ISIT'2013 1569719277

# Optimal Locally Repairable Codes via Rank-Metric Codes

Natalia Silberstein, Ankit Singh Rawat, O. Ozan Koyluoglu, and Sriram Vishwanath
Department of Electrical and Computer Engineering
The University of Texas at Austin
TX 78712 USA
Email: {natalys, ankitsr, ozan, sriram}@austin.utexas.edu.

***Abstract*—This paper presents a new explicit construction for locally repairable codes (LRCs) for distributed storage systems which possess all-symbol locality and the largest possible minimum distance, or equivalently, can tolerate the maximum number of node failures. This construction, based on maximum rank distance (MRD) Gabidulin codes, provides new optimal vector and scalar LRCs. In addition, the paper also discusses mechanisms by which codes obtained using this construction can be used to construct LRCs with efficient local repair of failed nodes by combination of LRCs with regenerating codes.**

## I. Introduction

In distributed storage systems (DSSs), it is desirable that data be reliably stored over a network of nodes in such a way that a user (*data collector*) can retrieve the stored data even if some nodes fail. To achieve such a resilience against node failures, DSSs introduce data redundancy based on different coding techniques. For example, erasure codes are widely used in such systems: When using an $(n, k)$ code, data to be stored is first divided into $k$ blocks; subsequently, these $k$ information blocks are encoded into $n$ blocks stored on $n$ distinct nodes in the system. In addition, when a single node fails, the system reconstructs the data stored on the failed node to keep the required level of redundancy. This process of data reconstruction for a failed node is called *node repair process* [1]. During a node repair process, the node which is added to the system to replace the failed node downloads data from a set of appropriate and accessible nodes.

There are two important goals that guide the design of codes for DSSs: reducing the *repair bandwidth*, i.e. the amount of data downloaded from system nodes during the node repair process, and achieving *locality*, i.e. reducing the number of nodes participating in the node repair process. These goals underpin the design of two families of codes for DSSs called *regenerating codes* (see [1]–[8] and references therein) and *locally repairable codes* (see [9]–[22]), respectively.

In this paper, we focus on the locally repairable codes (LRCs). Recently, these codes have drawn significant attention within the research community. Oggier et al. [11], [12] present coding schemes which facilitate local node repair. In [19], Gopalan et al. establish an upper bound on the minimum distance of scalar LRCs, which is analogous to the Singleton bound. The paper also shows that pyramid codes, presented in [9], achieve this bound with information symbol locality. Subsequently, the work by Prakash et al. extends the bound to a more general definition of scalar LRCs [15]. (Han and Lastras-Montano [10] provide a similar upper bound which is coincident with the one in [15] for small minimum distances, and also present codes that attain this bound in the context of reliable memories.) In [14], Papailiopoulos and Dimakis generalize the bound in [19] to vector codes, and present locally repairable coding schemes which exhibits the MDS property at the cost of a small amount of additional storage per node.

The main contributions of this paper are as follows. First, in Section II, we generalize the definition of *scalar* locally repairable codes, presented in [15] to *vector* locally repairable codes. For such codes, every node storing $\alpha$ symbols from a given finite field $\mathbb{F}$, can be locally repaired by using data stored on at most $r$ other nodes from a group of nodes of size $r + \delta - 1 < n$, which we call a *local group*, where $n$ is the number of system nodes, and $r$ and $\delta$ are the given locality parameters. Subsequently, in Section III, we derive an upper bound on the minimum distance $d_{\min}$ of the vector codes that satisfy a given locality constraint, which establishes a trade-off between node failure resilience (i.e., $d_{\min}$) and per node storage $\alpha$. [1] The bound presented in [14] can be considered as a special case of our bound with $\delta = 2$. Further, we present an explicit construction for LRCs which attain this bound on minimum distance. This construction is based on maximum rank distance (MRD) Gabidulin codes, which are a rank-metric analog of Reed-Solomon codes. The *scalar* and *vector* LRCs that are obtained by this construction are the first explicit optimal locally repairable codes with $(r + \delta - 1) \nmid n$. Finally, in Section IV, we discuss how the scalar and vector codes obtained by this construction can be used for constructions of repair bandwidth efficient LRCs. We conclude the paper with Section V.

## II. Background

### A. System Parameters

Let $\mathbf{f}$ be a file of size $\mathcal{M}$ over a finite field $\mathbb{F}$ that needs to be stored on a DSS with $n$ nodes. Each node is assumed to store $\alpha$ symbols over $\mathbb{F}$.

[1] In a parallel and independent work, [20], Kamath et al. also provide upper bounds on minimum distance together with constructions and existence results for vector LRCs.

### B. Vector Codes

A linear $[n, \mathcal{M}, d_{\min}, \alpha]_q$ vector code $C$ over $\mathbb{F}_q$ of length $n$ is defined as a linear subspace of $\mathbb{F}_q^{\alpha n}$ of dimension $\mathcal{M}$. The symbols $\mathbf{c}_i$, $1 \leq i \leq n$, of a codeword $\mathbf{c} \in C$ belong to $\mathbb{F}_q^{\alpha}$. The minimum distance $d_{\min}$ of $C$ is defined as the minimum Hamming distance over $\mathbb{F}_q^{\alpha}$. An alternative definition for the minimum distance of an $[n, \mathcal{M}, d_{\min}, \alpha]_q$ vector code in terms of entropy is as follows [14]:

**Definition 1.** *The minimum distance $d_{\min}$ of a vector code $C$ of dimension $\mathcal{M}$ is defined as*

$$d_{\min} = n - \max_{\mathcal{A}: H(\mathbf{c}_{\mathcal{A}}) < \mathcal{M}} |\mathcal{A}|, \tag{1}$$

*where $\mathcal{A} = \{i_1, \ldots, i_{|\mathcal{A}|}\} \subseteq \{1, \ldots, n\}$, $\mathbf{c}_{\mathcal{A}} = (\mathbf{c}_{i_1}, \ldots, \mathbf{c}_{i_{|\mathcal{A}|}})$, and $H$ denotes entropy.*

Vector codes are also known as *array* codes. An $[n, \mathcal{M}, d_{\min}, \alpha]_q$ array code is called *MDS array code* if $d_{\min} = n - \mathcal{M} + 1$. Constructions for MDS array codes can be found e.g. in [27], [28].

### C. Locally Repairable Codes

In this subsection, we generalize the definition of *scalar* LRCs, presented in [15] to *vector* LRCs.

**Definition 2.** *We say that an $[n, \mathcal{M}, d_{\min}, \alpha]_q$ vector code $C$ has $(r, \delta)$* all-symbol locality *if for each symbol $\mathbf{c}_i \in \mathbb{F}_q^{\alpha}$, $1 \leq i \leq n$, of a codeword $\mathbf{c} = (\mathbf{c}_1, \ldots, \mathbf{c}_n) \in C$, there exists a set of indices $\Gamma(i)$ such that*

- $i \in \Gamma(i)$
- $|\Gamma(i)| \leq r + \delta - 1$
- $d_{\min}(C|_{\Gamma(i)}) \geq \delta$, *for a punctured code $C|_{\Gamma(i)}$.*

*Note that the last two properties imply that each element $j \in \Gamma(i)$ can be written as a function of a set of at most $r$ elements in $\Gamma(i)$ (not containing $j$) and that $H(\Gamma(i)) \leq r\alpha$.*

*Codes that satisfy these properties are called $(r, \delta, \alpha)$* locally repairable codes *(LRCs).*

Note, that the definition of LRCs presented in this paper generalizes the notion of LRCs given in [14], which is restricted to $\delta = 2$.

In order to store a file $\mathbf{f}$ on a DSS using an LRC, $\mathbf{f}$ is first encoded to a codeword of an LRC. Each symbol (from $\mathbb{F}_q^{\alpha}$) of the codeword is then stored on a different node. Note that a node $i$ in a locally repairable DSS can be repaired by downloading data from at most $r$ nodes in $\Gamma(i) \backslash \{i\}$.

**Remark 3.** *$(r, \delta, \alpha = 1)$ LRCs are named as $(r, \delta)$ scalar LRCs.*

In [15], Prakash et al. present the following upper bound on the minimum distance of an $(r, \delta)$ scalar LRC:

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

It was established in [15] that a family of pyramid codes, presented in [9], attains this bound and has *information locality*, i.e. only information symbols satisfy the locality constraint. However, an explicit construction of optimal scalar LRCs with *all-symbol locality* is known only for the case $n = \left\lceil \frac{M}{r} \right\rceil (r + \delta - 1)$ [10], [15]. The *existence* of optimal scalar codes with all-symbol locality is shown for the case when $(r + \delta - 1) | n$ and field size $|\mathbb{F}| > \mathcal{M} n^{\mathcal{M}}$ [15]. In this paper, we provide an explicit construction of optimal scalar LRCs with all-symbol locality relaxing the restriction of $(r + \delta - 1) | n$.

The following upper bound on the minimum distance of $(r, \delta = 2, \alpha)$ LRCs and a construction of codes that attain this bound was presented in [14]:

$$d_{\min} \leq n - \left\lceil \frac{\mathcal{M}}{\alpha} \right\rceil - \left\lceil \frac{\mathcal{M}}{r\alpha} \right\rceil + 2 \tag{3}$$

In the sequel, we generalize this bound for any $\delta \geq 2$ and present $(r, \delta, \alpha)$ LRCs that attain this bound.

### D. Maximum Rank Distance (MRD) Codes

For the construction presented in this paper we propose a precoding of a file $\mathbf{f}$ with a maximum rank distance code [23], [24].

Let $\mathbb{F}_{q^m}$ be an extension field of $\mathbb{F}_q$. Since $\mathbb{F}_{q^m}$ can be also considered as an $m$-dimensional vector space over $\mathbb{F}_q$, any element $\gamma \in \mathbb{F}_{q^m}$ can be represented as the vector $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_m) \in \mathbb{F}_q^m$, such that $\gamma = \sum_{i=1}^{m} b_i \gamma_i$, for a fixed basis $\{b_1, \ldots, b_m\}$ of the field extension. Similarly, any vector $\mathbf{v} = (v_1, \ldots, v_N) \in \mathbb{F}_{q^m}^N$ can be represented by an $m \times N$ matrix $\mathbf{V} = [v_{i,j}]$ over $\mathbb{F}_q$, where each entry $v_i$ of $\mathbf{v}$ is expanded as a column vector $(v_{i,1}, \ldots, v_{i,m})^T$.

**Definition 4.** *The* rank *of a vector $\mathbf{v} \in \mathbb{F}_{q^m}^N$, denoted by $\mathrm{rank}(\mathbf{v})$, is defined as the rank of the $m \times N$ matrix $\mathbf{V}$ over $\mathbb{F}_q$. Similarly, for two vectors $\mathbf{v}, \mathbf{u} \in \mathbb{F}_{q^m}^N$, the* rank distance *is defined by $d_R(\mathbf{v}, \mathbf{u}) = \mathrm{rank}(\mathbf{V} - \mathbf{U})$.*

An $[N, K, D]_{q^m}$ rank-metric code $\mathcal{C} \subseteq \mathbb{F}_{q^m}^N$ is a linear block code over $\mathbb{F}_{q^m}$ of length $N$, dimension $K$ and minimum rank distance $D$. A rank-metric code that attains the Singleton bound $D \leq N - K + 1$ for the rank metric is called a *maximum rank distance* (MRD) code. For $m \geq N$, a construction of MRD codes was presented by Gabidulin [23]. Similar to Reed-Solomon codes, Gabidulin codes can be obtained by evaluation of polynomials, however, for Gabidulin codes, a special family of polynomials, called *linearized polynomials*, is used:

**Definition 5.** *A linearized polynomial $f(x)$ over $\mathbb{F}_{q^m}$ of $q$-degree $t$ has the form $f(x) = \sum_{i=0}^{t} a_i x^{q^i}$, where $a_i \in \mathbb{F}_{q^m}$, and $a_t \neq 0$.*

Note, that evaluation of a linearized polynomial is an $\mathbb{F}_q$-linear transformation from $\mathbb{F}_{q^m}$ to itself, i.e., for any $a, b \in \mathbb{F}_q$ and $\gamma_1, \gamma_2 \in \mathbb{F}_{q^m}$, we have $f(a\gamma_1 + b\gamma_2) = af(\gamma_1) + bf(\gamma_2)$ [25].

A codeword of an $[N, K, D = N - K + 1]_{q^m}$ Gabidulin code $\mathcal{C}^{\mathrm{Gab}}$, $m \geq N$, is defined as $\mathbf{c} = (f(g_1), f(g_2), \ldots, f(g_N)) \in \mathbb{F}_{q^m}^N$, where $f(x)$ is a linearized polynomial over $\mathbb{F}_{q^m}$ of $q$-degree at most $K - 1$ with $K$ message symbols as its coefficients, and $g_1, \ldots, g_N \in \mathbb{F}_{q^m}$ are linearly independent over $\mathbb{F}_q$ [23].

An MRD code $\mathcal{C}^{\rm Gab}$ with minimum distance $D$ can correct any $D-1=N-K$ erasures, which we will call *rank erasures*. An algorithm for erasures correction of Gabidulin codes can be found e.g. in [26].

### E. Regenerating Codes

Regenerating codes are a family of codes for DSSs that allow for efficient repair of failed nodes. When using such codes, we assume that a data collector can reconstruct the original file by downloading the data stored on any set of $k$ out of $n$ nodes. When a node fails, its content can be reconstructed by downloading $\beta \leq \alpha$ symbols from each node in a set of $d$, $k \leq d \leq n-1$, surviving nodes. Given a file size $\mathcal{M}$, a trade-off between storage per node $\alpha$ and *repair bandwidth* $\gamma \triangleq d\beta$ can be established [1]. Two classes of codes that achieve two extreme points of this trade-off are known as *minimum storage regenerating (MSR)* codes and *minimum bandwidth regenerating (MBR)* codes. The parameters $(\alpha, \gamma)$ for MSR and MBR codes are given by $\left(\frac{\mathcal{M}}{k}, \frac{\mathcal{M}d}{k(d-k+1)}\right)$ and $\left(\frac{2\mathcal{M}d}{2kd-k^2+k}, \frac{2\mathcal{M}d}{2kd-k^2+k}\right)$, respectively [1].

## III. Optimal Locally Repairable Codes

In this section, we first derive an upper bound on the minimum distance of $(r, \delta, \alpha)$ LRCs. Next, we propose a general code construction which attains the derived bound on $d_{\min}$. Our approach is to apply a two-stage encoding, where we use Gabidulin codes along with MDS array codes. This construction can be viewed as a generalization of the construction proposed in [17].

### A. Upper Bound on $d_{\min}$ for an $(r, \delta, \alpha)$ LRC

We state a generic upper bound on the minimum distance $d_{\min}$ of an $(r, \delta, \alpha)$ LRC $C$ of length $n$ and dimension $\mathcal{M}$. The bound generalizes the $d_{\min}$-bound given in [14] for LRCs with a single local parity ($\delta = 2$) to LRCs with multiple local parities ($\delta \geq 2$).

**Theorem 6.** *Let $C$ be an $(r, \delta, \alpha)$ LRC of length $n$ and dimension $\mathcal{M}$. Then*

$$d_{\min}(C) \leq n - \left\lceil \frac{\mathcal{M}}{\alpha} \right\rceil + 1 - \left( \left\lceil \frac{\mathcal{M}}{r\alpha} \right\rceil - 1 \right) (\delta - 1). \quad (4)$$

*Proof.* (Sketch) We follow the proof technique of [14], [19]. In particular, the proof involves construction of a set of nodes $\mathcal{A}$ for a locally repairable DSS such that total entropy of the symbols stored on $\mathcal{A}$ is less than $\mathcal{M}$ and

$$|\mathcal{A}| \geq \left\lceil \frac{\mathcal{M}}{\alpha} \right\rceil - 1 + \left( \left\lceil \frac{\mathcal{M}}{r\alpha} \right\rceil - 1 \right) (\delta - 1). \quad (5)$$

Theorem 6 then follows from Definition 1 and (5). See [18] for the detailed proof. □

Remarkably, the theorem above establishes a trade-off between node failure resilience ($d_{\min}$) and per node storage ($\alpha$), where $\alpha$ can be increased to obtain higher $d_{\min}$. This is of particular interest for designing codes having both locality and high resilience to node failures.

**Remark 7.** *For the special case of $\delta = 2$, this bound matches with the bound (3) presented in [14]. For the case of $\alpha = 1$, the bound reduces to $d_{\min} \leq n - \mathcal{M} + 1 + (\lceil \mathcal{M}/r \rceil - 1)(\delta - 1)$, which is coincident with the bound (2) presented in [15].*

### B. Construction of $d_{\min}$-Optimal Vector LRCs

In this subsection we present a construction of an $(r, \delta, \alpha)$ LRC with length $n$ and dimension $\mathcal{M}$, which attains the bound given in Theorem 6.

**Construction I.** Consider a file $\mathbf{f}$ over $\mathbb{F} = \mathbb{F}_{q^m}$ of size $\mathcal{M} \geq r\alpha$, where $m$ will be defined in the sequel. We encode the file in two steps before storing it on a DSS. First, the file is encoded using a Gabidulin code. The codeword of the Gabidulin code is then partitioned into local groups and each local group is then encoded using an MDS array code over $\mathbb{F}_q$.

In particular, let $\mathcal{M}, n, r, \delta, \alpha$ be the positive integers such that $r + \delta - 1 < n$ and $\mathcal{M} \geq r\alpha$. We denote by $g = \left\lceil \frac{n}{r+\delta-1} \right\rceil$ the number of local groups in the system. We consider the following two cases:

**1)** $(r + \delta - 1) | n$**:** Let $N = \frac{nr\alpha}{r+\delta-1}$, $m \geq N$, and let $\mathcal{C}^{\rm Gab}$ be an $[N, \mathcal{M}, D = N - \mathcal{M} + 1]_{q^m}$ Gabidulin code. First, we encode $\mathbf{f} \in \mathbb{F}_{q^m}^{\mathcal{M}}$ to a codeword $\mathbf{c} \in \mathcal{C}^{\rm Gab}$ and partition $\mathbf{c}$ into $g = \frac{N}{r\alpha}$ disjoint groups, each of size $r\alpha$, and each group is stored on a different set of $r$ nodes, $\alpha$ symbols per node. In other words, the output of the first encoding step generates the encoded data stored on $rg$ nodes, each one containing $\alpha$ symbols of the Gabidulin codeword. Second, we generate $\delta - 1$ parity nodes per group by applying an $[(r + \delta - 1), r, \delta, \alpha]_q$ MDS array code on each local group of $r$ nodes, treating these $r$ nodes as input data blocks (of length $\alpha$) for the MDS array code. At the end of the second round of encoding, we have $n = g(r + \delta - 1) = \frac{N}{\alpha} + \frac{N}{r\alpha}(\delta - 1)$ nodes, each storing $\alpha$ symbols over $\mathbb{F}_{q^m}$, partitioned into $g$ local groups, each of size $r + \delta - 1$.

**2)** $n (\text{mod } r + \delta - 1) > (\delta - 1)$**:** Let $\beta_0$, $1 \leq \beta_0 \leq r - 1$ be an integer, such that $n = \lfloor \frac{n}{r+\delta-1} \rfloor (r + \delta - 1) + \beta_0 + \delta - 1 = (g-1)(r+\delta-1) + \beta_0 + \delta - 1$. Let $N = (g-1)r\alpha + \beta_0\alpha$, $m \geq N$, and let $\mathcal{C}^{\rm Gab}$ be an $[N, \mathcal{M}, D = N - \mathcal{M} + 1]_{q^m}$ Gabidulin code. First, we encode $\mathbf{f}$ to a codeword $\mathbf{c} \in \mathcal{C}^{\rm Gab}$ and partition $\mathbf{c}$ into $g-1$ disjoint groups of size $r\alpha$ and one additional group of size $\beta_0\alpha$. The first $g-1$ groups are stored on $(g-1)r$ nodes, and the last group is stored on $\beta_0$ nodes, each one containing $\alpha$ symbols of the Gabidulin codeword. Second, we generate $\delta - 1$ parity nodes per group by applying an $[(r+\delta-1), r, \delta, \alpha]_q$ MDS array code on each of the first $g - 1$ local groups of $r$ nodes, and by applying a $[(\beta_0 + \delta - 1), \beta_0, \delta, \alpha]_q$ MDS array code on the last local group. At the end of the second round of encoding, we have $n = (g-1)(r+\delta-1) + (\beta_0 + \delta - 1) = \frac{N}{\alpha} + \left\lceil \frac{N}{r\alpha} \right\rceil (\delta - 1)$ nodes, each storing $\alpha$ symbols over $\mathbb{F}_{q^m}$, partitioned into $g$ local groups, $g-1$ of which of size $r+\delta-1$ and one group of size $\beta_0 + \delta - 1$.

We denote the obtained code by $C^{\rm loc}$.

**Remark 8.** *Note, that since an MDS array code from Construction I is defined over $\mathbb{F}_q$, any symbol of any node of $C^{\rm loc}$ can be written as $\sum_{j=1}^{r\alpha} a_j c_{i_j} = \sum_{j=1}^{r\alpha} a_j f(g_{i_j}) =$*

$f(\sum_{j=1}^{r\alpha} a_j g_{i_j})$, *where* $a_j \in \mathbb{F}_q$, $c_{i_j} \in \mathbb{F}_{q^m}$ *are* $r\alpha$ *symbols of the same group of the codeword* $\mathbf{c} \in C^{\text{loc}}$, *and* $g_{i_j} \in \mathbb{F}_{q^m}$ *are linearly independent (over* $\mathbb{F}_q$*) evaluation points. Hence,* any $s \leq r\alpha$ *symbols inside a group of* $C^{\text{loc}}$ *are evaluations of* $f(x)$ *at* $s$ *linearly independent over* $\mathbb{F}_q$ *points. (If there is a group with* $\beta_0 < r$ *elements we have the same result substituting* $r$ *with* $\beta_0$*). Thus,* any $\delta - 1 + i$ node erasures *in a group correspond to* $i\alpha$ rank erasures. *Moreover, if we take any* $r\alpha$ *symbols of* $C^{\text{loc}}$ *from every group (and* $\beta_0\alpha$ *symbols from the smallest group, if it exists), we obtain a Gabidulin codeword, for a corresponding choice of evaluation points for a Gabidulin code, which encodes the given data* $\mathbf{f}$.

Next, we provide the conditions for parameters of the code $C^{\text{loc}}$ obtained from Construction I to be a $d_{\min}$-optimal $(r, \delta, \alpha)$ LRC.

**Theorem 9.** *Let* $C^{\text{loc}}$ *be an* $(r, \delta, \alpha)$ *LRC obtained by Construction I. Then,*

- *If* $(r+\delta-1)|n$, *then* $C^{\text{loc}}$ *over* $\mathbb{F} = \mathbb{F}_{q^m}$, *for* $m \geq \frac{nr\alpha}{r+\delta-1}$ *and* $q \geq (r+\delta-1)$, *attains the bound (4).*
- *If* $n(mod\ r+\delta-1) - (\delta-1) \geq \lceil \frac{\mathcal{M}}{\alpha} \rceil (mod\ r) > 0$, *then* $C^{\text{loc}}$ *over* $\mathbb{F} = \mathbb{F}_{q^m}$, *for* $m \geq \alpha \left( n - (\delta-1) \left( \left\lfloor \frac{n}{r+\delta-1} \right\rfloor + 1 \right) \right)$ *and* $q \geq (r+\delta-1)$, *attains the bound (4).*

*Proof.* The proof is based on Remark 8 and the observation that any $n - \lceil \frac{\mathcal{M}}{\alpha} \rceil - \left( \lceil \frac{\mathcal{M}}{r\alpha} \rceil - 1 \right)(\delta - 1)$ node erasures correspond to at most $D-1$ rank erasures which can be corrected by the Gabidulin code $\mathcal{C}^{\text{Gab}}$. See [29] for a detailed derivation of the result. □

**Remark 10.** *For the case* $\alpha = 1$ *Construction I provides* $d_{\min}$*-optimal scalar LRCs. Note that this is the first explicit construction of optimal scalar locally repairable codes with* $(r+\delta-1) \nmid n$.

**Remark 11.** *The required field size* $|\mathbb{F}| = q^m$ *for the proposed construction should satisfy* $m \geq N$, *for any choice of* $q$. *So we can assume that* $|\mathbb{F}| = q^N$, *for* $N$ *given in Construction I. Note that we can reduce the field size to* $|\mathbb{F}| = q^{N/\alpha}$ *by stacking [20] of* $\alpha$ *independent optimal scalar LRCs, obtained by Construction I.*

We illustrate the construction of $C^{\text{loc}}$ in the following examples. First, we consider the scalar case.

**Example 12.** *Consider the following system parameters:*

$$(\mathcal{M}, n, r, \delta, \alpha) = (9, 14, 4, 2, 1).$$

*Since* $n = \left\lfloor \frac{14}{4+2-1} \right\rfloor \cdot (4+2-1) + (3+2-1)$, *let* $N = \left\lfloor \frac{14}{4+2-1} \right\rfloor \cdot 4 + 3 = 11$. *First,* $\mathcal{M} = 9$ *symbols over* $\mathbb{F} = \mathbb{F}_{5^{11}}$ *are encoded into a codeword* $\mathbf{c}$ *of a* $[11, 9, 3]_{5^{11}}$ *Gabidulin code* $\mathcal{C}^{\text{Gab}}$. *This codeword is partitioned into three groups, two of size* 4 *and one of size* 3, *as follows:* $\mathbf{c} = (a_1, a_2, a_3, a_4 | b_1, b_2, b_3, b_4 | c_1, c_2, c_3)$. *Then, by applying a* $[5, 4, 2]$ *MDS code in the first two groups and a* $[4, 3, 2]$ *MDS code in the last group we add one parity to each group. The*

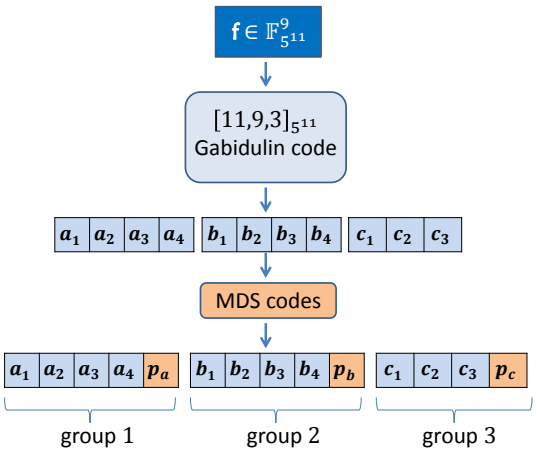


Fig. 1: Illustration of the construction of a scalar $(r = 4, \delta = 2, \alpha = 1)$ LRC for $n = 14, \mathcal{M} = 9$ and $d_{\min} = 4$.

*symbols of* $\mathbf{c}$ *with three new parities* $p_a, p_b, p_c$ *are stored on 14 nodes as shown in Fig. 1. By Theorem 6,* $d_{\min}(C^{\text{loc}}) \leq 4$. *By Remark 8, any* 3 *node erasures correspond to at most* 2 *rank erasures and then can be corrected by* $\mathcal{C}^{\text{Gab}}$, *hence* $d_{\min}(C^{\text{loc}}) = 4$. *In addition, when a single node fails, it can be repaired by using the data stored on all the other nodes from the same group.*

Next, we illustrate Construction I for a vector LRC.

**Example 13.** *We consider a DSS with the following parameters:*

$$(\mathcal{M}, n, r, \delta, \alpha) = (28, 15, 3, 3, 4).$$

*By (4) we have* $d_{\min}(C^{\text{loc}}) \leq 5$. *Let* $N = \frac{15 \cdot 3 \cdot 4}{3+3-1} = 36$ *and* $(a_1, \ldots, a_{12}, b_1, \ldots, b_{12}, c_1, \ldots, c_{12})$ *be a codeword of a* $[36, 28, 9]_{q^{36}}$ *code* $\mathcal{C}^{Gab}$, *which is obtained by encoding* $\mathcal{M} = 28$ *symbols over* $\mathbb{F} = \mathbb{F}_{q^{36}}$ *of the original file. The Gabidulin codeword is then partitioned into three groups* $(a_1, \ldots, a_{12})$, $(b_1, \ldots, b_{12})$, *and* $(c_1, \ldots, c_{12})$. *Encoded symbols in each group are stored on three storage nodes as shown in Fig. 2. In the second stage of encoding, a* $[5, 3, 3, 4]_q$ *MDS array code over* $\mathbb{F}_q$ *is applied on each local group to obtain* $\delta - 1 = 2$ *parity nodes per local group. The coding scheme is illustrated in Fig. 2.*

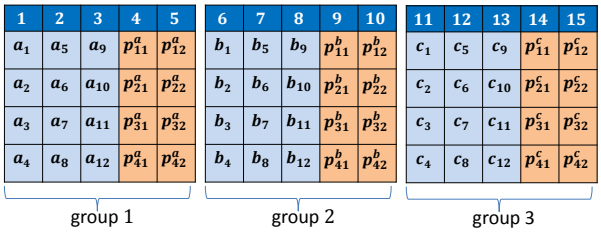


Fig. 2: Example of an $(r = 3, \delta = 3, \alpha = 4)$ LRC with $n = 15$ and $d_{\min} = 5$.

*By Remark 8, any* 4 *node failures correspond to at most* 8 *rank erasures in the corresponding codeword of* $\mathcal{C}^{Gab}$. *Since*

*the minimum rank distance of $\mathcal{C}^{\mathrm{Gab}}$ is 9, these node erasures can be corrected by $\mathcal{C}^{\mathrm{Gab}}$, and thus $d_{\min}(C^{\mathrm{loc}}) = 5$. In addition, when a single node fails, it can be repaired by using the data stored on any three other nodes from the same group.*

**Remark 14.** *The efficiency of the decoding of the codes obtained by Construction I depends on the efficiency of the decoding of the MDS codes and the Gabidulin codes.*

## IV. Repair Efficient LRCs

In this section, we discuss the hybrid codes which for a given locality parameters minimize the repair bandwidth. These codes are based on a combination of locally repairable codes with regenerating codes.

In a naïve repair process for a locally repairable code, a newcomer contacts $r$ nodes in its local group and downloads all the data stored on these nodes. Following the line of work of bandwidth efficient repair in DSSs given by [1], we allow a newcomer to contact any $d \geq r$ nodes in its local group and to download only $\beta \leq \alpha$ symbols stored on these nodes in order to repair the failed node. The motivation behind this is to lower the repair bandwidth for an LRC. The main idea here is to apply a regenerating code in each local group. (We note that, in a parallel and independent work, Kamath et al. [20] also proposed utilizing regenerating codes in the context of LRCs.)

In particular, by applying an $(r + \delta - 1, r, d, \alpha, \beta)$ MSR code in each local group instead of an MDS array code in the second step of Construction I we obtain a code, denoted by MSR-LRC, which has the maximal minimum distance (since an MSR code is also an MDS array code), the local minimum storage per node, and the minimized repair bandwidth. (The details of this construction can be found in [18].)

In addition, the optimal scalar codes obtained by Construction I can be used for construction of MBR-LRCs (codes with an MBR code in each local group) as it has been shown by Kamath et al. [20].

## V. Conclusion

We presented a novel construction for (scalar and vector) locally repairable codes. This construction is based on maximum rank distance codes. We derived an upper bound on minimum distance for vector LRCs and proved that our construction provides optimal codes for both scalar and vector cases. We also discussed how the codes obtained by this construction can be used to construct repair bandwidth efficient LRCs.


## Acknowledgment

The authors would like to thank Pascal Vontobel for his valuable comments on the early version of this paper.


## References


[1] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage system," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4539-4551, Sep. 2010.

[2] Y. Wu and A. G. Dimakis, "Reducing repair traffic for erasure coding-based storage via interference alignment," in *Proc. IEEE ISIT*, pp. 2276–2280, Jul. 2009.

[3] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, "Explicit codes minimizing repair bandwidth for distributed storage," in *Proc. IEEE ITW*, pp. 1–5, Jan. 2010.

[4] C. Suh and K. Ramchandran, "Exact-repair MDS codes for distributed storage using interference alignment," in *Proc. IEEE ISIT*, pp. 161–165, Jul. 2010.

[5] K. V. Rashmi, N. B. Shah, and P. V. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR point via a product-matrix construction," *IEEE Trans. Inf. Theory*, vol. 57, no. 57, pp. 5227-5239, Aug. 2011.

[6] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: MDS array codes with optimal rebuilding," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1597–1616, Mar. 2013.

[7] A. Datta and F. Oggier, "An overview of codes tailor-made for networked distributed data storage," *CoRR*, abs/1109.2317, Sep. 2011.

[8] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," in *Proc. of the IEEE*, pp. 476–489, Mar. 2011.

[9] C. Huang, M. Chen, and J. Li, "Pyramid code: flexible schemes to trade space for access efficiency in reliable data storage systems," in *Proc. 6th IEEE NCA*, pp. 79–86, Mar. 2007.

[10] J. Han and L. A. Lastras-Montano, "Reliable memories with subline accesses," in *Proc. IEEE ISIT*, pp. 2531–2535, Jun. 2007.

[11] F. E. Oggier and A. Datta, "Self-repairing codes for distributed storage - A projective geometric construction," in *Proc. IEEE ITW*, pp. 30–34, Oct. 2011.

[12] F. E. Oggier and A. Datta, "Self-repairing homomorphic codes for distributed storage systems," in *Proc. IEEE INFOCOM*, pp. 1215–1223, Apr. 2011.

[13] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, and S. Yekhanin,"Erasure coding in windows azure storage," *in Proc. USENIX Annual Technical Conference (ATC)*, Apr. 2012.

[14] D. S. Papailiopoulos and A. G. Dimakis, "Locally repairable codes," in *Proc. IEEE ISIT*, pp. 2771–2775, Jul. 2012.

[15] N. Prakash, G. M. Kamath, V. Lalitha, and P. V. Kumar, "Optimal linear codes with a local-error-correction property," in *Proc. IEEE ISIT*, pp. 2776–2780, Jul. 2012.

[16] A. S. Rawat and S. Vishwanath, "On locality in distributed storage systems," in *Proc. IEEE ITW*, pp. 497–501, Sep. 2012

[17] N. Silberstein, A. S. Rawat, and S. Vishwanath, "Error resilience in distributed storage via rank-metric codes," in *Proc. 50th Allerton Conference on Communication, Control, and Computing*, pp. 1150–1157, Oct. 2012.

[18] A. S. Rawat, O. O. Koyluoglu, N. Silberstein, and S. Vishwanath, "Optimal locally repairable and secure codes for distributed storage systems," *CoRR*, abs/1210.6954, Oct. 2012.

[19] P. Gopalan, C. Huang, H. Simitchi, and S. Yekhanin, "On the locality of codeword symbols," *IEEE Trans. Inf. Theory*, vol. 58, no. 11, pp. 6925-6934, Nov. 2012.

[20] G. M. Kamath, N. Prakash, V. Lalitha, and P. V. Kumar, "Codes with local regeneration," *CoRR*, abs/1211.1932, Nov. 2012.

[21] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A. G. Dimakis, R. Vadali, S. Chen, and D. Borthakur, "XORing elephants: Novel erasure codes for big data," *CoRR*, abs/1301.3791, Jan. 2013.

[22] H. D. L. Hollmann, "Storage codes - coding rate and repair locality," *CoRR*, abs/1301.4300, Jan. 2013.

[23] E. M. Gabidulin, "Theory of codes with maximum rank distance," *Problems of Information Transmission*, vol. 21, pp. 1-12, Jul. 1985.

[24] R. M. Roth, "Maximum-rank array codes and their application to crisscross error correction," *IEEE Trans. Inf. Theory*, vol. 37, pp. 328-336, Mar. 1991.

[25] F. J. MacWilliams and N. J. A. Sloane, *The theory of error-correcting codes*, North-Holland, 1978.

[26] E. M. Gabidulin and N. I. Pilipchuk, "Error and erasure correcting algorithms for rank codes," *Designs, Codes and Cryptography*, vol. 49, pp. 105–122, 2008.

[27] M. Blaum, J. Brady, J. Bruck, and J. Menon, "EVENODD: an efficient scheme for tolerating double disk failures in RAID architectures," *IEEE Trans. on Computers*, vol. 44, no. 2, pp. 192–202, Feb. 1995.

[28] Y. Cassuto and J. Bruck,"Cyclic low-density MDS array codes," in *Proc. IEEE ISIT*, pp. 2794–2798, Jul. 2006.

[29] N. Silberstein, A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Optimal locally repairable codes via rank-metric codes," *CoRR*, abs/1301.6331, Jan. 2013.