

Dual-Code Bounds on Multiple Concurrent (Local) Data Recovery

Gianira N. Alfarano^{*1}, Alberto Ravagnani^{†2}, and Emina Soljanin^{‡3}

¹University of Zurich, Switzerland

²Eindhoven University of Technology, the Netherlands

³Rutgers University, NJ, USA

Abstract

We are concerned with linear redundancy storage schemes regarding their ability to provide concurrent (local) recovery of multiple data objects. This paper initiates a study of such systems within the classical coding theory. We show how we can use the structural properties of the generator matrix defining the scheme to obtain a bounding polytope for the set of data access rates the system can support. We derive two dual distance outer bounds, which are sharp for some large classes of matrix families.

1 Introduction

Distributed computing systems rely on their storage layers to provide data access services for executing applications. Thus, the system’s overall performance depends on the underlying storage system’s data access performance. Distributed storage systems strive to maximize the number of concurrent data access requests they can support with fixed resources. Replicating data objects according to their relative popularity and access volume helps achieve this goal. However, these quantities are often unpredictable. In emerging applications such as edge computing, the expected number of users and their data interests fluctuate, and data storage schemes should support such dynamics [1]. Erasure-coding has emerged as an efficient and robust form of redundant storage, which can flexibly handle skews in the request rates.

Recent work on redundant distributed storage access introduced the notion of the *service rate region* of a redundancy scheme that includes all data access requests that the system can serve [2–7]. To understand this concept, consider a distributed system that stores k different data objects by encoding them into n and storing the n coded objects on n different nodes. Each of the n nodes can serve requests at a rate μ (i.e., has service capacity μ). The system can serve requests to access the k data objects that arrive at rates $\lambda_1, \lambda_2, \dots, \lambda_k$ if each

^{*}G. N. Alfarano is supported by Swiss Science Foundation grant n. 188430.

[†]A. Ravagnani is in part supported by Dutch Research Council grant VI.Vidi.203.045.

[‡]E. Soljanin is in part supported by NSF Award CIF-2122400.

request can be routed to a group of nodes that can jointly fulfill the request, and the total request rate allocated to each node does not exceed its service capacity μ . We call the set of such request vectors $(\lambda_1, \lambda_2, \dots, \lambda_k)$ the *service rate region* of a coded distributed system.

The overview paper [2] postulated the service rate region as an important consideration in the design of erasure-coded distributed systems. It highlights several open problems that can be grouped into two broad threads: 1) characterizing the service rate region of a given code and finding the optimal request allocation, and 2) designing the underlying erasure code for a given service rate region. The paper argued that the presented problems not only require expertise from different areas, but have also already been addressed in those areas in some special forms and under different names. Moreover, it explained how some problems associated with the service rate region generalize previously studied distributed problems such as batch codes, codes with locality and availability, and private information retrieval [8–16].

The numerous open problems described in [2] (in both groups mentioned above) could be seen as performance analysis and networking problems as well as coding theory and data allocation problems. These problems could be addressed by a wide variety of scientists according to their interests and expertise.

Our contribution. The goal of this paper is to initiate a study of redundancy schemes within the classical coding theory. We focus on characterizing the service rate region of a storage scheme defined by a rank k generator matrix G , which is a convex polytope in \mathbb{R}^k . Our ultimate goal is to establish a series of inequalities that, when combined, cut out the service region. This paper makes a first step towards this goal by showing how some structural properties of G can be used to find a polytope that contains the service rate region, giving an *outer bound* for the latter. More precisely, we establish a Total Capacity Bound for the service rate region determined by a matrix G , as well as two Dual Distance Bounds that take into account different structural properties of G . We also show that the *bounding* polytope we find coincides with the service rate region in some special cases, e.g. for some large classes of MDS codes.

This paper is organized as follows. Sec. 2 defines the problem. Sec. 3 shows some properties of the recovery sets of a linear redundancy scheme and introduces a way to compare storage schemes. Sec. 4 derives the Total Capacity Bound for the service rate region and two Dual Distance Bounds. Sec. 5 outlines future work plans.

2 Distributed Coded Recovery Systems

In this section we establish the notation for the rest of the paper, define distributed coded systems and their service rate region.

Notation 2.1. Throughout the paper, \mathbb{F}_q denotes the finite field with q elements where q is a prime power. We work with integers $n > k \geq 2$, a real number $\mu \geq 1$, and a fixed matrix $G \in \mathbb{F}_q^{k \times n}$ of rank k . We assume that G has no all-zero column and denote by G^j its j th column.

We consider coded distributed systems where k data objects are linearly encoded into n objects stored on n servers. Each server stores exactly one object and the objects are elements of \mathbb{F}_q .

Such a distributed coded system is fully specified by a rank k matrix $G \in \mathbb{F}_q^{k \times n}$, which we call the **generator matrix** of the system. If $(x_1, \dots, x_k) \in \mathbb{F}_q^k$ is the k -tuple of objects to be stored, then the j th server stores the j th component of the vector

$$(x_1, \dots, x_k) \cdot G \in \mathbb{F}_q^n.$$

We say that the matrix G is **systematic** if its first k columns form the identity $k \times k$ matrix.

We consider k -tuples $(\lambda_1, \dots, \lambda_k) \in \mathbb{R}^k$ of rate requests to access the k objects. More precisely, λ_i is the rate request for the i th object. Each of the n servers can serve request at the rate of at most $\mu \geq 1$. The parameter μ is called the server's **capacity**.

Each user gets assigned to a set of servers that, together, allow recovering the desired object. An object can be recovered from different server sets, which motivates the following terminology.

Definition 2.2. For $i \in \{1, \dots, k\}$, let

$$\mathcal{R}_i^{\text{all}} := \{R \subseteq \{1, \dots, n\} \mid e_i \in \langle G^j \mid j \in R \rangle\},$$

where $e_i \in \mathbb{F}_q^k$ denotes the i th standard basis vector and $\langle G^j \mid j \in R \rangle$ is the span of the columns of G indexed by R . The elements of $\mathcal{R}_i^{\text{all}}$ are the **recovery sets** for the i th object.

Note that, in the above definition, we have $\mathcal{R}_i^{\text{all}} \neq \emptyset$ for all $i \in \{1, \dots, k\}$. This is a simple consequence of the fact that G has rank k . Moreover, $R \neq \emptyset$ for all $i \in \{1, \dots, k\}$ and $R \in \mathcal{R}_i^{\text{all}}$. We use the superscript “all” to indicate that $\mathcal{R}_i^{\text{all}}$ contains *all* the recovery sets for the symbol i .

Example 2.3. Let $k = 2$, $n = 4$, $q = 3$ and

$$G := \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 2 & 1 \end{pmatrix} \in \mathbb{F}_3^{2 \times 4}.$$

Then, we have

$$\begin{aligned} \mathcal{R}_1^{\text{all}} &= \{\{1\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{2, 3, 4\}\}, \\ \mathcal{R}_2^{\text{all}} &= \{\{2\}, \{1, 3\}, \{1, 4\}, \{3, 4\}, \{1, 3, 4\}\}. \end{aligned}$$

When designing a recovery system starting from G , not all recovery sets need to be considered.

Definition 2.4. A **recovery G -system** is a k -tuple $\mathcal{R} = (\mathcal{R}_1, \dots, \mathcal{R}_k)$ of subsets of $\{1, \dots, n\}$ with $\mathcal{R}_i \subseteq \mathcal{R}_i^{\text{all}}$ and $\mathcal{R}_i \neq \emptyset$ for all $i \in \{1, \dots, k\}$.

The service rate region of a recovery G -system \mathcal{R} is the set of all request rate tuples $(\lambda_1, \dots, \lambda_k) \in \mathbb{R}^k$ that can be served by the system.

Definition 2.5. Let $\mathcal{R} = (\mathcal{R}_1, \dots, \mathcal{R}_k)$ be a recovery G -system. The **service rate region** associated with \mathcal{R} and μ is the set of all $(\lambda_1, \dots, \lambda_k) \in \mathbb{R}^k$ for which there exists a collection of real numbers

$$\{\lambda_{i,R} \mid i \in \{1, \dots, k\}, R \in \mathcal{R}_i\}$$

with the following properties:

$$\left\{ \begin{array}{l} \sum_{R \in \mathcal{R}_i} \lambda_{i,R} = \lambda_i \text{ for } 1 \leq i \leq k, \\ \sum_{i=1}^k \sum_{\substack{R \in \mathcal{R}_i \\ j \in R}} \lambda_{i,R} \leq \mu \text{ for } 1 \leq j \leq n, \\ \lambda_{i,R} \geq 0 \text{ for } 1 \leq i \leq k, R \in \mathcal{R}_i. \end{array} \right. \quad (1)$$

$$\left\{ \begin{array}{l} \sum_{i=1}^k \sum_{\substack{R \in \mathcal{R}_i \\ j \in R}} \lambda_{i,R} \leq \mu \text{ for } 1 \leq j \leq n, \\ \lambda_{i,R} \geq 0 \text{ for } 1 \leq i \leq k, R \in \mathcal{R}_i. \end{array} \right. \quad (2)$$

$$\lambda_{i,R} \geq 0 \text{ for } 1 \leq i \leq k, R \in \mathcal{R}_i. \quad (3)$$

A collection $\{\lambda_{i,R}\}$ of real numbers that satisfy the above three properties is called a **feasible allocation** for (\mathcal{R}, μ) . The service rate region associated with \mathcal{R} and μ is denoted by

$$\Lambda(\mathcal{R}, \mu) \subseteq \mathbb{R}^k.$$

Remark 2.6. We have $\Lambda(\mathcal{R}, \mu) = \mu \cdot \Lambda(\mathcal{R}, 1)$ for any recovery G -system \mathcal{R} . Therefore, we will simply call $\Lambda(\mathcal{R}, 1)$ the **service rate region** of G .

Our main goal is to establish outer bounds for the service rate region $\Lambda(\mathcal{R}, 1)$ of a recovery G -system \mathcal{R} , in the form of a *bounding* polytope.

Example 2.7. Let G be as in Example 2.3. Corollary 4.7 will give us the bounding polytope for $\Lambda(\mathcal{R}^{\text{all}}, 1)$ depicted in Figure 1. The outer bound is sharp.

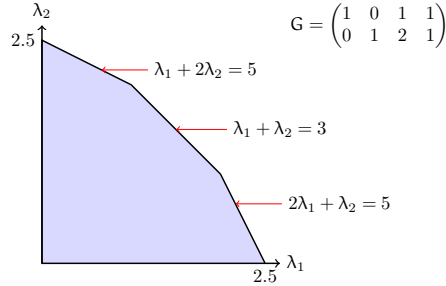


Figure 1: Service rate region for the G -system in Example 2.7. The lines bounding the polytope are obtained from Corollary 4.7.

3 Minimal Recovery Sets

A natural question is how the service rate region changes when changing the recovery G -system. An immediate observation is the following. The result easily follows from Definition 2.5 and is therefore left to the reader.

Proposition 3.1. Suppose that $\mathcal{R} = (\mathcal{R}_1, \dots, \mathcal{R}_k)$ and $\mathcal{R}' = (\mathcal{R}'_1, \dots, \mathcal{R}'_k)$ are recovery G -systems with $\mathcal{R}'_i \subseteq \mathcal{R}_i$ for all $i \in \{1, \dots, k\}$. Then $\Lambda(\mathcal{R}, 1) \supseteq \Lambda(\mathcal{R}', 1)$. In particular, $\Lambda(\mathcal{R}, \mu) \subseteq \Lambda(\mathcal{R}^{\text{all}}, \mu)$ for any recovery G -system \mathcal{R} .

The service rate region, however, does not change when selecting from \mathcal{R}^{all} the recovery sets that are minimal with respect to inclusion, in the following precise sense.

Definition 3.2. A set $R \in \mathcal{R}_i^{\text{all}}$ is *i-minimal* if there is no $R' \in \mathcal{R}_i^{\text{all}}$ with $R' \subsetneq R$. We define \mathcal{R}^{min} as the recovery G -system defined, for all i , by

$$\mathcal{R}_i^{\text{min}} := \{R \in \mathcal{R}_i^{\text{all}} \mid R \text{ is } i\text{-minimal}\}.$$

Proposition 3.3. We have $\Lambda(\mathcal{R}^{\text{min}}, 1) = \Lambda(\mathcal{R}^{\text{all}}, 1)$.

Therefore, when studying the service rate region of the system defined by G one may or may not restrict to the minimal recovery sets.

Proof of Proposition 3.3. The inclusion “ \subseteq ” follows directly from Proposition 3.1. To prove the inclusion “ \supseteq ”, let $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}^{\text{all}}, 1)$. By definition, there exist a collection of real numbers

$$\{\lambda_{i,R} \mid i \in \{1, \dots, k\}, R \in \mathcal{R}_i^{\text{all}}\},$$

satisfying (1)–(3). We will show that $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}^{\text{min}}, 1)$. In order to do so, for each $i \in \{1, \dots, k\}$ and $R \in \mathcal{R}_i^{\text{all}}$ we fix a set $\min_i(R) \in \mathcal{R}_i^{\text{min}}$ such that $R \supseteq \min_i(R)$. For $Q \in \mathcal{R}_i^{\text{min}}$ define

$$\lambda_{i,Q} := \sum_{\substack{R \in \mathcal{R}_i^{\text{all}} \\ \min_i(R) = Q}} \lambda_{i,R}.$$

Then the numbers $\{\lambda_{i,Q} \mid i \in \{1, \dots, k\}, Q \in \mathcal{R}_i^{\text{min}}\}$ satisfy (1)–(3). \square

4 A Total Capacity Bound

We here put forward a simple but powerful idea to obtain outer bounds for the service rate region of a coded system. As an application of this idea, we derive two dual-distance-type bounds in this context.

Lemma 4.1 (Total Capacity Bound). Let \mathcal{R} be a recovery G -system and let $\{\lambda_{i,R}\}$ be a feasible allocation for $(\mathcal{R}, 1)$. We have

$$\sum_{i=1}^k \sum_{R \in \mathcal{R}_i} |R| \lambda_{i,R} \leq n. \quad (4)$$

Proof. The bound is obtained by summing the inequalities in (2), for $1 \leq j \leq n$, \square

Remark 4.2. The Total Capacity Bound of Lemma 4.1 put in evidence the cardinality of the recovery sets. It can be applied directly when a lower bound for this quantity is known. More precisely, if every recovery set of a G -system \mathcal{R} has size at least M , then every $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$ satisfies

$$\lambda_1 + \dots + \lambda_k \leq n/M.$$

Lemma 4.1 motivates us to determine the properties of G that determine the cardinalities of the recovery sets, and how they shape the service rate region. This is the program we initiate in this paper.

4.1 First Dual Distance Bound

We next establish a connection between the recovery sets of a systematic coded system and the codewords of the code whose parity-check matrix is G . By combining this with the Total Capacity Bound, we obtain our First Dual Distance Bound.

Notation 4.3. In the sequel, we denote by \mathcal{C} the linear code generated by G , and by \mathcal{C}^\perp , its dual code. Note that \mathcal{C} is an $[n, k]_q$ code. We let d^\perp be the minimum distance of \mathcal{C}^\perp . Recall that the (**Hamming**) **support** of a vector $x \in \mathbb{F}_q^n$ is $\sigma(x) := \{1 \leq i \leq n \mid x_i \neq 0\}$.

Proposition 4.4. Suppose that G is systematic. Let $R \subseteq \{1, \dots, n\}$ and $i \in \{1, \dots, k\}$. Then $R \in \mathcal{R}_i^{\text{all}}$ if and only if $R = \{i\}$ or there exists a codeword $x \in \mathcal{C}^\perp$ with $\sigma(x) \subseteq R \cup \{i\}$ and $i \in \sigma(x)$.

Proof. By definition, $R \in \mathcal{R}_i^{\text{all}}$ if and only if the span of the columns of G indexed by $R \cup \{i\}$ contains e_i . This happens if and only if $R = \{i\}$ or there exists a linear combination of the columns of G indexed by R , where the i th column of G is taken with a nonzero coefficient, that gives zero. The latter condition is equivalent to the existence of a codeword $x \in \mathcal{C}^\perp$ with $i \in \sigma(x)$ and $\sigma(x) \subseteq R$. \square

Corollary 4.5. If G is systematic, $i \in \{1, \dots, k\}$ and $R \in \mathcal{R}_i^{\text{all}}$, then $R = \{i\}$ or $|R| \geq d^\perp - 1$.

We are now ready to present the First Dual Distance Bound.

Theorem 4.6 (First Dual Distance Bound). Suppose that G is systematic and let \mathcal{R} be a recovery G -system. If $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$, then

$$\sum_{i=1}^k \left(\min\{\lambda_i, 1\} + (d^\perp - 1) \max\{0, \lambda_i - 1\} \right) \leq n.$$

Proof. Let $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$ and let $\{\lambda_{i,R}\}$ be a feasible allocation. By Corollary 4.5, for every $i \in \{1, \dots, k\}$ and every set $R \in \mathcal{R}_i$ with $R \neq \{i\}$ we have $|R| \geq d^\perp - 1$. We can therefore rewrite and bound the LHS of (4) as follows:

$$\begin{aligned} & \sum_{i=1}^k \lambda_{i,\{i\}} + \sum_{i=1}^k \sum_{\substack{R \in \mathcal{R}_i \\ R \neq \{i\}}} |R| \lambda_{i,R} \\ & \geq \sum_{i=1}^k \lambda_{i,\{i\}} + (d^\perp - 1) \sum_{i=1}^k \sum_{\substack{R \in \mathcal{R}_i \\ R \neq \{i\}}} \lambda_{i,R} \\ & = \sum_{i=1}^k \lambda_{i,\{i\}} + (d^\perp - 1) \sum_{i=1}^k (\lambda_i - \lambda_{i,\{i\}}) \\ & = (d^\perp - 1) \sum_{i=1}^k \lambda_i - (d^\perp - 2) \sum_{i=1}^k \lambda_{i,\{i\}}. \end{aligned} \tag{5}$$

Since G has no all-zero column, we have $d^\perp \geq 2$. Therefore, since $\lambda_{i,\{i\}} \leq \min\{\lambda_i, 1\}$ for all i , we can further say that the right-hand side of (5) is at least

$$\begin{aligned} & (d^\perp - 1) \sum_{i=1}^k \lambda_i - (d^\perp - 2) \sum_{i=1}^k \min\{\lambda_i, 1\} \\ &= \sum_{i=1}^k \left(\min\{\lambda_i, 1\} + (d^\perp - 1) \max\{0, \lambda_i - 1\} \right), \end{aligned}$$

which, combined with (4), gives the statement. \square

We can apply Theorem 4.6 to families of systematic codes whose dual distance is known. For example, by considering the class of systematic MDS codes we obtain the following result.

Corollary 4.7. Suppose that G is systematic and that it generates an MDS code. Then for all $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$ we have

$$\sum_{i=1}^k \left(\min\{\lambda_i, 1\} + k \cdot \max\{0, \lambda_i - 1\} \right) \leq n.$$

Remark 4.8. The previous corollary on MDS codes is sharp whenever $k \leq n - k$; see [7, Theorem 2].

4.2 Second Dual Distance Bound

The goal of this subsection is to identify and study new, non-classical parameters of the matrix G that play a role in determining the associated service rate region. As an application, we obtain a second outer bound for the service rate region of a coded system, which refines the First Dual Distance Bound of Theorem 4.6 under some assumptions.

Notation 4.9. In the sequel, for all $i \in \{1, \dots, k\}$, we let

$$G_i = \left(G \mid e_i^\top \right) \in \mathbb{F}_q^{k \times (n+1)}$$

be the matrix obtained from G by appending the i th standard basis vector as the $(n+1)$ -th column. Then G_i generates an $[n+1, k]_q$ code, which we denote by $\mathcal{C}_i \subseteq \mathbb{F}_q^{n+1}$. Its dual is \mathcal{C}_i^\perp . For $i \in \{1, \dots, k\}$, we introduce the following sets and parameters:

$$\begin{aligned} \Gamma_i &:= \{\text{wt}(x) \mid x \in \mathcal{C}_i^\perp, n+1 \in \sigma(x)\}, \\ \gamma_i &:= |\Gamma_i| \quad \text{and} \quad \delta_i^1 := \min(\Gamma_i). \end{aligned}$$

Moreover, we let

$$\delta_i^2 := \min\{\text{wt}(x) \mid x \in \mathcal{C}_i^\perp, n+1 \in \sigma(x), \text{wt}(x) > \delta_i^1\},$$

with the convention that $\delta_i^2 = \delta_i^1$ when $\gamma_i = 1$. Finally, we define

$$\omega_i := \frac{|\{x \in \mathcal{C}_i^\perp : n+1 \in \sigma(x), \text{wt}(x) = \delta_i^1\}|}{q-1}.$$

The structural parameters defined in Notation 4.9 play an essential role in determining the corresponding service rate region. In this conference paper, we give preliminary evidence of this with a result that extends the First Dual Distance Bound of Theorem 4.6 to possibly non-systematic matrices.

We start by describing the recovery sets of \mathcal{R}^{all} in terms of the codes \mathcal{C}_i , $i \in \{1, \dots, k\}$. The proof of the next result is similar to the one of Proposition 4.4, and we, therefore, omit it.

Proposition 4.10. Let $R \subseteq \{1, \dots, n\}$ and $i \in \{1, \dots, k\}$. Then $R \in \mathcal{R}_i^{\text{all}}$ if and only if there is $x \in \mathcal{C}_i^\perp$ with $\sigma(x) \subseteq R \cup \{n+1\}$ and $n+1 \in \sigma(x)$. In particular, if $R \in \mathcal{R}_i^{\text{all}}$ then $|R| \geq \delta_i^1 - 1$.

Proposition 4.10 means that there are ω_1 sets $R \in \mathcal{R}_i^{\text{all}}$ with size δ_i^1 . All the other sets have size at least δ_i^2 .

Theorem 4.11 (Second Dual Distance Bound). Let \mathcal{R} be a recovery G -system and let $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$. For each $i \in \{1, \dots, k\}$, define $\ell_i := \min\{\lambda_i, 1\}$. Then

$$\sum_{i=1}^k \left((\delta_i^2 - 1)(\lambda_i - \omega_i \ell_i) + (\delta_i^1 - 1)\omega_i \ell_i \right) \leq n.$$

Proof. Let $(\lambda_1, \dots, \lambda_k) \in \Lambda(\mathcal{R}, 1)$ and let $\{\lambda_{i,R}\}$ be a feasible allocation. By Proposition 4.10 and (1) we have that, for all $i \in \{1, \dots, k\}$,

$$\sum_{\substack{R \in \mathcal{R}_i \\ |R| > \delta_i^1 - 1}} \lambda_{i,R} = \lambda_i - \sum_{\substack{R \in \mathcal{R}_i \\ |R| = \delta_i^1 - 1}} \lambda_{i,R}, \quad (6)$$

where the sum over an empty set is 0. Thus the LHS of (4) can be rewritten as

$$\sum_{i=1}^k \left((\delta_i^1 - 1) \sum_{\substack{R \in \mathcal{R}_i \\ |R| = \delta_i^1 - 1}} \lambda_{i,R} + \sum_{\substack{R \in \mathcal{R}_i \\ |R| > \delta_i^1 - 1}} |R| \lambda_{i,R} \right). \quad (7)$$

Using (6), Proposition 4.10, and the definition of δ_i^2 , we find that the quantity in (7) is at least

$$\begin{aligned} & \sum_{i=1}^k \left((\delta_i^1 - 1) \sum_{\substack{R \in \mathcal{R}_i \\ |R| = \delta_i^1 - 1}} \lambda_{i,R} + (\delta_i^2 - 1) \sum_{\substack{R \in \mathcal{R}_i \\ |R| > \delta_i^1 - 1}} \lambda_{i,R} \right) \\ &= \sum_{i=1}^k \left((\delta_i^2 - 1)\lambda_i - (\delta_i^2 - \delta_i^1) \sum_{\substack{R \in \mathcal{R}_i \\ |R| = \delta_i^1 - 1}} \lambda_{i,R} \right) \\ &\geq \sum_{i=1}^k \left((\delta_i^2 - 1)\lambda_i - (\delta_i^2 - \delta_i^1)\omega_i \ell_i \right), \end{aligned} \quad (8)$$

where the latter inequality follows from the fact that $\delta_i^2 - \delta_i^1 \geq 0$ and $\lambda_{i,R} \leq \ell_i$ for all $i \in \{1, \dots, k\}$. Finally, the last line of (8) is equal to

$$\sum_{i=1}^k \left((\delta_i^2 - 1) (\lambda_i - \omega_i \ell_i) + (\delta_i^1 - 1) \omega_i \ell_i \right),$$

which with (4) and (7) concludes the proof. \square

We illustrate the previous result with two examples where G is not systematic.

Example 4.12 (First-Order Reed-Muller Code). Consider the following non-systematic generator matrix $G \in \mathbb{F}_2^{4 \times 8}$ of the first order Reed-Muller code, namely

$$G := \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Consider the codes generated by G_i , for $i \in \{1, 2, 3, 4\}$. We have that $\delta_1^1 = \delta_2^1 = \delta_3^1 = 3$, $\omega_1 = \omega_2 = \omega_3 = 4$, $\delta_1^2 = \delta_2^2 = \delta_3^2 = 5$, $\delta_4^1 = 2$, $\omega_4 = 1$ and $\delta_4^2 = 4$. Thus Thm 4.11 reads, with $\ell_i = \min\{\lambda_i, 1\}$ for all i ,

$$4\lambda_1 + 4\lambda_2 + 4\lambda_3 - 8\ell_1 - 8\ell_2 - 8\ell_3 + 3\lambda_4 - 2\ell_4 \leq 8. \quad (9)$$

The bound is not sharp in this case. Indeed, suppose that $\lambda_2 = \lambda_3 = 0$.

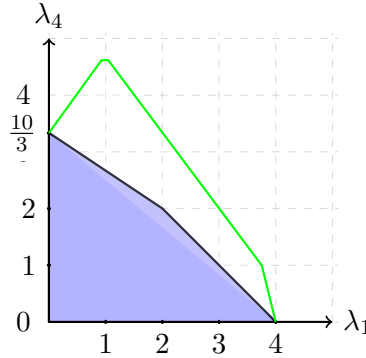


Figure 2: Service rate region for the G -system in Example 4.12, with $\lambda_2 = \lambda_3 = 0$. Eq. (9) is represented by the green line.

Example 4.13. Consider the finite field $\mathbb{F}_8 = \mathbb{F}_2[\alpha]$, where $\alpha^3 + \alpha + 1 = 0$. Let

$$G := \begin{pmatrix} 1 & \alpha & \alpha^4 & \alpha & \alpha^3 \\ \alpha & \alpha^3 & \alpha & \alpha^6 & \alpha^5 \\ \alpha^2 & \alpha & \alpha^4 & \alpha^2 & \alpha^3 \end{pmatrix} \in \mathbb{F}_8^{3 \times 5}.$$

We have $d^\perp = 2$. Moreover, the dual distances of the codes generated by G_i , for $i \in \{1, 2, 3\}$, are all equal to d^\perp . Further, we have $\delta_1^1 = 3$, $\omega_1 = 1$, $\delta_1^2 = 4$, $\delta_2^1 = 3$, $\omega_2 = 2$, $\delta_2^2 = 4$, $\delta_3^1 = 4$, $\omega_3 = 7$, and $\delta_3^2 = 5$. Thus Thm 4.11 reads, with $\ell_i = \min\{\lambda_i, 1\}$ for all i ,

$$3\lambda_1 + 3\lambda_2 + 4\lambda_3 - \ell_1 - 2\ell_2 - 7\ell_3 \leq 5.$$

We conclude this section by showing that Theorem 4.11 implies the First Dual Distance Bound of Theorem 4.6 when G is systematic and $d^\perp \geq 3$.

Proposition 4.14. Suppose that G is systematic and that $d^\perp \geq 3$. Then for all $i \in \{1, \dots, k\}$ we have $\delta_i^1 = 2$, $\omega_i = 1$, and $\delta_i^2 \geq d^\perp$.

Proof. We only prove the result for $i = 1$; the proof for the other indices is identical. We denote by $\pi : \mathbb{F}_q^{n+1} \rightarrow \mathbb{F}_q^n$ the projection onto the first n coordinates. That $\delta_1^1 = 2$ easily follows from the definitions, since the 1st and the $(n+1)$ th columns of G_1 are equal.

We claim that $(1, 0, \dots, 0, -1) \in \mathbb{F}_q^{n+1}$ is the only codeword $x \in \mathcal{C}_1^\perp$ (up to multiples) with $n+1 \in \sigma(x)$ and Hamming weight 2. Let $x = (1, 0, \dots, 0, -1)$ and suppose towards a contradiction that there exists $y \in \mathcal{C}_1^\perp$ with $y_{n+1} = 1$, Hamming weight 2, and linearly independent from x . Then $\pi(x+y) \in \mathcal{C}^\perp$ is non-zero and has weight at most 2, contradicting $d^\perp \geq 3$. This shows that $\omega_1 = 1$.

It remains to show that $\delta_1^2 \geq d^\perp$. Let $x \in \mathcal{C}_1^\perp$ be the codeword defined above.

- We start by showing that $\delta_1^2 \geq 3$. Take $y \in \mathcal{C}^\perp$ with $\text{wt}(y) = d^\perp \geq 3$. Then $(y, 0) + x \in \mathcal{C}_1^\perp$ has weight at least $d^\perp \geq 3$. Not all codewords of \mathcal{C}_1^\perp with $n+1$ in their support have weight 2. Therefore $\delta_1^2 \geq 3$ by definition of δ_1^2 .
- Let $y \in \mathcal{C}_1^\perp$ be a codeword with $\text{wt}(y) = \delta_1^2$ and $n+1 \in \sigma(y)$. By the previous item, $\text{wt}(y) \geq 3$.

If $1 \notin \sigma(y)$, then $\pi(x+y) \in \mathcal{C}$ is a non-zero codeword with the same weight as y . Therefore $\delta_i^2 = \text{wt}(y) \geq d^\perp$. On the other hand, if $1 \in \sigma(y)$ then $z := (y_1 + y_{n+1}, y_2, \dots, y_n) \in \mathcal{C}^\perp$. Since $\text{wt}(y) \geq 3$ we have $z \neq 0$. Therefore $d^\perp \leq \text{wt}(z) \leq \text{wt}(y) = \delta_1^2$. \square

Observe that when G is systematic and $d^\perp = 2$, Theorem 4.11 can be sharper than Theorem 4.6, since it takes into account possibly finer information.

Example 4.15. Let

$$G := \begin{pmatrix} 1 & 0 & 0 & 3 & 5 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 3 \end{pmatrix} \in \mathbb{F}_7^{3 \times 5}.$$

The bound of Theorem 4.6 reads $\lambda_1 + \lambda_2 + \lambda_3 \leq 5$, while the one of Theorem 4.11 reads $2\lambda_1 + 3\lambda_2 + 3\lambda_3 - 2 \sum_{i=1}^3 \min\{\lambda_i, 1\} \leq 5$. It can be shown that the latter region is strictly contained in the former.

The example illustrates that the service rate region of a G -system depends on G . How invertible operations on G change the region is an open problem.

5 Conclusions and Future Work

The service rate region is a new aspect of linear redundancy schemes, which measures their ability to provide simultaneous data recovery. The problem was studied in different frameworks. We initiate a study within the classical coding theory. This approach enabled us to derive some new results and recover some previously known bounds on the service rate region straightforwardly. A linear redundancy system is defined by a matrix akin to a code generator matrix. We believe that coding theory can be instrumental in identifying and using the matrix's structural properties to construct polytopes that contain the service rate region by establishing a series of outer bounds that, when combined, determine the region.

References

- [1] G. Yadgar, O. Kolosov, M. F. Aktas, and E. Soljanin, "Modeling the edge: Peer-to-peer reincarnated," in *2nd USENIX Workshop on Hot Topics in Edge Computing, HotEdge 2019, Renton, WA, USA, July 9, 2019*, I. Ahmad and S. Sundararaman, Eds. USENIX Association, 2019.
- [2] M. Aktas, G. Joshi, S. Kadhe, F. Kazemi, and E. Soljanin, "Service rate region: A new aspect of coded distributed system design," *IEEE Trans. Inform. The.*, Feb. 2022.
- [3] F. Kazemi, S. Kurz, and E. Soljanin, "Efficient storage schemes for desired service rate regions," in *2021 IEEE Information Theory Workshop (ITW)*, Apr. 2021.
- [4] —, "A geometric view of the service rates of codes problem and its application to the service rate of the first order reed-muller code," in *2020 IEEE International Symposium on Information Theory (ISIT)*, June 2020.
- [5] F. Kazemi, E. Karimi, E. Soljanin, and A. Sprintson, "A combinatorial view of the service rates of codes problem, its equivalence to fractional matching and its connection with batch codes," in *2020 IEEE International Symposium on Information Theory (ISIT)*, June 2020.
- [6] S. E. Anderson, A. Johnston, G. Joshi, G. L. Matthews, C. Mayer, and E. Soljanin, "Service capacity region of content access from erasure coded storage," in *IEEE Information Theory Workshop (ITW)*, Nov. 2018.
- [7] M. Aktas, S. E. Anderson, A. Johnston, G. Joshi, S. Kadhe, G. L. Matthews, C. Mayer, and E. Soljanin, "On the service capacity region of accessing erasure coded content," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2017, pp. 17–24.
- [8] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Batch codes and their applications," in *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, L. Babai, Ed., 2004, pp. 262–271.
- [9] A. Riet, V. Skachek, and E. K. Thomas, "Asynchronous batch and PIR codes from hypergraphs," in *IEEE Inform. Theory Workshop, ITW 2018, Guangzhou, China, November 25-29, 2018*. IEEE, 2018, pp. 1–5.

- [10] V. Skachek, *Batch and PIR Codes and Their Connections to Locally Repairable Codes*. Springer International Publishing, 2018.
- [11] A. Fazeli, A. Vardy, and E. Yaakobi, “PIR with low storage overhead: Coding instead of replication,” *CoRR*, vol. abs/1505.06241, 2015. [Online]. Available: <http://arxiv.org/abs/1505.06241>
- [12] A. Fazeli, A. Vardy, and E. Yaakobi, “Codes for distributed pir with low storage overhead,” in *2015 IEEE Internat. Symp. on Inform. Theory (ISIT)*, 2015, pp. 2852–2856.
- [13] T. Ernvall, T. Westerbäck, and C. Hollanti, “Linear locally repairable codes with random matrices,” *CoRR*, vol. abs/1408.0180, 2014. [Online]. Available: <http://arxiv.org/abs/1408.0180>
- [14] T. Ernvall, T. Westerback, and C. Hollanti, “Constructions of optimal and almost optimal locally repairable codes,” in *Wireless Communications, Vehicular Technology, Information Theory and Aerospace Electronic Systems (VITAE), 2014 4th International Conference on*, May 2014, pp. 1–5.
- [15] P. Gopalan, C. Huang, B. Jenkins, and S. Yekhanin, “Explicit maximally recoverable codes with locality,” *IEEE Trans. Inform. Th.*, vol. 60, no. 9, pp. 5245–5256, 2014.
- [16] I. Tamo and A. Barg, “A family of optimal locally recoverable codes,” *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 4661–4676, Aug 2014.