SCIENCE CHINA
Information Sciences

• REVIEW •

October 2018, Vol. 61 100301:1–100301:45
https://doi.org/10.1007/s11432-018-9482-6

Special Focus on Distributed Storage Coding

Erasure coding for distributed storage: an overview†

S. B. BALAJI[1], M. Nikhil KRISHNAN[1], Myna VAJHA[1], Vinayak RAMKUMAR[1], Birenjith SASIDHARAN[1] & P. Vijay KUMAR[1,2]

[1]*Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore* 560012*, India;*
[2]*Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles* CA90089*, USA*

Received 5 April 2018/Revised 4 May 2018/Accepted 7 July 2018/Published online 6 September 2018

Abstract In a distributed storage system, code symbols are dispersed across space in nodes or storage units as opposed to time. In settings such as that of a large data center, an important consideration is the efficient repair of a failed node. Efficient repair calls for erasure codes that in the face of node failure, are efficient in terms of minimizing the amount of repair data transferred over the network, the amount of data accessed at a helper node as well as the number of helper nodes contacted. Coding theory has evolved to handle these challenges by introducing two new classes of erasure codes, namely regenerating codes and locally recoverable codes as well as by coming up with novel ways to repair the ubiquitous Reed-Solomon code. This survey provides an overview of the efforts in this direction that have taken place over the past decade.

Keywords distributed storage, regenerating codes, locally recoverable codes, codes with locality, erasure codes, node repair

Citation Balaji S B, Krishnan M N, Vajha M, et al. Erasure coding for distributed storage: an overview. Sci China Inf Sci, 2018, 61(10): 100301, https://doi.org/10.1007/s11432-018-9482-6

1 Introduction

This survey article deals with the use of erasure coding for the reliable and efficient storage of large amounts of data in settings such as that of a data center. The amount of data stored in a single data center can run into tens or hundreds of petabytes. Reliability of data storage is ensured in part by introducing redundancy in some form, ranging from simple replication to the use of more sophisticated erasure-coding schemes such as Reed-Solomon (RS) codes. Minimizing the storage overhead that comes with ensuring reliability is a key consideration in the choice of erasure-coding scheme. More recently a second problem has surfaced, namely, that of node repair. In [1, 2], the authors study the Facebook warehouse cluster and analyze the frequency of node failures as well as the resultant network traffic relating to node repair. It was observed in [1] that a median of 50 nodes is unavailable per day and that a median of 180 TB of cross-rack traffic is generated as a result of node unavailability. It was also reported that 98.08% of the cases have exactly one block missing in a stripe. The erasure code that was deployed in this instance was an $[n = 14, k = 10]$ RS code. Here n denotes the block length of the code and k the dimension. The conventional repair of an $[n, k]$ RS code is inefficient in that the repair of a single node, calls for contacting k other (helper) nodes and downloading k times the amount of data stored in the failed node, which is clearly inefficient. Thus there is significant practical interest in the design of erasure-coding techniques that offer both low overhead and which can also be repaired efficiently.

* Corresponding author (email: pvk1729@gmail.com)
† Invited article

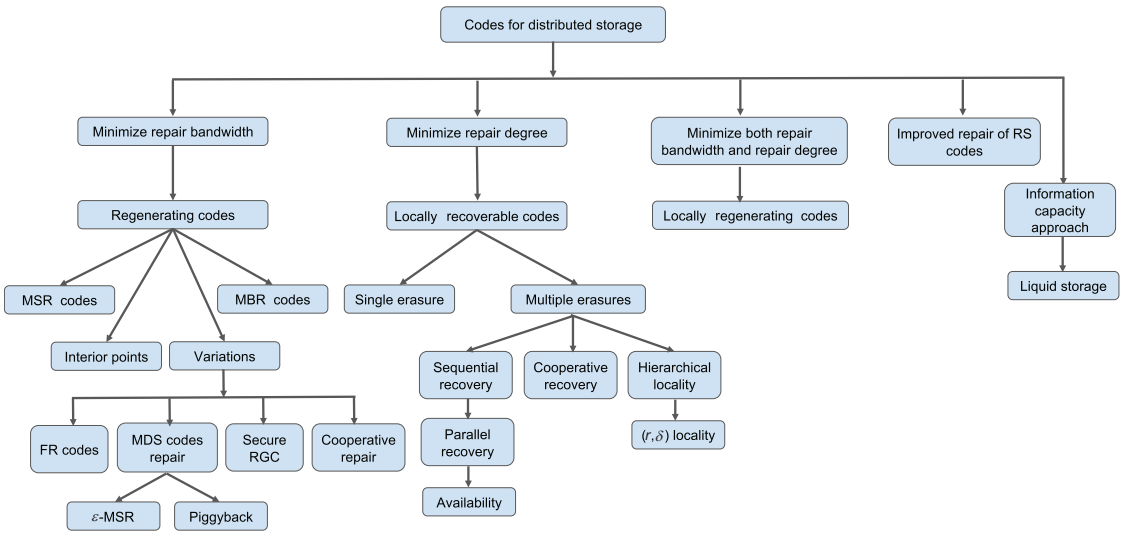

Figure 1 (Color online) An overview of the different classes of codes for distributed storage discussed in this survey article.

Coding theorists have responded to this need by coming up with two new classes of codes, namely regenerating (RG) and locally recoverable (LR) codes. The focus in an RG code is on minimizing the amount of data download needed to repair a failed node, termed the repair bandwidth while LR codes seek to minimize the number of helper nodes contacted for node repair, termed the repair degree. In a different direction, coding theorists have also re-examined the problem of node repair in RS codes and have come up with new and more efficient repair techniques. This survey provides an overview of these recent developments. An outline of the survey itself appears in Figure 1.

RG codes are discussed in Section 2. The two principal classes of RG codes, namely minimum bandwidth regenerating (MBR) and minimum storage regeneration (MSR) appear in the two sections that follow. These two classes of codes are at the two extreme ends of a tradeoff known as the storage-repair bandwidth (S-RB) tradeoff. A discussion on codes that correspond to the interior points of this tradeoff appears in Section 5. The theory of RG codes has been extended in several directions and these are explored in Section 6. Section 7 examines LR codes. There have been several approaches at extending the theory of LR codes to handle multiple erasures and these are dealt with in Section 8. A class of codes known as locally regenerating (LRG) codes that offer both low repair bandwidth and low repair degree within a single erasure code is discussed in Section 9. This is followed by Section 10 that discusses recent advances in the repair of RS codes. A brief description of a different approach based on capacity considerations and leading to the development of a liquid cloud storage system appears in Section 11. The final section, discusses practical evaluations and implementations.

Disclaimer. This survey is presented from the perspective of the authors and is biased in this respect. Given the explosion of research activity in this area, the survey also does not claim to be comprehensive and we offer our apologies to the authors whose work has inadvertently or for lack of space, not been appropriately cited. We direct the interested reader to some of the excellent surveys of codes on distributed storage contained in the literature including [3–6].

2 RG codes

Definition 1 ([7]). Let $\mathbb{F}_q$ denote a finite field of size q. Then an RG code $\mathcal{C}$ over $\mathbb{F}_q$ having integer parameter set $((n,k,d),(\alpha,\beta),B)$ where $1 \leqslant k \leqslant n-1$, $k \leqslant d \leqslant n-1$, $\beta \leqslant \alpha$, maps a file $\underline{u} \in \mathbb{F}_q^B$ on to a collection $\{\underline{c}_i\}_{i=1}^n$ of n α-tuples over $\mathbb{F}_q$ using an encoding map

$$E(\underline{u}) = [\underline{c}_1^{\mathrm{T}}, \underline{c}_2^{\mathrm{T}}, \ldots, \underline{c}_n^{\mathrm{T}}]^{\mathrm{T}}$$

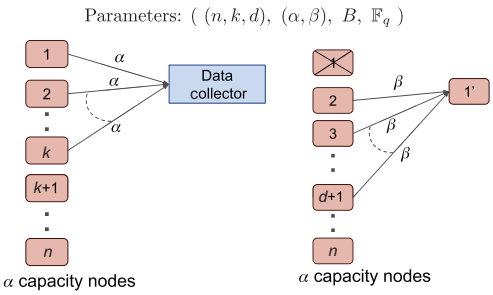

Figure 2 (Color online) An illustration of the data collection and node repair properties of an RG code.

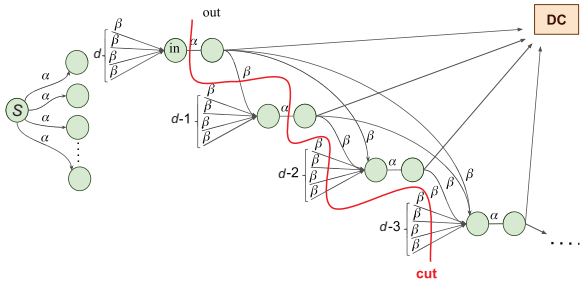

Figure 3 (Color online) The graph behind the cut-set file size bound.

with the α components of $\underline{c}_i$ stored on the i-th node in such a way that the following two properties (see Figure 2) are satisfied:

• Data collection. the message $\underline{u}$ can be uniquely recovered from the contents $\{c_{ij}\}_{j=1}^{k}$ of any k nodes.

• Node repair. If the f-th node storing $\underline{c}_f$ fails, then a replacement node can (1) contact any subset $D \subseteq [n] \setminus \{f\}$ of the remaining $(n-1)$ nodes of size $|D| = d$; (2) map the α contents $\underline{c}_h$ of each helper node $h \in D$ on to a collection of β repair symbols $\underline{a}_{h,f}^{D} \in \mathbb{F}_q^{\beta}$; (3) pool together the $d\beta$ repair symbols thus computed to use them to create a replacement vector $\hat{\underline{c}}_f \in \mathbb{F}_q^{\alpha}$ whose α components are stored in the replacement node, in such a way that the contents of the resultant nodes, with the replacement node replacing the failed node, once again forms an RG code.

An RG code is said to be exact-repair (ER) RG code if the contents of the replacement node are exactly same as that of the failed node, ie., $\hat{\underline{c}}_f = \underline{c}_f$. Else the code is said to be functional-repair (FR) RG code. An RG code is said to be linear if (1) $E(\underline{u}_1 + \theta\underline{u}_2) = E(\underline{u}_1) + \theta E(\underline{u}_2)$, $\underline{u}_1, \underline{u}_2 \in \mathbb{F}_q^B$, $\theta \in \mathbb{F}_q$ and (2) the map mapping the contents $\underline{c}_h$ of the h-th helper node on to the corresponding β repair symbols $\underline{a}_{h,f}^{D}$ is linear over $\mathbb{F}_q$.

Thus an RG code is a code over a vector alphabet $\mathbb{F}_q^{\alpha}$ and the quantity α is termed the sub-packetization level of the RG code. The total number $d\beta$ of $\mathbb{F}_q$ symbols to be transferred for repair of failure node is called the repair bandwidth of the RG code. The rate of the RG code is given by $R = \frac{B}{n\alpha}$. Its reciprocal $\frac{n\alpha}{B}$ is the storage overhead.

2.1 Cut-set bound

Let us assume that $\mathcal{C}$ is an FR RG code having parameter set $((n,\ k,\ d),\ (\alpha,\ \beta),\ B)$. Since an ER RG code is also an FR code, this subsumes the case when $\mathcal{C}$ is an ER RG code. Over time, nodes will undergo failures and every failed node will be replaced by a replacement node. Let us assume to begin with, that we are only interested in the behavior of the RG code over a finite-but-large number $N \gg n$ of node repairs. For simplicity, we assume that repair is carried out instantaneously. Then at any given time instant t, there are n functioning nodes whose contents taken together comprise an RG code. At this time instant, a data collector could connect to k nodes, download all of their contents and decode to recover underlying message vector $\underline{u}$. Thus in all, there are at most $N\binom{n}{k}$ distinct data collectors which are distinguished based on the particular set of k nodes to which the data collector connects.

Next, we create a source node that possesses the B message symbols $\{u_i\}_{i=1}^{B}$, and draw edges connecting the source to the initial set of n nodes. We also draw edges between the d helper nodes that assist a replacement node and the replacement node itself as well as edges connecting each data collector with the corresponding set of k nodes from which the data collector downloads data. All edges are directed in the direction of information flow. We associate a capacity β with edges emanating from a helper node to a replacement node and an ∞ capacity with all other edges. Each node can only store α symbols over $\mathbb{F}_q$. We take this constraint into account using a standard graph-theory construct, in which a node is replaced by 2 nodes separated by a directed edge (leading towards a data collector) of capacity α. We have in this way, arrived at a graph (Figure 3) in which there is one source S and at most $N\binom{n}{k}$ sinks $\{T_i\}$.

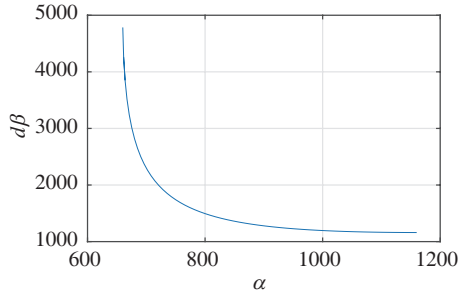

Figure 4 (Color online) Storage-repair bandwidth tradeoff. Here, $n = 60$, $k = 51$, $d = 58$, $B = 33660$.

Each sink T_i would like to be able to reconstruct all the B source symbols $\{u_i\}$ from the symbols it receives. This is precisely the multicast setting of network coding. A principal result in network coding tells us that in a multicast setting, one can transmit messages along the edges of the graph in such a way that each sink T_i is able to reconstruct the source data, provided that the minimum capacity of a cut separating S from T_i is $\geqslant B$. A cut separating S from T_i is simply a partition of the nodes of the network into 2 sets: A_i containing S and A_i^c containing T_i. The capacity of the cut is the sum of capacities of the edges leading from a node in A_i to a node in A_i^c. A careful examination of the graph will reveal that the minimum capacity Q of a cut separating a sink T_i from source S is given by $Q = \sum_{i=0}^{k-1} \min\{\alpha, (d-i)\beta\}$ (Figure 3 shows an example cut separating source from sink). This leads to the following upper bound on file size [7]:

$$B \leqslant \sum_{i=0}^{k-1} \min\{\alpha, (d-i)\beta\}. \tag{1}$$

Network coding also tells us that when only a finite number of regenerations take place, this bound is achievable and furthermore achievable using linear network coding, i.e., using only linear operations at each node in the network when the size q of the finite field $\mathbb{F}_q$ is sufficiently large. In a subsequent result [8], Wu established using the specific structure of the graph, that even in the case when the number of sinks is infinite, the upper bound in (1) continues to be achievable using linear network coding.

In summary, by drawing upon network coding, we have been able to characterize the maximum file size of an RG code given parameters $\{k, d, \alpha, \beta\}$ for the case of functional repair when there is constraint placed on the size q of the finite field $\mathbb{F}_q$. Note interestingly, that the upper bound on file size is independent of n. Quite possibly, the role played by n is that of determining the smallest value of field size q for which a linear network code can be found having file size B satisfying (1). A functional RG code having parameters $((n,\ k,\ d),\ (\alpha,\ \beta),\ B)$ is said to be optimal provided (a) the file size B achieves the bound in (1) with equality and (b) reducing either α or β will cause the bound in (1) to be violated.

2.2 Storage-repair bandwidth tradeoff

We have thus far, specified code parameters $(k, d)(\alpha, \beta)$ and asked what is the largest possible value of file size B. If however, we fix parameters (n, k, d, B) and ask instead what are the smallest values of (α, β) for which one can hope to achieve (1), it turns out, as might be evident from the form of the summands on the RHS of (1), that there are several pairs (α, β) for which equality holds in (1). In other words, there are different flavors of optimality.

For a given file size B, the storage overhead and normalized repair bandwidth are given respectively by $\frac{n\alpha}{B}$ and $\frac{d\beta}{B}$. Thus α reflects the amount of storage overhead while β determines the normalized repair bandwidth. The several pairs (α, β) for which equality holds in (1), represent a tradeoff between storage overhead on the one hand and normalized repair bandwidth on the other as can be seen from the example plot in Figure 4. Clearly, the smallest value of α for which the equality can hold in (1) is given by $\alpha = \frac{B}{k}$. Given $\alpha = \frac{B}{k}$, the smallest permissible value of β is given by $\beta = \frac{\alpha}{d-k+1}$. This represents the MSR point

and codes achieving (1) with $\alpha = \frac{B}{k}$ and $\beta = \frac{\alpha}{d-k+1}$ are known as MSR codes. At the other end of the tradeoff, we have the MBR code whose associated (α, β) values are given by $\beta = \frac{B}{dk-\binom{k}{2}}$, $\alpha = d\beta$.

Remark 1. Since an RG code can tolerate $(n-k)$ erasures by the data collection property, it follows that the minimum Hamming weight $d_{\min}$ of an RG code must satisfy $d_{\min} \geqslant (n-k+1)$. By the singleton bound, the largest size M of a code of block length n and minimum distance $d_{\min}$ is given by $M \leqslant Q^{n-d_{\min}+1} \leqslant Q^k$, where Q is the size of alphabet of the code. Since $Q = q^\alpha$ in the case of RG code, it follows that the size M of an RG code must satisfy $M \leqslant q^{k\alpha}$, or equivalently $q^B \leqslant q^{k\alpha}$, i.e., $B \leqslant k\alpha$. But $B = k\alpha$ in the case of an MSR code and it follows that an MSR code is a maximum distance separable (MDS) code over a vector alphabet. Such codes also go by the name MDS array code.

From a practical perspective, ER RG codes are easier to implement as the contents of the n nodes in operation do not change with time. Partly for this reason and partly for reasons of tractability, with few exceptions, most constructions of RG codes belong to the class of ER RG codes. Examples of FR RG code include the $d = (k+1)$ construction in [9] as well as the construction in [10].

Early constructions of RG codes focused on the two extreme points of the S-RB tradeoff, namely the MSR and MBR points. The various constructions of MBR and MSR codes are described in Sections 3 and 4. Not surprisingly, given the vast amount of data stored, the storage industry places a premium on low storage overhead. In this connection, we note that the maximum rate of an MBR code is given by

$$R_{\mathrm{MBR}} = \frac{B}{n\alpha} = \frac{(dk - \binom{k}{2})\beta}{nd\beta} = \frac{dk - \binom{k}{2}}{nd},$$

which can be shown to be upper bounded by $R_{\mathrm{MBR}} \leqslant \frac{1}{2}$ and is achieved when $k = d = (n-1)$. In the case of MSR codes, there is no such limitation and MSR codes can have rates approaching 1.

An RG code is said to be a help-by-transfer (HBT) RG code if repair of a failed node can be accomplished without incurring any computation at a helper node. If no computation is required at either helper node or at the replacement node, then the code is termed a repair-by-transfer (RBT) RG code. Clearly, an RBT RG code is also an HBT RG code.

3 MBR codes

Remark 2. If the B message symbols are drawn randomly with uniform distribution from $\mathbb{F}_q^B$, it can be shown that in any RG code achieving the cut-set bound, the contents of each node correspond to a random variable that is uniform over $\mathbb{F}_q^\alpha$. In an MBR code, repair is accomplished by downloading a total of just α symbols which clearly, is the minimum possible.

Remark 3. Let $\mathcal{C}$ be an MBR code. If $\mathcal{C}$ has the RBT property, it trivially follows that all scalar code-symbols of $\mathcal{C}$ are replicated at least twice. In [11], it is shown that for an MBR code it is not possible to have even a single scalar code-symbol replicated more than twice. Thus the RBT property implies that the collection of $n\alpha$ scalar code-symbols associated with a codeword represent a set of $\frac{n\alpha}{2}$ distinct code symbols, each repeated twice. The converse is not true in general. However when $d = (n-1)$, it can be shown that the two properties are equivalent.

Remark 4. In [12], it is shown that for $d < (n-1)$, it is not possible to construct an MBR code that has the HBT property.

3.1 Polygonal MBR codes

In the following, we describe with the help of an example, one of the first explicit families of MBR codes [13]. We term these codes as polygonal MBR codes. The construction holds for parameters $k \leqslant d = n-1$, $\beta = 1$ and the constructed MBR codes possess the RBT property.

Example 1. Consider the parameters $n = 5$, $k = 3$, $d = 4$ and $\beta = 1$. Thus $B = kd\beta - \binom{k}{2}\beta = 9$. First construct a complete graph with $n = 5$ vertices and $N = \binom{5}{2} = 10$ edges. The nine message symbols are then encoded using a $[10, 9]$ MDS code to produce ten code-symbols. Each code-symbol is then

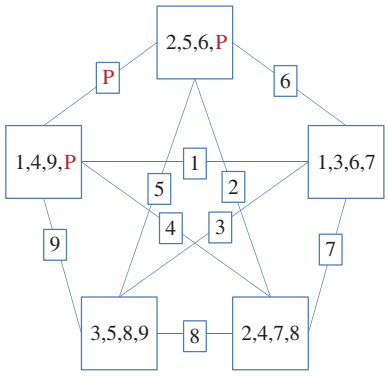

Figure 5 (Color online) An example RBT MBR code for the parameters $n = 5$, $k = 3$, $d = 4$. Here file size is 9.

uniquely assigned an edge. Each node of the MBR code stores the code-symbols corresponding to the edges incident on that node (Figure 5). The data collection property follows as any collection of $k = 3$ nodes yields nine distinct (MDS) code-symbols. If a node fails, the replacement node can download from each of the remaining four nodes, the code-symbol corresponding to the edge it shares with the failed node. Hence repair is accomplished by merely transferring the data without any computation (RBT).

Remark 5. For the general construction, in order to construct an $[n, k, d = n - 1]$, $\beta = 1$ MBR code, one first forms the complete graph on n vertices. Each edge is then mapped to a code-symbol of an $[N, B]$ MDS code, where $N = \binom{n}{2}$ and B is the file size parameter. An $O(n^2)$ field-size requirement is thus imposed by the underlying scalar MDS code.

3.2 Product-matrix (PM) MBR codes

A second, general construction for MBR codes is the PM construction [14] which derives its name from the fact that the contents of n nodes can be expressed in the form of a product of two matrices. The two matrices are respectively an encoding matrix and a second, message matrix containing the message symbols. This construction yields MBR codes for all feasible parameters $k \leqslant d \leqslant n - 1$, $\beta = 1$, with an $O(n)$ field-size requirement. The $(n \times d)$ encoding matrix ψ is of the form: $\psi = [\phi\ \Delta]$, where ϕ, Δ are $(n \times k)$, $(n \times (d - k))$ matrices, respectively. Let the i-th row of ψ be denoted by ψ_i^{T}. The sub-matrices ϕ and Δ are here chosen such that any d rows of ψ and any k rows of ϕ are linearly independent. The $(d \times d)$ symmetric message matrix M is derived from the $B = kd - \binom{k}{2}$ message symbols as $M = \left[\begin{smallmatrix} S & V \\ V^{\mathrm{T}} & 0 \end{smallmatrix}\right]$, where S is a symmetric $(k \times k)$ matrix and V a $(k \times (d - k))$ matrix.

The i-th node, under the PM-MBR construction, stores the matrix product $\psi_i^{\mathrm{T}} M$. The repair data passed on by helper node j to replacement node i is given by $\psi_j^{\mathrm{T}} M \psi_i$.

3.3 Other work

In [15], the authors introduce a family of RBT MBR codes for $d = n - 1$, that are constructed based on a congruent transformation applied to a skew-symmetric matrix of message symbols. In comparison with the $O(n^2)$ field requirement of polygonal MBR codes, in this construction, a field-size of $O(n)$ suffices. In [16], the authors stay within the PM framework, but provide a different set of encoding matrices for MSR and MBR codes that have least-possible update complexity within the PM framework. The authors of [16] also analyze the codes for their ability to correct errors and provide corresponding decoding algorithms. Ref. [12] proves the non-existence of HBT MBR codes with $d < (n - 1)$. The paper also provides PM-based constructions for two relaxations, namely (i) any failed node which is a part of a collection of systematic nodes can be recovered in HBT fashion from any d other nodes and (ii) for every failed node, there exists a corresponding set of d helper nodes which permit HBT repair. Ref. [11] provides binary MBR constructions for the parameters $(k = d = n - 2)$, $(k + 1 = d = n - 2)$ and studies the existence of MBR codes with inherent double replication, for all parameters. In [17], the authors provide regenerating-code constructions that asymptotically achieve the MSR or MBR point

as k increases and these codes can be constructed over any field, provided the file size is large enough. In [18], the authors introduce some extensions to the classical MBR framework by permitting the presence of a certain number of error-prone nodes during repair/reconstruction and by introducing flexibility in choosing the parameter d during node repair.

Open problems 1. Determine the smallest possible field size q of an MBR code for given $\{(n, k, d), (\alpha, \beta)\}$.

4 MSR codes

Among the class of RG codes, MSR codes have received the greatest attention, and the reasons include the fact that (a) the storage overhead of an MSR code can be made as small as desired, (b) MSR codes are MDS codes and (c) MSR codes have been challenging to construct.

4.1 Introduction

As noted previously, an MSR code with parameters (n, k, d, α) has file size $B = k\alpha$ and $\beta = \frac{\alpha}{d-k+1}$. Although MSR codes are vector MDS codes that have optimum repair-bandwidth of $d\beta$ for the repair of any node among the n nodes, there are papers in the literature that refer to a code as an MSR code even if optimal repair holds only for the systematic nodes. In the current paper, we refer to such codes as systematic MSR codes. While only β symbols are sent by each of the d helper nodes, the number of symbols accessed by the helper node in order to generate these β symbols could be $> \beta$. The class of MSR codes that access at each helper node, only as many symbols as are transferred, are termed optimal-access MSR codes. MSR codes that alter a minimum number of parity symbols while updating a single, systematic symbol, are called update-optimal MSR codes.

There are several ER MSR constructions available in the literature. Shah et al. [9] show that interference alignment (IA) is necessarily present in every ER MSR code, and use IA techniques to construct systematic MSR codes, known as MISER codes, for $d = n - 1 \geqslant 2k - 1$. The IA condition in the context of MSR codes (observed earlier in [19]) demands that the interference components in the data passed by helper nodes must be aligned so that they can be cancelled at the replacement node by data received from the systematic helper nodes. In [20], Suh et al. build on [9] to construct MSR codes for $d \geqslant 2k - 1$ with optimal repair bandwidth for all nodes, under the condition that the helper-node set necessarily includes systematic nodes. In [14], the well-known PM framework is introduced to provide MSR constructions for $d \geqslant 2k-1$, thereby settling the problem of MSR code construction in the low-rate regime, $k/n \leqslant 0.5$. While the method adopted in [14] to provide a construction for $d > 2k - 1$ is to suitably shorten a code for $d = 2k - 1$, an extension of the PM framework that yields constructions for any $d \geqslant 2k - 1$ in a single step is provided in [21]. Apart from a few notable constructions such as the Hadamard-design-based code [22] for $(k + 2, k)$ and its generalization for $(n - k) > 2$ for systematic node-repair, the problem of high-rate constructions (i.e., $k/n \geqslant 0.5$) for all-node repair remained open. The first major result in this direction, is due to Cadambe et al. [23] where the authors apply the notion of symbol extension in IA where multiple symbols are grouped together to form a single vector symbol, to jointly achieve IA. The symbol-extension viewpoint is then used to show that ER MSR codes exist for all (n, k, d), as B goes to infinity. The second major development was the zigzag code construction [24,25], the first non-asymptotic high-rate MSR code construction with $d = (n-1)$ permitting rates as close as 1 as desired, with additional desirable properties such as optimal access and optimal update. Zigzag codes however, require a sub-packetization level (α) that grows exponentially with k and a very large finite field size, while the earlier PM codes for the low-rate regime, have $\alpha = (k+1)$ and field-size that is linear in n. In a subsequent work [26], the authors present a systematic MSR construction having $\alpha = \frac{k^2}{4}$ and rate $R = 2/3$. A second systematic MSR code with $\alpha = r^{\frac{k}{r+1}}$ is presented in [27]. A lower bound on sub-packetization level α of a general MSR code is derived in [28]. The same paper shows that $\alpha \geqslant r^{\frac{k-1}{r}}$

in the case of an optimal-access MSR code. An improved lower bound for general MSR codes

$$2\log_2\alpha\left(\log_{(\frac{r}{r-1})}\alpha+1\right)+1\geqslant k \tag{2}$$

appears in [29]. These developments made it clear that the ultimate goal in MSR code construction was to construct a high-rate MSR code that simultaneously had low sub-packetization level α, low field-size q, arbitrary repair degree d and the optimal-access property.

In [30], a parity-check viewpoint is adopted to construct a high-rate MSR code for $d=n-1$ with a sub-packetization level $r^{\frac{n}{r}}$, requiring however, a large field-size. The construction was extended in [31], to d satisfying $k\leqslant d\leqslant n-1$. In [32], the authors provide a construction of MSR codes that holds for all $k\leqslant d\leqslant n-1$, but which once again required large field size. In [33], the authors provide a construction for an optimal-access systematic MSR code that holds for any parameter set $(n,k,d=n-1)$ having sub-packetization α matching the lower bound given in [28]. In [24–27, 30–33], combinatorial nullstellensatz [34] is used to prove the MDS property due to which the codes are non-explicit and have large field sizes.

In [35], an explicit optimal-access, systematic MSR code is constructed with optimal α, but for limited values of $n-k=2,3$. In [36], the authors present two different classes of explicit MSR constructions, one of which possessed the optimal-access property. Both constructions are for any (n,k,d) with sub-packetization level growing exponential in n.

In a major advance, Ye and Barg [37] present an explicit construction of a high-rate, optimal-access MSR code with $\alpha=r^{\lceil\frac{n}{r}\rceil}$, field size no larger than $r\lceil\frac{n}{r}\rceil$, and $d=(n-1)$. Essentially the same construction was independently rediscovered in [38] from a different coupled-layer perspective, where layers of an arbitrary MDS codes are coupled by a simple pairwise coupling transform to yield an MSR code. Just prior to the appearance of these two papers, in an earlier version of [39], the authors show how a systematic MSR code can be converted into an MSR code by increasing the sub-packetization level by a factor of $r=(n-k)$ using a pairwise symbol transformation. This result is then extended in [39], to present a technique that takes an MDS code, increases sub-packetization level by a factor of r and converts it into a code in which the optimal repair of r nodes can be carried out. By applying this transform repeatedly $\lceil\frac{n}{r}\rceil$ times, it is shown that any scalar MDS code can be transformed into an MSR code. It turns out that the three papers [37–39], either explicitly or implicitly, employed as a key part of the construction, essentially the same pairwise-coupling transform.

Let $s=(d-k+1)$. More recently, the lower bound $\alpha\geqslant s^{\frac{n}{s}}$ was derived in [40] for optimal-access MSR codes. The same paper also shows that the sub-packetization level of an MDS code that can optimally repair any w of the n nodes must satisfy $\alpha\geqslant s^{\lceil\frac{w}{s}\rceil}$. These results established that the earlier constructions in [30, 31, 37–39, 41] were optimal in terms of sub-packetization level α. It is also shown in [40], that a vector MDS code that can repair failed nodes belonging to a fixed set of Q nodes with minimum repair bandwidth and in optimal-access fashion, and having minimum sub-packetization level $\alpha=s^{\frac{n}{s}}$ must necessarily have a coupled-layer structure, similar to that found in [37–39]. An explicit construction of MSR codes for $d<(n-1)$ with α achieving the lower bound $\alpha\geqslant s^{\frac{n}{s}}$ for $s=2,3,4$ was recently provided in [41]. Please refer to Table 1 for a summary of MSR code constructions in existing literature.

Open problems 2. Derive a tight lower bound on the sub-packetization level of MSR codes and provide matching constructions.

Open problems 3. Constructions for explicit optimal-access MSR codes for any (n,k,d) with optimal sub-packetization.

4.2 Constructions of MSR codes

Product matrix construction [14]**.** We provide a brief description of the PM construction for parameter set $(n,k,d=2k-2)$, $(\alpha=k-1,\ \beta=1,\ B=k(k-1))$. The message symbols $\{u_i\}_{i=1}^{B}$ are arranged in the form of a $(d\times\alpha)$ matrix M: $M=[S_1\ S_2]^{\mathrm{T}}$, where the S_1, S_2 are symmetric $(k-1)\times(k-1)$ matrices containing the $B=k(k-1)$ message symbols. Encoding is carried out using a $(n\times d)$ matrix

Table 1 A list of MSR constructions and the parameters. In the table $r = n-k$, $s = d-k+1$ and when all node repair is No, the constructions are systematic MSR. By 'non-explicit' field-size, we mean that the order of the size of the field from which coefficients are picked is not given explicitly

MSR code	Parameters	α	Field size	All node repair	Optimal access	Notes
Shah et al. [9]	$(n, k, d = n-1 \geqslant 2k-1)$	r	$2r$	No	Yes	IA framework
Suh et al. [20]	$(n, k, d \geqslant 2k-1)$ $(n, k \leqslant 3, d)$	s	$2r$	Yes	No	IA framework
Rashmi et al. [14]	$(n \geqslant 2k-1, k, d)$	r	n	Yes	No	Product matrix framework
Papailiopoulos et al. [22]	$(n, k, d = n-1)$	r^k	non-explicit	No	No	High rate systematic MSR
Tamo et al. [24] Wang et al. [25]	$(n, k, d = n-1)$	$r^{k+1} \leqslant 4$	when $r \leqslant 3$, else non-explicit	Yes	Yes	High rate MSR known as Zigzag codes
Cadambe et al. [26]	$(n \geqslant \frac{3k}{2}, k, d = n-1)$	$O(k^2)$	non-explicit	No	Yes	
Sasidharan et al. [30]	$(n, k, d = n-1)$	$r^{\lceil \frac{n}{r} \rceil}$	$O(n^r)$	Yes	Yes	Introduced parity-check viewpoint, optimal α
Goparaju et al. [32]	(n, k, d)	$s^{k\binom{r}{s}}$	–	No	Yes	Very large field-size needed See Sec. IV in [32] for details
Rawat et al. [31]	(n, k, d)	$s^{\lceil \frac{n}{s} \rceil}$	$O(n^r)$	Yes	Yes	Extended [30] for $d < n-1$
Ye et al. [36]	(n, k, d)	s^n	sn	Yes	No	
	(n, k, d)	s^{n-1}	$n+1$	Yes	Yes	
Ye et al. [37] Sasidharan et al. [38] Li et al. [39]	$(n, k, d = n-1)$	$r^{\lceil \frac{n}{r} \rceil}$	$r\lceil \frac{n}{r} \rceil$	Yes	Yes	Optimal α for optimal-access MSR
Vajha et al. [41]	(n, k, d) $d \in \{k+1, k+2, k+3\}$	$s^{\lceil \frac{n}{s} \rceil}$	$O(n)$	Yes	Yes	

$\Psi = [\Phi\ \Lambda\Phi]$, where Φ is an $n \times (k-1)$ matrix and Λ is a diagonal matrix. Let the i-th row of Ψ be ψ_i^{T}, the i-th row of Φ be ϕ_i^{T} and the i-th diagonal element in Λ be λ_i. The α symbols stored in node i are given by $\underline{c}_i^{\mathrm{T}} = \psi_i^{\mathrm{T}} M = \phi_i^{\mathrm{T}} S_1 + \lambda_i \phi_i^{\mathrm{T}} S_2$. The matrix Ψ is required to satisfy the properties (1) any d rows of Ψ are linearly independent, (2) any α rows of Φ are linearly independent and (3) the n diagonal elements of Λ are distinct.

• Node repair. Let f be the index of failed node, thus the aim is to reconstruct $\underline{c}_f$. The i-th helper node, h_i, $i \in [d]$, passes on the information: $\underline{c}_{h_i}^{\mathrm{T}} \phi_f = \psi_{h_i}^{\mathrm{T}} M \phi_f$. Upon aggregating the repair information we obtain the vector $[\psi_{h_1}\ \psi_{h_2}\ \cdots\ \psi_{h_d}]^{\mathrm{T}}[M\phi_f]$. As any d-rows of Ψ are linearly independent, the vector $M\phi_f$ can be recovered. From $M\phi_f$, we can obtain $S_1\phi_f$ and $S_2\phi_f$. Since S_1 and S_2 are symmetric, we can recover the contents $\underline{c}_f^{\mathrm{T}} = \phi_f^{\mathrm{T}} S_1 + \lambda_f \phi_f^{\mathrm{T}} S_2$ of the replacement node.

• Data collection. Let $\Psi_{\mathrm{DC}} = [\Phi_{\mathrm{DC}}\ \Lambda_{\mathrm{DC}}\Phi_{\mathrm{DC}}]$ be the $(k \times d)$ sub matrix of Ψ corresponding to the k nodes contacted for data collection. We wish to retrieve M from $\Psi_{\mathrm{DC}} M = \Phi_{\mathrm{DC}} S_1 + \Lambda_{\mathrm{DC}}\Phi_{\mathrm{DC}} S_2$. This can be done in three steps.

(1) First compute $\Psi_{\mathrm{DC}} M \Phi_{\mathrm{DC}}^{\mathrm{T}} = \Phi_{\mathrm{DC}} S_1 \Phi_{\mathrm{DC}}^{\mathrm{T}} + \Lambda_{\mathrm{DC}}\Phi_{\mathrm{DC}} S_2 \Phi_{\mathrm{DC}}^{\mathrm{T}}$ and set $P = \Phi_{\mathrm{DC}} S_1 \Phi_{\mathrm{DC}}^{\mathrm{T}}$, $Q = \Phi_{\mathrm{DC}} S_2 \Phi_{\mathrm{DC}}^{\mathrm{T}}$.

(2) It is clear that P, Q are symmetric. Thus we know both $P_{ij} + \lambda_i Q_{ij}$ and $P_{ij} + \lambda_j Q_{ij}$. Since $\lambda_i \neq \lambda_j$ for $i \neq j$, we can recover P_{ij} and Q_{ij} for all $i \neq j$.

(3) Since we know P_{ij} for $j \neq i$, we can compute the vector $\phi_i^{\mathrm{T}} S_1 [\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_k]$. Since any α rows of Φ are linearly independent, we can recover $\{\phi_i^{\mathrm{T}} S_1 | 1 \leqslant i \leqslant k\}$. For any set of α distinct elements ϕ_i^{T}, we can compute $[\phi_1\ \cdots\ \phi_\alpha]^{\mathrm{T}} S_1$, from which S_1 can be recovered. S_2 can be similarly recovered from Q. The present description assumes data collection from the first k nodes, while a similar argument holds true for any arbitrary set of k nodes.

Coupled layer code. We present here the constructions in [37–39] from a coupled-layer perspective. We explain the construction here only for parameter sets of the form

$$(n = st,\ k = s(t-1),\ d = n-1),\ (\alpha = s^t,\ \beta = s^{t-1}),\ q \geqslant n,$$

where $s \geqslant 1$, $t \geqslant 2$. (The construction can however, be extended to yield MSR codes for any $(n, k, d = n-1)$ using a technique called shortening). The coupled-layer code can be constructed in two steps: (a) we layer α, (n, k) MDS codewords to form an uncoupled data-cube; (b) the symbols within the uncoupled-data cube are transformed using a pairwise-forward-transform (PFT) to obtain the coupled

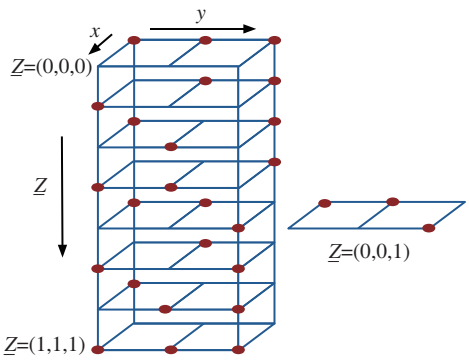

Figure 6 (Color online) Uncoupled data cube for $s = 2$, $t = 3$. The red dots represent plane-index $\underline{z}$.

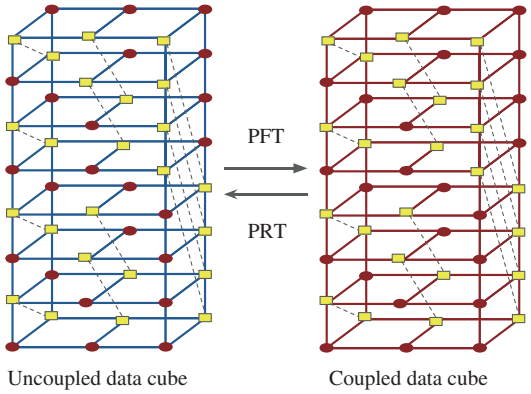

Figure 7 (Color online) Paired symbols are shown using yellow rectangles connected by dotted lines. Uncoupled symbols are transformed using PFT to get the coupled symbols in the coupled data cube.

layer code. While we discuss only the case when the MDS code employed in the layers is a scalar MDS code, there is a straightforward extension that permits the use of vector MDS codes [38].

Let us first consider the $n\alpha$ symbols $\{U(x, y, \underline{z}) \mid (x, y) \in \mathbb{Z}_s \times \mathbb{Z}_t, \underline{z} \in \mathbb{Z}_s^t\}$ of an uncoupled code $\mathcal{U}$ where each code symbol $U(x, y, \cdot)$ is a vector of α symbols in $\mathbb{F}_q$. These $n\alpha$ symbols can be organized to form a three-dimensional (3D) data cube (Figure 6), where $(x, y) \in \mathbb{Z}_s \times \mathbb{Z}_t$ is the node index and where $\underline{z} \in \mathbb{Z}_s^t$ serves to index the contents of a node. For fixed $\underline{z} \in \mathbb{Z}_s^t$, we think of the symbols $\{U(x, y, \underline{z}) \mid (x, y) \in \mathbb{Z}_s \times \mathbb{Z}_t\}$ as forming a plane or a layer and thus the value of $\underline{z}$ may be regarded as identifying a plane or layer. The symbols in each layer of the uncoupled data cube form an (n, k) MDS code.

Let Θ be the $((n - k) \times n)$ parity check (p-c) matrix of an arbitrarily chosen (n, k) scalar MDS code defined over $\mathbb{F}_q$. Let $\theta_{x,y}(\ell)$ denote the element of Θ lying in the ℓ-th row, and (x, y)-th column. Then the symbols of the uncoupled code satisfy the p-c equations

$$\sum_{(x,y)\in\mathbb{Z}_s\times\mathbb{Z}_t} \theta_{x,y}(\ell)U(x, y; \underline{z}) = 0, \quad \forall \ell \in [0, n - k - 1], \ \forall \underline{z} \in \mathbb{Z}_s^t. \tag{3}$$

Next, consider an identical data-cube (Figure 7) containing the $n\alpha$ symbols $\{C(x, y, \underline{z}) \mid (x, y) \in \mathbb{Z}_s \times \mathbb{Z}_t, \underline{z} \in \mathbb{Z}_s^t\}$ corresponding to the coupled-layer code. This data-cube will be referred to as the coupled data cube. The symbols of the coupled data cube are derived from the symbols of the uncoupled data cube as follows. Let γ be an element in $\mathbb{F}_q \setminus \{0\}$, $\gamma^2 \neq 1$. Let us define $\underline{z}(y, x) = (z_0, \ldots, z_{y-1}, x, z_{y+1}, \ldots, z_{t-1})$. Each symbol $C(x, y, \underline{z})$ which is such that $z_y \neq x$ is paired with a symbol $C(z_y, y, \underline{z}(y, x))$. The values of the symbols so paired, are derived from those of their counterparts in the uncoupled data cube as per the (2×2) linear transformation given below, termed as the PFT

$$\begin{bmatrix} C(x, y, \underline{z}) \\ C(z_y, y, \underline{z}(y, x)) \end{bmatrix} = \begin{bmatrix} 1 & \gamma \\ \gamma & 1 \end{bmatrix}^{-1} \begin{bmatrix} U(x, y, \underline{z}) \\ U(z_y, y, \underline{z}(y, x)) \end{bmatrix}. \tag{4}$$

In the case of the symbols $C(x, y, \underline{z})$ when $z_y = x$, the relation between symbols in the two data cubes is even simpler and given by $C(x, y, \underline{z}) = U(x, y, \underline{z})$. The pairwise reverse transform (PRT) is simply the inverse of the PFT and is used to obtain the uncoupled symbols $U(\cdot)$ from the coupled symbols $C(\cdot)$. The p-c equations satisfied by the coupled-layer code can be derived using the p-c equations (3) satisfied by the symbols in the uncoupled data cube and the PRT

$$\sum_{(x,y)\in\mathbb{Z}_s\times\mathbb{Z}_t} \theta_{x,y}(\ell)C(x, y, \underline{z}) + \sum_{y\in\mathbb{Z}_t} \sum_{x\neq z_y} \gamma\theta_{x,y}(\ell)C(z_y, y, \underline{z}(y, x)) = 0, \quad \forall \underline{z} \in \mathbb{Z}_s^t, \ \ell \in [0, n - k - 1]. \tag{5}$$

• Node repair. Let (x_0, y_0) be the failed node. To recover the symbols $\{C(x_0, y_0, \underline{z}) \mid \underline{z} \in \mathbb{Z}_s^t\}$, each of the remaining nodes $(x, y) \neq (x_0, y_0)$ sends helper information $\{C(x, y, \underline{z}) \mid \underline{z} \in \mathbb{Z}_s^t,\ z_{y_0} = x_0\}$. Focusing on (5) for $\underline{z}$ such that $z_{y_0} = x_0$ and retaining on the left side the unknown symbols, leads to equations of the form

$$\theta_{x_0,y_0}(\ell)C(x_0, y_0, \underline{z}) + \sum_{x \neq x_0} \gamma\theta_{x,y_0}(\ell)C(x_0, y_0, \underline{z}(y_0, x)) = \kappa^*, \quad \forall \ell \in [0, n-k-1], \tag{6}$$

where κ^* is a known value. These equations can be solved for the contents of the replacement node.

• Data collection. Please refer to [38] for the proof of data collection property.

Ye-Barg codes [36]**.** In [36], the authors present two constructions, for non optimal-access MSR and optimal-access MSR codes, respectively. These are the only known MSR constructions that are explicit and yield MSR codes for any parameter set (n, k, d). The same codes are also optimal for the repair of multiple nodes. We describe here, for simplicity, the construction of (n, k, d) MSR codes having parameters (n, k, d), $(\alpha = s^n, \beta = s^{n-1})$, $q \geqslant sn$ where $s = d-k+1$, defined over finite field $\mathbb{F}_q$ for $s \geqslant 1$. Let $\{C(i, \underline{z}) \mid i \in [n], \underline{z} \in \mathbb{Z}_s^n\}$ be the collection of $n\alpha$ symbols of a codeword, where i is the node index and $\underline{z}$ is the scalar symbol index. The code is defined via the p-c equations given as follows:

$$\sum_{i \in [n]} \lambda_{i,z_i}^{\ell} C(i; \underline{z}) = 0, \quad \forall \underline{z} \in \mathbb{Z}_s^n, \quad \ell \in [0, n-k-1], \tag{7}$$

where the $\{\lambda_{i,j}, i \in [n], j \in [0, s-1]\}$ are all distinct, thereby requiring a field size $q \geqslant sn$.

• Node repair. Let f be the failed node, D be the set of d helper nodes. The helper information sent by a node $i \in D$ is given by $\{\mu_f^i(\underline{z}) = \sum_{j=0}^{s-1} C(i, \underline{z}(f, j)) \mid \underline{z} \in \mathbb{Z}_s^n, z_f = 0\}$. Next, fixing $z_i,\ \forall i \in [n] \setminus \{f\}$ and summing equations (7) over the values of z_f, we get

$$\sum_{z_f=0}^{s-1} \lambda_{f,z_f}^{\ell} C(f, \underline{z}) + \sum_{i \in [n] \setminus \{f\}} \lambda_{i,z_i}^{\ell} \mu_f^i(\underline{z}) = 0, \quad \forall \ell \in [0, n-k-1]. \tag{8}$$

It can be shown that the collection of symbols $\{\mu_f^i(\underline{z}) | i \in [n] \setminus \{f\}\}$ form an $[n-1, d]$ MDS code. Therefore, all the $\mu_f^i(\underline{z})$ can be computed from the known d values supplied by the helper nodes and the symbols $\{C(f, \underline{z}) \mid \underline{z} \in \mathbb{Z}_s^n\}$ can thus be recovered from (8).

• Data collection. For every $\underline{z} \in \mathbb{Z}_s^n$, the collection $\{C(i, \underline{z}) | i \in [n]\}$ forms an (n, k) MDS code. Therefore, any $(n-k)$ erased symbols can be recovered.

Multiple node repair. Let $1 \leqslant t \leqslant n-k$ be the number of erasures to be recovered. It was shown in [23] that the minimum repair bandwidth required to repair t erasures in an MDS code having sub-packetization level α is lower bounded by $\gamma_t \geqslant \frac{t(n-t)\alpha}{n-k}$. Given that $k \leqslant d \leqslant n-t$ is the number of helper nodes that need to be contacted during the repair of t nodes, γ_t is lower bounded by $\gamma_t \geqslant \frac{td\alpha}{d+t-k}$. The Ye-Barg code presented above achieves this bound [36]. The t node repair discussed here assumes a centralized repair setting whereas an alternate, cooperative repair approach is discussed in Subsection 6.1.

Adaptive repair. Adaptive-repair (n, k) MSR codes are MSR codes that can repair a failed node by connecting to any d nodes, for any $d \in [k, n-1]$ and can reconstruct the failed node by downloading $\frac{\alpha}{d-k+1}$ symbols each from the d helper nodes. Constructions of MSR codes with adaptive repair can be found in [32, 36, 42].

5 On the S-RB tradeoff under exact repair

We distinguish between the S-RB tradeoffs for exact and FR RG code, by referring to them as the ER and FR tradeoff respectively. The file size B under exact repair cannot exceed that in the FR case since ER may be regarded as a trivial instance of FR. However, unlike in the case of FR codes, the data collection problem in the ER setting, cannot be identified with a multicast problem simply because each replacement node for a failed node acts as a sink for a different set of data. Thus it is not clear that

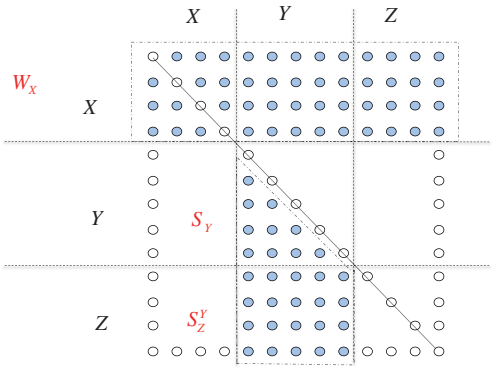

Figure 8 (Color online) The repair matrix.

the cut-set bound for FR can be achieved under ER, leaving the door open for an S-RB tradeoff in the case of ER that lies strictly above and to the right of the FR tradeoff in the (α, β)-plane. There do exist constructions of ER MBR and MSR codes meeting the cut-set bound with equality, showing that the ER tradeoff coincides with the FR tradeoff at the extreme MSR and MBR points.

5.1 The non-existence of ER codes achieving FR tradeoff

The first major result on the ER tradeoff was the result in [43], showing that apart from the MBR point and a small region adjacent to the MSR point, there do not exist ER codes whose (α, β) values lie on the interior point of the FR tradeoff. We set $\alpha_{\rm MSR} = \beta(d-k+1)$ to be the value of α at the MSR point.

Theorem 1. For any given values of $(n, k \geqslant 3, d)$, ER codes having parameters (α, β, B) corresponding to an interior point on the FR tradeoff do not exist, except possibly for α in the range

$$\alpha_{\rm MSR} \leqslant \alpha \leqslant \alpha_{\rm MSR}\left(1+\frac{1}{\alpha_{\rm MSR}(\alpha_{\rm MSR}+1)}\right) \tag{9}$$

corresponding to a small region in the neighborhood of the MSR point.

Proof. (Sketch) By restricting attention to any $(d+1)$ symbols of an RG code having parameter set $(n, k, d, (\alpha, \beta), B)$ one obtains a second RG code with parameter set $((d+1), k, d, (\alpha, \beta), B)$ in which all the remaining nodes participate in the repair of a failed node. This simplifies the analysis of the repair setting and with this in mind, in the proof, we set $n = (d+1)$. When the message vector $\underline{u}$ is picked uniformly at random, we have associated nodal random variables $\{W_i \mid i \in [n]\}$ and repair data variables $\{S_i^j \mid i \in [n] \setminus j\}$, where S_i^j denotes the data passed from node i to replacement node j. The repair matrix $\mathbb{S}$ (Figure 8) is an $(n \times n)$ matrix whose (i, j)-th entry $i \neq j$, is S_i^j. The diagonal elements of $\mathbb{S}$ do not figure in the discussion and maybe set equal to 0. Given subsets $H, N \subset [n]$, we set $W_N = \{W_i \mid i \in N\}$, $S_H^N = \{S_i^j \mid i \in H, j \in N\}$. We introduce the index sets $X = \{1, 2, \ldots, m\}$, $Y = [k] \setminus X$ and $Z = [k+1,\ d+1]$ for $m \leqslant k$. The file size B can be expressed in terms of the joint entropy of the node and repair-data variables (with logs computed to base q)

$$B = H(W_X, W_Y) = H(W_X, S_{Y\cup Z}^Y) \tag{10}$$

$$= H(W_X) + H(S_{Y\cup Z}^Y \mid W_X) \leqslant H(W_X) + \sum_{j=m+1}^{k} H\left(S_{[m+1,j-1]\cup Z}^j \mid W_X\right) \tag{11}$$

$$\leqslant m\alpha + \sum_{i=m+1}^{k}(d-i+1)\beta := B_m, \quad m = 0, 1, \ldots, k. \tag{12}$$

The cut-set bound in (1) corresponds to the inequalities: $B \leqslant \min_{m=0,1,\ldots,k} B_m$. For the bound to hold with equality, the joint random variables S_Y^Y and S_Z^Y must have maximum entropy. However it can be shown that the entropy of a row in the repair matrix is limited by β if the cut-set bound holds with equality. This leads to a contradiction, concluding the proof.

Theorem 1 does not however, rule out the possibility of an ER code having tradeoff approaching the FR tradeoff asymptotically i.e., as the file size $B \to \infty$.

5.2 The S-RB tradeoff for $(4, 3, 3)$

It is possible that the entropies of the random variables involved satisfy Shannon inequalities other than the ones we have noted and which shed light on the ER tradeoff. For the particular case $(n, k, d) = (4, 3, 3)$, Tian [44] was able to identify such an inequality with the help of a modified version of the information theory inequality prover (ITIP) [45, 46].

Let $\bar{\alpha} = \frac{\alpha}{B}$, $\bar{\beta} = \frac{\beta}{B}$ represent the normalization of α and β with respect to file size B. A point $(\bar{\alpha}, \bar{\beta})$ is said to be achievable if for any $\epsilon > 0$, there exists an ER-RG code whose $(\bar{\alpha}_1, \bar{\beta}_1)$ is ϵ-close to $(\bar{\alpha}, \bar{\beta})$. The normalized tradeoff, i.e., the tradeoff expressed in terms of $\bar{\alpha}$ and $\bar{\beta}$ allows comparison of codes across file sizes B. In the limit as $B \to \infty$, the S-RB tradeoff becomes a smooth curve. Let C_1, C_2 be RG codes over $\mathbb{F}_q$ having respective parameter sets $(n, k, d, (\alpha_1, \beta_1), B_1)$ and $(n, k, d, (\alpha_2, \beta_2), B_2)$. Consider a codeword array $\underline{c}$ obtained by vertically stacking M_1 codeword arrays of C_1 and M_2 codeword arrays of C_2. The code C comprising of all such arrays is said to be the space-shared code of C_1 and C_2. Then C is also an RG code with parameter set $(n, k, d, (M_1\alpha_1 + M_2\alpha_2, M_1\beta_1 + M_2\beta_2), M_1B_1 + M_2B_2)$. The notion of space-sharing clearly extends to multiple codes.

Theorem 2. For $(n, k, d) = (4, 3, 3)$, the achievable region $\mathcal{R}$ is given by

$$\mathcal{R} = \left\{(\bar{\alpha}, \bar{\beta}) \middle| 3\bar{\alpha} \geqslant 1,\ 2\bar{\alpha} + \bar{\beta} \geqslant 1,\ 6\bar{\beta} \geqslant 1,\ 4\bar{\alpha} + 3\bar{\beta} \geqslant 1\right\}. \tag{13}$$

Proof. Of the four inequalities listed, the first 3 follow the entropy constraints listed in (12) above. The last inequality $4\bar{\alpha} + 3\bar{\beta} \geqslant 1$ does not follow from (12), and was found in [44] using an ITIP. It remains to construct a code that operate on points on the $(\bar{\alpha}, \bar{\beta})$-plane, satisfying the inequalities with equality. A $[4, 3]$ single parity-check code serves as an MSR code C_1 for $(4, 3, 3)$. A $(4, 3, 3)$ MBR code C_2 can be constructed using the polygonal construction described in Section 3. A hand-crafted code C_3 operating at the interior point of deflection (Figure 9) is given in [44]. Every point on the lines determined by equality in (13) is achieved by a code obtained by space-sharing among C_1, C_2 and C_3.

5.3 Layered codes for interior points

A simple code-construction technique based on the layering of MDS codes turns out to provide codes that perform well with respect to file size in the interior region of the S-RB tradeoff. Let $\mathcal{C}_{\text{MDS}}$ be an MDS code having parameters $[w + \gamma, w, \gamma + 1]$. Let n be such that $w + \gamma \leqslant n$ and $L = \binom{n}{w+\gamma}$. Let $\{S_i \subset [n] \mid i = 1, 2, \ldots, L\}$ denote an ordering of the collection of all possible $(w + \gamma)$ subsets of $[n]$. Let $\underline{u}_i \in \mathbb{F}_q^w$, $i = 1, 2, \ldots, L$ be L message vectors, not necessarily distinct, and $\underline{c}_i$ be the codeword in $\mathcal{C}_{\text{MDS}}$ associated with $\underline{u}_i$. We create an $(L \times n)$ array in which we place the symbols of codeword $\underline{c}_i$ in the location specified by subset S_i. It turns out that this array represents an array code which possesses the data collection property of an RG code, but not the repair property. By replicating the array a certain number V of times, it turns out that one obtains an RG code with parameters $(n,\ k = n - \gamma,\ d = k,\ B_0 = LVw)$, operating between the MSR and MBR points. Further details can be found in [47]. We will refer to this code as the canonical layered code $\mathcal{C}_{\text{can}}$ (Figure 10). The canonical layered-code construction has been extended to construct codes with $k < d$ by making use of an outer code designed using linearized polynomials. An alternate generalization of the canonical code to the case of $k < d$ involved adding additional layers consisting of carefully designed parity symbols. Such an approach leads to the improved layered codes in [48], that turn out to be optimal for the set of parameters $(n,\ k = 3,\ d = n - 1)$.

5.4 ER tradeoff strictly away from FR tradeoff for all (n, k, d)

In [49], it was shown that the ER tradeoff cannot approach the FR tradeoff even when $B \to \infty$ for any value of (n, k, d). This was established by deriving a positive lower bound $0 < \delta < \beta$ on the gap between the ER and FR tradeoffs.

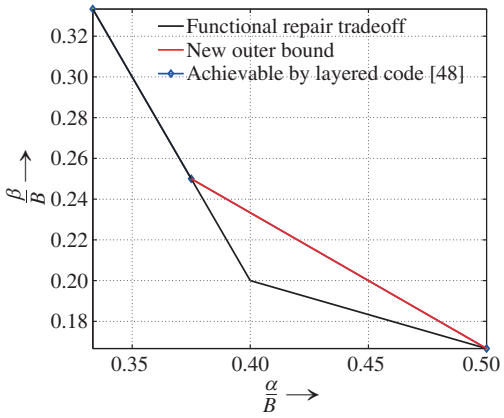

Figure 9 (Color online) The (4,3,3) normalized tradeoff.

1	2	3	4	5
c_{11}	c_{12}	c_{13}		
	c_{21}	c_{22}	c_{23}	
c_{m1}		c_{m2}	c_{m3}	
⋮	⋮	⋮	⋮	⋮

Figure 10 (Color online) An ($n = 5$, $k = 4$, $d = 4$) canonical layered code.

Theorem 3. The ER tradeoff between $\bar{\alpha}$ and $\bar{\beta}$ for any ER RG code, with $k \geqslant 3$ is strictly separated from the FR tradeoff, apart from the MSR and MBR endpoints as well as the region surrounding the MSR point appearing in (9).

The proof the theorem involves identifying contradicting bounds on the entropy of various trapezoidal-shaped subsets within the repair matrix. Subsequent papers [50, 51] derive better bounds, thereby improving the gap δ to go beyond β. In [52], the authors adopt a different approach by first providing three different expression for the entropy B of the data file involving mutual information between various repair-data variables, and taking a linear combination of these expressions that leads to a significantly tighter bound on B:

$$B \leqslant \min_{0 \leqslant p \leqslant k} \frac{(3k-2p)\alpha + \frac{p(2(d-k)+p+1)\beta}{2} + (d-k+1)\min\{\alpha, p\beta\}}{3}. \tag{14}$$

The authors in [53] improve upon the result in (14) using repair-matrix techniques, in combination with the bound in Theorem 3, leading to the best-known outer bound on the ER tradeoff. For the case of (n, $k = 3$, $d = n-1$), the outer bound is achieved by the improved layered codes, thus characterizing the ER tradeoff. The bound also characterizes certain interior points when $k = 4$ [49].

5.5 Determinant codes for interior points

The construction given in [54] has parameters $\alpha = \binom{k}{m}$, $\beta = \binom{k-1}{m-1}$ and file size $B = m\binom{k+1}{m+1}$, where $m \in \{1, 2, \ldots, k\}$ is an auxiliary parameter. The message symbols are first precoded to obtain $k\binom{k}{m}$ symbols, and these are then arranged in a data matrix M of size $(k \times \alpha)$ in a particular manner. The codeword array is then obtained as in the case of the product matrix framework introduced in [14], by setting $C_{n\times\alpha} = \psi_{n\times k} M_{k\times\alpha}$, where $\psi_{n\times k}$ is a Vandermonde matrix. The data collection and repair properties of the code are proved by making use of the Laplace expansion of determinants, and the codes for this reason, are called determinant codes. The codes achieve an outer bound discussed in the next subsection, and thus form an optimal family of codes for parameters (n, k, k). An extension of the construction to include the parameter set (n, k, $d = k+1$) can be found in [55].

5.6 ER tradeoff under linear setting

In [51, 56, 57], the authors characterize the ER tradeoff for (n, $k = n-1$, $d = n-1$) for the subclass of linear codes, using an approach that involves lower bounding the rank of the parity-check matrix of an RG code. The upper bound in [56] holds in general for any (n, k, $d = k$).

Table 2 Parameters of explicit constructions of cooperative RG codes

Type	Code parameters	Ref.
MBCR	$n,\ k,\ k \leqslant d \leqslant (n-t),\ t \geqslant 1$	[61]
MSCR	$n = d+2,\ k = t = 2$	[63]
MSCR	$n = 2k,\ d = n-2,\ k \geqslant 2,\ t = 2$	[62]
MSCR	$n = 2k,\ d = n-t,\ k \geqslant 2,\ k \geqslant t \geqslant 2$ (Repair of systematic nodes only)	[62]
MSCR	$n,\ k,\ k \leqslant d \leqslant (n-t),\ t \geqslant 1$	[64]

Theorem 4. Consider an ER linear RG code with parameters $\{(n \geqslant 4, k, d), (\alpha, \beta)\}$ and file size $B = n\alpha - \rho$. Then

$$\rho \geqslant \begin{cases} \left\lceil \dfrac{2rn\alpha - n(n-1)\beta}{r^2+r} \right\rceil, & \dfrac{d\beta}{r} \leqslant \alpha \leqslant \dfrac{d\beta}{r-1}, \quad 2 \leqslant r \leqslant n-2, \\ 2\alpha - \beta, & \dfrac{d\beta}{n-1} \leqslant \alpha \leqslant \dfrac{d\beta}{n-2}. \end{cases} \tag{15}$$

The corresponding bound on file size B coincides with the achievable region of layered codes when $k = d = (n-1)$. Determinant codes achieve the above bound in general for (n, k, k), thus characterizing the linear ER tradeoff in this case.

Open problems 4. Characterization of ER tradeoff for general (n, k, d) in both the linear and non-linear settings.

6 Variations on the theme of RG codes

6.1 Cooperative repair

This subsection was contributed at the request of the authors, by Kenneth Shum. The potential benefit of allowing data exchange among the nodes being regenerated while repairing multiple node failures simultaneously, was first investigated by Hu et al. [58]. The cooperative-repair process consists of two phases. In the first phase, each of the new nodes selects a set of d surviving nodes, and downloads a total of $d\beta_1$ symbols from them. In the second phase, a new node downloads β_2 symbols from each of the other new nodes. If t new nodes are re-built at the same time, the repair bandwidth per new node is $d\beta_1 + (t-1)\beta_2$. As in the non-cooperative case, there is a tradeoff between the amount of data stored in a node and the repair bandwidth. In the following, we denote the repair bandwidth per new node by γ. The minimum-storage cooperative regenerating (MSCR) point and minimum-bandwidth cooperative regenerating (MBCR) point are determined in [59, 60], and are given by

$$(\alpha_{\mathrm{MSCR}}, \gamma_{\mathrm{MSCR}}) = \left(\frac{B}{k}, \frac{B(d+t-1)}{k(d+t-k)}\right), \quad (\alpha_{\mathrm{MBCR}}, \gamma_{\mathrm{MBCR}}) = \frac{B(2d+t-1)}{k(2d+t-k)}\,(1,1),$$

where t is the number of nodes to be repaired simultaneously. When $t = 1$, they reduce to the corresponding operative points for single-node repair. The full FR tradeoff curve between storage and repair bandwidth per node is derived in [60]. In the case of exact repair, the explicit construction of cooperative RG codes for all parameters at the minimum-bandwidth point was first presented in [61]. The construction in [61] is presented in an alternate way in [62]. Constructions for minimum-storage cooperative codes are relatively rare (e.g., [62, 63]). Table 2 summarizes the existing constructions of MSCR and MBCR codes. We note that the MSCR codes in [62] share the same encoding method as in [9, 20]. It is shown in [62] that with the MSR codes in [9,20], we can repair multiple systematic nodes with repair bandwidth achieving the MSCR point. In [64], the authors present constructions for any $(n,\ k,\ k \leqslant d \leqslant n-t,\ t)$ MSCR codes.

The cooperative repair model was extended to partial cooperative repair in [65]. The first phase of repair is the same as described above. Each of the t new nodes contacts d other nodes and download a

total of β_1 data packets. In the second phase, a new node exchanges β_2 data packets with $t-s$ other new nodes, where s is a system parameter between 1 and t. When $s=t$, it is the original single-loss repair model. When $s=1$, it reduces to the cooperative repair model. The minimum-storage and minimum bandwidth point are derived in [65]. With partial collaboration, the minimum-storage and minimum-bandwidth operating points are given respectively by

$$(\alpha,\gamma)=\left(\frac{B}{k},\frac{B(d+t-s)}{k(d-k+t-s+1)}\right) \text{ and } (\alpha,\gamma)=\frac{B(2d+t-s)}{k(2d-k+t-s+1)}\,(1,1).$$

Two explicit codes for partial collaborative repair are presented in [66]. The code construction in [62] for MBCR codes can be extended to achieve all minimum-bandwidth points with partial collaboration. The security of cooperative RG codes is investigated in [67, 68].

6.2 MDS codes with repair capability

We discuss in this subsection, vector MDS codes that are not MSR, which nevertheless offer some savings in repair bandwidth in comparison to the conventional repair of RS codes while keeping the sub-packetization level α small. The piggybacking framework introduced in [69], was one of the first such efforts. In [70], the authors introduce codes that offer a choice of sub-packetization levels, namely, $\alpha=r^p$ for $1\leqslant p<\lceil\frac{n}{r}\rceil$. The corresponding repair download from each helper node is given by $\beta=(1+\frac{1}{p})r^{p-1}$. When $p=\lceil\frac{n}{r}\rceil$ these codes coincide with the construction in [30]. A similar approach was followed by the authors of [71] where they provide constructions for MDS codes for any given $1\leqslant\alpha\leqslant r^{\lceil\frac{k}{r}\rceil}$. However, the constructions here are restricted to systematic node repair and the bandwidth needed from each helper node is not uniform. These constructions are motivated by the systematic MSR code with $\alpha=r^{\lceil\frac{k}{r}\rceil}$ appearing in [33]. In more recent work [72], the ϵ-MSR framework was introduced to construct MDS codes that somewhat surprisingly, have sub-packetization α that is logarithmic in n for a modest increase in repair bandwidth by a multiplicative factor $(1+\epsilon)$.

Piggybacking framework. The piggybacking framework [69] begins with a collection of α codewords drawn form an MDS code and proceeds to modify the code symbols as described below. Let $\mathcal{C}$ be an MDS code and let $(f_1(u),f_2(u),\ldots,f_n(u))$ represent the codeword corresponding to message u. Next, consider codewords of $\mathcal{C}$ corresponding to α distinct messages, $u_1,\ldots,u_\alpha$. The α code symbols $f_j(u_i)$, $i=1,2,\ldots,\alpha$ are stored on node j. We first modify the code by adding a function $g_{ij}(u_1,\ldots,u_{i-1})$ to the j-th symbol of i-th codeword $f_j(u_i)$, for all $i\in\{2,\ldots,\alpha\}$, $j\in\{1,\ldots,n\}$. The values so added are termed as piggybacks. This modification does not affect our ability to decode the code, if the codewords are decoded in sequence. Applying an invertible linear transform T_i to the α code symbols in the i-th node, similarly does not affect our ability to decode the α codewords, nor a node's ability to serve as a helper node. By carefully choosing the piggybacking functions and the set T_i of invertible linear transformations it possible to reduce the repair bandwidth for the collective repair of the α MDS codewords in comparison with the repair bandwidth needed for the conventional repair of α MDS codewords. Three families of piggybacking-based MDS codes with reduced repair bandwidth and disk read are constructed in [69]. The piggybacking framework typically provides savings between 25% to 50% depending up on the parameters and choice of piggybacking functions. For example, Figure 11 shows modification of a $[4,2]$ MDS code with sub-packetization level 2 in such a way that the systematic nodes can be repaired by reading 3 symbols (instead of the 4 symbols required for MDS decoding), resulting in a 25% repair bandwidth and disk read saving.

ϵ-MSR framework. The motivation for constructing ϵ-MSR codes [72] is the larger sub-packetization level of an MSR code, which could possibly prove to be a hurdle in its practical implementation. The authors of [72] provide a generic way to transform an MSR code into an ϵ-MSR code.

Definition 2. An MDS code $\mathcal{C}$ with sub-packetization α over a finite field $\mathbb{B}$ is said to be an $(n,\ k,\ d=n-1,\ \alpha)_{\mathbb{B}}$ $\epsilon-$MSR code, $\epsilon>0$, if for every $i\in[n]$ there exists a linear repair scheme for the code symbol c_i which downloads $\beta_{ij}\leqslant(1+\epsilon)\frac{\ell}{n-k}$ symbols over $\mathbb{B}$ from the $(n-1)$ nodes storing code symbols c_j, for $j\in[n]\setminus\{i\}$.

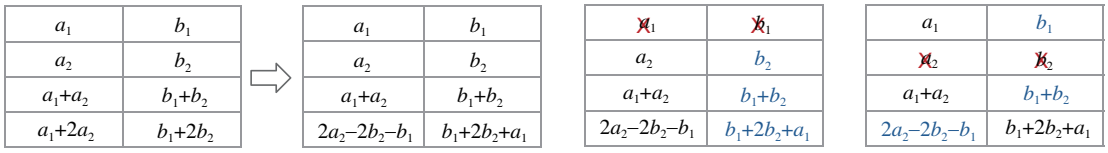

Figure 11 (Color online) Here two codewords of a [4,2] MDS code are piggybacked. The first systematic node can be repaired by reading b_2, b_1+b_2 and $b_1+2b_2+a_1$, whereas the second systematic node repair requires b_1, b_1+b_2 and $2a_2-2b_2-b_1$.

The construction of an ϵ-MSR code presented in [72] combines a short block-length MSR code with a code having large minimum distance. Let $\mathcal{C}_{\rm I}$ be an $(n=k+r,\, k,\, d=n-1,\, \alpha)_{\mathbb{B}}$ MSR code having parity check matrix

$$H=\begin{bmatrix} H_{1,1} & H_{1,2} & \dots & H_{1,n} \\ \vdots & \vdots & & \vdots \\ H_{r,1} & H_{r,2} & \dots & H_{r,n} \end{bmatrix},$$

where the sub-matrices $H_{i,j}$ are of size $(\alpha\times\alpha)$. Next, let $\mathcal{C}_{\rm II}$ be a (not necessarily linear) code having block length N, size M and minimum distance $D=\delta N$ over an alphabet $\mathbb{G}$ of size $|\mathbb{G}|\leqslant n$. Let us associate with every codeword $c=(c_1,\dots,c_N)$ of $\mathcal{C}_{\rm II}$, an $(rN\alpha\times N\alpha)$ matrix

$$\mathcal{H}_c=\begin{bmatrix} u_{1,c}\ \mathrm{Diag}(H_{1,c_1},\dots,H_{1,c_N}) \\ \vdots \\ u_{r,c}\ \mathrm{Diag}(H_{r,c_1},\dots,H_{r,c_N}) \end{bmatrix},$$

where the $\{u_{i,c}\}$ are non-zero coefficients, drawn from $\mathbb{B}$. Next, using the fact that the number of codewords in $\mathcal{C}_{\rm II}$ is M, let us form an $(rN\alpha\times MN\alpha)$ matrix $\mathcal{H}$ with each of the M 'thick' columns $\mathcal{H}_c$ corresponding to a different codeword $c\in\mathcal{C}_{\rm II}$. It can be shown that the code having $\mathcal{H}$ as its parity-check matrix is an $(M,\ M-r,\ d=M-1,\ N\alpha)_{\mathbb{B}}$ ϵ-MSR code, where $\epsilon=(r-1)(1-\delta)$. Ensuring this requires judicious selection of the base MSR code $\mathcal{C}_I$ as well as the non-zero scalars $\{u_{i,c}\}$. An additional requirement is that for a given $\epsilon>0$, the code $\mathcal{C}_{\rm II}$ should be chosen such that the parameter δ satisfies $\delta\geqslant 1-\frac{\epsilon}{r-1}$. The ϵ-MSR codes constructed using this approach can have sub-packetization level scaling logarithmically in the block length.

In [72], ϵ-MSR codes are constructed by picking the non-optimal-access MSR constructions as $\mathcal{C}_{\rm I}$. For instance, using $\mathcal{C}_{\rm I}$ with parameters $(n=3,\ k=1,\ d=2,\ \alpha=2^3=8)$ and $\mathcal{C}_{\rm II}$ with parameters $N=20$, $M=27$ and $D=13$ over $\mathbb{F}_3$ one can construct a $(M=27,\ M-r=25,\ M-1=26,\ N\alpha=160)$ ϵ-MSR code. Note that the MSR code $\mathcal{C}_{\rm I}$ with parameters $(n=27,\ k=25,\ d=26)$ requires a sub-packetization level of 2^{27}, whereas this ϵ-MSR code has sub-packetization level of 160 ($\ll 2^{27}$) and repair bandwidth is within 1.35 times that of the MSR code.

6.3 Fractional repetition codes

Fractional repetition codes [73] are regarded as codes that generalize the RBT MBR construction in [13]. A fractional repetition code is associated with the parameter set $\{n,k,\alpha,\rho\}$, where n is the number of nodes and k is the smallest number such that one can retrieve the entire data file from connecting to any set of that many nodes. Let K be the file size of the fractional repetition code. To encode and store data, a fractional repetition code begins by encoding a collection $\{u_1,\dots,u_K\}$ of message symbols drawn from a finite field $\mathbb{F}_q$ using a scalar $[N,K]$ MDS code $\mathcal{A}$, also referred to as the DRESS code in [74]. Let $(v_1,v_2,\dots,v_N)$ denote the symbols of a codeword in $\mathcal{A}$. Each of the N scalar code symbols is replicated ρ times and the resultant ρN symbols are stored across the n nodes in such a way that there are α symbols per node and each code symbol is present in precisely ρ distinct nodes. Combinatorial techniques such as t-designs are used to make such an assignment possible. For this to happen, we must have that $n\alpha=N\rho$. In order to be able to recover the entire data file by connecting to any k nodes we must clearly have that

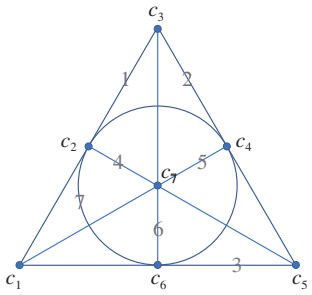

Figure 12 (Color online) Each of the seven lines in the Fano plane indicates a node and points within a line denote the code symbols stored in the corresponding node. For instance, $N_1 = \{c_1, c_2, c_3\}$.

$R_{\mathcal{C}}(k) \triangleq \min_{J\subseteq[n]:|J|=k} |\cup_{j\in J} N_j| \geqslant K$, where N_j indicates the set of α code symbols stored in j-th node, $j \in [n]$. Note that $R_{\mathcal{C}}(k)$ is defined with respect to a given collection $\{N_j\}_{j=1}^n$. Let $C_{\rm fr}(n, k, \alpha, \rho)$ denote the maximum $R_{\mathcal{C}}(k)$ possible across all possibilities of $\{N_j\}_{j=1}^n$, which conform to the parameters n, α and ρ. Hence a fractional repetition code is said to be k-optimal [75], if it satisfies $K = C_{\rm fr}(n, k, \alpha, \rho)$.

In contrast to an MBR code, a fractional repetition code requires the existence of just a single set of $d = \alpha$ helper nodes to perform RBT. However it follows naturally from the ρ-replication of code symbols that such a set of d helper nodes is available, even in the presence of $(\rho - 1)$ node failures.

Example 2 ([73])**.** Consider a fractional repetition code $\mathcal{C}$ with parameters $n = 7$, $k = 3$, $d = 3$, $\rho = 3$. The code is described using the Fano plane as shown in Figure 12. Here $R_{\mathcal{C}}(k) = 6$. By choosing the outer MDS code to be the $[7, 6]$ single parity check code, data collection property follows. As each symbol is shared by three lines, $\rho = 3$ and hence $\mathcal{C}$ permits RBT up to 2 node failures.

The following bound on the maximum rate of a fractional repetition code with parameters (n, k, α, ρ), is derived in [73].

$$C_{\rm fr}(n, k, \alpha, \rho) \leqslant \min\left\{\left\lceil \frac{n\alpha}{\rho}\left(1 - \frac{\binom{n-\rho}{k}}{\binom{n}{k}}\right)\right\rfloor,\ g(n, k, \alpha, \rho)\right\},$$

where $g(n, 1, \alpha, \rho) = 1$, and $g(n, k+1, \alpha, \rho) = g(n, k, \alpha, \rho) + \alpha - \lceil \frac{\rho g(n,k,\alpha,\rho) - k\alpha}{n-k} \rceil$.

Ref. [75] considers fractional repetition codes with parameters $\alpha \geqslant k$, $\beta = 1$ and provides several k-optimal constructions. Ref. [76] considers fractional repetition codes with parameter $\beta \geqslant 1$ and also introduces a certain notion of locally recoverable fractional repetition codes where the parameter $\alpha < k$. In [77], the authors study fractional repetition codes that have α much larger than replication degree, ρ. In [78], the authors identify necessary and sufficient conditions for the existence of fractional repetition codes.

6.4 Secure RG codes

Three secrecy models in the context of an RG code are introduced in [79]: (a) a passive eavesdropper model, where the eavesdropper can read the contents of any ℓ nodes but cannot modify the content of these nodes, (b) an active omniscient adversary model, where the adversary can read the content of $\ell = k$ nodes and can also modify the content of b nodes where $2b \leqslant k$, and (c) an active limited-knowledge adversary model, where the adversary can read the content of $\ell < k$ nodes and can modify the content of $b \leqslant \ell$ nodes. In the case of a passive eavesdropper, the secrecy capacity (B_s) is the maximum amount of information that can be stored without any information being revealed to the eavesdropper. In the active eavesdropper model, the resiliency capacity (B_r) is the maximum amount of information that can be stored such that it can be reliably made available to a legitimate data collector, in spite of the tampering on the data in b nodes done by the eavesdropper. In [79], the following upper bound on secrecy capacity of the passive eavesdropper model was derived:

$$B_s(\alpha, \gamma = d\beta) \leqslant \sum_{i=\ell+1}^{k} \min\{(d - i + 1)\beta, \alpha\}. \tag{16}$$

If α is not constrained, then the resultant bandwidth-limited secrecy capacity $B_{s,\mathrm{BL}}$ becomes a function of (k,d,β) alone. The value of $B_{s,\mathrm{BL}}$ is determined [79] for $d=(n-1)$ by providing a bound and an optimal construction. It was also shown that the resiliency capacity satisfies $B_r(\alpha,\gamma)\leqslant\sum_{i=i_0}^{k}\min\{(d-i+1)\beta,\alpha\}$, where i_0 is equal to $2b+1$ for omniscient case and $b+1$ for the limited knowledge case.

In an alternate setting, Rashmi et al. [80] assume a noisy channel for transmission of data during repair and reconstruction, and introduce the notion of an (s,t)-resilient RG code that can correct up to t errors and s errors during both repair and reconstruction. The model is aligned with the active eavesdropper model where the eavesdropper can tamper the contents of b nodes. An (s,t)-resilient RG code is shown to satisfy $B\leqslant\sum_{i=1}^{k}\min\{(d-i+1)\beta,\alpha\}$ where, $d=\Delta-2t-s$, $k=\kappa-2t-s$ and Δ, κ are the number of nodes contacted during repair and reconstruction respectively. Constructions of MSR and MBR codes that are (s,t) resilient are also provided in [80]. In [36], the authors extend this model to the repair of multiple nodes and provide MSR constructions that are resilient to t errors during repair. In [81], the authors extend the passive eavesdropper model to the setting where out of the ℓ nodes accessed, the eavesdropper can read the contents of ℓ_1 nodes and can observe the information passed on for the repair of $\ell_2=\ell-\ell_1$ nodes. The upper bound in (16) also holds for this extended case. In the case of an MBR code, since the amount of data stored equals the amount of data received for node repair, the breakup between ℓ_1, ℓ_2 is immaterial.

However in the case of an MSR code, $d\beta>\alpha$. In [81], the authors provide explicit, secure MBR, and low-rate MSR code constructions that achieve the upper bound (16) for $\ell_2=0$. The secure MSR construction from [81] provides a lower bound to the secure file size of an MSR code $B_s\geqslant(k-\ell)(\alpha-\ell_2\beta)$ for $\ell_2>0$.

The upper bound on secure MSR file size $B_s\leqslant(k-\ell)\alpha$ given by (16) is improved in [82–85]. In [86], Rawat established that the secrecy capacity of an MSR codes is given by $B_s=(k-\ell)(1-\frac{1}{n-k})^{\ell_2}\alpha$ by providing an MSR construction. An upper bound that matches with Rawat's construction is proved by Goparaju et.al [84] under the constraint of linearity. In [87], secure MSR codes with smaller field sizes for all parameters were constructed. In [88,89], the ER tradeoff is studied for secure RG codes.

7 Locally recoverable codes

The earliest-known appearance of LR codes can be found in [90,91]. A construction for a code with locality appears in [92]. A formal treatment of codes with locality with a bound on minimum distance (discussed below) appears in [93]. The extension to the non-linear case for all-symbol (AS) and information-symbol (IS) locality appear in [94,95], respectively.

Let $\mathcal{C}$ be an $[n,k]$ linear code over $\mathbb{F}_q$. For a subset $S\subseteq[n]$, we use $\mathcal{C}|_S$ to denote the restriction of $\mathcal{C}$ to the coordinates in S. Let G be a $(k\times n)$ generator matrix for $\mathcal{C}$ having columns $\{\underline{g}_i\}_{i=1}^{n}$, i.e., $G=[\underline{g}_1,\underline{g}_2,\ldots,\underline{g}_n]$. An information set $E=\{e_1,e_2,\ldots,e_k\}$ is any subset of $[n]$ of size k satisfying $\mathrm{rk}(G|_E)=\mathrm{rk}[\underline{g}_{e_1},\ldots,\underline{g}_{e_k}]=k$. An $[n,k]$ code $\mathcal{C}$ is said to have (r,δ) IS locality if there is an information set $E=\{e_1,e_2,\ldots,e_k\}$ such that for every $e_i\in E$, there exists a subset $S_i\subseteq[n]$, with $e_i\in S_i$,

$$\dim(\mathcal{C}|_{S_i})\leqslant r,\quad d_{\min}(\mathcal{C}|_{S_i})\geqslant\delta, \tag{17}$$

$\mathcal{C}$ is said to have (r,δ) AS locality if for every coordinate $i\in[n]$, there exists a subset $S_i\subseteq[n]$ with $i\in S_i$, such that (17) holds. Clearly, a code with AS locality also possesses IS locality.

7.1 Bound on minimum distance

A major result in the theory of LR codes is the minimum distance bound derived in [93], which in the context of the theorem below, was derived for $\delta=2$. An analogous proof for $\delta=2$ and nonlinear codes can be found in [94,95]. The bound in [93] was extended adopting the same approach as in [93], to the general case $\delta>2$ in [96] and appears in Theorem 5 below. The extension to codes over a vector alphabet can be found in [97].

Theorem 5 ([96]). Let $\mathcal{C}$ be an $[n,k]$ linear code over $\mathbb{F}_q$ having (r,δ) IS locality. Then

$$d_{\min} \leqslant (n-k+1) - \left(\left\lceil \frac{k}{r} \right\rceil - 1\right)(\delta - 1). \tag{18}$$

Our proof will make use of Lemma 1.

Lemma 1. Let $\mathcal{C}$ be an $[n,k]$ code and let $S \subseteq [n]$ such that $\mathrm{rk}(G|_S) \leqslant k-1$. Then $d_{\min}(\mathcal{C}) \leqslant n - |S|$.

Proof. Since $\mathrm{rk}(G|_S) \leqslant k-1$, it follows that there exists a nonzero message vector $\underline{u}$ such that $\underline{u}^{\mathrm{T}} G|_S = 0$. Let $\underline{c} = \underline{u}^{\mathrm{T}} G$, then $0 < \mathrm{wt}(c) \leqslant n - |S|$ and the result follows.

Proof. (of Theorem 5) Let $E = \{e_1, e_2, \ldots, e_k\}$ be the information set with respect to which $\mathcal{C}$ has IS locality. Let the subsets $S_i \subseteq [n]$, $1 \leqslant i \leqslant k$, be such that $e_i \in S_i$, $\mathcal{C}|_{S_i}$ is an (r,δ) code, i.e., $\dim(\mathcal{C}|_{S_i}) \leqslant r$, $d_{\min}(\mathcal{C}|_{S_i}) \geqslant \delta$. Let V_i denote the column space of $G|_{S_i}$. Next, over the course of several iterations, we incrementally build up a set S, beginning with $S = \phi$. We use j to indicate the iteration number and begin with $j = 1$. On the j-th iteration, $j \geqslant 1$, we first search for an index i such that $V_i \not\subset \mathrm{Col}(G|_S)$ ($\mathrm{Col}(A)$ refers to the column space of A). This will always be possible, as we always ensure $\mathrm{rk}(G|_S) \leqslant k-1$. Having found such an index i, we next examine the $\mathrm{rk}(G|_{S\cup S_i})$. If $\mathrm{rk}(G|_{S\cup S_i}) \leqslant k-2$, we set

$$a_j = |S \cup S_i| - |S|, \quad \gamma_j = \mathrm{rk}(G|_{S\cup S_i}) - \mathrm{rk}(G|_S), \quad S = S \cup S_i, \quad j = j+1 \tag{19}$$

in order from left to right, and repeat the procedure in $(j+1)$-th iteration by searching for an index i such that $V_i \not\subset \mathrm{Col}(G|_S)$. If at the j-th iteration, for any j, we find that

Case (i). $\mathrm{rk}(G|_{S\cup S_i}) = k-1$, we then replace the procedure in (19) with the steps $a_j = |S \cup S_i| - |S|$, $\gamma_j = \mathrm{rk}(G|_{S\cup S_i}) - \mathrm{rk}(G|_S)$, $S = S \cup S_i$, $m = j$, and terminate the program.

Case (ii). $\mathrm{rk}(G|_{S\cup S_i}) = k$. In this case, we replace the procedure in (19) by selecting a subset $T_i \subseteq S_i$ such that $\mathrm{rk}(G|_{S\cup T_i}) = k-1$ (this can always be done), and then setting $a_j = |S \cup T_i| - |S|$, $\gamma_j = \mathrm{rk}(G|_{S\cup T_i}) - \mathrm{rk}(G|_S)$, $S = S \cup T_i$, $m = j$, and then terminating the program.

Thus m indicates the number of iterations that took place before the program was terminated. Note that since for every i, $\mathrm{rk}(G|_{S_i}) \leqslant r$, we have that $\gamma_j \leqslant r$. Let $j \geqslant 1$. At the j-th iteration, let i be the index chosen such that $V_i \not\subset \mathrm{Col}(G|_S)$ and Let $R_i \subseteq S_i \backslash S$ be such that $|R_i| = \gamma_j - 1$ and $\mathrm{rk}(G|_{R_i}) = \gamma_j - 1$. Since the code having generator matrix $G|_{S_i}$ has minimum distance $\geqslant \delta$ and since $\mathrm{rk}(G|_{(S\cap S_i)\cup R_i}) \leqslant r-1$, by Lemma 1, $\delta \leqslant |S_i| - |(S \cap S_i) \cup R_i| = |S_i| - |(S \cap S_i)| - |R_i| = |S_i \setminus S| - (\gamma_j - 1)$. It follows from this that $a_j \geqslant \gamma_j + (\delta - 1)$.

• Algorithm terminates under Case (i). Since the incremental rank is at most r, it follows that the number of iterations m satisfies $m \geqslant \lceil \frac{k-1}{r} \rceil$. We thus have

$$|S| = \sum_{j=1}^{m} a_j \geqslant \sum_{j=1}^{m} (\gamma_j + \delta - 1) = (k-1) + (\delta - 1)m \geqslant (k-1) + \left\lceil \frac{k-1}{r} \right\rceil (\delta - 1).$$

• Algorithm terminates under Case (ii). Arguing similarly, we have that $m \geqslant \lceil \frac{k}{r} \rceil$ and

$$|S| = \sum_{j=1}^{m} a_i \geqslant \sum_{j=1}^{m-1} (\gamma_j + \delta - 1) + \gamma_m = (k-1) + (\delta - 1)(m-1) \geqslant (k-1) + \left(\left\lceil \frac{k}{r} \right\rceil - 1\right)(\delta - 1).$$

Case (ii) leads to a smaller lower bound on $|S|$. Hence from Lemma 1 it follows that

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

We note the following:

(1) Setting $\delta = 1$ (i.e., no locality constraint) in (18), one recovers the classical Singleton bound. For this reason, the bound in (18) is commonly referred to in the context of locality as the Singleton bound.

(2) The pyramid-code construction in Subsection 7.2.1 provides a general construction of codes with IS locality that achieves the Singleton bound for all parameters (n,k,r,δ).

(3) For many parameter sets, one can construct codes with AS locality that achieve the bound in (18), include all cases where $(r+1)|n$, see Subsection 7.2.2 below.

(4) For $\delta = 2$, bounds for AS locality that are tighter than the Singleton bound for IS locality appearing in (18), can be found in [98–101]. Constructions for codes achieving the tightened bound in [99] for the case of $n_1 > n_2$ where $n_1 = \lceil \frac{n}{r+1} \rceil$, $n_2 = n_1(r+1) - n$ and having exponential field size can also be found there.

(5) It is shown in [102] that one can construct codes with AS locality and field size of order n whose minimum distance is within 1 of the bound in (18) provided $r \nmid k$, $n \neq 1 \pmod{r+1}$. In [103], it is shown that this can be achieved for any parameter set if one permits the field size to be exponential in n.

7.2 Constructions

7.2.1 *Pyramid code construction*

The pyramid code construction technique which appeared in [91], allows us to construct for any given parameter set $\{n, k, r, \delta\}$ a code with (r, δ) IS locality achieving the $d_{\min}$ bound in (18). We sketch the construction for the case $k = 2r$. The general case $k = ar$, $a > 2$ or even when $r \nmid k$, follows along similar lines. The construction begins with the systematic generator matrix G_{MDS} of an $[n_1, k]$ scalar MDS code $\mathcal{C}_{\text{MDS}}$ having block length $n_1 = n - (\delta - 1)$. It then reorganizes the sub-matrices of G_{MDS} to create the generator matrix G_{PYR} of the pyramid code:

$$G_{\text{MDS}} = \begin{bmatrix} I_r & & P_1 & Q_1 \\ & I_r & \underbrace{P_2}_{(r\times(\delta-1))} & \underbrace{Q_2}_{(r\times s)} \end{bmatrix} \Rightarrow G_{\text{PYR}} = \begin{bmatrix} I_r & P_1 & & Q_1 \\ & & I_r & P_2 \; Q_2 \end{bmatrix},$$

where $s = n_1 - 2r - (\delta - 1)$. It is not hard to show that the $[n, k]$ code $\mathcal{C}_{\text{PYR}}$ generated by G_{PYR} has (r, δ) IS locality and that $d_{\min}(\mathcal{C}_{\text{PYR}}) \geqslant d_{\min}(\mathcal{C}_{\text{MDS}})$. It follows that $d_{\min}(\mathcal{C}_{\text{PYR}}) \geqslant d_{\min}(\mathcal{C}_{\text{MDS}}) = n_1 - k + 1 = (n - k + 1) - (\delta - 1)$, and the code $\mathcal{C}_{\text{PYR}}$ is thus optimal with respect to the $d_{\min}$ bound in (18).

7.2.2 *The Tamo-Barg construction*

The construction below by Tamo and Barg [102], provides a construction for LR codes with AS locality. While for simplicity, we present the construction for the case $\delta = 2$, the construction has a natural extension to the general case $\delta > 2$ [102]. We will refer to the construction in the sequel as the Tamo-Barg (T-B) construction.

Theorem 6. Let $\mathbb{F}_q$ be a finite field of size q, let $r \geqslant 2$, $n = m(r+1) \leqslant q$, with $m \geqslant 2$ and $2 \leqslant k \leqslant (n-1)$. Set $k = ar + b$, $0 \leqslant b \leqslant (r-1)$. Let $A = \{\theta_1, \theta_2, \ldots, \theta_n\} \subseteq \mathbb{F}_q$ and $A_i \subset A$, $1 \leqslant i \leqslant m$, $|A_i| = (r+1)$, $A_i \cap A_j = \phi$, $i \neq j$ represent a partitioning $A = \cup_{i=1}^{m} A_i$ of A. Let $g(x)$ be a 'good' polynomial, by which is meant, a polynomial over $\mathbb{F}_q$ that is constant on each A_i and of degree $(r+1)$. Let

$$f(x) = \sum_{j=0}^{a-1} \sum_{i=0}^{r-1} a_{ij} [g(x)]^j x^i + \sum_{j=a}^{b-1} \sum_{i=0}^{b-1} a_{ij} [g(x)]^j x^i, \tag{20}$$

where the $a_{ij} \in \mathbb{F}_q$ are the message symbols and where the second term is vacuous for $b = 0$, i.e., when $r|k$. Consider the code $\mathcal{C}$ of block length n and dimension k where the code symbols are obtained through evaluation of the above collection of polynomials at the elements in A. Then $\mathcal{C}$ is an (r, δ) AS locality code with $\delta = 2$ and is optimal with respect to the $d_{\min}$ bound in (18). The i-th local code has support set A_i.

Proof. In (20), it can be checked that by varying $\{a_{ij}\}$, one obtains a collection of k linearly independent polynomials and since $k < n$, it follows that the code has dimension k. Let $g(\theta) = \gamma_\ell$, all $\theta \in A_\ell$. Next, let $\theta \in A_\ell$. Then we have

$$f(x)|_{\theta \in A_\ell} = \sum_{j=0}^{a-1} \sum_{i=0}^{r-1} a_{ij} [\gamma_\ell]^j x^i + \sum_{j=a}^{b-1} \sum_{i=0}^{b-1} a_{ij} [\gamma_\ell]^j x^i,$$

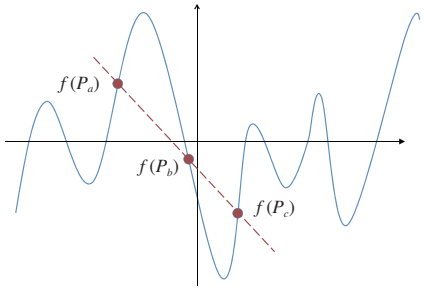

Figure 13 (Color online) In the T-B construction, code symbols in the local codes of length $(r+1)$ correspond to the evaluations of polynomials of degree $\leqslant (r-1)$. Here, $r=2$ implying evaluation at 3 points of a linear polynomial.

which is a polynomial of degree $\leqslant (r-1)$, see Figure 13, and hence the corresponding evaluation code, when restricted to A_i has $d_{\min} \geqslant 2$, leading to the desired locality and ability to recover from a single erasure. To determine $d_{\min}$, assume $b \geqslant 1$. The maximum degree of a polynomial $f(x)$ then equals

$$a(r+1)+b-1 \;=\; (ar+b)+(a-1) = k + \left\lceil \frac{k}{r} \right\rceil - 2.$$

When $b=0$ and hence $k=ar$, the maximum degree equals

$$(a-1)(r+1)+(r-1) \;=\; (ar)+(a-2) = k + \left\lceil \frac{k}{r} \right\rceil - 2.$$

It follows that the code is optimal as $d_{\min} \geqslant (n-k+1) - (\lceil \frac{k}{r} \rceil - 1)$.

An example of how good polynomials may be constructed is given below, corresponding to the annihilator polynomial of a multiplicative subgroup G of $\mathbb{F}_q^*$.

Example 3. Let $H < G \leqslant \mathbb{F}_q^*$ be a chain of cyclic subgroups, where $|H| = (r+1)$, $|G| = n$ so that $(r+1)|n|(q-1)$. Let $n=(r+1)t$. Let $\{A_i = \gamma_i H \mid i \in \{1,2,\ldots,t\}\}$ be the t multiplicative cosets of H in G, with γ_1 being the multiplicative identity so that $A_1 = H$. It follows that

$$\prod_{\beta \in A_i} (x-\beta) \;=\; x^{r+1} - \gamma_i^{r+1},$$

so that x^{r+1} is constant on all the cosets of H in G and may be selected as the good polynomial $g(x)$ i.e., $g(x) = x^{r+1}$ is one possible choice of good polynomial based on multiplicative group H.

Further examples may be found in [102,104,105]. For constructions meeting the Singleton bound with field size of $O(n)$ and more flexible value of r [106]. Construction of LR codes meeting the Singleton bound with $O(n)$ field size can also be found in [107].

7.3 Alphabet-size dependent bounds on code rate

7.3.1 *General bound*

The bound in Theorem 5 as well as the bounds for non-linear and vector codes derived in [94, 97] hold regardless of the size q of the underlying finite field. The theorem below takes the size q of the code symbol alphabet into account and provides a tighter upper bound on the dimension of a code with locality that is valid even for nonlinear codes. The 'dimension' of a nonlinear code $\mathcal{C}$ over an alphabet $\mathbb{Q}$ of size $q = |\mathbb{Q}|$ is defined to be the quantity $k = \log_q(|\mathcal{C}|)$.

Theorem 7 ([108]). For any (n,k,d) code $\mathcal{C}$ that is an LR code with parameter r over an alphabet $\mathbb{Q}$ of size $q = |\mathbb{Q}|$,

$$k \;\leqslant\; \min_{t \in \mathbb{Z}_+} \Big[tr + k_{\text{opt}}^{(q)}(n - t(r+1), d) \Big], \tag{21}$$

where $k_{\text{opt}}^{(q)}(n-t(r+1), d)$ is the largest possible dimension of a code over $\mathbb{Q}$ having block length $(n-t(r+1))$ and minimum distance d.

Table 3 A comparison of upper bounds on the dimension k of binary LR code, for given $(n, d_{\min}, r, q)$

$n = 31$, $q = 2$, $d_{\min} = 5$					
r (locality)	2	3	4	5	6
Bound (21)	17	19	20	20	20
Bound in [112]	15	18	20	22	23
Bound (22)	16	18	19	20	20

Proof. (Sketch of proof) The bound holds for linear as well as nonlinear codes. In the linear case, with $\mathbb{Q} = \mathbb{F}_q$, the derivation proceeds as follows. Let G be a $(k \times n)$ generator matrix of the LR code $\mathcal{C}$. Then it can be shown that for any integer $t > 0$, there exists an index set $\mathcal{I}$ such that $|\mathcal{I}| = \min(t(r+1), n)$ and $\text{rank}(G|_{\mathcal{I}}) = s \leqslant tr$. This implies that $\mathcal{C}$ has a generator matrix of the form (after permutation of columns):

$$G = \begin{bmatrix} \underbrace{A}_{(s\times|\mathcal{I}|)} & B \\ [0] & D \end{bmatrix}.$$

In turn, this implies that the rowspace of D defines an $[n - t(r+1), k - s \geqslant k - tr, d]$ code over $\mathbb{F}_q$, if $k - tr > 0$. It follows that $k \leqslant tr + k^{(q)}_{\text{opt}}(n - t(r+1), d)$ and the result follows. Note that the row space of D corresponds to a shortening $\mathcal{C}^S$ of $\mathcal{C}$ with respect to the coordinates $\mathcal{I} \subseteq [n]$. The proof in the general case is a (nontrivial) extension to the nonlinear setting.

Remark 6. The above bound was obtained by showing that shortening of an $[n, k, d]$ LR code with parameter r, leads to an $[n - t(r+1), \geqslant k - tr, d]$ code. Classical bounds on coding theory can be applied to this shortened code, to yield "lifted" bounds on the parent code having locality. This shortening approach, presented for the first time in [108], has since been employed in subsequent papers in [109,110].

An alphabet-size-dependent bound on $d_{\min}$ (based on the shortening approach in [108]), and which uses upper bounds on generalized Hamming weights (GHW) [111] of the dual code derived in [98], appears in [110]. The approach in [110] can also be used to derive the following upper bound on dimension which is in general tighter than (21)

$$k \leqslant \min_{\{i: e_i < n-d+1\}} \Big[e_i - i + k^{(q)}_{\text{opt}}(n - e_i, d) \Big]. \tag{22}$$

The integers $\{e_i\}_i$ appearing here can be recursively computed for a given (n, r), and represent upper bounds on the GHW of the dual code (Subsection 8.2.3). A bound on the dimension of a binary LR code for a given $(n, r, d_{\min})$ based on the Hamming bound for $d_{\min} \geqslant 5$ and $2 \leqslant r \leqslant \frac{n}{2} - 2$ appears in [112]. This bound is shown to be tighter than (21) for some cases including $5 \leqslant d_{\min} \leqslant 8$ for n large.

In [109], the authors employ the shortening approach to derive an alphabet-size-dependent bound on the minimum distance and dimension of codes having IS locality. An example comparison of the bounds on dimension for linear LR codes in (21), (22) and the Hamming-bound based bound in [112] is presented in Table 3.

7.3.2 *Bounds with disjoint repair groups*

Bounds on the dimension of a binary LR code $\mathcal{C}$ for a given n, r, $d_{\min}$ under the assumption that the local codes $(\mathcal{C}|_{S_i})$ have pairwise disjoint support appear in [112–114]. The bound in [112] make use of the Hamming bound and is shown to be tighter than (21) for some cases. A tightening of this bound appears in [113]. The tightest known bounds for this setting appear in [114] and are based on linear programming.

7.3.3 *Bounds on the dimension of cyclic LR code*

A linear-programming-based upper bound on the dimension of cyclic LR codes appears in [115]. Other bounds can be found in [116,117].

7.3.4 *Asymptotic bounds*

Upper bounds on asymptotic rate $R^q(r,1,\Delta)$ (see Subsection 8.2.5 for a definition) for a given fractional minimum distance of a binary LR code appear in [114], that represent a slight tightening of the asymptotic version of the bound in (21). An achievable asymptotic Gilbert-Varshamov type lower bound for LR code appear in [118] to be

$$R^q(r,1,\Delta) \geqslant 1 - \min_{0<s\leqslant 1}\left(\frac{1}{r+1}\log_q((1+(q-1)s)^{r+1}+(q-1)(1-s)^{r+1}) - \Delta\log_q(s)\right). \tag{23}$$

Constructions achieving the lower bound (23) can also be found in [108]. An improved lower bound obtained via a construction that makes use of algebraic-geometric codes based on the Garcia-Stichtenoth curves appear in [119]

$$R^q(r,1,\Delta) \geqslant \frac{r}{r+1}\left(1-\Delta-\frac{\sqrt{q}+r}{q-1}\right) \quad \text{for} \quad (r+1)|(\sqrt{q}+1).$$

Constructions based on algebraic geometry and covering a wider range of parameters can be found in [120]. The algebraic-geometry-based constructions improve upon the GV-type bound in (23) for some selected range of parameters.

7.4 Small-alphabet constructions

7.4.1 *Construction of binary codes*

Constructions for binary codes that achieve the bound on dimension given in (21) for binary codes, appear in [121–123]. While [121, 123] provide constructions for $d_{\min}=4$ and $d_{\min}=6$ respectively, the constructions in [122] handle the case of larger minimum distance but have locality parameter restricted to $r\in\{2,3\}$. In [109], the authors give optimal binary constructions with information and all symbol locality with $d_{\min}\in\{3,4\}$. The construction is optimal with respect to a bound similar to (21) derived in [109]. Constructions achieving the bound on dimension appearing in [112] and the further tightened bound for disjoint repair groups given in [113] for binary codes, appear respectively, in [112, 113]. These constructions are for the case $d_{\min}=6$. In [123], the authors present a characterization of binary LR codes that achieve the Singleton bound (18). In [124], the authors present constructions of binary codes meeting the Singleton bound. These codes are a subclass of the codes characterized in [123] for the case $d_{\min}\leqslant 4$.

7.4.2 *Constructions with small, non-binary alphabet*

In [125], the authors characterize ternary LR codes achieving the Singleton bound (18). In [123,124,126], the authors provide constructions for codes over a field of size $O(r)$ that achieve the Singleton bound in (18) for $d_{\min}\leqslant 5$. Some codes from algebraic geometry achieving the Singleton bound (18) for restricted parameter sets are presented in [127].

7.4.3 *Construction of cyclic LR codes*

Cyclic LR codes can be constructed by carefully selecting the generator polynomial $g(x)$ of the cyclic code. We illustrate a key idea behind the construction of a cyclic LR code by means of an example.

Example 4. Let α be a primitive element of $\mathbb{F}_{16}$ satisfying $x^4+x+1=0$. Let $\mathcal{C}_1$ be a cyclic $[n=15,k=10]$ code having generator polynomial $g_1(x)=(x+1)(x^4+x+1)$. Since the consecutive powers $\{1,\alpha,\alpha^2\}$ of α are zeros of $g_1(x)$, it follows that $d_{\min}(\mathcal{C})\geqslant 3+1=4$ by the BCH bound. Suppose we desire to ensure that a code $\mathcal{C}$ having generator polynomial $g(x)$ has $d_{\min}\geqslant 4$ and in addition, is locally recoverable with parameter $(r+1)=5$, then we do the following. Set $s=\frac{n}{(r+1)}=3$. Let $g_2(x)=\prod_{l=0}^{s-1=2}(x-\alpha^{5l})$ and $g(x)=\text{lcm}\{g_1(x),g_2(x)\}=g_1(x)g_2(x)/(x+1)$. It follows that

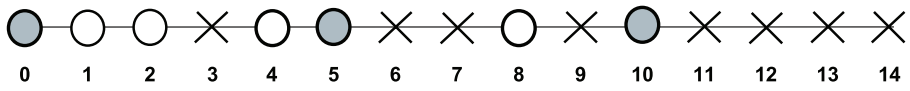

Figure 14 Zeros of the generator polynomial $g(x) = \frac{g_1(x)g_2(x)}{(x+1)}$ of the cyclic code in Example 4 are identified by circles. The unshaded circles along with the shaded circle corresponding to $\alpha^0 = 1$ indicate the zeros $\{1, \alpha, \alpha^2, \alpha^4, \alpha^8\}$ of $g_1(x)$ selected to impart the code with $d_{\min} \geqslant 4$. The shaded circles indicate the periodic train of zeros $\{1, \alpha^5, \alpha^{10}\}$ introduced to cause the code to be locally recoverable with parameter $(r+1) = 5$. The common element 1 is helpful both to impart increased minimum distance as well as locality.

$\sum_{t=0}^{14} c_t \alpha^{5lt} = 0$, $l = 0, 1, 2$. Summing over l we obtain

$$\sum_{l=0}^{2} \sum_{t=0}^{14} c_t \alpha^{5lt} = 0 \Rightarrow \sum_{t:t=0 \pmod 3} c_t = 0.$$

It follows that the symbols $\{c_t | t = 0 \pmod 3\}$ of $\mathcal{C}$ form a local code as they satisfy the constraint of an overall parity-check. Since the code $\mathcal{C}$ is cyclic the same holds for the code symbols $\{c_{t+\tau} | t = 0 \pmod 3\}$, for $\tau = 0, 1, 2$. Thus through this selection of generator polynomial $g(x)$, we have obtained a code that has both locality and $d_{\min} \geqslant 4$. The zeros of $g(x)$ are illustrated in Figure 14. The code $\mathcal{C}$ has parameters $[n = 15, k = 8, d_{\min} \geqslant 4]$ and $r = 4$. Note that the price we pay for introduction of locality is a loss in code dimension, equal to the degree of the polynomial $\frac{g_2(x)}{\gcd\{g_1(x), g_2(x)\}}$. Thus an efficient code will choose the zeros of $g_1(x)$, $g_2(x)$ for maximum overlap.

The above idea of constructing cyclic LR code was introduced in [116] and extended in [115, 117, 128, 129]. In [130], the use of locality for reducing the complexity of decoding a cyclic code is explored. The same paper also makes a connection with earlier work [131] that can be interpreted in terms of locality of a cyclic code. In [116], a construction of binary cyclic LR codes for $r = 2$ an $d_{\min} \in \{2, 6, 10\}$ achieving a bound derived within the same paper for binary codes is provided. In [128], the authors give constructions of optimal binary, ternary codes meeting the Singleton bound (18) for $d_{\min} = 4$, $r \in \{1, 3\}$ and $d_{\min} = 6$, $r = 2$ as well as a construction of a binary code meeting the bound given in [112] for $d_{\min} = 6$, $r = 2$ based on concatenating cyclic codes. A discussion on the locality of classical binary cyclic codes as well as of codes derived from them through simple operations such as shortening, can be found in [109, 132]. The principal idea here is that any cyclic code has locality $d^{\perp} - 1$ where $d^{\perp}$ is the minimum distance of the dual code $\mathcal{C}^{\perp}$. In [117], the authors construct optimal cyclic codes under the constraint that the local code is either a simplex code or else, a Reed-Muller code. In [115], the authors provide a construction of cyclic codes with field size $O(n)$ achieving the Singleton bound (18) and also study the locality of subfield subcodes as well as their duals, the trace codes. In [129], constructions of cyclic LR codes with $d_{\min} \in \{3, 4\}$ for any q and flexible n are provided.

7.5 Maximal recoverable (MR) codes

An $[n, k]$ MDS code can recover from any pattern of $(n-k)$ erasures. MR codes [133] are codes that operate under some pre-specified linearity constraints and which can recover from any pattern of $(n-k)$ erasures that is not precluded by the pre-specified linearity constraints imposed. In the context of locality, these constraints are the ones imposed on the local codes. A different perspective of MR codes based on k-core subsets (defined below) is given in [93].

Definition 3. Let H_0 be an $(\rho \times n)$ matrix over $\mathbb{F}_q$ whose row space has $m = q^{\rho} - 1$ nonzero vectors with respective support sets $A_i \subseteq [n]$, $i = 1, 2, \ldots, m$. We view H_0 as the matrix that imposes locality constraints. Let us define a subset $S \subset [n]$ to be a k-core with respect to H_0 if $|S| = k$ and $|A_i \cap S^c| \geqslant 1$, for all $i = 1, 2, \ldots, m$. Then with respect to H_0, an MR code is an $[n, k, H_0, q]$ code $\mathcal{C}$ possessing a $(k \times n)$ generator matrix G with $k \leqslant n - \rho$ satisfying the property that $H_0 G^{\mathrm{T}} = [0]$ and for any k-core S,

$$\mathrm{rank}\,(G|_S) = k. \tag{24}$$

Remark 7. Let $H = [\frac{H_0}{H_1}]$ denote the parity-check matrix of the MR code, where H_1 represents the additional parity-checks that need to be imposed to satisfy the requirements of an MR code. It could happen that the elements of H_0 belong to a small base field $\mathbb{B}$ and over that field it is not possible to find a matrix H_1 which will result in an MR code. It turns out that in such instances, one can always choose the elements of H_1 to lie in a suitable extension field $\mathbb{F}_q$ of $\mathbb{B}$, resulting in an MR code over $\mathbb{F}_q$.

Remark 8. The condition in (24) imposed on the k-core subsets S is equivalent to the following condition. Let $B \subseteq [n]$ be such that $|B^c \cap A_i| \geqslant 1, \forall i = 1, 2, \ldots, m$. Then $G|_B$ is a generator matrix of an $[n = |B|, k]$ MDS code. This follows since any k columns of $G|_B$ are required to be linearly independent.

7.5.1 *General construction with exponential field size*

The following construction is based on parity check matrix. There is an equivalent construction based on generator matrix which is presented in [93]. Saying that S is a k-core is equivalent to saying that S is an information set since the k underlying message symbols can be uniquely recovered from the k code symbols $\{c_i | i \in S\}$. From the perspective of the parity check matrix H, S is a k-core if and only if $\mathrm{rk}\,(H\,|_{S^c}) = (n-k)$. This suggests a construction technique. Setting $H = [\,{}^{H_0}_{H_1}\,]$ as earlier, we regard the symbols in the $((n-k-\rho)\times n)$ matrix H_1 as variables. We need to select H_1 such that any $(n-k)\times(n-k)$ sub-matrix of H corresponding to the complement S^c of a k-core, has nonzero determinant. Let $P(H_1)$ be the polynomial in the symbols of H_1 obtained by taking the product of these determinants. Note that the definition of a k-core ensures that each of these determinants are non-zero polynomials. The product polynomial is a polynomial in the entries (variables) of the matrix H_1 and each variable appears with degree at most $\binom{n-1}{n-k-1}$. By the combinatorial nullstellensatz [34], it follows that there is a field of size $q > \binom{n-1}{n-k-1}$ such that this product of determinants can be made nonzero. Thus an MR code always exists of field size $q > \binom{n-1}{n-k-1}$. The interest is of course, in explicit constructions of MR codes having low field size q. It is also possible to use linearized polynomials to construct MR codes, but while this results in an explicit construction, the field size is still in general, of exponential size.

7.5.2 *Partial MDS codes*

In the literature, the focus motivated by practical considerations, is on the following subclass of MR codes, also sometimes termed as partial MDS (P-MDS) codes [134].

Definition 4. An (r, δ, s) MR code or P-MDS code is defined as an $[n = m(r+\delta), k = mr - s]$ code over $\mathbb{F}_q$ in which the n code symbols can be arranged as an array of $(m \times (r+\delta))$ code symbols in such a way that each row in the array forms an $[r+\delta, r, \delta+1]$ MDS code and upon puncturing any δ code symbols from each row of the array, the resulting code becomes an $[mr, mr-s]$ MDS code.

A tabular listing of some constructions of P-MDS codes [107, 134–141] appears in Table 4. In [142], the authors characterize the weight enumerators and higher support weights of an $(r, 1, s)$ MR code.

8 LR codes for multiple erasures

We begin with an overview of the different classes (Figure 15) of LR codes that are capable of recovering from multiple erasures proposed in the literature. All the codes defined in this section are over the finite field $\mathbb{F}_q$.

8.1 Various classes of multiple-erasure LR codes

• Sequential-recovery LR codes. An (n, k, r, t) sequential-recovery LR code (abbreviated as S-LR code) is an $[n, k]$ linear code $\mathcal{C}$ having the following property. Given a collection of $s \leqslant t$ erased code symbols, there is an ordering $(c_{i_1}, c_{i_2}, \ldots, c_{i_s})$ of these s erased symbols such that for each index i_j, there exists a subset $S_j \subseteq [n]$ satisfying

Table 4 Constructions for P-MDS codes

Ref.	Parameters of MR code	Field size
		General r, δ, s
[135]	(r, δ, s)	$(q')^{mr}$ where q' is a prime power $\geqslant r+\delta$.
[136]	(r, δ, s)	$\geqslant \max((q')^{\delta+s}m^{s-1}, (q')^{s(\delta+s)})$ with q' a prime power $\geqslant r+\delta$.
		$\delta = 1$
[134]	$(r, 1, s)$	$O(2^n)$
[137]	$(r, 1, s)$	$O(m^{\lceil (s-1)(1-\frac{1}{2^r}) \rceil})$ or $\geqslant n^{\frac{m+s}{2}}$ for $m+s$ even and $\geqslant 2n^{\frac{m+s-1}{2}}$ for $m+s$ odd, when $r+1$ and m are powers of 2.
[138]	$(r, 1, s)$	$\geqslant (q')^{\lfloor (1-\frac{1}{m})s \rfloor + m - 1}$ (q' is prime power $\geqslant n$) and for some special case, the field size of their construction is $\geqslant (q')^{\lfloor (1-\frac{1}{m})s \rfloor + m - 2}$. For $m = 2$, $4 \vert s$, $\geqslant (q')^{\frac{s}{2}}$ where $q' \geqslant n$ is a power of 2.
[136]	$(r, 1, s)$	$\geqslant 2^{\ell(1+(s-1)\lceil \log_{2^\ell}(m) \rceil)}$ where $\ell = \lceil \frac{s+1}{2} \rceil \lceil \log_2(r+\delta) \rceil$.
		$s = 1$
[134]	$(r, \delta, 1)$	$O(\max(m, r+\delta))$
[139]	$(r, \delta, 1)$	$O(r+\delta)$
		$s = 2$
[140]	$(r, 1, 2)$	$O(n)$
[141]	$(r, \delta, 2)$	$\geqslant m((\delta+1)(r-1)+1) \approx \delta \times n$
[107]	$(r, \delta, 2)$	$O(n)$
		$s = 3$
[137]	$(r, 1, 3)$	$O(k^{\frac{3}{2}})$
[136]	$(r, \delta, 3)$	if $m < (r+\delta)^3$ then $O((r+\delta)^{3(\delta+3)})$ otherwise $O((r+\delta)^{\delta+3}m^{1.5})$
		$s = 4$
[137]	$(r, 1, 4)$	$O(k^{\frac{7}{3}})$

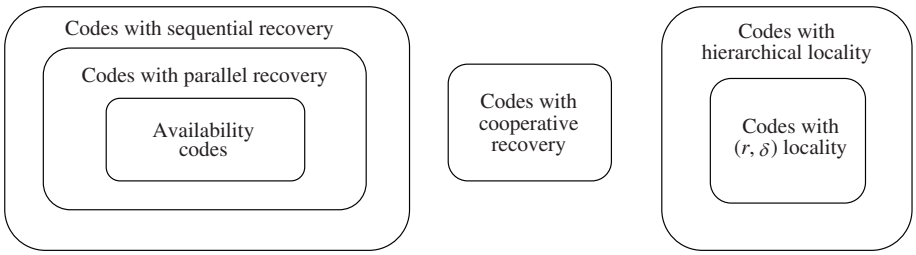

Figure 15 The various code classes corresponding to different approaches to recovery from multiple erasures.

$$\begin{aligned} &\text{(i)} \quad |S_j| \leqslant r, \\ &\text{(ii)} \quad S_j \cap \{i_j, i_{j+1}, \ldots, i_s\} = \phi, \\ &\text{(iii)} \quad c_{i_j} = \sum_{\ell \in S_j} u_\ell c_\ell, \quad u_\ell \in \mathbb{F}_q. \end{aligned} \tag{25}$$

It follows from the definition that an (n, k, r, t) S-LR code can recover from the erasure of s code symbols $c_{i_1}, c_{i_2}, \ldots, c_{i_s}$, for $1 \leqslant s \leqslant t$ by using (25) to recover the symbols c_{i_j}, $j = 1, 2, \ldots, s$, in succession.

• Parallel-recovery LR codes. If in the definition of the S-LR code, we replace the condition (ii) in (25) by the more stringent requirement $S_j \cap \{i_1, i_2, \ldots, i_s\} = \phi$, then the LR code will be referred to as a parallel recovery LR code, abbreviated as P-LR code. Clearly the class of P-LR codes is a subclass of S-LR codes. From a practical perspective, P-LR codes are preferred since as the name suggests, the erased symbols can be recovered in parallel. However, this will in general come at the expense of storage overhead. We note that under parallel recovery, depending upon the specific code, this may require the same helper (i.e., non-erased) code symbol to participate in the repair of more than one erased symbol c_{i_j}.

• Availability codes. An (n, k, r, t) availability LR code, is an LR code having the property that in the

event of a single but arbitrary erased symbol c_i, there exist t recovery sets $\{R_j^i\}_{j=1}^t$ which are pair-wise disjoint and of size $|R_j^i| \leqslant r$ with $R_j^i \subseteq [n] - \{i\}$ such that for each j, $1 \leqslant j \leqslant t$, c_i can be expressed as

$$c_i = \sum_{\ell \in R_j^i} a_{i\ell} c_\ell, \quad a_{i\ell} \in \mathbb{F}_q.$$

An (n, k, r, t) availability code is also an (n, k, r, t) P-LR code. This follows because the presence of at most t erasures implies, that there will be at least one recovery set for each erased code symbol all of whose symbols remain unerased. If the t disjoint recovery sets are available only for code symbols corresponding to an information set, the code is said to be an IS availability code as opposed to the AS availability implicit in the previous definition.

• (r, δ) codes. Recovery from t erasures can also be accomplished by using the codes with (r, δ) locality introduced in Section 7, if one ensures that the code has $d_{\min} \geqslant t + 1$. However in this case, repair is local only in those cases where the erasure pattern is such that the number of erasures e_i within each local code satisfies $e_i \leqslant \delta - 1$. Thus one may regard (r, δ) codes as offering probabilistic guarantees of local recovery in the presence of $\leqslant t$ erasures in exchange for a potential increase in code rate. Of course, one could always employ an (r, δ) locality with each local code being an MDS code and $\delta \geqslant t + 1$, but this would result in a significant rate penalty.

• Cooperative recovery codes. A cooperative recovery (n, k, r, t) LR (C-LR) code is an LR code such that if a subset $(c_{i_1}, c_{i_2}, \ldots, c_{i_s})$, $1 \leqslant s \leqslant t$ of symbols are erased, then there exists a subset $\{c_{j_1}, c_{j_2}, \ldots, c_{j_r}\}$ of r other code symbols (i.e., $i_a \neq j_b$ for any a, b) such that for all $a \in [s]$, $c_{i_a} = \sum_{b=1}^r \theta_{a,b} c_{j_b}$, $\theta_{a,b} \in \mathbb{F}_q$. Clearly an (n, k, r, t) C-LR code is also an (n, k, r, t) P-LR code, but the r in the case of a C-LR code will tend to be significantly larger. One may regard C-LR codes as codes that seek to minimize the number of unerased symbols contacted per erased symbol on average, rather than insist that each code symbol be repaired by contacting r other code symbols.

8.2 Availability codes

8.2.1 *Bounds on code rate*

The following upper bound on the rate of an availability code was given in [118].

Theorem 8 ([118]). If $\mathcal{C}$ is an (n, k, r, t) availability code, then its rate R must satisfy

$$R = \frac{k}{n} \leqslant \frac{1}{\prod_{j=1}^t (1 + \frac{1}{jr})}. \tag{26}$$

The parity check matrix of an availability code can be written in the form $H^{\mathrm{T}} = [H_a^{\mathrm{T}}\ H_b^{\mathrm{T}}]$ where the rows of H_a are the distinct parity checks associated with the recovery sets R_j^i, $\forall i \in [n]$, $j \in [t]$ and where the matrix H_b contains all the remaining parity checks. Clearly the Hamming weight of each row of H_a is $\leqslant (r + 1)$ and the column weight $\geqslant t$.

• Codes with strict availability. Codes with strict availability (SA-LR codes) are simply the subclass of availability codes where each row of H_a has weight equal to $(r + 1)$ and each column of H_a has weight equal to t. Thus the number m of rows of H_a must satisfy $m(r + 1) = nt$. Further, if the support sets of the rows in H_a having a non-zero entry in the i-th column are given respectively by $S_j^{(i)}$, $j = 1, 2, \ldots, t$, then we must have by the disjointness of the recovery sets, that $S_j^{(i)} \cap S_l^{(i)} = \{i\}$, $\forall\ 1 \leqslant j \neq l \leqslant t$. Each code symbol c_i in an SA-LR code is thus protected by a collection of t 'orthogonal' parity checks, each of weight $(r + 1)$.

Theorem 9 ([110]). Let $R = \frac{k}{n}$ be the maximum possible rate of an (n, k, r, t) SA-LR code. Then R must satisfy the upper bound

$$R \leqslant 1 - \left(\frac{t}{r+1}\right) + \left(\frac{t}{r+1}\right)\left(\frac{1}{\prod_{j=1}^{r+1}(1 + \frac{1}{j(t-1)})}\right). \tag{27}$$

The above bound (27), derived in [110], is tighter than (26) as r increases for any fixed t. An upper bound on rate of an $(n, k, r = 2, t)$ SA-LR code over $\mathbb{F}_2$ that for large t, becomes tighter in comparison with the bounds in either (26) or (27), is presented in [143]. Also contained in [143], is an upper bound on the rate of an $(n, k, r, 3)$ SA-LR code over $\mathbb{F}_2$ which is tighter than the bound in either (26) or (27) for $r > 72$ and which makes use of the "transpose"-based rate equation appearing in [110].

8.2.2 *Constructions*

• The product code. Consider the $[(r+1)^t, r^t]$ product code in t dimensions. Clearly this is an $(n = (r+1)^t, k = r^t, r, t)$ availability code, having rate $R = (\frac{r}{r+1})^t$.

• The Wang et al. [144] construction. For any given parameter pair (r, t), Wang et al. [144] provide a construction for an (n, k, r, t) availability code which is defined through its parity-check matrix. Let S be a set of $m = (r + t)$ elements. Then in the construction, each row of H corresponds to a distinct subset of S of cardinality $(t - 1)$ and each column, to a distinct subset of S of cardinality t. We set $h_{ij} = 1$ if the i-th $(t - 1)$-subset belongs to the j-th t-subset and zero otherwise. Thus H is of size $\binom{m}{t-1} \times \binom{m}{t}$. It is easy to verify that each row of H has constant row weight $(r + 1)$ and each column of H has constant weight t. It turns out that the rank of H is given by $\binom{m-1}{t-1}$ and that H defines an (n, k, r, t) availability code, having parameters $n = \binom{m}{t}$, $k = \binom{m}{t} - \binom{m-1}{t-1}$ and rate $R = \frac{r}{r+t}$. Thus this code provides improved rate in comparison with the product code. Since $\binom{r+t}{t} < (r + 1)^t$, the code has smaller block length as well.

• Direct-sum construction. It is shown in [143] that the direct sum of m copies of the $[7, 3]$ simplex code yields an SA-LR code with parameters $(7m, 3m, 2, 3)$ having maximum possible rate for $n = 7m$, $r = 2$, $t = 3$, $q = 2$.

8.2.3 *Bounds on minimum distance*

Let $d_{\min}(n, k, r, t)$ be the maximum possible minimum distance of an (n, k, r, t) availability code. In [145], the following bound on the minimum distance of an information symbol availability code (and hence applicable to the case of AS availability codes as well) was presented

$$d_{\min}(n, k, r, t) \leqslant n - k + 2 - \left\lceil \frac{t(k-1)+1}{t(r-1)+1} \right\rceil. \tag{28}$$

This bound was derived by adopting the approach employed in Gopalan et al. [93] to bound the minimum distance of an availability code. An improved minimum-distance estimate appears in [118]

$$d_{\min}(n, k, r, t) \leqslant n - \sum_{i=0}^{t} \left\lfloor \frac{k-1}{r^i} \right\rfloor. \tag{29}$$

• Approach via minimum support weights. The next bound on minimum distance relies upon an easy-to-compute sequence that represents upper bounds on the GHW of the dual of an availability code. Let there be b subsets $\{S_1, \ldots, S_b\}$ of $[n]$, each of size at most $r + 1$. We assume that $[n] = \cup_{i=1}^{b} S_i$. Let f_i be the minimum size of the union of any i out of the b subsets, i.e., $f_i = \min_{\{T:T\subseteq[b]:|T|=i\}} |\cup_{j\in T} S_j|$. Then $f_i \leqslant e_i$ [98] where the $\{e_i\}_{i=1}^{b}$ are recursively calculated in the reverse direction as follows: set $e_b = n$, and for $2 \leqslant i \leqslant b$, set

$$e_{i-1} = \min\left\{ e_i, e_i - \left\lceil \frac{2e_i}{i} \right\rceil + r + 1 \right\}. \tag{30}$$

From the definition of e_j, it is clear that e_j is an upper bound on the j-th minimum support weight or j-th GHW of a code containing b linearly independent codewords with the i-th codeword having support S_i, $i \in [b]$. We will refer to the sequence $\{e_i\}$ associated with a given parameter set (n, r, b) as the minimum-support-weight (MSW) sequence associated to (n, r, b). The bound below in (32) appeared in [110] and makes use of the fact that shortening of an (n, k, r, t) availability code results in a second

availability code with parameters $(n-\Delta_n, k-\Delta_k, r, t)$ having the same or larger $d_{\min}$. By applying the bound in (29) to the shortened code, one often obtains a bound on the original code (i.e., the parent code before shortening) that is significantly tighter. To estimate (Δ_n, Δ_k), the bound makes use of the MSW sequence discussed above.

Theorem 10 ([110]). Let $b = \lceil n(1-\rho(r,t))\rceil$ and e_i be calculated as per (30), where

$$\rho(r,t) = \begin{cases} \dfrac{r}{r+t}, & \text{if } t \in \{1,2\}, \\ \dfrac{r^2}{(r+1)^2}, & \text{if } t = 3, \\ \dfrac{1}{\prod_{j=1}^{t}(1+\frac{1}{jr})}, & \text{if } t > 3. \end{cases} \tag{31}$$

Then,

$$d_{\min}(n,k,r,t) \leqslant \min_{1\leqslant i\leqslant b,\ e_i - i < k} \left\{ n-k-i+1-\sum_{j=1}^{t} \left\lfloor \frac{k+i-e_i-1}{r^j} \right\rfloor \right\}. \tag{32}$$

The calculation of $\rho(r,t)$ for $t=1$ was not explicitly stated in [110] but is well known. Also contained in [110] is an improved upper bound on $d_{\min}$ in the case of codes with strict availability.

8.2.4 *Alphabet-size dependent bounds on $d_{\min}$*

Let $d^q_{\min}(n,k,r,t)$ be the maximum possible minimum distance of an (n,k,r,t) availability code over $\mathbb{F}_q$. In [109], the authors provide a bound on minimum distance of an (n,k,r,t) IS availability code (the bound thus also applies to AS availability codes as well) that depends on the size q of the underlying finite field $\mathbb{F}_q$

$$d^q_{\min}(n,k,r,t) \leqslant \min_{\substack{1\leqslant x\leqslant \lceil \frac{k}{(r-1)t+1}\rceil, \\ x\in\mathbb{Z}^+,\, y\in[t]^x,\, A(r,x,y)<k}} d^q(n-B(r,x,y), k-A(r,x,y)), \tag{33}$$

where $A(r,x,y) = \sum_{j=1}^{x}(r-1)y_j + x$, $B(r,x,y) = \sum_{j=1}^{x} r y_j + x$ and $d^q(n,k)$ is the maximum possible minimum distance of a classical (i.e., no locality necessary) $[n,k]$ block code over $\mathbb{F}_q$. There is a similar bound on the dimension of an availability code with parameters n, r, t, $d_{\min}$ over $\mathbb{F}_q$.

The following bound on the minimum distance of an (n,k,r,t) availability code over $\mathbb{F}_q$ that is tighter than the bound in (33) appears in [110] and is currently the tightest-known bound on $d^q_{\min}(n,k,r,t)$:

$$d^q_{\min}(n,k,r,t) \leqslant \min_{i\in S} d^q_{\min}(n-e_i, k+i-e_i, r, t), \tag{34}$$

where $S = \{i : e_i - i < k, 1 \leqslant i \leqslant b\}$ and $b = \lceil n(1-\rho(r,t))\rceil$ and e_i is calculated as per (30). This bound is also based on the shortening approach introduced in [108].

8.2.5 *Asymptotic bounds on rate*

Let $R^q(r,t,\Delta) = \limsup_{n\to\infty} \frac{\log_q(A_q(n,r,t,\lceil \Delta n\rceil))}{n}$, where $A_q(n,r,t,d)$ is the maximum number of codewords in an availability code with parameters (n,r,t) with minimum distance d over $\mathbb{F}_q$. The only known upper bounds on $\sup_q R^q(r,t,\Delta)$ are based on converting the minimum distance bounds appearing in (28), (29) and (32) into asymptotic bounds. There are constructions which provide lower bounds on $R^q(r,t,\Delta)$. A lower bound on $\sup_q R^q(r,t,\Delta)$ for any $r \geqslant t$ is provided in [118]. For the specific case $t=2$, [118] provides lower bounds on $R^q(r,2,\Delta)$:

$$R^q(r,2,\Delta) \geqslant \frac{r}{r+2} - \min_{0<s\leqslant 1} \left(\frac{1}{\binom{r+2}{2}} \log_q(g^{(2)}_q(s)) - \Delta \log_q(s) \right) \quad \text{valid for any } q, \tag{35}$$

$$g_2^{(2)}(s) = \frac{1}{2^{r+2}} \sum_{i=0}^{r+2} \binom{r+2}{i} (1+s)^{\binom{r+2}{2}-i(r+2-i)} (1-s)^{i(r+2-i)} \quad \text{valid only for } q=2. \tag{36}$$

The reader is referred to [118] for an expression for $g_q^{(2)}(s)$ for general q as well as a lower bound on $\sup_q R^q(r,t,\Delta)$ for any $r \geqslant t$. A further lower bound on $R^q(r,t,\Delta)$ for the case $t=2$ and based on algebraic geometry codes appears in [119].

8.3 Codes with sequential recovery

Somewhat surprisingly, the maximum possible rate of an (n,k,r,t) S-LR code has been precisely determined via a tight upper bound and a matching construction. The case $t=2,3$ is respectively settled in [98,146], where the authors derive the respective bounds

$$n \geqslant k + \left\lceil \frac{2k}{r} \right\rceil \quad \text{for } t=2, \quad n \geqslant k + \left\lceil \frac{2k + \lceil \frac{k}{r} \rceil}{r} \right\rceil \text{ for } t=3,$$

and provide matching constructions in each case. Matching constructions for the $t=2$ case can be derived either from complete graphs or Turan graphs [98]. Interestingly, the construction based on Turan graphs turns out to be optimal with respect to GHW as well. The general $t \geqslant 4$ case was settled in [147, 148] and is presented below.

Theorem 11 ([147,148]). Let $\mathcal{C}$ be an (n,k,r,t) S-LR code over a finite field $\mathbb{F}_q$. Let $r \geqslant 3$. Then

$$\frac{k}{n} \leqslant \begin{cases} \dfrac{r^{\frac{t}{2}}}{r^{\frac{t}{2}} + 2\sum_{i=0}^{\frac{t}{2}-1} r^i}, & t \text{ even}, \\ \dfrac{r^s}{r^s + 2\sum_{i=1}^{s-1} r^i + 1}, & \text{for } t \text{ odd}, \end{cases} \tag{37}$$

where $s = \frac{t+1}{2}$. Moreover, there exist binary codes (i.e., codes over $\mathbb{F}_q$ with $q=2$) that achieve this bound.

The rate bound given in (37) proves a conjecture given in [149] for maximum achievable rate of an (n,k,r,t) S-LRC. The proof of the bound (37) given in [147,148], shows that a code achieving the above rate bound must have a parity check matrix (upto a permutation of rows and columns) with a specific, sparse, staircase structure. An example of this for the case $t=8$ is shown in Figure 16. Based on this, it can shown that a binary code achieving the rate bound (37) and hence having parity check matrix of the form as shown in Figure 16, must be based on a tree-like graph with girth $\geqslant t+1$ with degree $r+1$ for most nodes, where each edge of the graph represents a code symbol and each node represents a parity check of the code symbols incident on it. Codes achieving the rate bound (37) appeared in [147,148,150] and are based on constructing these tree-like graphs with girth $\geqslant t+1$.

We note that a construction of codes based on $(r+1)$-regular bipartite graphs having girth $t+1$ and achieving rate close to (37) was suggested earlier in [151]. It was noted that these codes have rate $\geqslant \frac{r-1}{r+1}$. It is not hard to show that these codes have rate equal to $\frac{r-1}{r+1} + \frac{1}{n}$ [147]. For certain n, the resultant codes achieve the rate bound in (37). However these values of n correspond to the existence of Moore graphs of degree $r+1$, and girth $= t+1$ with that number n of edges. For $r \geqslant 2$, Moore graphs exist only for $t \in \{2,3,4,5,7,11\}$ [152].

In Figure 17, we compare the tight bound in (37) on the rate of an S-LR code with the upper bound in (26), due to Tamo et al. [118] on the rate of a code with availability. The plots suggest that codes with sequential recovery offer a significant rate advantage.

8.4 (r,δ) codes

The (Singleton) bound on the minimum distance of a code with (r,δ) locality was presented above in (18). We collect together in this subsection, other results on this class of codes that have appeared in the literature.

$$\left[\begin{array}{c|c|c|c|c} D_0 & A_1 & 0 & 0 & 0 \\ \hline 0 & D_1 & A_2 & 0 & 0 \\ \hline 0 & 0 & D_2 & A_3 & 0 \\ \hline 0 & 0 & 0 & D_3 & C \end{array}\right]$$

Figure 16 The general form [147,148] of the parity-check matrix H of a rate-optimal S-LR code for $t = 8$.

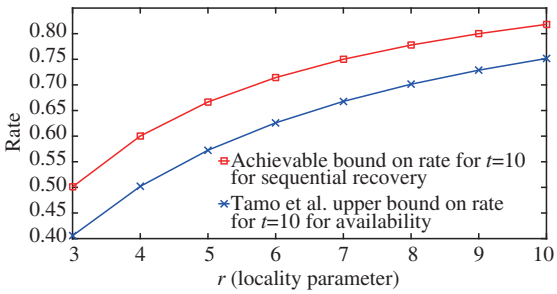

Figure 17 (Color online) Comparison of rate bounds on codes with sequential recovery (37) and codes with availability (26) for $t = 10$.

8.4.1 *Constructions and characterization of distance optimal* (r, δ) *codes*

We focus here only on optimal constructions having low field size. A detailed investigation of codes which achieve the Singleton bound on minimum distance of a code with (r, δ) locality for all symbols appears in [153] (see in particular, Figure 2 of [153] which provides a characterization of the existence of codes achieving the Singleton bound). In [102], a construction of codes achieving (18) with field size $O(n)$ for the case $(r + \delta - 1)|n$ is provided. A construction of cyclic codes with (r, δ) locality achieving the bound (18) for $(r + \delta - 1)|n$ and field size of $O(n)$ appears in [154].

8.4.2 (r, δ) *codes with small alphabet size*

• Upper bounds on dimension. Several alphabet-size dependent bounds on dimension for a code with (r, δ) AS locality and given minimum distance $d_{\min}$ appear in [114]. The bounds take on the form

$$k \leqslant \left(\left\lceil \frac{n - d + 1}{r + \delta - 1} \right\rceil + 1\right) \log_q(B(r + \delta - 1, \delta)),$$

where $B(r+\delta-1, \delta)$ is an upper bound on the number of codewords in a code of block length $(r+\delta-1)$ and minimum distance δ and is log-convex in the block length. The different bounds are obtained by substituting various bounds for $B(r + \delta - 1, \delta)$. The authors also present bounds for disjoint local codes derived based on association schemes and linear programming which provide the tightest-known bounds in the literature on codes with (r, δ) locality with disjoint local codes.

• Binary codes with (r, δ) locality. In [155], distance-optimal (codes achieving the Singleton bound) binary codes are characterized and the authors of [155], prove that there are only 2 classes of binary, distance-optimal codes for $\delta > 2$. They make use of the fact in their proof that since the code is binary and achieves the Singleton bound on minimum distance, the code after shortening a sufficient number of selected symbols must be an $[\ell, 1, \ell]$ MDS code for some $\ell < n$.

8.4.3 *Achievability results on asymptotic rate*

In [119], the following GV-type bound is derived

$$R^q(r, \delta, \Delta) \geqslant \frac{r}{r + \delta - 1} - \min_{0<s\leqslant 1} \left(\frac{\log_q(b_\delta(s))}{r + \delta - 1} - \Delta \log_q(s) \right),$$

where Δ denotes the fractional minimum distance and δ is the parameter associated with (r, δ) locality and where

$$b_\delta(s) = 1 + (q - 1) \sum_{w=\delta}^{r+\delta-1} \binom{r + \delta - 1}{w} s^w q^{w-\delta} \sum_{j=0}^{w-\delta} \binom{w - 1}{j} (-q)^{-j}.$$

A second lower bound, based on a construction appearing in [119] applies whenever $r + \delta - 1 = \sqrt{q}$ and improves upon the above GV-type bound in some parameter range $R^q(r, \delta, \Delta) \geqslant \frac{r}{r+\delta-1}(1 - \Delta - \frac{3}{\sqrt{q}+1})$.

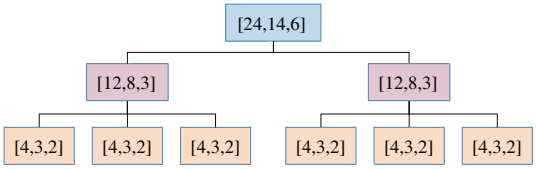

Figure 18 (Color online) Illustration of a code with hierarchical locality. Each code symbol is protected by a $[4, 3, 2]$ local code. Each local code is contained in a $[12, 8, 3]$ middle code.

8.5 Codes with hierarchical locality

Codes with hierarchical locality are codes proposed in [156] having multiple tiers of locality. We restrict the discussion for simplicity here to 2 tiers. The motivation here is that in a code with 2-tier locality, the higher probability single-erasure event can be repaired with the help of a short local code, while the lower-probability, multiple-erasure event can be handled by accessing a larger number of symbols from the next level local code, termed here as the 'middle' code. A hierarchical topology of local codes as illustrated by the example shown in Figure 18 is proposed in [156] and a bound on the minimum distance derived for the general case. The bound for a two-level hierarchy is presented below.

Theorem 12. Let $\mathcal{C}$ be an $[n, k, d]$-linear code with hierarchical locality with the local and middle codes having dimensions at most r_1, r_2 respectively, and minimum distances at least δ_1, δ_2 respectively. Then

$$d \leqslant n - k + 1 - \left(\left\lceil \frac{k}{r_2} \right\rceil - 1\right)(\delta_2 - 1) - \left(\left\lceil \frac{k}{r_1} \right\rceil - 1\right)(\delta_1 - \delta_2). \tag{38}$$

Optimal constructions are provided in [156,157]. We note that in the context of a practical distributed-storage system, the authors in [158] had previously suggested the topology of hierarchical codes and compared hierarchical codes with RS codes in terms of repair-efficiency using real data.

8.6 LR code with cooperative recovery (C-LR code)

Let $d_{\min}(n, k, r, t)$ be the maximum possible minimum distance of a C-LR code with parameters (n, k, r, t). In [151], the authors introduce the notion of cooperative local repair and provide the following bound on minimum distance for both linear as well as non-linear codes

$$d_{\min}(n, k, r, t) \leqslant n - k + 1 - t\left\lfloor \frac{k-t}{r} \right\rfloor.$$

They also give a second bound for $r \geqslant t$. The paper also contains the following alphabet-size dependent bound on dimension

$$k \leqslant \min_{\gamma \leqslant \min(\lfloor \frac{n}{r+t} \rfloor, \lfloor \frac{k-1}{r} \rfloor)} r\gamma + \log_q(A_q(n - \gamma(r+t), d)),$$

where $A_q(n, d)$ is the maximum size of a q-ary code of block length n and minimum distance d.

Open problems 5 (Codes for multiple erasures). (1) For a given (n, k, r, δ), what is the maximum achievable minimum distance of codes having (r, δ) locality for a given constraint on field size?

(2) For a given (n, k, r), what is the minimum field size over which we can construct a code with locality $(\delta = 2)$ meeting the Singleton bound?

(3) The construction of codes with locality $(\delta = 2)$ over a field $\mathbb{F}_q$ of size $q = O(1)$ for a larger range of $(d_{\min}, r)$ (say large $d_{\min}, r$) which are $d_{\min}$ optimal over $\mathbb{F}_q$.

(4) The construction of MR codes with smaller field size for a wide range of parameters.

(5) What is the maximum achievable rate $\frac{k}{n}$ for a given (r, t) of codes with availability and C-LR codes?

(6) For a given (n, k, r, t), what is the maximum achievable minimum distance of an S-LR code, a code with availability, or a C-LR code?

(7) Questions 5 and 6 when restricted to a finite field $\mathbb{F}_q$.

(8) All the above questions on minimum distance can be rephrased as a question on maximum achievable dimension for a given $(n, d_{\min}, r, t)$ over a finite field $\mathbb{F}_q$.

9 Locally RG codes

As is clear from the discussion in the preceding sections, while RG codes aim to minimize the repair bandwidth, LR codes focus in keeping the repair degree low. It is natural to ask if it is possible to construct codes that possess both low repair bandwidth and repair degree. The class of LRG codes introduced independently in [83, 159], answers this question in the affirmative. These codes are perhaps best viewed as codes with locality in which the local codes are RG codes.

9.1 Locality in vector codes

We begin by studying the notion of locality in a vector code, i.e., a code over a vector alphabet. Let $\mathcal{C}$ be an $[[n, K, d_{\min}, \alpha]]$ vector code over the vector alphabet $\mathbb{F}_q^{\alpha}$ having block length n and minimum Hamming distance $d_{\min}$. Let K be the dimension of the code viewing the code as a vector space over $\mathbb{F}_q$. Let $\mathcal{C}_s$ be the scalar code of length $n\alpha$ obtained from $\mathcal{C}$ by replacing each vector symbol by the corresponding α scalar symbols. Let G be a generator matrix for $\mathcal{C}_s$, where the first α columns correspond to the first vector code symbol of $\mathcal{C}$ and so on. For $1 \leqslant i \leqslant n$, we use the terminology i-th thick column to denote the set of columns $[(i-1)\alpha + 1, i\alpha]$ of G corresponding to the i-th vector code symbol of $\mathcal{C}$. Clearly, the scalar code $\mathcal{C}_s$ has dimension K.

For a subset $S \subseteq [n]$, of indices, let $\mathcal{C}|_S$ denote the vector code obtained by restricting the code $\mathcal{C}$ to the thick columns associated with the indices in S. We similarly define $G|_S$ to be the restriction of G to the thick columns associated to S. The definition below is a natural extension of the notion of locality to a code over vector alphabet.

Definition 5. For $i \in [0, n-1]$ and $\delta \geqslant 2$, the i-th vector code symbol of $\mathcal{C}$ is said to have (r, δ) locality if there exists a set $S_i \subseteq [n]$ such that $i \in S_i$, $|S_i| \leqslant r + \delta - 1$ and $d_{\min}(\mathcal{C}|_{S_i}) \geqslant \delta$. The restriction of $\mathcal{C}$ to S, i.e., code $\mathcal{C}|_{S_i}$ will be referred to as the local code associated to S_i.

Definition 6. A vector code $\mathcal{C}$ is said to have (r, δ) IS locality if there exists $\mathcal{I} \subseteq [n]$ such that $\mathrm{rank}(G|_{\mathcal{I}}) = K$ and the i-th vector code symbol of $\mathcal{C}$ has (r, δ) locality for all $i \in \mathcal{I}$.

$\mathcal{C}$ is said to have (r, δ) AS locality if $\mathcal{I}$ can be set to be $[n]$ in the definition above. If for a code having (r, δ) AS locality, $S_i = S_j$ or $|S_i \cap S_j| = 0$, for all $i \neq j$, then the code is said to have disjoint locality.

Definition 7. An $[[n, K, d_{\min}, \alpha]]$ vector code $\mathcal{C}$ is said to have the uniform rank accumulation (URA) property if there exists a sequence $\{a_i\}_{i=1}^{n}$ of non-negative integers satisfying (i) $a_1 = \alpha$, (ii) $\mathrm{rank}(G|_{\mathcal{I}}) = \sum_{j=1}^{i} a_i$, $\forall \mathcal{I} \subseteq [n] : |\mathcal{I}| = i$. The integer sequence $\{a_i, i \in [n]\}$ is referred to as the rank profile of $\mathcal{C}$.

Remark 9. It is shown in [12] that both MSR and MBR codes possess the URA property. The rank profile in the case of $((n, k, d), (\alpha, \beta), K)$ MSR, MBR codes, are respectively given by

$$\underbrace{a_i}_{\mathrm{MSR}} = \begin{cases} \alpha, & 1 \leqslant i \leqslant k, \\ 0, & (k+1) \leqslant i \leqslant n, \end{cases} \qquad \underbrace{a_i}_{\mathrm{MBR}} = \begin{cases} \alpha - (i-1)\beta, & 1 \leqslant i \leqslant k, \\ 0, & (k+1) \leqslant i \leqslant n. \end{cases}$$

Definition 8. An $[[n, K, d_{\min}, \alpha]]$ vector code $\mathcal{C}$ is said to have URA locality, if the code has either information or AS locality and if in addition, local codes are $[[n_\ell, K_\ell, d_\ell, \alpha]]$ vector codes having the URA property with identical rank profiles.

Consider the vector code $\mathcal{C}$ having URA locality with parameters as in Definition 8. The rank profile for any given $[[n_\ell, K_\ell, d_\ell, \alpha]]$ local code is denoted by $\{a_i\}_{i=1}^{n_\ell}$. Let $\{b_i\}_{i=1}^{\infty}$ be a periodic sequence, where $b_i = a_i$ for $1 \leqslant i \leqslant n_\ell$ and $b_{n_\ell + j} = b_j$ for $j \geqslant 1$. Define $P(s) \triangleq \sum_{i=1}^{s} b_i : \ s \geqslant 1$. For $x \geqslant 1$, set $P^{(\mathrm{inv})}(x)$ to be the smallest integer y such that $P(y) \geqslant x$, i.e., $P^{(\mathrm{inv})}(x) = y$.

Theorem 13 ([159])**.** Let $\mathcal{C}$ be an $[[n, K, d_{\min}, \alpha]]$ code with URA locality, where the local codes have parameter set $[[n_\ell, K_\ell, d_\ell, \alpha]]$. Then, we have $d_{\min}(\mathcal{C}) \leqslant n - P^{(\mathrm{inv})}(K) + 1$.

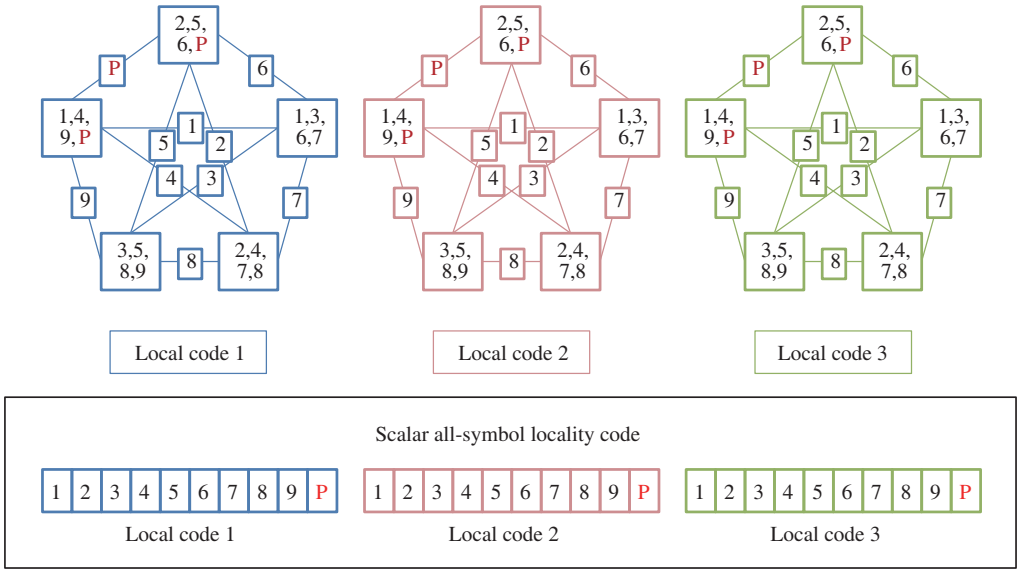

Figure 19 (Color online) An $[[n = 15, K = 20, d_{\min} = 5, \alpha = 4]]$ LRG code $\mathcal{C}$ where local codes are MBR codes. Here the local codes are $((n_\ell = 5, r = 3, d = 4), (\alpha = 4, \beta = 1), K_\ell = 9)$ MBR codes.

Corollary 1. Consider the case of a vector with locality, where the local codes are $((n_\ell, r, d), (\alpha, \beta), K_\ell)$ MSR codes. Using Remark 9 and Theorem 13, it follows that [83, 159]

$$d_{\min}(\mathcal{C}) \leqslant n - \left\lceil \frac{K}{\alpha} \right\rceil + 1 - \left(\left\lceil \frac{K}{\alpha r} \right\rceil - 1 \right)(\delta - 1).$$

In [159], the authors give minimum-distance bounds for general vector codes with locality and a tighter bound for the case when the local codes have the URA property. LRG codes with MSR or MBR AS locality, and IS locality that meet the minimum-distance bound, are provided for various parameters. The field-size requirement here is at least $O(n^2)$ for the AS locality code constructions. In [83], the authors present an explicit construction of a vector code with MSR AS locality, that requires a field-size that is exponential in n. In [160], the authors construct a related family of vector codes with IS locality, where the local codes are vector MDS codes with near-optimal bandwidth and small sub-packetization (α) levels. In [161, 162], the authors consider vector codes with locality featuring functional repair and achieving a reduction in repair bandwidth by carefully choosing for each failed node, a set of $r \leqslant k$ helper nodes. In [163], the authors provide linear field-size constructions for LRG codes with AS locality, where the local codes are either MSR or MBR.

9.2 Codes where local codes are MSR/MBR

It is possible to construct LRG codes which are minimum-distance optimal where the local codes are MSR or MBR using the T-B construction of optimal scalar LR codes.

Example 5 ([159]). An LRG code $\mathcal{C}$ having parameters $[[n = 15, K = 20, d_{\min} = 5, \alpha = 4]]$ where the local codes are $((n_\ell = 5, r = 3, d = 4), (\alpha = 4, \beta = 1), K_\ell = 9)$ MBR codes, can be constructed as follows. Let $N_\ell \triangleq \binom{n_\ell}{2} = 10$, $\delta' = N_\ell - K_\ell + 1 = 2$, $\nu = \frac{n}{n_\ell} = 3$. Take a minimum-distance optimal $[\nu N_\ell = 30, K = 20, 9]$ scalar T-B code $\mathcal{C}'$ with $(K_\ell = 9, \delta' = 2)$ AS locality. Note that each local code of $\mathcal{C}'$ is a $[N_\ell = 10, K_\ell = 9]$ MDS code. The LRG code with the required parameters is obtained by mapping each such local MDS code to an MBR code, using the polygonal MBR construction. The resultant code (Figure 19) is shown to be minimum-distance optimal in [159].

Example 6 ([163]). From the discussion in Subsection 7.2.2, it can be inferred that each local code in a T-B code is an MDS code. Let $(n_\ell - r)|r$ and $n_\ell|n$. In order to construct a code with MSR local regeneration, we initially stack $\alpha = (n_\ell - r)^{\frac{n_\ell}{n_\ell - r}}$ independent layers of codewords from an $[n, k, d_{\text{TB}}]$ T-B code with (r, δ) AS locality. We then perform the pairwise forward transform (introduced in Subsection

4.2) independently, for each local code. This results in an $[[n, K = k\alpha, d_{\min}, \alpha]]$ LRG code $\mathcal{C}$ where local codes are $((n_\ell, r, d), (\alpha, \beta), K_\ell)$ MSR codes, with $d = n_\ell - 1$. Let d_{TB} denote the (optimal) minimum-distance of the underlying T-B code. The code will be minimum-distance optimal if $d_{\mathrm{TB}} \leqslant 2(n_\ell - r + 1)$.

10 Repairing RS codes

The conventional repair of an $[n, k]$ scalar MDS code treats each code symbol as an indivisible unit and leads to a total repair bandwidth of k times the amount of data stored in the failed node, where k is the dimension of the code. Over the past couple of years, new techniques have surfaced that present a different picture for the repair of scalar MDS codes, particularly for RS codes. These techniques realize that the code symbols (say, over $\mathbb{F}_q$) of a scalar MDS code can be viewed as vectors whose entries are over some subfield, $\mathbb{B} \subseteq \mathbb{F}_q$. For example, consider the $[16, 8]$ RS code obtained by evaluating message polynomials of degree $\leqslant 7$ over all the elements in $\mathbb{F}_{2^4}$. Under the traditional repair, the repair bandwidth will be 8 code symbols over $\mathbb{F}_{2^4}$, which is equivalent to 32 bits. As we will shortly see, it is possible to perform single-node repair in this instance, by downloading just 1 bit from each of the fifteen surviving nodes. This results in a repair bandwidth of 15 bits, which is a clear improvement over the 32 bits downloaded under the conventional scheme. This line of work which vectorizes scalar MDS codes and performs repair operations over a suitable subfield $\mathbb{B}$ for bandwidth gains, began with the pioneering work of Shanmugam et al. [164] who showed the existence of an efficient repair scheme for systematic node repair, when $k = n - 2$, that improved up on the traditional repair bandwidth. In a subsequent paper, Guruswami and Wootters [165] consider generalized Reed-Solomon (GRS) codes and all-node repair. There have been other papers since as well.

Let t be the degree of the field extension $[\mathbb{F}_q : \mathbb{B}]$. Clearly, through vector representation over the subfield $\mathbb{B}$ of over $\mathbb{F}$, t can be regarded as the sub-packetization level of the MDS code. Traditional RS codes have code lengths typically on the order of $|\mathbb{F}_q|$ corresponding to a sub-packetization level which is logarithmic in code-length. On the other hand, there are fundamental bounds (Subsection 4.1) that require the sub-packetization to be exponential in code length (for a fixed r) in order to achieve the cut-set bound. This leads to the natural and interesting question: what is the least possible repair bandwidth that can be achieved in a low-sub-packetization-level setting?

10.1 Linear repair schemes for scalar MDS codes

In this subsection, we consider the single-node repair of linear, scalar, MDS codes over $\mathbb{F}_q$, where $q = p^t$ for p a prime power and t a positive integer. Let $\mathbb{B}$ be a subfield of $\mathbb{F}_q$ of size $|\mathbb{B}| = p$. In this setting, by linear repair scheme, we will mean that all repair operations correspond to linear operations over $\mathbb{B}$. For $i \in [n]$, let b_i denote the least possible repair bandwidth (measured by the number of $\mathbb{B}$-symbols downloaded) to repair the i-th code symbol. The repair bandwidth b is then defined as: $b \triangleq \max_{i\in[n]} b_i$. In the discussion below, by dimension we will throughout mean dimension as a vector space over $\mathbb{B}$.

Theorem 14 ([165])**.** Let $\mathcal{C}$ be a scalar MDS code. Then a linear repair scheme for $\mathcal{C}$ with repair bandwidth b exists iff for each code coordinate $i \in [n]$, there exists a subset $\mathcal{A}_i$ of t codewords in the dual code $\mathcal{C}^\perp$ such that

$$\dim(\langle a_i,\ \underline{a} \in \mathcal{A}_i\rangle) = t \quad \text{and} \quad \max_{i\in[n]} \left(\sum_{j\in[n]\setminus i} \dim(\langle a_j,\ \underline{a} \in \mathcal{A}_i\rangle) \right) \leqslant b.$$

It is easy to see the 'if' part above. The trace function from $\mathbb{F}_q$ to $\mathbb{B}$ is the $\mathbb{B}$-linear map given by $T(\gamma) = \sum_{m=0}^{t-1} \gamma^{p^m}$. Given a basis $\{\rho_m\}_{m=1}^t$ for $\mathbb{F}_q$ over $\mathbb{B}$, it is known [166] that there always exists a second basis $\{\gamma_m\}_{m=1}^t$ for $\mathbb{F}_q$ over $\mathbb{B}$, termed the trace-dual basis of $\{\rho_m\}$, such that any $x \in \mathbb{F}_q$ can be expressed in the form $x = \sum_{m=1}^t T(x\rho_m)\gamma_m$. Let $\mathcal{A}_i$ be as defined in Theorem 14. For $\underline{a} \in \mathcal{A}_i \subseteq \mathcal{C}^\perp$ and

$\underline{c} \in \mathcal{C}$, we have that $c_i a_i = -\sum_{j=1, j\neq i}^{n} c_j a_j$. Hence

$$T(c_i a_i) = -\sum_{j=1, j\neq i}^{n} T(c_j a_j), \quad \text{for } \underline{a} \in \mathcal{A}_i. \tag{39}$$

The definition of $\mathcal{A}_i$ implies that $\dim(\langle a_i,\ \underline{a} \in \mathcal{A}_i\rangle) = t$. Let b_{ij} denote the dimension of the set $\{a_j\}_{\underline{a}\in\mathcal{A}_i}$ and let $\mathcal{B}_{ij}$ denote a basis for the vector space spanned by $\{a_j\}_{\underline{a}\in\mathcal{A}_i}$. Using the $\mathbb{B}$-linearity of the trace function, it suffices to compute the b_{ij} trace values $\{T(c_j x) : x \in \mathcal{B}_{ij}\}$ which can be used to obtain $\{T(c_j a_j) : \underline{a} \in \mathcal{A}_i\}$. Hence by downloading $\sum_{j=1, j\neq i}^{n} b_{ij}$ symbols over $\mathbb{B}$, one can compute $\{T(c_i a_i) : \underline{a} \in \mathcal{A}_i\}$ using (39). Using the trace-dual basis, c_i can be reconstructed from these t traces.

Next, consider the specific case of an $[n, k]$ GRS code $\mathcal{C}$, whose symbols are n (scaled) evaluations of message polynomials $f(x) \in \mathbb{F}[x]$ of degree $\leqslant k-1$. Let the evaluation points be denoted by the set $\mathcal{A} = \{\alpha_j\}_{j=1}^{n}$. As the dual of a GRS code is a GRS code, codewords in the dual are scaled evaluations of message polynomials of degree $\leqslant (n-k-1)$. Thus in the context of a GRS code and ignoring w.o.l.o.g. the scaling coefficients, (39) takes on the form

$$T(f(\alpha_i)g(\alpha_i)) = -\sum_{j=1, j\neq i}^{n} T(f(\alpha_j)g(\alpha_j)), \quad \text{for all} \quad g(x) \in \mathcal{P}_i, \tag{40}$$

where $f(x)$ and $g(x)$ are polynomials having degrees at most $k-1$ and $n-k-1$, respectively, $\mathcal{P}_i$ is the set of t message polynomials having degree at most $n-k-1$ corresponding to the t dual codewords in $\mathcal{A}_i$.

10.2 Guruswami-Wootters GRS repair scheme

Let $k \leqslant n - p^{t-1}$ for a GRS code. Then it is possible repair each code-symbol (say, i-th) by downloading just one symbol over $\mathbb{B}$ each from the remaining $(n-1)$ nodes. The scheme is as follows. Consider the set of t polynomials $\mathcal{P}_i = \{g_{i,m}(x)\}_{m=1}^{t}$ and a basis $\{\rho_m\}_{m=1}^{t}$, where

$$g_{i,m}(x) = \frac{T\big(\rho_m(x-\alpha_i)\big)}{(x-\alpha_i)} = \sum_{s=0}^{t-1} \rho_m^{p^s} (x-\alpha_i)^{p^s-1}.$$

Each polynomial $g_{i,m}(x)$ has degree $p^{t-1} - 1 \leqslant n-k-1$. Hence the evaluations of this polynomial represent a codeword in $\mathcal{C}^{\perp}$. Note that $\{g_{i,m}(\alpha_i) = \rho_m, m \in [t]\}$ forms a basis for $\mathbb{F}$ over $\mathbb{B}$, i.e., $\dim_{\mathbb{F}}\langle\{g_{i,m}(\alpha_i), m \in [t]\}\rangle = t$. Also, $\dim_{\mathbb{F}}\langle\{g_{i.m}(\alpha_j), m \in [t]\}\rangle = 1\ \forall j \in [n] \setminus \{i\}$.

Theorem 15. Let $\mathcal{C}$ be an $[n, k]$ MDS code over $\mathbb{F}_q$. For any linear repair scheme for $\mathcal{C}$ over $\mathbb{B}$, the repair bandwidth, b (counted according to the number of symbols from $\mathbb{B}$) satisfies

$$b \geqslant (n-1)\log_{|B|}\left(\frac{n-1}{n-k+\frac{k-1}{|F|}}\right).$$

By Theorem 15, the repair scheme discussed above is optimal when $\mathcal{A} = \mathbb{F}_q$ and $n - k = p^{t-1}$.

10.3 Other related work

In [167], the authors improve the Guruswami-Wootters approach to a larger class of parameters. In [168], the authors provide a family of RS codes that has asymptotically optimal repair bandwidth with respect to the cut-set bound. This result is further developed in [169] to reduce the sub-packetization levels. In [170], the authors present RS codes that meet the MSR point for all parameters $k < d < n-1$. Bandwidth-efficient recovery from multiple erasures in RS codes is addressed in [171] and is further extended to include general scalar MDS codes in [172]. In [173], the authors present codes that universally achieve the optimal bandwidth points for all parameters $h \leqslant n-k$ and $k \leqslant d \leqslant n-h$ simultaneously (see Table 5 for a summary of RS repair schemes appearing in the literature).

Table 5 A summary of schemes appearing in the literature for the repair of RS codes

Ref.	Bandwidth	Sub-packetization	Cut-set bound achievability	Remarks
[165]	$n-1$	$\log_p n$	No	Single node repair; $(n-k) \geqslant p^{t-1}$
[167]	$(n-1)t(1-\log_n(n-k))$	$\log_p n$	No	Single node repair; $(n-k) \geqslant p^{\ell}; \ell \in [t-1]$
[168]	$< \frac{t(n+1)}{n-k}$	$(n-k)^n$	Asymptotically	Single node repair
[169]	$< \frac{t(n-1+3(n-k))}{n-k}; (n-k) = s^m$	s^{m+n-1}	Asymptotically	Single node repair
[170]	$\frac{td}{d-k+1}$	$\approxeq n^n$	Yes	Codes exist for any given $d \in [k, n-1]$
[171]	$2(n-1)$(2-erasures); $3(n-1)$(3-erasures)	$\log_p n$	No	Distributed repair
[171]	$2(n-2)$(2-erasures); $3(n-3)$(3-erasures)	$\log_p n$	No	Centralized repair
[172]	$h(n-h)-(p-1)(h-1)$ (h-erasures)	$\log_p n$	No	Centralized repair $h \leqslant \sqrt{\log n}$
[173]	$\frac{tdh}{d-k+h}$	$\approxeq n^n$	Yes; bound in [23]	Code works simultaneously for any given number of failures, $h : h \in [1, n-k]$ and any $d : d \in [k, n-h]$

11 An information capacity approach

• Capacity bounds. In [174], a generic distributed storage system model is introduced and fundamental limits presented. The notion of information capacity of a distributed system is introduced. Let m denote the source data size in bits. Consider a distributed storage system with N nodes, each storing s bits of data. If Δ denotes the average time between node failures, the erasure rate ϵ can be defined as $\epsilon = \frac{s}{\Delta}$. When a node failure takes place, a repairer carries out node repair in a manner which ensures that the source data can be recovered from the data in the surviving nodes at any point of time. The mean time to data loss (MTTDL) is the average amount of time over which the source data can be recovered. Let γ denote the repair rate, which is the rate at which the repairer reads and writes data. Let $\sigma = \frac{\gamma}{\epsilon}$ denote the repair rate to erasure rate ratio. The information capacity of a distributed storage system is then defined as the largest amount of source data m for which a large MTTDL is possible. In [174], it is shown that the information capacity approaches $(1 - \frac{1}{2\sigma})Ns$ bits as σ and N grow.

• Liquid storage. In [175], the idea of liquid cloud storage was proposed in which codes of large block length (for example, authors use a code of block length 3010 in one of their simulations) are used to spread data stored pertaining to every object over a large number of nodes. Liquid storage employs a lazy repair strategy where the repair runs slowly in the background. The authors present simulation results that shows that liquid storage gives better MTTDL performance in comparison with systems based on small block length codes. The performance of liquid storage systems is shown to approach the fundamental limits proved in [174].

12 Codes in practice

Distributed systems such as Hadoop, Google file system and Windows Azure have evolved to include support for erasure codes within their systems, in order to enjoy the benefits of improved storage efficiency in comparison with triple replication. However, the use of traditional erasure codes results in additional repair traffic resulting in larger repair times. This led to several theoretical code constructions for efficient node repair and these were discussed in the preceding sections of this article. Among the biggest success stories is undoubtedly the adoption of LR codes in the Windows Azure production cluster.

• LR codes. In [176], the authors compare performance-evaluation results of an $(n = 16, k = 12, r = 6)$ LR code with that of an $[n = 16, k = 12]$ RS code in the Azure production cluster and demonstrate the repair savings offered by the LR code. Subsequently, the authors implemented an $(n = 18, k = 14, r = 7)$ LR code in Windows Azure storage and showed that this code has repair degree comparable to that of an $[9, 6]$ RS code, but has storage overhead 1.29 versus 1.5 in the case of the RS code. This code has

reportedly resulted in the savings of millions of dollars for Microsoft [177]. The authors of [2] implemented HDFS-Xorbas which uses LR codes in place of RS codes in HDFS-RAID. Xorbas LR code is build on top of an RS code by adding extra local XOR parties. The experimental evaluation of Xorbas was carried out in Amazon EC2 and a cluster in Facebook, in which the repair performance of $(n = 16, k = 10, r = 5)$ LR code was compared against a $[14, 10]$ RS code. A second distributed storage system that has an LR code plug-in [178] is Ceph.

• MDS codes with bandwidth savings. The Hitchhiker erasure coded system presented in [179] is a practical implementation of the piggybacking framework introduced in [69]. The authors implemented the Hitchhiker in HDFS and evaluated its performance on a data-warehouse cluster at Facebook. The Hitchhiker has now been incorporated into Apache Hadoop. In [180], the HDFS implementation of a class of MDS array codes called HashTag codes is discussed. The theoretical framework of HashTag codes was presented in [71]. These codes allow low sub-packetization levels at the expense of increased repair bandwidth and are designed to efficiently repair systematic nodes.

• RG codes. The NCCloud [10] is one of the earliest works that dealt with the practical performance evaluation of RG codes. The NCCloud storage system is build on top of a 2-parity functional MSR code. In [181], the performance of the pentagon code (which is a RBT MBR code) and a heptagon-local code (which is an LRG code) in a Hadoop setting are studied. These two codes possess inherent double replication of code symbols, have storage overhead slightly greater than 2 and their performance is compared against double and triple replication. In [182], the authors present an optimal-access version of the PM MSR code, which they refer to as the PM-RBT code. The results of an experimental evaluation of a rate $\frac{1}{2}$ PM-RBT code on Amazon EC2 instances is reported. In [183], the authors introduced erasure codes termed Beehive that are built on top of MSR codes. These codes repair multiple failures simultaneously and are implemented using the PM MSR in C++ using the Intel storage acceleration library (ISAL). In [184], the authors present the evaluation of a high-rate MSR code known as the butterfly code in both Ceph and HDFS. This code is a simplified version of the MSR codes presented in [185] corresponding to the presence of two parity nodes. This code possesses the optimal-access property except in the case of the repair of a single parity node, and has sub-packetization level $\alpha = 2^{k-1}$. More recently in [186], the authors present Clay code that corresponds to the codes in [37–39]. The Clay code is implemented over Ceph based on the coupled-layer perspective in [38] and is evaluated over an Amazon AWS cluster. The Clay code is simultaneously optimal in terms of storage overhead, repair bandwidth, optimal access and sub-packetization level. As a part of this work, vector code support has been added to Ceph and the Clay code is under consideration to become a part of Ceph's master code-base.

Acknowledgements This work was supported in part by National Science Foundation of USA (Grant No. 1421848) and in part by an India-Israel UGC-ISF Joint Research Program Grant.

References

1 Rashmi K V, Shah N B, Gu D, et al. A solution to the network challenges of data recovery in erasure-coded distributed storage systems: a study on the facebook warehouse cluster. In: Proceedings of the 5th USENIX Workshop on Hot Topics in Storage and File Systems, San Jose, 2013
2 Sathiamoorthy M, Asteris M, Papailiopoulos D, et al. XORing elephants. Proc VLDB Endow, 2013, 6: 325–336
3 Dimakis A G, Ramchandran K, Wu Y N, et al. A survey on network codes for distributed storage. Proc IEEE, 2011, 99: 476–489
4 Datta A, Oggier F. An overview of codes tailor-made for better repairability in networked distributed storage systems. SIGACT News, 2013, 44: 89
5 Li J, Li B. Erasure coding for cloud storage systems: a survey. Tinshhua Sci Technol, 2013, 18: 259–272
6 Liu S Q, Oggier F. An overview of coding for distributed storage systems. In: Network Coding and Subspace Designs. Berlin: Springer, 2018. 363–383
7 Dimakis A G, Godfrey P B, Wu Y, et al. Network coding for distributed storage systems. IEEE Trans Inf Theory, 2010, 56: 4539–4551
8 Wu Y. Existence and construction of capacity-achieving network codes for distributed storage. IEEE J Sel Areas Commun, 2010, 28: 277–288
9 Shah N B, Rashmi K V, Kumar P V, et al. Interference alignment in regenerating codes for distributed storage: necessity and code constructions. IEEE Trans Inf Theory, 2012, 58: 2134–2158

10 Hu Y C, Chen H C H, Lee P P C, et al. NCCloud: applying network coding for the storage repair in a cloud-of-clouds. In: Proceedings of the 10th USENIX Conference on File and Storage Technologies, San Jose, 2012
11 Krishnan M N, Kumar P V. On MBR codes with replication. In: Proceedings of IEEE International Symposium on Information Theory, Barcelona, 2016. 71–75
12 Shah N B. On minimizing data-read and download for storage-node recovery. IEEE Commun Lett, 2013, 17: 964–967
13 Rashmi K V, Shah N B, Kumar P V, et al. Explicit construction of optimal exact regenerating codes for distributed storage. In: Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2009. 1243–1249
14 Rashmi K V, Shah N B, Kumar P V. Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction. IEEE Trans Inf Theory, 2011, 57: 5227–5239
15 Lin S J, Chung W H. Novel repair-by-transfer codes and systematic exact-mbr codes with lower complexities and smaller field sizes. IEEE Trans Parallel Distrib Syst, 2014, 25: 3232–3241
16 Han Y S, Pai H T, Zheng R, et al. Update-efficient error-correcting product-matrix codes. IEEE Trans Commun, 2015, 63: 1925–1938
17 Raviv N. Asymptotically optimal regenerating codes over any field. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 1416–1420
18 Mahdaviani K, Khisti A, Mohajer S. Bandwidth adaptive & error resilient MBR exact repair regenerating codes. 2017. ArXiv:1711.02770
19 Wu Y, Dimakis A G. Reducing repair traffic for erasure coding-based storage via interference alignment. In: Proceedings of IEEE International Symposium on Information Theory, Seoul, 2009. 2276–2280
20 Suh C, Ramchandran K. Exact-repair mds code construction using interference alignment. IEEE Trans Inf Theory, 2011, 57: 1425–1442
21 Lin S J, Chung W H, Han Y S, et al. A unified form of exact-MSR codes via product-matrix frameworks. IEEE Trans Inf Theory, 2015, 61: 873–886
22 Papailiopoulos D S, Dimakis A G, Cadambe V R. Repair optimal erasure codes through Hadamard designs. IEEE Trans Inf Theory, 2013, 59: 3021–3037
23 Cadambe V R, Jafar S A, Maleki H, et al. Asymptotic interference alignment for optimal repair of MDS codes in distributed storage. IEEE Trans Inf Theory, 2013, 59: 2974–2987
24 Tamo I, Wang Z, Bruck J. Zigzag codes: MDS array codes with optimal rebuilding. IEEE Trans Inf Theory, 2013, 59: 1597–1616
25 Wang Z Y, Tamo I, Bruck J. On codes for optimal rebuilding access. In: Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2011. 1374–1381
26 Cadambe V R, Huang C, Li J, et al. Polynomial length MDS codes with optimal repair in distributed storage. In: Proceedings of the 45th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, 2011. 1850–1854
27 Wang Z Y, Tamo I, Bruck J. Long MDS codes for optimal repair bandwidth. In: Proceedings of IEEE International Symposium on Information Theory, Cambridge, 2012. 1182–1186
28 Tamo I, Wang Z Y, Bruck J. Access versus bandwidth in codes for storage. IEEE Trans Inf Theory, 2014, 60: 2028–2037
29 Goparaju S, Tamo I, Calderbank R. An improved sub-packetization bound for minimum storage regenerating codes. IEEE Trans Inf Theory, 2014, 60: 2770–2779
30 Sasidharan B, Agarwal G K, Kumar P V. A high-rate MSR code with polynomial sub-packetization level. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 2051–2055
31 Rawat A S, Koyluoglu O O, Vishwanath S. Progress on high-rate MSR codes: enabling arbitrary number of helper nodes. In: Proceedings of Information Theory and Applications Workshop, La Jolla, 2016
32 Goparaju S, Fazeli A, Vardy A. Minimum storage regenerating codes for all parameters. IEEE Trans Inf Theory, 2017, 63: 6318–6328
33 Agarwal G K, Sasidharan B, Kumar P V. An alternate construction of an access-optimal regenerating code with optimal sub-packetization level. In: Proceedings of the 21st National Conference on Communications, Mumbai, 2015
34 Alon N. Combinatorial nullstellensatz. Combinator Probab Comp, 1999, 8: 7–29
35 Raviv N, Silberstein N, Etzion T. Constructions of high-rate minimum storage regenerating codes over small fields. IEEE Trans Inf Theory, 2017, 63: 2015–2038
36 Ye M, Barg A. Explicit constructions of high-rate MDS array codes with optimal repair bandwidth. IEEE Trans Inf Theory, 2017, 63: 2001–2014
37 Ye M, Barg A. Explicit constructions of optimal-access MDS codes with nearly optimal sub-packetization. IEEE Trans Inf Theory, 2017, 63: 6307–6317
38 Sasidharan B, Vajha M, Kumar P V. An explicit, coupled-layer construction of a high-rate MSR code with low sub-packetization level, small field size and all-node repair. 2016. ArXiv:1607.07335
39 Li J, Tang X H, Tian C. A generic transformation for optimal repair bandwidth and rebuilding access in MDS codes. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 1623–1627
40 Balaji S B, Kumar P V. A tight lower bound on the sub-packetization level of optimal-access MSR and MDS codes. 2018. ArXiv:1710.05876
41 Vajha M, Balaji S B, Kumar P V. Explicit MSR codes with optimal access, optimal sub-packetization and small field size for $d = k + 1, k + 2, k + 3$. 2018. ArXiv:1804.00598
42 Mahdaviani K, Mohajer S, Khisti A. Product matrix MSR codes with bandwidth adaptive exact repair. IEEE Trans

Inf Theory, 2018, 64: 3121–3135
43 Shah N B, Rashmi K V, Kumar P V, et al. Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff. IEEE Trans Inf Theory, 2012, 58: 1837–1852
44 Tian C. Characterizing the rate region of the (4,3,3) exact-repair regenerating codes. IEEE J Sel Areas Commun, 2014, 32: 967–975
45 Information theory inequality prover. 2016. http://user-www.ie.cuhk.edu.hk/~ITIP/
46 Yeung R W. A framework for linear information inequalities. IEEE Trans Inf Theory, 1997, 43: 1924–1934
47 Tian C, Sasidharan B, Aggarwal V, et al. Layered exact-repair regenerating codes via embedded error correction and block designs. IEEE Trans Inf Theory, 2015, 61: 1933–1947
48 Senthoor K, Sasidharan B, Kumar P V. Improved layered regenerating codes characterizing the exact-repair storage-repair bandwidth tradeoff for certain parameter sets. In: Proceedings of IEEE Information Theory Workshop, Jerusalem, 2015
49 Sasidharan B, Senthoor K, Kumar P V. An improved outer bound on the storage repair-bandwidth tradeoff of exact-repair regenerating codes. In: Proceedings of IEEE International Symposium on Information Theory, Honolulu, 2014. 2430–2434
50 Duursma I M. Outer bounds for exact repair codes. 2014. ArXiv:1406.4852
51 Duursma I M. Shortened regenerating codes. IEEE Trans Inf Theory (Early Access), 2018. doi: 10.1109/TIT.2018.2840995
52 Mohajer S, Tandon R. New bounds on the (n, k, d) storage systems with exact repair. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 2056–2060
53 Sasidharan B, Prakash N, Krishnan M N, et al. Outer bounds on the storage-repair bandwidth trade-off of exact-repair regenerating codes. Int J Inf Coding Theory, 2016, 3: 255–298
54 Elyasi M, Mohajer S. Determinant coding: a novel framework for exact-repair regenerating codes. IEEE Trans Inf Theory, 2016, 62: 6683–6697
55 Elyasi M, Mohajer S. Exact-repair trade-off for $(n, k = d - 1, d)$ regenerating codes. In: Proceedings of the 55th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2017. 934–941
56 Prakash N, Krishnan M N. The storage-repair-bandwidth trade-off of exact repair linear regenerating codes for the case $d = k = n - 1$. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 859–863
57 Elyasi M, Mohajer S, Tandon R. Linear exact repair rate region of $(k + 1, k, k)$ distributed storage systems: a new approach. In: Proceedings of IEEE International Symposium on Information Theory (ISIT), Hong Kong, 2015. 2061–2065
58 Hu Y, Xu Y, Wang X, et al. Cooperative recovery of distributed storage systems from multiple losses with network coding. IEEE J Sel Areas Commun, 2010, 28: 268–276
59 Kermarrec A M, Scouarnec N L, Straub G. Repairing multiple failures with coordinated and adaptive regenerating codes. In: Proceedings of International Symposium on Networking Coding, Beijing, 2011
60 Shum K W, Hu Y. Cooperative regenerating codes. IEEE Trans Inf Theory, 2013, 59: 7229–7258
61 Wang A Y, Zhang Z F. Exact cooperative regenerating codes with minimum-repair-bandwidth for distributed storage. In: Proceedings of IEEE INFOCOM, Turin, 2013. 400–404
62 Shum K W, Chen J. Cooperative repair of multiple node failures in distributed storage systems. Int J Inf Coding Theory, 2016, 3: 299–323
63 Scouarnec N L. Exact scalar minimum storage coordinated regenerating codes. In: Proceedings of IEEE International Symposium on Information Theory, Cambridge, 2012. 1197–1201
64 Ye M, Barg A. Optimal MDS codes for cooperative repair. 2018. ArXiv:1801.09665
65 Liu S Q, Oggier F E. On storage codes allowing partially collaborative repairs. In: Proceedings of IEEE International Symposium on Information Theory Proceedings, Honolulu, 2014. 2440–2444
66 Liu S Q, Oggier F E. Two storage code constructions allowing partially collaborative repairs. In: Proceedings of International Symposium on Information Theory and its Applications, Melbourne, 2014. 378–382
67 Koyluoglu O O, Rawat A S, Vishwanath S. Secure cooperative regenerating codes for distributed storage systems. IEEE Trans Inf Theory, 2014, 60: 5228–5244
68 Huang K, Parampalli U, Xian M. Security concerns in minimum storage cooperative regenerating codes. IEEE Trans Inf Theory, 2016, 62: 6218–6232
69 Rashmi K V, Shah N B, Ramchandran K. A piggybacking design framework for read-and download-efficient distributed storage codes. IEEE Trans Inf Theory, 2017, 63: 5802–5820
70 Guruswami V, Rawat A S. MDS code constructions with small sub-packetization and near-optimal repair bandwidth. In: Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms, Barcelona, 2017. 2109–2122
71 Kralevska K, Gligoroski D, Øverby H. General sub-packetized access-optimal regenerating codes. IEEE Commun Lett, 2016, 20: 1281–1284
72 Rawat A S, Tamo I, Guruswami V, et al. ϵ-MSR codes with small sub-packetization. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 2043–2047
73 Rouayheb S Y E, Ramchandran K. Fractional repetition codes for repair in distributed storage systems. In: Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing, Allerton, 2010
74 Pawar S, Noorshams N, Rouayheb S Y E, et al. DRESS codes for the storage cloud: simple randomized constructions. In: Proceedings of IEEE International Symposium on Information Theory Proceedings, St. Petersburg, 2011.

2338–2342
75 Silberstein N, Etzion T. Optimal fractional repetition codes based on graphs and designs. IEEE Trans Inf Theory, 2015, 61: 4164–4180
76 Olmez O, Ramamoorthy A. Fractional repetition codes with flexible repair from combinatorial designs. IEEE Trans Inf Theory, 2016, 62: 1565–1591
77 Koo J C, Gill J T. Scalable constructions of fractional repetition codes in distributed storage systems. In: Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2011. 1366–1373
78 Ernvall T. The existence of fractional repetition codes. 2012. ArXiv:1201.3547
79 Pawar S, Rouayheb S Y E, Ramchandran K. Securing dynamic distributed storage systems against eavesdropping and adversarial attacks. IEEE Trans Inf Theory, 2011, 57: 6734–6753
80 Rashmi K V, Shah N B, Ramchandran K, et al. Regenerating codes for errors and erasures in distributed storage. In: Proceeding of IEEE International Symposium on Information Theory Proceedings, Cambridge, 2012. 1202–1206
81 Rashmi K V, Shah N B, Ramchandran K, et al. Information-theoretically secure erasure codes for distributed storage. IEEE Trans Inf Theory, 2018, 64: 1621–1646
82 Tandon R, Amuru S D, Clancy T C, et al. Toward optimal secure distributed storage systems with exact repair. IEEE Trans Inf Theory, 2016, 62: 3477–3492
83 Rawat A S, Koyluoglu O O, Silberstein N, et al. Optimal locally repairable and secure codes for distributed storage systems. IEEE Trans Inf Theory, 2014, 60: 212–236
84 Goparaju S, Rouayheb S Y E, Calderbank R, et al. Data secrecy in distributed storage systems under exact repair. In: Proceedings of International Symposium on Network Coding, Calgary, 2013
85 Huang K, Parampalli U, Xian M. On secrecy capacity of minimum storage regenerating codes. IEEE Trans Inf Theory, 2017, 63: 1510–1524
86 Rawat A S. Secrecy capacity of minimum storage regenerating codes. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 1406–1410
87 Kadhe S, Sprintson A. Security for minimum storage regenerating codes and locally repairable codes. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 1028–1032
88 Ye F, Shum K W, Yeung R W. The rate region for secure distributed storage systems. IEEE Trans Inf Theory, 2017, 63: 7038–7051
89 Shao S, Liu T, Tian C, et al. On the tradeoff region of secure exact-repair regenerating codes. IEEE Trans Inf Theory, 2017, 63: 7253–7266
90 Han J, Lastras-Montano L A. Reliable memories with subline accesses. In: Proceedings of IEEE International Symposium on Information Theory, Nice, 2007. 2531–2535
91 Huang C, Chen M H, Li J. Pyramid codes: flexible schemes to trade space for access efficiency in reliable data storage systems. In: Proceedings of the 6th IEEE International Symposium on Network Computing and Applications, Cambridge, 2007. 79–86
92 Oggier F, Datta A. Self-repairing homomorphic codes for distributed storage systems. In: Proceedings of IEEE INFOCOM, Shanghai, 2011. 1215–1223
93 Gopalan P, Huang C, Simitci H, et al. On the locality of codeword symbols. IEEE Trans Inf Theory, 2012, 58: 6925–6934
94 Papailiopoulos D, Dimakis A. Locally repairable codes. In: Proceedings of IEEE International Symposium on Information Theory, Cambridge, 2012. 2771–2775
95 Forbes M, Yekhanin S. On the locality of codeword symbols in non-linear codes. Discrete Math, 2014, 324: 78–84
96 Prakash N, Kamath G M, Lalitha V, et al. Optimal linear codes with a local-error-correction property. In: Proceedings of IEEE International Symposium on Information Theory Proceedings, Cambridge, 2012. 2776–2780
97 Silberstein N, Rawat A S, Koyluoglu O O, et al. Optimal locally repairable codes via rank-metric codes. In: Proceedings of IEEE International Symposium on Information Theory, Istanbul, 2013. 1819–1823
98 Prakash N, Lalitha V, Kumar P V. Codes with locality for two erasures. In: Proceedings of IEEE International Symposium on Information Theory, Honolulu, 2014. 1962–1966
99 Wang A, Zhang Z. An integer programming-based bound for locally repairable codes. IEEE Trans Inf Theory, 2015, 61: 5280–5294
100 Zhang J, Wang X, Ge G N. Some improvements on locally repairable codes. 2015. ArXiv:1506.04822
101 Mehrabi M, Ardakani M. On minimum distance of locally repairable codes. In: Proceedings of the 15th Canadian Workshop on Information Theory, Quebec, 2017
102 Tamo I, Barg A. A family of optimal locally recoverable codes. IEEE Trans Inf Theory, 2014, 60: 4661–4676
103 Ernvall T, Westerback T, Hollanti C. Constructions of optimal and almost optimal locally repairable codes. In: Proceedings of the 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace Electronic Systems, Aalborg, 2014
104 Liu J, Mesnager S, Chen L. New constructions of optimal locally recoverable codes via good polynomials. IEEE Trans Inf Theory, 2018, 64: 889–899
105 Kolosov O, Barg A, Tamo I, et al. Optimal LRC codes for all lenghts $n \leqslant q$. 2018. ArXiv:1802.00157
106 Jin L F, Ma L M, Xing C P. Construction of optimal locally repairable codes via automorphism groups of rational function fields. 2017. ArXiv:1710.09638
107 Balaji S B, Kumar P V. On partial maximally-recoverable and maximally-recoverable codes. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 1881–1885

108 Cadambe V R, Mazumdar A. Bounds on the size of locally recoverable codes. IEEE Trans Inf Theory, 2015, 61: 5787–5794
109 Huang P, Yaakobi E, Uchikawa H, et al. Binary linear locally repairable codes. IEEE Trans Inf Theory, 2016, 62: 6268–6283
110 Balaji S B, Kumar P V. Bounds on the rate and minimum distance of codes with availability. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 3155–3159
111 Wei V K. Generalized Hamming weights for linear codes. IEEE Trans Inf Theory, 1991, 37: 1412–1418
112 Wang A Y, Zhang Z F, Lin D D. Bounds and constructions for linear locally repairable codes over binary fields. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 2033–2037
113 Ma J X, Ge G N. Optimal binary linear locally repairable codes with disjoint repair groups. 2017. ArXiv:1711.07138
114 Agarwal A, Barg A, Hu S, et al. Combinatorial alphabet-dependent bounds for locally recoverable codes. IEEE Trans Inf Theory, 2018, 64: 3481–3492
115 Tamo I, Barg A, Goparaju S, et al. Cyclic LRC codes, binary LRC codes, and upper bounds on the distance of cyclic codes. 2016. ArXiv:1603.08878
116 Goparaju S, Calderbank R. Binary cyclic codes that are locally repairable. In: Proceedings of IEEE International Symposium on Information Theory, Honolulu, 2014. 676–680
117 Zeh A, Yaakobi E. Optimal linear and cyclic locally repairable codes over small fields. In: Proceedings of IEEE Information Theory Workshop, Jerusalem, 2015
118 Tamo I, Barg A, Frolov A. Bounds on the parameters of locally recoverable codes. IEEE Trans Inf Theory, 2016, 62: 3070–3083
119 Barg A, Tamo I, Vladut S. Locally recoverable codes on algebraic curves. IEEE Trans Inf Theory, 2017, 63: 4928–4939
120 Li X D, Ma L M, Xing C P. Construction of asymptotically good locally repairable codes via automorphism groups of function fields. 2017. ArXiv:1711.07703
121 Nam M Y, Song H Y. Binary locally repairable codes with minimum distance at least six based on partial t-spreads. IEEE Commun Lett, 2017, 21: 1683–1686
122 Silberstein N, Zeh A. Optimal binary locally repairable codes via anticodes. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 1247–1251
123 Hao J, Xia S T, Chen B. Some results on optimal locally repairable codes. In: Proceedings of IEEE International Symposium on Information Theory, Barcelona, 2016. 440–444
124 Shahabinejad M, Khabbazian M, Ardakani M. A class of binary locally repairable codes. IEEE Trans Commun, 2016, 64: 3182–3193
125 Hao J, Xia S T, Chen B. On optimal ternary locally repairable codes. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 171–175
126 Hao J, Xia S. Bounds and constructions of locally repairable codes: parity-check matrix approach. 2016. ArXiv:1601.05595
127 Li X D, Ma L M, Xing C P. Optimal locally repairable codes via elliptic curves. 2017. ArXiv:1712.03744
128 Kim C, No J S. New constructions of binary and ternary locally repairable codes using cyclic codes. IEEE Commun Lett, 2018, 22: 228–231
129 Luo Y, Xing C P, Yuan C. Optimal locally repairable codes of distance 3 and 4 via cyclic codes. 2018. ArXiv:1801.03623
130 Krishnan M N, Puranik B, Kumar P V, et al. Exploiting locality for improved decoding of binary cyclic codes. IEEE Trans Commun, 2018, 66: 2346–2358
131 Vardy A, Be'ery Y. Maximum-likelihood soft decision decoding of BCH codes. IEEE Trans Inf Theory, 1994, 40: 546–554
132 Huang P, Yaakobi E, Uchikawa H, et al. Cyclic linear binary locally repairable codes. In: Proceedings of IEEE Information Theory Workshop, Jerusalem, 2015
133 Chen M H, Huang C, Li J. On the maximally recoverable property for multi-protection group codes. In: Proceedings of IEEE International Symposium on Information Theory, Nice, 2007. 486–490
134 Blaum M, Hafner J L, Hetzler S. Partial-MDS codes and their application to RAID type of architectures. IEEE Trans Inf Theory, 2013, 59: 4510–4519
135 Calis G, Koyluoglu O O. A general construction for PMDS codes. IEEE Commun Lett, 2017, 21: 452–455
136 Gabrys R, Yaakobi E, Blaum M, et al. Constructions of partial MDS codes over small fields. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017
137 Gopalan P, Huang C, Jenkins B, et al. Explicit maximally recoverable codes with locality. IEEE Trans Inf Theory, 2014, 60: 5245–5256
138 Hu G, Yekhanin S. New constructions of SD and MR codes over small finite fields. In: Proceedings of IEEE International Symposium on Information Theory, Barcelona, 2016. 1591–1595
139 Chen J Y, Shum K W, Yu Q, et al. Sector-disk codes and partial MDS codes with up to three global parities. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 1876–1880
140 Blaum M. Construction of PMDS and SD codes extending RAID 5. 2013. ArXiv:1305.0032
141 Blaum M, Plank J S, Schwartz M, et al. Construction of partial MDS and sector-disk codes with two global parity symbols. IEEE Trans Inf Theory, 2016, 62: 2673–2681
142 Lalitha V, Lokam S V. Weight enumerators and higher support weights of maximally recoverable codes. In: Proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2015.

835–842
143 Kadhe S, Calderbank R. Rate optimal binary linear locally repairable codes with small availability. 2017. ArXiv:1701.02456
144 Wang A Y, Zhang Z F, Liu M L. Achieving arbitrary locality and availability in binary codes. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 1866–1870
145 Wang A Y, Zhang Z F. Repair locality with multiple erasure tolerance. IEEE Trans Inf Theory, 2014, 60: 6979–6987
146 Song W T, Yuen C. Locally repairable codes with functional repair and multiple erasure tolerance. 2015. ArXiv:1507.02796
147 Balaji S B, Kini G R, Kumar P V. A tight rate bound and a matching construction for locally recoverable codes with sequential recovery from any number of multiple erasures. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 1778–1782
148 Balaji S B, Kini G R, Kumar P V. A bound on rate of codes with locality with sequential recovery from multiple erasures. 2016. ArXiv:1611.08561
149 Song W T, Cai K, Yuen C, et al. On sequential locally repairable codes. IEEE Trans Inf Theory, 2018, 64: 3513–3527
150 Balaji S B, Kini G R, Kumar P V. A rate-optimal construction of codes with sequential recovery with low block length. In: Proceedings of National Conference on Communications, Hyderabad, 2018
151 Rawat A S, Mazumdar A, Vishwanath S. Cooperative local repair in distributed storage. 2014. ArXiv:1409.3900
152 Exoo G, Jajcay R. Dynamic cage survey. The Electronic Journal Combinatorics, 2013. http://pdfs.semanticscholar.org/43b8/2016a2ef8f394f2cb1954439248198d2c274.pdf
153 Song W, Dau S H, Yuen C, et al. Optimal locally repairable linear codes. IEEE J Sel Areas Commun, 2014, 32: 1019–1036
154 Chen B, Xia S T, Hao J, et al. Constructions of optimal cyclic (r, δ) locally repairable codes. IEEE Trans Inf Theory, 2018, 64: 2499–2511
155 Hao J, Xia S T, Chen B. On the linear codes with (r, δ)-locality for distributed storage. In: Proceedings of IEEE International Conference on Communications, Paris, 2017
156 Sasidharan B, Agarwal G K, Kumar P V. Codes with hierarchical locality. In: Proceedings of International Symposium on Information Theory (ISIT), Hong Kong, 2015. 1257–1261
157 Ballentine S, Barg A. Codes on curves with hierarchical locality. In: Proceedings of IEEE International Symposium on Information Theory (accepted), Hong Kong, 2018
158 Duminuco A, Biersack E. Hierarchical codes: how to make erasure codes attractive for peer-to-peer storage systems. In: Proceedings of the 8th International Conference on Peer-to-Peer Computing, Aachen, 2008. 89–98
159 Kamath G M, Prakash N, Lalitha V, et al. Codes with local regeneration and erasure correction. IEEE Trans Inf Theory, 2014, 60: 4637–4660
160 Gligoroski D, Kralevska K, Jensen R E, et al. Repair duality with locally repairable and locally regenerating codes. 2017. ArXiv:1701.06664
161 Hollmann H D L. On the minimum storage overhead of distributed storage codes with a given repair locality. In: Proceedings of IEEE International Symposium on Information Theory, Honolulu, 2014. 1041–1045
162 Ahmad I, Wang C C. When locally repairable codes meet regenerating codes — what if some helpers are unavailable. In: Proceedings of IEEE International Symposium on Information Theory, Hong Kong, 2015. 849–853
163 Krishnan M N, Anantha N R, Kumar P V. Codes with combined locality and regeneration having optimal Rate, $d_{\min}$ and linear field size. 2018. ArXiv:1804.00564
164 Shanmugam K, Papailiopoulos D S, Dimakis A G, et al. A repair framework for scalar MDS codes. IEEE J Sel Areas Commun, 2014, 32: 998–1007
165 Guruswami V, Wootters M. Repairing Reed-Solomon codes. IEEE Trans Inf Theory, 2017, 63: 5684–5698
166 MacWilliams F J, Sloane N J A. The theory of error-correcting codes. Elsevier, 1977, 68: 185–186
167 Dau H, Milenkovic O. Optimal repair schemes for some families of full-length Reed-Solomon codes. In: Proceedings of IEEE International Symposium on Information Theory, Aachen, 2017. 346–350
168 Ye M, Barg A. Explicit constructions of MDS array codes and RS codes with optimal repair bandwidth. In: Proceedings of IEEE International Symposium on Information Theory, Barcelona, 2016. 1202–1206
169 Chowdhury A, Vardy A. Improved schemes for asymptotically optimal repair of MDS codes. In: Proceedings of the 55th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2017. 950–957
170 Tamo I, Ye M, Barg A. Optimal repair of reed-solomon codes: achieving the cut-set bound. In: Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science, Berkeley, 2017. 216–227
171 Dau S H, Duursma I M, Kiah H M, et al. Repairing Reed-Solomon codes with multiple erasures. 2016. ArXiv:1612.01361
172 Bartan B, Wootters M. Repairing multiple failures for scalar MDS codes. In: Proceedings of the 55th Annual Allerton Conference on Communication, Control, and Computing, Monticello, 2017. 1145–1152
173 Ye M, Barg A. Repairing Reed-Solomon codes: universally achieving the cut-set bound for any number of erasures. 2017. ArXiv:1710.07216
174 Luby M. Capacity bounds for distributed storage. 2016. ArXiv:1610.03541
175 Luby M, Padovani R, Richardson T J, et al. Liquid cloud storage. 2017. ArXiv:1705.07983
176 Huang C, Simitci H, Xu Y, et al. Erasure coding in windows azure storage. In: Proceedings of USENIX Annual Technical Conference, Boston, 2012. 15–26
177 Gantenbein D. A better way to store data. Microsoft research blog. https://www.microsoft.com/en-us/research/

blog/better-way-store-data/
178 CEPH. Locally repairable erasure code plugin. http://docs.ceph.com/docs/master/rados/operations/erasure-code-lrc/
179 Rashmi K V, Shah N B, Gu D, et al. A "hitchhiker's" guide to fast and efficient data reconstruction in erasure-coded data centers. In: Proceedings of ACM SIGCOMM Conference, Chicago, 2014. 331–342
180 Kralevska K, Gligoroski D, Jensen R E, et al. Hashtag erasure codes: from theory to practice. IEEE Trans Big Data, 2017. doi: 10.1109/TBDATA.2017.2749255
181 Krishnan M N, Prakash N, Lalitha V, et al. Evaluation of codes with inherent double replication for Hadoop. In: Proceedings of the 6th USENIX Workshop on Hot Topics in Storage and File Systems, Philadelphia, 2014
182 Rashmi K V, Nakkiran P, Wang J, et al. Having your cake and eating it too: jointly optimal erasure codes for I/O, storage, and network-bandwidth. In: Proceedings of the 13th USENIX Conference on File and Storage Technologies, Santa Clara, 2015. 81–94
183 Li J, Li B. Beehive: erasure codes for fixing multiple failures in distributed storage systems. IEEE Trans Parallel Distrib Syst, 2017, 28: 1257–1270
184 Pamies-Juarez L, Blagojevic F, Mateescu R, et al. Opening the chrysalis: on the real repair performance of MSR codes. In: Proceedings of the 14th USENIX Conference on File and Storage Technologies, Santa Clara, 2016. 81–94
185 Gad E E, Mateescu R, Blagojevic F, et al. Repair-optimal MDS array codes over GF(2). In: Proceedings of IEEE International Symposium on Information Theory, Istanbul, 2013. 887–891
186 Vajha M, Ramkumar V, Puranik B, et al. Clay codes: moulding MDS codes to yield an MSR code. In: Proceedings of the 16th USENIX Conference on File and Storage Technologies, Oakland, 2018. 139–154