Rawat *et al. EURASIP Journal on Advances in Signal Processing* (2015) 2015:107
DOI 10.1186/s13634-015-0292-0

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Cooperative local repair in distributed storage

Ankit Singh Rawat[1†], Arya Mazumdar[2*†] and Sriram Vishwanath[1]

Abstract

Erasure-correcting codes, which 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 repairable codes. In particular, this paper studies a class of codes with the following property: any small set of codeword symbols can be reconstructed (repaired) 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 the 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/Simplex codes.

Keywords: Coding for distributed storage, Locally repairable codes, Codes on graphs, Cooperative repair

AMS subject classification: 94B05, 94B25, 94B65, 94B60

1 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, ℓ)-cooperative locality which allow for any ℓ failed code 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. Each node stores one code symbol from the codeword.

The task of recovering the code symbols stored on failed nodes with the help of the code symbols stored on intact nodes is referred 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 the 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, ℓ)-cooperative locality: any set of ℓ code symbols are functions of at most r other code symbols. This allows for *cooperative local repair* of code symbols, where any group of ℓ 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

*Correspondence: arya@umn.edu
†Equal contributors
[2]Department of Electrical and Computer Engineering, University of Minnesota Twin Cities, 55455 Minneapolis, MN, USA
Full list of author information is available at the end of the article

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, ℓ)-cooperative locality: (1) obtaining trade-offs among minimum distance, dimension (rate), and locality parameters (r, ℓ) for such code and (2) presenting explicit constructions for codes with (r, ℓ)-cooperative locality that are close to the obtained trade-offs. Towards designing codes with (r, ℓ)-cooperative locality, we mainly focus on codes with maximum possible rate. We construct a code with (r, ℓ)-cooperative locality that has rate at least $\frac{r-\ell}{r+\ell}$. This code construction is based on the regular bipartite graphs with girth at least $\ell + 1$. In the light of an upper bound $\frac{r}{r+\ell}$ on the rate of a code with (r, ℓ)-cooperative 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, ℓ)-cooperative 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.

1.1 Contributions and organization

In Section 2, we first present a formal definition of codes with (r, ℓ)-cooperative locality and highlight the connections between the notion of cooperative locality as defined in this paper and various other contemporary notions from distributed storage literature [9–11, 16–20] that aim to generalize locally repairable codes (LRCs) [6, 7]. In Section 2.1, we comment on the cooperative locality parameters of the codes with multiple small-sized disjoint repair groups for each code symbol [17]. In Section 2.2, we highlight both the differences and similarities between the codes with (r, ℓ)-cooperative locality and the codes with $(\tilde{r}, \delta)$-locality [16].

In Section 3, we obtain an upper bound on the minimum distance of a code with (r, ℓ)-cooperative locality which encodes k information symbols to n symbols long codewords. As a special case of this result, we then obtain a bound on the best possible rate for a code with (r, ℓ)-cooperative locality with no further minimum distance requirement.

We address the issue of providing explicit constructions for codes with (r, ℓ)-cooperative locality in Sections 4, 6.1 and 6.2.

In Section 4, we present two simple constructions for the codes that have (r, ℓ)-cooperative locality and comment on their rates with respect to the bound obtained in Section 3. In Section 6.1, 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 allows for cooperative local repair of $g - 1$ failed code symbols. We further study cooperative locality of the codes based on expander graphs in Section 6.2. 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 1 summarizes the rates and distances obtained by various code constructions considered in this paper.

Certain families of classical algebraic codes may possess local repair property. In Section 7, 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 $(r = \ell+1, \ell)$-cooperative locality for any $\ell \leq \frac{n-1}{2}$. We conclude this paper in Section 8 with some directions for future work.

A short note on notation: we use bold lowercase 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 rate$(\mathcal{C})$ and $d_{\min}(\mathcal{C})$ to denote its rate and minimum distance, respectively.

1.2 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

Table 1 Summary of the constructions of codes with (r, ℓ)-cooperative locality considered in this paper. For the codes based on unbalanced bipartite expander graphs (Section 6.2.1), we assume that the underlying bipartite graphs is bi-regular with h and Δ representing its left and right degrees, respectively. Moreover, the graph exhibits expansions from left to right of any set of at most αn 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 (Section 6.2.2), Δ and λ denote the degree and the second largest absolute eigenvalue of the underlying graph, respectively. This construction utilizes smaller code of minimum distance at least $\delta\Delta$ to define local constraints at the vertices of the double cover

Construction	Cooperative locality	Rate rate($\mathcal{C}$)	Minimum distance $d_{\min}(\mathcal{C})$
Partition code (Section 4.1)	(r, ℓ)	$\frac{r}{r+\ell^2}$	$n-k+1-\ell\left(\frac{k\ell}{r}-1\right)$
Product code (Section 4.2)	(r, ℓ)	$\left(\frac{r}{r+1}\right)^{\ell}$	$\ell+1$
Concatenated code (Section 5)	(r, ℓ)	$\frac{\ell+2}{\ell+4}\frac{r}{r+2}$	$\ell+1$
Regular bipartite graph-based code (Section 6.1)	(r, ℓ)	$\geq \frac{r-\ell}{r+\ell}$	$\geq \ell+1$
Unbalanced bipartite expander graph-based code (Section 6.2.1)	(r, ℓ)	$\geq 1+\frac{h}{\Delta}\frac{r}{\ell}-h$	$\geq \left(2-\epsilon-\frac{\epsilon}{t}\right)\alpha n$
Double cover of regular expander graph based-code (Section 6.2.2)	(r, ℓ)	$\geq 2\frac{r}{\ell\Delta}-1$	$\delta\left(\delta-\frac{\lambda}{\Delta}\right)n$
Hadamard code (Section 7)	$(r=\ell+1, \ell), \forall 1 \leq \ell \leq \frac{n-1}{2}$	$\frac{\log(n+1)}{n}$	$\frac{n+1}{2}$

ourselves to only two erasures. In Section 6.1.2, we show that the codes based on regular bipartite graphs with high girth are rate-wise (almost) optimal under the natural generalization of [24] to more than two erasures.

Recently, the codes that enable 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 2.1, we comment on the implication of this line of work for the issue of cooperative locality.

2 Codes with (r, ℓ)-cooperative locality

Definition 1. A q-ary code $\mathcal{C}$ with length n and dimension $k \equiv \log_q |\mathcal{C}|$ is called an (n, k) code. We define an (n, k) code $\mathcal{C}$ to be a code with (r, ℓ)-cooperative locality if for each $\mathcal{S} \subset [n]$ with $|\mathcal{S}| = \ell$, we have a set $\Gamma_{\mathcal{S}} \subseteq [n] \setminus \mathcal{S}$ such that

1. $|\Gamma_{\mathcal{S}}| \leq r$,
2. For any codeword $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in \mathcal{C}$, the ℓ code symbols $\mathbf{c}_{\mathcal{S}} := \{c_i : i \in \mathcal{S}\}$ are functions of the code symbols $\mathbf{c}_{\Gamma_{\mathcal{S}}} := \{c_i : i \in \Gamma_{\mathcal{S}}\}$.

Note that Definition 1 ensures that any ℓ 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], 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 [6, 7].*

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

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

2.1 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 $(\tilde{r}i, \ell = i)$-cooperative locality for each $i \in [t]$. Without loss of generality, we establish this for $i = t$, i.e., we argue that a code with t disjoint repairs groups (each of size at most $\tilde{r}$) has $(\tilde{r}t, \ell = t)$-cooperative locality.

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 consideration 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 leave 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 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 $(\tilde{r}t, \ell = t)$-cooperative 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 $(\tilde{r}\ell, \ell)$-cooperative locality and rate $\frac{\tilde{r}}{\tilde{r}+\ell}$.

Remark 2. *Here, we would like to note that the definition of the codes with (r, ℓ)-cooperative locality (Definition 1) is more general. In particular, we show in Sections 4.1, 5, and 6.1 that it is possible to have codes with (r, ℓ)-cooperative locality that do not have at least $t = \ell$ disjoint local repair groups for all code symbols (or information symbols).*

2.2 Comparison with the codes with $(\tilde{r}, \delta)$-locality [9, 10]

In [16], Prakash et al. propose to study codes with $(\tilde{r}, \delta)$-locality, a generalization that enforces additional requirements which the local repair groups of an LRC need to satisfy. In particular, a code $\mathcal{C}$ is said to have $(\tilde{r}, \delta)$-locality if there is a set of codes $\{\mathcal{C}_i\}_{i\in\mathcal{L}}$ obtained by puncturing the code $\mathcal{C}$, for some index set $\mathcal{L}$, such that the following three requirements hold: (1) for each $i \in \mathcal{L}$, the support of $\mathcal{C}_i$ is no more than $\tilde{r} + \delta - 1$; (2) for each $i \in \mathcal{L}$, the minimum distance of $\mathcal{C}_i$ is larger than or equal to δ; and (3) each code symbol is contained in the support of at least one of the punctured codes $\mathcal{C}_i$, $i \in \mathcal{L}$. The rate vs. distance trade-offs for the codes with $(\tilde{r}, \delta)$-locality and the constructions attaining these trade-offs are presented in [9, 10].

Note that a code with (r, δ)-locality ensures repair of any $\delta - 1$ failures within each punctured code. Here, we would like to highlight that the notion of (r, ℓ)-cooperative locality is different from that of (r, δ)-locality. In particular, codes with (r, ℓ)-cooperative locality are not required to meet the requirement (2) in the aforementioned definition of the codes with (r, δ)-locality. As a result, there are families of codes which satisfy the requirements of (r, ℓ)-cooperative locality but that do not meet the definition of the codes with (r, δ)-locality. We illustrate this with the help of the following example.

Let $\mathcal{C}$ be a code which encodes three message symbols $\mathbf{m} = (a, b, c)$ to a seven-symbol-long codeword

$$\mathbf{c} = (a, b, c, a + b, b + c, c + a, a + b + c).$$

We note that the code $\mathcal{C}$ is nothing but a $[7, 3, 4]$ Simplex code which we study in Section 7. It follows from the analysis presented in Section 7 that this code has $(r = 3, \ell = 2)$-cooperative locality, i.e., any set of $\ell = 2$ failed code symbols can be recovered by contacting $r = 3$ other code symbols. Let us assume that the code symbols a and $a + b$ are in failure. In this case, we can recover both the failed code symbols from the set of $r = 3$ code symbols $(b, b + c, a + b + c)$. In other words, $(a, a + b, b, b + c, a + b + c)$ form a punctured code of the original code $\mathcal{C}$ at $r + \ell = 5$ indices. However, this punctured code does not have minimum distance at least $\ell + 1 = \delta = 3$. This can easily be observed from the fact that the punctured sub-code does not allow the repair of two code symbols $b + c$ and $a + b + c$ from the remaining set of three code symbols $(a, a + b, b)$. Moreover, there is no punctured codes of the original code at at most $r + \ell = 5$ indices which has minimum distance at least $\ell + 1 = 3$. Therefore, $\mathcal{C}$ is an example of a code with $(r = 3, \ell = 2)$-cooperative locality which does not have $(r = 3, \delta = \ell + 1 = 3)$-locality as defined in [16].

This also shows that the definition of (r, ℓ)-cooperative locality is not a strengthening of the definition of (r, δ)-locality. Hence, one cannot directly invoke the impossibility results for the codes with (r, δ)-locality to obtain impossibility results for the codes with (r, ℓ)-cooperative locality. However, as far as the achievability is concerned, a construction for a code with $(\tilde{r}, \delta)$-locality gives a construction with $(r = \ell\tilde{r}, \ell = \delta - 1)$-cooperative locality as explained in Section 4.

3 Rate vs. distance trade-off for codes with (r, ℓ)-cooperative locality

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

Theorem 1. *Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an (n, k) code (linear, or non-linear) over the finite field $\mathbb{F}_q$ with (r, ℓ)-cooperative locality. Then, the minimum distance of $\mathcal{C}$ satisfies*

$$d_{\min}(\mathcal{C}) \leq n - k + 1 - \ell \left\lfloor \frac{k - \ell}{r} \right\rfloor. \tag{2}$$

Furthermore, when we have $r \geq \ell$, the minimum distance of $\mathcal{C}$ satisfies the following.

$$d_{\min}(\mathcal{C}) \leq n - k + 1 - \ell \left(\left\lceil \frac{k}{r} \right\rceil - 1 \right). \tag{3}$$

Proof. The proof involves construction of a sub-code $\mathcal{C}' \subset \mathcal{C} \subseteq \mathbb{F}_q^n$ such that all but a small number of coordinates in every codeword of $\mathcal{C}'$ are fixed. The coordinates of the codewords in $\mathcal{C}$ are fixed in an iterative manner as follows. In each iteration, we consider a set of ℓ coordinates which have not been fixed so far. Then, we pick the set of r other coordinates such that the code symbols associated with these r coordinates allow us to repair the code symbols associated with the ℓ coordinates under consideration. The current iteration ends with fixing these $r + \ell$ coordinates to some specific values. Note that some of the r coordinates may have been fixed in the previous

Algorithm: Construction of sub-code $\mathcal{C}' \subset \mathcal{C}$.

Input: (n, k) code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with (r, ℓ)-cooperative locality.

$\mathcal{C}_0 = \mathcal{C}$
$j = 0$
while $|\mathcal{C}_j| > q^\ell$ **do**
  $j = j + 1$.
  Choose $i_1^j,\ i_2^j, \ldots, i_\ell^j \in [n]$ 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.
  Let $\mathcal{R}_j = \Gamma_{\{i_1^j, \ldots, i_\ell^j\}}$ be the index of at most r code symbols that cooperatively repair ℓ code symbols indexed by $\{i_1^j, \ldots, i_\ell^j\}$.
  Let $\mathbf{y} \in \mathbb{F}_q^{|\mathcal{R}_j|}$ be the most frequent element in the multi-set $\{\mathbf{x}_{\mathcal{R}_j} : \mathbf{x} \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^n\}$.
  Define $\mathcal{C}_j := \{\mathbf{x} : \mathbf{x} \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^n \text{ and } \mathbf{x}_{\mathcal{R}_j} = \mathbf{y}\}$.
  if $1 < |\mathcal{C}_j| \leq q^\ell$ **then**
    end while
  else if $|\mathcal{C}_j| = 1$ **then**
    Pick a maximal subset $\widetilde{\mathcal{R}}_j \subseteq \mathcal{R}_j$ such that $|\tilde{\mathcal{C}}_j| > 1$, where $\widetilde{\mathcal{C}}_j := \{\mathbf{x} : \mathbf{x} \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^n, \mathbf{x}_{\widetilde{\mathcal{R}}_j} = \widetilde{\mathbf{y}}_j\}$ and $\widetilde{\mathbf{y}}_j \in \mathbb{F}_q^{|\widetilde{\mathcal{R}}_j|}$ be the most frequent element in the multi-set $\{\mathbf{x}_{\widetilde{\mathcal{R}}_j} : \mathbf{x} \in \mathcal{C}_{j-1} \subset \mathbb{F}_q^n\}$.
    $\mathcal{C}_j = \widetilde{\mathcal{C}}_j$.
    end while.
  end if
end while

Output: $\mathcal{C}' = \mathcal{C}_j$.

Fig. 1 Construction of sub-code $\mathcal{C}' \subset \mathcal{C}$

iterations. We describe the iterative construction of the sub-code $\mathcal{C}'$ in Fig. 1. Given the sub-code $\mathcal{C}' \subset \mathcal{C}$, we have

$$d_{\min}(\mathcal{C}) \leq d_{\min}(\mathcal{C}'). \tag{4}$$

Given $\mathcal{C}'$, one can obtain a code $\mathcal{C}''$ with $|\mathcal{C}''| = |\mathcal{C}'|$ by removing fixed coordinates from all the codeword in $\mathcal{C}'$. This implies that $d_{\min}(\mathcal{C}'') = d_{\min}(\mathcal{C}')$, which along with (4) give us the following.

$$d_{\min}(\mathcal{C}) \leq d_{\min}(\mathcal{C}''). \tag{5}$$

We refer the reader to Section 8.1 for the complete proof. □

Remark 3. *It is possible to obtain a bound on the minimum distance of codes with (r, ℓ)-cooperative locality that depends on the alphabet size, in the spirit of [22]. Indeed, a more general version of Theorem 1 will give,*

$$k \leq \min_{t \leq \min\left\{\left\lfloor \frac{n}{r+\ell} \right\rfloor, \left\lfloor \frac{k-1}{r} \right\rfloor\right\}} rt + \log_q A_q(n - t(r+\ell), d),$$

where $A_q(n, d)$ is the maximum size of a q-ary error-correcting code of length n and distance d. The proof of this bound is straightforward.

Note that an (n, k) code with (r, ℓ)-cooperative locality has its minimum distance at least $\ell + 1$ as it can recover from the erasure of any ℓ code symbols (cf. Definition 1). Combining this observation with Theorem 1, we obtain the following result.

Corollary 1. *The rate of an (n, k) code with (r, ℓ)-cooperative locality is bounded as*

$$\frac{k}{n} \leq \frac{r}{r+\ell} + \frac{1}{n}\frac{\ell^2}{r}. \tag{6}$$

Furthermore, for the case when we have $r \geq \ell$, the rate of an (n, k) code with (r, ℓ)-cooperative locality satisfies

$$\frac{k}{n} \leq \frac{r}{r+\ell}. \tag{7}$$

Proof. It follows from (2) and the fact $d_{\min}(\mathcal{C}) \geq \ell + 1$ that

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

By using $\left\lfloor \frac{k-\ell}{r} \right\rfloor \geq \frac{k-\ell}{r} - 1$, we get

$$\frac{k}{n} \leq \frac{r}{r+\ell} + \frac{1}{n}\frac{\ell^2}{r}. \tag{8}$$

In the case when we have $r \geq \ell$, we can combine (3) with the observation $d_{\min}(\mathcal{C}) \geq \ell + 1$ to obtain the following.

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

By using $\left\lceil \frac{k}{r} \right\rceil - 1 \geq \frac{k}{r} - 1$, we get

$$\frac{k}{n} \leq \frac{r}{r+\ell}. \tag{9}$$

□

Remark 4. *Here, we note that the assumption $r \geq \ell$ is a natural assumption as it always holds for linear codes with (r, ℓ)-cooperative locality and dimension at least ℓ, i.e., $k \geq \ell$. Note that the additional term $\frac{1}{n}\frac{\ell^2}{r}$ that we have for the case when $r < \ell$ vanishes as n becomes large as compared to ℓ.*

4 Naive constructions of codes with (r, ℓ)-cooperative locality

In this section, we address the issue of constructing high rate codes that have (r, ℓ)-cooperative locality. In particular, we describe two simple constructions that ensure cooperative local repair for the failure of any ℓ code symbols: (1) partition code and (2) product code. In partition code, we partition the information symbols in groups of $\frac{r}{\ell}$ symbol and encode each group with an $\left(\frac{r}{\ell} + \ell, \frac{r}{\ell}\right)$ maximum distance separable (MDS) code (cf. Section 4.1). On the other hand, a product code is obtained by arranging $k = \left(\frac{r}{\ell}\right)^{\ell}$ information symbols in an ℓ-dimensional array and then introducing parity symbols along different dimensions of the array (cf. Section 4.2).

4.1 Partition code

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

1. Partition k information symbols into $p = \frac{k\ell}{r}$ groups of size $\frac{r}{\ell}$ each.
2. Encode the symbols in each of the p groups using an $\left(\frac{r}{\ell} + \ell, \frac{r}{\ell}\right)$ MDS code over $\mathbb{F}_q$. We refer to the $\frac{r}{\ell} + \ell$ code symbols obtained by encoding $\frac{r}{\ell}$ information symbols in the ith group as ith local group.

As it is clear from the construction, partition code has the rate $\frac{k}{n} = \frac{r}{r+\ell^2}$. Moreover, a code symbol can be recovered from any $\frac{r}{\ell}$ other code symbols from its local group. In the worst case, when ℓ failed code symbols belong to ℓ distinct local groups, we can recover all ℓ symbols from $\ell\frac{r}{\ell} = r$ code symbols, downloading $\frac{r}{\ell}$ symbols from each of the ℓ local groups containing one failed code symbol.

Remark 5. *Note that the partition codes presented here are special cases of codes with $(\frac{r}{\ell}, \delta = \ell + 1)$-locality as studied in [9, 10] (cf. Section 2.2). 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 $k\frac{r+\ell^2}{r}$ and attain the following relationship between the minimum distance $d_{\min}$ and the code dimension k [9]*

$$d_{\min}(\mathcal{C}) = n - k + 1 - \ell \left(\frac{k\ell}{r} - 1 \right). \tag{10}$$

In the above construction of the partition codes, we use an $\left(\frac{r}{\ell} + \ell, \frac{r}{\ell}\right)$ MDS code to encode disjoint groups of $\frac{r}{\ell}$ message symbols. Note that the rate of this MDS code governs the rate of the overall code. One can potentially use some other code $\mathcal{C}^{\text{local}}$ of minimum distance at least $\ell + 1$ to encode disjoint groups of $\frac{r}{\ell}$ message symbols. Now, we use $r(x), x \in [\ell]$ to denote the number of symbols that needs to be contacted to repair x erasure in one local group. For the case when an $\left(\frac{r}{\ell} + \ell, \frac{r}{\ell}\right)$ MDS code is used, we have $r(x) = \frac{r}{\ell}$ for $x \in [\ell]$. 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 ℓ erasures can be represented by a vector $(l_1, l_2, \ldots, l_p)$. Here, l_i denotes the number of erasures within the ith local group. Note that we have $\sum_{i=1}^{p} l_i = \ell$.

For a given local code $\mathcal{C}^{\text{local}}$, 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.

$$\sum_{i=1}^{p} r(l_i) \leq \sum_{i=1}^{p} r^*(l_i) \leq p r^* \left(\frac{\sum_{i=1}^{p} l_i}{p} \right) = p r^* \left(\frac{\ell}{p} \right). \tag{11}$$

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 (11). This approach optimizes the value of r for a given choice of ℓ and $\mathcal{C}^{\text{local}}$.

Example 1. It is possible to achieve better locality parameters in the partition code than just to use the copies of MDS codes. Consider a partition code with two blocks, each being a punctured Hadamard [7,3,4] code. From Theorem 4, we know that $r(x) = r^*(x) = x + 1$, for $1 \leq x \leq 3$ for these Hadamard codes. Hence, we have an $(14, 6)$ code with $(5, 3)$-cooperative locality.

On the other hand, consider a partition code with two blocks of $(7, 3)$ MDS codes. For this $(14, 6)$ code, we may need to access up to six symbols to repair even two symbols. Indeed, the overall code has $(6, 3)$-cooperative locality.

4.2 Product code

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

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

Remark 6. *An ℓ-dimensional product code enables ℓ disjoint repair groups for all code symbols. For example, every code symbol in a two-dimensional product code has two disjoint repair groups, associated with its row and column, respectively. Therefore, cooperative locality of product codes follows from the discussion in Section 2.1. We note that product codes along with their minimum distance have been previously been considered in [11, 19] in the context of codes with small locality.*

We now compare the rate of partition code and product code with the bound in (7). For any $\ell \geq 1$, we have

$$\left(\frac{r}{r+\ell}\right)^{\ell} \leq \frac{r}{r+\ell^2}. \tag{12}$$

Note that (12) follows from the fact that

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

Therefore, partition code approach provides (r,ℓ)-cooperative locality with a better rate. However, for all system parameters, the rate of partition code is smaller than the known bound (7), 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 (7) gets smaller as the parameter r becomes large as compared to the parameter ℓ. It is an interesting problem to either tighten the bound in (7) or present a construction for codes with (r,ℓ)-cooperative locality which have higher rate than that of the partition code. In the next two sections, we present two approaches to achieve this goal.

5 Concatenated codes with (r,ℓ)-cooperative locality

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

$$R = \text{ rate}(\mathcal{C}) = \frac{r}{r+x\ell} \cdot \frac{k_{\text{out}}}{n_{\text{out}}} \tag{13}$$

Before we describe the concatenated codes with (r,ℓ)-cooperative locality for general ℓ, let us consider a few examples for small values of ℓ.

5.1 When $\ell = 3$

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

If $\ell = 3$ erasures lie in the inner codewords of three different super symbols, then we can repair each of these erasures by contacting $\frac{r}{3}$ other code symbols. This amounts to using $3\frac{r}{3} = r$ symbols over $\mathbb{F}_q$. If at least two erasures belong to the inner codeword of a super symbol, we can employ $(r_{\text{out}}, 1)$-cooperative 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 two 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 four. Now, we compare the rate of the obtained concatenated code with that of the partition code described in Section 4.1, which has the rate $\frac{r}{r+9}$.

$$\frac{2r}{3(r+3)} > \frac{r}{r+9} \Rightarrow \ r < 9. \tag{14}$$

Hence, for all $r < 9$, the concatenated codes have a higher rate than the partition codes.

5.2 When $\ell = 4$

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

$$R = \frac{r/4}{r/4+2} \cdot \frac{r_{\text{out}}}{r_{\text{out}}+1} = \frac{3r}{4(r+8)}.$$

Moreover, the concatenated codes obtained in this manner have minimum distance at leat $2 \times 3 = 6$. We obtain better rate as compared to that of the partition codes (cf. Section 4.1), iff

$$\frac{3r}{4(r+8)} > \frac{r}{r+16} \Rightarrow \quad r < 16. \tag{15}$$

5.3 General values of ℓ

Here, in addition to $\ell | r$, we assume that ℓ is even[2]. For $1 \leq x \leq \ell - 1$, we now take an $\left[\frac{r}{\ell}+x, \frac{r}{\ell}, x+1\right]$ MDS code over $\mathbb{F}_q$ as the inner code. For outer code, we employ a 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. Sections 4.1 and 2.1)

$$\frac{\tilde{r}}{\tilde{r}+\left\lfloor \frac{\ell}{x+1} \right\rfloor},$$

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

Now, consider the case where all ℓ erasures lie in the inner codewords corresponding to different super symbols, we can repair all ℓ erasures by contacting $\ell \times \frac{r}{\ell} = r$ 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

$$y\tilde{r}\frac{r}{\ell} + (\ell - y(x+1))\frac{r}{\ell}$$

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

$$y\tilde{r}\frac{r}{\ell} + (\ell - y(x+1))\frac{r}{\ell} \leq r,$$

we get $\tilde{r} \leq x+1$. Therefore, the rate of the concatenated code we get is

$$R(x) = \frac{x+1}{x+1+\left\lfloor \frac{\ell}{x+1} \right\rfloor} \cdot \frac{\frac{r}{\ell}}{\frac{r}{\ell}+x} = \frac{x+1}{x+1+\left\lfloor \frac{\ell}{x+1} \right\rfloor} \cdot \frac{r}{r+x\ell}. \tag{16}$$

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

Remark 7. *If we substitute $x = \frac{\ell}{2}$ in* (16)*, we obtain a code with rate*

$$R\left(\frac{\ell}{2}\right) = \frac{\frac{\ell}{2}+1}{\frac{\ell}{2}+1+1} \cdot \frac{r}{r+2} = \frac{\ell+2}{\ell+4} \cdot \frac{r}{r+2}. \tag{17}$$

The rate $R\left(\frac{\ell}{2}\right)$ in (17) is strictly greater than the rate of the partition codes $\frac{r}{r+\ell^2}$ as long as $r < \frac{\ell^3}{4}$.

6 Cooperative locally repairable codes using codes on graphs

The concatenated codes described in Section 5 enable (r, ℓ)-cooperative 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 graph-based codes that improve upon the previously described approaches for a large range of system parameters.

6.1 Bipartite graphs with large girth

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 the bipartite graph enforces 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 $\mathcal{G} = (\mathcal{U} \cup \mathcal{V}, \mathcal{E})$ be a bipartite graph where $\mathcal{U}$ and $\mathcal{V}$ 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

degrees Δ_1 and Δ_2, respectively, then we refer to such a bipartite graph as a (Δ_1, Δ_2)-regular bipartite graph. In the case, where we have $\Delta_1 = \Delta_2 = \Delta$, we simply call the bipartite graph as Δ-regular bipartite graph. Given the bipartite graph $\mathcal{G}$, we obtain a code $\mathcal{C}$ (over $\mathbb{F}_q$) in the following manner:

- We assign each edge in the bipartite graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with a code symbol in the codewords of $\mathcal{C}$. That is, $\mathcal{C} \subseteq \mathbb{F}_q^{|\mathcal{E}|}$.
- 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 ℓ. Note that any two edges in $\mathcal{G}$ (code symbols in a codeword of $\mathcal{C}$) 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 $\mathcal{C}$ has $(2(\Delta_{\max} - 1), 2)$-cooperative locality, where $\Delta_{\max}$ denotes the maximum degree of the underlying bipartite graph $\mathcal{G}$. 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. Figure 2a, b illustrates 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 $3(\Delta_{\max} - 1)$ other code symbols; hence, the code $\mathcal{C}$ has $(3(\Delta_{\max} - 1), 3)$-cooperative locality. In order to cooperatively repair $\ell > 3$ erasures in $\mathcal{C}$, we utilize the fact that the underlying bipartite graph $\mathcal{G}$ has high girth.

Theorem 2. *Let $\mathcal{G}$ be a Δ-regular bipartite graph with girth g, then the code $\mathcal{C}$ obtained from the construction described above has $((g-1)(\Delta-1), g-1)$-cooperative 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 $\mathcal{C}$ can correct up to three erasures without any assumption on the girth of the bipartite graph $\mathcal{G}$. Therefore, without loss of generality, we can assume that the girth of the bipartite graph $\mathcal{G}$ is at least six[3], 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 $\mathcal{C}$ can recover from three erasures in a cooperative manner.

Now as an inductive hypothesis, we assume that the code $\mathcal{C}$ 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 ℓ erasures. In order to show a contradiction, we assume that no such local constraint exists. We start with a vertex say $u_1 \in \mathcal{U}$ with at least two of the code symbols associated with the edges incident on it in erasure. We then traverse along one of the

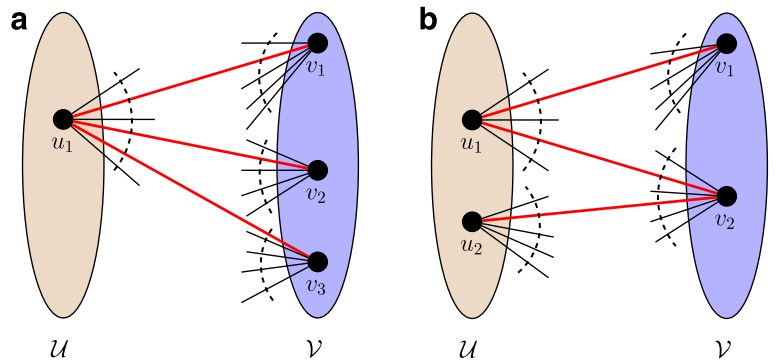

Fig. 2 Two representative patterns of three erasures. The edges associated with the three erasures are colored in *red*. **a** In this pattern, all three erased code symbols appear together in a local constraint associated with the left vertex u_1. Therefore, all the erased symbols must appear in different local constraints corresponding to the right vertices. This allows for the recovery of all three erased code symbols by using local constraints associated with the right vertices v_1, v_2, and v_3. **b** In this pattern, two erased code symbols together in the local constraint associated with the left vertex u_1. This leads to both the erased symbols appearing in two different local constraints defined by the right vertices v_1 and v_2. Since the third erased symbol can participate in only one of the local constraints associated with the right vertices (v_2 here), one can recover all three erased symbols

edges out of the vertex u_1 which have their corresponding code symbols in erasure. (Note that there are at least two of such edges.) Let $v_1 \in \mathcal{V}$ denote the vertex that we arrive at after traversing the edge. Since v_1 has at least two 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 $u_2 \in \mathcal{U}$ which is different from u_1. We continue this process until we cannot 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 ℓ 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 $(g-1)(\Delta-1)$ intact code symbols. □

Remark 8. *(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 ℓ, one can design a code using a regular bipartite graph from [15] which ensures cooperative local repair of any ℓ erasures.*

6.1.1 Rate and distance of $\mathcal{C}$ obtained from a regular bipartite graph

When $\mathcal{G}$ is a regular bipartite graph of degree Δ, the number of independent linear constrains on the codewords is at most $\frac{2|\mathcal{E}|}{\Delta}$. Hence, the rate of the code is

$$\text{rate}(\mathcal{C}) \geq \frac{|\mathcal{E}| - 2|\mathcal{E}|/\Delta}{|\mathcal{E}|} = \frac{\Delta-2}{\Delta}.$$

Note that Theorem 2 establish that the code $\mathcal{C}$ obtained using a Δ-regular graph with girth g has $((g-1)(\Delta-1), g-1)$-cooperative locality. If we set $(g-1)(\Delta-1) = r$ and $g-1 = \ell$, then the following holds for the code $\mathcal{C}$ with (r,ℓ)-cooperative locality.

$$\text{rate}(\mathcal{C}) \geq \frac{\frac{r}{\ell}-1}{\frac{r}{\ell}+1} = \frac{r-\ell}{r+\ell}. \tag{18}$$

As far as the minimum distance $d_{\min}(\mathcal{C})$ of a code $\mathcal{C}$ based on a Δ-regular bipartite graph $\mathcal{G}$ with girth g is concerned, we have the following trivial bound from Theorem 2.

$$d_{\min}(\mathcal{C}) \geq g. \tag{19}$$

One can construct a Tanner graph $\mathcal{H}$ corresponding to the graph $\mathcal{G}$. The left vertices and right vertices in this Tanner graph correspond to the edges in the graph $\mathcal{G}$ and the vertices in the graph $\mathcal{G}$, respectively. The Tanner graph $\mathcal{H}$ is a bi-regular bipartite graph with left degree 2 and right degree Δ. Moreover, the girth of $\mathcal{H}$ is $2g$. We can now use ([26] Theorem 2) to conclude that

$$d_{\min}(\mathcal{C}) \geq \tilde{d}_{\min}\frac{(\tilde{d}_{\min}-1)^{g/2}-1}{\tilde{d}_{\min}-2}, \tag{20}$$

where $\tilde{d}_{\min}$ is the minimum distance of the smaller code associated with each vertex in the graph $\mathcal{G}$. For our case of $\tilde{d}_{\min} = 2$, (20) does not give us anything better than (19).

Remark 9. *The relationship between stopping number, the smallest number of erasures that cannot be corrected under iterative decoding, and the girth of the Tanner graph associated with a code have been previously explored in the literature [27]. As described above, we can obtain a Tanner graph $\mathcal{H}$ corresponding to the graph $\mathcal{G}$. This allows us to draw the connections between Theorem 2 and the literature on stopping number.*

Remark 10. *Compared to* (7)*, this achievability result has a loss of at most $\frac{\ell}{r+\ell}$ from the optimal possible rate.*

6.1.2 Comparison with the work in [24]

Recently, Prakash et al. study codes which allow for local repair of two 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 $\frac{\tilde{r}}{\tilde{r}+2}$.

Note that their model can be generalized to $\ell \geq 2$ erasures, and one can consider codes that enable successive local repairs from ℓ erasures by contacting ℓ 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. (18)) is at least $\frac{\tilde{r}-1}{\tilde{r}+1}$. Since the upper bound $\frac{\tilde{r}}{\tilde{r}+2}$ from [24] still applies to these codes, they exhibit almost optimal rate.

6.2 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.

6.2.1 Unbalanced bipartite expanders

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

$$\Gamma(\mathcal{S}) \geq (1-\epsilon)h|\mathcal{S}|. \tag{21}$$

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

We now associate a code symbol with each of the left nodes in the bipartite graph $\mathcal{G}$. For $v \in \mathcal{V}$, let $\Gamma(v) \subseteq \mathcal{U}$ denote the neighborhood of the node v. Consider a code $\mathcal{C}$ such that for each $v \in \mathcal{V}$, the code symbols associated with $\Gamma(v)$ constitute a codeword in a shorter MDS code $\mathcal{C}_0$ with length $\Delta = |\Gamma(v)|$ and minimum distance at least $t+1$. Note that this approach of constructing codes from unbalanced expander graphs is proposed in [26, 28] and references therein.

Next, we argue that for small enough ϵ (cf. (21)), the code $\mathcal{C}$ should be able to correct any set of at most ℓ erasures. Note that the locality parameter r is dictated by the degrees of the right nodes in the graph $\mathcal{G}$.

Theorem 3. *Let $\mathcal{G}$ be an unbalanced (left) expander bipartite graph as defined in* (21)*. If we have $\epsilon < 1 - \frac{1}{t+1}$, then the code $\mathcal{C}$ can be locally repaired from any ℓ or less number of erasures by contacting at most $\ell\Delta \cdot rate(\mathcal{C}_0)$ code symbols.*

Proof. We prove the claim using induction on ℓ. Note that a single erasure can be repaired by using one of the local constraints the erased code symbol participates in. Now assume that at most $\ell - 1$ erasures can be repaired by using local constraints defined by the graph $\mathcal{G}$. We now show that any set of ℓ erasures can also be repaired using local constraints.

Let $\mathcal{S} \subseteq \mathcal{U}$ with $|\mathcal{S}| \leq \ell$ denote the set of ℓ erased code symbols. In order to repair these ℓ 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 $\mathcal{C}$. 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 $\widehat{\mathcal{G}}$ defined by the nodes $\mathcal{S} \cup \Gamma(\mathcal{S})$ has at least $t+1$ edges incident on every node in $\Gamma(\mathcal{S})$ from the nodes in $\mathcal{S}$. Therefore, we have

$$\begin{aligned}(t+1)|\Gamma(\mathcal{S})| &\leq \text{number of edges in } \widehat{\mathcal{G}} = h|\mathcal{S}| \\ \Rightarrow |\Gamma(\mathcal{S})| &\leq \frac{h|\mathcal{S}|}{t+1}. \end{aligned} \tag{22}$$

However, for $\epsilon < 1 - \frac{1}{t+1}$, it follows from (21) that

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

This along with (22) leads to a contradiction. Hence, in the presence of at most ℓ erasures, it is possible to find the desirable right node (with at most t erasures among the code symbols associated with its neighborhood).

Now the claim that $r \leq \ell\Delta \cdot \text{rate}(\mathcal{C}_0)$ follows from the fact that correcting each erasure requires contacting at least $\Delta \cdot \text{rate}(\mathcal{C}_0)$ code symbols from a codeword of the shorter code $\mathcal{C}_0$. □

Let αn be such that the graph $\mathcal{G}$ allows for expansion of all sets $\mathcal{S} \subseteq \mathcal{U}$ of size at most αn by a factor of at least $(1-\epsilon)h$, i.e.[4],

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

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

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

A proof of this fact, which is an extension of existing results (such as [28]) is provided in Section 8.2. We further assume that the bipartite graph $\mathcal{G}$ is bi-regular with Δ denoting its right degree, *i.e.,* $nh = m\Delta$. Moreover, let $\mathcal{C}_0$ represent the shorter code of length Δ used to define the code $\mathcal{C}$. Then we have,

$$\text{rate}(\mathcal{C}) \geq \frac{n - m\Delta(1 - \text{rate}(\mathcal{C}_0))}{n} = 1 + \frac{h}{\Delta}\frac{r}{\ell} - h,$$

where $r = \ell\Delta \cdot \text{rate}(\mathcal{C}_0)$ denotes the maximum number of intact code symbols that need to be contacted to repair ℓ erasures.

Remark 11. *Here, we note that for any constant $\epsilon > 0$ and $\delta < 1$, it is possible to explicitly construct unbalanced expander graphs with constant degree h, $m = \delta n$ and expansion factor $(1-\epsilon)h$ for $\Omega(n)$ sized subsets of left vertices [29].*

6.2.2 Regular expander graph

We now study the cooperative locality of the codes obtained by the double covers of Δ-regular expander graphs [28]. The analysis of the cooperative locality is based on the analysis of the decoding algorithm for these codes presented in [30]. Note that we naturally modify the decoding algorithm from [30] to perform erasure correction in a cooperative manner.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a Δ-regular graph with $|\mathcal{V}| = N$ and λ as the second (absolute) largest eigenvalue of its adjacency matrix[5]. Given $\mathcal{G}$, we construct a bipartite graph $\widetilde{\mathcal{G}} = (\mathcal{V}_0 \cup \mathcal{V}_1, \widetilde{\mathcal{E}})$ with $|\mathcal{V}_0| = |\mathcal{V}_1| = N$ in the following manner (see Fig. 3):

- Each vertex $u \in \mathcal{V}$ in the original graph $\mathcal{G}$ corresponds to a left node $u_l \in \mathcal{V}_0$ and a right node $u_r \in \mathcal{V}_1$ in the graph $\widetilde{\mathcal{G}}$.
- For a pair of vertices $(u_l, v_r) \in \mathcal{V}_0 \times \mathcal{V}_1$, there exists an edge $(u_l, v_r) \in \widetilde{\mathcal{E}}$ iff there is an edge between the vertices u and v in the original graph $\mathcal{G}$, i.e., $(u, v) \in \mathcal{E}$.

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

Lemma 1. ***(Expander mixing lemma)*** *[31] Let $\widetilde{\mathcal{G}}$ be the Δ-regular bipartite graph as described above. Then, for every $\mathcal{S} \subseteq \mathcal{V}_0$ and $\mathcal{T} \subseteq \mathcal{V}_1$, we have*

$$\left| \widetilde{\mathcal{E}}(\mathcal{S} \times \mathcal{T}) - \frac{d|\mathcal{S}||\mathcal{T}|}{N} \right| \leq \lambda \sqrt{|\mathcal{S}||\mathcal{T}|}, \tag{23}$$

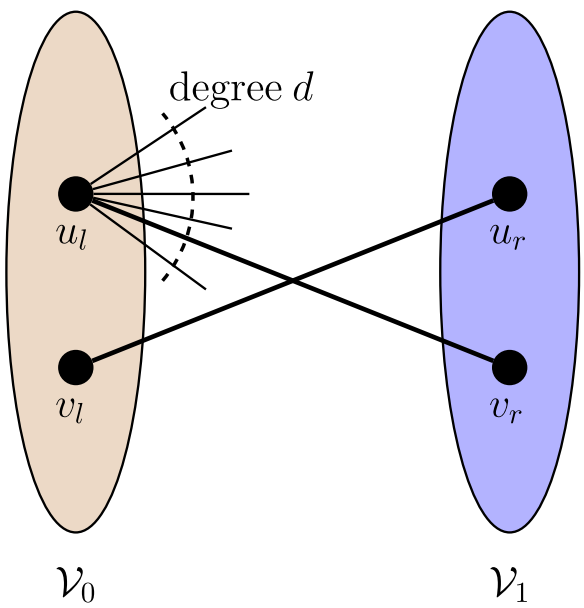

Fig. 3 Illustration of the double cover $\widetilde{\mathcal{G}}$ of the Δ-regular graph $\mathcal{G}$. An edge (u, v) in the original graph $\mathcal{G}$ contributes to two edges (u_l, v_r) and (v_l, u_r) in the the bipartite graph $\widetilde{\mathcal{G}}$

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

Given the bipartite graph $\widetilde{\mathcal{G}}$ and a code $\mathcal{C}_0$ with Δ-symbol long codewords, we define a code $\mathcal{C}$ as a slight generalization of the method of Section 6.1. Each edge in $\widetilde{\mathcal{G}}$ corresponds to a code symbol in the codewords of $\mathcal{C}$. For each node in the bipartite graph $\widetilde{\mathcal{G}}$, the Δ code symbols associated with the Δ edges incident on the node constitute a codeword in the code $\mathcal{C}_0$. Note that we assume the local code $\mathcal{C}_0$ to be an MDS code throughout this paper. In Fig. 4, we present an algorithm which corrects any ℓ erasures in $\mathcal{C}$ an cooperative manner by contacting at most $\ell\Delta \cdot \text{rate}(\mathcal{C}_0)$ code symbols. The algorithm alternates between the left nodes $\mathcal{V}_0$ and the right nodes $\mathcal{V}_1$ in order to utilize the smaller code $\mathcal{C}_0$ associated with the vertices to correct the erasures.

Let $\mathcal{S}^1 \subseteq \mathcal{V}_0$ denote 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 $\mathcal{S}^1$ has at least $d_{\min}(\mathcal{C}_0)$ erasures among the code symbols associated with its Δ edges. Therefore, we have

$$|\mathcal{S}^1| \leq \frac{\ell}{d_{\min}(\mathcal{C}_0)}. \tag{24}$$

We use $\mathcal{S}^i$ for $i \geq 2$ to denote the set of (left or right) vertices that have erasures among the Δ code symbols associated with them in the beginning of ith round and did not attempt to correct those erasures. Note that $\mathcal{S}^i \subseteq \mathcal{V}_0$ and $\mathcal{S}^i \subseteq \mathcal{V}_1$ when i is an odd and even round of decoding, respectively. Next, we employ the expander mixing lemma (cf. Lemma 1) to show that $\{|\mathcal{S}^1|, |\mathcal{S}^2|, |\mathcal{S}^3|, \dots\}$ is a strictly decreasing sequence.

Lemma 2. *Let $\mathcal{S}^1, \mathcal{S}^2, \dots$ be the sequence of sets of (left or right) vertices in the bipartite graph $\widetilde{\mathcal{G}}$ as defined above. Assume that the minimum distance of $\mathcal{C}_0$ is at least $(1+\epsilon)\lambda$ and $\ell \leq \frac{N\lambda\epsilon\delta}{2} = \frac{n\lambda\epsilon\delta}{2\Delta}$. Then, for $i \geq 1$, we have*

$$|\mathcal{S}^{i+1}| \leq \frac{|\mathcal{S}^i|}{1+\epsilon}. \tag{25}$$

Proof. We prove the relation in (25) 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 $\mathcal{S}^1$. By the definition of the set $\mathcal{S}^2$, it has at least $d_{\min}(\mathcal{C}_0)$ erasures among the Δ code symbols associated with it after the first round of decoding. In other words, this implies that each vertex in the set $\mathcal{S}^2$ has at least $d_{\min}(\mathcal{C}_0)$ edges

Algorithm: Erasure correction in $\mathcal{C}$.

Input: A codeword from $\mathcal{C}$ with at most ℓ erasures.

1: $j = 0$.
2: **while** not all the erasures are corrected **do**
3: For every vertex $u \in \mathcal{V}_j$ such that $1 \leq e \leq d_{\min}(\mathcal{C}_0) - 1$ code symbols among Δ code symbols associated with it are in erasure, use erasure correcting algorithm for $\mathcal{C}_0$ to recover from these erasures.
4: $j = j + 1 \pmod 2$.
5: **end while**

Fig. 4 Cooperative erasure correction in the code based on the double cover of a regular expander graph

incident on it which are emanating from the vertices from the set $\mathcal{S}^1$. Therefore, we have

$$\begin{aligned} |\mathcal{S}^2| d_{\min}(\mathcal{C}_0) &\leq |\widetilde{\mathcal{E}}\left(\mathcal{S}^1 \times \mathcal{S}^2\right)| \\ &\overset{(a)}{\leq} \frac{\Delta |\mathcal{S}^1||\mathcal{S}^2|}{N} + \lambda \sqrt{|\mathcal{S}^1||\mathcal{S}^2|} \\ &\overset{(b)}{\leq} \frac{\Delta |\mathcal{S}^1||\mathcal{S}^2|}{N} + \lambda \frac{|\mathcal{S}^1| + |\mathcal{S}^2|}{2} \\ &\overset{(c)}{\leq} \frac{\Delta \ell |\mathcal{S}^2|}{N \cdot d_{\min}(\mathcal{C}_0)} + \lambda \frac{|\mathcal{S}^1| + |\mathcal{S}^2|}{2}, \end{aligned} \tag{26}$$

where (a) and (c) follows from Lemma 1 and (24), respectively. Note that we employ the AM-GM inequality to obtain (b). It follows from (26) that

$$\left|\mathcal{S}^2\right| \leq \frac{\lambda}{2 \cdot d_{\min}(\mathcal{C}_0) - \lambda - 2\Delta\ell/(N \cdot d_{\min}(\mathcal{C}_0))} \left|\mathcal{S}^1\right|. \tag{27}$$

By replacing $d_{\min}(\mathcal{C}_0) = \delta\Delta$ in (27), we get

$$\left|\mathcal{S}^2\right| \leq \frac{\lambda}{2\delta\Delta - \lambda - 2\ell/(N\delta)} \left|\mathcal{S}^1\right|. \tag{28}$$

Under our assumption that $\frac{2\ell}{N\delta} \leq \epsilon\lambda$, it follows from (28) that

$$\left|\mathcal{S}^2\right| \leq \frac{\lambda}{2\delta\Delta - (1+\epsilon)\lambda} \left|\mathcal{S}^1\right|. \tag{29}$$

Now, under the assumption that $\delta\Delta \geq (1+\epsilon)\lambda$, we get from (29) that

$$\left|\mathcal{S}^2\right| \leq \frac{\left|\mathcal{S}^1\right|}{1+\epsilon}. \tag{30}$$

□

It follows from the Lemma 2 that in at most logarithmic (in ℓ) rounds of decoding, the algorithm described in Fig. 4 can correct ℓ erasures.

The codes based on the double covers of Δ-regular expander graphs have been studied in the coding theory literature before (see, e.g., [30]). The rate and the minimum distance of the code $\mathcal{C}$ depends on the rate and the minimum distance of the code $\mathcal{C}_0$. Note that $\mathcal{C}_0$ characterizes the local constraints associated with the vertices in the bipartite graph $\widetilde{\mathcal{G}}$. In particular, if $\text{rate}(\mathcal{C}_0) = R$ and $d_{\min}(\mathcal{C}_0) = \delta\Delta$, then we have that $\text{rate}(\mathcal{C}) \geq 2R - 1$ and $d_{\min}(\mathcal{C}) \geq \delta(\delta - \frac{\lambda}{\Delta})n$ [28, 30].

As we show in this section, for an $\epsilon > 0$ and local code $\mathcal{C}_0$ such that $d_{\min}(\mathcal{C}_0) = \delta\Delta \geq (1+\epsilon)\lambda$, it is possible to correct $\ell \leq \frac{N\lambda\epsilon\delta}{2}$ erasures using the algorithm described in Fig. 4. Moreover, in the worst correction of each erasure involves contacting at most $\text{rate}(\mathcal{C}_0)\Delta \leq (1 + \text{rate}(\mathcal{C}))\Delta/2$ other intact code symbols (assuming that the local code $\mathcal{C}_0$ is an MDS code). Therefore, the codes based on the double cover of a Δ-regular expander graph and a local code $\mathcal{C}_0$ have (r, ℓ)-cooperative locality for any $\ell \leq \frac{N\lambda\epsilon\delta}{2} = \frac{n\lambda\epsilon\delta}{2\Delta}$ and $r = \ell(1 + \text{rate}(\mathcal{C}))\Delta/2$, that is,

$$\text{rate}(\mathcal{C}) \geq \frac{2r}{\ell\Delta} - 1.$$

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.

7 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) [32] and have multiple disjoint repair groups for each code symbols. 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 $(r = \ell + 1, \ell)$-cooperative locality for any $\ell \leq \frac{n-1}{2}$.

An $[n = 2^k - 1, k, 2^{k-1}]_2$ punctured Hadamard code encodes a k bits long message $(m_1, m_2, \ldots, m_k)$ to an $n = 2^k - 1$ codeword $\mathbf{c} = \left(c_1, c_2, \ldots, c_{n=2^k-1}\right)$ such that

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

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

Theorem 4. *In an* $\left[n = 2^k - 1, k, 2^{k-1}\right]_2$ *punctured Hadamard code, any* $1 \leq \ell \leq \frac{n-1}{2}$ *erasures can be corrected by contacting at most* $\ell + 1$ *other code symbols.*

Proof. We prove the theorem by using induction over k. For base case, we consider $k = 2$, where the $[n = 3 = 2^2 - 1, 2, 2]_2$ 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 $1 \leq \ell \leq \frac{3-1}{2} = 1$ 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 $\left[n = 2^k - 1, k, 2^{k-1}\right]_2$ punctured Hadamard code of dimension k, and two cases regarding the positions of ℓ erased code symbols.

- *Case 1: There are* $x \leq 2^{k-2} - 1$ *erasures among the first* $\widehat{n} = 2^{k-1} - 1$ *code symbols.* Note that the first $\widehat{n} = 2^{k-1} - 1$ code symbols constitute a codeword of an $\left[\widehat{n} = 2^{k-1} - 1, k-1, 2^{k-2}\right]_2$ punctured Hadamard code. Therefore, from the inductive hypothesis, one can correct the x erasures among the first $\widehat{n}$ code symbols by contacting $x+1$ other code symbols out of these $\widehat{n}$ code symbols. Now, if the symbol $c_{2^{k-1}}$ in erasure, we can recover it by contacting one of the intact symbol among $\left\{c_{2^{k-1}+1}, c_{2^{k-1}+2}, \ldots, c_{n=2^k-1}\right\}$ say $c_{2^{k-1}+j}$ and the corresponding code symbol c_j from the first $\widehat{n}$ code symbols. Now, we can repair the remaining erased symbols among $\left\{c_{2^{k-1}+1}, c_{2^{k-1}+2}, \ldots, c_{n=2^k-1}\right\}$ from $c_{2^{k-1}}$ and the corresponding code symbol among the first $\widehat{n}$ 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 ℓ erasures.
- *Case 2: There are* $x \geq 2^{k-2}$ *erasures among the first* $\widehat{n} = 2^{k-1} - 1$ *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 $\frac{n-1}{2} = 2^{k-1}$ distinct pairs of code symbols $\left\{c_i, c_{2^{k-1}+i}\right\}_{i \in [2^{k-1}]}$ that can recover $c_{2^{k-1}}$. Since we have at most $\frac{n-1}{2} - 1 = 2^{k-1} - 1$ erasures apart from $c_{2^{k-1}}$, one of the 2^{k-1} pairs $\left\{c_i, c_{2^{k-1}+i}\right\}_{i \in [2^{k-1}]}$ 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 $\left\{c_{2^{k-1}+1}, c_{2^{k-1}+2}, \ldots, c_{n=2^k-1}\right\}$. Similarly, we can add m_k to any of the first $\widehat{n} = 2^{k-1} - 1$ 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 ℓ_1 erasures by contacting at most $\ell_1 + 1$ code symbols.

Combining both cases completes the proof. □

Remark 12. *Note that, for each symbol, the punctured Hadamard code provides* $\frac{n-1}{2}$ *disjoint repair groups. Moreover, each of these repair groups comprises two symbols. Therefore, it easily follows from the discussion of Section 2.1 that the punctured Hadamard code has* $(2\ell, \ell)$*-cooperative locality for* $\ell \leq \frac{n-1}{2}$*. Here, we show that these codes allow for more efficient cooperative local repair mechanism by establishing* $(\ell + 1, \ell)$*-cooperative locality for them.*

8 Conclusions

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 ℓ erasures are uniformly distributed among the code symbols. Moreover, we assume r and ℓ to be large enough. In that case, we claim that even the simple partition codes of Section 4.1 are asymptotically optimal. This is true, because with reasonably high probability (depending on r and ℓ), every local group (a total p of them) experiences less than about $t \equiv \Theta\left(\frac{\ell}{p} \log \frac{\ell}{p}\right)$ number of erasures. Therefore, with high probability, one can perform cooperative local repair of ℓ random erasures even if an $\left(\frac{r}{\ell} + t, \frac{r}{\ell}\right)$ MDS code in employed in the construction of the partition code (cf. Section 4.1). This translates to a coding scheme with the overall rate of $\frac{r}{r+\ell t}$. One can take p large enough to optimize this value. Indeed, it is possible to attain a rate of $\frac{r}{r+\ell^{1+\epsilon}}$ for some $\epsilon > 0$. Comparing with (7), we see that partition codes are near-optimal in this case. Here, note that it was shown in [33] that for a random erasure channel, the partition codes are asymptotically optimal in terms of achieving capacity.

8.1 Part of the proof of Theorem 1

Before proceeding with the analysis, we argue the correctness of the algorithm in Fig. 1. Note that it is always possible to find ℓ coordinates $\left\{i_1^j, i_2^j, \ldots, i_\ell^j\right\}$ at line 5. When the algorithm reaches line 5, the sub-code $\mathcal{C}_{j-1}$ has more than q^ℓ codewords. Therefore, there must be at least ℓ coordinates in the codewords in $\mathcal{C}_{j-1}$ that are not fixed in the previous iterations. This also implies that, for $m \in [\ell]$,

$$i_m^j \notin \mathcal{I}_{j-1} := \bigcup_{j' \in [j-1]} \left(\mathcal{R}_{j'} \cup \left\{i_1^{j'}, \ldots, i_\ell^{j'}\right\}\right) \subset [n]. \quad (31)$$

Note that the code symbols indexed by $\mathcal{I}_{j-1}$ are fixed in $\mathcal{C}_{j-1}$. This further implies that

$$\mathcal{R}_j = \Gamma_{\left\{i_1^j, \ldots, i_\ell^j\right\}} \not\subset \mathcal{I}_{j-1},$$

i.e., not all of the code symbols contacted to repair the ℓ symbols indexed by the set $\left\{i_1^j, \ldots, i_\ell^j\right\}$ can be fixed in the previous iterations. Otherwise, the symbols indexed by the set $\left\{i_1^j, \ldots, i_\ell^j\right\}$ would also have been fixed in the previous iterations.

For the construction of a sub-code as described in Fig. 1, we define $\mathcal{A}_j = \mathcal{I}_j \backslash \mathcal{I}_{j-1} \subseteq \mathcal{R}_j \cup \left\{i_1^j, \ldots, i_\ell^j\right\}$ and $a_j = |\mathcal{A}_j|$. Assuming that the while loop in Fig. 1 ends with $j = t$, for $j \in [t]$, we have

$$\mathcal{I}_j = \bigcup_{j' \in [j]} \mathcal{A}_{j'},$$

where we take union of the disjoint sets $\mathcal{A}_{j'}$, $j' \in [j]$.

Note that none of the indices in the set $\left\{i_1^j, \ldots, i_\ell^j\right\}$ corresponds to fixed symbols. Thus, by the definition of $\mathcal{A}_j$ and a_j, only $a_j - \ell$ code symbols among the code symbols indexed by the set $\mathcal{R}_j$ are not fixed in the previous iterations. Hence, at line 7, there are at most $q^{a_j - \ell}$ possibilities for $\mathbf{y}_j$. This implies that

$$|\mathcal{C}_j| \geq |\mathcal{C}_{j-1}|/q^{a_j - \ell}. \quad (32)$$

The construction of the subcode $\mathcal{C}'$ can end at either line 10 or line 14. Here, we analyze only the case when the construction ends at line 10. (The similar analysis holds for the other case as well). In this case, we have $|\mathcal{C}_t| \leq q^\ell$, or

$$\ell \geq \log_q |\mathcal{C}_t| \geq k - \sum_{j=0}^{t-1} \left(a_{j+1} - \ell\right). \quad (33)$$

Now, using that $a_j \leq |\mathcal{A}_j| \leq |\mathcal{R}_j \cup \left\{i_1^j, \ldots, i_\ell^j\right\}| \leq r + \ell$, we get

$$k - \ell \leq \sum_{j=0}^{t-1} \left(a_{j+1} - \ell\right) \leq tr. \quad (34)$$

This implies that

$$t \geq \left\lfloor \frac{k-\ell}{r} \right\rfloor. \quad (35)$$

Furthermore, for the case when we have $r \geq \ell$, it follows from (34) that

$$t \geq \left\lceil \frac{k}{r} \right\rceil - 1. \quad (36)$$

Note that sub-code $\mathcal{C}' = \mathcal{C}_t$. Therefore,

$$\begin{aligned} \log_q |\mathcal{C}'| &= \log_q |\mathcal{C}_t| \\ &\geq \log_q |\mathcal{C}| - \sum_{j=0}^{t-1} \left(a_{j+1} - \ell\right) \\ &= k - \sum_{j=0}^{t-1} a_{j+1} + t\ell \\ &\overset{(a)}{=} k - |\mathcal{I}_t| + t\ell \end{aligned} \quad (37)$$

where (a) follows from the fact that $\mathcal{I}_t$ is union of the disjoint sets $\mathcal{A}_j$.

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

$$\begin{aligned} d_{\min}(C) \leq d_{\min}(\mathcal{C}'') &\leq n - |\mathcal{I}_t| - \log_q |\mathcal{C}''| + 1 \\ &\leq n - |\mathcal{I}_t| - (k - |\mathcal{I}_t| + t\ell) + 1 \\ &= n - k - t\ell + 1, \end{aligned} \quad (38)$$

It follows from (38) and (35) that

$$d_{\min}(\mathcal{C}) \leq n - k + 1 - \ell \left\lfloor \frac{k-\ell}{r} \right\rfloor. \quad (39)$$

For the setting where we have $r \geq \ell$, we can use (38) along with (36) to obtain that

$$d_{\min}(\mathcal{C}) \leq n - k + 1 - \ell \left(\left\lceil \frac{k}{r} \right\rceil - 1\right). \quad (40)$$

This completes the proof.

8.2 Proof of Proposition 1

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

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

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

For any codeword of $\mathcal{C}$ whose support is given by the vertex set $\mathcal{S} \subset \mathcal{U}$, we must have $U_t(\mathcal{S}) = \emptyset$. Clearly, when $|\mathcal{S}| \leq \alpha n$, $|U_t(\mathcal{S})| \geq (1 - \epsilon - \epsilon/t)h|\mathcal{S}| > 0$. Let us assume

$|\mathcal{S}| > \alpha n$ but $|\mathcal{S}| \leq (2-\epsilon-\epsilon/t)\alpha n$. Let $\mathcal{Q}$ be a proper subset of $\mathcal{S}$ such that $|\mathcal{Q}| = \alpha n$. The number of edges coming out of $\mathcal{S} \setminus \mathcal{Q}$ is $h(|\mathcal{S}| - \alpha n) \leq h(1-\epsilon-\epsilon/t)\alpha n$. On the other hand, $U_t(\mathcal{Q}) \geq (1-\epsilon-\epsilon/t)h\alpha n$. Hence, $U_t(\mathcal{S}) \neq \emptyset$.

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

Endnotes

[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.

[2]This assumption is just for the ease of exposition and a similar construction for odd ℓ can also be proposed.

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

[4]As shown above, $\alpha n \geq \ell$ is a sufficient condition for the code obtained from the bipartite graph $\mathcal{G}$ to be able to allow for cooperative repair of ℓ erasures.

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

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
A. Mazumdar's research in this paper is supported by NSF CAREER grant CCF 1453121 and grant CCF1318093. S. Vishwanath would like to acknowledge support from Army Research Office under grant W911NF1110258. A part of this paper was presented at the 48th Annual Conference on Information Sciences and Systems on March 2014.

Author details
[1]Department of Electrical and Computer Engineering, The University of Texas at Austin, 78751 Austin, TX, USA. [2]Department of Electrical and Computer Engineering, University of Minnesota Twin Cities, 55455 Minneapolis, MN, USA.

Received: 3 July 2015 Accepted: 29 November 2015
Published online: 23 December 2015

References

1. AG Dimakis, P Godfrey, Y Wu, M Wainwright, K Ramchandran, Network coding for distributed storage systems. IEEE Trans. Inf. Theory. **56**(9), 4539–4551 (2010). doi:10.1109/TIT.2010.2054295
2. K Rashmi, N Shah, P Kumar, Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction. IEEE Trans. Inf. Theory. **57**, 5227–5239 (2011). doi:10.1109/TIT.2011.2159049
3. I Tamo, Z Wang, J Bruck, Zigzag codes: MDS array codes with optimal rebuilding. IEEE Trans. Inf. Theory. **59**(3), 1597–1616 (2013). doi:10.1109/TIT.2012.2227110
4. DS Papailiopoulos, AG Dimakis, V Cadambe, Repair optimal erasure codes through Hadamard designs. IEEE Trans. Inf. Theory. **59**(5), 3021–3037 (2013). doi:10.1109/TIT.2013.2241819
5. O Khan, R Burns, J Park, C Huang, in *Proceedings of the 3rd USENIX Conference on Hot Topics in Storage and File Systems. HotStorage'11*. In search of I/O-optimal recovery from disk failures (USENIX Association, Berkeley, CA, USA, 2011). http://dl.acm.org/citation.cfm?id=2002218.2002224
6. P Gopalan, C Huang, H Simitci, S Yekhanin, On the locality of codeword symbols. IEEE Trans. Inf. Theory. **58**(11), 6925–6934 (2012). doi:10.1109/TIT.2012.2208937
7. DS Papailiopoulos, AG Dimakis, Locally Repairable Codes. IEEE Trans. Inf. Theory. **60**(10), 5843-5855 (2014). doi:10.1109/TIT.2014.2325570
8. F Oggier, A Datta, in *Proceedings 2011 IEEE INFOCOM*. Self-repairing homomorphic codes for distributed storage systems (IEEE Press, Piscataway, NJ, USA, 2011), pp. 1215–1223. doi:10.1109/INFCOM.2011.5934901
9. AS Rawat, OO Koyluoglu, N Silberstein, S Vishwanath, Optimal locally repairable and secure codes for distributed storage systems. IEEE Trans. Inf. Theory. **60**(1), 212–236 (2014)
10. GM Kamath, N Prakash, V Lalitha, PV Kumar, Codes with local regeneration and erasure correction. IEEE Trans. Inf. Theory. **60**(8), 4637–4660 (2014). doi:10.1109/TIT.2014.2329872
11. I Tamo, A Barg, A family of optimal locally recoverable codes. IEEE Trans. Inf. Theory. **60**(8), 4661–4676 (2014). doi:10.1109/TIT.2014.2321280
12. D Ford, F Labelle, F Popovici, M Stokely, V-A Truong, L Barroso, C Grimes, S Quinlan, in *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*. Availability in globally distributed storage systems (USENIX, Berkeley, CA, 2010)
13. AM Kermarrec, NL Scouarnec, G Straub, in *Proc. International Symposium on Network Coding*. Repairing multiple failures with coordinated and adaptive regenerating codes (IEEE Press, Piscataway, NJ, USA, 2011)
14. KW Shum, Y Hu, Cooperative regenerating codes. IEEE Trans. Inf. Theory. **59**(11), 7229–7258 (2013)
15. F Lazebnik, VA Ustimenko, AJ Woldar, A new series of dense graphs of high girth. Bull. Am. Math. Soc. **32**(1), 73–79 (1995)
16. N Prakash, GM Kamath, V Lalitha, PV Kumar, in *Proc. 2012 IEEE International Symposium on Information Theory (ISIT)*. Optimal linear codes with a local-error-correction property (IEEE Press, Piscataway, NJ, USA, 2012), pp. 2776–2780. doi:10.1109/ISIT.2012.6284028
17. L Pamies-Juarez, HDL Hollmann, F Oggier, in *Proc. 2013 IEEE International Symposium on Information Theory (ISIT)*. Locally repairable codes with multiple repair alternatives (IEEE Press, Piscataway, NJ, USA, 2013), pp. 892–896. doi:10.1109/ISIT.2013.6620355
18. I Tamo, A Barg, in *Proc. 2014 IEEE International Symposium on Information Theory (ISIT)*. Bounds on locally recoverable codes with multiple recovering sets (IEEE Press, Piscataway, NJ, USA, 2014), pp. 691–695. doi:10.1109/ISIT.2014.6874921
19. A Wang, Z Zhang, Repair locality with multiple erasure tolerance. IEEE Trans. Inf. Theory. **60**(11), 6979–6987 (2014). doi:10.1109/TIT.2014.2351404
20. AS Rawat, DS Papailiopoulos, AG Dimakis, S Vishwanath, in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*. Locality and availability in distributed storage (IEEE Press, Piscataway, NJ, USA, 2014), pp. 681–685
21. DS Papailiopoulos, J Luo, AG Dimakis, C Huang, J Li, in *Proceedings 2012 IEEE INFOCOM*. Simple regenerating codes: network coding for cloud storage (IEEE Press, Piscataway, NJ, USA, 2012), pp. 2801–2805. doi:10.1109/INFCOM.2012.6195703
22. V Cadambe, A Mazumdar, in *2013 International Symposium on Network Coding (NetCod)*. An upper bound on the size of locally recoverable codes (IEEE Press, Piscataway, NJ, USA, 2013), pp. 1–5. doi:10.1109/NetCod.2013.6570829
23. M Forbes, S Yekhanin, On the locality of codeword symbols in non-linear codes. Discrete Math. **324**, 7 (2014). doi:10.1016/j.disc.2014.01.016, http://dx.doi.org/10.1016/j.disc.2014.01.016
24. N Prakash, V Lalitha, PV Kumar, in *Proc. 2014 IEEE International Symposium on Information Theory (ISIT)*. Codes with locality for two erasures (IEEE Press, Piscataway, NJ, USA, 2014), pp. 1962–1966. doi:10.1109/ISIT.2014.6875176
25. F Lazebnik, VA Ustimenko, Explicit construction of graphs with an arbitrary large girth and of large size. Discret. Appl. Math. **60**(1–3), 275–284 (1995)
26. R Tanner, A recursive approach to low complexity codes. IEEE Trans. Inf. Theor. **27**(5), 533–547 (1981). doi:10.1109/TIT.1981.1056404
27. A Orlitsky, R Urbanke, K Viswanathan, J Zhang, in *Proc. 2002 IEEE International Symposium on Information Theory (ISIT)*. Stopping sets and the girth of tanner graphs (IEEE Press, Piscataway, NJ, USA, 2002), pp. 2–2
28. M Sipser, DA Spielman, Expander codes. IEEE Trans. Inf. Theory. **42**(6), 1710–1722 (1996). doi:10.1109/18.556667
29. M Capalbo, O Reingold, S Vadhan, A Wigderson, in *Proceedings of the Thiry-fourth Annual ACM Symposium on Theory of Computing. STOC '02*. Randomness conductors and constant-degree lossless expanders

(ACM, New York, NY, USA, 2002), pp. 659–668. doi:10.1145/509907.510003, http://doi.acm.org/10.1145/509907.510003
30. G Zemor, On expander codes. IEEE Trans. Inf. Theory. **47**(2), 835–837 (2001). doi:10.1109/18.910593
31. N Alon, FRK Chung, Explicit construction of linear sized tolerant networks. Discrete Math. **72**(1-3), 15–19 (1988). doi:10.1016/0012-365X(88)90189-6
32. S Yekhanin, *Locally Decodable Codes and Private Information Retrieval Schemes*. (Springer, New York, USA, 2010). doi:10.1007/978-3-642-14358-8
33. A Mazumdar, V Chandar, GW Wornell, in *Proc. 2013 Information Theory and Applications Workshop (ITA)*. Local recovery properties of capacity achieving codes (IEEE Press, Piscataway, NJ, USA, 2013), pp. 1–3. doi:10.1109/ITA.2013.6502958