ORIGINAL RESEARCH

Extension-based constructions of locally repairable fractional repetition codes

Mehmet Dağlı[1]

Abstract

Fractional Repetition (FR) codes break the data into smaller fragments and distribute these fragments across multiple storage nodes. This paper focuses on extension-based construction methods for obtaining new FR codes from existing FR codes. Additionally, it formulates a particular case of extension codes as an exact cover problem, enabling an algorithmic construction of FR codes.

Keywords Distributed storage systems · Fractional repetition codes · Regenerating codes · The exact cover problem

Mathematics Subject Classification 05B05 · 94B25

1 Introduction

In today's data-driven landscape, the management and preservation of vast amounts of information have become critical. Large-scale data storage systems are used in a variety of domains, from social networking and video streaming sites to the ever-expanding realm of cloud storage solutions. In these contexts, ensuring the integrity and availability of stored data is paramount, especially when storage nodes are individually unreliable. The fundamental challenge of maintaining data resiliency and accessibility in the face of node failures is traditionally addressed by introducing redundancy through data replication and erasure coding [3]. Distributed storage systems are a key solution to this challenge. These systems effectively distribute data across multiple nodes or servers, which are often geographically dispersed. In addition to ensuring reliable data access, they have a unique property: after a regeneration process, the resulting system retains the essential characteristics of the original configuration. The

✉ Mehmet Dağlı
    mehmet.dagli@amasya.edu.tr

1   Department of Mathematics, Amasya University, 05100 Amasya, Turkey

🙋 Springer

regeneration process may require a balance between optimizing certain metrics, such as the speed of regeneration and the minimum number of nodes involved. The design of coding techniques tailored to meet these objectives has consequently become an area of research interest [3, 4, 12, 16, 18].

In the general framework, a Distributed Storage System (DSS) with parameters (n, k, d), or simply an (n, k, d)-DSS, consists of n storage nodes. Each of these nodes stores packets, or equivalently symbols, in such a way that allows the data collector to reconstruct the stored file by contacting any k nodes. This property is referred to as the Maximum Distance Separability (MDS) property of the system. In the event of a node failure, the system connects to any set of $d \geq k$ surviving nodes and downloads β packets from each of them. This collective operation results in a total repair bandwidth of $\gamma = d\beta$ packets. The parameters d, β, and γ are called the repair degree, normalized repair bandwidth, and total repair bandwidth, respectively.

In certain DSSs, replacement nodes are designed to connect only to predetermined subsets of nodes for repair. In these systems, the repair process is exact and uncoded, accomplished by simply downloading packets from the surviving nodes [9, 14, 17, 19, 24]. The encoding process begins with the use of an outer MDS code. The encoded symbols are then placed into storage nodes using an inner fractional repetition code to meet specific desired characteristics. This dual-layered approach allows the entire DSS to function without requiring any computation at the surviving nodes or the replacement nodes [1, 10, 11, 18, 19, 22]. Notably, the known constructions for exact repair processes typically operates at one of two distinct points: the Minimum Bandwidth Regenerating (MBR) point, minimizing the repair bandwidth γ; the Minimum Storage Regenerating (MSR) point, minimizing the storage per node. In parallel, several studies have investigated code constructions, called local codes, aimed at minimizing the number of nodes involved in the regeneration process [4, 12, 15, 21]. In these codes, the repair degree d is strictly less than k.

In this paper, we present novel constructions based on an extension method that adds new symbols to the storage nodes of existing FR codes. In particular, we obtain an infinite family of codes through the extension process associated with affine planes and projective planes. The resultant codes may achieve optimality without altering the minimum distances. Additionally, we introduce families of locally repairable codes originating from net FR codes. Furthermore, we outline the construction of extensions to certain FR codes that arised from partitioning the edges of complete bipartite graphs.

2 Fractional repetition codes

Let θ represent the number of encoded symbols generated by an outer (n, k)-MDS code, such as the Reed–Solomon code. Copies of these symbols are distributed across n nodes, with each symbol occurring ρ times and each node containing α symbols. In the event of a node failure, recovery is possible by downloading exactly β packets from a predetermined set of d surviving nodes, resulting in a total repair bandwidth of $d\beta$. In other words, $\alpha = d\beta$.
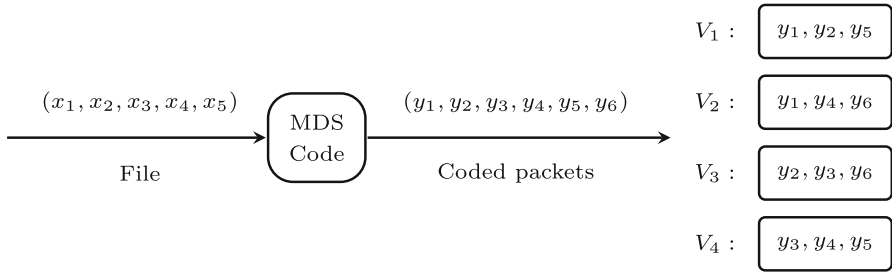
$V_1 :$ y_1, y_2, y_5

$V_2 :$ y_1, y_4, y_6

$(x_1, x_2, x_3, x_4, x_5)$ MDS Code $(y_1, y_2, y_3, y_4, y_5, y_6)$ $V_3 :$ y_2, y_3, y_6

File   Coded packets $V_4 :$ y_3, y_4, y_5

Fig. 1 A $(4, 2, 3)$-DSS

Example 1 Let $X = (x_1, x_2, x_3, x_4, x_5) \in \mathbb{F}_q^5$ be a file of 5 packets to be stored in a DSS. Consider the $(6, 5)$-MDS code that inputs the file X, and outputs the coded file $Y = (y_1, y_2, y_3, y_4, y_5, y_6)$ of 6 packets, where $y_i = x_i$ for $1 \leq i \leq 5$, and $y_6 = \sum_{i=1}^{5} x_i$. The coded packets are then distributed among the storage nodes $V_1 = \{y_1, y_2, y_5\}$, $V_2 = \{y_1, y_4, y_6\}$, $V_3 = \{y_2, y_3, y_6\}$ and $V_4 = \{y_3, y_4, y_5\}$, as shown in Fig. 1. In this configuration, every node contains a subset of 3 packets from the set $\{y_1, y_2, \ldots, y_6\}$, and each y_i is replicated twice in the storage system. When a node fails, the system can connect to the remaining 3 nodes to download 1 packet from each to repair the failed node. As any distinct pair of nodes shares exactly one symbol in common, any two storage nodes contain at least 5 distinct symbols together. Consequently, the entire file can be recovered from any two nodes by means of the MDS property.

The nodes of a DSS can be represented by a binary matrix $N = [n_{ij}]$ of size $\theta \times n$ defined as follows:

$$n_{ij} = \begin{cases} 1; & \text{if } y_i \in V_j, \\ 0; & \text{otherwise.} \end{cases}$$

For example, the nodes of the DSS in Example 1 can be represented by the following matrix.

$$N = \begin{array}{c} \\ \\ \end{array} \begin{array}{cccc} V_1 & V_2 & V_3 & V_4 \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} & \begin{array}{l} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{array} \end{array}$$

Definition 1 Let $\Omega = \{1, 2, \ldots, \theta\}$ denote the index set of coded packets. An FR code $\mathcal{C}$, with repetition degree ρ, for an (n, k, d)-DSS is a collection of n subsets $V_1, V_2, \ldots, V_n$ of Ω that satisfy the following conditions:

1. Each subset V_i has a cardinality of d for all $1 \leq i \leq n$.
2. Each element of Ω belongs to exactly ρ sets within the collection.

Constructing FR codes based on combinatorial structures, including regular graphs and combinatorial designs such as affine planes and projective planes, were presented in [9, 13, 14, 17, 19, 21, 23].

In a DSS with parameters (n, k, d), the stored file should be constructed from an arbitrary set of k nodes. The file size depends on the parameter k and is defined as follows.

Definition 2 The supported file size of an FR code $\mathcal{C}$ for an (n, k, d)-DSS, denoted as $M_k(\mathcal{C})$, corresponds to the minimum number of distinct symbols stored within any set of k nodes. In other words,

$$M_k(\mathcal{C}) = \min_{I \subset [n], |I| = k} |\cup_{i \in I} V_i| .$$

Rouayheb and Ramchandran [17] derived two upper bounds for the file size that an FR code can support. The first bound is given by the formula

$$M_k(\mathcal{C}) \leq \left\lfloor \frac{nd}{\rho} \left(1 - \frac{\binom{n-\rho}{k}}{\binom{n}{k}} \right) \right\rfloor .$$

The second bound is sharper than the first bound, and it is given as $M_k(\mathcal{C}) \leq \phi(k)$, where $\phi(k)$ is recursively defined by

$$\phi(1) = d,$$

$$\phi(k + 1) = \phi(k) + d - \left\lceil \frac{\rho \phi(k) - kd}{n - k} \right\rceil .$$

The minimum distance, denoted as $d_{\min}$, is another metric considered in DSSs. It represents the size of the smallest subset of nodes whose failure ensures that the file stored in the system cannot be reconstructed from the remaining nodes. The following lemma establishes a Singleton-like bound on the minimum distance of an FR code.

Lemma 1 [14, Lemma 1] *The minimum distance of an FR code $\mathcal{C}$ for an (n, k, d)-DSS is bounded from above by*

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

An FR code $\mathcal{C}$ designed for an (n, k, d)-DSS is called *locally recoverable* if $d < k$. Locally repairable codes suffer a penalty on the maximum possible minimum distance [4].

Lemma 2 [15, Theorem 1] *Consider an (n, k, d)-DSS. If the file size is M, then*

$$d_{\min}(\mathcal{C}) \leq n - \left\lceil \frac{M}{\alpha} \right\rceil - \left\lceil \frac{M}{d\alpha} \right\rceil + 2.$$

Assume that each node is part of a local structure that forms a (local) FR code. We have the following bound for locally recoverable codes.

Lemma 3 [21, Theorem 3] *Let* $\mathcal{C} = (\Omega, \mathcal{V})$ *be a locally recoverable FR code designed for an* $(n, k, d = \alpha)$-*DSS, where each node is part of a local FR code* $\mathcal{C}' = (\Omega', \mathcal{V}')$ *with* $|\Omega'| = \theta'$, $|\mathcal{V}'| = n'$ *per-node storage* $\alpha' = \alpha$ *and repetition degree* $\rho' = \rho$. *Then the minimum distance of* $\mathcal{C}$ *is bounded from above by*

$$d_{\min}(\mathcal{C}) \leq n - \left\lceil \frac{(\rho' - 1)\theta' \left\lfloor \frac{M_k(\mathcal{C})-1}{\theta'} \right\rfloor + M_k(\mathcal{C})}{\alpha} \right\rceil + 1. \tag{1}$$

3 Extension-based code constructions

In this section, we will construct new FR codes based on existing FR codes. Our strategy is to keep the number of storage nodes in the DSS constant, while increasing the number of symbols within each node. To this end, we begin with a motivating example. Consider the following FR code:

$$\mathcal{C} = \{123, 456, 789, 147, 258, 369, 159, 267, 348\}.$$

The file size of $\mathcal{C}$ is $M_3(\mathcal{C}) = 6$. For example, users connecting to the second, fourth, and eighth nodes can download six distinct symbols. In other words, only six symbols are retrievable from the nodes containing symbols 456, 147, and 267. However, the upper bound of the file size is $\phi(3) = 7$. Now, we extend the number of symbols in each node by adding a certain symbol from the set $\{a, b, c\}$ as follows:

$$\mathcal{C}' = \{123a, 456a, 789a, 147b, 258b, 369b, 159c, 267c, 348c\}.$$

The file size of $\mathcal{C}'$ is $M_3(\mathcal{C}') = 9$, which meets the upper bound. Recall that an FR code for an (n, k, d)-DSS attains the Singleton-like bound if $k = \left\lceil \frac{M_k(\mathcal{C})}{\alpha} \right\rceil$. Both $\mathcal{C}$ and $\mathcal{C}'$ require the failure of at least seven nodes to guarantee that the file stored in the DSS cannot be reconstructed from the surviving nodes. Consequently, $\mathcal{C}$ does not meet the Singleton-like bound while $\mathcal{C}'$ achieves optimality with respect to this bound.

In summary, we have constructed an optimal FR code for a $(9, 3, 4)$-DSS using an existing code without altering the minimum distance. To generalize this idea, we can now define code extensions.

Definition 3 Consider an (n, k, d)-DSS, whose inner fractional repetition code $\mathcal{C} = (\Omega_1, \mathcal{V})$ is characterized by the parameters $(n, \theta_1, \alpha_1, \rho_1)$. Let Ω_2 be a finite set of symbols. Extend each node of the DSS by adding α_2 symbols from Ω_2, resulting in the creation of the following nodes:

$$\mathcal{W} = \{(V|W) : V \subset \Omega_1, W \subset \Omega_2, |V| = \alpha_1, |W| = \alpha_2\}.$$

If $\mathcal{C}' = (\Omega_1 \cup \Omega_2, \mathcal{W}')$ forms a fractional repetition code for $\mathcal{W}' \subseteq \mathcal{W}$, then we say that $\mathcal{C}'$ is an extension of $\mathcal{C}$.

It is worth highlighting the operation in defining the nodes in $\mathcal{C}'$. This operation adjoins symbols to each node of a given FR code $\mathcal{C}$ to form a new FR code $\mathcal{C}'$. In particular it takes each node V of size α_1 from $\mathcal{C}$, and adjoins to it a set W of size α_2, resulting in a node $(V|W)$ of size $\alpha_1 + \alpha_2$. This set W comprises specific symbols from Ω_2, rather than encompassing all possible α_2-element subsets of Ω_2.

Projective planes and affine planes serve as important tools for constructing code extensions. These structures belong to the category of balanced incomplete block designs. Let us now revisit their definitions.

Definition 4 A $(\theta, \alpha, \lambda)$ Balanced Incomplete Block Design (BIBD) is a pair $(\Omega, \mathcal{V})$, where Ω is a set with θ elements, called *points*, and $\mathcal{V}$ is a collection of subsets of Ω, called *blocks*. These blocks satisfy the following conditions: each block has precisely α points, and every 2-subset of Ω is contained in exactly λ blocks.

Let $|\mathcal{V}| = n$. Then it is easy to verify that

$$n\alpha = \theta\rho \quad \text{and} \quad \rho(\alpha - 1) = \lambda(\theta - 1)$$

where ρ represents the replication number of each point in the design. We can use the notation $(\theta, \rho, \alpha, \lambda)$-BIBD in order to capture the parameter ρ.

Remark 1 A BIBD is in fact an FR code, with the additional property that every pair of distinct points is contained in exactly λ blocks [13].

A BIBD with parameters $(n^2 + n + 1, n + 1, 1)$ where $n \geq 2$, is referred to as a projective plane of order n. Correspondingly, an $(n^2, n, 1)$-BIBD is called an affine plane of order n. Projective and affine planes of order n exist for prime power values of n. We refer to [20] for further details.

Definition 5 Let $\mathcal{C} = (\Omega, \mathcal{V})$ be an FR code with $\mathcal{V} = \{V_1, V_2, \ldots, V_n\}$. A subset $P \subset \mathcal{V}$ is called a parallel class if $V_i \cap V_j = \emptyset$ for all $V_i, V_j \in P$ with $i \neq j$, and $\cup_{V_i \in P} V_i = \Omega$.

Theorem 1 *An extension of an FR code for a $(p^2, p, p + 1)$-DSS exists if p is an odd prime. Moreover, the code extension is optimal with respect to the Singleton-like bound for $k = 3$.*

Proof Let p be an odd prime. Then there exists an affine plane of order p, which is a $(p^2, p, p + 1)$-BIBD. This affine plane has $p(p + 1)$ lines that fall into $p + 1$ parallel classes, with each class consisting of p lines. A projective plane can be formed by adjoining a set of $p + 1$ points in a one-to-one correspondence with parallel classes. In this case, each new point is incident to all lines of the corresponding parallel class. Observe that any p distinct parallel classes of the affine plane form an FR code $\mathcal{C}$ with parameters $(n = p^2, \theta = p^2, \alpha = p, \rho = p)$. By adjoining p new symbols in one-to-one correspondence with these p parallel classes, we obtain an extension $\mathcal{C}'$ of $\mathcal{C}$ with parameters $(n = p^2, \theta = p^2 + p, \alpha = p + 1, \rho = p)$.

Next, we show that $\mathcal{C}'$ is optimal for $k = 3$. In the projective plane, any two lines intersect in exactly one point. Since the new nodes are part of the projective plane of

order p, the union of any $k = 3$ nodes will have at least $3(p+1) - \binom{3}{2} = 3p$ symbols. As $3 = \lceil \frac{3p}{p+1} \rceil$, $\mathcal{C}'$ is optimal with respect to the Singleton-like bound for $k = 3$. This completes the proof. $\square$

Theorem 2 *Let q be a prime power. Consider the FR code $\mathcal{C}$ obtained from q distinct parallel classes of the affine plane. Then the file size of the extension $\mathcal{C}'$ of $\mathcal{C}$ is given by $M_k(\mathcal{C}') = q^2 + q - \binom{q}{2}$ for $k = q$.*

Proof It can be deduced from Lemma 9 in [14] that the DSS achieves the minimum file size by selecting storage nodes from different parallel classes. These storage nodes introduce q additional new symbols through the extension process. By applying the inclusion–exclusion principle, it can be seen that at least $q^2 + q - \binom{q}{2}$ symbols are covered by any q storage nodes. Consequently, we have demonstrated a set of nodes that attains this bound. $\square$

Next, we present an extension family of FR codes where the new code has a larger minimum distance than the existing code. Before proceeding, let us state a result known as Corrádi's bound [2], which establishes bounds on the size of the union of certain subsets.

Lemma 4 *Let $V_1, V_2, \ldots, V_n$ be α-element sets and let X be their union. If $|V_i \cap V_j| \le k$ for all $i \ne j$, then*

$$|X| \ge \frac{\alpha^2 n}{\alpha + (n-1)k}.$$

Theorem 3 *Let q be a prime power. Consider an FR code $\mathcal{C} = (\Omega, \mathcal{V})$ obtained from a projective plane of order q. Suppose there exists an FR code $\mathcal{C}'$ where extra symbols are adjoined to the nodes of $\mathcal{C}$ in a one-to-one correspondence with the elements of Ω such that any pair of distinct symbols appears together in at most 2 nodes in the resulting FR code. Then,*
$$d_{\min}(\mathcal{C}') \ge d_{\min}(\mathcal{C}) + 1.$$

Proof The Corrádi's bound is optimal for projective planes [6]. Therefore,

$$M_{q^2}(\mathcal{C}) = \left\lceil \frac{(q+1)^2 q^2}{q+1+(q^2-1)1} \right\rceil = q^2 + q.$$

Given that each symbol is present in $q + 1$ nodes, and any distinct pair of nodes intersect in exactly one symbol, the minimum distance $d_{\min}(\mathcal{C})$ is equal to $2q + 1$. Now assume that there exists an extension $\mathcal{C}'$ with nodes $(N_i \mid i)$, where $i \notin N_i$ and for every $j \in \Omega$, the count of nodes containing both i and j is at most 2. The failure of certain $2q + 1$ nodes guarantees that 2 symbols from the projective plane cannot be recovered. However, according to our assumption, there will be at least one copy of these symbols that will survive in the new FR code obtained by the extension technique. Therefore, $d_{\min}(\mathcal{C}') \ge d_{\min}(\mathcal{C}) + 1$. $\square$

In the rest of this section, we will focus on locally recoverable codes. In particular, we will investigate an infinite family of locally recoverable FR codes obtained by the extension technique. But let us recall the definition of resolvable FR codes first [13].

Definition 6 Let $\mathcal{C} = (\Omega, \mathcal{V})$ be an FR code. A resolution is a partition of $\mathcal{V}$ into r parallel classes. If a resolution exists, then the code is called resolvable.

Before stating our next theorem, let us present a motivating example. Consider the FR code $\mathcal{C}_1 = \{12, 34, 13, 24\}$ and its isomorphic copy $\mathcal{C}_2 = \{56, 78, 57, 68\}$. Adjoin symbols from x_1, x_2 and y_1, y_2 to the non-parallel classes of $\mathcal{C}_1$ and $\mathcal{C}_2$ as follows:

$$\mathcal{C} = \left\{ \begin{array}{l} 12x_1, \ 34x_1, \ 13x_2, \ 24x_2, \ 56x_1, \ 78x_1, \ 57x_2, \ 68x_2 \\ 12y_1, \ 34y_1, \ 13y_2, \ 24y_2, \ 56y_1, \ 78y_1, \ 57y_2, \ 68y_2 \end{array} \right\}. \tag{2}$$

This forms a locally recoverable FR code for an $(16, 5, 3)$-DSS. Each node is part of a local FR code with parameters $(n = 4, \theta = 6, \alpha = 3, \rho = 2)$ where the local FR codes are obtained from isomorphic copies of the extension of $\mathcal{C}_1$ and $\mathcal{C}_2$ by adding a symbol to each node in a given parallel class.

We are now ready to generalize the idea illustrated in this example.

Theorem 4 Consider a resolvable FR code $\mathcal{C}_1 = (\Omega_1, \mathcal{V}_1)$ with parameters $(n = a^2, \theta = a^2, \alpha = a, \rho = a)$ such that any pair of non-parallel nodes intersect in exactly one symbol. Let $\mathcal{C}_2 = (\Omega_2, \mathcal{V}_2)$ be an isomorphic copy of $\mathcal{C}_1$ with $\Omega_1 \cap \Omega_2 = \emptyset$. Also, let $x_1, \ldots, x_a$ and $y_1, \ldots, y_a$ be symbols not in $\Omega_1 \cup \Omega_2$. Denote the j-th node in the i-th parallel classes of $\mathcal{C}_1$ and $\mathcal{C}_2$ by $T_{i,j}$ and $U_{i,j}$, respectively. Then the collection $\mathcal{C}$ of all the extended nodes

$$\left(T_{i,j} \mid x_i\right), \left(T_{i,j} \mid y_i\right), \left(U_{i,j} \mid x_i\right), \left(U_{i,j} \mid y_i\right)$$

forms an FR code with parameters $(n = 4a^2, \theta = 2(a^2 + a), \alpha = a + 1, \rho = 2a)$. Moreover, $\mathcal{C}$ is optimal with respect to the bound given in (1) for the file size $a^2 + a + 1$.

Proof Let $\mathcal{C}$ be defined as above. It is easy to observe from its construction that $\mathcal{C}$ is a locally recoverable FR code for an $(4a^2, a^2 + 1, a + 1)$-DSS. In this setting, each node is part of a local FR code with parameters $(n' = a^2, \theta' = a^2 + a, \alpha' = a + 1, \rho' = a)$. Consequently, we have the following bound on the minimum distance:

$$d_{\min}(\mathcal{C}) \leq 4a^2 - \left\lceil \frac{(a - 1)(a^2 + a) \left\lfloor \frac{a^2 + a}{a^2 + a} \right\rfloor + a^2 + a + 1}{a + 1} \right\rceil + 1.$$

Thus, $d_{\min}(\mathcal{C}) \leq 3a^2$. Now choose $k < 3a^2$ nodes and suppose that the file is not recoverable. This implies that a minimum of $a^2 + a$ symbols are inaccessible within the DSS. Without loss of generality, assume that all symbols in $\Omega_1 \cup \{x_1, \ldots, x_a\}$ are not recoverable in the DSS. Hence, all the nodes $\left(T_{i,j} \mid x_i\right)$, $\left(T_{i,j} \mid y_i\right)$, and $\left(U_{i,j} \mid x_i\right)$ should fail. As a result, we are left with only $a^2 + a$ symbols, and in this situation, recovering the file in the DSS is not possible. Since all the local codes are isomorphic

Fig. 2 The complete bipartite graph $K_{4,2}$
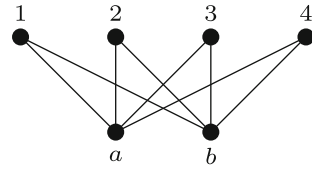
to each other, the above calculation remains valid when x_i is replaced by y_i and Ω_1 is replaced by Ω_2. Now consider the case where $a^2 + a$ symbols are chosen from $\Omega_1 \cup \Omega_2$. This implies that we need to select at least a of the symbols from Ω_1 and Ω_2. To minimize the number of node failures required to ensure that a symbols are not recoverable, such a symbols should be contained in a single node. Based on this observation, eliminating the symbols would necessitate a minimum of $a(a-1)+1$ node failures in each of the four categories of extended nodes. However, this contradicts our assumption that $k < 3a^2$. A similar calculation for the case in which $a^2 + a$ symbols are chosen from $\Omega_1 \cup \Omega_2$ and $\Omega_3 = \{x_1, \ldots, x_a, y_1, \ldots, y_a\}$ will lead to the conclusion that $d_{\min}(\mathcal{C}) = 3a^2$. □

Example 2 Consider the FR code $\mathcal{C}$ constructed in (2). Let us compare our code with the bound given in (1). Here, we have $n = 16$, $\rho' = 2$, $\theta' = 6$, and $M_5(\mathcal{C}) = 7$. It can easily be seen that $d_{\min} \leq 12$. Hence, the failure of at least 12 nodes guarantees that the file stored in the DSS cannot be reconstructed from the surviving nodes.

4 FR codes from complete bipartite graphs

In this section, we introduce a family of FR codes that can be obtained by edge partitioning of the complete bipartite graph $K_{c,d}$ into copies of the complete bipartite graph $K_{a,b}$. We begin our discussion with an example.

Example 3 Consider the complete bipartite graph $K_{4,2}$ whose vertices are partitioned into the subsets $V = \{1, 2, 3, 4\}$ and $W = \{a, b\}$. We can partition $K_{4,2}$ into four copies of $K_{2,1}$. Each copy of $K_{2,1}$ can be represented by the vertices involved in the partition. For example, $\{12a, 34a, 14b, 23b\}$ represents one of such partitions. Note that this partition served as storage nodes in Example 1.

A partition of $K_{c,d}$ into the copies of $K_{a,b}$ may or may not exist. The following theorem by Hoffman and Liatti [5] gives necessary and sufficient conditions for the partition problem on complete bipartite graphs.

Theorem 5 *Let a, b, c, and d be positive integers. Let $g = (a, b)$, the greatest common divisor of a and b; let e, f be integers satisfying $ae - bf = g$, and let $h = ae + bf$. For each integer x, let: $\alpha(x) = \lceil \frac{xf}{a} \rceil$, $\beta(x) = \lfloor \frac{xe}{b} \rfloor$, and $\gamma(x) = \frac{x}{ab}$. Then the edges of the complete bipartite graph $K_{c,d}$ can be partitioned into copies of the complete bipartite graph $K_{a,b}$ if and only if the following conditions are true:*

1. $ab \mid cd$,
2. $g \mid c$ and $\alpha(c) \leq \beta(c)$,

3. $g \mid d$ and $\alpha(d) \leq \beta(d)$,
4. $c\alpha(d) + d\alpha(c) \leq h\gamma(cd) \leq c\beta(d) + d\beta(c)$.

The task of partitioning the edges of the complete bipartite graph $K_{c,d}$ into copies of the complete bipartite graph $K_{a,b}$ can be formalized as an instance of the exact cover problem. Formally, given a set X and a collection S of subsets of X, the exact cover problem aims to find a subset S^* of S such that every element of X is included in exactly one subset of S^*. The exact cover problem can effectively be addressed using Knuth's Algorithm X which employs a straightforward recursive, nondeterministic, depth-first, backtracking approach to find all possible exact covers for a given collection of sets [8]. The problem is formulated as a binary matrix in Algorithm X. Then the objective is to identify a subset of the rows in such a way that the number 1 appears exactly once in each column.

Next, we will demonstrate how this partitioning can provide an infinite family of FR codes where we can apply enumeration algorithms. Let $\Gamma(V, W)$ be the complete bipartite graph. Let $i \in V$, $j \in W$ and X be the set of all pairs (i, j) such that (i, j) is an edge in the complete bipartite graph. Let K be the set of all s-element subsets of V. Let $S = \{(v|w) : v \in K, w \in W\}$. We will define a matrix M whose rows are indexed by the elements of S and columns are indexed by the elements of X as follows:

$$M_{(v|w),(i,j)} = \begin{cases} 1; & \text{if } i \in v \text{ and } w = j, \\ 0; & \text{otherwise.} \end{cases}$$

Let's apply the Algorithm X to the matrix M to construct FR codes.

Example 4 Consider the complete bipartite graph $K_{4,2}$ given in Fig. 2. Suppose we want to partition the edges of $K_{4,2}$ into the copies of $K_{2,1}$. Let us index the columns of M with the pairs $(1, a), (2, a), (3, a), (4, a), (1, b), (2, b), (3, b), (4, b)$ and the rows of M with the blocks $(12 \mid a), (13 \mid a), (14 \mid a), (23 \mid a), (24 \mid a), (34 \mid a), (12 \mid b), (13 \mid b), (14 \mid b), (23 \mid b), (24 \mid b), (34 \mid b)$. Both the rows and columns are indexed in the given order. Then the binary matrix M has the form $M = \begin{bmatrix} N & 0 \\ 0 & N \end{bmatrix}$ where

$$N = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

The matrix N has 3 exact covers. Specifically, the first and sixth, second and fifth, as well as the third and fourth rows all provide exact covers. Since M has a block diagonal structure with repeated blocks of N, it has $3^2 = 9$ exact covers. Therefore, by extending the exact covers of N or applying Algorithm X to the matrix M, we can find all 9 distinct solutions to the exact cover problem. These solutions correspond to

the following FR codes:

$$\{12a, 34a, 13b, 24b\}, \{12a, 34a, 14b, 23b\}, \{13a, 24a, 12b, 34b\},$$
$$\{13a, 24a, 14b, 23b\}, \{14a, 23a, 12b, 34b\}, \{14a, 23a, 13b, 24b\},$$
$$\{12a, 34a, 12b, 34b\}, \{13a, 24a, 13b, 24b\}, \{14a, 23a, 14b, 23b\}.$$

Note that the codes in the first two rows have the property $M_2(\mathcal{C}) = 5$. The remaining codes have $M_2(\mathcal{C}) = 4$.

We can perform similar calculations for larger complete bipartite graphs. For instance, we can obtain FR codes with parameters $(n = 9, \theta = 9, \alpha = 3, \rho = 3)$ by partitioning the edges of $K_{6,3}$ into copies of $K_{2,1}$. This case has a total of $15^3 = 3375$ solutions. Out of these solutions, 375 of them have the property $M_k(\mathcal{C}) = 5$, while the remaining 3000 solutions have $M_k(\mathcal{C}) = 6$ for $k = 3$. The table below provides a summary of the enumeration results for larger values of k.

k	$M_k(\mathcal{C}) = 5$	$M_k(\mathcal{C}) = 6$	$M_k(\mathcal{C}) = 7$	$M_k(\mathcal{C}) = 8$
4	0	2895	480	0
5	0	0	2895	480
6	0	0	375	3000

5 Conclusion

Addressing multiple node failures in DSSs is crucial for ensuring data integrity and availability. FR codes have emerged as powerful tools offering efficient solutions to this challenge, finding applications in diverse fields such as cloud storage, network coding, and data center architectures. Recognizing their importance in modern information systems, this paper presented extension-based constructions of new FR codes, and studied their characteristics. In this framework, we derived an infinite family of optimal FR codes based on the proposed method related to affine and projective planes. We then constructed locally recoverable FR codes that can achieve a Singleton-like upper bound with equality. In addition, we provided another construction of FR codes, realized as an extension of FR codes, by partitioning the edges of complete bipartite graphs. After presenting the necessary and sufficient conditions for the existence of such partitions, we applied Algorithm X to investigate two specific cases and obtained comprehensive enumeration results.

Declarations

Conflict of interest The author declares no conflicts of interest.

References

1. Benerjee, K.G., Gupta, M.K.: On non-uniform flower codes. Cryptogr. Commun. **12**, 613–643 (2020). https://doi.org/10.1007/s12095-020-00430-6
2. Corrádi, K.: Problem at Schweitzer competition. Mat. Lapok **20**, 159–162 (1969)
3. Dimakis, A.G., Godfrey, P.B., Wu, Y., Wainwright, M.J., Ramchandran, K.: Network coding for distributed storage systems. IEEE Trans. Inform. Theory **56**(9), 4539–4551 (2010). https://doi.org/10.1109/TIT.2010.2054295
4. Gopalan, P., Huang, C., Simitci, H., Yekhanin, S.: On the locality of codeword symbols. IEEE Trans. Inform. Theory **58**(11), 6925–6934 (2012). https://doi.org/10.1109/TIT.2012.2208937
5. Hoffman, D.G., Liatti, M.: Bipartite designs. J. Comb. Des. **3**(6), 449–454 (1995). https://doi.org/10.1002/jcd.3180030608
6. Jukna, S.: Extremal Combinatorics: With Applications in Computer Science. Springer, Berlin (2011)
7. Kamath, G.M., Prakash, N., Lalitha, V., Kumar, P.V.: Codes with local regeneration and erasure correction. IEEE Trans. Inform. Theory **60**(8), 4637–4660 (2014). https://doi.org/10.1109/TIT.2014.2329872
8. Knuth, D.E.: Dancing links. (2000) Preprint. https://doi.org/10.48550/arXiv.cs/0011047
9. Koo, J.C., Gill, J.T.: Scalable constructions of fractional repetition codes in distributed storage systems. In: Proc. 49th Ann. Allerton Conf. on Commun., Control, Comp., pp. 1366–1373. (2011) https://doi.org/10.1109/Allerton.2011.6120326
10. 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. **25**(12), 3232–3241 (2014). https://doi.org/10.1109/TPDS.2013.2297109
11. Lin, S.J., Chung, W.H., Han, Y.S., Al-Naffouri, T.Y.: A unified form of exact-MSR codes via product-matrix frameworks. IEEE Trans. Inform. Theory **61**(2), 873–886 (2015). https://doi.org/10.1109/TIT.2014.2378255
12. Oggier, F., Datta, A.: Self-repairing homomorphic codes for distributed storage systems. In: Proceedings IEEE INFOCOM, pp. 1215–1223 (2011). https://doi.org/10.1109/INFCOM.2011.5934901
13. Olmez, O., Ramamoorthy, A.: Repairable replication-based storage systems using resolvable designs. In: Proc. 50th Ann. Allerton Conf. on Commun., Control, Comp., pp. 1174–1181 (2012) https://doi.org/10.1109/Allerton.2012.6483351
14. Olmez, O., Ramamoorthy, A.: Fractional repetition codes with flexible repair from combinatorial designs. IEEE Trans. Inform. Theory **62**(4), 1565–1591 (2016). https://doi.org/10.1109/TIT.2016.2531720
15. Papailiopoulos, D.S., Dimakis, A.G.: Locally repairable codes. IEEE Trans. Inform. Theory **60**(10), 5843–5855 (2014). https://doi.org/10.1109/TIT.2014.2325570
16. 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. Inform. Theory **57**(8), 5227–5239 (2011). https://doi.org/10.1109/TIT.2011.2159049
17. Rouayheb, S.E., Ramchandran, K.: Fractional repetition codes for repair in distributed storage systems. In: Proc. 48th Ann. Allerton Conf. on Commun., Control, Comput., pp. 1510–1517 (2010) https://doi.org/10.1109/ALLERTON.2010.5707092
18. Shah, N.B., Rashmi, K.V., Kumar, P.V., Ramchandran, K.: Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff. IEEE Trans. Inform. Theory **58**(3), 1837–1852 (2012). https://doi.org/10.1109/TIT.2011.2173792
19. Silberstein, N., Etzion, T.: Optimal fractional repetition codes based on graphs and designs. IEEE Trans. Inform. Theory **61**(8), 4164–4180 (2015). https://doi.org/10.1109/TIT.2015.2442231

20. Stinson, D.R.: Combinatorial Designs. Springer, Berlin (2003)
21. Su, Y.S.: Optimal pliable fractional repetition codes that are locally recoverable: a bipartite graph approach. IEEE Trans. Inform. Theory **65**(2), 985–999 (2018). https://doi.org/10.1109/TIT.2018.2876284
22. Wang, Z., Tamo, I., Bruck, J.: Explicit minimum storage regenerating codes. IEEE Trans. Inform. Theory **62**(8), 4466–4480 (2016). https://doi.org/10.1109/TIT.2016.2553675
23. Zhu, B., Shum, K.W., Li, H., Hou, H.: General fractional repetition codes for distributed storage systems. IEEE Commun. Lett. **18**(4), 660–663 (2014). https://doi.org/10.1109/LCOMM.2014.030114.132694
24. Zhu, B., Shum, K.W., Li, H.: Fractional repetition codes with optimal reconstruction degree. IEEE Trans. Inform. Theory **66**(2), 983–994 (2019). https://doi.org/10.1109/TIT.2019.2929788