdot \frac{r}{r + x\ell}.
\]

(13)

Note that the concatenated code obtained in this section has minimum distance at least \((x + 1)(\left\lfloor \frac{\ell}{x+1} \right\rfloor + 1)\).

**Remark 5.** If we substitute \( x = \frac{\ell}{2} \) in (13), we obtain a code with rate

\[
R \left( \frac{\ell}{2} \right) = \frac{\frac{\ell}{2} + 1}{\frac{\ell}{2} + 1 + 1} \cdot \frac{r}{r + 2}.
\]

(14)

\[
= \frac{\frac{\ell}{2} + 2}{\frac{\ell}{2} + 4} \cdot \frac{r}{r + 2}.
\]

The rate \( R \left( \frac{\ell}{2} \right) \) in (14) is strictly greater than the rate of the partition codes \( \frac{r}{r + \ell} \) as long as \( r < \frac{\ell}{2} \).

VI. COOPERATIVE LOCALLY REPAIRABLE CODES USING CODES ON GRAPHS

The concatenated codes described in Section [VI] enable \((r, \ell)\)-locality with better rate and minimum distance as compared to those of Partition codes. However, the improvements obtained by concatenated code approach are small and limited to the bounded values of the parameter \( r \). In this section, we present various graphs based codes that improve upon the previously described approaches for a large range of system parameters.
The girth of a graph is the number of vertices in the shortest cycle of the graph. In this section, we explore a particular class of codes based on bipartite graphs with high girth. In this construction, the code symbols are associated with the edges of a bipartite graph and both the left and right vertices in the bipartite graph enforce the constraints on the code symbols associated with the edges incident on these vertices. The analysis of the cooperative locality of the codes obtained in this manner is based on the fact that the underlying bipartite graph has high girth.

Let $G = (U \cup V, E)$ be a bipartite graph where $U$ and $V$ denote the set of left and right vertices, respectively. In particular, we work with the bipartite graphs that are bi-regular, i.e., all the vertices from one part have the same degree. If all the left vertices and right vertices have degree $d_1$ and $d_2$, respectively, then we refer to such a bipartite graph as a $(d_1, d_2)$-regular bipartite graph. In the case, where we have $d_1 = d_2 = d$, we simply call the bipartite graph as $d$-regular bipartite graph. Given the bipartite graph $G$, we obtain a code $C$ (over $\mathbb{F}_q$) in the following manner:

- We assign each edge in the bipartite graph $G(V, E)$ with a code symbol in the codewords of $C$. That is, $C \subseteq \mathbb{F}_q^{\mid E\mid}$.
- For every (left or right) vertex in the bipartite graph, all the code symbols associated with the edges incident on the vertex satisfy a linear constraint (over $\mathbb{F}_q$).

Before stating our general result on the cooperative locality of the codes obtained in this manner, we consider small values of $\ell$. Note that any two edges in $G$ (code symbols in a codeword of $C$) share at most one vertex (appear together in at most one local constraint). Thus, for $\ell = 2$ code symbols in erasure, it is possible to find two local constraints that contain exactly one of the two erased symbols. This allows for the repair of both the erased symbols by utilizing these two local constraints. In other words, the code $C$ has $(2(d_{\text{max}} - 1), 2)$-locality, where $d_{\text{max}}$ denotes the maximum degree of the underlying bipartite graph $G$. Similarly, even in the presence of $\ell = 3$ erasures, one can find at least two local constraints such that there is only one erasure among the code symbols participating in each of these constraints. Fig. 4a and 4b illustrate this fact by considering two possible patterns of $\ell = 3$ erasures. Now, using these two constraints one can repair two erasures, which leaves only one erased symbol which can then be recovered with the help of any of the two local constraints it appears in. The repair of $\ell = 3$ erasures involves at most $3(d_{\text{max}} - 1)$ other code symbols; hence, the code $C$ has $(3(d_{\text{max}} - 1), 3)$-locality. In order to cooperatively repair $\ell > 3$ erasures in $C$, we utilize the fact that the underlying bipartite graph $G$ has high girth.

**Theorem 2.** Let $G$ be a $d$-regular bipartite graph with girth $g$, then the code $C$ obtained from the construction described above has $((g - 1)(d - 1), g - 1)$-locality.

**Proof:** A bipartite graph can only have cycles of even length (number of vertices or edges). Note that as we explain this before stating this theorem, the code $C$ can correct up to 3 erasures without any assumption on the girth of the bipartite graph $G$. Therefore, without loss of generality we can assume that the girth of the bipartite graph $G$ is at least six, i.e., $g \geq 6$. We use induction over the number of erasures to prove the claim. For the base case we consider the case of $\ell = 3$ erasures. As described in the paragraph preceding the statement of this theorem, the code $C$ can recover from 3 erasures in a cooperative manner.

Now as an inductive hypothesis, we assume that the code $C$ can repair at most $\ell \leq g - 2$ erasures in a cooperative manner and show that it is also possible to repair $\ell + 1 \leq g - 1$ erasures. Towards this, we show that given $\ell + 1$ erasures, it is possible to obtain a local constraint which has a single erasure among the code symbols appearing in the constraint. Finding such a constraint allow for the recovery of one erasure leaving $\ell$ erasures. In order to show a contradiction, we assume

Note that Theorem 2 guarantees cooperative local repair of only $\ell = 3$ erasures when $g = 4$. 

---

**Fig. 4:** Two representative patterns of 3 erasures. The edges associated with the three erasures are colored in red.
that no such local constraint exists. We start with a vertex say \( u_1 \in U \) with at least 2 of the code symbols associated with the edges incident on it in erasure. We then traverse along one of the edges out of the vertex \( u_1 \) which have their corresponding code symbols in erasure. (Note that there are at least 2 of such edges.) Let \( v_1 \in V \) denote the vertex that we arrive at after traversing the edge. Since \( v_1 \) has at least 2 code symbols associated with its edges in erasure, we can now pick an edge associated with one of the erased symbol to reach another vertex \( u_2 \in U \) which is different from \( u_1 \). We continue this process until we can not traverse to an unexplored vertex through an edge with its associated symbol in erasure. Note that this process is bound to end in at most \( \ell + 1 \) steps as there are only \( \ell + 1 \) erasures. This process can end with two possibilities: 1) we have traversed through all edges associated with erased symbols or 2) all the unexplored edges from the last vertex leads to previously visited vertices. The first possibility is not feasible under our assumption as it implies that the last vertex has only single erasure associated with the edges incident on it. The second possibility leads to the existence of cycle of length at most \( \ell + 1 \) which is infeasible as \( \ell + 1 \leq g - 1 \). This leads to a contradiction. Thus, it is possible to obtain a local constraint which has a single erasure among the code symbols appearing in the constraint. Now that we are remained with \( \ell \) erasures, we can employ the inductive hypothesis to complete the proof.

As for the total number of intact code symbols contacted during the repair process, in the worst case, we may need to utilize \( g - 1 \) different local constraints to recover from \( g - 1 \) erasures. This amounts to contacting \( (g-1)(d-1) \) intact code symbols.

**Remark 6.** (Construction of regular bipartite graphs with large girth) The problem of constructing regular bipartite graphs with large girth has received significant attention in the past. Here, we like to point out the work presented in [15], [25] and references therein. For an odd integer \( k \geq 1 \) and prime power \( q \), Lazebnik et al. present explicit construction for \( q \)-regular bipartite graphs with girth at least \( k + 5 \) and number of edges \( q^{k-1} \) [15]. Therefore, for any \( \ell \), one can design a code using a regular bipartite graph from [15] which ensures cooperative local repair of any \( \ell \) erasures.

1) **Rate and distance of \( C \) obtained from a regular bipartite graph:** When \( G \) is a regular bipartite graph of degree \( d \), the number of independent linear constrains on the codewords is at most \( \frac{2|E|}{d} \). Hence the rate of the code is

\[
\text{rate}(C) \geq \frac{|E| - 2|E|/d}{|E|} = \frac{d - 2}{d}.
\]

Note that Lemma 2 establishes that the code \( C \) obtained using a \( d \)-regular graph with girth \( g \) has \((g-1)(d-1), g-1)\)-locality. If we set \((g-1)(d-1) = r \) and \( g-1 = \ell \), then the following holds for the code \( C \) with \((r, \ell)\)-locality.

\[
\text{rate}(C) \geq \frac{\frac{r}{\ell}}{\frac{\ell}{r}} = \frac{r - \ell}{\ell + r},
\]

**Remark 7.** Compared to [5], this achievability result has a loss of at most \( \frac{1}{\ell+1} \) from the optimal possible rate.

2) **Comparison with the work in [24]:** Recently, Prakash et al. study codes which allow for local repair of 2 erasures [24]. In their model, they perform the repair of the two erasures in a successive manner, where a parity constraint of weight at most \( \tilde{r} + 1 \) is used to repair each of the two erasures. In [24], Prakash et al. show that such codes have their rates upper bounded by \( \frac{r}{\ell+1} \).

Note that their model can be generalized to \( \ell \geq 2 \) erasures, and one can consider codes that enable successive local repairs from \( \ell \) erasures by contacting \( \ell \) parity constraints of weight at most \( \tilde{r} + 1 \). The codes based on bipartite graphs with high girth, as proposed in this section, fall under this setting. Taking \( \tilde{r} = \frac{r}{\ell} \), their rate (cf. (15)) is at least \( \frac{r - \ell}{\ell + r} \). Since the upper bound \( \frac{r}{\ell+1} \) from [24] still applies to these codes, they exhibit almost optimal rate.

**B. Expander graphs**

The above analysis of the construction based on bipartite graphs fails to show a high minimum distance on top of the local repair property. However with the graphical construction it is also possible to have high distance, and hence protection against catastrophic failures. Next we show how the expansion property of graphs leads to such conclusion.

1) Unbalanced bipartite expanders: Let \( G = (U \cup V, E) \) be an unbalanced left regular bipartite graph with \(|U| = n \geq |V| = m \) and left degree \( h \). We assume that the graph \( G \) is an expander graph where expansion happens from left nodes to right nodes. In particular, we assume that for all \( S \subseteq U \) such that \(|S| \leq \ell \), we have

\[
\Gamma(S) \geq (1-\epsilon)h|S|.
\]

Here, \( \Gamma(S) \subseteq V \) denotes the set of right nodes that constitute the neighborhood of the nodes in the set \( S \).

We now associate a code symbol with each of the left nodes in the bipartite graph \( G \). For \( v \in V \), let \( \Gamma(v) \subseteq U \) denote the neighborhood of the node \( v \). Consider a code \( C \) such that for each \( v \in V \), the code symbols associated with \( \Gamma(v) \) constitute a codeword in a shorter code with length \( |\Gamma(v)| \) and minimum distance at least \( \ell + 1 \). Note that this approach of constructing codes from unbalanced expander graphs is proposed in [26], [27] and references therein.

Next, we argue that for sufficiently small enough \( \epsilon \) (cf. (16)), the code \( C \) should be able to correct any set of at most \( \ell \) erasures. Note that the locality parameter \( r \) is dictated by the degrees of the right nodes in the graph \( G \).

**Theorem 3.** Let \( G \) be an unbalanced (left) expander bipartite graph as defined in (16). If we have \( \epsilon < 1 - \frac{1}{\ell+1} \), then the code \( C \) can be locally repaired from any \( \ell \) or less number of erasures.

**Proof:** We prove the claim using induction on \( \ell \). Note that 1 erasure can be repaired by using one of the local constraints the failed code symbol participates in. Now assume that at most \( \ell - 1 \) erasures can be repaired by using local constraints defined by the graph \( G \). We now show that any set of \( \ell \) erasures can also be repaired using local constraints.

Let \( S \subseteq U \) with \(|S| \leq \ell \) denote the set of \( \ell \) erased code symbols. In order to repair these \( \ell \) erasures, we start with a
right node which has at most $t$ of the code symbols associated
with its neighborhood in erasure. These $t$ erasures can be
corrected under the local constraints satisfied by the code $C$.
We can then utilize the inductive hypothesis to complete the
proof.

Note that what remains to be shown is that the desirable
right node with at most $t$ associated erasures exists. Towards
this, we assume that there is no such right node. In other
words, this implies that the induced subgraph $\tilde{G}$ defined by
the nodes $S \cup \Gamma(S)$ has at least $t+1$ edges incident on every
node in $\Gamma(S)$ from the nodes in $S$. Therefore, we have

$$(t+1)|\Gamma(S)| \leq \text{number of edges in } \tilde{G} = h|S|$$

$$\Rightarrow |\Gamma(S)| \leq \frac{h|S|}{t+1}. \tag{17}$$

However, for $\epsilon < 1 - \frac{1}{t+1}$, it follows from (16) that

$$|\Gamma(S)| > \frac{h|S|}{t+1}.$$  

This along with (17) leads to a contradiction. Hence, in the
presence of at most $t$ erasures it is possible to find the desirable
right node (with at most $t$ erasures among the code symbols
associated with its neighborhood).

Let $\epsilon n$ be such that the graph $\tilde{G}$ allows for expansion of all
sets $S \subseteq U$ of size at most $\epsilon n$ by a factor of at least $(1-\epsilon)h$, i.e.,

$$\Gamma(S) \geq (1-\epsilon)h|S| \text{ for all } S \subseteq U \text{ with } |S| \leq \epsilon n.$$  

Proposition 1. (For the code $C$ based on the bipartite graph
$\tilde{G}$ above and local codes of minimum distance $t+1$, we have,

$$d_{\min}(C) \geq \left(2 - \epsilon - \frac{\epsilon}{t} \right)\epsilon n.$$  

A proof of this fact, which is an extension of existing results
(such as [26]) is provided in Appendix [1]. We further assume
that the bipartite graph $\tilde{G}$ is bi-regular with $d$ denoting its right
degree, i.e., $nh = md$. Moreover, let $C_0$ represent the shorter
code of length $d$ used to define the code $C$. Then we have,

$$\text{rate}(C) \geq \frac{n - md(1 - \text{rate}(C_0))}{n} = 1 + \frac{hr}{dN} - h,$$

where $r = \ell d \cdot \text{rate}(C_0)$ denotes the maximum number of intact
code symbols that need to be contacted to repair $\ell$ erasures.

2) Regular expander graph: We now study the cooperative
locality of the codes obtained by the double covers of $d$-regular
expander graphs [26]. The analysis of the cooperative locality
is based on the analysis of the decoding algorithm for these
codes presented in [28]. Note that we naturally modify the
decoding algorithm from [28] to perform erasure correction in
a cooperative manner.

Let $\tilde{G} = (V, E)$ be a $d$-regular graph with $|V| = N$ and
$\lambda$ as the second (absoluate) largest eigenvalue of its adjacency
matrix. Given $\tilde{G}$, we construct a bipartite graph $G = (V_0 \cup
V_1, E)$ where $V_0 \cup V_1 = V$.

$\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_N$ be $N$ eigenvalues of the adjacency
matrix of $G$, then $\lambda = \max\{\lambda_2, |\lambda_N|\}$.  

As shown above, $\epsilon n \geq \ell$ is a sufficient condition for the code obtained
from the bipartite graph $G$ to be able to allow for cooperative repair of $\ell$
erasures.

If $d = \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_N$ be $N$ eigenvalues of the adjacency
matrix of $G$, then $\lambda = \max\{\lambda_2, |\lambda_N|\}$.  

Fig. 5: Illustration of the double cover $\tilde{G}$ of the $d$-regular graph
$G$. An edge $(u, v)$ in the original graph $G$ contributes to two
degrees $(u_l, v_r)$ and $(v_l, u_r)$ in the the bipartite graph $\tilde{G}$.

$V_1, \tilde{E}$ with $|V_0| = |V_1| = N$ in the following manner (see Fig. [5]):

- Each vertex $u \in V$ in the original graph $G$ corresponds
to a left node $u_l \in V_0$ and a right node $u_r \in V_1$ in the
graph $\tilde{G}$.
- For a pair of vertices $(u_l, v_r) \in V_0 \times V_1$, there exists an
dege $(u_l, v_r) \in \tilde{E}$ if there is an edge between the vertices
$u$ and $v$ in the original graph $G$, i.e., $(u, v) \in E$.

The bipartite graph $\tilde{G}$ is referred to as the double cover of
the graph $G$. Note that the bipartite graph $\tilde{G}$ is $d$-regular with
total $n = Nd$ edges. Moreover, the following result holds for
the bipartite graph $\tilde{G}$.

Lemma 1. (Expander Mixing Lemma) Let $\tilde{G}$ be the $d$-
regular bipartite graph as described above. Then, for every
$S \subseteq V_0$ and $T \subseteq V_1$, we have,

$$|\tilde{E}(S \times T) - \frac{d|S||T|}{N}| \leq \lambda \sqrt{|S||T|}, \tag{18}$$

where $\tilde{E}(S \times T)$ denotes the collection of the edges from
the nodes in the set $S$ to the nodes in the set $T$.

Given the bipartite graph $\tilde{G}$ and a code $C_0$ with $d$-symbol
long codewords, we define a code $C$ as a slight generalization
of the method of Sec. [VI-A]. Each edge in $\tilde{G}$ corresponds to a
code symbol in the codewords of $C$. For each node in the
bipartite graph $\tilde{G}$, the $d$ code symbols associated with the $d$
edges incident on the node constitute a codeword in the code
$C_0$. Note that we assume the local code $C_0$ to be an MDS
code throughout this paper. In Fig. [6] we present an algorithm
which corrects any $\ell$ erasures in $C$ in a cooperative manner by
contacting at most $\ell d \cdot \text{rate}(C_0)$ code symbols. The algorithm
alternates between the left nodes $V_0$ and the right nodes $V_1$ in
order to utilize the smaller code $C_0$ associated with the vertices
to correct the erasures.

Let $S^1 \subseteq V_0$ denotes the set of nodes that have erasures
among the code symbols associated with their edges and did
not attempt to correct those erasures in the first round of the
algorithm. This implies that each vertex in $S^1$ has at least
$d_{\min}(C_0)$ erasures among the code symbols associated with
Algorithm: Erasure correction in $C$.

Input: A codeword from $C$ with at most $\ell$ erasures.
1: $j = 0$.
2: while not all the erasures are corrected do
3: For every vertex $u \in V_i$ such that $1 \leq e \leq d_{\min}(C_0) - 1$
   code symbols among $d$ code symbols associated with it
   are in erasure, use erasure correcting algorithm for $C_0$
   to recover from these erasures.
4: $j = j + 1$ (mod 2).
5: end while

Fig. 6: Cooperative erasure correction in the code based on
the double cover of a regular expander graph.

its $d$ edges. Therefore, we have
\[ |S^1| \leq \frac{\ell}{d_{\min}(C_0)}. \tag{19} \]

We use $S^i$, for $i \geq 2$ to denote the set of (left or
right) vertices that have erasures among the $d$ code symbols
associated with them in the beginning of $i$-th round and did
not attempt to correct those erasures. Note that $S^i \in V_0$ and
$S^i \in V_1$ when $i$ is an odd and even round of decoding,
respectively. Next, we employ the expander mixing lemma
(cf. Lemma 1) to show that $\{|S^1|, |S^2|, |S^3|, \ldots \}$ is a strictly
decreasing sequence.

Lemma 2. Let $S^1, S^2, \ldots$ be the sequence of sets of (left
of right) vertices in the bipartite graph $\mathcal{G}$ as defined above.
Assume that the minimum distance of $C_0$ is at least $(1 + \epsilon)\lambda$
and $\ell \leq \frac{N \lambda \delta}{2} = \frac{n \lambda \delta}{2d}$. Then, for $i \geq 1$, we have
\[ |S^{i+1}| \leq \frac{|S^i|}{1 + \epsilon}. \tag{20} \]

Proof: We prove the relation in (20) for $i = 1$; the
proof for general $i$ involves steps similar to those in the proof
of the $i = 1$ case. Note that each code symbol that is in
erasures after the first round of decoding is associated with
some edge incident on a left node belonging to the set $S^1$. By
the definition of the set $S^2$, it has at least $d_{\min}(C_0)$ erasures
among the $d$ code symbols associated with it after the first
round of decoding. In other words, this implies that each vertex
in the set $S^2$ has at least $d_{\min}(C_0)$ edges incident on it which
are emanating from the vertices from the set $S^1$. Therefore, we have
\[ |S^2|d_{\min}(C_0) \leq |\mathcal{E}(S^1 \times S^2)| \]
\[ \leq \frac{d}{N} |S^1||S^2| + \lambda \sqrt{|S^1||S^2|} \]
\[ \leq \frac{d}{N} |S^1||S^2| + \lambda \frac{|S^1| + |S^2|}{2} \]
\[ \leq \frac{d}{N \cdot d_{\min}(C_0)} + \lambda \frac{|S^1| + |S^2|}{2}, \tag{21} \]
where (a) and (c) follows from Lemma 1 and 19, respectively. Note that we employ the AM-GM inequality to obtain
\[ |S^2| \leq \frac{\lambda}{2 \cdot d_{\min}(C_0) - \lambda - 2d\ell/(N \cdot d_{\min}(C_0))}|S^1|. \tag{22} \]
By replacing $d_{\min}(C_0) = \delta d$ in (22), we get
\[ |S^2| \leq \frac{\lambda}{2\delta d - \lambda - 2\ell/(N \delta)}|S^1|. \tag{23} \]
Under our assumption that $\frac{\lambda}{\delta d} \leq \epsilon$, it follows from (23) that
\[ |S^2| \leq \frac{\lambda}{2\delta d - (1 + \epsilon)\lambda}|S^1|. \tag{24} \]

Now, under the assumption that $\delta d \geq (1 + \epsilon)\lambda$, we get from (24) that
\[ |S^2| \leq \frac{|S^1|}{1 + \epsilon}. \tag{25} \]

It follows from the Lemma 2 that in at most logarithmic (in
$\ell$) rounds of decoding, the algorithm described in Fig. 6 can
correct $\ell$ erasures.

The codes based on the double covers of $d$-regular expander
graphs have been studied in the coding theory literature before
(see e.g. [28]). The rate and the minimum distance of the code
$C$ depends on the rate and the minimum distance of the code $C_0$.
Note that $C_0$ characterizes the local constraint associated
with the vertices in the bipartite graph $\mathcal{G}$. In particular, if rate($C_0$) = $R$ and $d_{\min}(C_0) = \delta d$, then we have that rate($C$) \geq 2R - 1
and $d_{\min}(C) \geq \delta(d - \frac{\delta}{3})n$ in [26], [28].

As we show in this section, for an $\epsilon > 0$ and local code $C_0$
such that $d_{\min}(C_0) = \delta d \geq (1 + \epsilon)\lambda$, it is possible to correct
$\ell \leq \frac{N \lambda \delta}{2}$ erasures using the algorithm described in Fig. 6.
Moreover, in the worst correction of each erasure involves
contacting at most rate($C_0$)$d \leq (1 + rate(C))d/2$ other intact
code symbols (assuming that the local code $C_0$ is an MDS code).
Therefore, the codes based on the double cover of a $d$-
regular expander graph and a local code $C_0$ have $(r, \ell)$-locality
for any $\ell \leq \frac{N \lambda \delta}{2d}$ and $r = \ell(1 + rate(C))d/2$, that is,
rate($C$) \geq \frac{2r}{\ell d} - 1.

In the next section, we show an explicit family of algebraic
codes that exhibit very strong cooperative local repair property,
as well as a very high minimum distance.

VII. COOPERATIVE LOCAL REPAIR FOR HADAMARD CODES

In this section, we study the cooperative locality for punctured
Hadamard codes. Punctured Hadamard codes are also referred to as Simplex codes, which are the dual codes of
Hamming codes. These codes are well known to be locally
decodable codes (LDCs) [29] and have multiple disjoint repair
groups for each code symbol. Here, we comment on the
exact parameters for the cooperative locality of these codes. In
particular, we show that an $[n = 2^k - 1, k, 2^{k-1}]_2$ punctured
Hadamard code has $(r = \ell + 1, \ell)$-locality for any $\ell \leq \frac{n - 1}{2}$. 
An \( n = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code encodes a \( k \) bits long message \((m_1, m_2, \ldots, m_k)\) to an \( n = 2^k - 1 \) codeword \( c = (c_1, c_2, \ldots, c_n=2^k-1) \) such that

\[
c_i = \sum_{j=1}^{k} m_j b_j^i \pmod{2}.
\]

Here \( b^i = (b_1^i, b_2^i, \ldots, b_k^i) \in \mathbb{F}_2^k \) denotes the binary representation of the integer \( i \in [2^k - 1] \). In an \( n = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code, we have \( c_1 + c_2 = c_{i+2} \), where \( 1 \leq j \leq k-1 \) and \( i \in [2^k - 1] \). Moreover, we note that an \( n = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code has a particular structural property: for any \( 2 \leq k < k \), the prefix of length \( 2^k - 1 \) of each codeword is a codeword of the \( \hat{n} = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code which encodes the message \((m_1, m_2, \ldots, m_k)\). We now present the main result of this subsection:

**Lemma 3.** In an \( n = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code, any \( 1 \leq \ell \leq \frac{2^k}{k} - 1 \) erasures can be corrected by contacting at most \( \ell + 1 \) other code symbols.

**Proof:** We prove the Lemma by using induction over \( k \). For base case, we consider \( k = 2 \), where the \( n = 3 = 2^2 - 1, 2, 2^{2-1} \) punctured Hadamard code encodes the message \((m_1, m_2)\) to a codeword \((c_1, c_2, c_3) = (m_1, m_2, m_1 + m_2)\). In this case any \( 1 \leq \ell \leq \frac{3}{1} - 1 = 1 \) erasure can be recovered by contacting other \( \ell + 1 = 2 \) code symbols. For example, one can recover \( c_2 = m_2 \) from \((c_1, c_3) = (m_1, m_1 + m_3)\).

For inductive step, we assume that the Lemma holds for any punctured code of dimension up to \( k - 1 \). Consider the \( n = 2^k - 1, k, 2^{k-1} \) punctured Hadamard code of dimension \( k \), and two cases regarding the positions of \( \ell \) erased code symbols.

- **Case 1:** There are \( x \leq 2^{k-2} - 1 \) erasures among the first \( \hat{n} = 2^{k-1} - 1 \) code symbols. Note that the first \( \hat{n} = 2^{k-1} - 1 \) code symbols constitute a codeword of an \( n = 2^{k-1} - 1, k-1, 2^{k-2} \) punctured Hadamard code. Therefore, from the inductive hypothesis, one can correct the \( x \) erasures among the first \( \hat{n} \) code symbols by contacting \( x + 1 \) other code symbols out of these \( \hat{n} \) code symbols. Now, if the symbol \( c_{2^k-1} \) is in erasure, we can recover it by contacting one of the unerased symbol among \( \{c_{2^k-1+j}, c_{2^k-1+2}, \ldots, c_{n=2^k-1}\} \) say \( c_{2^k-1+j} \) and the corresponding code symbol \( c_j \) from the first \( \hat{n} \) code symbols. Now, we can repair the remaining erased symbols among \( \{c_{2^k-1+j}, c_{2^k-1+2}, \ldots, c_{n=2^k-1}\} \) from \( c_{2^k-1} \) and the corresponding code symbol among the first \( \hat{n} \) code symbols. For example, if we want to recover the symbol \( c_{2^k-1+m} \), we can use \( c_{2^k-1} \) and \( c_m \) to reconstruct \( c_{2^k-1+m} \). In the worst case, we contact \( \ell+1 \) code symbols during the repair of all \( \ell \) erasures.

- **Case 2:** There are \( x \geq 2^{k-2} \) erasures among the first \( \hat{n} = 2^{k-1} - 1 \) code symbols. In this case, we first recover the code symbol \( c_{2^k-1} \), if it is in erasure. Without loss of generality we assume that \( c_{2^k-1} \) is in erasure. Note that there are \( \frac{2^k}{k} = 2^{k-1} \) distinct pairs of code symbols \( \{c_i, c_{2^k-1+i}\} \in [2^k-1] \) that can recover \( c_{2^k-1} \). Since we have at most \( \frac{2^k}{k} - 1 = 2^{k-1} - 1 \) erasures apart from \( c_{2^k-1} \), one of the \( 2^{k-1} \) pairs \( \{c_i, c_{2^k-1+i}\} \in [2^k-1] \) must be intact. This pair allows us to recover \( c_{2^k-1} \).

Now that we know the symbol \( c_{2^k-1} = m_k \), we can remove the contribution of \( m_k \) from any of the last \( 2^k - 1 - 2^k \) code symbols \( \{c_{2^k-1+1}, c_{2^k-1+2}, \ldots, c_{n=2^k-1}\} \). Similarly, we can add \( m_k \) to any of the first \( \hat{n} = 2^{k-1} - 1 \) code symbols \( \{c_1, c_2, \ldots, c_{2^k-1}\} \). Therefore, we can reduce the Case 2 to Case 1 of the proof, and repair any \( \ell+1 \) erasures by contacting at most \( \ell+1 \) code symbols.

Combining both cases completes the proof.

**Remark 8.** Note that, for each symbol, the punctured Hadamard code provides \( \frac{n-1}{2} \) disjoint repair groups. Moreover, each of these repair groups comprises 2 symbols. Therefore, it easily follows from the discussion of Section II-A that the punctured Hadamard code has \((2\ell, \ell)\)-locality for \( \ell \leq \frac{n-1}{2} \). Here, we show that these codes allow for more efficient cooperative local repair mechanism by establishing \((\ell+1, \ell)\)-locality for them.

**VIII. CONCLUSION: COMMENT OF RANDOM ERASURES**

All the constructions of this paper are designed to allow for the cooperative local repairs in the case of adversarial erasure patterns. One can consider the setting where erasures occur according to a random model. Here, we briefly comment on the setting where \( \ell \) erasures are uniformly distributed among the code symbols. Moreover, we assume \( r \) and \( \ell \) to be large enough. In that case, we claim that even the simple Partition codes of Section IV-A are asymptotically optimal. This is true, because with reasonably high probability (depending on \( r \) and \( \ell \)), every local group (a total \( p \) of them) experiences less than about \( t \approx \Theta\left(\frac{p \log^2 \ell}{p}\right) \) number of erasures. Therefore, with high probability, one can perform cooperative local repair of \( \ell \) random erasures even if an \((\ell+1, \ell)\)-MDS code in employed in the construction of the Partition code (cf. Section IV-A). This translates to a coding scheme with the overall rate of \( \frac{r}{r+\ell} \). One can take \( p \) large enough to optimize this value. Indeed, using the techniques of [30], it can be shown that, for random erasures, the Partition codes are asymptotically optimal.

**REFERENCES**

[1] A. G. Dimakis, P. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran. Network coding for distributed storage systems. *IEEE Trans. Inf. Theory*, 56(9):4539–4551, 2010.

[2] K. Rashmi, N. Shah, and P. Kumar. Optimal exact-repairing codes for distributed storage at the MSR and MBR points via a product-matrix construction. *IEEE Trans. Inf. Theory*, 57:5227–5239, 2011.

[3] I. Tamo, Z. Wang, and J. Bruck. Zigzag codes: Mds array codes with optimal rebuilding. *IEEE Trans. Inf. Theory*, 59(3):1597–1616, 2013.

[4] D. S. Papailiopoulos, A. G. Dimakis, and V. Cadambe. Repair optimal erasure codes through hadamard designs. *IEEE Trans. Inf. Theory*, 59(5):3021–3037, 2013.

[5] O. Khan, R. Burns, J. Park, and C. Huang. In search of i/o-optimal recovery from disk failures. In *Proceedings of the 3rd USENIX Conference on Hot Topics in Storage and File Systems*, HotStorage’11, Berkeley, CA, USA, 2011. USENIX Association.

[6] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin. On the locality of codeword symbols. *IEEE Trans. Inf. Theory*, 58(11):6925–6934, 2012.

[7] D. S. Papailiopoulos and A. G. Dimakis. Locally repairable codes. In *Proc. 2012 IEEE International Symposium on Information Theory (ISIT)*, pages 2771–2775, 2012.
Before proceeding with the analysis, we argue the correctness of the algorithm in Fig. [I]. Note that it is always possible to find \( \ell \) coordinates \( \{i_1, i_2, \ldots, i_\ell\} \) at line 5. When the algorithm reaches line 5, the subcode \( C_{j-1} \) has more than \( q^r \) codewords. Therefore, there must be at least \( \ell \) coordinates in the codewords in \( C_{j-1} \) that are not fixed in the previous iterations. This also implies that, for \( m \in [\ell] \),

\[
i_m^j \notin I_{j-1} := \bigcup_{j' \in [j-1]} \left( R_{j'} \cup \{i_1', \ldots, i_\ell'\} \right) \subset [n].
\]

(26)

Note that the code symbols indexed by \( I_{j-1} \) are fixed in \( C_{j-1} \). This further implies that,

\[
R_j = \Gamma_{\{i_1', \ldots, i_\ell'\}} \setminus I_{j-1},
\]

i.e., not all of the code symbols contacted to repair the \( \ell \) symbols indexed by the set \( \{i_1', \ldots, i_\ell'\} \) can be fixed in the previous iterations. Otherwise, the symbols indexed by the set \( \{i_1', \ldots, i_\ell'\} \) would also have been fixed in the previous iterations.

For the construction of a subcode as described in Fig. [I], we define \( A_j = I_j \setminus I_{j-1} \subseteq R_j \cup \{i_1^j, \ldots, i_\ell^j\} \) and \( a_j = |A_j| \). Assuming that the whole loop in Fig. [I] ends with \( j = t \), for \( j \in [t] \), we have

\[
I_j = \bigcup_{j' \in [j]} A_{j'},
\]

where we take union of the disjoint sets \( A_{j'} \), \( j' \in [j] \).

Note that none of the indices in the set \( \{i_1', \ldots, i_\ell'\} \) corresponds to fixed symbols. Thus, by the definition of \( A_j \) and \( a_j \), only \( a_j - \ell \) code symbols among the code symbols indexed by the set \( R_j \) are not fixed in the previous iterations. Hence, at line 7, there are at most \( q^{a_j-\ell} \) possibilities for \( y_j \). This implies that

\[
|C_j| \geq |C_{j-1}|/q^{a_j-\ell}.
\]

(27)

The construction of the subcode \( C' \) can end at either line 11 or line 15. Here we analyze only the case when the construction ends at line 11. In this case, we have \( |C_t| \leq q^r \), or

\[
1 \geq \log_q |C_t| \geq k - \sum_{j=0}^{t-1} (a_{j+1} - \ell).
\]

(28)

Now, using that \( a_j \leq |A_j| \leq |R_j \cup \{i_1^j, \ldots, i_\ell^j\}| \leq r + \ell \), we get

\[
k - 1 \leq \sum_{j=0}^{t-1} (a_{j+1} - \ell) \leq tr,
\]

(29)

which gives us that

\[
t \geq \left\lceil \frac{k}{r} \right\rceil - 1.
\]

(30)

Note that sub-code \( C' = C_t \). Therefore,

\[
\log_q |C'| = \log_q |C_t| \geq \log_q |C| - \sum_{j=0}^{t-1} (a_{j+1} - \ell)
\]

\[
\geq \log_q |C| - \sum_{j=0}^{t-1} a_{j+1} + t \ell
\]

\[
= k - t \ell - |I_t| + t \ell
\]

(31)
where $(a)$ follows from the fact that $I_t$ is union of the disjoint sets $A_j$.

Now, we define $C'' = C'|_{I_t}$ which denotes the code obtained by puncturing the codewords in $C'$ at the coordinates associated with the set $I_t$. We have $|C''| = |C'|$ and $d_{\min}(C'') = d_{\min}(C')$. Moreover, the length of the codewords in $C''$ is $n - |I_t|$. Next, applying the Singleton bound on $C''$ gives us

$$d_{\min}(C) \leq d_{\min}(C'') \leq n - |I_t| - \log_q |C''| + 1 \leq n - |I_t| - (k - |I_t| + t\ell) + 1 = n - k - t\ell + 1,$$

It follows from (32) and (30) that

$$d_{\min}(C) \leq n - k + 1 - \ell \left(\left\lceil \frac{k}{r} \right\rceil - 1\right).$$

This completes the proof.

**APPENDIX B**

**PROOF OF PROP. [1]**

Define $U_t(A), A \subset \mathcal{U}$, to be the set of neighbors of $A$ such that each vertex of $U_t(A)$ is connected to at most $t$ vertices from $A$. Notice that for any $A : |A| \leq \alpha n$, $\Gamma(A) \geq (1 - \epsilon)h|A|$. Furthermore,

$$|U_t(A)| + |\Gamma(A) \setminus U_t(A)| (t + 1) \leq h|A|.$$

Therefore, $|U_t(A)| \geq (1 - \epsilon - \epsilon/t)h|A|$.

For any codeword of $C$ whose support is given by the vertex-set $S \subset \mathcal{U}$, we must have $U_t(S) = \emptyset$. Clearly, when $|S| \leq \alpha n$, $|U_t(S)| \geq (1 - \epsilon - \epsilon/t)h|S| > 0$. Let us assume, $|S| > \alpha n$ but $|S| \leq (2 - \epsilon - \epsilon/t)\alpha n$. Let $Q$ be a proper subset of $S$ such that $|Q| = \alpha n$. The number of edges coming out of $S \setminus Q$ is $h(|S| - \alpha n) \leq h(1 - \epsilon - \epsilon/t)\alpha n$. On the other hand, $U_t(Q) \geq (1 - \epsilon - \epsilon/t)h\alpha n$, Hence $U_t(S) \neq \emptyset$.

This proves that the minimum distance of the expander code is at least $(2 - \epsilon - \epsilon/t)\alpha n$. 