Ph.D. DISSERTATION

Outer Bounds on the Storage-Bandwidth Tradeoff of Linear Exact-Repair Regenerating Codes

선형 동일 복구 재생 부호의 저장량과 통신량 간 상충 관계의 외부 경계에 관한 연구

Hyuk Lee

August 2017

Department of Electrical Engineering and Computer Science
College of Engineering
Seoul National University

Ph.D. DISSERTATION

Outer Bounds on the Storage-Bandwidth Tradeoff of Linear Exact-Repair Regenerating Codes

선형 동일 복구 재생 부호의 저장량과 통신량 간 상충 관계의 외부 경계에 관한 연구

Hyuk Lee

August 2017

Department of Electrical Engineering and Computer Science
College of Engineering
Seoul National University

Outer Bounds on the Storage-Bandwidth Tradeoff of Linear Exact-Repair Regenerating Codes

Advisor: Jungwoo Lee

Presented to the Graduate School of Seoul National University in Partial Fulfillment of the Requirements for

The Degree of Doctor of Philosophy

May 2017

by

Hyuk Lee
Department of Electrical Engineering and Computer Science
College of Engineering
Seoul National University

This dissertation is approved for

The Degree of Doctor of Philosophy

June 2017

Chairman ____________________

Vice chairman ____________________

Member ____________________

Member ____________________

Member ____________________

Outer Bounds on the Storage-Bandwidth Tradeoff of Linear Exact-Repair Regenerating Codes

선형 동일 복구 재생 부호의 저장량과 통신량 간 상충 관계의 외부 경계에 관한 연구

지도교수 이 정 우

이 논문을 공학박사 학위논문으로 제출함

2017년 5월

서울대학교 대학원

전기·컴퓨터 공학부

이 혁

이혁의 공학박사 학위 논문을 인준함

2017년 6월

위 원 장: ______________________

부위원장: ______________________

위 원: ______________________

위 원: ______________________

위 원: ______________________

Abstract

Distributed storage systems disperse data to a large number of storage nodes connected in a network. When some of the storage nodes fail, a storage system should be able to repair them by downloading data from other surviving nodes. The amount of data traffic during the repair, called repair bandwidth, is one of the important performance metrics of distributed storage systems. Cooperative regenerating codes are a class of recently developed erasure codes which are optimal in terms of minimizing the repair bandwidth. An (n, k, d, r)-cooperative regenerating code has n storage nodes, where k arbitrary nodes are enough to reconstruct the original data, and r failed nodes can be repaired cooperatively with the help of d arbitrary surviving nodes.

In the regenerating-code framework, there exists a tradeoff between the storage capacity of each node α and the repair bandwidth γ. The tradeoff of functional repair codes are fully characterized by Shum et al, but the problem of specifying the optimal storage-bandwidth tradeoff of the exact repair codes remains open. In this dissertation, two outer bounds on the storage-bandwidth tradeoff under the exact repair model are proposed. The outer bounds suggest the (α, γ) pairs that no exact repair codes can achieve but only functional repair codes can.

The first outer bound considers general set of parameters (n, k, d, r). This result can be regarded as a generalization of the outer bound proposed by Prakash et al., which specifies the optimal tradeoff of exact-repair regenerating codes for the case of $d = k = n - 1$ and $r = 1$. It is verified that the proposed outer bound becomes more effective when k is large, r is small, or d ($\geq k$) is close to k.

The second outer bound is developed for the case of single node repair ($r = 1$). The bound is union of two independently derived sub-bounds. Each sub-bound has its own condition to be tighter than the other. One sub-bound can be regarded as an extension of the first outer bound for $r = 1$, and becomes more effective in high rates ($k/n > \frac{1}{2}$). The other sub-bound is derived based on the symmetric property of the storage nodes, and is tight in low rates ($k/n < \frac{1}{2}$).

keywords: regenerating codes, cooperative regenerating codes, repair bandwidth, exact repair model, distributed storage systems

student number: 2013-30256

Contents

List of Figures

Chapter 1

Introduction

In distributed storage systems (DSS), a data file is encoded into multiple fragments, and dispersed across a number of multiple storage nodes that are connected in a network. Failure of storage nodes can occur frequently in large-scale storage systems, due to the large number of storage nodes and unreliability of data disks. Redundancy must be introduced, in order to protect original data against node failures. *Erasure codes* are known as an efficient coding scheme for distributed storage, in that maximum reliability can be achieved for a given amount of storage overhead. In practice, Reed-Solomon (RS) codes, which are a kind of erasure codes, have been employed by several storage systems, including Facebook [1], Windows Azure Storage [2] and Hadoop Distributed File System (HDFS) [32], because of their high storage efficiency.

When a node fails, a new node (*newcomer*) that replaces it is generated by downloading information symbols from surviving nodes (*helpers*). The amount of data transmitted to repair a failed node is called *repair bandwidth*, and is an important performance metric to measure the network efficiency of distributed storage systems. Even though conventional erasure codes are beneficial in terms of storage efficiency, they have considerable disadvantage in network efficiency since they requires large repair bandwidth. In a naive repair scheme, the repair bandwidth of (n, k)-erasure codes is equal to the amount of information stored in k nodes, since the whole data file is required to be reconstructed even for the repair of a single failed node. Motivated by this problem, various codes for distributed storages with their own efficient repair scheme such as locally repairable codes [25–27], and repair-efficient RS codes [28, 29] have been introduced recently. In this dissertation, we focus on *Regenerating codes* that are a kind of erasure codes optimized in terms of minimizing repair bandwidth [3].

1.1 The Family of Regenerating Codes

Regenerating codes with the parameters of (n, k, d) consist of n storage nodes that store α data symbols each, and satisfy following two properties.

- *Data collection*: A data collector can obtain the original data of size B by downloading α symbols from each of k $(< n)$ arbitrary nodes.

- *Node repair*: A newcomer node replacing a failed node can be generated by downloading β ($\leq \alpha$) symbols from each of d ($\geq k$) arbitrary surviving nodes.

Note that the node repair process of regenerating codes is designed to be performed for a single node failure. However in large-scale storage systems, multiple nodes can fail at the same time. In order to recover r node failures by a regenerating code, $rd\beta$ symbols should be transmitted across the network. It is known that if the cooperation of multiple newcomers is allowed, the repair of multiple nodes can be performed with a total repair bandwidth smaller than $rd\beta$. Cooperative repairing implies that the exchange of a certain amount of information between multiple newcomers is allowed. The idea of cooperative repair is proposed in [4] only for the case of $d = n - r$ and generalized to have arbitrary d in [5]. In [6], the optimal storage-bandwidth (S-B) tradeoff of cooperative repair is derived, and the codes that achieve the tradeoff are called *cooperative regenerating codes*. If the number of simultaneously recovered node failures is r, two properties of (n, k, d, r)-cooperative regenerating codes comprising n nodes are as follows.

- *Data collection*: A data collector can obtain the original data of size B by downloading α symbols from each of k ($< n$) arbitrary nodes.

- *Cooperative node repair*: r newcomers can be generated through two phases. In Phase 1, each newcomer downloads β_1 ($\leq \alpha$) symbols from each of d ($\geq k$) arbitrary surviving nodes. In Phase 2, each newcomer downloads β_2 ($\leq d\beta_1$) symbols from each of the other $r-1$ newcomers.

In this case, repair bandwidth per one failed node γ equals $d\beta_1 + (r-1)\beta_2$. Note that $k\alpha \geq B$, $d\beta_1 \geq \beta_2$, and $\gamma \geq \alpha$ hold due to the information flow in the data collection and the node repair processes, and the reason why $d \geq k$ holds is that $d < k$ is contradictory since d nodes are enough for a data collector to obtain the original data file of size B by repairing $k-d$ nodes due to the node repair property. The maximum size of an original data file B that can be stored in a system using cooperative regenerating codes is determined by parameters α, β_1, and β_2 [6], and is expressed as

$$B \leq \sum_{h=1}^{g} l_h \min\left(\alpha, \left(d - \sum_{t=1}^{h-1} l_t\right)\beta_1 + (r - l_h)\beta_2\right), \tag{1.1}$$

where $\mathbf{l} = (l_1, ..., l_g)$ is an arbitrary vector such that each of its elements is an integer from 1 to r, and the sum of all elements is $\sum_{h=1}^{g} l_h = k$. Equation (1.1) is usually called the *cutset bound*, since it originated from the network coding results. In [7], the set of (α, γ) pairs satisfying (1.1) with equality is derived in a closed-form expression, and this forms the storage-bandwidth tradeoff of cooperative regenerating codes. There are two extreme points on the tradeoff curve that correspond to minimum values of α and γ, respectively. These two extreme points are called the *minimum storage cooperative regenerating (MSCR) points* and the *minimum bandwidth cooperative regenerating (MBCR) points*, respectively, and the points between them are called the *interior points*.

Cooperative regenerating codes are the generalized version of regenerating codes, and reduced to regenerating codes when $r = 1$. By substituting $r = 1$, $\beta_1 = \beta$, and $\beta_2 = 0$, (1.1) can be converted to

$$B \leq \sum_{i=1}^{k} \min(\alpha, (d - i + 1)\beta). \quad (1.2)$$

In the case of $r = 1$, the two extreme points are usually called the *minimum storage regenerating (MSR) points* and the *minimum bandwidth regenerating (MBR) points*, respectively.

1.2 The Exact Repair Model

The storage-bandwidth tradeoff of cooperative regenerating codes given by (1.1) assumes the *functional repair* model. In the functional repair model, information symbols of failed nodes are allowed to be replaced by different symbols if the newly formed n nodes including the newcomers can operate the functionalities of cooperative regenerating codes. However, the functional repair model is not usually employed for practical reasons. Firstly, huge network overhead is incurred since encoding and decoding rules should be updated every time the node repair occurs. Secondly, under the functional repair model, a systematic form of codes cannot be maintained. To solve these problems, the symbols of newcomer nodes need to be regenerated to be the exact replica of failed nodes, and this repair model is called the *exact repair* model.

Existing explicit designs of regenerating codes usually assume the exact repair model, and most of them are constructed at two extreme points, the MSCR and the MBCR points. For the case of single node repair ($r = 1$), constructions of exact-repair regenerating codes in two extreme points are shown to be possible in general (n, k, d) parameters. Explicit construction of exact repair codes in the MSR points considered in [8–10, 33] and the construction of the MBR codes is introduced in [9] and [11]. In the case of cooperative repairing ($r \geq 2$), the design method for the exact-repair MSCR codes with $d = k$ [12] and $k = 2,\ d \geq k$ [13] were proposed. In [14], it was proved that construction of the exact MSCR codes with $r = 2$ is possible for general (n, k, d) parameters, by showing that $(n, k, d, 1)$-MSR codes can always be converted to $(n, k, d-1, 2)$-MSCR codes, and vice versa. (In the functional repair case, it was proved that the node repair properties of $(n, k, d-r+1, r)$-MSCR points for $1 \leq r \leq d-k+1$ can be satisfied with the same code construction. the parameter r can be chosen opportunistically depending on the number of available helper nodes. The interested reader is referred to [30,31].) Shum et al. [15] first designed the exact-repair MBCR codes in the case of $n = d + r,\ d = k$, and it was generalized to $n = d + r,\ d \leq k$ case in [16]. Wang et al. [17] proposed the design method for the exact MBCR codes for general (n, k, d, r).

1.3 Existing Results on the S-B Tradeoff of Exact Repair Codes

The size of the original file B stored in exact-repair regenerating codes also satisfies the upper bound given in (1.1), because exact-repair codes are also a kind of functional-repair codes. As stated in the previous section, at the two extreme points, the MSCR and MBCR points, exact-repair cooperative regenerating codes can be built, and the condition of exact repair does not impose any penalty (except for the MSCR points with $r \geq 3$). However, in interior points, it is known that cooperative regenerating codes with (α, γ) parameters satisfying (1.1) with equality cannot be constructed with the exact repair model in general. The problem of specifying the storage-bandwidth tradeoff of exact-repair regenerating codes remains open except for the cases of two extreme points.

An example of the storage-bandwidth tradeoff curve is illustrated in Figure

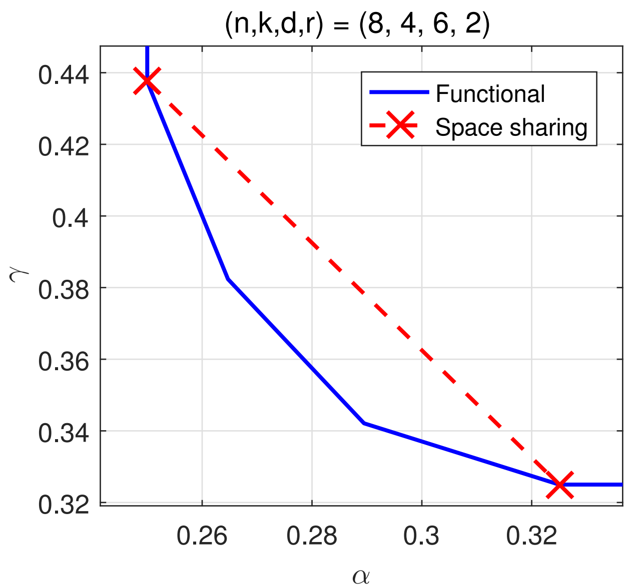

Figure 1.1: The storage α vs. bandwidth γ tradeoff of $(n, k, d, r) = (8, 6, 4, 2)$-cooperative regenerating code.

1.1. The curve with the blue and solid line is the set of (α, γ) points satisfying (1.1) with equality. Since any exact repair code is also a functional repair code, the curve can be viewed as an *outer* bound on the S-B tradeoff of the exact-repair codes. If an (α, γ) is below an outer bound, it is impossible to construct a regenerating code operating at the pair of (α, β). The curve with red and dashed line is usually called the *space-sharing* line, which is a line segment connecting the MSCR and MBCR points. A regenerating code operating at a point on the space-sharing line can be simply obtained by space-sharing scheme where an MSCR code is used in a fraction of original file and the remaining fraction is encoded by an MBCR code. The space-sharing line is an *inner* bound on the S-B tradeoff of exact repair codes. The optimal S-B tradeoff curve of exact repair regenerating codes must locate in between two curves.

In the case of single repair ($r = 1$), several works regarding the inner and outer bound on the S-B tradeoff of exact-repair regenerating codes have recently been reported. In [11], it was shown that interior points of the cutset bound (1.1) are impossible to achieve with exact-repair regenerating codes, except for a small region close to the MSR points. Tian derived the optimal S-B tradeoff of exact-repair regenerating codes in the case $(n, k, d) = (4, 3, 3)$ [18] by using the computer-aided proof (CAP) approach [36], and also extended this to the $(5, 4, 4)$ case [19]. It means the functional repair tradeoff (1.1) is not even achievable asymptotically with any exact repair codes.

Recently, in [20–23, 34, 35, 37–39], improved inner and outer bounds on the S-B tradeoff of exact repair regenerating codes for more general (n, k, d) parameters are proposed. Specifically, In [21], it was shown that if regenerating codes with parameters of $k = d = n - 1$ have linear encoding and decoding procedures, the properties of the regenerating codes can be expressed by some conditions of its parity check matrix. By exploiting those conditions, in [23], an outer bound on the S-B tradeoff of linear regenerating codes was proposed. The outer bound is identical to the inner bound proposed in [24] in the case of $k = d = n - 1$, which implies that the optimal S-B tradeoff of exact repair linear regenerating codes is characterized in that case.

1.4 Main Contribution

In this dissertation, we propose the two outer bounds on the storage-bandwidth tradeoff on the S-B tradeoff of linear regenerating codes. We propose generalized conditions of the dual codes of cooperative regenerating codes, and derive the outer bounds from them. The outer bounds suggest the (α, γ) pairs that no exact repair codes can achieve but only functional repair codes can.

The following theorem describes the first outer bound on the S-B tradeoff of linear cooperative regenerating codes which is mainly discussed in Chapter

2.

Theorem 1. Assume an (n, k, d, r)-linear cooperative regenerating code. If the exact repair model is used, then an upper bound of the file size B is expressed as

$$B \leq \frac{s-1}{s+1}(d+r)\alpha + \frac{2}{s(s+1)} \sum_{h=1}^{g} l_h \min\left(s\alpha, h\alpha, \Delta_h^{\mathbf{l}}\right), \tag{1.3}$$

where $\mathbf{l} = (l_1, \ \cdots, \ l_g)$ denotes a vector whose elements are integers satisfying $1 \leq l_h \leq r$ for $1 \leq h \leq g$ and the sum of its elements $\sum_{h=1}^{g} l_h$ equals k. s is an integer with $1 \leq s \leq g$ and $\Delta_h^{\mathbf{l}}$ is defined as

$$\Delta_h^{\mathbf{l}} := \begin{cases} (d-k+\sum_{t=1}^{h} l_t)\beta_1 + (r-l_h)\beta_2, & \text{if } s=1, \\ (d-k+\sum_{t=1}^{h} l_t)\beta_1 + c_h(r-l_h)\beta_2, & \text{if } 2 \leq s \leq g, \end{cases} \tag{1.4}$$

where $c_h = h - b_h$ and b_h is defined as

$$b_h := \max_{\substack{A \subset \{0,\cdots,h-1\}, \\ \sum_{t \in A} l_t \leq r - l_h}} |A|. \tag{1.5}$$

Proof. The proof of Theorem 1 will be discussed in Section 2.4. □

To the best of our knowledge there have been no results considering the outer bounds on the S-B tradeoff of the multiple repair case ($r \geq 2$). We derive the outer bound described in Theorem 1 by constructing a rank lower bound of parity check matrices of cooperative regenerating codes. This method is first used in [23], which proposed an outer bound on the S-B tradeoff for the case of $d = k = n - 1$ and $r = 1$.

The second outer bound, considered in Chapter 3, is developed for the case of single node repair ($r = 1$). We derived the two sub-bounds independently, where one is stated in Theorem 2, and the other is in Theorem 3.

Theorem 2 (Sub-bound 1)**.** Suppose a linear regenerating code under the exact repair model with parameters (n, k, d, α, β). Define $\tau = d - k + 1$ and $Q = \lfloor \frac{d+1}{\tau} \rfloor$. The size of data file B is bounded by

$$\begin{aligned} \frac{s(s+1)}{2}B \quad \leq \quad & \frac{s(s-1)}{2}k\alpha + sp\alpha + \frac{q(q-1)}{2}R(\alpha,\beta) \\ & + \frac{1}{2}(k-p-(q-1)\tau)(k-p+(q+1)\tau-1)\beta \\ & + \sum_{t=1}^{s-1} \min\left(t\tau\alpha, (q-1)R(\alpha,\beta) + \tau(k-p-(q-1)\tau)\beta\right) \end{aligned} \tag{1.6}$$

where s, q and p are arbitrary integers satisfying $1 \leq q \leq Q$, $0 \leq p \leq k-(q-1)\tau$, and $1 \leq s \leq d - (\tau - 1)q - p$. $R(\alpha, \beta)$ is defined as

$$R(\alpha,\beta) = \sum_{i=1}^{\tau} \min(\alpha, (\tau + i - 1)\beta). \tag{1.7}$$

Proof. The proof of this theorem will be discussed in Section 3.4.1. □

Theorem 3 (Sub-bound 2)**.** Suppose a linear regenerating code under the exact repair model with parameters (n, k, d, α, β). The size of data file B is bounded by

$$\sigma_B B \leq \sigma_\alpha \alpha + \sigma_\beta \beta \tag{1.8}$$

where σ_B, σ_α and σ_β is defined as

$$\sigma_B = (2\tau + k - p - 1)(s^2 + s - 2) + 2(k - p - 1),$$

$$\sigma_\alpha = 2(k - p - 1)p + (2\tau + k - p - 1)(sk + 2p)(s - 1),$$

$$\sigma_\beta = (2\tau + k - p - 1)(k - p)(k - p + 2(s - 1)\tau - 1),$$

and s, p are arbitrary integers satisfying $1 \leq s \leq k - p - 1$ and $2 \leq p \leq k$.

Proof. The proof of this theorem will be discussed in Section 3.4.2. □

Each sub-bound has its own condition to be tighter than the other. One sub-bound is more effective in high rates ($k/n > \frac{1}{2}$), but the other sub-bound becomes tighter when the code rate is low ($k/n < \frac{1}{2}$). In addition we shall verify that the two sub-bounds are asymptotically optimal in very high or low rates.

Chapter 2

An Outer Bound on the Storage-Bandwidth Trade-off of Cooperative Regenerating Codes

2.1 Conditions for Parity Check Matrices of Linear Cooperative Regenerating Codes

Suppose (n, k, d, r)-cooperative regenerating codes encode a $1 \times B$ message vector $\mathbf{m}$ into a $1 \times n\alpha$ codeword $\mathbf{c}$. The first α symbols of $\mathbf{c}$ correspond to the α symbols stored in the first node, the next α symbols are stored in the second node, and so on. Let $\mathbf{c}_1, \cdots, \mathbf{c}_n$ denote n code symbols of $\mathbf{c}$, each of which has length α, i.e.,

$$\mathbf{c} = [\mathbf{c}_1 \ \mathbf{c}_2 \ \cdots \ \mathbf{c}_n]. \tag{2.1}$$

We define *linear* cooperative regenerating codes as the cooperative regenerating codes where the encoding and decoding process performed in the data collection and node repair process is linear. A linear cooperative regenerating code with parameters of (n, k, d, r) can be regarded as a kind of $(n\alpha, B)$-linear codes, and there exist the $B \times n\alpha$ generator matrix $\mathbf{G}$ and the $(n\alpha - B) \times n\alpha$ parity check matrix $\mathbf{H}$ which satisfy

$$\mathbf{c} = \mathbf{m}\mathbf{G}, \quad \mathbf{G}\mathbf{H}^T = \mathbf{0}, \tag{2.2}$$

$$\mathsf{rank}(\mathbf{G}) = B, \quad \mathsf{rank}(\mathbf{H}) = n\alpha - B. \tag{2.3}$$

Linearity of data collection process follows from (2.2). The following sufficient conditions ensure that the node repair process of (n, k, d, r)-cooperative regenerating codes are linear.

- Each of the β_1 symbols sent from a helper to a newcomer in Phase 1 is a linear combination of the α symbols that the helper stores.
- Each of the β_2 symbols sent from newcomer m to newcomer i $(\neq m)$ in Phase 2 is a linear combination of $d\beta_1$ symbols that newcomer m received from its corresponding d helpers in Phase 1.
- Each of the α symbols a newcomer obtained is a linear combination of $\gamma = d\beta_1 + (r-1)\beta_2$ symbols that the newcomer received from the d helpers in Phase 1 and from the other $r - 1$ newcomers in Phase 2.

Lemma 1 gives some conditions that the generator and parity check matrices of linear cooperative regenerating codes must satisfy. In order to simplify the notation, we shall use the concept of thick columns and thick rows. Assume a matrix $\mathbf{M}$ consists of mn submatrices as

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} & \dots & \mathbf{M}_{1n} \\ \mathbf{M}_{21} & \mathbf{M}_{22} & \dots & \mathbf{M}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{m1} & \mathbf{M}_{m2} & \dots & \mathbf{M}_{mn} \end{bmatrix}.$$

For sets $I \subset [m]$ $(:= \{1, \cdots, m\})$ and $J \subset [n]$, let $\mathbf{M}_{IJ}$ be the matrix constructed by collecting submatrices whose indices i and j belong to I and J, respectively. Let $\mathbf{M}_{i[n]} = [\mathbf{M}_{i1} \ \cdots \ \mathbf{M}_{in}]$ be the ith thick row for $1 \leq i \leq m$ and $\mathbf{M}_{[m]j} = \left[\mathbf{M}_{1j}^T \ \cdots \ \mathbf{M}_{mj}^T\right]^T$ be the jth thick column for $1 \leq j \leq n$, where the superscript T denotes the transpose operator. Specifically, out of $n\alpha$ columns of $\mathbf{G}$ and $\mathbf{H}$, the first α columns form the first thick column, the next α columns correspond to the second thick column, and so on. In addition, we will use the following notations in the rest of the dissertation. $|A|$ denotes the cardinality of a set A. For some integers m and n, $[m]$ and $[m, n]$ denote the sets $\{1, 2, \cdots, n\}$ and $\{m, m+1, \cdots, n\}$, respectively. For a matrix $\mathbf{M}$, let $S(\mathbf{M})$ and $S(\mathbf{M}^T)$ be its column and row spaces. $\mathbf{I}_n$ denotes the $n \times n$ identity matrix and $\mathbf{0}$ denotes a zero matrix where every element is 0.

Lemma 1. Consider an (n, k, d, r)-linear cooperative regenerating code with $n = d + r$. The parity check matrix $\mathbf{H}$ satisfies the following two conditions, (i) and (ii).

(i) The rank of a matrix constructed by collecting $n - k$ arbitrary thick columns of $\mathbf{H}$ is $(n - k)\alpha$.

(ii) For any index set $R = \{i_1, \cdots, i_r\}$ $(i_1 < \cdots < i_r)$ which is a subset of $[n]$ and satisfies $|R| = r$, there exists an $r\alpha \times n\alpha$ matrix

$$\mathbf{H}_R = \begin{bmatrix} \mathbf{A}^R_{i_1 1} & \mathbf{A}^R_{i_1 2} & \dots & \mathbf{A}^R_{i_1 n} \\ \mathbf{A}^R_{i_2 1} & \mathbf{A}^R_{i_2 2} & \dots & \mathbf{A}^R_{i_2 n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^R_{i_r 1} & \mathbf{A}^R_{i_r 2} & \dots & \mathbf{A}^R_{i_r n} \end{bmatrix}$$

satisfying the following Condition (a)-(c), where $\mathbf{A}^R_{ij}$ is the $\alpha \times \alpha$ submatrix of $\mathbf{H}_R$ for $i \in R$ and $j \in [n]$.

(a) $\mathbf{A}^R_{RR} = \mathbf{I}_{r\alpha}$

(b) If $j \in D := [n] \setminus R$, then for all $i \in R$,

$$\mathbf{A}^R_{ij} = \mathbf{P}_{ij} + \mathbf{T}_{ij} = \mathbf{P}_{ij} + \sum_{m \in R \setminus \{i\}} \mathbf{T}_{ij,m}, \tag{2.4}$$

such that

$$S(\mathbf{P}^T_{ij}) \subset U^P_{ij}, \tag{2.5}$$

$$S(\mathbf{T}_{ij,m}^{T}) \subset U_{mj}^{P}, \tag{2.6}$$

and

$$\mathsf{rank}(\mathbf{T}_{iD,m}) \leq \beta_2, \tag{2.7}$$

where U_{ij}^{P} is a subspace whose dimension is smaller than or equal to β_1.

(c) $S(\mathbf{H}_R^T) \subset S(\mathbf{H}^T)$

Proof. See Subsection 2.1.1. □

Remark 1. For the case of $k = d$, Condition (i) follows from Condition (ii). Let $R_0 \subset [n]$ be a set of indices of $n - k$ $(= r)$ thick columns. Since $\mathbf{H}_{R_0}$ obtained from Condition (ii) contains $\mathbf{I}_{(n-k)\alpha}$ in the location of the $n - k$ thick columns by Condition (ii)-(a), the row space of the matrix that consists of the $n - k$ thick columns includes $S(\mathbf{I}_{(n-k)\alpha})$.

Remark 2. Lemma 1 can be regarded as a generalization of the conditions for linear regenerating codes stated in [21]. If $r = 1$, Lemma 1 is reduced to Proposition 2.1 of [21].

Remark 3. Condition (ii) originates from the cooperative node repair property. The index sets R and D used in Condition (ii) of Lemma 1 correspond to the sets of newcomers and helpers, respectively. $\mathbf{H}_R$ describes the case that the r

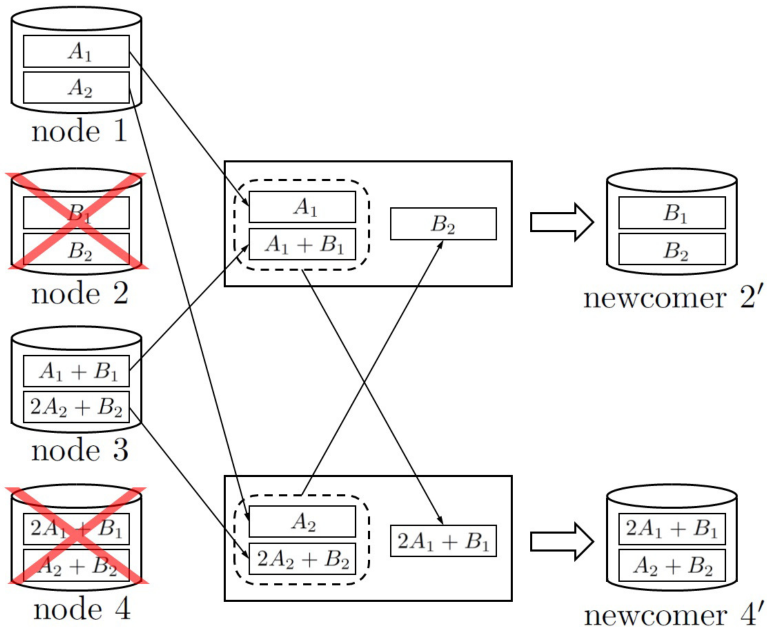

Figure 2.1: An example of node repair process of $(4, 2, 2, 2)$-cooperative regenerating codes with $(\alpha, \beta_1, \beta_2) = (2, 1, 1)$. This example is borrowed from [7].

nodes which belong to R are repaired with the help of d nodes which belong to D. $\mathbf{P}_{ij}$ and $\mathbf{T}_{ij}$ are related to Phase 1 and Phase 2, respectively. Each row of $\mathbf{H}_R$ corresponds to one of $r\alpha$ symbols of r newcomers. According to Condition (ii)-(c), every row of $\mathbf{H}_R$ must be orthogonal to all of $n\alpha$ codewords. It implies that each of $r\alpha$ symbols of r newcomers can be represented by a linear combination of $d\alpha$ symbols of d helpers. Refer to Subsection 2.1.1 for details.

We present a simple example of $(4, 2, 2, 2)$-cooperative regenerating codes for reader's better understanding. Consider an $(8, 4)$-linear code with generator matrix

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}. \tag{2.8}$$

The message vector $\mathbf{m} = (A_1, A_2, B_1, B_2)$ is encoded into the codeword $\mathbf{c} = (A_1, A_2, B_1, B_2, A_1{+}B_1, 2A_2{+}B_2, 2A_1{+}B_1, A_2{+}B_2)$. This code is a $(n, k, d, r) = (4, 2, 2, 2)$-cooperative regenerating code with $(\alpha, \beta_1, \beta_2) = (2, 1, 1)$, and we will verify that its parity check matrix $\mathbf{H}$ satisfies the conditions of Lemma 1. As illustrated in Figure 2.1, there are $n = 4$ nodes, each of which has $\alpha = 2$ symbols. The parity check matrix $\mathbf{H}$ is expressed as

$$\mathbf{H} = \begin{bmatrix} -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -2 & 0 & -1 & 0 & 1 & 0 & 0 \\ -2 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.9}$$

$\mathbf{H}$ has 4 thick columns, and it can be easily verified that Condition (i) is satisfied since every matrix made of $n - k = 2$ arbitrary thick columns has full rank.

In Figure 2.1, generation of two newcomers, newcomer $2'$ and $4'$, which replaces two failed nodes, node 2 and 4, are described. Condition (ii) of Lemma

1 gives

$$\mathbf{H}_{\{2,4\}} = \begin{bmatrix} \mathbf{A}_{21}^{\{2,4\}} & \mathbf{A}_{22}^{\{2,4\}} & \mathbf{A}_{23}^{\{2,4\}} & \mathbf{A}_{24}^{\{2,4\}} \\ \mathbf{A}_{41}^{\{2,4\}} & \mathbf{A}_{42}^{\{2,4\}} & \mathbf{A}_{43}^{\{2,4\}} & \mathbf{A}_{44}^{\{2,4\}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \tag{2.10}$$

which corresponds to the set of two indices of failed nodes $R = \{2, 4\}$ can be obtained from Condition (ii) of Lemma 1. It can be verified that each rows of $\mathbf{H}_{\{2,4\}}$ is orthogonal to every row of $\mathbf{G}$, and this implies the row space of $\mathbf{H}_{\{2,4\}}$ belongs to the row space of $\mathbf{H}$ (Condition (ii)-(c)).

To verify that $\mathbf{H}_{\{2,4\}}$ satisfies Condition (ii)-(b) of Lemma 1, consider the repair of the two symbols of newcomer $2'$. As shown in Figure 2.1, newcomer $2'$ uses three symbols A_1 (downloaded from node 1 in Phase 1), $A_1 + B_2$ (downloaded from node 3 in Phase 1), and B_2 (downloaded from newcomer $4'$ in Phase 1) to generate B_1 and B_2 as

$$B_1 = -A_1 + (A_1 + B_2) + 0(B_2), \tag{2.11}$$

$$B_2 = 0A_1 + 0(A_1 + B_2) + 1(B_2). \tag{2.12}$$

Although B_2 is downloaded from newcomer $4'$, it also originates from the symbols sent by node 1 and 3 to newcomer $4'$. We have

$$B_2 = (-2)A_2 + (2A_2 + B_2). \tag{2.13}$$

By combining (2.11)-(2.13), we have

$$\begin{bmatrix} B_1 & B_2 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} A_1 & A_2 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$

$$+ \begin{bmatrix} A_1 + B_2 & 2A_2 + B_2 \end{bmatrix} * \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \tag{2.14}$$

$$= \begin{bmatrix} B_1 & B_2 \end{bmatrix} \mathbf{A}_{22}^{\{2,4\}} + \begin{bmatrix} A_1 & A_2 \end{bmatrix} \mathbf{A}_{21}^{\{2,4\}}$$

$$+ \begin{bmatrix} A_1 + B_2 & 2A_2 + B_2 \end{bmatrix} \mathbf{A}_{23}^{\{2,4\}} = 0. \tag{2.15}$$

By classifying the symbols according to the phase in which the symbol was delivered, $\mathbf{A}_{21}^{\{2,4\}}$ and $\mathbf{A}_{23}^{\{2,4\}}$ can be decomposed into two components as

$$\begin{aligned} \mathbf{A}_{21}^{\{2,4\}} &= \mathbf{P}_{21} + \mathbf{T}_{21} \\ &= \mathbf{P}_{21} + \mathbf{T}_{21,4} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}, \end{aligned} \tag{2.16}$$

$$\begin{aligned}\mathbf{A}_{23}^{\{2,4\}} &= \mathbf{P}_{23} + \mathbf{T}_{23} \\ &= \mathbf{P}_{23} + \mathbf{T}_{23,4} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix}, \end{aligned} \tag{2.17}$$

where $\mathbf{P}_{21}$ and $\mathbf{P}_{23}$ are related to Phase 1, and $\mathbf{T}_{21}$ and $\mathbf{T}_{23}$ are related to Phase 2. Similarly, by considering the repair of newcomer $4'$, we can obtain $\mathbf{A}_{41}^{\{2,4\}}$ and $\mathbf{A}_{43}^{\{2,4\}}$ as

$$\begin{aligned}\mathbf{A}_{41}^{\{2,4\}} &= \mathbf{P}_{41} + \mathbf{T}_{41} \\ &= \mathbf{P}_{41} + \mathbf{T}_{41,2} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & +1 \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix}, \end{aligned} \tag{2.18}$$

$$\begin{aligned}\mathbf{A}_{43}^{\{2,4\}} &= \mathbf{P}_{43} + \mathbf{T}_{43} \\ &= \mathbf{P}_{43} + \mathbf{T}_{43,2} \\ &= \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned} \tag{2.19}$$

If subspaces U_{21}^P, U_{41}^P, U_{23}^P, and U_{43}^P are defined as $U_{21}^P = S(\mathbf{P}_{21}^T)$, $U_{41}^P = S(\mathbf{P}_{41}^T)$, $U_{23}^P = S(\mathbf{P}_{23}^T)$, and $U_{43}^P = S(\mathbf{P}_{43}^T)$, (2.5) is straightforward, and (2.6) is also satisfied since

$$S(\mathbf{T}_{21,4}^T) \subset S(\mathbf{P}_{41}^T) = U_{41}^P, \;\; S(\mathbf{T}_{41,2}^T) \subset S(\mathbf{P}_{21}^T) = U_{21}^P,$$

$$S(\mathbf{T}_{23,4}^T) \subset S(\mathbf{P}_{43}^T) = U_{43}^P, \quad \text{and } S(\mathbf{T}_{43,2}^T) \subset S(\mathbf{P}_{23}^T) = U_{23}^P.$$

Lastly, (2.7) can also be verified since

$$\mathsf{rank}([\mathbf{T}_{21,4}\ \mathbf{T}_{23,4}]) = 1 \leq \beta_2$$

$$\text{and } \mathsf{rank}([\mathbf{T}_{41,2}\ \mathbf{T}_{43,2}]) = 1 \leq \beta_2.$$

2.1.1 Proof of Lemma 1

Condition (i) and (ii) of Lemma 1 can be derived based on the data collection property and the node repair property of cooperative regenerating codes, respectively. Proof of (i) and (ii) is as follows.

Proof of (i). Let K be an arbitrary subset of $[n]$ whose cardinality is k, and define $\bar{K} = [n] \setminus K$. Suppose $(n\alpha, B)$-linear codes encode a $1 \times B$ message vector $\mathbf{m}$ into a $1 \times n\alpha$ codeword $\mathbf{c}$. Let $\mathbf{c}_K$ be the $k\alpha \times 1$ vector which contains k $1 \times \alpha$ code symbols that correspond to the index set K, and let $\mathbf{H}_K$ be the matrix formed by collecting k thick columns of $\mathbf{H}$ that correspond to the index set K. Similarly, $\mathbf{c}_{\bar{K}}$ and $\mathbf{H}_{\bar{K}}$ can be defined by using $\bar{K}$.

Assume that $\mathsf{rank}(\mathbf{H}_{\bar{K}})$ is smaller than $(n-k)\alpha$. Since the columns of $\mathbf{H}_{\bar{K}}$ are linearly dependent, there exists a nonzero vector $\mathbf{s}$ of length $(n-k)\alpha$ such that $\mathbf{s}\mathbf{H}_{\bar{K}}^T = \mathbf{0}$. Let $\mathbf{c}'$ be a row vector of length $n\alpha$ with $\mathbf{c}'_K = \mathbf{0}$ and $\mathbf{c}'_{\bar{K}} = \mathbf{s}$.

$\mathbf{c}'$ is a codeword because

$$\mathbf{c}'\mathbf{H}^T = \mathbf{0}\mathbf{H}_K^T + \mathbf{s}\mathbf{H}_{\bar{K}}^T = \mathbf{0}. \tag{2.20}$$

However, since $\mathbf{c}'_K = \mathbf{0}$, the message vector $\mathbf{m}$ cannot be repaired from $\mathbf{c}'_K$, and it contradicts the data collection property.

□

Proof of (ii). Condition (ii) of Lemma 1 states that for an index set $R = \{i_1, \cdots, i_r\}$ ($i_1 < \cdots < i_r$) such that $R \subset [n]$ and $|R| = r$, there exists an $r\alpha \times n\alpha$ matrix $\mathbf{H}_R$ with rn $\alpha \times \alpha$ submatrices $A^R_{i_1,1}, \cdots, A^R_{i_r,n}$ satisfying the following conditions (a)-(c)

(a) $\mathbf{A}^R_{RR} = \mathbf{I}_{r\alpha}$

(b) If $j \in D := [n] \setminus R$, then for all $i \in R$,

$$\mathbf{A}^R_{ij} = \mathbf{P}_{ij} + \mathbf{T}_{ij} = \mathbf{P}_{ij} + \sum_{m \in R \setminus \{i\}} \mathbf{T}_{ij,m}, \tag{2.21}$$

such that

$$S(\mathbf{P}^T_{ij}) \subset U^P_{ij}, \tag{2.22}$$

$$S(\mathbf{T}^T_{ij,m}) \subset U^P_{mj}, \tag{2.23}$$

and

$$\mathsf{rank}(\mathbf{T}_{iD,m}) \leq \beta_2, \tag{2.24}$$

where U^P_{ij} is a subspace whose dimension is smaller than or equal to β_1.

(c) $S(\mathbf{H}_R^T) \subset S(\mathbf{H}^T)$

For a fixed index set R, consider a node repair process where the r newcomers that corresponds to the index set R are cooperatively repaired with the help of d nodes that corresponds to $D = [n] \setminus R$. For $i \in R$ and $j \in D$, node j sends node i a $1 \times \beta_1$ vector $\mathbf{s}_{ij}$ whose elements are linear combinations of elements of $\mathbf{c}_j = (c_{(j-1)\alpha+1}, \ \cdots, \ c_{j\alpha})$ in the first phase of the node repair process. This encoding process can be specified by $\alpha \times \beta_1$ matrix Φ_{ij} as

$$\mathbf{s}_{ij} = \mathbf{c}_j \Phi_{ij} \qquad \text{for } i \in R \text{ and } j \in D. \tag{2.25}$$

In the next phase, node $m \in R \setminus \{i\}$ sends node i a $1 \times \beta_2$ vector $\mathbf{t}_{im}$ whose elements are linear combinations of the $d\beta_1$ elements of $\mathbf{s}_{mj_1}, \cdots, \mathbf{s}_{mj_d}$. i.e.,

$$\begin{aligned} \mathbf{t}_{im} &= [\mathbf{s}_{mj_1} \ \cdots \ \mathbf{s}_{mj_d}] \begin{bmatrix} \Psi_{im}^{j_1} \\ \vdots \\ \Psi_{im}^{j_d} \end{bmatrix}. & (2.26) \\ &= \sum_{j \in D} \mathbf{s}_{mj} \Psi_{im}^{j} & (2.27) \\ &= \sum_{j \in D} \mathbf{c}_j \Phi_{mj} \Psi_{im}^{j}, & (2.28) \end{aligned}$$

where Ψ_{im}^{j} is a $\beta_1 \times \beta_2$ matrix for $m \in R \setminus \{i\}$. According to the node repair property, $\mathbf{c}_i = (c_{(i-1)\alpha+1}, \ \cdots, \ c_{i\alpha})$ can be reconstructed by linearly combining

$d\beta_1$ symbols of $\mathbf{s}_{ji}s$ $(j \in D)$ and $(r-1)\beta_2$ symbols of $\mathbf{t}_{im}s$ $(i \in R \setminus \{i\})$ as

$$\begin{aligned} \mathbf{c}_i &= \sum_{j \in D} \mathbf{s}_{ij} \mathbf{L}_{ij}^{\beta_1} + \sum_{m \in R \setminus \{i\}} \mathbf{t}_{im} \mathbf{L}_{im}^{\beta_2} \\ &= \sum_{j \in D} \mathbf{c}_j \{ \Phi_{ij} \mathbf{L}_{ij}^{\beta_1} + \sum_{m \in R \setminus \{i\}} \Phi_{mj} \Psi_{im}^{j} \mathbf{L}_{im}^{\beta_2} \}, \end{aligned} \tag{2.29}$$

where $\mathbf{L}_{ij}^{\beta_1}$ and $\mathbf{L}_{im}^{\beta_2}$ are encoding matrices with size of $\beta_1 \times \alpha$ and $\beta_2 \times \alpha$, respectively for $j \in D$ and $m \in R \setminus \{i\}$.

For given indices of newcomer nodes $R = \{i_1,\ i_2,\ \cdots,\ i_r\}$ $(i_1 < i_2 < \cdots < i_r)$, define an $r\alpha \times n\alpha$ matrix $\mathbf{H}_R$ as

$$\mathbf{H}_R = \begin{bmatrix} \mathbf{A}_{i_1 1}^R & \dots & \mathbf{A}_{i_1 n}^R \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{i_r 1}^R & \dots & \mathbf{A}_{i_r n}^R \end{bmatrix} \tag{2.30}$$

where $\mathbf{A}_{ij}^R$ $(i \in R$ and $j \in [n])$ is an $\alpha \times \alpha$ submatrix of $\mathbf{H}_R$ and defined as follows. If $j \in R$,

$$\mathbf{A}_{ij}^R = \begin{cases} \mathbf{I}_\alpha & \text{if } j = i \\ \mathbf{0} & \text{if } j \in R \setminus \{i\}. \end{cases} \tag{2.31}$$

If $j \in D$, $\mathbf{A}_{ij}^R$ is the sum of two $\alpha \times \alpha$ matrices, $\mathbf{P}_{ij}$ and $\mathbf{T}_{ij}$, where

$$\mathbf{P}_{ij}^T = -\Phi_{ij} \mathbf{L}_{ij}^{\beta_1}, \tag{2.32}$$

$$\mathbf{T}_{ij}^T = -\sum_{m \in R \setminus \{i\}} \Phi_{mj} \Psi_{im}^{j} \mathbf{L}_{im}^{\beta_2} = \sum_{m \in R \setminus \{i\}} \mathbf{T}_{ij,m}^T. \tag{2.33}$$

By (2.31), it is clear that $\mathbf{H}_R$ satisfies Condition (a). Moreover, by (2.32) and (2.33), it can be easily verified that $\mathbf{H}_R$ satisfies (2.5) by letting $U_{ij}^P =$

$S\left((\mathbf{L}_{ij}^{\beta_1})^T\right)$ for all $i \in R$ and $j \in D$. Since

$$\begin{aligned}
&[\mathbf{T}_{ij_1,m} \ \cdots \ \mathbf{T}_{ij_d,m}] \\
&\quad = \quad -\left[\left(\Phi_{mj_1}\Psi_{im}^{j_1}\mathbf{L}_{im}^{\beta_2}\right)^T \ \cdots \ \left(\Phi_{mj_1}\Psi_{im}^{j_1}\mathbf{L}_{im}^{\beta_2}\right)^T\right] \\
&\quad = \quad -\left(\mathbf{L}_{im}^{\beta_2}\right)^T\left[\left(\Phi_{mj_1}\Psi_{im}^{j_1}\right)^T \ \cdots \ \left(\Phi_{mj_1}\Psi_{im}^{j_1}\right)^T\right],
\end{aligned} \tag{2.34}$$

$\mathsf{rank}([\mathbf{T}_{ij_1,m} \ \cdots \ \mathbf{T}_{ij_d,m}]) \leq \mathsf{rank}(\mathbf{L}_{im}^{\beta_2}) \leq \beta_2$ is satisfied and (2.7) can also be verified.

By using (2.31)-(2.33), (2.29) can be converted to

$$\mathbf{c}_i + \sum_{j \in D} \mathbf{c}_j (\mathbf{A}_{ij}^R)^T = \mathbf{c}[\mathbf{A}_{i1}^R \ \mathbf{A}_{i2}^R \ \cdots \ \mathbf{A}_{in}^R]^T = \mathbf{0}. \tag{2.35}$$

Since (2.35) must be satisfied for every codewords $\mathbf{c}$, $S(\mathbf{H}_R^T)$ must be orthogonal to $S(\mathbf{G}^T)$ and belong to $S(\mathbf{H}^T)$. This implies Condition (c).

□

2.2 An Alternative Proof of Functional Repair Cutset Bound

In this section, we prove the cutset bound of functional-repair cooperative regenerating codes (1.1) by using conditions given by Lemma 1.

Before the specific description, it should be emphasized that while we derive the cutset bound (1.1) and the proposed outer bound (in Section 2.2 and 2.4, respectively), we only consider the case of $n = d + r$. This is because every (n, k, d, r)-cooperative regenerating code with $n > d + r$ can be regarded as a $(d + r, k, d, r)$-cooperative regenerating code if some of $d + r$ nodes are chosen. Note that both the cutset bound (1.1) and the proposed outer bound (1.3) do not depend on the value of n, and yield the same outer bound under the same values of k, d, and r regardless of n. Therefore, we assume $n = d+r$ in the rest of the chapter.

The proof of (1.1) can be summarized in a few steps as follows.

(1) Choose an arbitrary vector $\mathbf{l} = (l_1, \cdots, l_g)$ whose elements are integers satisfying $1 \leq l_h \leq r$ for every $1 \leq h \leq g$ and $\sum_{h=1}^{g} l_h = k$.

(2) Construct the $n\alpha \times n\alpha$ matrix $\mathbf{H}_{repair}$ that corresponds to $\mathbf{l}$ by properly combining the rows of $\mathbf{H}_R$ given by Lemma 1.

(3) Find a lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$. Since $S(\mathbf{H}_{repair}^T) \subset S(\mathbf{H}^T)$, the lower bound is also a lower bound of $\mathsf{rank}(\mathbf{H})$.

(4) By using the fact that $B = n\alpha - \mathsf{rank}(\mathbf{H})$, an upper bound of B can be derived.

We will use the technique that uses the lower bounds of $\mathsf{rank}(\mathbf{H})$ to find the upper bounds of B not only in the proof of (1.1), but also in the proof of the proposed outer bound (Theorem 1) described in Section 2.4. In addition, we will reuse the matrix $\mathbf{H}_{repair}$ constructed in this section in Section 2.4.

2.2.1 Construction of $\mathbf{H}_{repair}$

Consider a vector $\mathbf{l} = (l_1, \ \cdots, \ l_g)$ such that $1 \leq l_h \leq r$ for $h \in [g]$ and $\sum_{h=1}^{g} l_h = k$. By adding an element $l_0 = n - k$ to the left side of $\mathbf{l}$, if it is extended to $\mathbf{l}^* = (l_0, \ l_1, \ \cdots, \ l_g)$, $\sum_{h=0}^{g} l_h = n$ is satisfied. Define sets $R_1, R_2, \cdots, R_g$ as

$$R_h = R_h^{'} \cup N_h \quad \text{for } 1 \leq h \leq g, \tag{2.36}$$

where

$$R_h^{'} := \left[\sum_{t=0}^{h-1} l_t + 1, \sum_{t=0}^{h} l_t \right] \tag{2.37}$$

and N_h is defined to satisfy

$$N_h \subset \left[\sum_{t=0}^{h-1} l_t \right] \text{ such that } |N_h| = r - l_h. \tag{2.38}$$

By Condition (ii) of Lemma 1, for every $1 \leq h \leq g$,

$$\mathbf{H}_{R_h} = \begin{bmatrix} \mathbf{A}_{i_1^h 1}^{R_h} & \mathbf{A}_{i_1^h 2}^{R_h} & \cdots & \mathbf{A}_{i_1^h n}^{R_h} \\ \mathbf{A}_{i_2^h 1}^{R_h} & \mathbf{A}_{i_2^h 2}^{R_h} & \cdots & \mathbf{A}_{i_2^h n}^{R_h} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{i_r^h 1}^{R_h} & \mathbf{A}_{i_r^h 2}^{R_h} & \cdots & \mathbf{A}_{i_r^h n}^{R_h} \end{bmatrix}$$

can be obtained, which corresponds to $R_h := \{i_1^h, \cdots i_r^h\}$ ($i_1^h < \cdots < i_r^h$) defined in (2.36), where $\mathbf{A}_{ij}^{R_h}$ ($i \in R_h, j \in [n]$) is an $\alpha \times \alpha$ submatrix of $\mathbf{H}_{R_h}$. Note that $N_h = \{i_1^h, \cdots, i_{r-l_h}^h\}$ and $R_h^{'} = \{i_{r-l_h+1}^h, \cdots, i_r^h\}$, since every element in N_h is smaller than any element in $R_h^{'}$. By collecting the last l_h thick rows out of r thick rows of $\mathbf{H}_{R_h}$, an $l_h\alpha \times n\alpha$ matrix $\mathbf{A}_{R_h^{'}[n]}^{R_h}$ can be obtained. By combining them vertically, we can obtain an $n\alpha \times n\alpha$ matrix

$$\mathbf{H}_{repair} = \begin{bmatrix} \tilde{\mathbf{H}}^{\dagger}\mathbf{H} \\ \mathbf{A}_{R_1^{'}[n]}^{R_1} \\ \vdots \\ \mathbf{A}_{R_g^{'}[n]}^{R_g} \end{bmatrix}, \tag{2.39}$$

where $\tilde{\mathbf{H}}$ is the $(n\alpha - B) \times (n-k)\alpha$ matrix constructed by collecting the first $n-k$ thick columns of the parity check matrix $\mathbf{H}$, and $\tilde{\mathbf{H}}^{\dagger}$ ($:= (\tilde{\mathbf{H}}^T\tilde{\mathbf{H}})^{-1}\tilde{\mathbf{H}}^T$) is its left inverse such that $\tilde{\mathbf{H}}^{\dagger}\tilde{\mathbf{H}} = \mathbf{I}_{(n-k)\alpha}$. According to Condition (i) of Lemma 1, $\tilde{\mathbf{H}}$ must have full column rank, and its left inverse always exists. Note that the first $(n-k)\alpha$ columns of $\tilde{\mathbf{H}}^{\dagger}\mathbf{H}$ are equal to $\tilde{\mathbf{H}}^{\dagger}\tilde{\mathbf{H}} = \mathbf{I}_{(n-k)\alpha}$.

Let $n\alpha$ rows of $\mathbf{H}_{repair}$ be grouped in the pattern of $l_0\alpha, \cdots, l_g\alpha$. By grouping its columns in the same pattern, $\mathbf{H}_{repair}$ has $(g+1)^2$ submatrices as

$$\mathbf{H}_{repair} = \begin{bmatrix} \mathbf{H}_{0,0} & \mathbf{H}_{0,1} & \dots & \mathbf{H}_{0,g} \\ \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \dots & \mathbf{H}_{1,g} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{g,0} & \mathbf{H}_{g,1} & \dots & \mathbf{H}_{g,g} \end{bmatrix}, \tag{2.40}$$

where $\mathbf{H}_{h,t}$ denotes the submatrix with the size of $l_h\alpha \times l_t\alpha$ contained commonly in the hth thick row and the tth thick column of $\mathbf{H}_{repair}$. Note that

$$[\mathbf{H}_{0,0} \cdots \mathbf{H}_{0,g}] = \tilde{\mathbf{H}}^{\dagger}\mathbf{H} \tag{2.41}$$

and

$$\mathbf{H}_{h,t} = \mathbf{A}^{R_h}_{R'_h R'_t}, \text{ for } 1 \leq h \leq g. \tag{2.42}$$

Specifically, $\mathbf{H}_{h,h}$ is the hth diagonal submatrix with the size of $l_h\alpha \times l_h\alpha$. We have already mentioned that $\mathbf{H}_{0,0} = \tilde{\mathbf{H}}^{\dagger}\tilde{\mathbf{H}} = \mathbf{I}_{(n-k)\alpha}$. In addition, it is easily verified that

$$\mathbf{H}_{h,h} = \mathbf{I}_{l_h\alpha} \qquad \text{for } 1 \leq h \leq g, \tag{2.43}$$

according to Condition (ii)-(a) of Lemma 1.

Figure 2.2 illustrates an example of constructing $\mathbf{H}_{repair}$ that corresponds to $\mathbf{l}^* = (l_0, l_1, l_2, l_3, l_4) = (6, 1, 2, 3, 1)$. The parameters of the cooperative regenerating code are set to be $(n, k, d, r) = (13, 7, 8, 5)$. The left side of Figure 2.2

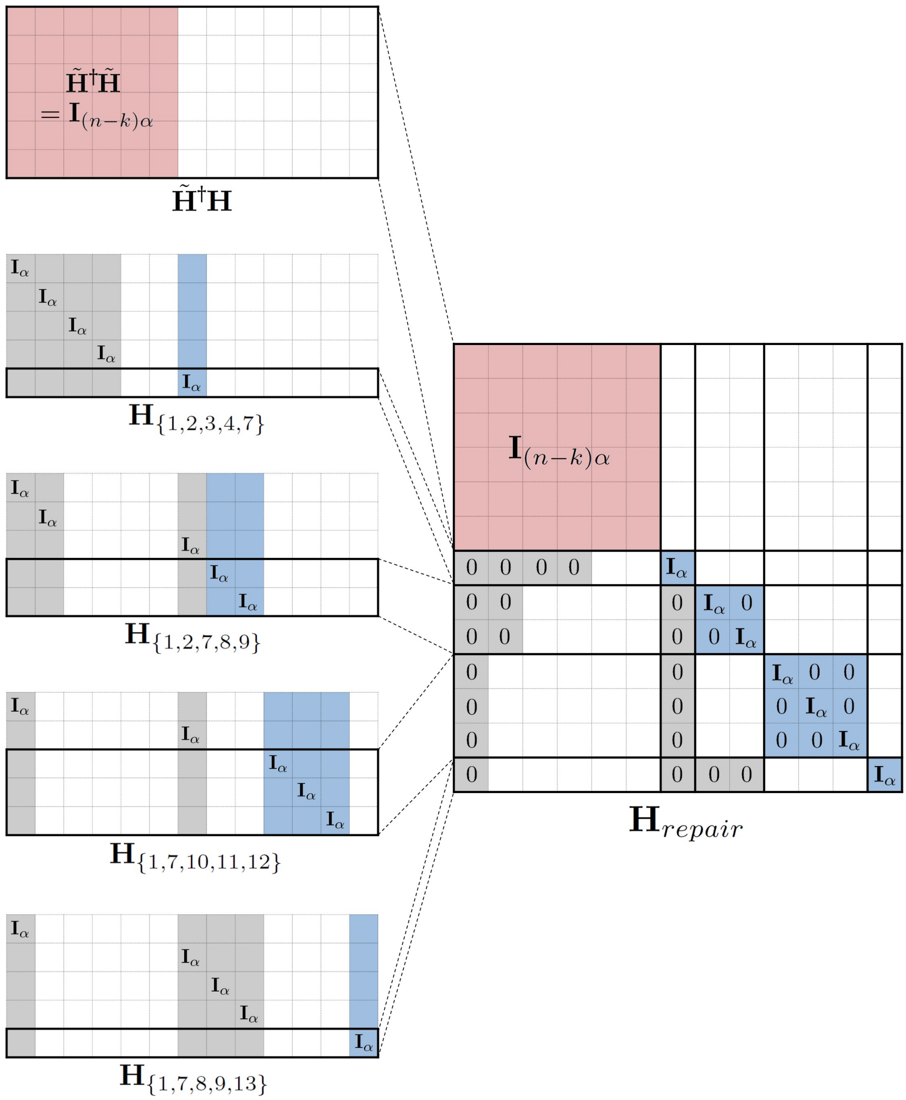

Figure 2.2: An example of construction of $\mathbf{H}_{repair}$. The code parameters $(n, k, d, r) = (13, 7, 8, 5)$ and $\mathbf{l}^* = (l_0, l_1, l_2, l_3, l_4) = (6, 1, 2, 3, 1)$ are used.

expresses $\mathbf{H}_{R_1}, \cdots, \mathbf{H}_{R_4}$, and $\tilde{\mathbf{H}}^{\dagger}\mathbf{H}$. The smallest squares denote $\alpha \times \alpha$ components. For $1 \leq h \leq 4$, the $r^2 = 25$ shaded squares correspond to $\mathbf{A}^{R_h}_{R_h R_h}$ which equals $\mathbf{I}_{r\alpha}$ by Condition (ii)-(a) of Lemma 1. The rectangle enclosed by bold lines corresponds to $\mathbf{A}^{R_h}_{R_h^{'}[n]}$, which participates in the construction of $\mathbf{H}_{repair}$. It can be verified that the lower part of the blue-shaded squares becomes $\mathbf{H}_{h,h}$, the hth diagonal submatrix of $\mathbf{H}_{repair}$, which is still an identity matrix, $\mathbf{I}_{l_h \alpha}$. The right side of Figure 2.2 illustrates $\mathbf{H}_{repair}$, and its $(g+1)^2 = 5^2$ submatrices are emphasized by bold lines.

Remark 4. As stated in (2.36), R_h is the union of two disjoint subsets $R_h^{'}$ and N_h for $1 \leq h \leq g$. $R_h^{'}$s are given deterministically by (2.37). However, N_hs are not deterministic, and there can be various forms of N_h that satisfy $N_h \subset [\sum_{t=0}^{h-1} l_t]$ and $|N_h| = r - l_h$. The gray-shaded region of Figure 2.2 corresponds to $\mathbf{A}^{R_h}_{R_h^{'} N_h}$ for $1 \leq h \leq g$. $\mathbf{A}^{R_h}_{R_h^{'} N_h}$ becomes a zero matrix because it is a part of the lower triangular part of $\mathbf{A}^{R_h}_{R_h R_h}$ $(= \mathbf{I}_{r\alpha})$. Hence, the part of $\mathbf{A}^{R_h}_{R_h^{'}[\sum_{t=0}^{h-1} l_t]}$ that is a zero matrix can be controlled by properly selecting N_h. Though the position of zero matrices is not important in this section, it will play an important role in the proof of Theorem 1 in Section 2.4.

2.2.2 Lower Bounds of $\mathsf{rank}(\mathbf{H}_{repair})$

$\mathbf{H}_{repair}$ has $g+1$ thick columns, $\mathbf{H}_{[0,g],0}, \cdots, \mathbf{H}_{[0,g],g}$. Define $\delta_0, \delta_1, \cdots, \delta_g$ as

$$\delta_0 = \mathsf{rank}(\mathbf{H}_{[0,g],0}), \tag{2.44}$$

and

$$\delta_h = \mathsf{rank}([\mathbf{H}_{[0,g],0} \cdots \mathbf{H}_{[0,g],h}]) - \mathsf{rank}([\mathbf{H}_{[0,g],0} \cdots \mathbf{H}_{[0,g],h-1}]), \quad \text{if } 1 \le h \le g. \tag{2.45}$$

Therefore, δ_h indicates the increment of rank after the hth thick column is added.

Since $\mathbf{H}_{0,0} = \mathbf{I}_{(n-k)\alpha}$, $n-k$ columns of $\mathbf{H}_{[0,g],0}$ are linearly independent. This implies

$$\delta_0 = \mathsf{rank}(\mathbf{H}_{0,0}) = (n-k)\alpha \tag{2.46}$$

If $h \ge 1$, δ_h is lower bounded by

$$\delta_h \ge \mathsf{rank}([\mathbf{H}_{h,0} \ \cdots \ \mathbf{H}_{h,h}]) - \mathsf{rank}([\mathbf{H}_{h,0} \ \cdots \ \mathbf{H}_{h,h-1}]) \tag{2.47}$$

$$\ge \mathsf{rank}(\mathbf{H}_{h,h}) - \mathsf{rank}([\mathbf{H}_{h,0} \ \cdots \ \mathbf{H}_{h,h-1}]) \tag{2.48}$$

$$= l_h\alpha - \mathsf{rank}\left(\mathbf{A}^{R_h}_{R'_h L_h}\right), \tag{2.49}$$

where L_h is defined as

$$L_h := \left[\sum_{t=0}^{h-1} l_t\right] \setminus N_h, \tag{2.50}$$

and (2.49) follows from the relations $\mathbf{H}_{h,h} = \mathbf{I}_{l_h\alpha}$ and $\mathbf{A}^{R_h}_{R'_h N_h} = \mathbf{0}$ (See Remark 4).

Because of the fact that $L_h \subset [n] \setminus R_h$, every $\alpha \times \alpha$ submatrix contained in $\mathbf{A}^{R_h}_{R'_h L_h}$ can be expressed as

$$\mathbf{A}^{R_h}_{ij} = \mathbf{P}_{ij} + \mathbf{T}_{ij} = \mathbf{P}_{ij} + \sum_{m \in R_h \setminus \{i\}} \mathbf{T}_{ij,m} \tag{2.51}$$

by Condition (ii)-(b) of Lemma 1 where definitions of $\mathbf{P}_{ij}$, $\mathbf{T}_{ij}$ and $\mathbf{T}_{ij,m}$ are given in Lemma 1. Define matrices $\mathbf{P}'_{ij}$ and $\mathbf{T}'_{ij}$ as

$$\mathbf{P}'_{ij} := \mathbf{P}_{ij} + \sum_{m \in R'_h \setminus \{i\}} \mathbf{T}_{ij,m}, \quad \mathbf{T}'_{ij} := \sum_{m \in N_h} \mathbf{T}_{ij,m}. \tag{2.52}$$

for $i \in R'_h$ and $j \in L_h$. Therefore, $\mathbf{A}^{R_h}_{ij} = \mathbf{P}'_{ij} + \mathbf{T}'_{ij}$ is satisfied for every $i \in R'_h$ and $j \in L_h$ and we have

$$\mathbf{A}^{R_h}_{R'_h L_h} = \mathbf{P}'_{R'_h L_h} + \mathbf{T}'_{R'_h L_h}. \tag{2.53}$$

By using Condition (ii)-(b) of Lemma 1, $\mathsf{rank}(\mathbf{P}'_{R'_h L_h})$ and $\mathsf{rank}(\mathbf{T}'_{R'_h L_h})$ are upper bounded by

$$\begin{aligned}
\mathsf{rank}(\mathbf{P}'_{R'_h L_h}) &\leq \sum_{j\in L_h} \mathsf{rank}(\mathbf{P}'_{R'_h j}) &(2.54)\\
&\leq \sum_{j\in L_h} \mathsf{dim}(\bigoplus_{l\in R'_h} U_{lj}) &(2.55)\\
&\leq \sum_{j\in L_h} \sum_{l\in R'_h} \mathsf{dim}(U_{lj}) &(2.56)\\
&\leq |R'_h||L_h|\beta_1 &(2.57)\\
&= |R'_h| \left(\sum_{t=0}^{h-1} l_t - |N_h| \right) \beta_1 &(2.58)\\
&= l_h(d-k+\sum_{t=1}^{h} l_t)\beta_1 &(2.59)
\end{aligned}$$

and

$$\begin{aligned}
\mathsf{rank}(\mathbf{T}'_{R'_h L_h}) &\leq \sum_{i\in R'_h} \mathsf{rank}(\mathbf{T}'_{iL_h}) &(2.60)\\
&= \sum_{i\in R'_h} \sum_{m\in N_h} \mathsf{rank}(\mathbf{T}_{iL_h,m}) &(2.61)\\
&\leq \sum_{i\in R'_h} \sum_{m\in N_h} \mathsf{rank}(\mathbf{T}_{i([n]\setminus R_h),m}) &(2.62)\\
&\leq |R'_h||N_h|\beta_2 &(2.63)\\
&= l_h(r-l_h)\beta_2, &(2.64)
\end{aligned}$$

respectively, where the operator $\bigoplus$ denotes the sum of subspaces, and (2.55) follows from

$$\begin{aligned} S((\mathbf{P}'_{ij})^T) &= S(\mathbf{P}_{ij}^T + \sum_{m \in R'_h \setminus \{i\}} \mathbf{T}_{ij,m}^T) &(2.65)\\ &\subset S(\mathbf{P}_{ij}^T) \oplus \bigoplus_{m \in R'_h \setminus \{i\}} S(\mathbf{T}_{ij,m}^T) &(2.66)\\ &\subset U_{ij} \oplus \bigoplus_{m \in R'_h \setminus \{i\}} U_{mj} &(2.67)\\ &= \bigoplus_{l \in R'_h} U_{lj} &(2.68)\end{aligned}$$

$$\text{for every } i \in R'_h$$

By applying (2.59) and (2.64) to (2.49), we have

$$\begin{aligned} \delta_h &\geq l_h\alpha - \mathsf{rank}\left(\mathbf{A}^{R_h}_{R'_h L_h}\right) &(2.69)\\ &\geq l_h\alpha - \left(\mathsf{rank}\left(\mathbf{P}'_{R'_h L_h}\right) + \mathsf{rank}\left(\mathbf{T}'_{R'_h L_h}\right)\right) &(2.70)\\ &\geq l_h\alpha - \left(l_h(d-k+\sum_{t=1}^{h} l_t)\beta_1 + l_h(r-l_h)\beta_2\right). &(2.71)\end{aligned}$$

In addition, since δ_h, a rank increment, must be positive, δ_h is lower bounded by

$$\delta_h \geq l_h \min\left(0, \alpha - \left((d-k+\sum_{t=1}^{h} l_t)\beta_1 + (r-l_h)\beta_2\right)\right). \qquad (2.72)$$

Since $\mathsf{rank}(\mathbf{H}_{repair}) = \sum_{h=0}^{g} \delta_h$, $\mathsf{rank}(\mathbf{H}_{repair})$ is lower bounded by

$$\begin{aligned} \mathsf{rank}(\mathbf{H}_{repair}) &= \sum_{h=0}^{g} \delta_h \\ &\geq (n-k)\alpha + \sum_{h=1}^{g} l_h \min\left(0, \alpha - \left((d-k+\sum_{t=1}^{h} l_t)\beta_1 + (r-l_h)\beta_2\right)\right). \qquad (2.73)\end{aligned}$$

2.2.3 Upper Bounds of B

Since the rows of $\mathbf{H}_{repair}$ originate from $\mathbf{H}_{R_0}, \cdots, \mathbf{H}_{R_g}$, by Condition (ii)-(c) of Lemma 1, $S(\mathbf{H}^T_{repair})$ must be a subspace of $S(\mathbf{H}^T)$. Hence, the lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ is also a lower bound of $\mathsf{rank}(\mathbf{H})$. By using $B = n\alpha - \mathsf{rank}(\mathbf{H})$, an upper bounds of B can be derived as

$$\begin{aligned} B &= n\alpha - \mathsf{rank}(\mathbf{H}) && (2.74)\\ &\leq n\alpha - \mathsf{rank}(\mathbf{H}_{repair}) && (2.75)\\ &= (n-k+\sum_{h=1}^{g} l_h)\alpha - \mathsf{rank}(\mathbf{H}_{repair}) && (2.76)\\ &= \sum_{h=1}^{g} l_h \min\big(\alpha, (d-k+\sum_{t=1}^{h} l_t)\beta_1 + (r-l_h)\beta_2\big) && (2.77)\\ &= \sum_{h=1}^{g} l_h \min\big(\alpha, (d-\sum_{t=h+1}^{g} l_t)\beta_1 + (r-l_h)\beta_2\big), && (2.78) \end{aligned}$$

By using $(l'_g,\ l'_{g-1},\ \cdots,\ l'_1) := (l_1,\ l_2,\ \cdots,\ l_g)$, (1.1) is derived as

$$B \leq \sum_{h=1}^{g} l'_h \min\big[\alpha, (d-\sum_{t=1}^{h-1} l'_t)\beta_1 + (r-l'_h)\beta_2\big]. \tag{2.79}$$

2.3 Block Matrices with Full-Rank Diagonal Blocks

Our objective is to find a tight upper bound on the file size B stored in a given linear cooperative regenerating code with parameters $(n, k, d, r, \alpha, \beta)$. In Section 2.2, we tried to find a lower bound of $\mathbf{H}_{repair}$, and converted it into an upper bound of B based on the relation that

$$\begin{aligned} B &= n\alpha - \mathsf{rank}(\mathbf{H}) & (2.80) \\ &\leq n\alpha - \mathsf{rank}(\mathbf{H}_{repair}). & (2.81) \end{aligned}$$

where (2.81) holds since the matrix $\mathbf{H}_{repair}$ is constructed to satisfy $S(\mathbf{H}_{repair}^T) \subset S(\mathbf{H}^T)$.

$\mathbf{H}_{repair}$ is a block matrix which has g^2 submatrices as

$$\mathbf{H}_{repair} = \begin{bmatrix} \mathbf{H}_{0,0} & \mathbf{H}_{0,1} & \ldots & \mathbf{H}_{0,g} \\ \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \ldots & \mathbf{H}_{1,g} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{g,0} & \mathbf{H}_{g,1} & \ldots & \mathbf{H}_{g,g} \end{bmatrix}, \quad (2.82)$$

In order to find a tighter lower bound on the $\mathsf{rank}(\mathbf{H}_{repair})$, the property of $\mathbf{H}_{repair}$ we shall focus on is that the every diagonal submatrices of $\mathbf{H}_{repair}$ is nonsingular as

$$\mathbf{H}_{h,h} = \mathbf{I}_{l_h \alpha} \qquad \text{for } 0 \leq h \leq g. \quad (2.83)$$

In this section, we shall derive the general properties of the block matrices with full-rank diagonal blocks. The properties and definitions provided in this section will be used not only in the derivation of Theorem 1 in this chapter, but also in the derivation of the second outer bound (Theorem 2 and 3) which will be discussed in the next chapter.

2.3.1 Definitions

In this subsection, we present several definitions of the block matrices, which will be used in common for the derivations of the outer bounds on the regenerating codes in this dissertation.

Suppose a block matrix $\mathbf{M}$ is broken into n^2 submatrices as

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{1,1} & \mathbf{M}_{1,2} & \dots & \mathbf{M}_{1,n} \\ \mathbf{M}_{2,1} & \mathbf{M}_{2,2} & \dots & \mathbf{M}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{n,1} & \mathbf{M}_{n,2} & \dots & \mathbf{M}_{n,n} \end{bmatrix} = [\mathbf{M}_1 \cdots \mathbf{M}_n].$$

where $\mathbf{M}_1, \cdots, \mathbf{M}_n$ are n thick columns of $\mathbf{M}$. The number of columns (rows) in each thick column (thick row) does not have to be identical.

We will define $n-1$ matrices $\mathbf{M}^{(1)}, \cdots, \mathbf{M}^{(n-1)}$, which originate from $\mathbf{M}$. First of all, define $\mathbf{M}^{(1)} := \mathbf{M}$. $\mathbf{M}^{(1)}$ has n^2 submatrices and n thick columns as $\mathbf{M}$ does. Let $\mathbf{M}^{(1)}_{i,j}$ be a submatrix of $\mathbf{M}^{(1)}$ for $i, j \in [n]$ and $\mathbf{M}^{(1)}_i$ be the ith thick column of $\mathbf{M}^{(1)}$ for $i \in [n]$ such that

$$\mathbf{M}^{(1)} = \begin{bmatrix} \mathbf{M}^{(1)}_{1,1} & \mathbf{M}^{(1)}_{1,2} & \dots & \mathbf{M}^{(1)}_{1,n} \\ \mathbf{M}^{(1)}_{2,1} & \mathbf{M}^{(1)}_{2,2} & \dots & \mathbf{M}^{(1)}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}^{(1)}_{n,1} & \mathbf{M}^{(1)}_{n,2} & \dots & \mathbf{M}^{(1)}_{n,n} \end{bmatrix} = \left[\mathbf{M}^{(1)}_1 \cdots \mathbf{M}^{(1)}_n\right].$$

For $2 \leq s \leq n-1$, let $\mathbf{M}^{(s)}$ denote a block matrix with $n(n-s+1)$ submatrices $\mathbf{M}^{(s)}_{1,s}, \cdots, \mathbf{M}^{(s)}_{n,n}$ and $n-s+1$ thick columns $\mathbf{M}^{(s)}_{s}, \cdots, \mathbf{M}^{(s)}_{n}$ such that

$$\mathbf{M}^{(s)} = \begin{bmatrix} \mathbf{M}^{(s)}_{1,s} & \mathbf{M}^{(s)}_{1,s+1} & \dots & \mathbf{M}^{(s)}_{1,n} \\ \mathbf{M}^{(s)}_{2,s} & \mathbf{M}^{(s)}_{2,s+1} & \dots & \mathbf{M}^{(s)}_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}^{(s)}_{n,s} & \mathbf{M}^{(s)}_{n,s+1} & \dots & \mathbf{M}^{(s)}_{n,n} \end{bmatrix} = \left[\mathbf{M}^{(s)}_{s} \cdots \mathbf{M}^{(s)}_{n}\right].$$

$\mathbf{M}^{(2)}, \mathbf{M}^{(3)}, \cdots, \mathbf{M}^{(n-1)}$ are defined in a recursive manner as follows. $\mathbf{M}^{(s-1)}$ has $n-s+2$ thick columns $\mathbf{M}^{(s-1)}_{s-1}, \cdots, \mathbf{M}^{(s-1)}_{n}$. For $s \leq i \leq n$, define

$$V^{(s)}_{i} := S(\mathbf{M}^{(s-1)}_{i}) \cap S([\mathbf{M}^{(s-1)}_{s-1} \cdots \mathbf{M}^{(s-1)}_{i-1}]).$$

and let $\mathbf{M}^{(s)}_{i}$ be the matrix with $\mathsf{dim}(V^{(s)}_{i})$ columns that are the basis vectors of $V^{(s)}_{i}$. The pattern of partitioning the thick rows of $\mathbf{M}^{(s)}$ is assumed to be the same with the partitioning pattern of thick rows of $\mathbf{M}^{(1)}$ for every $2 \leq s \leq n-1$.

Let us define additional notations related to the rank of submatrices of $\mathbf{M}^{(s)}$ for $1 \leq s \leq n-1$ as follows.

- $\delta^{(s)}_{i}(\mathbf{M}) := \begin{cases} \mathsf{rank}(\mathbf{M}^{(s)}_{s}), & \text{if } i = s \\ \mathsf{rank}\left(\left[\mathbf{M}^{(s)}_{s} \cdots \mathbf{M}^{(s)}_{i}\right]\right) - \mathsf{rank}\left(\left[\mathbf{M}^{(s)}_{s} \cdots \mathbf{M}^{(s)}_{i-1}\right]\right), & \text{if } s+1 \leq i \leq n \end{cases}$
- $\rho^{(s)}(\mathbf{M}) := \mathsf{rank}(\mathbf{M}^{(s)}) = \sum_{i=s}^{n} \delta^{(s)}_{i}(\mathbf{M})$

- $T_i^{(s)}(\mathbf{M}) := \begin{cases} 0, & \text{if } i = s, \\ \sum_{j=s}^{i-1} \mathsf{rank}(\mathbf{M}_{i,j}^{(s)}), & \text{if } s+1 \le i \le n. \end{cases}$

- $\bar{T}_i^{(s)}(\mathbf{M}) := \begin{cases} 0, & \text{if } i = s, \\ \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-1}^{(s)}\right]\right), & \text{if } s+1 \le i \le n. \end{cases}$

2.3.2 Properties of Block Matrices with Full-Rank Diagonal Blocks

In this subsection, we consider block matrices of which the every diagonal block has full column rank. Specifically, suppose that $\mathbf{M}$ satisfies the following two conditions (i) and (ii):

(i) the columns of $\mathbf{M}_i$ are linearly independent for every $1 \le i \le n$

(ii) $\mathsf{rank}(\mathbf{M}_{i,i}) = \mathsf{rank}(\mathbf{M}_i)$ for every $1 \le i \le n$,

The following proposition states that Conditions (i) and (ii) are inherited to $\mathbf{M}^{(1)}, \cdots, \mathbf{M}^{(n-1)}$.

Proposition 1. If $\mathbf{M}$ $(= \mathbf{M}^{(1)})$ satisfies Condition (i) and (ii), then $\mathbf{M}^{(2)}, \cdots, \mathbf{M}^{(n-1)}$ have similar properties, which are:

(i)′ the columns of $\mathbf{M}_i^{(s)}$ are linearly independent for every $s \le i \le n$,

(ii)′ $\mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) = \mathsf{rank}(\mathbf{M}_i^{(s)})$ for every $s \le i \le n$.

Proof. The first condition is straightforward, since the columns of $\mathbf{M}_i^{(s)}$ are the basis of $V_i^{(s)}$. Since $\mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) \leq \mathsf{rank}(\mathbf{M}_i^{(s)})$, in order to derive the second condition, we need to show $\mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) \geq \mathsf{rank}(\mathbf{M}_i^{(s)})$. We will show this by induction. Assume $\mathsf{rank}(\mathbf{M}_{i,i}^{(s-1)}) = \mathsf{rank}(\mathbf{M}_i^{(s-1)})$ holds for some $2 \leq s \leq n-1$. Suppose that the basis of $S(\mathbf{M}_i^{(s)})$ is extended to the basis of $S(\mathbf{M}_i^{(s-1)})$ by adding additional $\mathsf{dim}(S(\mathbf{M}_i^{(s-1)})) - \mathsf{dim}(S(\mathbf{M}_i^{(s)}))$ linearly independent columns. Let us focus on the part of these $\mathsf{dim}(S(\mathbf{M}_i^{(s-1)})) - \mathsf{dim}(S(\mathbf{M}_i^{(s)}))$ columns that correspond to the position of the ith thick row. We can observe that the subspace spanned by these $\mathsf{dim}(S(\mathbf{M}_i^{(s-1)})) - \mathsf{dim}(S(\mathbf{M}_i^{(s)}))$ (small) columns and $S(\mathbf{M}_{i,i}^{(s)})$ is exactly the same as $S(\mathbf{M}_{i,i}^{(s-1)})$. This implies

$$\mathsf{rank}(\mathbf{M}_{i,i}^{(s-1)}) \leq \mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) + \mathsf{rank}(\mathbf{M}_i^{(s-1)}) - \mathsf{rank}(\mathbf{M}_i^{(s)}).$$

Since we assumed that $\mathsf{rank}(\mathbf{M}_{i,i}^{(s-1)}) = \mathsf{rank}(\mathbf{M}_i^{(s-1)})$ holds, this leads to $\mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) \geq \mathsf{rank}(\mathbf{M}_i^{(s)})$. □

In addition, we introduce the following propositions.

Proposition 2. For $1 \leq s \leq n-2$ and $s+1 \leq i \leq n$,

$$\delta_i^{(s)}(\mathbf{M}) = \mathsf{rank}(\mathbf{M}_{i,i}^{(s)}) - \mathsf{rank}(\mathbf{M}_{i,i}^{(s+1)}), \tag{2.84}$$

$$\bar{T}_i^{(s)}(\mathbf{M}) \leq T_i^{(s)}(\mathbf{M}) - T_i^{(s+1)}(\mathbf{M}). \tag{2.85}$$

Proof. We can obtain (2.84) as

$$\begin{aligned}
\delta_i^{(s)}(\mathbf{M}) &= \mathsf{rank}\left(\left[\mathbf{M}_s^{(s)} \cdots \mathbf{M}_i^{(s)}\right]\right) \\
&\quad -\mathsf{rank}\left(\left[\mathbf{M}_s^{(s)} \cdots \mathbf{M}_{i-1}^{(s)}\right]\right) \\
&= \mathsf{dim}\left(S\left(\left[\mathbf{M}_s^{(s)} \cdots \mathbf{M}_{i-1}^{(s)}\right]\right) \oplus S\left(\mathbf{M}_i^{(s)}\right)\right) \\
&\quad -\mathsf{dim}\left(S\left(\left[\mathbf{M}_s^{(s)} \cdots \mathbf{M}_{i-1}^{(s)}\right]\right)\right) \\
&= \mathsf{rank}\left(\mathbf{M}_i^{(s)}\right) \\
&\quad -\mathsf{dim}\left(S\left(\left[\mathbf{M}_s^{(s)} \cdots \mathbf{M}_{i-1}^{(s)}\right]\right) \cap S\left(\mathbf{M}_i^{(s)}\right)\right) \qquad (2.86) \\
&= \mathsf{rank}\left(\mathbf{M}_i^{(s)}\right) - \mathsf{rank}\left(\mathbf{M}_i^{(s+1)}\right) \\
&= \mathsf{rank}\left(\mathbf{M}_{i,i}^{(s)}\right) - \mathsf{rank}\left(\mathbf{M}_{i,i}^{(s+1)}\right) \qquad (2.87)
\end{aligned}$$

where (2.86) holds since for any two subspace U and V,

$$\mathsf{dim}(U \oplus V) = \mathsf{dim}(U) + \mathsf{dim}(V) - \mathsf{dim}(U \cap V). \qquad (2.88)$$

Equation (2.85) can also be derived by using (2.88). The case of $i = s+1$ is trivial since $\bar{T}_{s+1}^{(s)}(\mathbf{M}) = T_{s+1}^{(s)}(\mathbf{M})$ and $T_{s+1}^{(s+1)}(\mathbf{M}) = 0$. For $s+2 \leq i \leq n$, using the definition of $\bar{T}_i^{(s)}(\mathbf{M})$, we have

$$\begin{aligned}
\bar{T}_i^{(s)}(\mathbf{M}) = &\ \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-1}^{(s)}\right]\right) \\
= &\ \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-2}^{(s)}\right]\right) + \mathsf{rank}(\mathbf{M}_{i,i-1}^{(s)}) \\
&-\mathsf{dim}\left(S\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-2}^{(s)}\right]\right) \cap S\left(\mathbf{M}_{i,i-1}^{(s)}\right)\right) && (2.89) \\
\leq &\ \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-2}^{(s)}\right]\right) + \mathsf{rank}(\mathbf{M}_{i,i-1}^{(s)}) \\
&-\mathsf{rank}(\mathbf{M}_{i,i-1}^{(s+1)}) && (2.90) \\
\leq &\ \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-3}^{(s)}\right]\right) \\
&+\mathsf{rank}(\mathbf{M}_{i,i-2}^{(s)}) - \mathsf{rank}(\mathbf{M}_{i,i-2}^{(s+1)}) \\
&+\mathsf{rank}(\mathbf{M}_{i,i-1}^{(s)}) - \mathsf{rank}(\mathbf{M}_{i,i-1}^{(s+1)}) && (2.91) \\
&\vdots \\
\leq &\ \mathsf{rank}\left(\mathbf{M}_{i,s}^{(s)}\right) \\
&+ \sum_{j=s+1}^{i-1} \left(\mathsf{rank}(\mathbf{M}_{i,j}^{(s)}) - \mathsf{rank}(\mathbf{M}_{i,j}^{(s+1)})\right) && (2.92) \\
= &\ T_i^{(s)}(\mathbf{M}) - T_i^{(s+1)}(\mathbf{M}),
\end{aligned}$$

where (2.89) follows from (2.88) with $U = S\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-2}^{(s)}\right]\right)$ and $V = S\left(\mathbf{M}_{i,i-1}^{(s)}\right)$, and (2.90) holds since each vector in $S(\mathbf{M}_{i,i-1}^{(s+1)})$ must be contained in both $S\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-2}^{(s)}\right]\right)$ and $S\left(\mathbf{M}_{i,i-1}^{(s)}\right)$. We can obtain (2.91) and (2.92) by repeating the similar steps done in (2.89)-(2.90) recursively. □

As discussed at the beginning of this section, the goal we want to achieve is to find a tight lower bound of $\mathsf{rank}(\mathbf{M})$, since we are going to use $\mathbf{H}_{repair}$ for $\mathbf{M}$. The following lemma deals with lower bounds of $\mathsf{rank}(\mathbf{M})$. We will use Lemma 2 to derive the lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ for $2 \leq s \leq g$ in the next section. Note that the following theorem is extended from Theorem 3.3 of [23] (See Remark 5).

Lemma 2. Suppose a block matrix $\mathbf{M}$ with n^2 submatrices satisfies the following Condition (i) and (ii),

(i) For any $i \in [n]$, the thick column $\mathbf{M}_i$ has linearly independent columns.

(ii) $\mathsf{rank}(\mathbf{M}_{i,i}) = \mathsf{rank}(\mathbf{M}_i)$ for every $i \in [n]$.

then, for any positive integer $s \geq 1$, $\mathsf{rank}(\mathbf{M})$ is lower bounded by

$$\frac{s(s+1)}{2}\mathsf{rank}(\mathbf{M}) \geq \sum_{i=1}^{n} \max\left(0, (s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}), s\mathsf{rank}(\mathbf{M}_{i,i}) - T_i^{(1)}(\mathbf{M})\right). \tag{2.93}$$

Proof. For $1 \leq s \leq n-1$, by using (2.84), we have

$$\begin{aligned} \delta_s^{(s)}(\mathbf{M}) &= \mathsf{rank}(\mathbf{M}_{s,s}^{(s)}) \\ &= \mathsf{rank}(\mathbf{M}_{s,s}^{(s-1)}) - \delta_s^{(s-1)}(\mathbf{M}) \\ &= \mathsf{rank}(\mathbf{M}_{s,s}^{(s-2)}) - \delta_s^{(s-2)}(\mathbf{M}) - \delta_s^{(s-1)}(\mathbf{M}) \\ &\vdots \\ &= \mathsf{rank}(\mathbf{M}_{s,s}^{(1)}) - \sum_{j=1}^{s-1} \delta_s^{(j)}(\mathbf{M}). \end{aligned} \tag{2.94}$$

Similarly, for $1 \leq s \leq n-1$ and $s+1 \leq i \leq n$,

$$\begin{aligned} \delta_i^{(s)}(\mathbf{M}) &\geq \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i}^{(s)}\right]\right) - \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i-1}^{(s)}\right]\right) \\ &= \mathsf{rank}\left(\left[\mathbf{M}_{i,s}^{(s)} \cdots \mathbf{M}_{i,i}^{(s)}\right]\right) - \bar{T}_i^{(s)}(\mathbf{M}) \\ &\geq \mathsf{rank}\left(\mathbf{M}_{i,i}^{(s)}\right) - \bar{T}_i^{(s)}(\mathbf{M}) \\ &= \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{j=1}^{s-1} \delta_i^{(j)}(\mathbf{M}) - \bar{T}_i^{(s)}(\mathbf{M}). \end{aligned} \tag{2.95}$$

By collecting every $\delta_i^{(s)}(\mathbf{M})$ terms on the left hand side of (2.94) and (2.95), we can derive the lower bounds of $\sum_{j=1}^{s} \delta_i^{(j)}(\mathbf{M})$ for $s \leq i \leq n$ as

$$\sum_{j=1}^{s} \delta_i^{(j)}(\mathbf{M}) \geq \begin{cases} \mathsf{rank}(\mathbf{M}_{s,s}^{(1)}) & \text{if } i = s, \\ \mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - \bar{T}_i^{(s)}(\mathbf{M}) & \text{if } s+1 \leq i \leq n. \end{cases} \tag{2.96}$$

For the next step, we will find the lower bounds of $\sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M})$ for $1 \leq i \leq n$ by using (2.96). If $i = 1$,

$$\sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M}) = s\delta_1^{(1)}(\mathbf{M}) = s\mathsf{rank}(\mathbf{M}_{1,1}^{(1)}). \tag{2.97}$$

For $2 \leq i \leq s$,

$$\begin{aligned}
&\sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M}) \\
&\quad = (s-i+1)\sum_{q=1}^{i} \delta_i^{(q)}(\mathbf{M}) + \sum_{q=1}^{i-1}\sum_{j=1}^{q} \delta_i^{(j)}(\mathbf{M}) \\
&\quad \geq (s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) + \sum_{q=1}^{i-1}\sum_{j=1}^{q} \delta_i^{(j)}(\mathbf{M}) && (2.98) \\
&\quad \geq (s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) + \sum_{q=1}^{i-1} \Big(\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - \bar{T}_i^{(q)}(\mathbf{M})\Big) && (2.99) \\
&\quad \geq s\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - \sum_{q=1}^{i-1} \bar{T}_i^{(q)}(\mathbf{M}) \\
&\quad \geq s\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - T_i^{(1)}(\mathbf{M}), && (2.100)
\end{aligned}$$

and for $s+1 \leq i \leq n$,

$$\begin{aligned}
\sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M}) &= \sum_{q=1}^{s} (s-q+1)\delta_i^{(q)}(\mathbf{M}) \\
&= \sum_{q=1}^{s}\sum_{j=1}^{q} \delta_i^{(j)}(\mathbf{M}) && (2.101) \\
&\geq s\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - \sum_{q=1}^{s} \bar{T}_i^{(q)}(\mathbf{M}) && (2.102) \\
&\geq s\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - T_i^{(1)}(\mathbf{M}), && (2.103)
\end{aligned}$$

where (2.99) and (2.102) follow from (2.96), and (2.100) and (2.103) follow from

(2.85) as

$$\begin{aligned}\sum_{q=1}^{i-1} \bar{T}_i^{(q)}(\mathbf{M}) &= \bar{T}_i^{(i-1)}(\mathbf{M}) + \sum_{q=1}^{i-2} \left(T_i^{(q)}(\mathbf{M}) - T_i^{(q+1)}(\mathbf{M}) \right) \\ &= T_i^{(1)}(\mathbf{M}) - T_i^{(i-1)}(\mathbf{M}) + \bar{T}_i^{(i-1)}(\mathbf{M}) \\ &\leq T_i^{(1)}(\mathbf{M}).\end{aligned}$$

for $1 \leq i \leq s$, and

$$\begin{aligned}\sum_{q=1}^{s} \bar{T}_i^{(q)}(\mathbf{M}) &= \bar{T}_i^{(s)}(\mathbf{M}) + \sum_{q=1}^{s-1} \left(T_i^{(q)}(\mathbf{M}) - T_i^{(q+1)}(\mathbf{M}) \right) \\ &= T_i^{(1)}(\mathbf{M}) - T_i^{(s)}(\mathbf{M}) + \bar{T}_i^{(s)}(\mathbf{M}) \\ &\leq T_i^{(1)}(\mathbf{M}).\end{aligned}$$

for $s+1 \leq i \leq n$, Since $\delta_i^{(s)}(\mathbf{M})$ is always positive for every s and i, we have

$$\sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M}) \geq \begin{cases} (s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}), & \text{if } 2 \leq i \leq s \\ 0, & \text{if } s+1 \leq i \leq n. \end{cases}$$
(2.104)

from (2.101) and (2.98).

By combining (2.97), (2.100), (2.103) and (2.104), we have

$$\sum_{i=1}^{n} \sum_{q=1}^{\min(i,s)} (s-q+1)\delta_i^{(q)}(\mathbf{M}) \geq \sum_{i=1}^{n} \max\left(0, (s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}), s\mathsf{rank}(\mathbf{M}_{i,i}^{(1)}) - T_i^{(1)}(\mathbf{M})\right). \quad (2.105)$$

Since $\mathbf{M}^{(s)}$ originates from $\mathbf{M}^{(s-1)}$, we have $\rho^{(1)}(\mathbf{M}) \geq \rho^{(2)}(\mathbf{M}) \geq \cdots \geq$

$\rho^{(n-1)}(\mathbf{M})$. By using this, we note that the left hand side of (2.105) is upper bounded by

$$
\begin{aligned}
\sum_{i=1}^{n}\sum_{q=1}^{\min(i,s)}(s-q+1)\delta_i^{(q)}(\mathbf{M}) &= \sum_{q=1}^{s}(s-q+1)\sum_{i=q}^{n}\delta_i^{(q)}(\mathbf{M}) \\
&= \sum_{q=1}^{s}(s-q+1)\rho^{(q)}(\mathbf{M}) \\
&\le \frac{s(s+1)}{2}\rho^{(1)}(\mathbf{M}), \qquad (2.106)
\end{aligned}
$$

and hence we complete the derivation of (2.93) for the case of $1 \le s \le n-1$.

When $s \ge n$, we prove (2.93) by induction. Suppose that (2.93) holds for some $s \ge n-1$. By adding the term $(s+1)\mathsf{rank}(\mathbf{M})$ to both sides, (2.93) for $s+1$ also holds as

$$
\begin{aligned}
&\frac{(s+1)(s+2)}{2}\mathsf{rank}(\mathbf{M}) \\
&\quad \ge \sum_{i=1}^{n}\max\left((s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}), s\mathsf{rank}(\mathbf{M}_{i,i}) - T_i^{(1)}(\mathbf{M})\right) + (s+1)\mathsf{rank}(\mathbf{M}) \\
&\quad \ge \sum_{i=1}^{n}\max\left((s-i+1)\mathsf{rank}(\mathbf{M}_{i,i}), s\mathsf{rank}(\mathbf{M}_{i,i}) - T_i^{(1)}(\mathbf{M})\right) \\
&\qquad\quad + \sum_{i=1}^{n}\mathsf{rank}(\mathbf{M}_{i,i}) \qquad (2.107) \\
&\quad \ge \sum_{i=1}^{n}\max\left((s-i+2)\mathsf{rank}(\mathbf{M}_{i,i}), (s+1)\mathsf{rank}(\mathbf{M}_{i,i}) - T_i^{(1)}(\mathbf{M})\right),
\end{aligned}
$$

where (2.107) follows from the fact that

$$
\sum_{i=1}^{n}\mathsf{rank}(\mathbf{M}_{i,i}) \le n\mathsf{rank}(\mathbf{M}) \le (s+1)\mathsf{rank}(\mathbf{M}).
$$

Since we have already proved the case of $s = n-1$, we can verify that (2.93) holds for every $s \geq n$ in a recursive manner.

$\square$

Remark 5. Lemma 2 is an extension of Theorem 3.3 of [23]. The difference from [23] is the existence of the max(·) operation. Theorem 3.3 of [23] is equivalent to

$$\begin{aligned}\frac{s(s+1)}{2}\mathsf{rank}(\mathbf{M}) &\geq \sum_{i=1}^{n}(s\mathsf{rank}(\mathbf{M}_{i,i}) - T_i)\\ &= s\sum_{i=1}^{n}\mathsf{rank}(\mathbf{M}_{i,i}) - \sum_{i=2}^{n}\sum_{j=1}^{i-1}\mathsf{rank}(\mathbf{M}_{i,j})\end{aligned}$$

instead of (2.93).

In addition, we introduce another lower bound of $\mathsf{rank}(\mathbf{M})$ which will do an important role in Chapter 3.

Lemma 3. Assume a block matrix $\mathbf{M}$ with n^2 submatrices which satisfies Condition (i) and (ii),

(i) For any $i \in [n]$, the thick column $\mathbf{M}_i$ has linearly independent columns.

(ii) $\mathsf{rank}(\mathbf{M}_{i,i}) = \mathsf{rank}(\mathbf{M}_i)$ for every $i \in [n]$.

For any positive integer $1 \le s \le n-1$, $\mathsf{rank}(\mathbf{M})$ is lower bounded by

$$
\begin{aligned}
s\mathsf{rank}(\mathbf{M}) \;\ge\; & \sum_{i=1}^{n} \mathsf{rank}(\mathbf{M}_{i,i}) - \sum_{i=s}^{n} \bar{T}_i^{(s)}(\mathbf{M}) \\
& + \max\left(0, (s-1)\mathsf{rank}(\mathbf{M}_{n,n}) - \sum_{i=1}^{s-1} \bar{T}_n^{(i)}(\mathbf{M})\right). \qquad (2.108)
\end{aligned}
$$

Proof. At first, we derive a lower bound of $\sum_{j=1}^{s} \rho^{(j)}$. $\rho^{(s)}(\mathbf{M})$ is lower bounded by

$$
\begin{aligned}
\rho^{(s)}(\mathbf{M}) &= \sum_{i=s}^{n} \delta_i^{(s)} \\
&\ge \sum_{i=s}^{n} \left\{ \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{j=1}^{s-1} \delta_i^{(j)}(\mathbf{M}) - \bar{T}_i^{(s)}(\mathbf{M}) \right\} \qquad (2.109) \\
&= \sum_{i=s}^{n} \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{j=1}^{s-1}\sum_{i=s}^{n} \delta_i^{(j)}(\mathbf{M}) - \sum_{i=s}^{n} \bar{T}_i^{(s)}(\mathbf{M}) \\
&= \left\{ \sum_{i=1}^{n} \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=1}^{s-1} \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) \right\} - \sum_{i=s}^{n} \bar{T}_i^{(s)}(\mathbf{M}) \\
&\quad - \sum_{j=1}^{s-1} \left\{ \sum_{i=j}^{n} \delta_i^{(j)}(\mathbf{M}) - \sum_{i=j}^{s-1} \delta_i^{(j)}(\mathbf{M}) \right\} \\
&= \sum_{i=1}^{n} \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n} \bar{T}_i^{(s)}(\mathbf{M}) - \sum_{j=1}^{s-1} \rho^{(j)}(\mathbf{M}) \\
&\quad - \sum_{i=1}^{s-1} \left\{ \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{j=1}^{i} \delta_i^{(j)}(\mathbf{M}) \right\} \qquad (2.110) \\
&= \sum_{i=1}^{n} \mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n} \bar{T}_i^{(s)}(\mathbf{M}) - \sum_{j=1}^{s-1} \rho^{(j)}(\mathbf{M}) \qquad (2.111)
\end{aligned}
$$

where (2.109) follows from (2.94) and (2.95), (2.110) holds because of the fact that

$$\sum_{j=1}^{s-1}\sum_{i=j}^{s-1}\delta_i^{(j)} = \sum_{i=1}^{s-1}\sum_{j=1}^{i}\delta_i^{(j)},$$

and (2.111) follows from (2.96). By moving the last term of the right side, we have

$$\sum_{j=1}^{s}\rho^{(j)}(\mathbf{M}) \geq \sum_{i=1}^{n}\mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n}\bar{T}_i^{(s)}(\mathbf{M}). \tag{2.112}$$

Next, the quantity $\rho^{(i)}(\mathbf{M}) - \rho^{(i+1)}(\mathbf{M})$ is lower bounded by,

$$\begin{aligned}
\rho^{(i)}(\mathbf{M}) - \rho^{(i+1)}(\mathbf{M}) &= \mathsf{rank}(\mathbf{M}^{(i)}) - \mathsf{rank}(\mathbf{M}^{(i+1)}) \\
&\geq \mathsf{rank}(\mathbf{M}^{(i)}) - \mathsf{rank}\left(\left[\mathbf{M}_i^{(i)} \ \cdots \ \mathbf{M}_{n-1}^{(i)}\right]\right) \quad (2.113)\\
&= \delta_n^{(i)}(\mathbf{M}) \quad (2.114)
\end{aligned}$$

for $1 \leq i \leq s-1$, where (2.113) holds since

$$\begin{aligned}
S(\mathbf{M}_j^{(i+1)}) &\subset S\left(\left[\mathbf{M}_i^{(i)} \ \cdots \ \mathbf{M}_{j-1}^{(i)}\right]\right) \\
&\subset S\left(\left[\mathbf{M}_i^{(i)} \ \cdots \ \mathbf{M}_{n-1}^{(i)}\right]\right).
\end{aligned}$$

for every $i+1 \leq j \leq n$. By using (2.112) and (2.114), we have

$$
\begin{aligned}
s\mathsf{rank}(\mathbf{M}) &= s\rho^{(1)}(\mathbf{M}) \\
&= \sum_{j=1}^{s}\rho^{(j)} + \sum_{j=2}^{s}\sum_{i=1}^{j-1}\left(\rho^{(i+1)}(\mathbf{M}) - \rho^{(i)}(\mathbf{M})\right) \\
&\geq \sum_{i=1}^{n}\mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n}\bar{T}_i^{(s)} + \sum_{j=2}^{s}\sum_{i=1}^{j-1}\delta_n^{(i)}(\mathbf{M}) \qquad (2.115) \\
&\geq \sum_{i=1}^{n}\mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n}\bar{T}_i^{(s)} + \sum_{j=2}^{s}\max\left(0, \mathbf{M}_{n,n}^{(1)} - \bar{T}_n^{(j-1)}\right) \qquad (2.116) \\
&\geq \sum_{i=1}^{n}\mathsf{rank}\left(\mathbf{M}_{i,i}^{(1)}\right) - \sum_{i=s}^{n}\bar{T}_i^{(s)} + \max\left(0, (s-1)\mathbf{M}_{n,n}^{(1)} - \sum_{j=2}^{s}\bar{T}_n^{(j-1)}\right)
\end{aligned}
$$

where (2.115) follows from (2.112) and (2.114), and (2.116) follows from (2.96).

□

2.4 An Outer Bound of Linear and Exact-Repair Cooperative Regenerating Codes

In this section, we derive the outer bound on the file size of linear cooperative regenerating codes (1.3). First of all, for a given vector $\mathbf{l} = (l_1, \cdots, l_g)$, if $s = 1$, (1.3) is reduced to

$$
\begin{aligned}
B &\leq \sum_{h=1}^{g} l_h \min\left(\alpha, \Delta_h^{\mathbf{l}}\right) \\
&= \sum_{h=1}^{g} l_h \min\left(\alpha, (d-k+\sum_{t=1}^{h} l_t)\beta_1 + (r-l_h)\beta_2\right), \qquad (2.117)
\end{aligned}
$$

which is equivalent to (2.78). Thus, we need to show that (1.3) holds for $2 \le s \le g$.

The proof of Theorem 1 is almost similar to the proof of (1.1) in Section 2.2. After constructing $\mathbf{H}_{repair}$ that satisfies $S(\mathbf{H}_{repair}^T) \subset S(\mathbf{H}^T)$, we convert the lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ to an upper bound of B.

2.4.1 Construction of $\mathbf{H}_{repair}$

For a given arbitrary vector $\mathbf{l} = (l_1, \cdots, l_g)$ which satisfies $1 \le l_h \le r$ and $\sum_{h=1}^{g} l_h = k = d$, construct $\mathbf{H}_{repair}$ in a similar way as in Section 2.2.1. The only difference is that N_h must be chosen under stricter conditions. For $1 \le h \le g$, N_h is defined as

$$N_h = \underset{\substack{N_h \subset [\sum_{t=0}^{h-1} l_t], \\ |N_h| = r - l_h}}{\arg\max} \sum_{t=0}^{h-1} \mathbb{I}[\mathbf{H}_{h,t} = \mathbf{0}], \tag{2.118}$$

where $\mathbb{I}[\cdot]$ denotes the indicator function, which has 1 as its value if the statement inside brackets is true, and has value of 0, otherwise. Thus, the elements of N_h must be selected to maximize the number of zero matrices out of $\mathbf{H}_{h,0}, \cdots, \mathbf{H}_{h,h-1}$.

For example, $N_1, \cdots, N_4$ of $\mathbf{H}_{repair}$ described in Figure 2.2 satisfy (2.118). For $h = 1$, there is no N_h that makes $\mathbf{H}_{1,0}$ a zero matrix. For the cases of $h = 2$ and $h = 3$, $\mathbf{H}_{2,1} = \mathbf{0}$ and $\mathbf{H}_{3,1} = \mathbf{0}$, since $7 \in N_2$ and $7 \in N_3$. We chose N_4 to satisfy $\{7, 8, 9\} \subset N_4$ in order to make $\mathbf{H}_{4,2}$ and $\mathbf{H}_{4,3}$ zero matrices.

2.4.2 Lower Bound of $\mathsf{rank}(\mathbf{H}_{repair})$

For a given vector $\mathbf{l}$, $\mathbf{H}_{repair}$ has $(g+1)^2$ submatrices. We use Lemma 2 to derive the lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ for $2 \leq s \leq g$. Note that the Lemma 2 is extended from Theorem 3.3 of [23] (See Remark 5).

As discussed in (2.43), $g+1$ diagonal submatrices $\mathbf{H}_{0,0}, \mathbf{H}_{1,1}, \cdots, \mathbf{H}_{g,g}$ are identity matrices. Thus, $\mathbf{H}_{repair}$ satisfies Conditions (i) and (ii) of Lemma 2, and can be used for the matrix $\mathbf{M}$ in Lemma 2. The lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ derived by using (2.93) for $2 \leq s \leq g$ is expressed as

$$\frac{s(s+1)}{2}\mathsf{rank}(\mathbf{H}_{repair}) \geq s(n-k)\alpha + \sum_{h=1}^{g} \max\left(0, (s-h)l_h\alpha, sl_h\alpha - T_h\right), \quad (2.119)$$

where

$$T_h = \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{H}_{h,t}). \quad (2.120)$$

For a given vector $\mathbf{l}$, the terms in the right hand side of (2.119) are fixed except for the term T_h. Thus, minimizing T_h is important for tighter lower bound.

For $1 \leq h \leq g$, $[\mathbf{H}_{h,0} \ \cdots \ \mathbf{H}_{h,h-1}] = \mathbf{A}^{R_h}_{R'_h\left[\sum_{t=1}^{h-1} l_t\right]}$ has $l_h \sum_{t=0}^{h-1} l_t$ numbers of $\alpha \times \alpha$ components. It is important to note that some part of $[\mathbf{H}_{h,0} \ \cdots \ \mathbf{H}_{h,h-1}]$, $\mathbf{A}^{R_h}_{R'_h N_h}$ is a zero matrix, since entries of $\mathbf{A}^{R_h}_{R'_h N_h}$ are contained in the lower triangular entries of $\mathbf{I}_{r\alpha}$ (see Remark 4).

As shown in (2.51)-(2.52), if $j \in \left[\sum_{t=0}^{h-1} l_t\right] \setminus R_h$, $\mathbf{A}_{ij}$ is then the sum of two components $\mathbf{P}'_{ij}$ and $\mathbf{T}'_{ij}$, as

$$\begin{aligned} \mathbf{A}_{ij} &= \mathbf{P}'_{ij} + \mathbf{T}'_{ij} && (2.121) \\ &= \left(\mathbf{P}_{ij} + \sum_{m \in R'_h \setminus \{i\}} \mathbf{T}_{ij,m} \right) + \sum_{m \in R_h \setminus R'_h} \mathbf{T}_{ij,m} && (2.122) \end{aligned}$$

For the case of $j \in N_h$, where $\mathbf{A}_{ij}$ is a zero matrix, if we define $\mathbf{A}_{ij} = \mathbf{P}'_{ij} = \mathbf{T}'_{ij} = \mathbf{0}$, then $\mathbf{A}_{ij} = \mathbf{P}'_{ij} + \mathbf{T}'_{ij}$ holds for every case of $j \in \left[\sum_{t=0}^{h-1} l_t\right]$. For $0 \leq t \leq h-1$, $\mathbf{H}_{h,t}$ is also the sum of two components, as

$$\mathbf{H}_{h,t} = \mathbf{P}'_{R'_h R'_t} + \mathbf{T}'_{R'_h R'_t}. \qquad (2.123)$$

The upper bounds of $\sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{P}'_{R'_h R'_t})$ can be derived similarly to (2.54)-(2.59) as

$$\begin{aligned} \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{P}'_{R'_h R'_t}) &\leq \sum_{j \in L_h} \mathsf{rank}(\mathbf{P}'_{R'_h j}) && (2.124) \\ &\leq l_h(d - k + \sum_{t=1}^{h} l_t)\beta_1, && (2.125) \end{aligned}$$

The upper bound of $\sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{T}'_{R'_h R'_t})$ can be derived as

$$\begin{aligned}
\sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{T}'_{R'_h R'_t}) &\leq \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{T}'_{R'_h R'_t})\mathbb{I}[\mathbf{H}_{h,t} = \mathbf{0}] && (2.126)\\
&\leq \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{T}'_{R'_h([n]\setminus R_h)})\mathbb{I}[\mathbf{H}_{h,t} = \mathbf{0}] && (2.127)\\
&= c_h \mathsf{rank}(\mathbf{T}'_{R'_h([n]\setminus R_h)}) && (2.128)\\
&= c_h \sum_{i\in R'_h} \sum_{m\in N_h} \mathsf{rank}(\mathbf{T}_{i([n]\setminus R_h),m}) && (2.129)\\
&\leq c_h |R'_h||N_h|\beta_2 && (2.130)\\
&= c_h l_h (r - l_h)\beta_2, && (2.131)
\end{aligned}$$

where (2.129) follows from the fact that

$$\begin{aligned}
b_h &:= \max_{\substack{A\subset\{0,\cdots,h-1\},\\ \sum_{t\in A} l_t \leq r-l_h}} |A|. && (2.132)\\
&= \sum_{t=0}^{h-1} \mathbb{I}[\mathbf{H}_{h,t} \neq \mathbf{0}], && (2.133)\\
&= h - \sum_{t=0}^{h-1} \mathbb{I}[\mathbf{H}_{h,t} = \mathbf{0}], && (2.134)
\end{aligned}$$

and $c_h := h - b_h$ for $1 \leq h \leq g$.

By combining (2.125) and (2.53), for $1 \leq h \leq g$, $T_h = \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{H}_{h,t})$ is upper bounded by

$$
\begin{aligned}
T_h &\leq \sum_{t=0}^{h-1} \mathsf{rank}(\mathbf{P}'_{R'_h R'_t}) + \mathsf{rank}(\mathbf{T}'_{R'_h R'_t}) \\
&\leq l_h(d-k+\sum_{t=1}^{h} l_t)\beta_1 + c_h l_h (r - l_h)\beta_2 \\
&= l_h \Delta_h^{\mathbf{l}},
\end{aligned}
\tag{2.135}
$$

where $\Delta_h^{\mathbf{l}}$ was defined in (1.4). By applying this to (2.119), the lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ can be derived as

$$
\begin{aligned}
&\mathsf{rank}(\mathbf{H}_{repair}) \\
&\quad \geq \frac{2(n-k)\alpha}{s+1} + \frac{2}{s(s+1)} \sum_{h=1}^{g} l_h \max\left(0, (s-h)\alpha, s\alpha - \Delta_h^{\mathbf{l}}\right).
\end{aligned}
\tag{2.136}
$$

2.4.3 Derivation of the Proposed Outer Bound

By using the relation $\mathsf{rank}(\mathbf{H}) = n\alpha - B$, the lower bound of $\mathsf{rank}(\mathbf{H})$ can be converted to an upper bound of B as

$$
\begin{aligned}
B &= n\alpha - \mathsf{rank}(\mathbf{H}) \\
&\leq n\alpha - \mathsf{rank}(\mathbf{H}_{repair}) \\
&\leq n\alpha - \frac{2}{s+1}(n-k)\alpha - \frac{2}{s+1}\sum_{h=1}^{g} l_h\alpha \\
&\quad - \frac{2}{s(s+1)} \sum_{h=1}^{g} l_h \max\left(-s\alpha, -h\alpha, -\Delta_h^{\mathbf{l}}\right) \\
&\leq \frac{s-1}{s+1} n\alpha + \frac{2}{s(s+1)} \sum_{h=1}^{g} l_h \min\left(s\alpha, h\alpha, \Delta_h^{\mathbf{l}}\right).
\end{aligned}
\tag{2.137}
$$

Remark 6. We only defined (1.3) for the case of $2 \leq s \leq g$, even if (2.93) holds for every positive integer $s \geq 1$. In fact, we can also derive (1.3) for $s = 1$ and $s \geq g+1$ in the same manner described in this section, but we do not have to use them. When $s = 1$, this is because the functional repair bound (1.1) is tighter than (1.3) with $s = 1$. The reason why we did not consider $s \geq g+1$ is that every lower bound of $\mathsf{rank}(\mathbf{H}_{repair})$ from (2.93) for $s \geq g+1$ is always smaller than the lower bound obtained for $s = g$, since every diagonal submatrix of $\mathbf{H}_{repair}$ is square and nonsingular.

To verify this, we will show that for a matrix $\mathbf{M}$ which has n^2 submatrices and satisfies Conditions (i) and (ii) of Lemma 2, if $\mathbf{M}_{i,i}$ is square and nonsingular for every $i \in [n]$, then the lower bound of $\mathsf{rank}(\mathbf{M})$ given in (2.93) decreases as s increases when $s \geq n$. For a given $i \in [n]$, if $\mathbf{M}_{i,i}$ is nonsingular, it must be that

$$\mathsf{rank}(\mathbf{M}_{i,i}) \geq \mathsf{rank}(\mathbf{M}_{i,j}) \text{ and } \mathsf{rank}(\mathbf{M}_{i,i}) \geq \mathsf{rank}(\mathbf{M}_{j,i}) \quad \text{for every } j \in [n] \setminus i. \tag{2.138}$$

According to (2.93), for a given $s \geq n-1$, $\mathsf{rank}(\mathbf{M})$ is lower bounded by

$$\frac{2}{s(s+1)} \sum_{i=1}^{n} \left(s\mathsf{rank}(\mathbf{M}_{i,i}) - T_i\right),$$

where the max operation disappeared since

$$(i-1)\mathsf{rank}(\mathbf{M}_{i,i}) \geq \sum_{j=1}^{i-1} \mathsf{rank}(\mathbf{M}_{i,j}) = T_i, \tag{2.139}$$

which follows from (2.138). The difference between the lower bounds for $s = s_0$ and $s = s_0 + 1$ is

$$\begin{aligned} &\frac{2}{s_0(s_0+1)} \sum_{i=1}^{n} (s_0\mathsf{rank}(\mathbf{M}_{i,i}) - T_i) \\ &- \frac{2}{(s_0+1)(s_0+2)} \sum_{i=1}^{n} ((s_0+1)\mathsf{rank}(\mathbf{M}_{i,i}) - T_i) \\ &= \frac{2}{s(s+1)(s+2)} \sum_{i=1}^{n} (s\mathsf{rank}(\mathbf{M}_{i,i}) - 2T_i)\,. \end{aligned} \tag{2.140}$$

This value is always positive, since

$$\begin{aligned} &\sum_{i=1}^{n} (s\mathsf{rank}(\mathbf{M}_{i,i}) - 2T_i) \\ &\geq \sum_{i=1}^{n} ((n-1)\mathsf{rank}(\mathbf{M}_{i,i}) - 2T_i) \qquad (2.141) \\ &= (n-1)\sum_{i=1}^{n} \mathsf{rank}(\mathbf{M}_{i,i}) - 2\sum_{i=2}^{n}\sum_{j=1}^{i-1} \mathsf{rank}(\mathbf{M}_{i,j}) \\ &= (n-1)\sum_{i=1}^{n} \mathsf{rank}(\mathbf{M}_{i,i}) \\ &- \sum_{i=2}^{n}\sum_{j=1}^{i-1} \mathsf{rank}(\mathbf{M}_{i,j}) - \sum_{j=1}^{n-1}\sum_{i=j+1}^{n} \mathsf{rank}(\mathbf{M}_{i,j}) \\ &\geq 0, \qquad (2.142) \end{aligned}$$

where (2.141) follows from $s \geq n-1$ and (2.142) follows from (2.138).

2.5 Evaluation of the Proposed Outer Bound

In this section, we discuss the performance of the proposed outer bound of Theorem 1 by evaluating it on the (α, γ)-plane for various parameters. Given a vector $\mathbf{l}$ and an integer s, the right side of (2.137) is a function of $(\alpha, \beta_1, \beta_2)$ since $\Delta_h^{\mathbf{l}}$ is a fucntion of β_1 and β_2. For a fixed set of parameters $(\alpha, \beta_1, \beta_2)$, we can obtain the least upper bound of B by minimizing the right side of (2.137) over $\mathbf{l} \in \mathcal{L}$ and $1 \leq s \leq g$. As a result, we have

$$\begin{aligned} B &\leq \min_{\substack{\mathbf{l}\in\mathcal{L} \\ 1\leq s\leq g}} \left\{ \frac{s-1}{s+1} n\alpha + \frac{2}{s(s+1)} \sum_{h=1}^{g} l_h \min\left(s\alpha, h\alpha, \Delta_h^{\mathbf{l}}\right) \right\} \\ &:= \hat{B}(\alpha, \beta_1, \beta_2), \end{aligned} \tag{2.143}$$

where we defined $\hat{B}(\alpha, \beta_1, \beta_2)$ as the least upper bound of B, and $\mathcal{L}$ is the set of all vectors $\mathbf{l} = \{l_1, \cdots, l_g\}$ such that $1 \leq l_h \leq r$ for every $h \in [g]$ and $\sum_{h=1}^{g} l_h = k$. For a given value of γ, by maximizing $\hat{B}(\alpha, \beta_1, \beta_2)$ over β_1 and β_2 with $\gamma = d\beta_1 + (r-1)\beta_2$, $\hat{B}(\alpha, \beta_1, \beta_2)$ is transformed into a function of (α, γ) as

$$B \leq \max_{d\beta_1+(r-1)\beta_2=\gamma} \hat{B}(\alpha, \beta_1, \beta_2). \tag{2.144}$$

In Figures 2.3 and 2.4, (α, γ) values satisfying

$$1 = \max_{d\beta_1+(r-1)\beta_2=\gamma} \hat{B}(\alpha, \beta_1, \beta_2) \tag{2.145}$$

are plotted for various (n, k, d, r). The set of the points forms a piece-wise linear curve on the (α, γ)-plane.

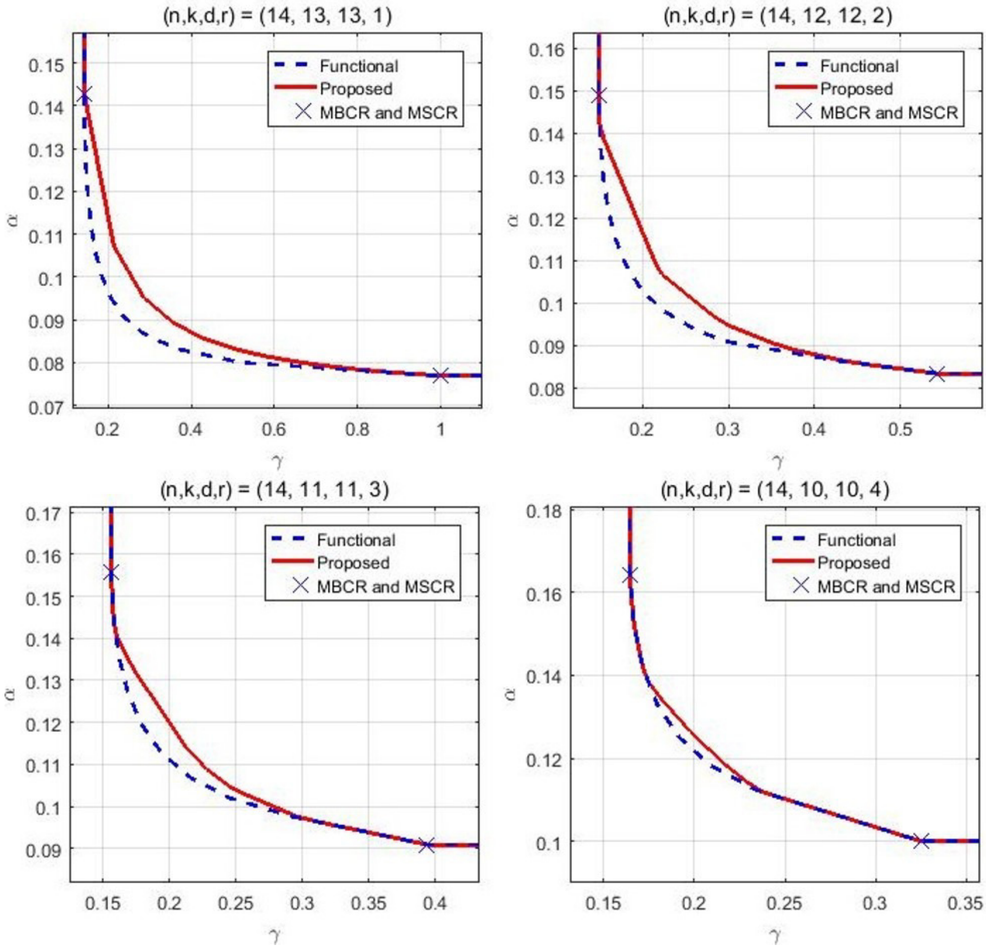

Figure 2.3: Performance comparison between the proposed outer bound (Theorem 1) and the cutset bound (1.1) for different values of r when $n = 14$ and $d = k = n - r$.

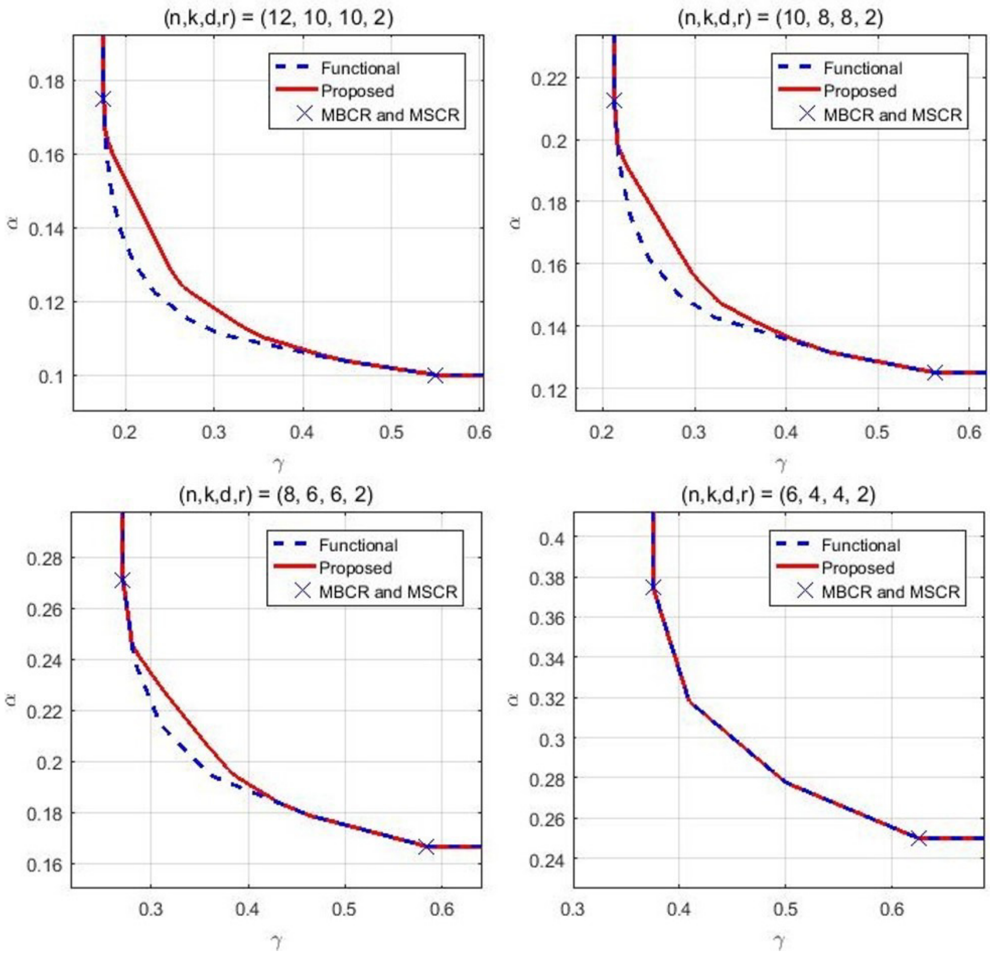

Figure 2.4: Performance comparison between the proposed outer bound (Theorem 1) and the cutset bound (1.1) for different values of k when $r = 2$ and $d = k = n - r$.

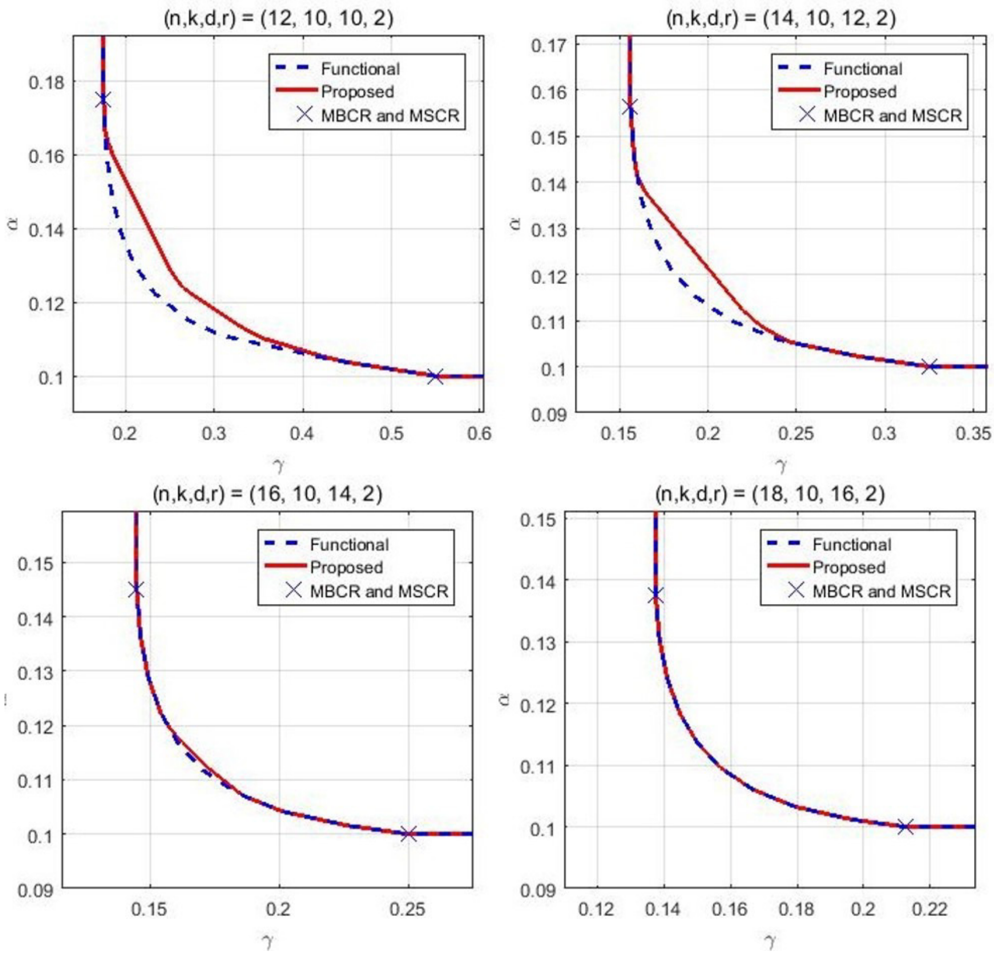

Figure 2.5: Performance comparison between the proposed outer bound (Theorem 1) and the cutset bound (1.1) for different values of d when $r = 2$ and $k = 10$.

In Figure 2.3 we plot the performance of the bounds for different r when $n = k + r = 14$. The (α, γ) points in the region above the cutset bound and under the proposed bound are not achievable with the exact repair model, but achievable with the functional repair model. This region becomes smaller as r increases. Figure 2.4 illustrates the piece-wise linear curves for different k when fixed $r = 2$, and Figure 2.5 includes the (α, γ) curves for different values of d when $r = 2$ and $k = 10$. It is observed that for fixed r the proposed outer bound becomes tighter as k increases and $d - k$ decreases, compared to the cutset bound.

To sum up, the proposed outer bound stated in Theorem 1 is effective if r and d are small, or k is large, when compared with the functional-repair cutset bound. This is because the lower bound of $\mathsf{rank}(\mathbf{H}_{h,t})$ becomes loose when it is approximated by the sum of ranks of its $\alpha \times \alpha$ components. It is observed that the lower bound becomes tighter if r is small compared to the value of n.

Chapter 3

An Improved Outer Bound for the Case of Single Node Repair

3.1 Symmetric Exact-Repair codes

In this chapter, we restrict our discussion to the case of single node repair where $r = 1$. The proposed outer bound on the storage-bandwidth tradeoff of single-repair codes (The cooperative regenerating codes with $r = 1$ are simply called *regenerating codes.*), which was stated in Theorem 2 and 3, is tighter than the $r = 1$ case of the first outer bound discussed in Chapter 2. This improvement is motivated by the storage node symmetry of exact repair codes, which was first discussed by Tian in [18].

Symmetric regenerating codes are defined to be regenerating codes that are invariant to index permutation. When we discuss about outer bounds on the S-B tradeoff or conditions the regenerating codes must satisfy, it is sufficient to consider symmetric regenerating codes, since there is no operating point (α, β), which can only be achieved by non-symmetric regenerating codes. Let $\mathcal{C}$ be a set of codewords of an non-symmetric regenerating code operating at (α, β). Suppose a permutation code $\mathcal{C}_\pi$ where the indices of nodes $(1, 2, \cdots, n)$ is permuted into another order of n integers $\pi\left[(1, 2, \cdots, n)\right]$. Let $\mathcal{C}^{'}$ be a new code generated by space-sharing $n!$ possible $\mathcal{C}_\pi$s with the same fraction. Then $\mathcal{C}^{'}$ can be regarded as a symmetric code operating at the same point (α, β). Even though the size of the alphabet (*e.g.* finite fields) might become larger, it is not an interested problem when we want to verify the existence of such codes. Note that for a symmetric regenerating code, the amount of any information measure (*e.g.* entropy, rank) is not dependent on the particular choice of nodes, but only on the number of nodes.

3.2 Conditions for Parity Check Matrices of Single Repair Codes

Lemma 4 below gives some conditions that the parity check matrix $\mathbf{H}$ of an (n, k, d)-regenerating code must satisfy if the code is an $(n\alpha, B)$-linear code.

Lemma 4 is analogous to Lemma 1 in Chapter 2.

Lemma 4. The parity check matrix $\mathbf{H}$ of an $(n = d{+}1, k, d)$-linear regenerating code satisfy the following two conditions (i) and (ii).

(i) A $(n\alpha - B) \times (n-k)\alpha$ matrix constructed by collecting arbitrary $n-k$ thick columns of $\mathbf{H}$ has full rank $(n-k)\alpha$.

(ii) For an integer $i \in [n]$, there exists an $\alpha \times n\alpha$ matrix $\hat{\mathbf{H}}_i$ that satisfies the following conditions (a) and (b).

(a) Each of $\alpha \times \alpha$ submatrices of $\hat{\mathbf{H}}_i$ satisfies

$$\begin{cases} \hat{\mathbf{H}}_i^h = \mathbf{I}_\alpha, & \text{if } h = i, \\ \mathsf{rank}(\hat{\mathbf{H}}_i^h) \le \beta, & \text{if } h \in D, \end{cases} \tag{3.1}$$

where $\hat{\mathbf{H}}_i = \begin{bmatrix} \hat{\mathbf{H}}_i^1 & \cdots & \hat{\mathbf{H}}_i^n \end{bmatrix}$.

(b) $S(\hat{\mathbf{H}}_i^T) \subset S(\mathbf{H}^T)$.

Proof. Consider n code symbols $\mathbf{c}_1, \cdots, \mathbf{c}_n$ such that

$$\mathbf{c} = (\mathbf{c}_1\ \mathbf{c}_2\ \ldots\ \mathbf{c}_n), \tag{3.2}$$

where each symbol is a row vector of length α. For $A = \{a_1, a_2, \cdots, a_{|A|}\}$, a subset of $[n]$, define $\mathbf{c}_A$ as $\left(\mathbf{c}_{a_1}\ \mathbf{c}_{a_2}\ \ldots\ \mathbf{c}_{a_{|A|}}\right)$, and let $\mathbf{G}_A$ and $\mathbf{H}_A$ be the matrices formed by combining thick columns of $\mathbf{G}$ and $\mathbf{H}$ whose indices correspond to the elements of A.

Condition (i) can be derived from the data collection property of regenerating codes. Let K be a subset of $[n]$ with k elements and $\bar{K} = [n] \setminus K$. $\mathbf{s} = \mathbf{0}$ is the unique vector satisfying $\mathbf{s}\mathbf{H}_{\bar{K}}^T = \mathbf{0}$, since $\mathbf{c} = \mathbf{0}$ is the unique codeword satisfying $\mathbf{c}_K = \mathbf{0}$, according to the data collection property. This means every columns of $\mathbf{H}_{\bar{K}}^T$ are linearly independent.

Condition (ii) is derived from the node repair property. Suppose a node repair process where d helper nodes whose indices correspond to $D = [n] \setminus \{i\} = \{l_1, \cdots, l_d\}$ repair node i ($\notin D$). Each helper node produces β symbols by combining its own α symbols and transmits them to node i. Let $\mathbf{c}_j \Phi_{ji}$ be the β symbols node i downloaded from node $j \in D$, where Φ_{ji} is the $\alpha \times \beta$ encoding matrix. Node i can obtain $\mathbf{c}_i$ by combining $\mathbf{c}_{l_1}\Phi_{l_1 i}, \cdots, \mathbf{c}_{l_d}\Phi_{l_d i}$, and this procedure is expressed by

$$\mathbf{c}_i = \sum_{j=1}^{d} (\mathbf{c}_{l_j} \Phi_{i l_j}) \Psi_{l_j i}, \tag{3.3}$$

where $\Psi_{l_1 i},\ \Psi_{l_2 i}, \cdots, \Psi_{l_d i}$ are $\beta \times \alpha$ matrices. Define $\hat{\mathbf{H}}_i = \begin{bmatrix} \hat{\mathbf{H}}_i^1 & \cdots & \hat{\mathbf{H}}_i^n \end{bmatrix}$ as

$$\hat{\mathbf{H}}_i^h = \begin{cases} \mathbf{I}_\alpha, & \text{if } h = i, \\ -\Psi_{ti}^T \Phi_{it}^T, & \text{if } h \in D. \end{cases} \tag{3.4}$$

$\hat{\mathbf{H}}_i$ satisfies (3.1), since the rank of $\Phi_{i l_j} \Psi_{l_j i}$ is less than β. It can be shown that every row of $\hat{\mathbf{H}}_i$ belongs to the row space of $\mathbf{H}$ by verifying

$$\begin{aligned}
\mathbf{c}\hat{\mathbf{H}}_{i,D}^T &= \sum_{j=1}^{n} \mathbf{c}_j (\mathbf{H}_{i,D}^j)^T \\
&= \mathbf{c}_i \mathbf{I}_\alpha + \sum_{j=1}^{d} \mathbf{c}_{l_j} (-\Psi_{l_j i}^T \Phi_{i l_j}^T)^T \\
&= \mathbf{c}_i - \sum_{j=1}^{d} \mathbf{c}_{l_j} \Phi_{i l_j} \Psi_{l_j i} \\
&= \mathbf{0}
\end{aligned} \tag{3.5}$$

for any codeword $\mathbf{c}$.

□

As stated in the proof of Lemma 4, conditions (i) and (ii) are derived from two properties of regenerating codes, data collection and node repair, respectively. The following corollary can be regarded as a modification of condition (i) of Lemma 4.

Corollary 1. Suppose an (n, k, d)-linear regenerating code, and let $\mathbf{H}$ be its parity check matrix. Let K be a subset of $[n]$, and i be an element of $[n]$ such that $|K| = k$ and $i \notin K$. For arbitrarily chosen (i, K), there exists an $\alpha \times n\alpha$ matrix $\hat{\mathbf{H}}_{i,K}$ that satisfy the following conditions (a) and (b).

(a) Each of $\alpha \times \alpha$ submatrices of $\hat{\mathbf{H}}_{i,K}$ satisfies

$$\begin{cases} \hat{\mathbf{H}}_{i,K}^{t} = \mathbf{I}_{\alpha}, & \text{if } t = i, \\ \mathsf{rank}(\hat{\mathbf{H}}_{i,K}^{t}) \leq \alpha, & \text{if } t \in K, \\ \hat{\mathbf{H}}_{i,K}^{t} = \mathbf{0}, & \text{otherwise,} \end{cases} \tag{3.6}$$

where $\hat{\mathbf{H}}_{i,K} = \begin{pmatrix} \hat{\mathbf{H}}_{i,K}^{1} & \cdots & \hat{\mathbf{H}}_{i,K}^{n} \end{pmatrix}$.

(b) $S(\hat{\mathbf{H}}_{i,K}^{T}) \subset S(\mathbf{H}^{T})$.

Proof. Given a subset of $[n]$, $K = \{l_1, \cdots, l_k\}$, $\mathbf{m}$ can be obtained from $\mathbf{c}_K$ based on the data collection property. This procedure is expressed by

$$\mathbf{m} = \sum_{t=1}^{k} \mathbf{c}_{l_t} \Theta_{l_t}, \tag{3.7}$$

where $\Theta_{l_1}, \cdots, \Theta_{l_k}$ are $\alpha \times B$ matrices. Consequently, this implies that the symbols of $\mathbf{c}_i$ can also be expressed by linear combinations of $\mathbf{c}_K$ as

$$\begin{aligned} \mathbf{c}_i &= \mathbf{m}\mathbf{G}_i \\ &= \sum_{t=1}^{k} \mathbf{c}_{l_t} \Theta_{l_t} \mathbf{G}_i. \end{aligned} \tag{3.8}$$

Define $\hat{\mathbf{H}}_{i,K} = \begin{pmatrix} \hat{\mathbf{H}}_{i,K}^{1} & \cdots & \hat{\mathbf{H}}_{i,K}^{n} \end{pmatrix}$ where the submatrices satisfy

$$\hat{\mathbf{H}}_{i,K}^{t} = \begin{cases} \mathbf{I}_{\alpha}, & \text{if } t = i, \\ -\mathbf{G}_i^{T} \Theta_t^{T}, & \text{if } t \in K, \\ \mathbf{0}, & \text{otherwise.} \end{cases} \tag{3.9}$$

It can be easily verified that $\hat{\mathbf{H}}_{i,K}$ satisfies the condition (a). By (3.8), for every codeword $\mathbf{c}$,

$$\begin{aligned} \mathbf{c}\hat{\mathbf{H}}_{i,K}^T &= \mathbf{c}_i - \sum_{t=1}^{k} \mathbf{c}_{l_t} \Theta_{l_t} \mathbf{G}_i \\ &= 0 \end{aligned} \tag{3.10}$$

This implies the condition (b) is also satisfied, since the row space of $\hat{\mathbf{H}}_{i,K}$ is orthogonal to $S(\mathbf{G}^T)$ and belongs to $S(\mathbf{H}^T)$. □

3.3 Construction of $\mathbf{H}_{single}$

In this section, we shall define a block matrix matrix $\mathbf{H}_{single}$ which satisfies

$$S(\mathbf{H}_{single}^T) \subset S(\mathbf{H}), \tag{3.11}$$

as $\mathbf{H}_{repair}$ used in Chapter 2 satisfies

$$S(\mathbf{H}_{repair}^T) \subset S(\mathbf{H}^T). \tag{3.12}$$

$\mathbf{H}_{single}$ is made up of the rows of the parity check matrix $\mathbf{H}$ of the corresponding regenerating codes, as $\mathbf{H}_{repair}$ does. However, there is a notable difference between $\mathbf{H}_{single}$ and $\mathbf{H}_{repair}$ in combining direction of the rows of $\mathbf{H}$. While $\mathbf{H}_{repair}$ is constructed by combining the rows of $\mathbf{H}$ vertically as in (2.39), $\mathbf{H}_{single}$ uses the rows of $\mathbf{H}$ as its columns. The column space of $\mathbf{H}_{single}$

is contained in the column space of $\mathbf{H}$ as (3.11), where the column space of $\mathbf{H}_{repair}$ is contained in the columns space of $\mathbf{H}$. More specifically, $\mathbf{H}_{single}$ is constructed by combining $\alpha \times n\alpha$ matrices $\hat{\mathbf{H}}_i$ and $\hat{\mathbf{H}}_{i,K}$ horizontally given by Lemma 4 and Corollary 1.

Let the quotient and remainder when n $(= d+1)$ is divided by τ $(= d+1-k)$ be Q and R, respectively. i.e.,

$$n = d + 1 = Q\tau + R, \qquad 0 \leq R \leq \tau. \tag{3.13}$$

In this section, Q kinds of submatrix patterns will be considered. For $1 \leq q \leq Q$, Let $\mathbf{u}_q$ be a vector of length $g = d - q(\tau - 1) + 1$ such that

$$\mathbf{u}_q = (\underbrace{1, \cdots, 1}_{f=d-q\tau+1}, \underbrace{\tau, \cdots, \tau}_{q}). \tag{3.14}$$

The elements of $\mathbf{u}_q\alpha$ can be regarded as a pattern of widths of thick columns and rows of $\mathbf{H}_{single}$. The sum of all elements of $\mathbf{u}_q$ always equals n. Specifically, if $\mathbf{u}_q = (u_1, \cdots, u_g)$, $n\alpha$ columns of $\mathbf{H}_{repair}$ are partitioned into g $(= f + q = d - q(\tau - 1) + 1)$ thick columns, each of which consists of $u_1\alpha, \cdots, u_g\alpha$ columns, respectively. Let the thick rows of $\mathbf{H}_{single}$ be partitioned in the same way. In this manner, $\mathbf{H}_{single}$ is broken into g^2 submatrices $\mathbf{H}_{1,1}, \cdots, \mathbf{H}_{g,g}$ as

$$\mathbf{H}_{single} = \begin{bmatrix} \mathbf{H}_{1,1} & \mathbf{H}_{1,2} & \dots & \mathbf{H}_{1,g} \\ \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \dots & \mathbf{H}_{2,g} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{g,1} & \mathbf{H}_{g,2} & \dots & \mathbf{H}_{g,g} \end{bmatrix} = [\mathbf{H}_1 \ \mathbf{H}_2 \ \cdots \mathbf{H}_g], \tag{3.15}$$

where $\mathbf{H}_h = \left[\mathbf{H}_{1,h}^T \ \mathbf{H}_{2,h}^T \ \cdots \ \mathbf{H}_{g,h}^T\right]^T$ is the hth thick column of $\mathbf{H}_{single}$ for $1 \leq h \leq g$. Each of the g thick columns of $\mathbf{H}_{single}$ is defined as

$$\mathbf{H}_h = \begin{cases} \hat{\mathbf{H}}_h^T, & \text{if } 1 \leq h \leq f, \\ \mathbf{H}_h^1 - \mathbf{H}_h^2 & \text{if } f+1 \leq h \leq g, \end{cases} \tag{3.16}$$

where for $f+1 \leq h \leq g$, $\mathbf{H}_h^1$ and $\mathbf{H}_h^2$ are defined as

$$\mathbf{H}_h^1 = \left[\hat{\mathbf{H}}_{f_h^1}^T \ \cdots \ \hat{\mathbf{H}}_{f_h^\tau}^T\right]$$

and

$$\begin{aligned} \mathbf{H}_h^2 \ = \ & \left[\hat{\mathbf{H}}_{f_h^1,[d+1]\setminus\{f_h^1,\cdots,f_h^\tau\}}^T \ \cdots \ \hat{\mathbf{H}}_{f_h^\tau,[d+1]\setminus\{f_h^1,\cdots,f_h^\tau\}}^T\right] \\ & \times \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \hat{\mathbf{H}}_{f_h^2}^1 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \hat{\mathbf{H}}_{f_h^3}^1 & \hat{\mathbf{H}}_{f_h^3}^2 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \hat{\mathbf{H}}_{f_h^\tau}^1 & \hat{\mathbf{H}}_{f_h^\tau}^2 & \cdots & \hat{\mathbf{H}}_{f_h^\tau}^{\tau-1} & \mathbf{0} \end{bmatrix}^T, \end{aligned}$$

where $f_h^j := f + (q-1)\tau + j = (d - q\tau + 1) + (q-1)\tau + j$ for $1 \le j \le \tau$. Note that for $f+1 \le h \le g$, the two components of $\mathbf{H}_h$, $\mathbf{H}_h^1$ and $\mathbf{H}_h^2$, originate from $\hat{\mathbf{H}}_i$ and $\hat{\mathbf{H}}_{i,K}$ given in Lemma 4 and Corollary 1, respectively.

It can be easily verified that the every diagonal submatrices of $\mathbf{H}_{single}$ is nonsingular by using the conditions of $\hat{\mathbf{H}}_i$ and $\hat{\mathbf{H}}_{i,K}$ stated in 4 and Corollary 1. For $1 \le h \le f$

$$\mathbf{H}_{h,h} = \left(\hat{\mathbf{H}}_h^h\right)^T = \mathbf{I}_\alpha, \tag{3.17}$$

and for $f+1 \le h \le g$,

$$\begin{aligned}
\mathbf{H}_{h,h} &= \begin{bmatrix} \mathbf{I}_\alpha & \hat{\mathbf{H}}^2_{f_h^1} & \hat{\mathbf{H}}^3_{f_h^1} & \cdots & \hat{\mathbf{H}}^\tau_{f_h^1} \\ \hat{\mathbf{H}}^1_{f_h^2} & \mathbf{I}_\alpha & \hat{\mathbf{H}}^3_{f_h^2} & \cdots & \hat{\mathbf{H}}^\tau_{f_h^2} \\ \hat{\mathbf{H}}^1_{f_h^3} & \hat{\mathbf{H}}^2_{f_h^3} & \mathbf{I}_\alpha & \ddots & \hat{\mathbf{H}}^\tau_{f_h^3} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \hat{\mathbf{H}}^1_{f_h^\tau} & \hat{\mathbf{H}}^2_{f_h^\tau} & \cdots & \hat{\mathbf{H}}^{\tau-1}_{f_h^\tau} & \mathbf{I}_\alpha \end{bmatrix}^T - \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \hat{\mathbf{H}}^1_{f_h^2} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \hat{\mathbf{H}}^1_{f_h^3} & \hat{\mathbf{H}}^2_{f_h^3} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \hat{\mathbf{H}}^1_{f_h^\tau} & \hat{\mathbf{H}}^2_{f_h^\tau} & \cdots & \hat{\mathbf{H}}^{\tau-1}_{f_h^\tau} & \mathbf{0} \end{bmatrix}^T \\
&= \begin{bmatrix} \mathbf{I}_\alpha & \hat{\mathbf{H}}^2_{f_h^1} & \hat{\mathbf{H}}^3_{f_h^1} & \cdots & \hat{\mathbf{H}}^\tau_{f_h^1} \\ \mathbf{0} & \mathbf{I}_\alpha & \hat{\mathbf{H}}^3_{f_h^2} & \cdots & \hat{\mathbf{H}}^\tau_{f_h^2} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_\alpha & \ddots & \hat{\mathbf{H}}^\tau_{f_h^3} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_\alpha \end{bmatrix}^T
\end{aligned} \tag{3.18}$$

Thus,

$$\mathsf{rank}(\mathbf{H}_{h,h}) = \begin{cases} \alpha & \text{if } 1 \le h \le f, \\ \tau\alpha & \text{if } f+1 \le h \le g, \end{cases} \tag{3.19}$$

Additionally, we can verify the rank of each lower-triangular submatrix ($\mathbf{H}_{h,t}$ with $h > t$) is upper bounded by

$$\mathsf{rank}(\mathbf{H}_{h,t}) \le \begin{cases} \beta & \text{if } 1 \le t \le f \text{ and } 2 \le h \le f, \\ \tau\beta & \text{if } 1 \le t \le f \text{ and } f+1 \le h \le g, \\ R(\alpha,\beta) & \text{if } f+1 \le t \le g, \end{cases} \tag{3.20}$$

where the definition of $R(\alpha, \beta)$ is given by (1.7).

3.4 Derivation of Two Sub-Bounds

3.4.1 Proof of Theorem 2

Assume the matrix $\mathbf{H}_{single}$ is given for an arbirary $1 \le q \le Q = \lfloor \frac{d+1}{\tau} \rfloor$. For a given value of p, define a block matrix P_{single} with $(g-p)^2$ submatirces as

$$\begin{aligned}
\mathbf{P}_{single} &= \begin{bmatrix} \mathbf{H}_{1+p,1+p} & \mathbf{H}_{1+p,2+p} & \dots & \mathbf{H}_{1+p,g} \\ \mathbf{H}_{2+p,1+p} & \mathbf{H}_{2+p,2+p} & \dots & \mathbf{H}_{2+p,g} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{g,1+p} & \mathbf{H}_{g,2+p} & \dots & \mathbf{H}_{g,g} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{P}_{1,1} & \mathbf{P}_{1,2} & \dots & \mathbf{P}_{1,g-p} \\ \mathbf{P}_{2,1} & \mathbf{P}_{2,2} & \dots & \mathbf{P}_{2,g-p} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{g-p,1} & \mathbf{P}_{g-p,2} & \dots & \mathbf{P}_{g-p,g-p} \end{bmatrix} \\
&= [\mathbf{P}_1 \ \mathbf{P}_2 \ \cdots \mathbf{P}_{g-p}]
\end{aligned}$$

Since $p \le k-(q-1)\tau$, by using (3.19) and (3.20), we have

$$\mathsf{rank}(\mathbf{P}_{h,h}) = \begin{cases} \alpha & \text{if } 1 \le h \le f-p, \\ \tau\alpha & \text{if } f-p+1 \le h \le g-p, \end{cases} \tag{3.21}$$

and

$$\mathsf{rank}(\mathbf{P}_{h,t}) \leq \begin{cases} \beta & \text{if } 1 \leq t \leq f-p \text{ and } 2 \leq h \leq f-p, \\ \tau\beta & \text{if } 1 \leq t \leq f-p \text{ and } f-p+1 \leq h \leq g-p, \\ R(\alpha,\beta) & \text{if } f-p+1 \leq t \leq g-p, \end{cases} \tag{3.22}$$

Since every diagonal submatrices of $\mathbf{P}_{single}$ is nonsingular, we can utilize the properties discussed in Section 2.2. Substituting $\mathbf{P}_{single}$ for $\mathbf{M}$ of Lemma 3, we have

$$\begin{aligned} t\mathsf{rank}(\mathbf{P}_{single}) \geq & \sum_{i=1}^{g-p} \mathsf{rank}(\mathbf{P}_{i,i}) - \sum_{i=t}^{g-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) \\ & + \max\left(0, (t-1)\mathsf{rank}(\mathbf{P}_{g-p,g-p}) - \sum_{i=1}^{t-1} \bar{T}_{g-p}^{(i)}(\mathbf{P}_{single})\right), \end{aligned} \tag{3.23}$$

for $1 \leq t \leq g-p-1$. The summation of (3.23) for $1 \leq t \leq s$ yields that

$$\begin{aligned} \frac{s(s+1)}{2}\mathsf{rank}(\mathbf{P}_{single}) \geq\ & s\sum_{i=1}^{g-p} \mathsf{rank}(\mathbf{P}_{i,i}) - \sum_{t=1}^{s}\sum_{i=t}^{g-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) \\ & + \sum_{t=1}^{s} \max\left(0, (t-1)\mathsf{rank}(\mathbf{P}_{g-p,g-p}) - \sum_{i=1}^{t-1} \bar{T}_{g-p}^{(i)}(\mathbf{P}_{single})\right), \\ \geq\ & s(k+\tau-p)\alpha - \sum_{i=1}^{g-p} T_i^{(1)}(\mathbf{P}_{single}) \\ & + \sum_{t=1}^{s} \max\left(0, (t-1)\tau\alpha - T_{g-p}^{(1)}(\mathbf{P}_{single})\right), \end{aligned} \tag{3.24}$$

where (3.24) follows from (2.85) and (3.21). By using (3.22), $\sum_{i=1}^{g-p} T_i^{(1)}(\mathbf{P}_{single})$

and $T_{g-p}^{(1)}$ are upper bounded as

$$\begin{aligned}\sum_{i=1}^{g-p} T_i^{(1)}(\mathbf{P}_{single}) &\le \sum_{h=2}^{g-p}\sum_{t=1}^{h-1} \mathsf{rank}(\mathbf{P}_{h,t}) \\ &\le \frac{1}{2}(k-p-(q-1)\tau)(k-p+(q+1)\tau-1)\beta \\ &\quad +\frac{q(q-1)}{2}R(\alpha,\beta). \end{aligned} \tag{3.25}$$

and

$$T_i^{(g-p)}(\mathbf{P}_{single}) \le (q-1)R(\alpha,\beta)+\tau(k-p-(q-1)\tau)\beta. \tag{3.26}$$

By substituting (3.25), (3.26) and using the fact that

$$\begin{aligned} B &= n\alpha - \mathsf{rank}(\mathbf{H}) \\ &\le n\alpha - \mathsf{rank}(\mathbf{H}_{single}) \\ &\le n\alpha - \mathsf{rank}(\mathbf{P}_{single}), \end{aligned} \tag{3.27}$$

we can find the upper bound of B as

$$\begin{aligned}\frac{s(s+1)}{2}B &\le \frac{s(s-1)}{2}k\alpha + sp\alpha + \frac{q(q-1)}{2}R(\alpha,\beta) \\ &\quad +\frac{1}{2}(k-p-(q-1)\tau)(k-p+(q+1)\tau-1)\beta \\ &\quad +\sum_{t=2}^{s}\min\left((t-1)\tau\alpha, (q-1)R(\alpha,\beta)+\tau(k-p-(q-1)\tau)\beta\right), \end{aligned} \tag{3.28}$$

which is equivalent to (1.6).

3.4.2 Proof of Theorem 3

Assume that $\mathbf{P}_{single}$ is constructed for $q = 1$. Since $q = 1$ and $g = k + 1$, we have

$$\mathsf{rank}(\mathbf{P}_{h,h}) = \begin{cases} \alpha & \text{if } 1 \leq h \leq k-p, \\ \tau\alpha & \text{if } h = k-p+1, \end{cases} \tag{3.29}$$

and

$$\mathsf{rank}(\mathbf{P}_{h,t}) \leq \beta \tag{3.30}$$

for $1 \leq t \leq k-p$ and $t < h \leq k-p$. For the case of $h = k-p+1$, $\mathbf{P}_{h,t}$ is defined as

$$\mathbf{P}_{h,t} = \left[\hat{\mathbf{H}}_{t+p}^{k+1} \ \cdots \ \hat{\mathbf{H}}_{t+p,}^{d+1}\right]^T \tag{3.31}$$

for $1 \leq t \leq k$.

By substituting $\mathbf{P}_{single}$ for $\mathbf{M}$ of Lemma 3, the lower bound of $\mathsf{rank}(\mathbf{P}_{single})$ is expressed as

$$\begin{aligned} t\mathsf{rank}(\mathbf{P}_{single}) &\geq \sum_{i=1}^{k-p+1} \mathsf{rank}(\mathbf{P}_{i,i}) - \sum_{i=t}^{k-p+1} \bar{T}_i^{(t)}(\mathbf{P}_{single}) \\ &\quad + \max\left(0, (t-1)\mathsf{rank}(\mathbf{P}_{k-p+1,k-p+1}) - \sum_{i=1}^{t-1} \bar{T}_{k-p+1}^{(i)}(\mathbf{P}_{single})\right), \\ &\geq (k+t\tau-p)\alpha - \sum_{i=t}^{k-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) - \sum_{i=1}^{t} \bar{T}_{g-p}^{(i)}(\mathbf{P}_{single}), \end{aligned} \tag{3.32}$$

for $1 \leq t \leq k-p$. By using the relation (3.27), the lower bound of $t\mathsf{rank}(\mathbf{P}_{single})$

can be converted to an upper bound of B as

$$
\begin{aligned}
tB &\leq ((t-1)k+p)\alpha + \sum_{i=t}^{k-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) + \sum_{i=1}^{t} \bar{T}_{k-p+1}^{(i)}(\mathbf{P}_{single}) \\
&\leq ((t-1)k+p)\alpha + \sum_{i=t}^{k-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) + \bar{T}_{k-p+1}^{(1)}(\mathbf{P}_{single}) \qquad (3.33) \\
&\leq ((t-1)k+p)\alpha + \sum_{i=t}^{k-p} \bar{T}_i^{(t)}(\mathbf{P}_{single}) + \tau(k-p)\beta \qquad (3.34)
\end{aligned}
$$

where (3.33) follows from (2.85).

For $t = 1$, (3.34) is reduced to

$$
\begin{aligned}
B &\leq p\alpha + \sum_{i=1}^{k-p} \bar{T}_i^{(1)}(\mathbf{P}_{single}) + \bar{T}_{k-p+1}^{(i)}(\mathbf{P}_{single}) \\
&= p\alpha + \sum_{i=1}^{k-p} \bar{T}_i^{(1)}(\mathbf{P}_{single}) + \mathsf{rank}\left([\mathbf{P}_{k-p+1,1} \ \cdots \ \mathbf{P}_{k-p+1,k-p}]\right) \qquad (3.35) \\
&= p\alpha + \sum_{i=1}^{k-p} \bar{T}_i^{(1)}(\mathbf{P}_{single}) \\
&\quad + \mathsf{rank}\left(\begin{bmatrix} \left(\mathbf{H}^{k+1}_{1+p,[d+1]\setminus\{1+p\}}\right)^T & \cdots & \left(\mathbf{H}^{k+1}_{k,[d+1]\setminus\{k\}}\right)^T \\ \vdots & \ddots & \vdots \\ \left(\mathbf{H}^{d+1}_{1+p,[d+1]\setminus\{1+p\}}\right)^T & \cdots & \left(\mathbf{H}^{d+1}_{k,[d+1]\setminus\{k\}}\right)^T \end{bmatrix}\right) \qquad (3.36) \\
&\leq p\alpha + \sum_{i=1}^{k-p} \bar{T}_i^{(1)}(\mathbf{P}_{single}) \qquad (3.37) \\
&\quad + \tau\mathsf{rank}\left(\left[\left(\mathbf{H}^{k+1}_{1+p,[d+1]\setminus\{1+p\}}\right)^T \ \cdots \ \left(\mathbf{H}^{k+1}_{k,[d+1]\setminus\{k\}}\right)^T\right]\right) \qquad (3.38) \\
&\leq p\alpha + \left(\frac{2\tau}{k-p-1} + 1\right) \sum_{i=1}^{k-p} \bar{T}_i^{(1)}(\mathbf{P}_{single}) \qquad (3.39)
\end{aligned}
$$

where (3.38) and (3.39) follow from the property of symmetric regenerating codes. The summation of (3.34) for $2 \le t \le s$ yiels that

$$\left(\frac{s(s+1)}{2}-1\right)B \le \left(\frac{s(s-1)k}{2}+(s-1)p\right)\alpha+(s-1)\tau(k-p)\beta + \sum_{t=2}^{s}\sum_{i=t}^{k-p}\bar{T}_i^{(t)}(\mathbf{P}_{single}). \quad (3.40)$$

Since

$$\begin{aligned}\sum_{i=1}^{k-p}\bar{T}_i^{(1)}(\mathbf{P}_{single})+\sum_{t=2}^{s}\sum_{i=t}^{k-p}\bar{T}_i^{(t)}(\mathbf{P}_{single}) &= \sum_{t=1}^{s}\sum_{i=t}^{k-p}\bar{T}_i^{(t)}(\mathbf{P}_{single}) \\ &\le \sum_{i=t}^{k-p}T_i^{(1)}(\mathbf{P}_{single}) \quad (3.41)\\ &\le \frac{1}{2}(k-p)(k-p-1)\beta \quad (3.42)\end{aligned}$$

where (3.41) follows from (2.85), by adding (3.39) and $\left(\frac{2\tau}{k-p-1}+1\right)$ times of (3.40), we have

$$\begin{aligned}&\left(\left(\frac{s(s+1)}{2}-1\right)\left(\frac{2\tau}{k-p-1}+1\right)+1\right)B\\ &\le \left(\frac{2\tau}{k-p-1}+1\right)\left(\frac{s(s-1)}{2}k+(s-1)p\right)\alpha+p\alpha\\ &\quad+\left(\frac{2\tau}{k-p-1}+1\right)(s-1)\tau(k-p)\beta\\ &\quad+\left(\frac{2\tau}{k-p-1}+1\right)\left(\sum_{i=1}^{k-p}\bar{T}_i^{(1)}(\mathbf{P}_{single})+\sum_{t=2}^{s}\sum_{i=t}^{k-p}\bar{T}_i^{(t)}(\mathbf{P}_{single})\right)\\ &\le \left(\left(\frac{2\tau}{k-p-1}+1\right)\left(\frac{s(s-1)}{2}k+(s-1)p\right)+p\right)\alpha\\ &\quad+\left(\frac{2\tau}{k-p-1}+1\right)\left((s-1)\tau(k-p)+\frac{1}{2}(k-p)(k-p+1)\right)\beta,\end{aligned} \quad (3.43)$$

which is equivalent to (1.8).

3.5 Performance Evaluation

In this section, we provide a few examples to illustrate how the new outer bounds are tight compared to the other existing outer bounds and the cutset bound (1.2). In Figure 3.1 and 3.2, we have plotted the performance of the proposed outer bounds for different k when $n = 11$ and $d = 10$, together with the cutset bound and two other existing outer bounds [20, 35]. In Figure 3.1, the tradeoff curves are illustrated on the $\alpha - \gamma$ plane for the case of $k = 4, 5, 6, 7$. The sub-bound 2 (Theorem 3) performs better than the other ones, while in small region near the MSR points it is worse than the outer bound proposed in [20]. On the other hands, In Figure 3.2, we can find that the performance of sub-bound 1 (stated in Theorem 2) becomes better as k gets larger.

In order to check their difference clearer, we provide Figure 3.3 and 3.4, where the cases of extremely low and high rates are described. In Figure 3.3, the examples of extremely high rates is illustrated. As k/n goes to 1 with fixed $\tau = n - k = 3$, the sub-bound 1 in Theorem 2 and the existing inner bound [24] converge to the same point. In addition, the convergence point of the two curves is definitely located far away from the functional repair S-B tradeoff. This shows the existence of performance difference between the exact and functional repair model. In Figure 3.4, the examples of extremely low rates is illustrated. For a given $k = 10$, as n gets larger, the sub-bound 2 stated in

Theorem 3 looks converging to the space-sharing line, which is the trivial inner bound.

To summarize, in high rates where $k/n \cong 1$, the sub-bound 1 is superior than others, and in low rates when k is much smaller than n, the sub-bound 2 is tighter than other bounds. In the both of the extreme low and high rates, each bound gets closer to the optimal S-B tradeoff.

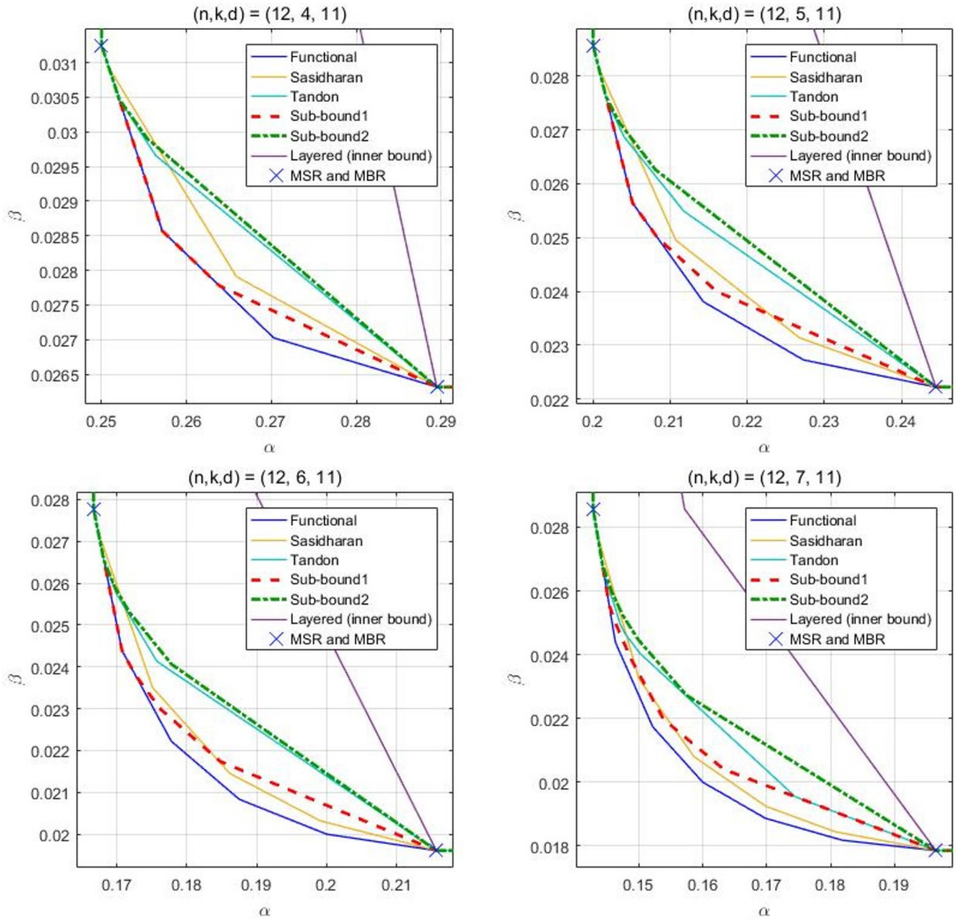

Figure 3.1: Comparison of functional-repair storage-bandwidth tradeoff (1.2), the outer bounds of [20,35], and the proposed outer bounds for various (n, k, d) values. Given fixed $n = d + 1 = 12$, $k = 4, 5, 6$, and 7 is used.

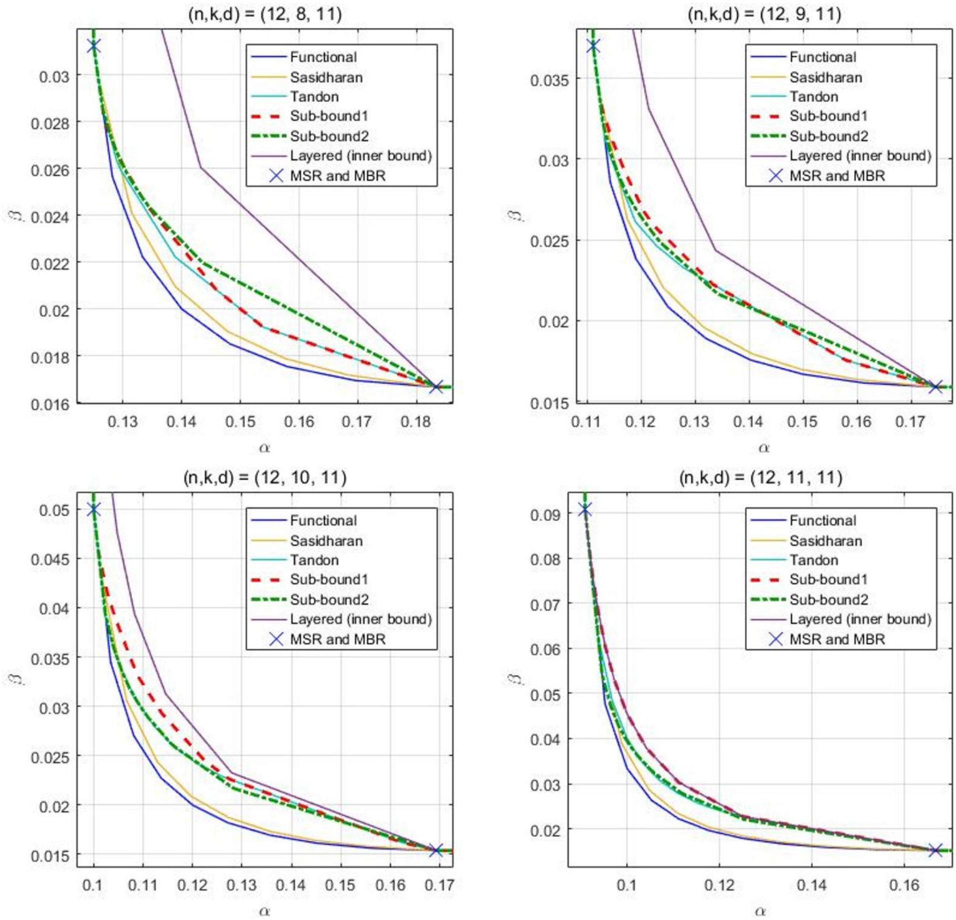

Figure 3.2: Comparison of functional-repair storage-bandwidth tradeoff (1.2), the outer bounds of [20,35], and the proposed outer bounds for various (n, k, d) values. Given fixed $n = d + 1 = 12$, $k = 8, 9, 10$, and 11 is used.

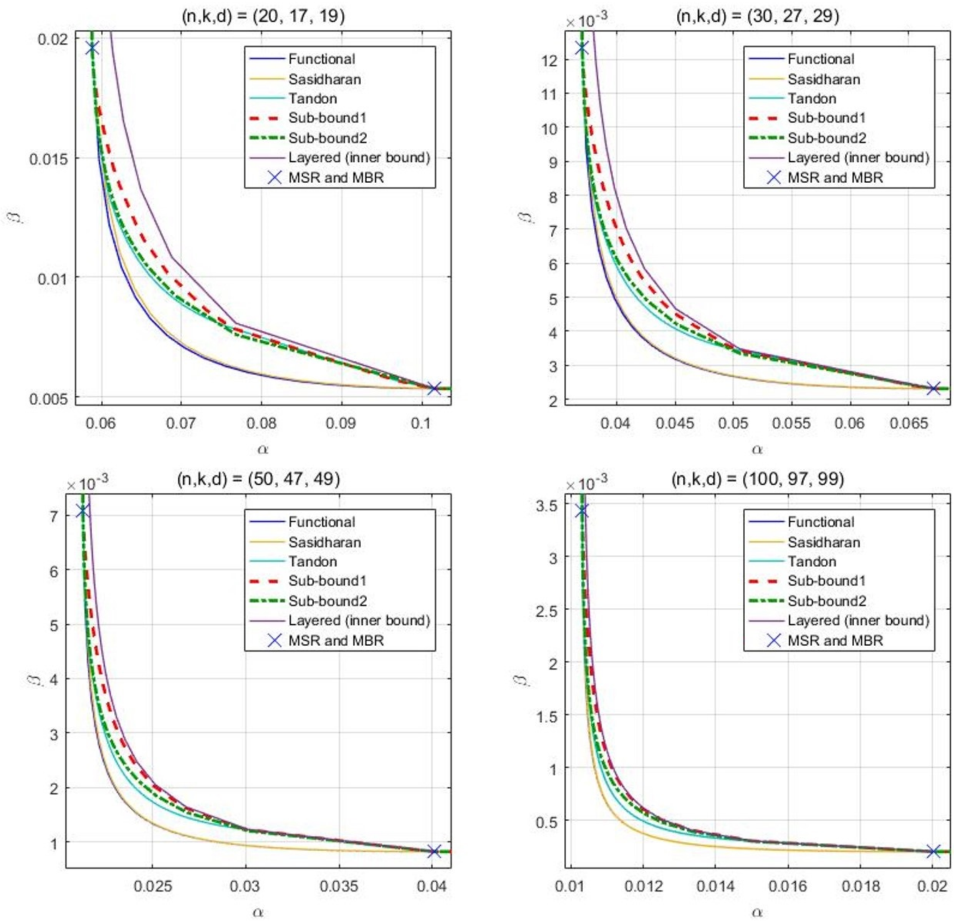

Figure 3.3: Comparison of several bounds on the exact-repair S-B tradeoff in extremely high rates. Given fixed $n - k = 3$, $n = 20, 30, 50$, and 100 is used.

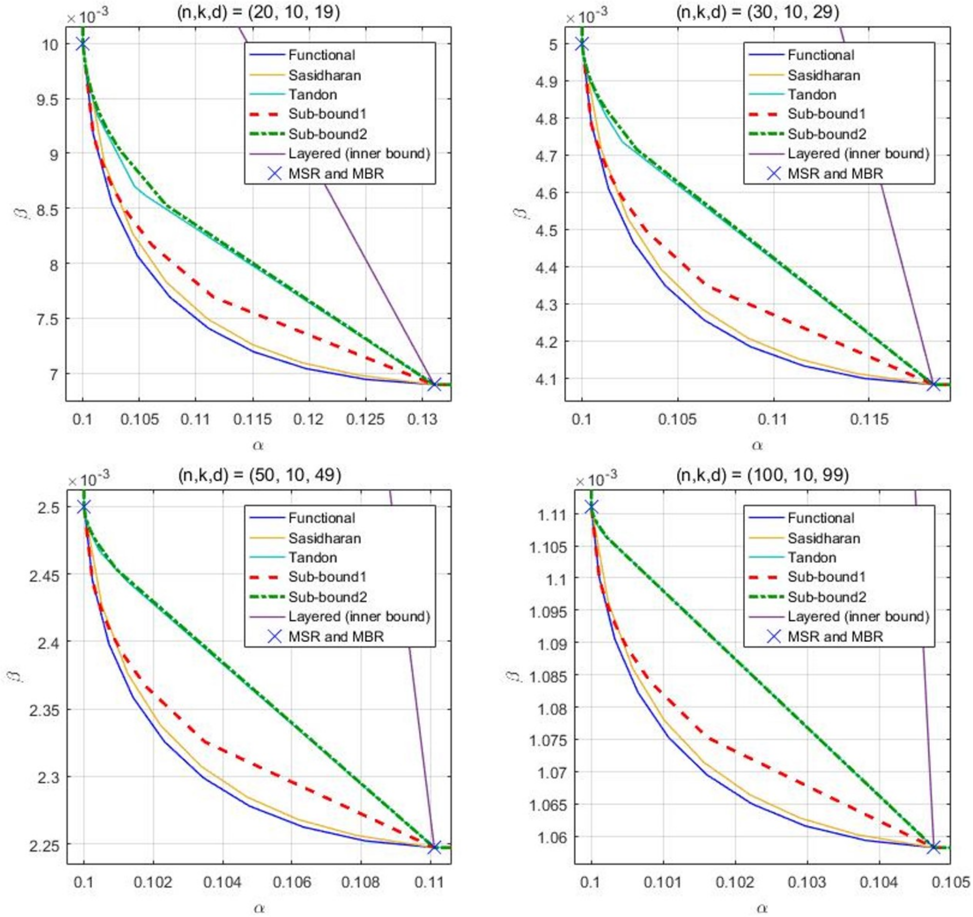

Figure 3.4: Comparison of several bounds on the exact-repair S-B tradeoff in extremely low rates. Given fixed $k = 10$, $n = 20, 30, 50$, and 100 is used.

Chapter 4

Conclusion

- We proposed an outer bound on the storage-bandwidth tradeoff of linear exact-repair cooperative regenerating codes. The proposed bound is a generalization of the $d = k = n - 1$ case (i.e., $r = 1$) proposed in [23]. In addition, we proposed the conditions that the parity check matrix $\mathbf{H}$ of a linear code must satisfy if the code is a cooperative regenerating code. Although the proposed outer bound is not always effective in arbitrary (n, k, d, r) when compared with the cutset bound (1.1), it becomes more effective as k increases, or r and $d - k$ decrease.

- The second contribution is to propose a new outer bound on the S-B tradeoff of exact-repair linear regenerating codes, where we assumed the case of single repair ($r = 1$). The proposed outer bound for single-repair codes consists of two sub-bounds. The two sub-bounds have different tendency according to the code rate k/n. One sub-bound is more effective in high rates ($k/n > \frac{1}{2}$), but the other sub-bound becomes tighter when the code rate is low ($k/n < \frac{1}{2}$). The proposed outer bound asymptotically gets closer to the optimal S-B tradeoff at extreme high or low rates, since the sub-bound 1 becomes closer to the existing inner bound proposed in [24] at extremely high rates and the sub-bound 2 becomes closer to the space-sharing inner bounds at extremely low rates.

Bibliography

[1] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A. G. Dimakis, R. Vadali, S Chen, and D. Borthakur, "XORing elephants: Novel erasure codes for big data," in *Proc. Very Large Data Based (VLDB) Endowment,* vol. 6, no. 5, pp. 325-336, Mar. 2013.

[2] B. Calder, J. Wang, A. Ogus *et al.*, "Windows Azure Storage: A highly available cloud storage service with strong consistency," in *Proc. ACM Symposium on Operating Systems Principles (SOSP)*, Oct. 2011.

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

[4] Y. Hu, Y. Xu, X. Wang, C. Zhan, and P. Li, "Cooperative recovery of distributed storage systems from multiple losses with network coding," in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM)*, May 2007.

[5] X. Wang, Y. Xu, Y. Hu, and K. Ou, "MFR: Multi-loss flexible recovery in distributed storage systems," in *Proc. IEEE Int. Conf. on Comm. (ICC), Capetown, South Africa*, May 2010.

[6] A. Kermarrec, N. Le Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *Proc. International Symposium on Network Coding (NetCod 2011)*, July 2011.

[7] K. W. Shum and Y. Hu, "Cooperative regenerating codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7229-7258, Nov. 2013.

[8] C. Suh and K. Ramchandran, "Exact-repair MDS code construction using interference alignment," *IEEE Trans. Inf. Theory*, vol. 57, no. 3, pp. 1425-1442, Mar. 2011.

[9] 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," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5227-5239, Aug. 2011.

[10] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, "Interference alignment in regenerating codes for distributed storage: Necessity and code constructions," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2134-2158, Apr. 2012.

[11] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, "Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1837-1852, Mar. 2012.

[12] K. W. Shum "Cooperative regenerating codes for distributed storage systems," in *IEEE Int. Conf. Comm. (ICC), Kyoto*, June 2011.

[13] N. Le Scouarnec, "Exact scalar minimum storage coordinated regenerating codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2012.

[14] J. Li and B. Li, "Cooperative repair with minimum-storage regenerating codes for distributed storage," in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM)*, May 2014.

[15] K. W. Shum and Y. Hu, "Exact minimum-repair-bandwidth cooperative regenerating codes for distributed storage systems," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aug. 2011.

[16] S. Jiekak and N. Le Scouarnec, "CROSS-MBCR: Exact minimum bandwidth coordinated regenerating codes," arXiv:1207.0854v1 [cs.IT], Jul. 2012.

[17] A. Wang and Z. Zhang, "Exact cooperative regenerating codes with minimum-repair-bandwidth for distributed storage," in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM)*, Apr. 2013.

[18] C. Tian, "Characterizing the rate region of the (4,3,3) exact-repair regenerating codes," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 5, pp. 967-975, May. 2014.

[19] C. Tian, "A note on the rate region of exact-repair regenerating codes," *CoRR,* abs/1503.00011, Mar. 2015. Available: http://arxiv.org/abs/1503.00011

[20] B. Sasidharan, K. Senthoor, and P. V. Kumar, "An improved outer bound on the storage-repair-bandwidth tradeoff of exact-repair regenerating codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, June 2014.

[21] I. M. Duursma, "Outer bounds for exact repair codes," *CoRR*, abs/1406.4852, June 2014. Available: http://arxiv.org/abs/1406.4852

[22] I. M. Duursma, "Shortened regenerating codes," *CoRR*, abs/1505.00178, MAy 2015. Available: http://arxiv.org/abs/1505.00178

[23] N. Prakash and N. Krishnan, "The storage-repair-bandwidth trade-off of exact repair linear regenerating codes for the case $d = k = n - 1$," in *Proc.*

IEEE Int. Symp. Inf. Theory (ISIT), June 2015.

[24] C. Tian, B. Sasidharan, V. Aggarwal, V. A. Vaishampayan, and P. V. Kumar, "Layered, exact-repair regenerating codes via embedded error correction and block designs," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1933-1947, Apr. 2015.

[25] F. Oggier and A. Datta, "Self-repairing homomorphic codes for distributed storage systems," in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM)*, Apr. 2011.

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

[27] D. Papailiopoulos and A. Dimakis, "Locally repairable codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2012.

[28] V. Guruswami and M. Wootters, "Repairing Reed-Solomon codes," in *Proc. Annu. Symp. Theory Comput. (STOC)*, June 2016.

[29] H. Dau, I. Duursma, H. M. Kiah, and O. Milenkovic, "Repairing Reed-Solomon codes with multiple erasures," *CoRR*, vol. abs/1612.0136, Dec. 2016. Available: http://arxiv.org/abs/1612.0136

[30] V. Aggarwal, C. Tian, V. A. Vaishampayan, and Y. F. R. Chen. "Distributed data storage systems with opportunistic repair." in *Proc. IEEE Int. Conf. on Computer Commun. (INFOCOM)*, Apr. 2014.

[31] A. Kermarrec, N. Le Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *Proc. IEEE Int. Symp. Network Coding (NetCod)*, Jul. 2011.

[32] D. Borthakur, "HDFS architecture guide," *Hadoop Apache Project*, 2008.

[33] V. R. Cadambe, S. A. Jafar, H. Maleki, K. Ramchandran, and C. Suh "Asymptotic interference alignment for optimal repair MDS codes in distributed storage," *IEEE Trans. Inf. Theory*, vol. 59, no. 5, pp. 2974-2987, May 2013.

[34] S. Mohajer and R. Tandon, "New bounds on the (n, k, d) storage systems with exact repair," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, June 2015.

[35] B. Sasidharan, N. Prakash, M. N. Krishnan, M. Vajha, K. Senthoor, and P. V. Kumar, "Outer bounds on the storage-repair bandwidth tradeoff of exact-repair regenerating codes," *CoRR,* abs/1606.04467, June 2015. Available: http://arxiv.org/abs/1606.04467

[36] "Information theory inequality prover," http://user-www.ie.cuhk.edu.hk/ ITIP/, accessed: 2016-Jun-02.

[37] S. Goparaju, S.E. Rouayheb, and R. Calderbank, "New codes and inner bounds for exact repair in distributed storage systems," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, June 2014.

[38] T. Ernvall, S. El Rouayheb, C. Hollanti, and H. V. Poor, "Capacity and security of heterogeneous distributed storage systems. in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, July 2013.

[39] B. Sasidharan and P. V. Kumar, "High-rate regenerating codes through layering," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, July 2013.

초 록

최근 SNS나 클라우드 서비스의 사용량 증가와 더불어, 대규모의 데이터를 네트워크상에 효율적이고 안정적으로 저장할 수 있는 분산 저장 시스템(distributed storage system)에 대한 연구가 활발하게 진행되고 있다. 분산 저장 시스템은 대규모의 데이터 파일을 네트워크로 연결된 다수의 노드에 분산적으로 저장하는 시스템을 말한다. 일부의 노드가 손실되었을 때, 손실된 노드는 다른 생존한 노드들로부터 전송받은 정보를 이용하여 복구될 수 있어야 한다. 이러한 복구 과정에서 필요한 총 정보량인 복구 대역폭(repair bandwidth)을 최소화하는 것은 분산 저장시스템의 중요한 성능 지표중 하나이다. 협력 재생 부호(Cooperative regenerating codes)는 높은 복구 대역폭을 최소화하는 erasure code의 일종이다. (n, k, d, r)-협력 재생 부호는 총 n개의 저장소 노드 중 일부의 k개의 노드에 저장된 정보만으로 원래의 파일을 복구할 수 있는 기능과 r개의 노드 손실이 발생했을때, 임의의 d개의 생존한 노드들로부터 정보를 전송받아 복구될 수 있는 기능을 가진다.

이 때, 재생 부호의 각 노드별 저장량 α와 복구 대역폭 γ는 일반적으로 상충관계에 놓여 있음이 알려져 있다. 하지만 새롭게 복구된 노드가 기존 노드와 다른 정보를 가지는 것을 허용하는 기능 복구(functional repair) 모델의 경우, 이 상충관계가 완벽히 밝혀져 있으나, 손실되기 전과 완전히 동일한 노드로의 복구를 요구하는 동일 복구(exact repair) 모델의 경우, 이 상충관계가 명확히 밝혀져

있지 않다. 본 논문에서는 동일 복구 모델의 상충 관계에 대한 두 종류의 외부 경계(outer bound)를 제시한다. 상충 관계의 외부 경계는 기능 복구 부호로는 가능하지만, 동일 복구 부호로는 설계가 불가능한 (α, γ) 동작점들을 제시한다.

첫 번째 외부 경계는 일반적인 (n, k, d, r) 파라미터를 가지는 협력 재생 부호를 가정하여 유도되었다. 이 외부 경계는 $d = k = n - 1$, $r = 1$을 만족하는 경우에 한하여 최적의 상충관계를 밝힌 Prakash 등의 연구 결과를 일반화한 것으로 볼 수 있다. 첫 번째 외부 경계는 k가 크거나 r이 작거나 k와 d가 비슷한 조건 하에서 더 좋은 성능을 보임을 확인할 수 있다.

두 번째 외부 경계는 한 번에 한 개의 손실된 노드만을 복구하는 경우로 한정하였을 때를 고려한다. 두 번째 외부 경계는 두 개의 독립적인 부경계(sub-bound)의 합집합으로 표현된다. 두 가지의 부경계들은 각각 성능이 좋아지는 조건이 다름을 실험을 통해 확인할 수 있다. 첫 번째 부경계는 본 논문에서 첫 번째로 제안된 외부 경계와 비슷하게 k/n으로 정의되는 코드의 부호화율이 1에 가까울수록 더 좋은 성능을 보이며, 두 번째 부 경계는 반대로 부호화율이 낮아질때 다른 기존의 외부경계들보다 더 좋은 성능을 보임을 확인할 수 있다.

주요어: 분산 저장 시스템, 재생 부호, 협력 재생 부호, 복구 대역폭, 동일 복구 모델

학번: 2013-30256