Finite Fields and Their Applications 96 (2024) 102421

Contents lists available at ScienceDirect

Finite Fields and Their Applications

journal homepage: www.elsevier.com/locate/ffa

Private information retrieval from locally repairable databases with colluding servers

Umberto Martínez-Peñas

IMUVa-Mathematics Research Institute, University of Valladolid, Spain

ARTICLE INFO

Article history:
Received 20 September 2021
Received in revised form 4 August 2023
Accepted 15 March 2024
Available online 28 March 2024
Communicated by Gary L. Mullen

MSC:
94A60
94B05
94C99

Keywords:
Distributed storage
Linearized Reed-Solomon codes
Locally repairable codes
Network coding
Private information retrieval

ABSTRACT

We consider information-theoretical private information retrieval (PIR) from a coded database with colluding servers. We target, for the first time, locally repairable storage codes (LRCs). We consider any number of local groups g, locality r, local distance δ and dimension k. Our main contribution is a PIR scheme for maximally recoverable (MR) LRCs based on linearized Reed–Solomon codes, which achieve the smallest field sizes among MR-LRCs for many parameter regimes. In our scheme, nodes are identified with codeword symbols and servers are identified with local groups of nodes. Only locally non-redundant information is downloaded from each server, that is, only r nodes (out of $r+\delta-1$) are downloaded per server. The PIR scheme achieves the (download) rate $R=(N-k-rt+1)/N$, where $N=gr$ is the length of the MDS code obtained after removing the local parities, and for any t colluding servers such that $k+rt\leq N$. For an unbounded number of stored files, the obtained rate is strictly larger than those of known PIR schemes that work for any MDS code. Finally, the obtained PIR scheme can also be adapted when communication between the user and each server is performed via linear network coding, achieving the same rate as previous PIR schemes for this scenario but with polynomial finite field sizes, instead of exponential. Our rates are equal to those of PIR schemes for Reed–Solomon codes, but Reed–Solomon codes are incompatible with the MR-LRC property or linear

E-mail address: umberto.martinez@uva.es.

https://doi.org/10.1016/j.ffa.2024.102421

network coding, thus our PIR scheme is less restrictive in its applications.

1. Introduction

Private information retrieval (PIR), introduced in [6,7], consists in retrieving a file from a database without revealing the index of the retrieved file to the servers, hence keeping the user's preference for a file private. In this work, we consider information-theoretical privacy, meaning that an undesired observer (e.g., the servers) with unbounded computational power may not obtain any information on the file index. Originally [6,7], databases were considered to store files using a *repetition* code, that is, each server stores one copy of each file, and servers were not considered to communicate with each other (*collude*) in order to gain information on the file index.

As it was pointed out in these seminal works, an obvious solution to the PIR problem is to download the entire database. However, this turns out to be wasteful and much higher *download rates*, or simply *rates* (the size of the file divided by the amount of downloaded data), can be achieved when more than one server is used. It was also shown in these seminal works that downloading the whole database is the only solution in the single-server case (for information-theoretical privacy).

As is well-known, databases often suffer from data erasures (due to disk failures). Using a repetition code, that is, storing copies of the same files across multiple servers, yields an unacceptably high overhead (i.e., unacceptably low information rate). PIR from a database where data is stored after being encoded by a non-repetition code (*coded database*) was considered in [2,10,30]. However, the number of servers in these works is either larger than the number of files or grows as the overhead of the storage code decreases, which are not practical scenarios.

Explicit PIR schemes from a database that uses an MDS storage code, and for $\tau \geq 1$ colluding servers, were obtained in [11,36,37,42]. The optimal rate for a fixed number m of stored files is

$$R = \frac{N-k-\tau+1}{N} \cdot \left(1 - \left(\frac{k+\tau-1}{N}\right)^m\right)^{-1}, \tag{1}$$

for the cases $\tau = 1$ [1,32] and $k = 1$ [33], although (1) is not optimal in general [34]. If the MDS storage code has dimension k and length N, *universal* PIR schemes (compatible with any (N,k) MDS storage code) were obtained in [36,37] with rate $R = 1/N$, for $\tau = N-k$, and rate $R = (N-k)/N$ for $\tau = 1$. For $\tau = 1$ or $k = 1$, these rates tend to the optimal rate (1) as the number of files m increases, and in fact they become very close already for moderate values of m (since m appears in the exponent). We also remark that, in practical scenarios, we usually have $m \gg N$.

For $\tau > 1$ and $k > 1$, the optimal rate is unknown in general, but the rate of the PIR scheme in [11],

$$R = \frac{N - k - \tau + 1}{N}, \tag{2}$$

is the highest known (and it is unmatched by other schemes) for an unrestricted number of files and is always strictly larger than that of known universal PIR schemes [36].

However, we will disregard in our rate comparisons the scheme in [11], since it is *more restrictive* as it only works for generalized Reed–Solomon (GRS) storage codes [28], but not for other MDS storage codes, which makes them incompatible with any MR-LRC coded database (except for the trivial cases). More concretely, the scheme in [11] uses the coordinate-wise product, and the only MDS codes that satisfy the desired properties with respect to such products are GRS codes [21]. Furthermore, GRS codes are not LRCs and can never be (except for trivial cases) the MDS codes obtained after removing the local parities of an MR-LRC. This is because GRS codes have linear field sizes and MR-LRCs require super-linear field sizes [15]. Hence the scheme from [11] may not be adapted to be used in an MR-LRC coded database. For similar reasons, the PIR scheme from [11] is also incompatible with linear network coding (see Subsection 5.3 and [38]).

As pointed out in the distributed storage literature, MDS codes are not suitable for large databases, which are increasingly more important due to the spread of Big Data. This is because repairing one single failed node with an MDS code requires contacting and downloading the content of a large number of nodes, resulting in a high repair latency. Locally repairable codes (LRCs), introduced in [14,18] and already applied in practice (by, e.g., Microsoft [16] and Facebook [29]), allow to repair a single erasure (or generally $\delta - 1$ erasures per *local group*, for a *local distance* δ) by contacting at most a number r, called *locality*, of other nodes. Maximally recoverable (MR) LRCs were introduced in [3,13] and are optimal in the following strong sense: Given parameters k, r, δ and number of local groups g, if there is an erasure pattern that an MR-LRC with such parameters cannot correct, then such a pattern cannot be corrected by any other LRC with such parameters. Such patterns can be described easily, see Definitions 1 and 2.

To the best of our knowledge, no work has provided PIR schemes with rates as high as (2) for optimal LRCs, MR-LRCs or MDS codes obtained from puncturing MR-LRCs for general parameters g, r, δ, k, τ and $N = gr$. As explained above, the only PIR schemes for some MDS coded databases with PIR rates as large as (2) are those from [11]. However, these PIR schemes only work for GRS codes, since they make use of coordinate-wise products and GRS codes are the only MDS codes suitable for such products [21]. Moreover, GRS codes can never be the MDS codes obtained from puncturing MR-LRCs (except for trivial cases) since they have linear field sizes and MR-LRCs require super-linear field sizes [15]. We will circumvent the limitations of [11] by making use of coordinate-wise matrix products (see Section 3).

In this work, we provide the first PIR scheme for a class of MR-LRC storage codes that cover general parameters. We consider the MR-LRCs from [24], based on linearized

Reed-Solomon codes [22]. We achieve download rates as in (2), matching GRS storage codes (which cannot be used in our scenario since they cannot be obtained as MDS codes from puncturing MR-LRCs, thus our scheme would be *less restrictive*).

It is worth mentioning that some works not only consider the download rate, but also the upload rate (see [31,40] and the references therein). However, in the case of our scheme, the upload cost may be considered negligible compared to the download one by folding the scheme by a large number (see the discussion before Definition 4). Thus, we do not consider the upload rate in this work. We also mention that PIR schemes for databases coded with Minimum Bandwidth Regenerating (MBR) codes were proposed in [19]. MBR codes are also used for local repair in distributed storage. However, their objective is minimizing the amount of downloaded data instead of the number of contacted nodes, as is the case for LRCs.

We remark that a PIR scheme from Gabidulin storage codes [12] has been recently given in [38]. Gabidulin codes can also be used to construct MR-LRCs [4], and the PIR scheme in [38] can be used for such MR-LRCs. However, the required field size would be at least 2^{gr} (see Subsection 5.3), exponential in g and the code length $N = gr$, in contrast with polynomial field sizes $\max\{r+\delta-1, g\}^r$ for our scheme (note also that r is preferably small). The main objective in [38] is to give a PIR scheme where communication between the user and each server is via linear network coding [20]. We will see in Subsection 5.3 that our PIR scheme can be used in the same scenario, achieving the same rate, but with polynomial field sizes as noted above.

The paper is organized as follows. In Section 2, we formulate general PIR schemes for MR-LRC databases. In Section 3, we develop the mathematical tools for our PIR scheme. In Section 4, we explicitly describe our PIR scheme. Finally, in Section 5, we discuss some further considerations.

2. Private information retrieval from MR-LRC databases

In this section, we describe the *private information retrieval* (PIR) model that we consider in this work. To the best of our knowledge, no general information-theoretical PIR model has yet been proposed for LRC databases.

Let q be a prime power. We will denote by $\mathbb{F}$ an arbitrary field, and by $\mathbb{F}_q$ the finite field with q elements. Usually, we will consider $\mathbb{F} = \mathbb{F}_{q^r}$, where $r \geq 1$ will be the locality of the considered storage codes. We will also denote by $\mathbb{F}^{m\times n}$ the set of $m \times n$ matrices with entries in $\mathbb{F}$, and we denote $\mathbb{F}^n = \mathbb{F}^{1\times n}$. For a positive integer n, we denote $[n] = \{1, 2, \ldots, n\}$. Given $\mathcal{R} \subseteq [n]$, we denote by $\mathbf{c}_{\mathcal{R}} \in \mathbb{F}^{|\mathcal{R}|}$, $A|_{\mathcal{R}} \in \mathbb{F}^{m\times|\mathcal{R}|}$ and $\mathcal{C}_{\mathcal{R}} \subseteq \mathbb{F}^{|\mathcal{R}|}$ the restrictions of a vector $\mathbf{c} \in \mathbb{F}^n$, a matrix $A \in \mathbb{F}^{m\times n}$ and a code $\mathcal{C} \subseteq \mathbb{F}^n$, respectively, to the coordinates indexed by $\mathcal{R}$.

Next, we recall the definitions of *locally repairable codes* [14,18].

Definition 1 *(Locally repairable codes [14,18]).* We say that a code $\mathcal{C} \subseteq \mathbb{F}^n$ is a *locally repairable code* (LRC) with (r, δ) localities if there exists a partition $[n] = \Gamma_1 \cup \Gamma_2 \cup \ldots \cup \Gamma_g$, where $\Gamma_i \cap \Gamma_j = \varnothing$ if $i \neq j$, such that

1. $|\Gamma_j| = r + \delta - 1$, and
2. $\mathrm{d}_H(\mathcal{C}_{\Gamma_j}) \geq \delta$,

for $j = 1, 2, \ldots, g$. The set Γ_j is called the jth local group, r is called the *locality*, and δ is called the *local distance*. A *node* will simply be an index $j \in [n]$.

Note that $n = (r + \delta - 1)g$, but necessarily it must hold that $k = \dim(\mathcal{C}) \leq gr$. Maximally recoverable LRCs, introduced in [3, Def. 2.1] and [13, Def. 6], can correct all information-theoretically correctable erasure patterns for the given locality constraints r, δ, k and g. Such patterns are formed by any $\delta - 1$ erasures per local group, plus any $h = gr - k$ extra erasures anywhere else, as the following definition shows.

Definition 2 *(Maximal recoverability [3,13]).* We say that an LRC $\mathcal{C} \subseteq \mathbb{F}^n$ with (r, δ) localities is *maximally recoverable* (MR) if, for any $\Delta_j \subseteq \Gamma_j$ with $|\Delta_j| = r$, for $j = 1, 2, \ldots, g$, the restricted code $\mathcal{C}_\Delta \subseteq \mathbb{F}^N$ is MDS, where $\Delta = \bigcup_{j=1}^g \Delta_j$ and $N = |\Delta| = gr$. We say for short that $\mathcal{C}$ is an MR-LRC. We will usually call $\mathcal{C}_\Delta \subseteq \mathbb{F}^N$ a *remaining MDS code* of $\mathcal{C}$ (there is one for each choice of Δ_j's).

MR-LRCs can correct more erasure patterns than most LRCs with optimal minimum distance with respect to the bound in [18, Th. 2.1], such as Tamo-Barg codes [39].

Our PIR schemes will work for the MDS codes that remain after puncturing an MR-LRC, as in Definition 2. As explained in Section 1, the PIR schemes from [11] only work for GRS codes, which may not be the MDS code $\mathcal{C}_\Delta$ that remains after puncturing an MR-LRC $\mathcal{C}$ as in Definition 2 since GRS have linear field sizes and MR-LRCs require super-linear field sizes (in the code length).

We will consider collusion patterns formed by unions of local groups. Note that collusion patterns strongly depend on each particular scenario. We now argue why in the LRC case it actually makes more sense to consider collusion patterns formed by unions of local groups than any set of nodes corresponding to codeword symbols:

1. Communication is much more frequent and necessary among nodes inside a local group, as local correction is the most frequent type and global correction is only left to catastrophic erasure patterns. For this reason, local groups could be considered as separate storage units or even be placed geographically apart. In the extreme case $h = gr - k = 0$, MR-LRCs are simply Cartesian products of codes. In this case, no communication across local groups is needed and they could store completely independent data. In contrast, for repetition codes and MDS codes, communication across servers is needed to correct average erasure patterns. For these reasons, we will

consider local groups $\Gamma_j \subseteq [n]$, rather than individual nodes $j \in [n]$, as *corruptable units*. In other words, the *jth server* will be identified with the jth local group. Thus a subset $T \subseteq [g]$ of *colluding servers* is the same as the corresponding $|T|$ local groups and $(r+\delta-1)|T|$ colluding nodes.

2. To help reduce the downloaded amount of data from the jth server (i.e. jth local group, see Item 1), we assume that only r stored symbols from each codeword are downloaded from that server, since the remaining $\delta - 1$ nodes only contain locally redundant information. This means that we consider an MDS code that remains after puncturing the MR-LRC as in Definition 2. As explained above, GRS codes (the MDS codes considered in [11]) cannot come from MR-LRC after puncturing since GRS codes have linear field sizes and MR-LRCs require super-linear field sizes [15], hence the scheme in [11] does not apply to the MR-LRC scenario.

Fix now positive integers b, m and $k \leq N = gr$, and let $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^m \in \mathbb{F}_{q^r}^{b\times k}$ be the m files to be stored. Arrange them as

$$X = \begin{pmatrix} \mathbf{x}^1 \\ \vdots \\ \mathbf{x}^m \end{pmatrix} \in \mathbb{F}_{q^r}^{bm\times k}.$$

Consider an arbitrary MR-LRC $\mathcal{C}_{glob} \subseteq \mathbb{F}_{q^r}^n$ (the *global code*) with a generator matrix of the form

$$G_{glob} = G_{out}\,\mathrm{Diag}_g(A) \in \mathbb{F}_{q^r}^{k\times n}, \tag{3}$$

where $n = g(r+\delta-1)$, $N = gr$, $G_{out} \in \mathbb{F}_{q^r}^{k\times N}$ is a generator matrix of some k-dimensional *outer code* $\mathcal{C}_{out} \subseteq \mathbb{F}_{q^r}^N$, $A \in \mathbb{F}_q^{r\times(r+\delta-1)}$ is a generator matrix of an $(r+\delta-1, r)$ MDS code (the *local code*), and $\mathrm{Diag}_g(A) = \mathrm{Diag}(A, A, \ldots, A) \in \mathbb{F}_q^{N\times n}$ is the block-diagonal matrix with A repeated in the main block-diagonal g times.

Each file $\mathbf{x}^i$ is then encoded into $\mathbf{y}^i = \mathbf{x}^i G_{glob} \in \mathbb{F}_{q^r}^{b\times n}$, where $\mathbf{y}^i_{\Gamma_j} \in \mathbb{F}_{q^r}^{b\times(r+\delta-1)}$ is stored in the jth server. Let $\Delta_j \subseteq \Gamma_j$ be the first r coordinates in Γ_j and assume that the first r columns of A form the identity matrix (i.e. A is systematic). Then if we disregard the nodes in $\Gamma_j \setminus \Delta_j$, the part of the ith file stored in the jth server is $\mathbf{z}^i_j \in \mathbb{F}_{q^r}^{b\times r}$, for $j = 1, 2, \ldots, g$, where

$$Z = XG_{out} = \begin{pmatrix} \mathbf{z}^1 \\ \vdots \\ \mathbf{z}^m \end{pmatrix} = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_g) = \begin{pmatrix} \mathbf{z}^1_1 & \ldots & \mathbf{z}^1_g \\ \vdots & \ddots & \vdots \\ \mathbf{z}^m_1 & \ldots & \mathbf{z}^m_g \end{pmatrix} \in \mathbb{F}_{q^r}^{bm\times N}. \tag{4}$$

Thus the remaining MDS code coincides with the outer code: $\mathcal{C}_\Delta = \mathcal{C}_{out}$.

The parameter b will be called the *folding parameter*. It is a common parameter that is usually considered in Coding Theory implicitly. It allows to store a larger amount of data while the encoding and decoding operations grow linearly with b.

We now formalize general private information retrieval schemes for MR-LRCs.

Definition 3. A private information retrieval (PIR) scheme for an MR-LRC distributed storage system as described above consists, for each $i = 1, 2, \ldots, m$, of:

1. Random *queries* sent to the jth server to retrieve the ith file:
$$\mathbf{q}_j^i = \left(\mathbf{q}_j^{i,1}, \mathbf{q}_j^{i,2}, \ldots, \mathbf{q}_j^{i,m}\right) \in \mathbb{F}_{q^r}^{bmr},$$
where $\mathbf{q}_j^{i,\ell} \in \mathbb{F}_{q^r}^{br}$, for $\ell = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, g$.
2. The corresponding *response* $\mathbf{r}_j^i = \mathbf{z}_j \cdot \mathbf{q}_j^i \in \mathbb{F}_{q^r}^r$ of the jth server when requested the ith file (the server only knows $\mathbf{z}_j$ and $\mathbf{q}_j^i$ in principle), for $j = 1, 2, \ldots, g$. The product $\cdot$ will be given in (14). We will denote $\mathbf{r}^i = (\mathbf{r}_1^i, \mathbf{r}_2^i, \ldots, \mathbf{r}_g^i) \in \mathbb{F}_{q^r}^N$.
3. A number s of iterations of Items 1 and 2, until the ith file can be recovered from the responses $\mathbf{r}^i$ in Item 2.
4. A reconstruction function with input the s responses $\mathbf{r}^i$ and output the ith file $\mathbf{x}^i$.

A major difference with [9,11] is that we do not use the usual inner product $\mathbf{z} \cdot \mathbf{q}$, but a generalization of it (see Definition 8 below).

As usual in the PIR literature, our goal is to maximize the download rate, which is defined as the file size divided by the amount of downloaded data. The upload cost may be considered negligible by further *folding* the scheme $b' \gg 1$ times, thus a total of bb' times. Disregarding also local redundancies, the download rate is as follows.

Definition 4. We define the *download rate*, or simply rate, of a PIR scheme given as in Definition 3 as
$$R = \frac{bk}{Ns}. \tag{5}$$

We require information-theoretical privacy for a given number t of colluding servers (i.e. colluding local groups).

Definition 5. We say that a PIR scheme as in Definition 3 protects against t colluding servers (i.e., colluding local groups) if, for every $T \subseteq [g]$ of size t, in each iteration of the scheme we have that
$$I\left(\left(\mathbf{q}_j^i\right)_{j \in T}; i\right) = 0,$$
where $I(X;Y)$ denotes the mutual information between two random variables X and Y (see [8, Ch. 12]).

3. Coordinate-wise and inner matrix products

In this section, we define and collect the main properties of inner and coordinate-wise matrix products used in our PIR scheme (see Item 2 in Definition 3). Such products will be crucial for the MR-LRCs from [24] based on linearized Reed-Solomon codes [22]. Thus we start by revisiting such codes.

3.1. MR-LRCs based on linearized Reed-Solomon codes

Fix $r \geq 1$ and let $\sigma : \mathbb{F}_{q^r} \longrightarrow \mathbb{F}_{q^r}$ be given by $\sigma(a) = a^q$, for all $a \in \mathbb{F}_{q^r}$. We next define linear operators as in [22, Def. 20].

Definition 6 *([22]).* Fix $a \in \mathbb{F}_{q^r}$, and define its ith norm as $N_i(a) = \sigma^{i-1}(a) \cdots \sigma(a)a$, for $i \in \mathbb{N}$. We define the $\mathbb{F}_q$-linear operator $\mathcal{D}_a^i : \mathbb{F}_{q^r} \longrightarrow \mathbb{F}_{q^r}$ by

$$\mathcal{D}_a^i(\beta) = \sigma^i(\beta) N_i(a), \tag{6}$$

for all $\beta \in \mathbb{F}_{q^r}$, and all $i \in \mathbb{N}$. Define also $\mathcal{D}_a = \mathcal{D}_a^1$ and observe that $\mathcal{D}_a^{i+1} = \mathcal{D}_a \circ \mathcal{D}_a^i$, for $i \in \mathbb{N}$. Denote by $\mathbb{F}_{q^r}[\mathcal{D}_a]$ the polynomial ring in the operator $\mathcal{D}_a$, for $a \in \mathbb{F}_{q^r}$.

Recall that the *skew polynomial* ring $\mathbb{F}_{q^r}[x;\sigma]$, introduced in [27], is the polynomial ring on the variable x but with non-commutative product given by the rule

$$x\beta = \sigma(\beta)x, \tag{7}$$

for all $\beta \in \mathbb{F}_{q^r}$. For $F = \sum_{i=0}^d F_i x^i \in \mathbb{F}_{q^r}[x;\sigma]$, we define

$$F^{\mathcal{D}_a} = \sum_{i=0}^d F_i \mathcal{D}_a^i \in \mathbb{F}_{q^r}[\mathcal{D}_a], \tag{8}$$

for $a \in \mathbb{F}_{q^r}$. In the following, for $F = \sum_{i=0}^d F_i x^i \in \mathbb{F}_{q^r}[x;\sigma]$, for $a \in \mathbb{F}_{q^r}$ and for $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_r) \in \mathbb{F}_{q^r}^r$, we use the notation

$$F^{\mathcal{D}_a}(\boldsymbol{\beta}) = \left(F^{\mathcal{D}_a}(\beta_1), F^{\mathcal{D}_a}(\beta_2), \ldots, F^{\mathcal{D}_a}(\beta_r)\right) \in \mathbb{F}_{q^r}^r. \tag{9}$$

Next, given $\mathbf{a} = (a_1, a_2, \ldots, a_g) \in \mathbb{F}_{q^r}^g$, we define the total evaluation vector of F at $(\mathbf{a}, \boldsymbol{\beta})$ as

$$F^{\mathcal{D}_{\mathbf{a}}}(\boldsymbol{\beta}) = \left(F^{\mathcal{D}_{a_1}}(\boldsymbol{\beta}), F^{\mathcal{D}_{a_2}}(\boldsymbol{\beta}), \ldots, F^{\mathcal{D}_{a_g}}(\boldsymbol{\beta})\right) \in \mathbb{F}_{q^r}^N, \tag{10}$$

where $N = gr$. The following definition is a particular case of [22, Def. 31].

Definition 7 *(Linearized Reed-Solomon codes [22]).* Fix a primitive element $\gamma \in \mathbb{F}_{q^r}^*$ and let $\mathbf{a} = (\gamma^0, \gamma^1, \dots, \gamma^{g-1}) \in \mathbb{F}_{q^r}^g$. Fix an ordered basis $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_r) \in \mathbb{F}_{q^r}^r$ of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$. For $k = 0, 1, 2, \dots, N$, where $N = gr$, we define the (N, k) linearized Reed-Solomon (LRS) code as

$$\mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta}) = \left\{ F^{\mathcal{D}_{\mathbf{a}}}(\boldsymbol{\beta}) \in \mathbb{F}_{q^r}^N \mid F \in \mathbb{F}_{q^r}[x;\sigma], \deg(F) < k \text{ or } F = 0 \right\} \subseteq \mathbb{F}_{q^r}^N.$$

Here, the degree $\deg(F)$ of a non-zero skew polynomial $F = \sum_{i \in \mathbb{N}} F_i x^i \in \mathbb{F}_{q^r}[x;\sigma]$, where $F_i \in \mathbb{F}_{q^r}$ for $i \in \mathbb{N}$, is defined as the maximum $i \in \mathbb{N}$ such that $F_i \neq 0$.

Linearized Reed-Solomon codes recover Reed-Solomon codes [28] by setting $r = 1$ and $\beta_1 = 1$, and they recover Gabidulin codes [12] by setting $g = 1$.

In [24, Const. 1], a construction of MR-LRCs was given based on linearized Reed-Solomon codes (Definition 7). This construction recovers Reed-Solomon codes if $r = \delta = 1$, and it recovers Cartesian products of codes if $h = gr - k = 0$.

Construction 1 *(LRS-based MR-LRC [24]).* Fix the positive integers g, r and δ, and choose any *base field* size $q > \max\{r + \delta - 3, g\}$. Next choose a dimension $k = 1, 2, \dots, N$, where $N = gr$, and:

1. *Outer code*: An (N, k) linearized Reed-Solomon code $\mathcal{C}_{out} = \mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta}) \subseteq \mathbb{F}_{q^r}^N$ as in Definition 7.
2. *Local codes*: Any linear $(r + \delta - 1, r)$ MDS code $\mathcal{C}_{loc} \subseteq \mathbb{F}_q^{r+\delta-1}$.
3. *Global code*: Let $\mathcal{C}_{glob} \subseteq \mathbb{F}_{q^r}^n$, with $n = (r + \delta - 1)g = N + (\delta - 1)g$, be given by

$$\mathcal{C}_{glob} = \mathcal{C}_{out} \operatorname{Diag}_g(A) \subseteq \mathbb{F}_{q^r}^n,$$

where $A \in \mathbb{F}_q^{r \times (r+\delta-1)}$ is any generator matrix of $\mathcal{C}_{loc}$.

The following result is [24, Th. 2] and states the MR and LRC properties of the global code $\mathcal{C}_{glob}$ in Construction 1.

Theorem 1 *([24]). Let $\mathcal{C}_{glob} \subseteq \mathbb{F}_{q^r}^n$ be the global code from Construction 1, and let $\Gamma_j \subseteq [n]$ be the subset of coordinates from $(r+\delta-1)(j-1)+1$ to $(r+\delta-1)j$, for $j = 1, 2, \dots, g$. Then $\mathcal{C}_{glob} \subseteq \mathbb{F}_{q^r}^n$ has (r, δ) localities, local groups $\Gamma_1, \Gamma_2, \dots, \Gamma_g$, and is maximally recoverable. Furthermore, its field size may be chosen as $\max\{r + \delta - 3, g\}^r$.*

3.2. Definition and linearity properties of the products

Fix an ordered basis $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_r) \in \mathbb{F}_{q^r}^r$ of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$. Denote by $M_{\boldsymbol{\beta}} : \mathbb{F}_{q^r}^r \longrightarrow \mathbb{F}_q^{r \times r}$ the corresponding *matrix-representation map*, given by

$$M_{\boldsymbol{\beta}}(\mathbf{x}) = \begin{pmatrix} x_1^1 & x_2^1 & \dots & x_r^1 \\ x_1^2 & x_2^2 & \dots & x_r^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^r & x_2^r & \dots & x_r^r \end{pmatrix}, \tag{11}$$

for $\mathbf{x} = (x_1, x_2, \dots, x_r) \in \mathbb{F}_{q^r}^r$, where $x_j^1, x_j^2, \dots, x_j^r \in \mathbb{F}_q$ are the unique scalars such that $x_j = \sum_{i=1}^r \beta_i x_j^i \in \mathbb{F}_{q^r}$, for $j = 1, 2, \dots, r$. Observe that $M_{\boldsymbol{\beta}}$ is an $\mathbb{F}_q$-linear vector space isomorphism, and it is the identity map if $r = 1$ and $\beta_1 = 1$.

Definition 8. Given $\mathbf{x}, \mathbf{y} \in \mathbb{F}_{q^r}^r$, we define their *matrix product* with respect to $\boldsymbol{\beta}$ as

$$\mathbf{x} \star \mathbf{y} = M_{\boldsymbol{\beta}}^{-1}(M_{\boldsymbol{\beta}}(\mathbf{x}) M_{\boldsymbol{\beta}}(\mathbf{y})) \in \mathbb{F}_{q^r}^r. \tag{12}$$

For $N = gr$, $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_g) \in \mathbb{F}_{q^r}^N$ and $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_g) \in \mathbb{F}_{q^r}^N$, where $\mathbf{x}_j, \mathbf{y}_j \in \mathbb{F}_{q^r}^r$, for $j = 1, 2, \dots, g$, we define their *coordinate-wise matrix product* as

$$\mathbf{x} * \mathbf{y} = (\mathbf{x}_1 \star \mathbf{y}_1, \mathbf{x}_2 \star \mathbf{y}_2, \dots, \mathbf{x}_g \star \mathbf{y}_g) \in \mathbb{F}_{q^r}^N, \tag{13}$$

and we define their *inner matrix product* $\cdot$ as

$$\mathbf{x} \cdot \mathbf{y} = \sum_{j=1}^{g} \mathbf{x}_j \star \mathbf{y}_j \in \mathbb{F}_{q^r}^r. \tag{14}$$

The products $\star$, $*$ and $\cdot$ all depend on the subfield $\mathbb{F}_q \subseteq \mathbb{F}_{q^r}$ (thus q and r) and the ordered basis $\boldsymbol{\beta}$, but we will not denote this dependence for simplicity. The classical coordinate-wise and inner products in $\mathbb{F}_q^N$, used in [11] for PIR (and in general in the literature, see [9]), are recovered by setting $r = 1$ and $\beta_1 = 1$ (thus $N = g$).

From the definitions, we note also that, if $\mathbf{x} = (x_1, x_2, \dots, x_r) \in \mathbb{F}_{q^r}^r$ and $\mathbf{y} = \sum_{i=1}^r \beta_i \mathbf{y}^i \in \mathbb{F}_{q^r}^r$, with $x_i \in \mathbb{F}_{q^r}$ and $\mathbf{y}^i \in \mathbb{F}_q^r$, for $i = 1, 2, \dots, r$, then

$$M_{\boldsymbol{\beta}}^{-1}(M_{\boldsymbol{\beta}}(\mathbf{x}) M_{\boldsymbol{\beta}}(\mathbf{y})) = \sum_{i=1}^{r} x_i \mathbf{y}^i \in \mathbb{F}_{q^r}^r. \tag{15}$$

From Equation (15) applied coordinate-wise, we deduce the following.

Lemma 9. *The coordinate-wise matrix product $*$ is $\mathbb{F}_q$-bilinear and $\mathbb{F}_{q^r}$-linear in the first component, that is,*

1. $(\mathbf{x} + \mathbf{x}') * \mathbf{y} = \mathbf{x} * \mathbf{y} + \mathbf{x}' * \mathbf{y}$ *and* $\mathbf{x} * (\mathbf{y} + \mathbf{y}') = \mathbf{x} * \mathbf{y} + \mathbf{x} * \mathbf{y}'$,
2. $(a\mathbf{x}) * \mathbf{y} = a(\mathbf{x} * \mathbf{y})$ *and* $\mathbf{x} * (b\mathbf{y}) = b(\mathbf{x} * \mathbf{y})$,

for all $\mathbf{x}, \mathbf{x}', \mathbf{y}, \mathbf{y}' \in \mathbb{F}_{q^r}^N$, all $a \in \mathbb{F}_{q^r}$ and all $b \in \mathbb{F}_q$.

3.3. Products of skew and linearized polynomials

We have the following important connection between the rings $\mathbb{F}_{q^r}[x;\sigma]$ and $\mathbb{F}_{q^r}[\mathcal{D}_a]$, for all $a \in \mathbb{F}_{q^r}$. We consider $\mathbb{F}_{q^r}[\mathcal{D}_a]$ as a ring with conventional addition and with composition of maps as multiplication, denoted by $\circ$.

Lemma 10. *For all $F, G \in \mathbb{F}_{q^r}[x;\sigma]$ and all $a \in \mathbb{F}_{q^r}$, it holds that*

$$(FG)^{\mathcal{D}_a} = F^{\mathcal{D}_a} \circ G^{\mathcal{D}_a}.$$

In particular, the map $\mathbb{F}_{q^r}[x;\sigma] \longrightarrow \mathbb{F}_{q^r}[\mathcal{D}_a]$ given by (8) is a (surjective) ring morphism.

Proof. Observe that

$$\mathcal{D}_a \circ (\beta \mathrm{Id}) = \sigma(\beta)\mathcal{D}_a, \tag{16}$$

for all $\beta \in \mathbb{F}_{q^r}$, where $\mathrm{Id} = \mathcal{D}_a^0$ is the multiplicative identity of $\mathbb{F}_{q^r}[\mathcal{D}_a]$, and note that (16) coincides with (7) if we set $x = \mathcal{D}_a$. Since (7) is the defining property of the product in the skew polynomial ring $\mathbb{F}_{q^r}[x;\sigma]$, the result follows. □

The main result of this section is showing that products of skew polynomials become coordinate-wise matrix products after evaluation via the operators $\mathcal{D}_a$.

Theorem 2. *Let $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_r) \in \mathbb{F}_{q^r}^r$ be an ordered basis of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$, and let coordinate-wise matrix products $*$ be defined via $\boldsymbol{\beta}$. Then it holds that*

$$(FG)^{\mathcal{D}_{\mathbf{a}}}(\boldsymbol{\beta}) = F^{\mathcal{D}_{\mathbf{a}}}(\boldsymbol{\beta}) * G^{\mathcal{D}_{\mathbf{a}}}(\boldsymbol{\beta}),$$

for all vectors $\mathbf{a} = (a_1, a_2, \ldots, a_g) \in \mathbb{F}_{q^r}^g$ and all skew polynomials $F, G \in \mathbb{F}_{q^r}[x;\sigma]$.

Proof. In [25, Prop. 1], it was proven that

$$\sigma^\ell(\boldsymbol{\beta}) \star \mathbf{y} = \sigma^\ell(\mathbf{y}), \tag{17}$$

for all $\ell \in \mathbb{N}$ and all $\mathbf{y} \in \mathbb{F}_{q^r}^r$. We recall the proof of (17) for convenience of the reader. If $\mathbf{y} = \sum_{i=1}^r \beta_i \mathbf{y}^i$, where $\mathbf{y}^i \in \mathbb{F}_q^r$, for $i = 1, 2, \ldots, r$, then

$$\sigma^\ell(\boldsymbol{\beta}) \star \mathbf{y} = \sum_{i=1}^r \sigma^\ell(\beta_i)\mathbf{y}^i = \sigma^\ell\left(\sum_{i=1}^r \beta_i \mathbf{y}^i\right) = \sigma^\ell(\mathbf{y}),$$

where the first equality is (15).

Since $\star$ is $\mathbb{F}_{q^r}$-linear in the first component (Lemma 9), and $\mathcal{D}_a^\ell = N_\ell(a)\sigma^\ell$, where $N_\ell(a) \in \mathbb{F}_{q^r}$, then we deduce from (17) that

$$\mathcal{D}_a^\ell(\boldsymbol{\beta}) \star \mathbf{y} = (N_\ell(a)\sigma^\ell(\boldsymbol{\beta})) \star \mathbf{y} = N_\ell(a)(\sigma^\ell(\boldsymbol{\beta}) \star \mathbf{y}) = N_\ell(a)\sigma^\ell(\mathbf{y}) = \mathcal{D}_a^\ell(\mathbf{y}),$$

for all $a \in \mathbb{F}_{q^r}$, all $\mathbf{y} \in \mathbb{F}_{q^r}^r$ and all $\ell \in \mathbb{N}$. Thus the case $g = 1$ follows by combining Lemmas 9 and 10.

Finally, the theorem for general g follows by applying the case $g = 1$ separately in each of the g coordinates over the alphabet $\mathbb{F}_{q^r}^r$, and applying Lemma 10. □

Setting $r = 1$ and $\beta_1 = 1$, the previous theorem is nothing but the well-known fact that coordinate-wise evaluation transforms conventional polynomial products into the conventional coordinate-wise product.

We conclude by deducing that the product of two linearized Reed-Solomon codes over the same ordered basis $\boldsymbol{\beta}$ is again a linearized Reed-Solomon code. For this purpose, given $\mathbb{F}_{q^r}$-linear codes $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathbb{F}_{q^r}^N$, we define their coordinate-wise matrix product as

$$\mathcal{C}_1 * \mathcal{C}_2 = \langle \{ \mathbf{c}_1 * \mathbf{c}_2 \mid \mathbf{c}_1 \in \mathcal{C}_1, \mathbf{c}_2 \in \mathcal{C}_2 \} \rangle \subseteq \mathbb{F}_{q^r}^N,$$

where $\langle \mathcal{A} \rangle$ denotes the $\mathbb{F}_q$-linear vector space generated by $\mathcal{A} \subseteq \mathbb{F}_{q^r}^N$.

Corollary 11. *Let $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_r) \in \mathbb{F}_{q^r}^r$ be an ordered basis of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$, and let coordinate-wise matrix products $*$ be defined via $\boldsymbol{\beta}$. Let also $\mathbf{a} = (\gamma^0, \gamma^1, \ldots, \gamma^{g-1}) \in \mathbb{F}_{q^r}^g$, for a primitive element $\gamma \in \mathbb{F}_{q^r}^*$. For any $k_1, k_2 = 0, 1, 2, \ldots, N$, with $k_1 \geq 1$,*

$$\mathcal{C}_{N,k_1}(\mathbf{a}, \boldsymbol{\beta}) * \mathcal{C}_{N,k_2}(\mathbf{a}, \boldsymbol{\beta}) = \mathcal{C}_{N,k_1+k_2-1}(\mathbf{a}, \boldsymbol{\beta})$$

if $k_1 + k_2 - 1 \leq N$, and $\mathcal{C}_{N,k_1}(\mathbf{a}, \boldsymbol{\beta}) * \mathcal{C}_{N,k_2}(\mathbf{a}, \boldsymbol{\beta}) = \mathbb{F}_{q^r}^N$ otherwise.

Proof. It follows by combining Lemma 9, Theorem 2 and the fact that

$$\deg(FG) = \deg(F) + \deg(G),$$

for all skew polynomials $F, G \in \mathbb{F}_{q^r}[x; \sigma]$. □

Setting $r = 1$ and $\beta_1 = 1$, Corollary 11 recovers the well-known fact that the classical coordinate-wise product of two Reed-Solomon codes is again a Reed-Solomon code. See for instance [11, Prop. 3]. Setting $g = 1$, Corollary 11 recovers the fact that the matrix product of two Gabidulin codes is again a Gabidulin code. See for instance [25, Lemma 10] or [38].

4. PIR schemes for LRS-based MR-LRC databases

In this section, we provide a concrete and explicit PIR scheme, as in Definition 3, for the MR-LRC storage codes from Construction 1. To that end, we will show how

to construct the queries and how to reconstruct the file from the responses. The set of server responses, given the queries, are as described in Definition 3.

Let the notation be as in Section 2, fix an ordered basis $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_r) \in \mathbb{F}_{q^r}^r$ of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$, and let coordinate-wise matrix products $*$ be defined via $\boldsymbol{\beta}$. We set throughout this section $t \geq 1$ as the target number of colluding servers, with the restriction

$$k + rt \leq N, \tag{18}$$

and we will set $c = N - k - rt + 1 > 0$.

For clarity, we present two schemes, being the first one (Subsection 4.1) a particular case of the second one (Subsection 4.2) by setting $b = 1$. The first scheme is added because it is a particular case that is simpler, it is easier to understand and does not require folding by setting $b > 1$. However, it requires that k is divisible by c. The second scheme is added simply because it is an extension of the first scheme that works for any set of parameters.

Both schemes will achieve the PIR rate

$$R = \frac{c}{N} = \frac{N - k - rt + 1}{N}, \tag{19}$$

coinciding with the rate of the PIR scheme in [11] for (N, k) GRS storage codes. However, as explained in Section 1, one cannot compare these two PIR schemes, since [11] only works for GRS codes, which cannot be the MDS codes obtained from puncturing MR-LRCs (since GRS have linear field sizes and MR-LRCs require super-linear field sizes [15]). Moreover, the PIR scheme from [11] only works for GRS codes since it uses coordinate-wise products, and GRS codes are the only MDS codes that satisfy the desired properties with respect to such products [21]. We circumvent this limitation by considering coordinate-wise matrix products and LRS codes.

We also remark here that, mathematically speaking, the scheme in [11] is precisely the particular case of our second scheme by setting $r = \delta = 1$. Being able to extend it to arbitrary r and δ requires a somewhat different partition of the considered vectors and matrices, and the careful use of the coordinate-wise matrix products from Section 3. On the Coding-Theoretic side, the obvious and important difference with [11] is that setting $r = \delta = 1$ simply does not allow for local repair.

4.1. First scheme: no folding

Our first scheme assumes no minimum folding of the files, that is, $b = 1$, and is precisely the particular case of our second scheme obtained by setting $b = 1$. Note that the stored codewords may however be further folded $b' \gg 1$ times without folding the PIR scheme. The main disadvantage of choosing $b = 1$ is that the dimension k must be divisible by $c = N - k - rt + 1$. Relaxing this divisibility assumption is the advantage of the second scheme.

As stated above, assume that $k = sc$, for some $s \in \mathbb{N}$, which will be the number of iterations of the scheme. This can be trivially assumed if $k + rt = N$, thus $c = 1$ and $s = k$. Note that this is the best choice for the competing parameters k and t satisfying (18), but it gives the smallest PIR rate $R = 1/N \leq c/N$ among our schemes, although $R = 1/N$ is still far better than downloading the whole database (which gives rate $1/m$), since $m \gg N$ in practice.

Fix file and iteration indices $i = 1, 2, \ldots, m$ and $u = 1, 2, \ldots, s$, respectively. We now describe the two steps of the uth iteration in Definition 3 to privately retrieve the ith file.

Step 1, Queries: Choose m codewords $\mathbf{d}^\ell = (\mathbf{d}_1^\ell, \mathbf{d}_2^\ell, \ldots, \mathbf{d}_g^\ell) \in \mathbb{F}_{q^r}^N$ uniformly at random from $\mathcal{C}_{N,rt}(\mathbf{a}, \boldsymbol{\beta})$, where $\mathbf{d}_j^\ell \in \mathbb{F}_{q^r}^r$, for $\ell = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, g$. The random vectors $\mathbf{d}^\ell = \mathbf{d}^\ell(u)$ depend on the iteration index u (i.e. $\mathbf{d}^\ell(1), \mathbf{d}^\ell(2), \ldots, \mathbf{d}^\ell(s)$ are identically distributed and independent), but we sometimes omit the index u for simplicity. Set

$$\mathbf{d}_j = (\mathbf{d}_j^1, \mathbf{d}_j^2, \ldots, \mathbf{d}_j^m) \in \mathbb{F}_{q^r}^{rm},$$

for $j = 1, 2, \ldots, g$. Define the set

$$J_u = c(u-1) + [c],$$

where we use the notation $a + B = \{a + b \mid b \in B\}$, for $a \in \mathbb{Z}$ and $B \subseteq \mathbb{Z}$. Finally, for each server $j = 1, 2, \ldots, g$, we define its query by

$$\mathbf{q}_j^i(u) = \mathbf{d}_j(u) + \mathbf{e}_j^i(u) \in \mathbb{F}_{q^r}^{rm}, \tag{20}$$

where we define $\mathbf{e}_j^i(u) \in \mathbb{F}_{q^r}^{rm}$ as being zero everywhere except in the ith block of r coordinates over $\mathbb{F}_{q^r}$, where it is defined as

$$M_{\boldsymbol{\beta}}^{-1}(I_{((j-1)r+[r])\cap J_u}) \in \mathbb{F}_{q^r}^r. \tag{21}$$

Here, we define $I_J \in \mathbb{F}_q^{r\times r}$, for a set $J \subseteq (j-1)r + [r]$, as the diagonal matrix $I_J = \mathrm{Diag}(\delta_1^J, \delta_2^J, \ldots, \delta_r^J)$, where $\delta_\kappa^J = 1$ if $(j-1)r + \kappa \in J$, and $\delta_\kappa^J = 0$ otherwise. Note that $I_\varnothing \in \mathbb{F}_q^{r\times r}$ is the zero matrix.

Step 2, Responses: Due to the definition of the queries in (20) and the inner matrix product (14), the total response in the uth iteration is

$$\mathbf{r}^i(u) = \sum_{\ell=1}^{m} (\mathbf{z}^\ell * \mathbf{d}^\ell(u)) + (\mathbf{0}_{(u-1)c}, \mathbf{z}_{J_u}^i, \mathbf{0}_{N-uc}) \in \mathbb{F}_{q^r}^N,$$

where $\mathbf{0}_M \in \mathbb{F}_{q^r}^M$ is a zero vector of length M. This is because, by the definition of the inner matrix product $\cdot$ in $\mathbb{F}_{q^r}^{rm}$, the response by the jth server is given by

$$\begin{aligned}\mathbf{r}_j^i(u) = \mathbf{z}_j \cdot \mathbf{q}_j^i(u) = \sum_{\ell=1}^{m} \mathbf{z}_j^\ell \star \mathbf{q}_j^{i,\ell}(u) &= \sum_{\ell=1}^{m} \left(\mathbf{z}_j^\ell \star \mathbf{d}_j^\ell(u) + \mathbf{z}_j^\ell \star \mathbf{e}_j^{i,\ell}(u) \right) \\ &= \sum_{\ell=1}^{m} (\mathbf{z}_j^\ell \star \mathbf{d}_j^\ell(u)) + \sum_{\ell=1}^{m} \mathbf{z}_j^\ell \star \mathbf{e}_j^{i,\ell}(u) \\ &= \sum_{\ell=1}^{m} (\mathbf{z}_j^\ell \star \mathbf{d}_j^\ell(u)) + \mathbf{z}_j^i I_{((j-1)r+[r])\cap J_u},\end{aligned}$$

where $\mathbf{e}_j^{i,\ell}(u) \in \mathbb{F}_{q^r}^r$ is the ℓth block of r coordinates of the vector $\mathbf{e}_j^i(u) \in \mathbb{F}_{q^r}^{rm}$, and finally,

$$\begin{aligned}\left(\mathbf{z}_j^i I_{((j-1)r+[r])\cap J_u}\right)_{j=1}^{g} &= \left(\mathbf{z}_1^i I_{[r]\cap J_u}, \ldots, \mathbf{z}_g^i I_{((g-1)r+[r])\cap J_u}\right) \\ &= (\mathbf{0}_{(u-1)c}, \mathbf{z}_{J_u}^i, \mathbf{0}_{N-uc}) \in \mathbb{F}_{q^r}^N.\end{aligned}$$

In our view, this is the key step where we use that the jth server (i.e., jth corruptable unit) is the jth local group after removing the local redundancies. This allows the corresponding stored data $\mathbf{z}_j^\ell$ and query $\mathbf{q}_j^\ell$ to be seen as $r \times r$ matrices over $\mathbb{F}_q$, and thus we may perform the above operations.

Step 3, File reconstruction: We now describe how to recover the ith file by combining the responses from all s iterations. Let $H \in \mathbb{F}_{q^r}^{c\times N}$ be a parity-check matrix (any of them) of the linear code

$$\mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta}) = \mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta}) * \mathcal{C}_{N,rt}(\mathbf{a}, \boldsymbol{\beta})$$

(recall Corollary 11). For $u = 1, 2, \ldots, s$, we compute

$$\mathbf{r}^i(u)H^T = (\mathbf{0}_{(u-1)c}, \mathbf{z}_{J_u}^i, \mathbf{0}_{N-uc})H^T,$$

which holds since

$$\sum_{\ell=1}^{m} (\mathbf{z}^\ell * \mathbf{d}^\ell(u)) \in \mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta}).$$

Since $\mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta})$ is MDS by Theorem 1, its dual is also MDS, and we can recover the vector $\mathbf{z}_{J_u}^i \in \mathbb{F}_{q^r}^c$ from $\mathbf{r}^i(u)H^T$. Since we have that

$$[k] = J_1 \cup J_2 \cup \ldots \cup J_s,$$

collecting all such s restrictions $\mathbf{z}_{J_u}^i$, we obtain

$$(z_1^i, z_2^i, \ldots, z_k^i) = \mathbf{x}^i (G_{out})_{[k]} \in \mathbb{F}_{q^r}^k.$$

Now, since $\mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta})$ is MDS, again by Theorem 1, we may recover the ith file, $\mathbf{x}^i \in \mathbb{F}_{q^r}^k$, and we are done.

Note that the MDS property in this last step is not necessary: If we take the generator matrix G_{out} of the outer code $\mathcal{C}_{N,k}(\mathbf{a},\boldsymbol{\beta})$ to be systematic, with the identity in the first k columns, then it simply holds that $\mathbf{x}^i = (z_1^i, z_2^i, \dots, z_k^i)$.

Proof of privacy: We now show that the proposed PIR scheme protects against any t colluding servers as in Definition 5. Recall from Section 2 that we identify servers with local groups. Let $T \subseteq [g]$, such that $|T| = t$, be the set of colluding local groups. Therefore, this can be understood as an adversary gaining as information the values $\mathbf{q}_j^i(u) \in \mathbb{F}_{q^r}^{rm}$, for $j \in T$, and for all iterations $u = 1, 2, \dots, s$. We will just write $\mathbf{q}_j^i = \mathbf{q}_j^i(u)$ for simplicity. We need to prove that, for a given iteration, it holds that

$$I((\mathbf{q}_j^i)_{j\in T}; i) = 0.$$

Since $\mathcal{C}_{N,rt}(\mathbf{a},\boldsymbol{\beta}) \subseteq \mathbb{F}_{q^r}^N$ has dimension rt and is MDS by Theorem 1, it holds that any set of rt coordinates in $[N]$ constitute an information set for $\mathcal{C}_{N,rt}(\mathbf{a},\boldsymbol{\beta})$. In other words, the restricted code $\mathcal{C}_{N,rt}(\mathbf{a},\boldsymbol{\beta})_{\widetilde{T}} = \mathbb{F}_{q^r}^{rt}$ is the whole space, where $\widetilde{T} = \bigcup_{j\in T}((j-1)r + [r]) \subseteq [N]$ is the actual set of colluding nodes. This implies that the vectors

$$(\mathbf{d}_j^\ell)_{j\in T} \in \mathbb{F}_{q^r}^{rt}$$

are uniform random variables in $\mathbb{F}_{q^r}^{rt}$. Since the Cartesian product of independent and uniform random variables is again a uniform random variable, we deduce that

$$(\mathbf{d}_j)_{j\in T} \in \mathbb{F}_{q^r}^{rtm}$$

is a uniform random variable in $\mathbb{F}_{q^r}^{rtm}$. Since the vector of queries $(\mathbf{q}_j^i)_{j\in T}$ is a translation of the random variable $(\mathbf{d}_j)_{j\in T}$ by a deterministic vector, we deduce that $(\mathbf{q}_j^i)_{j\in T}$ is a uniform random variable in $\mathbb{F}_{q^r}^{rtm}$. Since there is only one uniform random variable in $\mathbb{F}_{q^r}^{rtm}$, independently of i, we deduce that $I((\mathbf{q}_j^i)_{j\in T}; i) = 0$, and we are done.

4.2. Second scheme: folding

In our second scheme, we avoid the constraint that k must be divisible by $c = N - k - rt + 1$. To that end, we will make use of the folding parameter b as done in [11]. Again, the stored codewords may be further folded $b' \gg 1$ times without further folding the PIR scheme. We emphasize here that the scheme in [11] is actually recovered from this second scheme by setting $r = \delta = 1$, which is the case in which linearized Reed-Solomon codes recover Reed-Solomon codes (see Subsection 3.1). Our first scheme is also recovered from this second scheme by setting $b = 1$. To avoid the divisibility assumption, we define

$$b = \frac{\operatorname{lcm}(c,k)}{k} \quad \text{and} \quad s = \frac{\operatorname{lcm}(c,k)}{c},$$

hence guaranteeing that $bk = sc$. Thus we may define

$$h = \frac{k}{s} = \frac{c}{b}.$$

Fix file and iteration indices $i = 1, 2, \ldots, m$ and $u = 1, 2, \ldots, s$, respectively. We now describe the two steps of uth iteration in Definition 3 to privately retrieve the ith file.

Step 1, Queries: Choose mb codewords $\mathbf{d}^{\ell,v} = (\mathbf{d}_1^{\ell,v}, \mathbf{d}_2^{\ell,v}, \ldots, \mathbf{d}_g^{\ell,v}) \in \mathbb{F}_{q^r}^N$, uniformly at random from $\mathcal{C}_{N,rt}(\mathbf{a}, \boldsymbol{\beta})$, where $\mathbf{d}_j^{\ell,v} \in \mathbb{F}_{q^r}^r$, for $\ell = 1, 2, \ldots, m$, $v = 1, 2, \ldots, b$, and $j = 1, 2, \ldots, g$. As before, $\mathbf{d}^{\ell,v} = \mathbf{d}^{\ell,v}(u)$ depends on u, but we sometimes drop this in the notation. We set

$$\mathbf{d}_j^\ell = (\mathbf{d}_j^{\ell,1}, \mathbf{d}_j^{\ell,2}, \ldots, \mathbf{d}_j^{\ell,b}) \in \mathbb{F}_{q^r}^{rb} \text{ and}$$
$$\mathbf{d}_j = (\mathbf{d}_j^1, \mathbf{d}_j^2, \ldots, \mathbf{d}_j^m) \in \mathbb{F}_{q^r}^{rbm},$$

for $\ell = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, g$. Define the sets

$$J_u^1 = h(u-1) + [h], J_u^2 = h + J_u^1, \ldots, J_u^b = h(b-1) + J_u^1.$$

Finally, for each server $j = 1, 2, \ldots, g$, we define its query by

$$\mathbf{q}_j^i(u) = \mathbf{d}_j(u) + \mathbf{e}_j^i(u) \in \mathbb{F}_{q^r}^{rbm}. \tag{22}$$

In this case, we define $\mathbf{e}_j^i(u) \in \mathbb{F}_{q^r}^{rbm}$ as being zero everywhere except in the $(b(i-1)+v)$th block of r coordinates over $\mathbb{F}_{q^r}$, for $v = 1, 2, \ldots, b$, where it is defined as

$$M_{\boldsymbol{\beta}}^{-1}(I_{((j-1)r+[r])\cap J_u^v}) \in \mathbb{F}_{q^r}^r. \tag{23}$$

As before, we define $I_J \in \mathbb{F}_q^{r\times r}$, for a set $J \subseteq (j-1)r + [r]$, as the diagonal matrix $I_J = \mathrm{Diag}(\delta_1^J, \delta_2^J, \ldots, \delta_r^J)$, where $\delta_\kappa^J = 1$ if $(j-1)r + \kappa \in J$, and $\delta_\kappa^J = 0$ otherwise. As before, $I_\varnothing \in \mathbb{F}_q^{r\times r}$ is the zero matrix.

Step 2, Responses: The reader can check that, from the definition of the queries in (22) and the inner matrix product (14), the total response in the first iteration is

$$\mathbf{r}^i = \sum_{\ell=1}^{m}\sum_{v=1}^{b}(\mathbf{z}^{\ell,v} * \mathbf{d}^{\ell,v}) + (\mathbf{z}_{J_1^1}^{i,1}, \mathbf{z}_{J_1^2}^{i,2}, \ldots, \mathbf{z}_{J_1^b}^{i,b}, \mathbf{0}) \in \mathbb{F}_{q^r}^N, \tag{24}$$

where $\mathbf{0}$ has length $N-c$. In the uth iteration, the response is obtained similarly, replacing $\mathbf{z}_{J_1^v}^{i,v}$ by $\mathbf{z}_{J_u^v}^{i,v}$, but placed in the coordinates indexed by J_u^v taking the cyclicity of the coordinates in $[N]$ into account, for $v = 1, 2, \ldots, b$.

Step 3, File reconstruction: We now describe how to recover the ith file by combining the responses from all s iterations. As before, let $H \in \mathbb{F}_{q^r}^{c\times N}$ be a parity-check matrix of

$$\mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta}) = \mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta}) * \mathcal{C}_{N,rt}(\mathbf{a}, \boldsymbol{\beta})$$

(recall Corollary 11). In the first iteration, we compute

$$\mathbf{r}^i H^T = (\mathbf{z}^{i,1}_{J_1^1}, \mathbf{z}^{i,2}_{J_1^2}, \ldots, \mathbf{z}^{i,b}_{J_1^b}, \mathbf{0}) H^T,$$

which holds since

$$\sum_{\ell=1}^{m} \sum_{v=1}^{b} (\mathbf{z}^{\ell,v} * \mathbf{d}^{\ell,v}) \in \mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta}).$$

As before, $\mathcal{C}_{N,k+rt-1}(\mathbf{a}, \boldsymbol{\beta})$ is MDS by Theorem 1, and thus its dual is also MDS. Therefore we may recover $\mathbf{z}^{i,1}_{J_1^1}, \mathbf{z}^{i,2}_{J_1^2}, \ldots, \mathbf{z}^{i,b}_{J_1^b} \in \mathbb{F}_{q^r}^h$ from $\mathbf{r}^i H^T$.

In a similar way, in the uth iteration, we recover $\mathbf{z}^{i,1}_{J_u^1}, \mathbf{z}^{i,2}_{J_u^2}, \ldots, \mathbf{z}^{i,b}_{J_u^b} \in \mathbb{F}_{q^r}^h$, for $u = 1, 2, \ldots, s$. For a given $v = 1, 2, \ldots, b$, the reader can check from their definition that the sets $J_1^v, J_2^v, \ldots, J_s^v$ are disjoint and the size of their union is $sh = k$. Therefore, we recover k symbols of $\mathbf{z}^{i,v} \in \mathbb{F}_{q^r}^N$ over the alphabet $\mathbb{F}_{q^r}$, together with their indices, given by $J_1^v \cup J_2^v \cup \ldots \cup J_s^v \subseteq [N]$. Since $\mathcal{C}_{N,k}(\mathbf{a}, \boldsymbol{\beta})$ is MDS, we recover the vth row of the ith file, that is, $\mathbf{x}^{i,v} \in \mathbb{F}_{q^r}^k$, for $v = 1, 2, \ldots, b$. Thus we are done by collecting all b rows, $\mathbf{x}^{i,1}, \mathbf{x}^{i,2}, \ldots, \mathbf{x}^{i,b}$, of the ith file.

Proof of privacy: Analogous to that in Subsection 4.1.

4.3. Summary of parameters and complexity

We now discuss the complexity of the three steps of the PIR scheme in this section. We only discuss the scheme without folding, since all complexities simply get multiplied by b in the folded case. We consider the three main steps of the scheme:

1. Queries: We need to generate sm uniformly random vectors in $\mathbb{F}_{q^r}^{rt}$ and multiply each of them by a generator matrix of $\mathcal{C}_{N,rt}(\mathbf{a}, \boldsymbol{\beta})$, that is, a matrix of size $rt \times N$. Thus this step has a complexity of $\mathcal{O}(smrtN) = \mathcal{O}(smN^2)$ operations in $\mathbb{F}_{q^r}$.
2. Responses: We need to perform sm products of two matrices in $\mathbb{F}_q^{r \times r}$ in order to compute the vectors $\mathbf{r}_j^i = \mathbf{z}_j \cdot \mathbf{q}_j^i$, hence this step has a complexity of $\mathcal{O}(smr^3)$ operations in $\mathbb{F}_q$.
3. Reconstruction: We need to compute s products of a vector in $\mathbb{F}_{q^r}^N$ with a matrix in $\mathbb{F}_{q^r}^{N \times c}$ in order to compute $\mathbf{r}^i(u) H^T$, hence this step has a complexity of $\mathcal{O}(scN)$ operations over $\mathbb{F}_{q^r}$. Finally, computing $\mathbf{x}^i (G_{out})_{[k]}$ is trivial if G_{out} is systematic (i.e. $(G_{out})_{[k]}$ is the identity).

As we can see, if the number of files m is much larger than the other parameters, then we may consider the total computational complexity as linear in m (recall that $m \gg N$ and $s, r, t \leq N$).

We next summarize the general PIR scheme (Subsection 4.2) in the following theorem by describing its parameters and computational complexity.

Theorem 3. *There exists a k-dimensional MR-LRC $\mathcal{C} \subseteq \mathbb{F}_{q^r}^n$ with (r, δ) localities as in Definitions 1 and 2 and, for a database coded with $\mathcal{C}$, there exists a PIR scheme as in Definition 3, with PIR rate*

$$R = \frac{N - k - rt + 1}{N},$$

where $N = gr$ and the other parameters are as follows:

Parameter	*Restrictions*	*Parameter*	*Restrictions*
r, δ, g	*None*	*Field size* q	$q > \max\{r + \delta - 3, g\}$
No. files m	*None*	*No. servers* n	$n = g(r + \delta - 1)$
Dimension k	$1 \leq k \leq N$	*Colluding servers* t	$k + rt \leq N$
Iterations s	$s = \frac{\operatorname{lcm}(k, N-k+rt+1)}{N-k+rt+1}$	*Folding* b	$b = \frac{\operatorname{lcm}(k, N-k+rt+1)}{k}$

In addition, such a PIR scheme has a complexity of $\mathcal{O}(smN^2)$ operations in $\mathbb{F}_{q^r}$ for the queries, $\mathcal{O}(smr^3)$ operations in $\mathbb{F}_q$ for the responses, and $\mathcal{O}(scN)$ operations over $\mathbb{F}_{q^r}$ for the file reconstruction. If m grows while all other parameters remain constant, such complexities are linear in m.

4.4. Worked example

In this subsection, we provide an example of the PIR scheme proposed in this manuscript. We will consider the simpler case of Subsection 4.1 (no folding or $b = 1$). We will consider dimension $k = 2$, locality $r = 2$, local distance $\delta = 2$, number of local groups $g = 2$ and protection against $t = 1$ colluding servers. The total number of nodes is $n = g(r + \delta - 1) = 6$, but after removing $\delta - 1 = 1$ redundant node per local groups, the number of remaining nodes is $N = gr = 4$. In order to consider Construction 1, we choose the field size $q = 3 > \max\{g, r + \delta - 3\}$. Hence $q^r = 9$. Choosing $s = 2$ iterations and $c = N - k - rt + 1 = 1$, we notice that the hypotheses of Subsection 4.1 are satisfied ($k = sc$ and $k + rt = N$). We keep the number of files m unrestricted (its value is not important for the example).

By considering the local codes in Construction 1 to be systematic, we may assume that, after removing the local redundancies, the remaining data is encoded with $\mathcal{C}_{out}$, that is, $\mathcal{C}_\Delta = \mathcal{C}_{out}$ as explained right after (4). Let $\alpha \in \mathbb{F}_9$ be such that $\alpha^2 = 2\alpha + 1$, which is a primitive element of $\mathbb{F}_9$. Let now $a_1 = 1$, $a_2 = \alpha$, $\beta_1 = 1$ and $\beta_2 = \alpha$. Notice that β_1 and β_2 are $\mathbb{F}_3$-linearly independent. Then we have

$$G_{out} = \left(\begin{array}{cc|cc} \beta_1 & \beta_2 & \beta_1 & \beta_2 \\ a_1\beta_1^q & a_1\beta_2^q & a_2\beta_1^q & a_2\beta_2^q \end{array}\right) = \left(\begin{array}{cc|cc} 1 & \alpha & 1 & \alpha \\ 1 & 2\alpha + 2 & \alpha & 2 \end{array}\right) \in \mathbb{F}_9^{2\times 4}.$$

Thus if $\mathbf{x}^i = (x_1^i, x_2^i) \in \mathbb{F}_{q^r}^k = \mathbb{F}_9^2$ is the ith file, we may consider its encoding as

$$(z_1^1, z_2^1, z_3^1, z_4^1) = (\mathbf{z}_1^1, \mathbf{z}_2^1) = (x_1^i, x_2^i) \left(\begin{array}{cc|cc} 1 & \alpha & 1 & \alpha \\ 1 & 2\alpha + 2 & \alpha & 2 \end{array}\right) =$$

$$\left(x_1^i + x_2^i, \quad \alpha x_1^i + (2\alpha+2)x_2^i \;\middle|\; x_1^i + \alpha x_2^i, \quad \alpha x_1^i + 2x_2^i \right) \in \mathbb{F}_9^4.$$

The first server stores $\mathbf{z}_1^1 = (x_1^i + x_2^i, \alpha x_1^i + (2\alpha+2)x_2^i) \in \mathbb{F}_9^2$ and the second server stores $\mathbf{z}_2^1 = (x_1^i + \alpha x_2^i, \alpha x_1^i + 2x_2^i) \in \mathbb{F}_9^2$.

We now show the three steps of the scheme for the first of the two iterations in order to recover the first file (i.e. $i = 1$):

Step 1, Queries: Generate independently and uniformly at random $\mathbf{w}_1^\ell, \mathbf{w}_2^\ell \in \mathbb{F}_9^2$ and, for $j = 1, 2$, compute

$$\mathbf{d}_j^\ell = (w_{j,1}^\ell, w_{j,2}^\ell) \left(\begin{array}{cc|cc} 1 & \alpha & 1 & \alpha \\ 1 & 2\alpha+2 & \alpha & 2 \end{array} \right) =$$

$$\left(w_{j,1}^\ell + w_{j,2}^\ell, \quad \alpha w_{j,1}^\ell + (2\alpha+2)w_{j,2}^\ell \;\middle|\; w_{j,1}^\ell + \alpha w_{j,2}^\ell, \quad \alpha w_{j,1}^\ell + 2w_{j,2}^\ell \right) \in \mathbb{F}_9^4.$$

By (11) and (21), we have

$$\mathbf{e}_1^1 = \left(M_{\boldsymbol{\beta}}^{-1} \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \mathbf{0} \right) = (\beta_1, 0, \ldots, 0) = (1, 0, \ldots, 0) \in \mathbb{F}_9^{2m},$$

and $\mathbf{e}_2^1 = \mathbf{0} \in \mathbb{F}_9^{2m}$. Hence the queries for the two servers are the vectors in $\mathbb{F}_9^{2m}$

$$\begin{array}{rcl} \mathbf{q}_1^1 & = & (\mathbf{d}_1^1, \mathbf{d}_1^2, \ldots, \mathbf{d}_1^m) + (1, 0, \ldots, 0), \text{ and} \\ \mathbf{q}_2^1 & = & (\mathbf{d}_2^1, \mathbf{d}_2^2, \ldots, \mathbf{d}_2^m), \end{array}$$

respectively.

Step 2, Responses: The responses from the servers are the vectors in $\mathbb{F}_9^2$

$$\begin{array}{rcl} \mathbf{r}_1^1 & = & \sum_{\ell=1}^m (\mathbf{z}_1^\ell \star \mathbf{d}_1^\ell) + \mathbf{z}_1^1 \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \sum_{\ell=1}^m (\mathbf{z}_1^\ell \star \mathbf{d}_1^\ell) + (z_1^1, 0), \text{ and} \\ \mathbf{r}_2^1 & = & \sum_{\ell=1}^m (\mathbf{z}_2^\ell \star \mathbf{d}_2^\ell), \end{array}$$

respectively.

Step 3, File reconstruction: The vector $\sum_{\ell=1}^m (\mathbf{z}_1^\ell \star \mathbf{d}_1^\ell)$ is a codeword in $\mathcal{C}_{4,3}(\mathbf{a}, \boldsymbol{\beta})$. Generator and parity-check matrices of such a code can be chosen, respectively, as

$$G = \begin{pmatrix} 1 & \alpha & 1 & \alpha \\ 1 & 2\alpha+2 & \alpha & 2 \\ 1 & \alpha & 2 & 2\alpha \end{pmatrix} \quad \text{and} \quad H = \begin{pmatrix} 2\alpha & 1 & 1 & 2\alpha+2 \end{pmatrix}.$$

Hence we have that

$$\mathbf{r}^1 H^T = (\mathbf{r}_1^1, \mathbf{r}_2^1) H^T = (z_1^1, 0, 0, 0) H^T = 2\alpha z_1^1.$$

Clearly we can recover z_1^1 from $2\alpha z_1^1$ since $2\alpha \neq 0$ is known. In this way, at the end of the first iteration we have obtained z_1^1. Analogously, in the second iteration we would obtain z_2^1. In other words, at the end of the whole process we recover

$$(z_1^1, z_2^1) = \mathbf{z}_1^1 = (x_1^1, x_2^1) \begin{pmatrix} 1 & \alpha \\ 1 & 2\alpha + 2 \end{pmatrix},$$

and since such a matrix is invertible, we may recover the first file $(x_1^1, x_2^1) \in \mathbb{F}_9^2$.

5. Further considerations

5.1. Unequal localities and local distances

The results in this work may be extended, in a straightforward way, to the case where each local group Γ_j has a different locality r_j and local distance δ_j, for $j = 1, 2, \ldots, g$. See the next subsection for a further extension. The MR-LRC in Construction 1 based on linearized Reed-Solomon codes can be extended to arbitrary equal or unequal localities and local distances as long as the field is $\mathbb{F}_{q^r}$, where $q > g$ and $r \geq \max\{r_1, r_2, \ldots, r_g\}$. See [24, Sec. III]. By choosing systematic generator matrices of the MDS local linear codes (which now are different), the remaining MDS storage code after removing all local redundancies is again a k-dimensional linearized Reed-Solomon code, although of length $N = \sum_{j=1}^{g} r_j$.

For $\tau \geq 1$ colluding nodes, the achieved rate would still be $R = (N - k - \tau + 1)/N$. However, $t \geq 1$ colluding local groups correspond in this case to a number of colluding servers that is different for different sets of local groups. In other words, the collusion pattern [35] is generated by maximal collusion sets of different sizes. We may still proceed with the strategy in this work, that is, we may consider protecting against any $\tau = \max\{\sum_{j \in T} r_j \mid T \subseteq [g], |T| = t\}$ colluding nodes. Improvements on the rate $R = (N - k - \tau + 1)/N$ may be possible for certain cases (as in [35, Sec. V]), which we leave open.

Finally, the main motivation behind unequal localities and local distances is that some local groups may require faster and/or more robust repair, for instance due to hot data, while global erasure correction may be improved by considering the different localities and local distances. See [5,17,41] for more details.

5.2. Arbitrary local linear codes and hierarchical localities

As before, the results in this work may be extended, in a straightforward way, to the case where each local group Γ_j uses an arbitrary r_j-dimensional local linear code $\mathcal{C}_{loc}^j \subseteq \mathbb{F}_q^{n_j}$, where $\Gamma_j = |n_j|$ and $r_j + \delta_j - 1 \leq n_j$, where $\delta_j = \mathrm{d}(\mathcal{C}_{loc}^j)$, for $j = 1, 2, \ldots, g$. Construction 1 still gives an MR-LRC for any choice of local linear codes, see [24, Sec. IV]. Furthermore, the local codes may be dynamically, efficiently and locally updated in

order to adapt to different distributed storage configurations, as discussed in [24, Subsec. V-A]. In particular, the local codes may be in turn MR-LRCs, giving rise to multi-layer or hierarchical MR-LRCs (see [24, Def. 7], [24, Subsec. V-B] and [26, Sec. II]), which have optimal global distance by [24, Th. 4]. As before, by choosing systematic generator matrices of the local codes, the remaining MDS storage code after removing the local redundancies is a linearized Reed-Solomon code of length $N = \sum_{j=1}^{g} r_j$.

5.3. PIR over linearly coded networks

Linear network coding [20] permits maximum information flow over a network from a source to several sinks simultaneously in one shot (*multicast*). In [38], PIR is considered where each server is formed by a number $r \geq 1$ of nodes in the database and communication between the user and each server is through a linearly coded network.

To avoid mixing information through the network for non-colluding sets of servers, it is assumed in [38] that the linearly coded networks between the user and the servers are pair-wise disjoint (after removing the user node). This makes the total transfer matrix from the user to the database and back have a block-diagonal shape $\mathrm{Diag}(A_1, A_2, \ldots, A_g) \in \mathbb{F}_q^{gr}$, where $A_j \in \mathbb{F}_{q^r}^{r}$ is the transfer matrix from the jth server to the user (we assume square transfer matrices for simplicity). In other words, the total linearly coded network from the user to the servers and back can be considered as a *multishot* linearly coded network as in [23], with one shot per server. The effect of such a channel is simply multiplying codewords by $\mathrm{Diag}(A_1, A_2, \ldots, A_g)$.

It was shown in [23, Subsec. V-F] that multiplying on the right a linearized Reed-Solomon code, as in Definition 7, by a block-diagonal matrix $\mathrm{Diag}(A_1, A_2, \ldots, A_g) \in \mathbb{F}_q^{gr}$ gives again a linearized Reed-Solomon code, possibly with erasures if the matrices A_j are not full-rank. Using this fact, our PIR scheme (Subsection 4.2) may be used mutatis mutandis in the scenario described in this subsection and in [38]. In the error-free and erasure-free case, the rate obtained in both works is

$$R = \frac{N - k - rt + 1}{N}.$$

However, since Gabidulin codes [12] are used in [38], the required field size is q_0^{gr}, where $q_0 \geq 2$ is the field size of the underlying linear network code. Note that q_0^{gr} is exponential in the number of servers g, whereas our scheme would still require the field size g^r, which is polynomial in the number of servers g.

5.4. Systematic and non-systematic codes

All of the results in this manuscript hold for any generator and parity-check matrix of the linear codes involved. Observe that we only need: 1) The fact that the coordinate-wise matrix product of two linearized Reed-Solomon codes is a linearized Reed-Solomon code, in Step 3 of our PIR scheme; 2) Using a parity-check matrix, systematic or not, of

such a coordinate-wise matrix product of linearized Reed-Solomon codes, in Step 3 of our PIR scheme; and 3) the fact that the remaining MDS code $\mathcal{C}_\Delta$, after removing all local redundancies $\Gamma_j \setminus \Delta_j$, is a linearized Reed-Solomon code. To ensure the last condition, we made the assumption in Section 2 that the generator matrix $A \in \mathbb{F}_q^{r\times(r+\delta-1)}$ is systematic, having its first r columns equal to those of the identity matrix. However, this assumption can be easily lifted. This is because, if $A_r \in \mathbb{F}_q^{r\times r}$ is formed by the first r columns of the matrix A, whether A_r is the identity matrix or not, it holds that

$$\mathcal{C}_{N,k}(\mathbf{a},\boldsymbol{\beta})A_r = \mathcal{C}_{N,k}(\mathbf{a},\boldsymbol{\beta}A_r),$$

with notation as in Definition 7, where $\boldsymbol{\beta}A_r \in \mathbb{F}_{q^r}^r$ is another ordered basis of $\mathbb{F}_{q^r}$ over $\mathbb{F}_q$ (see also [23, Subsec. V-F]). Thus our PIR scheme still works in this case, simply by replacing $\boldsymbol{\beta}$ by $\boldsymbol{\beta}A_r$.

Data availability

No data was used for the research described in the article.

Acknowledgment

The author gratefully acknowledges the support from The Independent Research Fund Denmark (Grant No. DFF-7027-00053B). The author also wishes to thank the anonymous reviewers, who helped improve the presentation of the manuscript.

References

[1] K. Banawan, S. Ulukus, The capacity of private information retrieval from coded databases, IEEE Trans. Inf. Theory 64 (3) (2018).
[2] S.R. Blackburn, T. Etzion, M.B. Paterson, PIR schemes with small download complexity and low storage requirements, in: Proc. IEEE Int. Symp. Info. Theory, June 2017, pp. 146–150.
[3] M. Blaum, J.L. Hafner, S. Hetzler, Partial-MDS codes and their application to RAID type of architectures, IEEE Trans. Inf. Theory 59 (7) (July 2013) 4510–4519.
[4] G. Calis, O.O. Koyluoglu, A general construction for PMDS codes, IEEE Commun. Lett. 21 (3) (March 2017) 452–455.
[5] B. Chen, S.T. Xia, J. Hao, Locally repairable codes with multiple (r_i, δ_i)-localities, in: Proc. IEEE Int. Symp. Info. Theory, June 2017, pp. 2038–2042.
[6] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, Private information retrieval, in: Proce. 36th Annual Symposium on Foundations of Computer Science, FOCS '95, 1995, p. 41.
[7] B. Chor, E. Kushilevitz, O. Goldreich, M. Sudan, Private information retrieval, J. ACM 45 (6) (November 1998) 965–981.
[8] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley-Interscience, 2006.
[9] R.G.L. D'Oliveira, S. El Rouayheb, One-shot PIR: refinement and lifting, IEEE Trans. Inf. Theory 66 (4) (2020) 2443–2455.
[10] A. Fazeli, A. Vardy, E. Yaakobi, Codes for distributed PIR with low storage overhead, in: Proc. IEEE Int. Symp. Info. Theory, June 2015, pp. 2852–2856.
[11] R. Freij-Hollanti, O. Gnilke, C. Hollanti, D. Karpuk, Private information retrieval from coded databases with colluding servers, SIAM J. Appl. Algebra Geom. 1 (1) (2017) 647–664.

[12] E.M. Gabidulin, Theory of codes with maximum rank distance, Probl. Inf. Transm. 21 (1) (1985) 1–12.
[13] P. Gopalan, C. Huang, B. Jenkins, S. Yekhanin, Explicit maximally recoverable codes with locality, IEEE Trans. Inf. Theory 60 (9) (Sept 2014) 5245–5256.
[14] P. Gopalan, C. Huang, H. Simitci, S. Yekhanin, On the locality of codeword symbols, IEEE Trans. Inf. Theory 58 (11) (Nov 2012) 6925–6934.
[15] S. Gopi, V. Guruswami, S. Yekhanin, On maximally recoverable local reconstruction codes, Electron. Colloq. Comput. Complex. 24 (183) (2017).
[16] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, S. Yekhanin, Erasure coding in Windows Azure storage, in: 2012 USENIX Annual Technical Conference, Boston, MA, 2012, pp. 15–26.
[17] S. Kadhe, A. Sprintson, Codes with unequal locality, in: Proc. IEEE Int. Symp. Info. Theory, July 2016, pp. 435–439.
[18] G.M. Kamath, N. Prakash, V. Lalitha, P.V. Kumar, Codes with local regeneration and erasure correction, IEEE Trans. Inf. Theory 60 (8) (Aug 2014) 4637–4660.
[19] J. Lavauzelle, R. Tajeddine, R. Freij-Hollanti, C. Hollanti, Private information retrieval schemes with product-matrix MBR codes, IEEE Trans. Inf. Forensics Secur. 16 (2020) 441–450.
[20] S.-Y.R. Li, R.W. Yeung, Ning Cai, Linear network coding, IEEE Trans. Inf. Theory 49 (2) (February 2003) 371–381.
[21] I. Márquez-Corbella, R. Pellikaan, A characterization of MDS codes that have an error correcting pair, Finite Fields Appl. 40 (2016) 224–245.
[22] U. Martínez-Peñas, Skew and linearized Reed-Solomon codes and maximum sum rank distance codes over any division ring, J. Algebra 504 (2018) 587–612.
[23] U. Martínez-Peñas, F.R. Kschischang, Reliable and secure multishot network coding using linearized Reed-Solomon codes, IEEE Trans. Inf. Theory 65 (8) (2019) 4785–4803.
[24] U. Martínez-Peñas, F.R. Kschischang, Universal and dynamic locally repairable codes with maximal recoverability via sum-rank codes, IEEE Trans. Inf. Theory 65 (12) (2019) 7790–7805.
[25] U. Martínez-Peñas, R. Pellikaan, Rank error-correcting pairs, Des. Codes Cryptogr. 84 (1–2) (2017) 261–281.
[26] A.M. Nair, V. Lalitha, Maximally recoverable codes with hierarchical locality, in: 2019 National Conference on Communications (NCC), 2019, pp. 1–6, Preprint, https://arxiv.org/abs/1901.02867.
[27] O. Ore, Theory of non-commutative polynomials, Ann. Math. (2) 34 (3) (1933) 480–508.
[28] I.S. Reed, G. Solomon, Polynomial codes over certain finite fields, J. Soc. Ind. Appl. Math. 8 (2) (1960) 300–304.
[29] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A.G. Dimakis, R. Vadali, S. Chen, D. Borthakur, XORing elephants: novel erasure codes for big data, in: Proc. 39th Int. Conf. Very Large Data Bases, PVLDB'13, 2013, pp. 325–336.
[30] N.B. Shah, K.V. Rashmi, K. Ramchandran, One extra bit of download ensures perfectly private information retrieval, in: Proc. IEEE Int. Symp. Info. Theory, June 2014, pp. 856–860.
[31] V. Skachek, Batch and PIR codes and their connections to locally repairable codes, in: Network Coding and Subspace Designs, 2018, pp. 427–442.
[32] H. Sun, S.A. Jafar, The capacity of private information retrieval, IEEE Trans. Inf. Theory 63 (7) (July 2017) 4075–4088.
[33] H. Sun, S.A. Jafar, The capacity of robust private information retrieval with colluding databases, IEEE Trans. Inf. Theory 64 (4) (April 2018) 2361–2370.
[34] H. Sun, S.A. Jafar, Private information retrieval from MDS coded data with colluding servers: settling a conjecture by Freij-Hollanti et al., IEEE Trans. Inf. Theory 64 (2) (Feb 2018) 1000–1022.
[35] R. Tajeddine, O.W. Gnilke, D. Karpuk, R. Freij-Hollanti, C. Hollanti, S.E. Rouayheb, Private information retrieval schemes for coded data with arbitrary collusion patterns, in: Proc. IEEE Int. Symp. Info. Theory, June 2017, pp. 1908–1912.
[36] R. Tajeddine, O.W. Gnilke, S. El Rouayheb, Private information retrieval from MDS coded data in distributed storage systems, IEEE Trans. Inf. Theory 64 (11) (Nov 2018) 7081–7093.
[37] R. Tajeddine, S. El Rouayheb, Private information retrieval from MDS coded data in distributed storage systems, in: Proc. IEEE Int. Symp. Info. Theory, July 2016, pp. 1411–1415.
[38] R. Tajeddine, A. Wachter-Zeh, C. Hollanti, Private information retrieval over random linear networks, IEEE Trans. Inf. Forensics Secur. 15 (2020) 790–799.
[39] I. Tamo, A. Barg, A family of optimal locally recoverable codes, IEEE Trans. Inf. Theory 60 (8) (Aug 2014) 4661–4676.

[40] S. Yekhanin, Locally Decodable Codes and Private Information Retrieval Schemes. Information Theory and Security, 1st edition, Springer, 2010.
[41] A. Zeh, E. Yaakobi, Bounds and constructions of codes with multiple localities, in: Proc. IEEE Int. Symp. Info. Theory, July 2016, pp. 640–644.
[42] Y. Zhang, G. Ge, A general private information retrieval scheme for MDS coded databases with colluding servers, Preprint, https://arxiv.org/abs/1704.06785, 2017.