Fractional Repetition Codes For Repair In Distributed Storage Systems

Salim El Rouayheb, Kannan Ramchandran
Dept. of Electrical Engineering and Computer Sciences
University of California, Berkeley
{salim, kannanr}@eecs.berkeley.edu

September 4, 2010

Abstract

We introduce a new class of *Exact Minimum-Bandwidth Regenerating* (MBR) codes for distributed storage systems, characterized by a low complexity *uncoded* repair process that is resilient to multiple node failures. Our model for repair is *table-based*, and thus, differs from the random access model adopted in the literature. We present code constructions based on *regular graphs* and *Steiner systems* for a large set of system parameters. The resulting codes are guaranteed to achieve the storage capacity for random access repair. We refer to these codes as *Fractional Repetition* codes since they consist of splitting the data on each node into several packets and storing multiple copies of each on different nodes in the system. The considered model motivates a new definition of capacity for distributed storage systems, that we call *Fractional Repetition* capacity. We provide upper bounds on this capacity, while a general expression remains an open problem.

I. Introduction

Despite being formed of nodes having a short lifespan, *distributed storage systems* (DSS) are required to store data for long periods of time with a very high reliability. Typically, nodes in the system will unexpectedly leave for different reasons, such as hardware failures in data centers, or peer churning in peer-to-peer (p2p) systems. To overcome this problem, a two-fold solution can be adopted based on *redundancy* and *repair*. Classical erasure codes can be used to introduce redundancy in the system to protect the data from being lost when nodes fail. In addition, to maintain a targeted high reliability, the system is repaired whenever a node fails by replacing it with a new one.

Erasure codes with repair capabilities for distributed storage systems (DSS), termed *Regenerating* codes, were first introduced and studied in the original work of Dimakis et al. in [1]. A distributed storage system is modeled as being formed of n nodes with certain storage capacities. The system gives the user the flexibility to recover its stored file by contacting any k nodes, with $k < n$. We call this property the

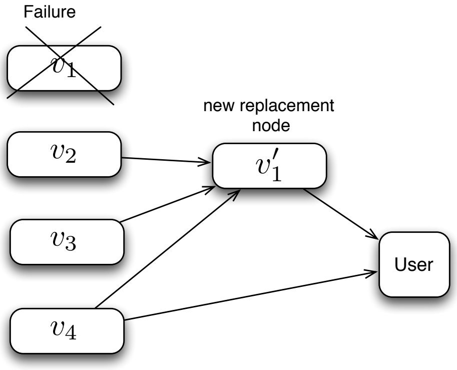

Fig. 1. An example of a distributed storage system with $(n, k, d) = (4, 2, 3)$. Initially, the system is formed of $n = 4$ nodes $v_1, \ldots, v_4$ storing coded packets of a file. The user contacts any $k = 2$ nodes and should be able to decode the stored file. When a node fails, it is replaced by a new one that contacts $d = 3$ nodes to download its data. The figure shows an instance where node v_1 fails and is replaced by node v_1', and a user connected to nodes v_1' and v_4.

MDS property of the DSS, in reference to Maximum Distance Separable (MDS) codes. When a node fails, the system is repaired by replacing the failed node by a new “blank” node. The new node contacts d surviving nodes, downloads encodings of their data and stores it, possibly after compression. The data stored on the new node should conserve the MDS property of the DSS. In analogy with classical codes defined by the two parameters (n, k), a DSS, and the associated Regenerating codes, are specified by the triplet (n, k, d), where the additional parameter d, referred to as the repair degree, accounts for the additional repair requirement. Fig. 1 depicts a $(4, 2, 3)$ DSS showing one repaired failure and one user.

In this work we are interested in constructing *Exact* Regenerating codes with bandwidth-optimal repair, known in the literature as Exact Minimum-Bandwidth Regenerating (MBR) codes. Exactness is a much desired property of Regenerating codes and refers to their ability to reproduce an exact copy of the lost data on a new replacement node. In the minimum-bandwidth regime, a replacement node obtains this exact copy by downloading data from d surviving nodes and storing it with *no compression*.

The common model adopted in the literature for repair is *random access*, where a new replacement node can contact any d arbitrarily chosen surviving nodes to download its data. In this case, the repair degree d indirectly determines the number ρ' of simultaneous failures that the repair process can tolerate, which is here $\rho' = n - d$. Since large number of nodes failing together is a rare event, the DSS should be designed for small values of ρ' (compared to n), in addition to a small values of d to reduce the repair delay and protocol overhead. These two ranges of d and ρ' seem to be conflicting under this model, and we risk over-provisioning the system for unreasonable large number of failures when designing codes with low repair degree. In this paper, we present a new approach that decouples d and ρ' by abandoning the random access model of repair and adopting a scheme that is based on a *repair table*. The repair table specifies for each failure pattern a list of nodes that can be contacted together for repair. This table-based model is very attractive from a practical point of view and goes along with existing tracker-based p2p systems. As a result of this relaxation, we are able to construct Exact MBR codes for the desired ranges of d and ρ', possessing low complexity repair capabilities that do not involve encoding during repair: a replacement node simply downloads a single packet from each of the d node it contacts and stores it. These codes are very well suited for large scale systems with strict requirement on the permitted system downtime and where multiple nodes simultaneously leaving the system is not a rare event.

Our codes are based on a generalization of the construction of Rashmi et al. in [2] and are formed by the concatenation of two constituent codes: an outer MDS code to ensure the required MDS property of the DSS, and an inner repetition code to guarantee efficient uncoded repair (see Fig. II). The design of the inner code represents the challenging task in this construction. We call the inner code a *Fractional Repetition (FR)* code since, in our proposed solution, the stored content of each node is split into d packets, each of which is repeated ρ times in the system, where $\rho = \rho' + 1$[1] is a design parameter representing the *repetition degree* of the FR code. We study the design of FR codes that can achieve the DSS capacity under random access repair. For single failures, i.e., $\rho = 2$, we provide a construction based on *regular graphs* for all feasible values of d for a fixed n. For the general case of multiple failures, i.e., $\rho > 2$, we devise code constructions based on *Steiner systems*. Of particular importance is a construction for DSS for small values of ρ and a repair degree that is a fraction of the surviving nodes. The table-based repair model motivates a new concept of storage capacity for distributed storage systems, referred to as *Fractional Repetition* (FR) capacity, which we define and study.

The rest of the paper is organized as follows. In Section II, we give two examples of Fractional Repetition Codes, then briefly summarize previous related work in Section III. We describe code constructions for the single failure case in Section IV, and for multiple failures in Section V. In Section VI, we define the Fractional Repetition capacity of a DSS and provide some bounds. We conclude in Section VII and discuss related open problems.

II. Examples and Model

An interesting tradeoff was shown to exist between nodes storage capacity and repair bandwidth, i.e., the total amount of data downloaded by a replacement node for repair, in a DSS [3]. An important operation point in this tradeoff corresponds to, what is known in the literature as, the Minimum-Bandwidth Regime. This regime is characterized by a minimum repair bandwidth, making it very appealing for practical systems where bandwidth is in general a more costly resource than storage. Under this regime, a new

[1] An extra copy is needed to achieve resilience to ρ' failures.

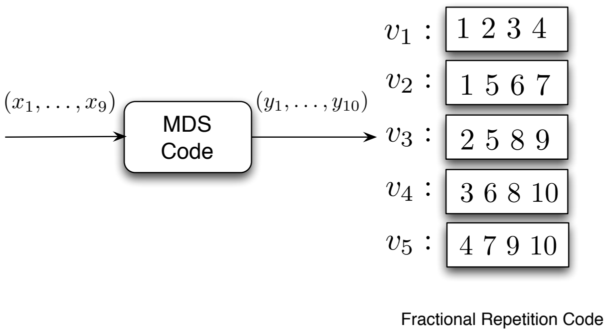

Fig. 2. An Exact Regenerating code for a $(4,3,3)$ DSS formed by an $(10,9)$ outer MDS code followed by a *Fractional Repetition* code of repetition degree $\rho = 2$. The content of each node is split into $d = 4$ packets, each is repeated twice in the system. This code can achieve the MDS property of the DSS, along with exact and uncoded repair in the case of one failure.

replacement node stores all the the data it downloads, without any compression, from the surviving nodes it contacts. We assume a symmetric repair where a replacement node downloads the same amount of data, referred to as a packet, from each node it contacts. In this case, the storage capacity $C_{MBR}(n,k,d)$, in packets, of a DSS with parameters (n,k,d) was proven in [3] to be

$$C_{MBR}(n,k,d) = kd - \binom{k}{2}, \tag{1}$$

assuming a *functional repair* model. Under functional repair, the only constraint on the data stored on a new replacement node is that it should maintain the MDS property of the DSS. It is, however, very desirable to have the replacement node store an exact copy of the lost data. In this case, the regenerating codes are called *Exact*. It was recently shown [4] that Exact MBR codes can always achieve the DSS capacity C_{MBR} of (1). These codes in general require that a node contacted for repair forward coded packets, consisting of linear combination of its data, to a new node.

Next, we present two constructions of Exact Regenerating codes that can also achieve the above capacity with the additional property of achieving *uncoded repair*. The first example is based on the code construction of exact codes for $d = n - 1$ of Rashmi et al. [2].

Example 1: Consider a $(5,3,4)$ DSS with storage capacity equal to 9 packets as given by (1). Let $X = (x_1,\ldots,x_9) \in \mathbb{F}_q^9$ denote the file of 9 packets to be stored on the system. Figure II depicts an Exact MBR code [2] that can achieve the above storage capacity. This code consists of the concatenation of two constituent codes: an outer $(10,9)$ parity check MDS code, followed by a special repetition code. The MDS code takes the file X as an input and outputs the codeword $Y = (y_1,\ldots,y_{10})$, where $y_i = x_i, i = 1,\ldots,9$, and y_{10} is a parity check packet, i.e., $y_{10} = \sum_{i=1}^{9} x_i$. The coded packets $y_1,\ldots,y_{10}$ are then placed on the 5 storage nodes following the pattern of the inner code in Fig. II. I.e., nodes $v_1,\ldots,v_5$

store, respectively, $\{y_1, y_2, y_3, y_4\}, \{y_1, y_5, y_6, y_7\}, \{y_2, y_5, y_8, y_9\}, \{y_3, y_6, y_8, y_{10}\}$ and $\{y_4, y_7, y_9, y_{10}\}$.

A user contacting a node can download all its stored data. Therefore, any user connecting to $k = 3$ nodes will be able to download exactly 9 distinct packets; 12 in total, of which 3 are repeated twice. Thus, due to the MDS property of the outer code, it can recover the whole file X. Moreover, the outer code is such that every storage node shares a distinct packet with each of the remaining nodes in the system, which guarantees exact repair in the case of a single node failure. Indeed, whenever a node fails, its stored data can be exactly recovered by contacting the four surviving nodes and downloading a single packet from each. For instance, when node v_1 fails, a replacement node contacts nodes $v_2, \dots, v_5$, and downloads packets $y_1, \dots, y_4$ from each, respectively. ■

In the inner code of the previous example, each packet y_i is repeated twice in the system. Due to this property, we call this inner code a *Fractional Repetition code* of repetition degree $\rho = 2$. In a Fractional Repetition code for an (n, k, d) DSS, the content of each node is split into d packets, where each is stored on ρ different nodes in the system. In addition to being exact and optimal, the code obtained by the above construction is characterized by *uncoded repair*. When contacted by a new node upon a failure, a survivor node simply forwards a particular packet depending on which nodes have failed. This uncoded repair mechanism is more restrictive than the original model of [3] where a node contacted for repair sends linear combinations of its stored packets to the replacement node. Despite being restrictive, uncoded repair is capacity achieving in this case.

From a system point of view, uncoded repair is a very desirable property since it allows a fast and low complexity repair of the system. In this paper, we assume that it is a design requirement for the repair process to be uncoded in addition to being exact, and we investigate the construction of Regenerating codes with these properties. The Exact MBR codes devised in [2] are an example of such codes, however, they are specialized to the case of $d = n - 1$ and require contacting $n - 1$ nodes in the case of a single failure. This may not always be feasible due, for example, to multiple failures occurring simultaneously, or certain nodes being busy serving the users. It is then important to build Exact MBR codes with uncoded repair for smaller values of d where a new replacement node contacts just a fraction of the nodes in the system. The next example describes such a code for a $(7, 3, 3)$ DSS that can achieve exact uncoded repair in the case of two nodes failure. This code is based on the projective plane $PG(2, 2)$ (Fig. 3(b)) and gives a hint on our general construction techniques that will be detailed in the following sections.

Example 2: The code that we propose for the $(7, 3, 3)$ DSS is also constructed by concatenating two constituent codes: an outer $(7, 6)$ MDS code followed by the Fractional Repetition code of degree $\rho = 3$ depicted in Fig. 3(a). It can be seen that each of the 7 packets that form the output of the MDS code is repeated 3 times in the DSS on 3 distinct nodes. Therefore, there is always a surviving copy of any packet in the system whenever two nodes fail. The code then guarantees exact uncoded repair for up to two node failures.

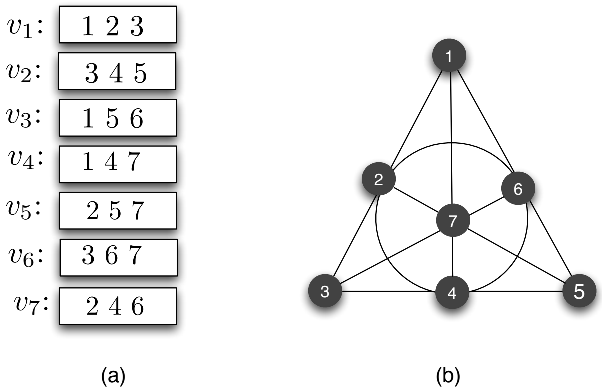

Fig. 3. (a) The Projective plane $PG(2,2)$, also known as the Fano matroid. (b) The corresponding Fractional Repetition code with repetition degree $\rho = 3$ for a DSS with $(n,k,d) = (7,3,3)$. Each of the 7 lines in $PG(2,2)$, including the circle, represents a storage node. The points lying on that line correspond to the packets that are stored on the node. This code can achieve the capacity $C_{MBR} = 6$ for this DSS, and has an exact and uncoded repair process.

The structure of the Fractional Repetition code is deduced from the projective plane $PG(2,2)$ (Fig 3(b)) of order 2, of 7 points and 7 lines (including the circle), also known as the Fano matroid. Each line in $PG(2,2)$ corresponds to a storage node in the DSS, and the three points belonging to that line give the indices of the packets stored on the node. In the projective plane, any two lines intersect in exactly one point. This implies that any user that contacts 3 nodes can get *at least* $3 \times 3 - \binom{3}{2} = 6$ distinct packets. For instance, a user contacting nodes v_2, v_4 and v_5 will get exactly 6 different packets, namely the ones having indices $\{1,2,3,4,5,7\}$, whereas another user contacting v_1, v_3 and v_4 will get all the 7 packets. In general, we are limited by the user that gets the least number of packets, which is 6 here. Therefore, the outer MDS code allows any user to recover a stored file of 6 packets which is exactly the capacity $C_{MBR}(7,3,3)$ of (1). ■

In the code of Example 2, a replacement node has to contact a *specific* set of d nodes for repair, depending on which nodes have failed. For example, when node v_1 fails, a replacement node can recover the lost packets by connecting to nodes $\{v_4, v_5, v_6\}$, but not $\{v_2, v_3, v_4\}$. We assume, therefore, that there is a *repair table* maintained in the system that is available to all the nodes in the DSS. The repair table indicates for each possible failure pattern the set of nodes that can be contacted for repair, and which packet to download from each. This repair model based on a repair table is inspired by tracker-based p2p file distribution systems and will be adopted throughout this paper. It differs from the random access model adopted in the literature where repair can be performed by contacting *any* d surviving node. Table I summarizes the differences between the repair model adopted here and the original model of [3].

The previous two examples suggest a general method for constructing Exact MBR codes with uncoded

Original repair model in [3]	Repair model of FR codes
Functional: replacement data should satisfy MDS property.	*Exact*: replacement data is an exact copy of the lost one.
Coded: new node downloads linear combination of packets.	*Uncoded*: new node downloads specific packet with no coding.
Random Access: the new node contacts any d surviving nodes.	*Repair Table*: a table specifies the set of d nodes to be contacted for repair.

TABLE I

A COMPARISON BETWEEN THE MODEL FOR REPAIR IN THE ORIGINAL WORK OF DIMAKIS ET AL. IN [3] AND THE MODEL FOR REPAIR FOR THE FRACTIONAL REPETITION CODES PROPOSED HERE.

repair process that is resilient to up to ρ' failures, by concatenating an outer MDS code with an inner Fractional Repetition code with repetition degree $\rho = \rho' + 1$. Since MDS codes exist for all feasible parameters provided that the packets are taken from an alphabet of large enough size, the challenging part of the suggested construction is designing the Fractional Repetition code. Assuming all the packets in the system are to be equally protected, we are motivated to provide the following general definition of FR codes:

Definition 3 (Fractional Repetition Codes): A *Fractional Repetition* (FR) code $\mathcal{C}$, with repetition degree ρ, for an (n,k,d) DSS, is a collection of n d-subsets[2], $\mathcal{C} = \{V_1, V_2, \ldots, V_n\}, |V_i| = d, i = 1, \ldots, n$, of a set $\Omega = \{1, \ldots, \theta\}$, such that each element of Ω belongs to exactly ρ sets in the collection.

In this definition, each set V_i contains the indices of the packets that are stored on node $v_i, i = 1, \ldots, n$ and which are output by the MDS code. The value of θ, which will be determined later, corresponds to the length of the codewords of the outer MDS code. For example, following this definition, the FR code of Example 2 can be written as $\mathcal{C} = \{V_1, \ldots, V_7\}$ with $V_1 = \{1,2,3\}, V_2 = \{3,4,5\}, V_3 = \{1,5,6\}, V_4 = \{1,4,7\}$ $V_5 = \{2,5,7\}, V_6 = \{3,6,7\}, V_7 = \{2,4,6\}$, where $\Omega = \{1, \ldots, 7\}$.

III. RELATED WORK

The pioneering work of Dimakis et al. in [1], [3], [5] introduced and studied Regenerating codes for storage and repair in distributed storage systems. The authors showed that there exists a tradeoff between storage capacity and repair bandwidth in these systems. Moreover, they determined their storage capacity using network coding techniques and showed that random linear Regenerating codes are optimal under a functional repair model.

[2] A d-subset is a subset of cardinality d.

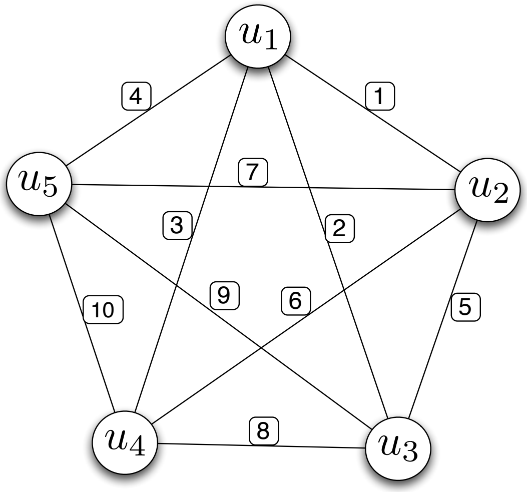

Fig. 4. The complete graph K_5 on 5 vertices. The labeling of the edges from 1 to $\binom{5}{2} = 10$ gives the FR code with $\rho = 2$ for the DSS $(5, 4, 3)$ depicted in Fig. II. The edges adjacent to vertex u_i give the indices of the packets stored on node v_i in the DSS.

Subsequent works focused on the design of Exact Regenerating codes that can repair the system by regenerating an exact copy of the data lost as a result of a failure. Rashmi et al. constructed Exact MBR codes first in [2] for the special case of $d = n - 1$, and then for all feasible values of the repair degree d in [6]. The existence of Exact Regenerating codes for another important regime, known as the Minimum Storage Regime (MSR), was demonstrated in [7], [8], and deterministic constructions were investigated in [9], [10], [11], [2]. The design of Regenerating codes that can protect the distributed storage system from eavesdropping or data corruption by malicious adversaries was studied in [12], [13], [14].

IV. Fractional Repetition Codes for Single Failures

In this section, we study Fractional Repetition codes with repetition degree 2 that can guarantee exact and uncoded repair in the case of only a single node failure. We provide a code construction based on *regular graphs* that can achieve the capacity C_{MBR} of (1) for all feasible values of n and d. To that end, we define the rate $R_{\mathcal{C}}(k)$ of an FR code $\mathcal{C}$ as the maximum file size, i.e., the maximum number of distinct packets, that the code can deliver to *any* user contacting k nodes.

Definition 4 (FR Code Rate): The rate $R_{\mathcal{C}}(k)$ of an FR code $\mathcal{C} = \{V_1, V_2, \dots, V_n\}$ for a DSS with parameters (n, k, d) is defined as

$$R_{\mathcal{C}}(k) := \min_{\substack{I \subset [n] \\ |I| = k}} | \cup_{i \in I} V_i|. \tag{2}$$

As it can be seen in the examples in the previous section, the DSS parameter k specifying the number of nodes contacted by a user, is not intrinsically related to the construction of the FR code. An FR code designed for a DSS with parameters (n, k_1, d) can be directly used for another with parameters (n, k_2, d). An FR code $\mathcal{C}$ is said to be a *universally good* code if its rate is guaranteed to be no less then the capacity C_{MBR} of the DSS under functional, coded and random access repair, i.e.,

$$R_{\mathcal{C}}(k) \geq C_{MBR}(n, k, d), \tag{3}$$

for all $k = 1, \dots, d$. The inequality in the previous definition is justified by the fact that FR code can have rates that exceed C_{MBR}, a property that will be investigated more in Section VI.

In total, an (n, k, d) DSS stores nd packets. When an FR code of degree ρ is used, θ distinct packets are stored in the system, where each is repeated exactly ρ times. Therefore, the following relation exists between the FR code parameters:

Proposition 5: The parameter θ in Def. 3 of an FR code of degree ρ for an (n, k, d) DSS is given by,

$$nd = \theta\rho. \tag{4}$$

As an application of Prop. 5, consider the design of an FR code with $\rho = 3$ for a $(7, 3, 3)$ DSS. This code should make use of $\theta = \frac{7\times 3}{3} = 7$ distinct packets, which is exactly the number of packets used in Example 2.

The Exact MBR codes of Rashmi et al. were proposed in [2] as capacity achieving codes for the special case of $d = n - 1$. In this case, when a node fails, all the remaining nodes in the system are contacted by the replacement node, which implies that the random access and table-based repair models are equivalent. These codes can be viewed as special FR codes as shown in Example 1. Their general construction can be described with the assistance of the complete graph K_n defined on n vertices $u_1, \dots, u_n$, with edges indexed from 1 to $\binom{n}{2}$. Prop 5 gives $\theta = \frac{n(n-1)}{2} = \binom{n}{2}$ distinct packets. The FR code is then obtained by storing on node $v_i, i = 1, \dots, n$, the packets indexed by the edges adjacent to vertex u_i in K_n. Figure IV depicts the complete graph K_5 with its edges indexed in a way to give the FR code of Fig. II.

For the above codes, the repair process is very costly since it involves contacting all the nodes that are "up" in the system. This may not be always feasible for systems with a large number of nodes. Thus, it is important to provide constructions of FR codes for smaller values of d. For $\rho = 2$, Prop. 5 gives a necessary condition for the existence of FR codes, that is nd should be even. Next, we show that this is also a sufficient condition, and provide a general code construction based on regular graphs.

A d-regular graph $R_{n,d}$ on n vertices is a simple graph where all vertices have the same degree d. The graph $R_{n,d}$ has $nd/2$ vertices, and exists whenever nd is even [3].

[3] This follows, for example, from Gayle-Ryser theorem [15, Th. 2.1.3]

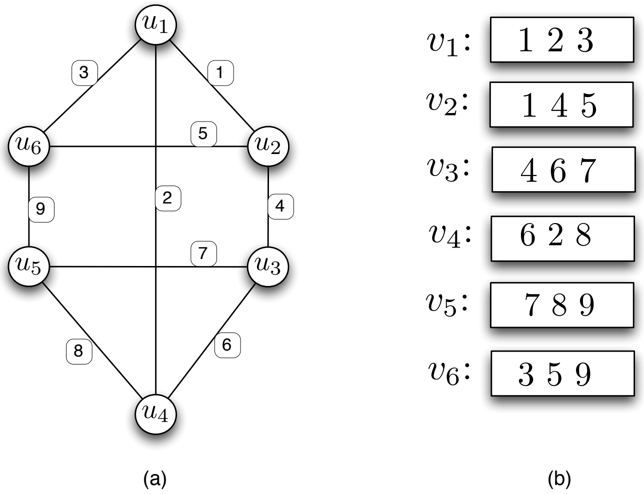

Fig. 5. (a) $R_{6,3}$ a 3-regular graph on 6 vertices with edges indexed from 1 to $\frac{6\times 3}{2} = 9$. (b) The corresponding universally good FR code with $\rho = 2$ obtained by Construction 6 for a DSS with $n = 6$ an $d = 3$.

Construction 6: An FR code with repetition degree $\rho = 2$ can be constructed for an (n, k, d) DSS, with nd even, in the following way:

1) Generate a d-regular graph $R_{n,d}$ on n vertices $u_1, \ldots, u_n$.
2) Index the edges of $R_{n,d}$ from 1 to $\frac{nd}{2}$.
3) Store on node v_i in the DSS the packets indexed by the edges that are adjacent to vertex u_i in the graph $R_{n,d}$.

The regular graph in step 1 can be randomly generated using efficient randomized algorithms that are well-studied in the literature, see for example [16]. The fact that the FR codes obtained by this construction have repetition degree $\rho = 2$ is a direct consequence of the graph being simple with edges being adjacent to exactly two vertices. This also implies that any two nodes cannot have in common more than one packet. Therefore, among any k nodes observed by a user, at most $\binom{k}{2}$ packets are duplicated. Therefore, we have the following lemma.

Lemma 7: The FR codes with repetition degree $\rho = 2$ obtained by Construction 6 are *universally good* codes.

Note that Construction 6 subsumes that described in [2] since a complete graph is a regular graph for which $d = n - 1$. Note that the existence condition of FR codes for single failures, i.e., nd is even, is not restrictive, since from a system designer perspective, n can be always chosen to be even. In this case, Construction 6 results in FR codes for all possible values of d. Fig. IV shows a 3-regular graph $R_{6,3}$ and the corresponding universally good FR code obtained by Construction 6 for the DSS with $n = 6$ and

$d = 3$.

The next lemma shows that for small values of k the rate of any Universally Good FR code with $\rho = 2$, including those obtained by Construction 6, is exactly C_{MBR}.

Lemma 8: For any Universally Good FR code $\mathcal{C}$ having a repetition degree $\rho = 2$ for an (n, k, d) DSS with $k \leq \frac{n}{n-d}$, the rate of the code is

$$R_{\mathcal{C}}(k) = kd - \binom{k}{2}.$$

Proof: Consider an FR code $\mathcal{C} = \{V_1, \ldots, V_n\}$ with $\rho = 2$ for an (n, k, d) DSS. Since $\mathcal{C}$ is Universally Good, two different subsets V_i and V_j intersect in at most one element. Construct a graph G on n vertices $u_1, \ldots, u_n$ by connecting two different vertices u_i and u_j if $V_i \cap V_j \neq \emptyset$. The graph G is d-regular and contains $\frac{nd}{2}$ edges. By Turan Theorem [21], G has clique of size least $\frac{n}{n-d}$. Therefore, there exists a collection of k sets in $\mathcal{C}$ such that any two intersect in distinct elements. ■

V. Fractional Repetition Codes for Multiple Failures

In this section, we propose constructions for universally good FR codes with $\rho > 2$ characterized by an uncoded repair process that is resilient to more than one failure. While the FR codes with $\rho = 2$ described in the previous section were based on regular graphs, the constructions in this section will be based on a combinatorial structure known as *Steiner System* which can be thought of as a generalization of the projective plane of Example 2.

A. Steiner Systems

Definition 9 (Steiner System): A Steiner system $S(t, k', v)$ is a collection of k'-subsets, $B_1, \ldots, B_b$, called *blocks*, of a set $\mathcal{V}$ of cardinality $|\mathcal{V}| = v$, with the property that any t-subset of $\mathcal{V}$ is contained in *exactly* one block.

It can be shown that in a Steiner system every element of $\mathcal{V}$ belongs to the same number of blocks denoted by r [17, p. 60]. We will be mostly interested in Steiner systems with $t = 2$. Simple counting arguments give the following two properties of $S(2, k', v)$.

Proposition 10: The parameters b and r of a Steiner system $S(2, k', v)$ are given by:

$$bk' = vr, \tag{5}$$

$$v - 1 = r(k' - 1). \tag{6}$$

Equation (5) is equivalent to (4) for FR codes. The Fano matroid of Fig. 3(a) is an example of a Steiner system where $\mathcal{V}$ is the set of 7 points and the blocks are the lines (including the circle). The Fano matroid is indeed $S(2, 3, 7)$ since there is a single line that goes through any two points. Prop. 10

gives $r = 3$, i.e. each point belong to exactly 3 lines, and $b = 7$, i.e., the non-Fano matroid contains 7 lines, which can be easily checked on the figure.

For a Steiner system $S(t, k, v)$ to exist, it is necessary that the parameters b and r given in Prop. 10 be integers. Wilson proved in [18] that this condition is also sufficient for a sufficiently large v.

Theorem 11: Given a positive integer k', Steiner systems $S(2, k', v)$ exist for all sufficiently large integers v for which the congruences

$$vr \equiv 0 \bmod k' \tag{7}$$

$$v - 1 \equiv 0 \bmod k' - 1 \tag{8}$$

are valid.

B. Code Constructions

In this section we present two constructions of universally good FR codes derived from Steiner systems. Example 2 suggests the following direct construction:

Construction 12: Given a Steiner system $S(2, d, \theta)$ with blocks $B_1, \ldots, B_n \subset \mathcal{V} = [\theta]$, an FR code $\mathcal{C}$ with repetition degree ρ for a DSS (n, k, d) can be obtained by taking $\mathcal{C} = \{B_1, \ldots, B_n\}$. The parameters n and ρ as given by Prop 10 are $n = \frac{\theta(\theta-1)}{d(d-1)}$ and $\rho = \frac{\theta-1}{d-1}$.

By definition, any two blocks in $S(t, k, v)$ cannot intersect in more than $t - 1$ elements. This implies that in the FR codes obtained by Construction 12, two nodes can have at most one packet in common. Thus, among any k nodes there are at most $\binom{k}{2}$ packets that are repeated twice. Therefore, the obtained FR codes can achieve the capacity C_{MBR} for all $k = 1, \ldots, d$.

Lemma 13: The FR codes obtained by Construction 12 are universally good.

Construction 12 is simple, however, it has the disadvantage that the two important parameters of the code design, n and ρ, cannot be explicitly chosen beforehand. Typically, an FR code should be designed for a given large number of nodes n, a small repetition degree ρ since it is very unlikely that many nodes fail simultaneously, and a repair degree d that is just a fraction of the surviving nodes. The second construction satisfies these requirements.

Construction 14: Given a Steiner system $S(2, \rho, n)$ with blocks $B_1, \ldots, B_\theta \subset \mathcal{V} = [n]$, an FR code $\mathcal{C} = \{V_1, \ldots, V_n\}$ with a repetition degree ρ for a DSS (n, k, d) can be obtained by taking

$$V_i = \{j | i \in B_j\}.$$

The parameters d and θ as given by Prop 10 are $d = \frac{n-1}{\rho-1}$ and $\theta = \frac{n(n-1)}{\rho(\rho-1)}$.

We refer to the codes obtained by this construction as *Transpose* codes since the role of the blocks and the elements of $\mathcal{V}$ are reversed. The blocks correspond to the stored packets in the DSS and the elements of $\mathcal{V}$ to the storage nodes. Therefore, any two nodes have *exactly* one packet in common, and the rate of Transpose codes can be exactly determined as given by the following lemma.

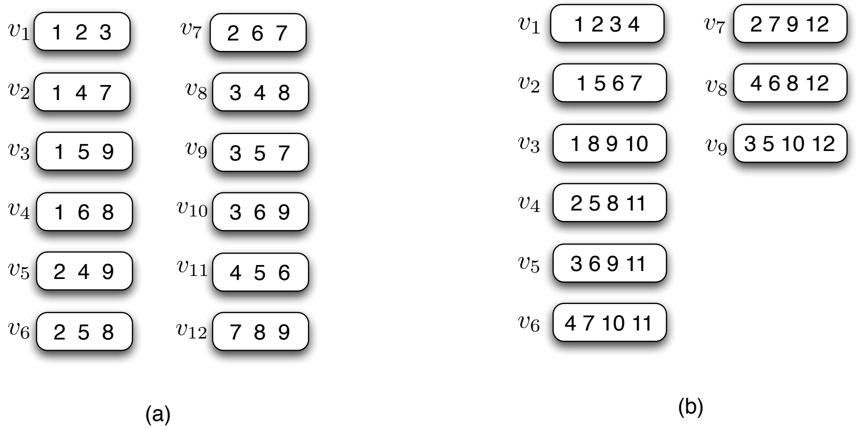

Fig. 6. (a) FR code with $\rho = 4$ for a DSS with $n = 12$ and $d = 3$ derived from the Steiner system S(2,3,9) using Construction 12. (b) FR code with $\rho = 3$ for a DSS with $n = 9$ and $d = 4$ derived from the same Steiner system using Construction 14.

Lemma 15: The rate of the Transpose codes $\mathcal{C}$ obtained by Construction 14 is exactly

$$R_{\mathcal{C}}(k) = kd - \binom{k}{2},$$

and these codes are universally good.

To highlight the difference between these two constructions, we give an example in Figure V-B when they are both applied to the (unique) Steiner system S(2,3,9) [19, p. 27]. Construction 12 gives an FR code with $\rho = 4$ for a DSS with $n = 12$ and $d = 3$, whereas Construction 12 gives an FR code with $\rho = 3$ for a DSS with $n = 9$ and $d = 4$. It can be seen that Construction 12 gives a better handle on the parameters n and ρ of the FR code, since $\rho = k'$ and $n = v$ for a Steiner system $S(2, k', v)$. Note that these two constructions will give the same FR code (up to relabeling) when applied to the Fano matroid of Fig. 3(b).

The next result strengthens Lemma 15 by showing that Transpose codes are rate optimal in the sense that no other Universally Good FR codes with the same family of parameters satisfying $d = \frac{n-1}{\rho-1}$, can achieve higher rates.

Lemma 16: Any Universally Good code $\mathcal{C}$ with repetition degree ρ and $d = \frac{n-1}{\rho-1}$ has a rate given by

$$R_{\mathcal{C}}(k) = kd - \binom{k}{2}$$

Proof: Consider an FR code $\mathcal{C} = \{V_1, \dots, V_n\}$ with repetition degree ρ and $d = \frac{n-1}{\rho-1}$. Since $\mathcal{C}$ is Universally Good, two different subsets in $\mathcal{C}$ are either disjoint, or have exactly one element in common. Moreover, the average number of distinct packets observed by a user contacting $k = 2$ distinct nodes is

$2d-1$ packets, i.e.,

$$\binom{n}{2}^{-1}\sum_{i\neq j}|V_i\cup V_j| = 2d-1,$$

as given by Lemma 19 which will be proven in the next section. Therefore, any two sets in $\mathcal{C}$ cannot be disjoint and intersect in exactly one packet. Therefore, $R_{\mathcal{C}}(k)\leq kd-\binom{k}{2}$. ■

The previous two constructions assume the existence of the Steiner system with the desired parameters, which is not always true. However, Steiner systems $S(2,k,v)$ are known to exist for small values of k, namely $k=2,\dots,5$, whenever the integrality conditions given by Prop. 10 are satisfied. This result in conjunction with Construction 14 gives the necessary and sufficient conditions for the existence of Transpose codes with low repetition degree, which is indeed the range of ρ that we are interested in.

Lemma 17: Transpose codes with repetition degree $\rho=2,\dots,5$ exist if and only if

$$n-1\equiv 0 \bmod \rho-1 \tag{9}$$

$$n(n-1)\equiv 0 \bmod \rho(\rho-1) \tag{10}$$

The second dominant failure pattern, after single node failures, is the simultaneous failure of two nodes. FR codes for this case having a repetition degree $\rho=3$ always exist whenever $n\equiv 1,3 \mod 6$ by the previous lemma. These codes can be obtained by Construction 14 using Steiner systems $S(2,3,n)$, known as *Steiner triple systems*. Steiner triple systems are historically the most investigated systems in the literature and explicit constructions, such as Bose and Skolem constructions, exist for all feasible values of n [20].

VI. Capacity under Exact Uncoded Repair

The Universally Good FR codes constructed in the previous sections are guaranteed to have a rate greater or equal to the capacity C_{MBR}, achieved by random Regenerating Codes as demonstrated in [3]. However, there exist cases where FR codes can achieve a storage capacity that exceeds C_{MBR}. Figure VI depicts an FR code for the $(6,3,3)$ DSS based on a 3×3 grid. The points in the grid represent the packets, and the lines the storage nodes. Since this grid does not contain any triangles, any user contacting 3 nodes have access to at least 7 distinct packets. Therefore, this code has a rate $R(3)=7>C_{MBR}=6$.

We refer to the maximum file size that a DSS with parameters (n,k,d) can store under exact and uncoded repair as its *Fractional Repetition (FR) capacity* C_{FR} defined as follows:

Definition 18 (Fractional Repetition Capacity): The Fractional Repair (FR) capacity, denoted by C_{FR} of a distributed storage system with parameters (n,k,d) when the repair process in the system is required to be uncoded, exact and resilient to $\rho-1$ failures is defined for all ρ satisfying $nd\equiv 0 \bmod \rho$ as

$$C_{FR}(k,\rho):=\max_{\mathcal{C}} R_{\mathcal{C}}(k),$$

where $\mathcal{C}$ is any FR code with repetition degree ρ for an (n,k,d) DSS.

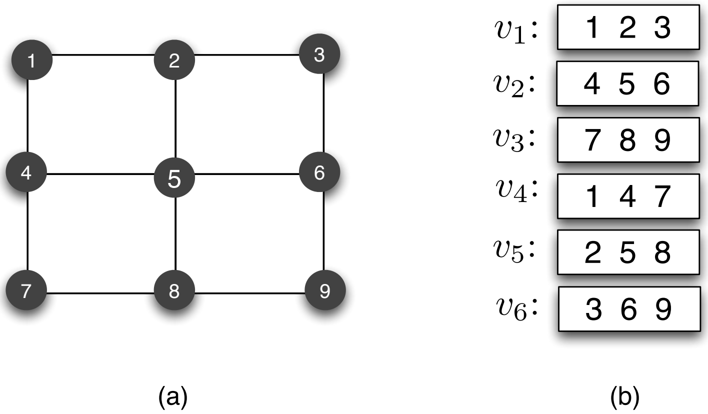

Fig. 7. (a) A 3×3 grid of 9 points and 6 lines. (b) The corresponding FR code achieving a storage capacity exceeding C_{MBR}.

The condition on ρ in the definition above is needed by Prop. 10 to guarantee the existence of an FR code $\mathcal{C}$. The code constructions of the previous sections imply lower bounds on the FR capacity. Next, we derive two upper bounds on C_{FR}. The first is based on an averaging argument and is presented in Lemma 19.

Lemma 19: For a DSS with parameters (n, k, d),

$$C_{FR}(k, \rho) \leq \left\lfloor \frac{nd}{\rho} \left(1 - \frac{\binom{n-\rho}{k}}{\binom{n}{k}} \right) \right\rfloor.$$

Proof: Let $\mathcal{C} = \{V_1, \ldots, V_n\}$ be an FR code with repetition degree ρ, where $V_i \subset [\theta], |V_i| = d$ and $\theta = \frac{nd}{\rho}$ as given by Prop. 5. Define the set $\mathcal{U}$ as

$$\mathcal{U} := \{U_I = \cup_{i \in I} V_i : I \subset [n], |I| = k\}.$$

The set U_I represent the set of packets observed by a user contacting the nodes in the DSS indexed by the elements in I. We want to show that the term on the right in the inequality is the average cardinality of the sets in $\mathcal{U}$ under uniform distribution. We denote this average by $\overline{U_I}$. To find $\overline{U_I}$, we count the following quantity $\sum_{U_I \in \mathcal{U}} |U_I|$ in two ways. First, we have by definition

$$\sum_{U_I \in \mathcal{U}} |U_I| = \binom{n}{k} \overline{U_I}.$$

But, each element in $[\theta]$ belongs to exactly $\binom{n}{k} - \binom{n-\rho}{k}$ sets in $\mathcal{U}$. Therefore,

$$\sum_{U_I \in \mathcal{U}} |U_I| = \theta \left(\binom{n}{k} - \binom{n-\rho}{k} \right).$$

The upper bounds follows then from the fact that there must be in $\mathcal{U}$ at least one set of cardinality less that the average. ■

For instance, for the DSS $\mathcal{D}(7,3,3)$, Lem. 19 implies that $R(3,3) \leq \lfloor 6.2 \rfloor = 6$. Therefore, the FR code of Example 2 is optimal and $C_{FR}(3,3) = 6$. However, the above upper bound has the disadvantage of becoming very loose for large values of n and k since the FR capacity is by definition a worst case measure.

Next, we give a second upper bound on the FR capacity of a DSS that is defined using a recursive function, and which is usually tighter than the previous one.

Lemma 20: For a DSS $\mathcal{D}(n,k,d)$, the FR capacity is upper bounded by the function $g(k)$,

$$C_{FR}(k,\rho) \leq g(k),$$

where $g(k)$ is defined recursively as

$$g(1) = d, \tag{11}$$

$$g(k+1) = g(k) + d - \left\lceil \frac{\rho g(k) - kd}{n-k} \right\rceil. \tag{12}$$

Proof: (sketch) The proof is established by induction on k. It is evident that the statement is true for $k=1$. Let us assume that it is true for $k = k_0$, and prove it for $k = k_0 + 1$. Consider an FR code $\mathcal{C} = \{V_1, \ldots, V_n\}$ of repetition degree ρ for an (n,k,d) DSS. Pick k_0 sets from $\mathcal{C}$. Without loss of generality, let these sets be $V_1, \ldots, V_{k_0}$, and their union $U := \cup_{i=1}^{k_0} V_i$. By the induction hypothesis, we have $|U| \leq g(k_0)$.

Now, since each element in U is repeated ρ times, there are $\rho g(k_0) - k_0 d$ copies of the elements in U that should be stored on the remaining $n - k_0$ nodes in the system. Therefore, there must exist one storage node $v_j, j \in \{k_0+1, \ldots, n\}$, where $|V_j \cap U| \geq \lceil \frac{\rho g(k_0) - k_0 d}{n - k_0} \rceil$. Therefore, a user contacting the $k_0 + 1$ nodes $v_1, \ldots, v_{k_0}$ in addition to node v_j will observe

$$|U \cup V_j| \leq g(k_0) + d - \lceil \frac{\rho g(k_0) - k_0 d}{n - k_0} \rceil,$$

distinct packets. ■

For storage systems with $d = \frac{n-1}{\rho-1}$, the previous lemma implies that for small values of $k, (k = O(n^{1/2}))$, Transpose codes achieve the FR capacity which is exactly C_{MBR}:

Corollary 21: For a DSS (n,k,d) with $d = \frac{n-1}{\rho-1}$ and $n > k + (\rho-2)\binom{k}{2}$,

$$C_{FR}(k,\rho) = kd - \binom{k}{2}.$$

VII. Conclusion and Open Problems

We proposed a new class of Exact Minimum-Bandwidth Regenerating (MBR) codes for distributed storage systems, that we call *Fractional Repetition* (FR) codes, characterized by a low complexity *uncoded* repair process. An FR code with repetition degree ρ is resilient to $\rho - 1$ failures, and consists of splitting the data on each node into multiple packets and storing ρ copies of each on distinct nodes in the system. An additional outer MDS code guarantees that a user contacting a sufficient number of storage nodes will be able to recover the stored file.

For single node failures, i.e. $\rho = 2$, we presented a construction of FR codes based on *regular graphs* for all feasible system parameters. For the multiple failures case, i.e., $\rho > 2$, we presented two code constructions based on *Steiner systems*. Of particular importance are the constructed *Transpose* codes where the nodes contacted for repair are just a fraction of the surviving ones. All the obtained codes are guaranteed to achieve the storage capacity under random access repair. The adopted table-based repair model motivates a new concept of Fractional Repetition (FR) capacity for distributed storage systems, which we studied and derived some upper bounds.

This work constitutes the first step in the study of Fractional Repetition codes and many important questions remain open. For instance, it is not known whether FR codes with $\rho > 2$ exist for system parameters not covered by our constructions. Moreover, a general expression of the FR capacity is still an open problem, as well as codes that can achieve it.

References

[1] A. G. Dimakis, P. B. Godfrey, M. J. Wainwright, and K. Ramchandran, “Network coding for distributed storage systems,” in *INFOCOM’07*, 2007.

[2] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, “Explicit codes minimizing repair bandwidth for distributed storage,” in *ITW*, 2010.

[3] A. Dimakis, P. Godfrey, Y. Wu, M. Wainright, and K. Ramchandran, “Network coding for distributed storage systems,” *to appear in IEEE Trans. Inform. Theory*.

[4] K. V. Rashmi, N. B. Shah, and P. V. Kumar, “Optimal exact-regenerating codes for distributed storage at the MSR and mbr points via a product-matrix construction,” in *arXiv:1005.4178v1 [cs.IT]*, 2010.

[5] Y. Wu, A. G. Dimakis, and K. Ramchandran, “Deterministic regenerating codes for distributed storage,” in *Proc. Allerton Conference on Control, Computing and Communication*, 2007.

[6] K. V. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran, “Explicit and optimal exact-regenerating codes for the minimum-bandwidth point in distributed storage,” in *Int. Sym. on Inf. Th. (ISIT’10)*, 2010.

[7] C. Suh and K. Ramchandran, “On the existence of optimal exact-repair mds codes for distributed storage,” tech. rep., 2010.

[8] V. R. Cadambe, S. A. Jafar, and H. Maleki, “Title: Distributed data storage with minimum storage regenerating codes - exact and functional repair are asymptotically equally efficient,” in *arXiv:1004.4299v1 [cs.IT]*, 2010.

[9] K. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran, “Exact regenerating codes for distributed storage,” in *Allerton Conference on Control, Computing, and Communication, Urbana-Champaign, IL*, 2009.

[10] Y. Wu and A. G. Dimakis, “Reducing repair traffic for erasure coding-based storage via interference alignment,” in *IEEE Internat. Symp. Inform. Th.*, 2009.

[11] C. Suh and K. Ramchandran, “Exact regeneration codes for distributed storage repair using interference alignment,” in *Proc. IEEE Intl Symp. on Information Theory (ISIT)*, 2010.

[12] S. Pawar, S. E. Rouayheb, and K. Ramchandran, “n secure distributed data storage under repair dynamics,” in *Int. Sym. on Inf. Th. (ISIT’10)*, 2010.

[13] S. Pawar, S. E. Rouayheb, and K. Ramchandran, “Securing dynamic distributed storage systems against eavesdropping and adversarial attacks,” *submitted to Special Issue of the IEEE Trans. on Inf. Th. (Facets of Coding Theory)*, 2010.

[14] T. K. Dikaliotis, A. G. Dimakis, and T. Ho, "Security in distributed storage systems by communicating a logarithmic number of bits," in *IEEE Internat. Symp. Inform. Th. (ISIT'10)*, 2010.

[15] R. A. Brualdi, *Combinatorial Matrix Classes*. Cambridge University Press, 2006.

[16] J. Kim and V. H. Vu, "Generating random regular graphs," in *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, (San Diego, CA, USA), pp. 213–222, 2003.

[17] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. North Holland, June 1988.

[18] R. M. Wilson, "An existence theory for pairwise balanced designs: Iii- proof of the existence conjectures," *J. Comb. Theory*, vol. 18A, pp. 71–79, 1975.

[19] C. J. Colbourn and J. H. Denitz, *Handbook of Combinatorial Designs, Second Edition*. Chapman and Hall/CRC, 2006.

[20] C. C. Lindner and C. A. Rodger, *Design Theory, Second Edition*. Chapman and Hall/CRC, 2008.

[21] R. Diestel, *Graph Theory*. Springer; 3rd edition, 2006.