Enabling optimal access and error correction for the repair of Reed-Solomon codes

Zitan Chen Min Ye Alexander Barg

***Abstract*—Recently Reed-Solomon (RS) codes were shown to possess a repair scheme that supports repair of failed nodes with optimal repair bandwidth. In this paper, we extend this result in two directions. First, we propose a new repair scheme for the RS codes constructed in [Tamo-Ye-Barg, *IEEE Transactions on Information Theory*, vol. 65, May 2019] and show that repair is robust to erroneous information provided by the helper nodes while maintaining the optimal repair bandwidth. Second, we construct a new family of RS codes with optimal access for the repair of any single failed node. We also show that the constructed codes can accommodate both features, supporting optimal-access repair with optimal error-correction capability.**

Going beyond RS codes, we also prove that any scalar MDS code with repair bandwidth attaining the cutset bound affords a repair scheme with optimal access property.

***Index Terms*—Distributed storage, Regenerating codes, Reed-Solomon codes, Error correction, Optimal access.**

I. Introduction

The problem of efficient erasure correction in various classes of algebraic codes has recently attracted renewed attention because of its links to applications of erasure coding for distributed storage. Compared to the classic setting of erasure correction, efficient functioning of distributed storage systems critically depends on the volume of communication exchanged between the nodes for the purposes of data recovery. The constraint on the amount of communication, termed "repair bandwidth," adds new features to the problem, and has motivated a large amount of research in coding theory in the last decade.

Consider an (n, k, l) array code $\mathcal{C}$ over a finite field F, i.e., a collection of codewords $c = (c_1, \dots, c_n)$, where $c_i = (c_{i,0}, c_{i,1}, \dots, c_{i,l-1})^T \in F^l, i = 1, \dots, n$. A node $c_i, i \in [n]$ can be repaired from a subset of $d \geqslant k$ helper nodes $\{c_j : j \in \mathcal{R}\}, \mathcal{R} \subseteq [n]\backslash\{i\}$, by downloading $\beta_i(\mathcal{R})$ symbols of F if there are numbers $\beta_{ij}, j \in \mathcal{R}$, functions $f_{ij} : F^l \to F^{\beta_{ij}}, j \in \mathcal{R}$, and a function $g_i : F^{\sum_{j\in\mathcal{R}} \beta_{ij}} \to F^l$ such that

$$c_i = g_i(\{f_{ij}(c_j), j \in \mathcal{R}\}) \text{ for all } c = (c_1, \dots, c_n) \in \mathcal{C}$$

and

$$\sum_{j\in\mathcal{R}} \beta_{ij} = \beta_i(\mathcal{R}).$$

Codes that we consider form linear spaces over F. If $\mathcal{C}$ is not linear over F^l, it is also called a *vector* code, while if it is, it is called *scalar* to stress the linearity property. A code $\mathcal{C}$ is called MDS if any k coordinates $\{c_{j_i}, i = 1, \dots, k\}$ of the codeword suffice to recover its remaining $n-k$ coordinates. In this paper we study the repair problem of scalar MDS codes.

It is well known [1] that for any MDS code $\mathcal{C}$ (scalar or vector), any $i \in [n]$, and any $\mathcal{R} \subseteq [n]\backslash\{i\}$ of cardinality $|\mathcal{R}| \geqslant k$, we have

$$\beta_i(\mathcal{R}) \geqslant \frac{l}{|\mathcal{R}| - k + 1}|\mathcal{R}|. \tag{1}$$

For an MDS code $\mathcal{C}$, we define the minimum bandwidth of repair of a node from a d-subset $\mathcal{R}$ of helper nodes as $\beta(d) = \max_{i\in[n]} \min_{\mathcal{R}\subset[n]\backslash\{i\},|\mathcal{R}|=d} \beta_i(\mathcal{R})$. It follows immediately from (1) that

$$\beta := \beta(d) \geqslant \frac{l}{d - k + 1}d. \tag{2}$$

An MDS code that attains the bound (2) with equality is said to afford optimal repair, and a repair scheme that attains this bound is called optimal. Such codes are also termed *minimium storage regenerating* or MSR codes, and the parameter l is called node size or sub-packetization. Multiple constructions of vector MDS codes with optimal repair are available in the literature, including papers [2], [3], [4], [5], [6], [7].

The basic repair problem of MDS codes has been extended to the case that some of the helper nodes provide erroneous information (or arbitrary nature). Suppose that a subset of e nodes out of d helpers provide erroneous information and define $\beta(d, e)$ to be the minimum number of symbols needed to repair a failed node in the presence of such errors. It was shown [8], [9] that for $d \geqslant k + 2e$,

$$\beta(d, e) \geqslant \frac{dl}{d - 2e - k + 1}. \tag{3}$$

A repair scheme that achieves this bound is said to have *optimal error correction capability*. Constructions of MDS array codes with optimal error correction capability are presented, for instance, in [4].

Another parameter of erasure codes for distributed storage that affects the system performance is the so-called access, or input-output cost of repair. Indeed, while the code may support parsimonious exchange between the helper nodes and the repair center, generation of the symbols to be transmitted from the helper node may require reading the entire contents of the node (trivial access), which increases delays in the

The results of this paper were presented in part at the 2020 IEEE International Symposium on Information Theory.

Zitan Chen is with Dept. of ECE and ISR, University of Maryland, College Park, MD 20742. Email: chenztan@umd.edu. His research was supported by NSF grants CCF1618603 and CCF1814487.

Min Ye is with the Data Science and Information Technology Research Center, Tsinghua-Berkeley Shenzhen Institute, Tsinghua Shenzhen International Graduate School, Shenzhen, 518055, China. Email: yeemmi@gmail.com

Alexander Barg is with Dept. of ECE and ISR, University of Maryland, College Park, MD 20742 and also with IITP, Russian Academy of Sciences, 127051 Moscow, Russia. Email: abarg@umd.edu. His research was supported by NSF grant CCF1618603 and CCF1814487.

system. The smallest number of symbols accessed on each of the helper nodes in an MSR code is $l/(d-k+1)$, and such codes are said to have the *optimal access property*. Advantages of having this property are well recognized in the literature starting with [10], and a number of papers were devoted to constraints that it imposes on the code parameters such as sub-packetization [11], [12]. Many families of MSR codes including early constructions in [3], [13] as well as code families for general parameters in [4], [5], [14], [15] have the optimal access property.

The optimal-access repair and optimal error correction capability can be combined. According to (3), we say that a code family/repair scheme have both properties if repair can be performed in the presence of e errors, while the number of symbols accessed on each of the helper nodes equals $1/(d-2e-k+1)$ proportion of the contents of each of d helper nodes.

While the aforementioned papers mostly deal with vector codes, in this paper we focus on the repair problem for scalar MDS codes, more specifically, for Reed-Solomon (RS) codes. This code family continues to attract attention in multiple aspects of theoretical research (list decoding of variants of RS codes, locally recoverable codes, to name a few) and it is also one of the most used coding methods in a vast variety of practical systems. The first work to isolate and advance the repair problem for RS codes was [16] which itself followed and developed the ideas in [17]. In [16], the authors view each coordinate of RS codes as a vector over some subfield and characterize linear repair schemes of RS codes over this subfield. For RS codes (and more generally for scalar codes), the node size l is defined as the degree of extension of the symbol field over the subfield. Following [16], several papers attempted to optimize the repair bandwidth of RS codes [18], [19], [20]. A family of optimal-repair RS codes in the case of repairing a single failed node as well as multiple nodes was constructed in [21]. Later this construction was extended to the case of the rack-aware storage model, resulting in a family of codes with optimal repair of a single node [15], and this problem was addressed again in [22].

In this paper we address two problems related to RS repair, namely,

(i) repair schemes of RS codes with optimal error correction, and
(ii) RS codes with optimal-access repair.

Error correction during repair of failed storage nodes was previously only considered for vector codes [8], [9], [4]. The problem of low-access RS codes was studied in [23]–[25]. In particular, the last of these works analyzed the access (input/output) cost of the family of RS codes of [21], providing an estimate of this parameter, but stopping short of achieving optimal access.

Our main results provide a solution to problems (i)-(ii). Specifically, we construct a repair scheme for RS codes in [21] that has optimal error correction capability (i.e., attains the bound (3)), and we also construct a family of RS codes with optimal access repair for any single failed node. Additionaly, we prove that the constructed codes can be furnished with a repair scheme that supports both optimal error correction and optimal-access repair.

Apart from this, we also show that any scalar MDS code with optimal repair of a single node from d helpers, $k \leqslant d \leqslant n-1$, affords a repair scheme with optimal access, and this includes the RS codes in [21]. While our arguments do not provide an explicit construction, we give a combinatorial search procedure, showing that it exists for any scalar MSR code. The resulting optimal access codes have the same sub-packetization as the original MDS codes.

The constructions are technically involved, and we begin in Sec. II with illustrating them in an example. The three sections that follow it are devoted to the results described above.

II. A simple example

In this section, we construct an RS code together with a repair scheme that can recover its *first node* with both optimal access and optimal error correction capability.

A. Preliminaries

1) We begin with some standard definitions. Recall that a *generalized RS code* (GRS code) of length n and dimension k over a finite field F is obtained by fixing a set of n distinct evaluation points $\Omega := \{\alpha_1, \alpha_2, \dots, \alpha_n\} \subset F$ and a vector $(v_1, \dots, v_n) \in (F^*)^n$ with no zero coordinates. Then the GRS code is the set of vectors

$$\mathrm{GRS}_F(n,k,v,\Omega) = \{(v_1 f(\alpha_1), v_2 f(\alpha_2), \dots, v_n f(\alpha_n)) : f \in F[x], \deg f < k\}.$$

In particular, if $(v_1, \dots, v_n) = (1, \dots, 1)$, then the GRS code is called the Reed-Solomon (RS) code and is denoted by $\mathrm{RS}_F(n,k,\Omega)$. It is a classic fact that the dual code $(\mathrm{RS}_F(n,k,\Omega))^\perp$ is $\mathrm{GRS}_F(n,n-k,v,\Omega)$, where $v \in (F^*)^n$ is some vector. In particular, if $c = (c_1, \dots, c_n) \in F^n$ is a vector such that $\sum_{i=1}^n c_i h(\alpha_i) = 0$ for every polynomial $h(x)$ of degree $\leqslant k-1$, then c is contained in a GRS_F code of dimension $n-k$. Rephrasing this, we have the following obvious proposition that will be frequently used below.

Proposition 1. *Let $c = (c_1, \dots, c_n) \in F^n$ and suppose that $\sum_{i=1}^n c_i \alpha_i^t = 0$ for all $t = 0, 1, \dots, k-1$. Then the vector c is contained in a code $\mathrm{GRS}_F(n, n-k, v, \Omega)$, where $v \in (F^*)^n$ and $\Omega = \{\alpha_1, \dots, \alpha_n\}$.*

Let E be an algebraic extension of F of degree s. The *trace mapping* $\mathrm{tr}_{E/F}$ is given by $x \mapsto 1 + x^{|F|} + x^{|F|^2} + \dots + x^{|F|^{s-1}}$. For any basis $\gamma_0, \dots, \gamma_{s-1}$ of E over F there exists a *trace-dual basis* $\delta_0, \dots, \delta_{s-1}$, which satisfies $\mathrm{tr}_{E/F}(\gamma_i \delta_j) = \mathbb{1}_{\{i=j\}}$ for all pairs i, j. For an element $x \in E$ the coefficients of its expansion in the basis (γ_i) are found using the dual basis, specifically, $x = \sum_{i=0}^{s-1} \mathrm{tr}_{E/F}(x\delta_i)\gamma_i$. As a consequence, for any basis (δ_i) the mapping $E \to F^s$ given by $x \mapsto (\mathrm{tr}(x\delta_i), i = 0, \dots, s-1)$ is a bijection.

2) Before we define the RS code that will be considered below, let us fix the parameters of the repair scheme. We attempt to repair a failed node using information from d helper nodes. Suppose that at most e of them provide erroneous information. Assume that $d-2e \geqslant k$, and let $s := d-2e-k+1$. Let F be a finite field of size $|F| \geqslant n-1$. Choose a set

of distinct evaluation points $\Omega := \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ such that $\alpha_i \in F$ for all $2 \leqslant i \leqslant n$ and α_1 is an algebraic element of degree s over F (which means that the extension field $E := F(\alpha_1)$ forms an s-dimensional vector space over F). Consider the code

$$\mathcal{C} := \mathrm{RS}_E(n, k, \Omega).$$

In this section we present a repair scheme of the code $\mathcal{C}$ that can repair the first node of $\mathcal{C}$ over the field F; in other words, we represent the coordinates of $\mathcal{C}$ as s-dimensional vectors over F in some basis of E over F. Thus, the node size of this code is s. We note that the code $\mathcal{C}$ represented in this way is still a scalar code.

The repair scheme presented below has the following two properties:

- the optimal error correction capability, i.e., the repair bandwidth achieves the bound (3) for any pair (d, e) such that $d - 2e = s + k - 1$;
- in the absence of errors it has the optimal access property, i.e., the number of symbols accessed during the repair process is d. Thus, in this case $e = 0$ and $s = d - k + 1$.

B. Repair scheme with optimal error correction capability

Let $c = (c_1, c_2, \ldots, c_n) \in \mathcal{C}$ be a codeword and suppose that c_1 is erased. Since $\mathcal{C}^\perp = \mathrm{GRS}_E(n, n-k, v, \Omega)$ for some $v \in (E^*)^n$, we have

$$v_1\alpha_1^t c_1 + v_2\alpha_2^t c_2 + \cdots + v_n\alpha_n^t c_n = 0, \quad t = 0, 1, \ldots, n-k-1,$$

or

$$v_1\alpha_1^t c_1 = -v_2\alpha_2^t c_2 - \cdots - v_n\alpha_n^t c_n, \quad t = 0, 1, \ldots, n-k-1. \tag{4}$$

Evaluating the trace $\mathrm{tr} = \mathrm{tr}_{E/F}$ on both sides of (4), we obtain the relation

$$\begin{aligned} \mathrm{tr}(v_1\alpha_1^t c_1) &= -\mathrm{tr}(v_2\alpha_2^t c_2) - \cdots - \mathrm{tr}(v_n\alpha_n^t c_n) \\ &= -\alpha_2^t \,\mathrm{tr}(v_2 c_2) - \cdots - \alpha_n^t \,\mathrm{tr}(v_n c_n), \\ &\qquad t = 0, 1, \ldots, n-k-1, \end{aligned} \tag{5}$$

where the second equality follows from the fact that $\alpha_2, \ldots, \alpha_n \in F$. Therefore, knowing the values of $(\mathrm{tr}(v_2c_2), \ldots, \mathrm{tr}(v_nc_n))$ enables us to compute $\mathrm{tr}(v_1\alpha_1^t c_1)$ for all $0 \leqslant t \leqslant n-k-1$. Since $\deg_F(\alpha_1) = s$, the elements $1, \alpha_1, \ldots, \alpha_1^{s-1}$ form a basis of E over F. As a consequence, one can recover c_1 from the values of $\{\mathrm{tr}(v_1\alpha_1^t c_1) : 0 \leqslant t \leqslant s-1\}$. By definition, $s - 1 = d - 2e - k \leqslant n - k - 1$, so $\{\mathrm{tr}(v_1\alpha_1^t c_1) : 0 \leqslant t \leqslant s-1\} \subseteq \{\mathrm{tr}(v_1\alpha_1^t c_1) : 0 \leqslant t \leqslant n-k-1\}$. Combining this with (5), we see that the value c_1 is fully determined by the set of elements $(\mathrm{tr}(v_2c_2), \ldots, \mathrm{tr}(v_nc_n))$.

Recalling our problem, we will show that in order to repair c_1, it suffices to acquire the values $\mathrm{tr}(v_ic_i)$ from any d helper nodes provided that at least $d - e = (d + s + k - 1)/2$ of these values are correct. This will follow from the following proposition.

Proposition 2. *Let $f(x) \in F[x]$ be the minimal polynomial of α_1. For any $s < n - k$ and any $c = (c_1, \ldots, c_n) \in \mathcal{C}$ the vectors $(f(\alpha_2)\,\mathrm{tr}(v_2c_2), \ldots, f(\alpha_n)\,\mathrm{tr}(v_nc_n))$ are contained in an $(n-1, s+k-1)$ GRS code over F.*

Proof. Let $\mathcal{T} := \{0, 1, \ldots, n-k-s-1\}$. Since $\alpha_i \in F, i = 2, \ldots, n$ by definition we have $f(\alpha_i) \neq 0$ for all such i. Next, $\deg(f) = s$, and thus for all $t \in \mathcal{T}$

$$(v_1\alpha_1^t f(\alpha_1), v_2\alpha_2^t f(\alpha_2), \ldots, v_n\alpha_n^t f(\alpha_n)) \in \mathcal{C}^\perp.$$

This implies that for all $t \in \mathcal{T}$

$$v_1\alpha_1^t f(\alpha_1)c_1 + v_2\alpha_2^t f(\alpha_2)c_2 + \cdots + v_n\alpha_n^t f(\alpha_n)c_n = 0,$$

but $f(\alpha_1) = 0$, so taking the trace, we obtain

$$\alpha_2^t f(\alpha_2)\,\mathrm{tr}(v_2c_2) + \cdots + \alpha_n^t f(\alpha_n)\,\mathrm{tr}(v_nc_n) = 0, \quad t \in \mathcal{T}. \tag{6}$$

By Proposition 1, this implies that the vectors $(f(\alpha_2)\,\mathrm{tr}(v_2c_2), \ldots, f(\alpha_n)\,\mathrm{tr}(v_nc_n))$ are contained in a GRS code of length $n-1$ with $n-s-k$ parities. □

The GRS code identified in this proposition can be punctured to any subset $\mathcal{R}$ of d coordinates, retaining the dimension and the MDS property. This means that the punctured code is capable of correcting any $e = (d-s-k+1)/2$ errors. Therefore, as long as no more than e helper nodes provide incorrect information, we can always recover $(\mathrm{tr}(v_2c_2), \ldots, \mathrm{tr}(v_nc_n))$ by acquiring a subset $\{\mathrm{tr}(v_{i_j}c_{i_j}), j = 1, \ldots, d\}$ from any d helper nodes and correcting the errors based on any decoding procedure of the underlying MDS code. Finally note that the case $s = n - k$ can be added trivially because then $d = n-1$ and $e = 0$, so all the helper nodes provide accurate information, and no error correction is required (or possible).

C. Optimal access property

Following the discussion in the first part of this section, we show that the code $\mathcal{C} = \mathrm{RS}_E(n, k, \Omega)$ defined above supports optimal-access repair of the node c_1. In this part we assume that the helper nodes provide accurate information about their contents, and we do not attempt error correction.

To represent the code, we choose a pair of trace-dual bases $(b_i), (b_i^*)$ of E over F, where we assume w.l.o.g. that $b_0 = 1$. Next, represent the ith coordinate of the code, $i \in \{1, \ldots, n\}$, using the basis $(v_i^{-i}b_m, m = 0, \ldots, s-1)$, where $(v_1, \ldots, v_n)$ is defined by the code $\mathcal{C}^\perp$. Namely, for a codeword $c \in \mathcal{C}$ we have

$$c_i = v_i^{-1} \sum_{m=0}^{s-1} c_{i,m} b_m^*, \tag{7}$$

where $c_{i,m} \in F$ for all $m = 0, 1, \ldots, s-1$. We assume that each storage node contains the vector $(c_{i,0}, c_{i,1}, \ldots, c_{i,s-1})$.

As discussed above, the value c_1 can be recovered from any d-subset of the set of elements $\{\mathrm{tr}(v_ic_i), j = 2, \ldots, n\}$. Further, for all $i = 2, \ldots, n$ and $m = 0, \ldots, s-1$ we have $\mathrm{tr}(v_ic_ib_m) = c_{i,m}$, so in particular,

$$\mathrm{tr}(v_ic_i) = c_{i,0}.$$

Thus, to repair c_1 it suffices to access and download a single symbol $c_{i,0}$ from the chosen subset of d helper nodes. According to the bound (1), the minimum number of symbols downloaded from a helper node for optimal repair is the $(1/s)$th proportion of the node's contents. Overall this shows that the repair scheme considered above has the optimal access property.

The above discussion sets the stage for constructing RS codes with optimal-access repair for each of the n coordinates. Namely, we took a basis $1, b_1, \ldots, b_{s-1}$ of E over F and represented each c_i in the basis $(v_i^{-1} b_i^*)$. The only element of the helper coordinate that we access and download is $c_{i,0}$. For more complicated constructions of RS codes, e.g., the ones constructed in [21] and below in the paper, we assume that E is an l-degree extension of F. The known repair schemes require to download elements of the form $\mathrm{tr}(v_i c_i a_0), \mathrm{tr}(v_i c_i a_1), \ldots, \mathrm{tr}(v_i c_i a_{(l/s)-1})$, where $a_0, a_1, \ldots, a_{(l/s)-1}$ are linearly independent over F. In this case, we can extend the set $a_0, a_1, \ldots, a_{(l/s)-1}$ to a basis (b_i) of E over F. Following the approach in (7), we store the code coordinate c_i as the vector of its coefficients $(c_{i,0}, c_{i,1}, \ldots, c_{i,l-1})$ in the dual basis $(b_i^*, i = 0, \ldots, l-1)$ of the basis (b_i). Since $c_{i,m} = \mathrm{tr}(v_i c_i a_m)$ for all $m = 0, 1, \ldots, l/s-1$, this choice of the basis enables one to achieve optimal access. This idea underlies the construction presented below in Sec. IV-A.

D. Optimal access with error correction

Thus far, we have assumed that errors are absent for optimal-access repair. To complete the picture, we address the case of codes with both optimal access and optimal error correcting capability for the repair of node c_1. It is easily seen that both properties can be combined. Indeed, since $\mathrm{tr}(v_i c_i) = c_{i,0}$ for all $i = 2, \ldots, n$, and since by Proposition 2 these elements form a codeword of a GRS code, it is immediately clear that c_1 can be repaired with optimal error correction capability and optimal access. To enable this property for any c_i, below we add extra features to the general repair scheme with optimal access. Specifically, error correction and optimal access are based on two different structures supported by the code. We show that it is possible to realize the error-correction structure in an extension field located between the base field and the symbol field of the code. Further reduction to the base field enables us to perform repair with optimal access. These ideas are implemented in detail in Sec. IV-B below.

III. Enabling error correction for repair of RS codes of [21]

In this section we propose a new repair scheme for the optimal-repair family of RS codes of [21] that supports the optimal error correction capability.

A. Preliminaries

We begin with briefly recalling the definition of the subfamily of RS codes of [21]. The construction depends on the number of helper nodes d used for the purpose of repair of a single node, $k \leqslant d \leqslant n-1$.

Definition 1 ([21]). *Let p be a prime, let $s := d - k + 1$, and let $p_1, \ldots, p_n$ be distinct primes that satisfy the condition $p_i \equiv 1 \bmod s, i = 1, \ldots, n$, Let $\mathcal{C} := RS_{\mathbb{K}}(n, k, \Omega)$ be a Reed-Solomon code, where*

- *$\Omega = \{\alpha_1, \ldots, \alpha_n\}$, where $\alpha_i, i = 1, \ldots, n$ is an algebraic element of degree p_i over $\mathbb{F}_p$,*
- *$\mathbb{K} = \mathbb{F}(\beta)$, where β is an algebraic element of degree s over $\mathbb{F} := \mathbb{F}_p(\alpha_1, \ldots, \alpha_n)$.*

As shown in [21], this code supports optimal repair of any node i from any set of d helper nodes in $[n] \backslash \{i\}$. Below we use this construction, choosing the value of s based not only on the number of helpers but also on the target number of errors tolerated by the repair procedure.

In this section we consider an RS code $\mathcal{C}$ given by Def. 1, where we take $s = d - 2e - k + 1$. For this code we will present a new repair scheme that has the property of optimal error correction. This repair scheme as well as the original repair scheme developed in [21] rely on the following lemma:

Lemma 3 ([21], Lemma 1). *Let F be a finite field. Let r be a prime such that $r \equiv 1 \bmod s$ for some $s \geqslant 1$. Let α be an element of degree r over F and β be of degree s over the field $F(\alpha)$. Let $K = F(\alpha, \beta)$ be the extension field of degree rs. Consider the F-linear subspace S of dimension r with the basis*

$$E := \{\beta^u \alpha^{u+qs} \mid u = 0, \ldots, s-1; q = 0, \ldots, \tfrac{r-1}{s} - 1\} \bigcup \Big\{ \sum_{u=0}^{s-1} \beta^u \alpha^{r-1} \Big\}.$$

Then $S + S\alpha + \cdots + S\alpha^{s-1} = K$, and this is a direct sum.

Without loss of generality, we only present the repair scheme for the first node c_1, and all the other nodes can be repaired in the same way (this is different from the previous section where the code was designed to support optimal repair *only* of the node c_1). The scheme is complicated, and we take time to develop it, occasionally repeating similar arguments more than once rather than compressing the presentation.

The repair of c_1 is conducted over the field $F_1 := \mathbb{F}_p(\alpha_2, \alpha_3, \ldots, \alpha_n)$. It is clear that $\mathbb{F} = F_1(\alpha_1)$ and $\mathbb{K} = \mathbb{F}(\beta)$, where $\deg_{F_1}(\alpha_1) = p_1$ and $\deg_{\mathbb{F}}(\beta) = s$. Below we use $\mathrm{tr} = \mathrm{tr}_{\mathbb{K}/F_1}$ to denote the trace mapping from $\mathbb{K}$ to F_1.

Define the set

$$E_1 := \{\beta^u \alpha_1^{u+qs} \mid u = 0, \ldots, s-1; q = 0, \ldots, \tfrac{p_1-1}{s} - 1\} \bigcup \Big\{ \sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1} \Big\}. \quad (8)$$

Clearly, $|E_1| = p_1$, and we write the elements in E_1 as $e_0, e_1, \ldots, e_{p_1-1}$. Then Lemma 3 implies that the set of elements

$$\{e_i \alpha_1^j : i = 0, \ldots, p_1 - 1, j = 0, \ldots, s-1\} \quad (9)$$

forms a basis of $\mathbb{K}$ over F_1.

Let $\mathcal{C}^\perp = \mathrm{GRS}_{\mathbb{K}}(n, n-k, v, \Omega)$ be the dual code. For every codeword $(c_1, \ldots, c_n) \in \mathcal{C}$ we have

$$v_1 \alpha_1^t c_1 + v_2 \alpha_2^t c_2 + \cdots + v_n \alpha_n^t c_n = 0, \qquad t = 0, 1, \ldots, n-k-1.$$

Multiplying by e_i on both sides of the equation and evaluating the trace, we obtain the relation

$$\mathrm{tr}(e_i v_1 \alpha_1^t c_1) = -\sum_{j=2}^{n} \mathrm{tr}(e_i v_j \alpha_j^t c_j)$$

$$= -\sum_{j=2}^{n} \alpha_j^t \operatorname{tr}(e_i v_j c_j), \quad t = 0, 1, \ldots, n-k-1, \quad (10)$$

where the second equality follows since $\alpha_j \in F_1$ for all $2 \leqslant j \leqslant n$. Therefore, the elements $\{\operatorname{tr}(e_i v_j c_j) : 2 \leqslant j \leqslant n\}$ suffice to compute $\{\operatorname{tr}(e_i v_1 \alpha_1^t c_1) : 0 \leqslant t \leqslant n-k-1\}$. Since $s = d - 2e - k + 1 \leqslant d - k + 1 \leqslant n - k$, we can calculate $\{\operatorname{tr}(e_i v_1 \alpha_1^t c_1) : 0 \leqslant t \leqslant s-1\}$ from $\{\operatorname{tr}(e_i v_j c_j) : 2 \leqslant j \leqslant n\}$. Thus knowing the values of $\{\operatorname{tr}(e_i v_j c_j) : 2 \leqslant j \leqslant n, 0 \leqslant i \leqslant p_1 - 1\}$ suffices to find the set of elements

$$\{\operatorname{tr}(e_i v_1 \alpha_1^t c_1) : 0 \leqslant t \leqslant s-1, 0 \leqslant i \leqslant p_1 - 1\}. \quad (11)$$

Since the set (9) forms a basis of $\mathbb{K}$ over F_1, the set $\{e_i v_1 \alpha^t : 0 \leqslant i \leqslant p_1 - 1, 0 \leqslant t \leqslant s-1\}$ also forms a basis of $\mathbb{K}$ over F_1, and therefore we can recover c_1 from (11). In conclusion, to recover c_1, it suffices to know the set of elements $\{\operatorname{tr}(e_i v_j c_j) : 2 \leqslant j \leqslant n, 0 \leqslant i \leqslant p_1 - 1\}$.

B. The repair scheme

For $j = 2, 3, \ldots, n$ define the vector $r_j := (\operatorname{tr}(e_i v_j c_j), i = 0, \ldots, p_1 - 1)$. In this section we design invertible linear transformations M_j that send these vectors to a set of vectors z_j that support error correction. The following proposition underlies our repair scheme.

Proposition 4. *Consider the set of vectors $z_j = (z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1}), j = 2, 3, \ldots, n$ defined by*

$$z_j^T = M_j r_j^T, \quad (12)$$

where $M_2, \ldots, M_n$ are invertible matrices of order p_1. Suppose that for every $i = 0, 1, \ldots, p_1 - 1$, the vector $(z_{2,i}, z_{3,i}, \ldots, z_{n,i})$ is contained in an MDS code of length $n-1$ and dimension $s+k-1$. Then there is a repair scheme of the code $\mathcal{C}$ that supports recovery of the node c_1 with optimal error correction capability.

Note that, by the closing remark in Sec. II-B, it suffices to assume that $s < n - k$.

Proof. If $(z_2, z_3, \ldots, z_n)$ is a codeword in an MDS array code of length $n-1$ and dimension $s+k-1$, then the punctured codeword $(z_j : j \in \mathcal{R})$ is contained in an MDS array code of length $d = |\mathcal{R}|$ and dimension $s + k - 1 = d - 2e$, and such the code can correct any e errors.

To repair the failed node c_1, we download p_1-dimensional vectors $\hat{r}_j, j \in \mathcal{R}$, where $\mathcal{R} \subset [n] \backslash \{1\}, |\mathcal{R}| = d$ is a set of d helper nodes. For all but e or fewer values of j, we have $\hat{r}_j = r_j$. The repair scheme consists of the following steps:

(i) Find the vectors $\hat{z}_j^T = M_j \hat{r}_j^T, j \in \mathcal{R}$,

(ii) Find the vectors $z_j, j \in \mathcal{R}$ using the error correction procedures of the underlying MDS codes,

(iii) For every $i = 0, \ldots, p_1 - 1$ use the d-subset $\{z_{j,i}, j \in \mathcal{R}\}$ to recover the codeword $(z_{2,i}, z_{3,i}, \ldots, z_{n,i})$,

(iv) Find the vectors $r_j^T = M_j^{-1} z_j^T, j = 2, \ldots, n-1$ and finally recover c_1.

Step (ii) is justified by the fact that, by assumption, at most e of the elements $\hat{z}_j$ are incorrect. In step (iii) we rely on the fact that d symbols of the MDS codeword suffice to recover the remaining $n - 1 - d$ symbols, and in step (iv) we use invertibility of the matrices M_j and recover c_1 using (10), (11).

The total number of downloaded symbols of F_1 equals $p_1 d$, and it is easy to verify that the repair bandwidth of our scheme meets the bound (3) with equality. □

Why do we need the matrices M_j and why were they not involved in the example in Sec. II-B? The answer is related to the fact that we need to remove the failed node from consideration and obtain a codeword of the MDS code that contains all the other nodes. In the example the degree of the minimal polynomial of α_1, denoted $f(x)$, is $s < n - k$, so the evaluations of $x^t f$ are dual codewords (see (6) in Prop. 2). This implies that the downloaded symbols form a codeword in an MDS code over F which supports error correction. Importantly, this codeword does not involve the erased coordinate.

Switching to the RS codes of [21] considered here, the element α_1 is of degree p_1 over the repair field $F(\alpha_2, \ldots, \alpha_n)$, and generally $p_1 > n - k - 1$, so the minimal polynomial of α_1 is not a dual codeword. This requires us to modify the above idea. In general terms, we will find suitable elements of the set E_1 such that Eq. (10) yields linear relations between the entries of the form $\operatorname{tr}(e_i v_j c_j)$. The coefficients of these relations form the rows of the matrix M_j.

C. The matrices M_j

In this section we will construct the matrices M_j and the vector z_j, and also prove the full rank condition. Rather than writing the expressions at this point in the text, We proceed in stages, by deriving p_1 linear relations involving components of the vectors on both sides of (12). (the notation is rather complicated and would not be intuitive; if desired, the reader may nevertheless consult Sec. III-D, particularly, Eq.(27)).

1) The first $p_1 - s - 1$ relations:

Proposition 5. *For all $0 \leqslant u \leqslant s-1$ and $0 \leqslant q \leqslant \frac{p_1 - 1}{s} - 2$, the vector*

$$\left(\alpha_j^s \operatorname{tr}(\beta^u \alpha_1^{u+qs} v_j c_j) - \operatorname{tr}(\beta^u \alpha_1^{u+(q+1)s} v_j c_j), j = 2, \ldots, n\right) \quad (13)$$

is a codeword in a GRS code of length $n-1$ and dimension $s+k-1$.

Proof. Let us write (10) for e_i of the form $e_i = \beta^u \alpha_1^{u+qs}$:

$$\operatorname{tr}(\beta^u \alpha_1^{u+qs+t} v_1 c_1) = -\sum_{j=2}^{n} \alpha_j^t \operatorname{tr}(\beta^u \alpha_1^{u+qs} v_j c_j), \quad t = 0, 1, \ldots, n-k-1$$

(see also (8)). Writing this as

$$\operatorname{tr}(\beta^u \alpha_1^{u+(q+1)s+t-s} v_1 c_1) = -\sum_{j=2}^{n} \alpha_j^{s+t-s} \operatorname{tr}(\beta^u \alpha_1^{u+qs} v_j c_j), \quad t = 0, 1, \ldots, n-k-1$$

and performing the change of variable $(t-s)\mapsto t$, we obtain the relation

$$\operatorname{tr}(\beta^u\alpha_1^{u+(q+1)s+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{s+t}\operatorname{tr}(\beta^u\alpha_1^{u+qs}v_jc_j), \tag{14}$$
$$t=-s,-s+1,\dots,-s+n-k-1.$$

On the other hand, substituting $e_i=\beta^u\alpha_1^{u+(q+1)s}$ into (10), we obtain

$$\operatorname{tr}(\beta^u\alpha_1^{u+(q+1)s+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{t}\operatorname{tr}(\beta^u\alpha_1^{u+(q+1)s}v_jc_j), \tag{15}$$
$$t=0,1,\dots,n-k-1.$$

Note that the left-hand sides of (14) and (15) conicide for $t=0,1,\dots,n-k-s-1$, and thus so do the right-hand sides. We obtain

$$\sum_{j=2}^{n}\alpha_j^{s+t}\operatorname{tr}(\beta^u\alpha_1^{u+qs}v_jc_j)=\sum_{j=2}^{n}\alpha_j^{t}\operatorname{tr}(\beta^u\alpha_1^{u+(q+1)s}v_jc_j)$$

or

$$\sum_{j=2}^{n}\alpha_j^{t}\big(\alpha_j^{s}\operatorname{tr}(\beta^u\alpha_1^{u+qs}v_jc_j)-\operatorname{tr}(\beta^u\alpha_1^{u+(q+1)s}v_jc_j)\big)=0,$$

for $t=0,1,\dots,n-k-s-1$. On account of Proposition 1 this implies the claim about the GRS code; moreover, since there are $n-k-s$ independent parity-check equations, the dimension of this code is $(n-1)-(n-k-s)=s+k-1$. □

We note that the components of the vector (13) are formed as linear combinations of the elements $\operatorname{tr}(e_iv_jc_j)$, and so this gives us p_1-s-1 vectors z_j.

2) One more relation:

Proposition 6. *The vector*

$$\Big(\sum_{u=0}^{s-1}\alpha_j^{s-u}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j)-\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big),\quad j=2,\dots,n\Big) \tag{16}$$

is a codeword in a GRS code of length $n-1$ and dimension $s+k-1$.

Proof. Going back to (10), take $e_i=\beta^u\alpha_1^{u+p_1-s-1}$ for $u=0,1,\dots,s-1$. We obtain the relation

$$\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{t}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$t=0,1,\dots,n-k-1.$$

Changing the variable $(t+u-s)\mapsto t$ in the above equation, we obtain that for every $u=0,1,\dots,s-1$,

$$\operatorname{tr}(\beta^u\alpha_1^{p_1-1+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j), \tag{17}$$
$$t=u-s,u-s+1,\dots,u-s+n-k-1.$$

Since

$$\bigcap_{u=0}^{s-1}\{u-s,u-s+1,\dots,u-s+n-k-1\} = \{-1,0,1,\dots,n-k-s-1\}, \tag{18}$$

we have

$$\operatorname{tr}(\beta^u\alpha_1^{p_1-1+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$-1\leqslant t\leqslant n-k-s-1,\quad 0\leqslant u\leqslant s-1.$$

Taking the cue from (18), let us sum these equations on $u=0,1,\dots,s-1$, and we obtain

$$\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1+t}v_1c_1\Big) = -\sum_{j=2}^{n}\sum_{u=0}^{s-1}\alpha_j^{t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j), \tag{19}$$
$$-1\leqslant t\leqslant n-k-s-1.$$

Turning to (8) again, let us substitute the element $\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}$ into (10):

$$\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1+t}v_1c_1\Big) = -\sum_{j=2}^{n}\alpha_j^{t}\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big), \tag{20}$$
$$0\leqslant t\leqslant n-k-1$$

From (19) and (20) we deduce the equality

$$\sum_{j=2}^{n}\sum_{u=0}^{s-1}\alpha_j^{t-u+s}\operatorname{tr}\Big(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j\Big) = \sum_{j=2}^{n}\alpha_j^{t}\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big),$$

or

$$\sum_{j=2}^{n}\alpha_j^{t}\Big(\sum_{u=0}^{s-1}\alpha_j^{s-u}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j) - \operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big)\Big)=0$$

for $0\leqslant t\leqslant n-k-s-1$. By Proposition 1, the proof is complete. □

3) The remaining s relations: Following the plan outlined in Sec. III-B, we have constructed p_1-s vectors z_j, listed in (13) and (16). In order to find the remaining s linear combinations of the elements $r_{i,j}$, we develop the idea used in the example in Sec. II-B.

We begin with introducing some notation. Let $f(x)$ be the minimal polynomial of α_1 over F_1. For $h=0,1,\dots,s-1$ define

$$f_h(x)=x^{p_1+h}(\operatorname{mod} f(x)), \tag{21}$$

then $\deg f_h<\deg f=p_1$ and $\alpha_1^{p_1+h}=f_h(\alpha_1)$. Let $f_{h,q}\in F_1[x], q=0,\dots,(p_1-1)/s-1$ be the (uniquely defined) polynomials such that

(i) $\deg f_{h,q} \leqslant s-1, q=0,1,\ldots,\frac{p_1-1}{s}-2$;
(ii) $\deg f_{h,(p_1-1)/s-1} \leqslant s$;
(iii)

$$f_h(x) = \sum_{q=0}^{(p_1-1)/s-1} x^{qs} f_{h,q}(x). \tag{22}$$

Proposition 7. *For every $h=0,1,\ldots,s-1$, the vector*

$$\Big(\sum_{q=0}^{(p_1-1)/s-1} \sum_{u=0}^{h} f_{h-u,q}(\alpha_j)\,\mathrm{tr}(\alpha_1^{u+qs}\beta^u v_j c_j) + \sum_{u=h+1}^{s-1} \alpha_j^{h+1-u+s}\,\mathrm{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j) - \alpha_j^{h+1}\,\mathrm{tr}\Big(\sum_{u=0}^{s-1}\beta^u \alpha_1^{p_1-1} v_j c_j\Big), j=2,3,\ldots,n\Big) \tag{23}$$

is contained in a GRS code of length $n-1$ and dimension $s+k-1$.

The proof of this proposition is rather long and technical, and is given in Appendix A.

Concluding, expressions (13), (16), and (23) yield p_1 linear combinations of the elements $(\mathrm{tr}(e_0 v_j c_j), \mathrm{tr}(e_1 v_j c_j), \ldots, \mathrm{tr}(e_{p_1-1} v_j c_j))$ for every $j \in \{2,3,\ldots,n\}$. It is these linear combinations that we denote by $z_j = (z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1})$ in (12). We have shown that for every $i \in \{0,1,\ldots,p_1-1\}$, the vector $(z_{2,i}, z_{3,i}, \ldots, z_{n,i})$ is contained in an MDS code of length $n-1$ and dimension $s+k-1$. The next subsection treats the remaining part of the assumptions in Proposition 4 above.

D. The linear transforms M_j are invertible

The object of this section is to show that the mapping

$$(\mathrm{tr}(e_0 v_j c_j), \mathrm{tr}(e_1 v_j c_j), \ldots, \mathrm{tr}(e_{p_1-1} v_j c_j)) \mapsto z_j = (z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1})$$

is invertible. In other words, we will show that $\mathrm{rank}(M_j) = p_1$ for all j. Let us first simplify the notation. Recall the set $E_1 = \{e_0, e_1, \ldots, e_{p_1-1}\}$ in (8) and let us order its elements in the order of increase of the powers of α_1 :

$$e_{u+qs} := \beta^u \alpha_1^{u+qs} \quad \text{for } u=0,1,\ldots,s-1 \text{ and } q=0,1,\ldots,\tfrac{p_1-1}{s}-1$$
$$e_{p_1-1} := \sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1}.$$

Using the notation $r_{i,j} = \mathrm{tr}(e_i v_j c_j)$ introduced above, the vectors in (13) can be written as

$$(\alpha_j^s r_{u+qs,j} - r_{u+qs+s,j}, j=2,\ldots,n)$$

for $0 \leqslant u \leqslant s-1$ and $0 \leqslant q \leqslant \frac{p_1-1}{s}-2$, or, writing $i = u+qs$, as

$$(\alpha_j^s r_{i,j} - r_{i+s,j}, j=2,\ldots,n) \tag{24}$$

for $0 \leqslant i \leqslant p_1-s-2$. Similarly, the vector in (16) can be written as

$$\Big(\sum_{u=0}^{s-1} \alpha_j^{s-u} r_{u+p_1-s-1,j} - r_{p_1-1,j}, j=2,\ldots,n\Big), \tag{25}$$

and the vectors in (23) can be written as

$$\Big(\sum_{q=0}^{(p_1-1)/s-1} \sum_{u=0}^{h} f_{h-u,q}(\alpha_j) r_{u+qs,j} + \sum_{u=h+1}^{s-1} \alpha_j^{h+1-u+s} r_{u+p_1-s-1,j} - \alpha_j^{h+1} r_{p_1-1,j}, j=2,\ldots,n\Big), \quad 0 \leqslant h \leqslant s-1. \tag{26}$$

For a fixed value of j, the entries in (24)–(26) form the vector $z_j = (z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1})$, and we list its coordinates according to the chosen order:

$$\left.\begin{aligned} z_{j,i} &:= \alpha_j^s r_{i,j} - r_{i+s,j} \quad \text{for } 0 \leqslant i \leqslant p_1-s-2, \\ z_{j,p_1-s-1} &:= \sum_{u=0}^{s-1} \alpha_j^{s-u} r_{u+p_1-s-1,j} - r_{p_1-1,j}, \\ z_{j,p_1-s+h} &:= \sum_{u=0}^{h} \sum_{q=0}^{(p_1-1)/s-1} f_{h-u,q}(\alpha_j) r_{u+qs,j} \\ &\quad + \sum_{u=h+1}^{s-1} \alpha_j^{h+1-u+s} r_{u+p_1-s-1,j} - \alpha_j^{h+1} r_{p_1-1,j} \end{aligned}\right\} \tag{27}$$

for $0 \leqslant h \leqslant s-1$. Our objective is to show that the linear mapping $(r_{0,j}, r_{1,j}, \ldots, r_{p_1-1,j}) \xrightarrow{M_j} (z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1})$ is invertible. This will follow once we show that its kernel is trivial, i.e., that if $(z_{j,0}, z_{j,1}, \ldots, z_{j,p_1-1})$ is an all-zeros vector, then so is $(r_{0,j}, r_{1,j}, \ldots, r_{p_1-1,j})$. If $z_{j,i} = \alpha_j^s r_{i,j} - r_{i+s,j} = 0$ for $0 \leqslant i \leqslant p_1-s-2$, then

$$r_{u+qs,j} = \alpha_j^s r_{u+(q-1)s,j} = \cdots = \alpha_j^{qs} r_{u,j} \quad \text{for } 0 \leqslant u \leqslant s-1 \text{ and } 1 \leqslant q \leqslant \frac{p_1-1}{s}-1. \tag{28}$$

Using (28) in the expression for $z_{j,p_1-s+h}, 0 \leqslant h \leqslant s-1$, we obtain the following s relations:

$$\begin{aligned} z_{j,p_1-s+h} &= \sum_{u=0}^{h} \sum_{q=0}^{(p_1-1)/s-1} f_{h-u,q}(\alpha_j) \alpha_j^{qs} r_{u,j} + \sum_{u=h+1}^{s-1} \alpha_j^{h+1-u+s} \alpha_j^{p_1-s-1} r_{u,j} - \alpha_j^{h+1} r_{p_1-1,j} \\ &= \sum_{u=0}^{h} f_{h-u}(\alpha_j) r_{u,j} + \sum_{u=h+1}^{s-1} \alpha_j^{p_1+h-u} r_{u,j} - \alpha_j^{h+1} r_{p_1-1,j}, \end{aligned} \tag{29}$$

where the second equality follows from (22). Using (28) in the expression for z_{j,p_1-s-1}, we obtain

$$\begin{aligned} z_{j,p_1-s-1} &= \sum_{u=0}^{s-1} \alpha_j^{s-u} \alpha_j^{p_1-s-1} r_{u,j} - r_{p_1-1,j} \\ &= \sum_{u=0}^{s-1} \alpha_j^{p_1-u-1} r_{u,j} - r_{p_1-1,j}. \end{aligned} \tag{30}$$

Since we assumed that the z-vector is zero, coordinates $z_{p_1-u}, u=s+1,s,\ldots,1$ that appear in (29), (30) are zero. Writing these conditions in matrix form using the above order,

$$\begin{bmatrix} \alpha_j^{p_1-1} & \alpha_j^{p_1-2} & \alpha_j^{p_1-3} & \dots & \alpha_j^{p_1-s} & -1 \\ f_0(\alpha_j) & \alpha_j^{p_1-1} & \alpha_j^{p_1-2} & \dots & \alpha_j^{p_1-s+1} & -\alpha_j \\ f_1(\alpha_j) & f_0(\alpha_j) & \alpha_j^{p_1-1} & \dots & \alpha_j^{p_1-s+2} & -\alpha_j^2 \\ f_2(\alpha_j) & f_1(\alpha_j) & f_0(\alpha_j) & \dots & \alpha_j^{p_1-s+3} & -\alpha_j^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f_{s-1}(\alpha_j) & f_{s-2}(\alpha_j) & f_{s-3}(\alpha_j) & \dots & f_0(\alpha_j) & -\alpha_j^s \end{bmatrix} \begin{bmatrix} r_{0,j} \\ r_{1,j} \\ r_{2,j} \\ \vdots \\ r_{s-1,j} \\ r_{p_1-1,j} \end{bmatrix} = 0, \tag{31}$$

we obtain relation (31). We aim to show that the matrix on the left-hand side is invertible.

Recall that $f(x)$ is the minimal polynomial of α_1 and from (22), $f(x) + f_0(x) = x^{p_1}$. Since $f(x)$ is irreducible over F_1 and $\alpha_j \in F_1$, we have $f(\alpha_j) \neq 0$ for all $j = 2, \dots, n$.

Multiplying the first row of the matrix in (31) by α_j and then subtracting the second row from the first row, we obtain

$$\begin{bmatrix} f(\alpha_j) & 0 & 0 & \dots & 0 & 0 \\ f_0(\alpha_j) & \alpha_j^{p_1-1} & \alpha_j^{p_1-2} & \dots & \alpha_j^{p_1-s+1} & -\alpha_j \\ f_1(\alpha_j) & f_0(\alpha_j) & \alpha_j^{p_1-1} & \dots & \alpha_j^{p_1-s+2} & -\alpha_j^2 \\ f_2(\alpha_j) & f_1(\alpha_j) & f_0(\alpha_j) & \dots & \alpha_j^{p_1-s+3} & -\alpha_j^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f_{s-1}(\alpha_j) & f_{s-2}(\alpha_j) & f_{s-3}(\alpha_j) & \dots & f_0(\alpha_j) & -\alpha_j^s \end{bmatrix}.$$

Since $f(\alpha_j) \neq 0$, we can use elementary row operations to erase the first column, obtaining

$$\begin{bmatrix} f(\alpha_j) & 0 & 0 & \dots & 0 & 0 \\ 0 & \alpha_j^{p_1-1} & \alpha_j^{p_1-2} & \dots & \alpha_j^{p_1-s+1} & -\alpha_j \\ 0 & f_0(\alpha_j) & \alpha_j^{p_1-1} & \dots & \alpha_j^{p_1-s+2} & -\alpha_j^2 \\ 0 & f_1(\alpha_j) & f_0(\alpha_j) & \dots & \alpha_j^{p_1-s+3} & -\alpha_j^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & f_{s-2}(\alpha_j) & f_{s-3}(\alpha_j) & \dots & f_0(\alpha_j) & -\alpha_j^s \end{bmatrix}.$$

Proceeding analogously, let us multiply the second row of this matrix by α_j and then subtract the third row from the second one to obtain

$$\begin{bmatrix} f(\alpha_j) & 0 & 0 & \dots & 0 & 0 \\ 0 & f(\alpha_j) & 0 & \dots & 0 & 0 \\ 0 & f_0(\alpha_j) & \alpha_j^{p_1-1} & \dots & \alpha_j^{p_1-s+2} & -\alpha_j^2 \\ 0 & f_1(\alpha_j) & f_0(\alpha_j) & \dots & \alpha_j^{p_1-s+3} & -\alpha_j^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & f_{s-2}(\alpha_j) & f_{s-3}(\alpha_j) & \dots & f_0(\alpha_j) & -\alpha_j^s \end{bmatrix}.$$

As above, we can eliminate all the nonzeros in the second column except for $f(\alpha_j)$, and so on. In the end we obtain the matrix $\mathrm{diag}(f(\alpha_j), \dots, f(\alpha_j), -\alpha_j^s)$ with nonzero diagonal. This proves that the matrix in (31) is invertible. Therefore, $r_{0,j} = r_{1,j} = \dots = r_{s-1,j} = r_{p_1-1,j} = 0$. Combining this with (28), we conclude that $r_{i,j} = 0$ for all $0 \leqslant i \leqslant p_1 - 1$. This proves that the matrices $M_j, j = 2, \dots, n$ in (12) are invertible, providing the last missing element to the justification of the repair scheme with optimal error correction.

IV. A family of optimal-access RS codes

In this section, we construct a new family of RS codes that is similar to the construction in [21] but affords repair with optimal access.

The input-output cost of node repair for the RS codes of [21] was analyzed in [25] for $d = n - 1$. According to (2), in this case the minimum access cost per helper node equals $\frac{l}{n-k}$. The authors of [25] showed that it is possible to adjust the repair scheme so that the access cost is $(1 + \frac{n-k-1}{p_i})\frac{l}{n-k}$, i.e., at most twice the optimal value. However, more is true: namely, it turns out that any *fixed* node in the construction of [21] (Def. 1) can be repaired with optimal access. This observation, which is the starting point of the new construction, is based on the fact that it is possible to construct a basis of the field $\mathbb{K}$ over the base field that reduces the access cost of the repair of the chosen node. If the option of choosing the basis for each erased node were available, we could use the arguments in Sec. II-C to perform repair with optimal access. The difficulty arises because this would entail rewriting the storage contents, which should be avoided. To address this issue, we construct the code over a field that contains n elements β_i instead of a single element β, and this supports efficient repair of any single failed node. This idea is developed below.

A. New construction

Consider the following sequence of algebraic extensions of $\mathbb{F}_p$: let $K_0 = \mathbb{F}_p$ and for $i = 1, \dots, n$ let

$$F_i = K_{i-1}(\alpha_i), K_i = F_i(\beta_i), \tag{32}$$

where α_i is an algebraic element of degree p_i over $\mathbb{F}_p$ and β_i is an element of degree $s = d - k + 1$ over F_i. In the end we obtain the field

$$\mathbb{K} := K_n = \mathbb{F}_p(\alpha_1, \dots, \alpha_n, \beta_1, \dots \beta_n). \tag{33}$$

We still assume that $p_1, \dots, p_n$ are distinct primes satisfying the condition $p_i \equiv 1 \bmod s$ for all $i = 1, \dots, n$. Consider the code $\mathcal{C} := RS_{\mathbb{K}}(n, k, \Omega)$, where as before, the set of evaluation points is given by $\Omega = \{\alpha_1, \dots, \alpha_n\}$. We will show that the code $\mathcal{C}$ affords optimal-access repair.

The repair scheme follows the general approach of [16] and its implementation in [21]. Let $c = (c_1, \dots, c_n) \in \mathcal{C}$ be a codeword. Suppose that the node i has failed (coordinate c_i is erased), and we would like to repair it from a set of helper nodes $\mathcal{R} \subseteq \{1, \dots, n\} \backslash \{i\}$ with $|\mathcal{R}| = d$. Let

$$h(x) = \prod_{j \in \{1,\dots,n\} \backslash (\mathcal{R} \cup \{i\})} (x - \alpha_j).$$

Clearly, we have $\deg(x^t h(x)) < n - k$ for $t = 0, \dots, s - 1$. Therefore, for some nonzero vector $v = (v_1, \dots, v_n)$, we have $(v_1 \alpha_1^t h(\alpha_1), \dots, v_n \alpha_n^t h(\alpha_n)) \in \mathcal{C}^\perp$ for $t = 0, \dots, s-1$, where $\mathcal{C}^\perp = GRS_{\mathbb{K}}(n, k, v, \Omega)$. In other words, we have

$$v_i \alpha_i^t h(\alpha_i) c_i = -\sum_{\substack{j=1 \\ j \neq i}}^{n} v_j \alpha_j^t h(\alpha_j) c_j, \ t = 0, \dots, s-1. \tag{34}$$

The repair scheme in [21] as well as in this paper relies on this set of s dual codewords to recover the value of c_i.

Remark 1. The dual codewords $x^t h(x)$ have zero values in the complement of the set $\hat{\mathcal{R}} := \mathcal{R} \cup \{i\}$. In other words, they are contained in the shortened code $(\mathcal{C}^\perp)^{\hat{\mathcal{R}}}$ of the dual code. Thinking dually, we can start with the code $\mathcal{C}^\perp$ and construct a repair scheme for its coordinates based on the punctured code $\mathcal{C}_{\hat{\mathcal{R}}}$ (coordinate projection of $\mathcal{C}$ on $\hat{\mathcal{R}}$). This approach is equivalent to the scheme used in [21] and in this paper because $((\mathcal{C}^\perp)^{\hat{\mathcal{R}}})^\perp \cong C_{\hat{\mathcal{R}}}$.

Let us establish a few simple properties of the tower of fields defined above in (32), (33).

Lemma 8. *The extension degrees in the field tower* $\mathbb{F}_p = K_0 \subset \cdots \subset K_i \subset \cdots \subset K_n = \mathbb{K}$ *are as follows:*

$$[K_i : \mathbb{F}_p] = s^i \prod_{j=1}^{i} p_j, i = 1, \dots, n$$

$$[\mathbb{K} : \mathbb{F}_p] = l := s^n \prod_{i=1}^{n} p_i.$$

Proof. The proof is obvious from the definition: for each i we adjoin two elements α_i, β_i to K_{i-1}, and their degrees over K_{i-1} are coprime, so they contribute sp_i to the result. □

We will use an explicit form of the basis of $\mathbb{K}$ over $\mathbb{F}_p$. For $m = 0, \dots, l-1$, let us write

$$m = (m_n, m_{n-1}, \dots, m_1, \bar{m}_n, \bar{m}_{n-1}, \dots, \bar{m}_1) \tag{35}$$

where $m_i = 0, \dots, p_i - 1$ and $\bar{m}_i = 0, \dots, s-1$ for $i = 1, \dots, n$.

Lemma 9. *Let*

$$A = \{a_m := \prod_{i=1}^{n} \alpha_i^{m_i} \prod_{j=1}^{n} \beta_j^{\bar{m}_j} \mid m_i = 0, \dots, p_i - 1, \\ \bar{m}_j = 0, \dots, s-1; m = 0, 1, \dots, l-1\}.$$

Then A is a basis for $\mathbb{K}$ over $\mathbb{F}_p$.

Proof. By co-primality, for $i = 1, \dots, n$ we have $\deg_{K_{i-1}}(\alpha_i) = p_i$, and by construction, we have $\deg_{F_i}(\beta_i) = s$. Thus, the elements $a_m, m = 0, \dots, l-1$ are linearly independent over $\mathbb{F}_p$. □

Lemma 10. *For $m = 0, \dots, l-1$ let $\mathcal{J} = \{j \in [n] : (\bar{m}_j, m_j) = (s-1, p_j - 1)\}$ and let*

$$b_m = \prod_{i=1}^{n} \alpha_i^{m_i} \cdot \prod_{j \in \mathcal{J}} \Big(\sum_{u=0}^{s-1} \beta_j^u \Big) \cdot \prod_{j \notin \mathcal{J}} \beta_j^{\bar{m}_j}.$$

Then the set $B := \{b_m \mid m = 0, \dots, l-1\}$ is a basis of $\mathbb{K}$ over $\mathbb{F}_p$.

Furthermore, for $i = 1, \dots, n$, let $A_i = \{a_m \in A \mid (m_i, \bar{m}_i) = (0,0)\}$ and $B_i = \{b_m \in B \mid (m_i, \bar{m}_i) = (0,0)\}$, then

$$\mathrm{Span}_{\mathbb{F}_p} A_i = \mathrm{Span}_{\mathbb{F}_p} B_i.$$

Proof. Since $|B| = l$, to prove that B is a basis it suffices to show that the elements a_m can be expressed as linear combinations of the elements in B. Let $\mathcal{J} \subset [n]$ and let $A(\mathcal{J}) = \{a_m \in A : (\bar{m}_j, m_j) = (s-1, p_j - 1), j \in \mathcal{J}; (\bar{m}_j, m_j) \neq (s-1, p_j - 1), j \notin \mathcal{J}\}$. We argue by induction on $|\mathcal{J}|$. If m is such that $\mathcal{J} = \varnothing$, then $a_m \in B$, and there is nothing to prove. Now assume that for all $\mathcal{J} \subset [n], |\mathcal{J}| \leqslant J-1$ the elements a_m are linearly generated by the elements in B, and let m be such that $|\mathcal{J}| = J$. We have

$$a_m = \prod_{i=1}^{n} \alpha_i^{m_i} \prod_{j \notin \mathcal{J}} \beta_j^{\bar{m}_j} \prod_{j \in \mathcal{J}} \beta_j^{s-1}$$

and

$$b_m = \prod_{i=1}^{n} \alpha_i^{m_i} \prod_{j \notin \mathcal{J}} \beta_j^{\bar{m}_j} \prod_{j \in \mathcal{J}} \sum_{u=0}^{s-1} \beta_j^u \\ = \prod_{i=1}^{n} \alpha_i^{m_i} \Big(\prod_{j \notin \mathcal{J}} \beta_j^{\bar{m}_j} \Big) \Big(\sum_{t_1, \dots, t_J = 0}^{s-1} \prod_{u=1}^{J} \beta_{j_u}^{t_u} \Big).$$

Multiplying out the sums on right-hand side, we note that the term with all $t_i = s-1$ equals a_m, while the remaining terms contain fewer than J factors of the form $\alpha_{j_u}^{p_{j_u}-1} \beta_{j_u}^{s-1}$. Each of such terms is contained in some $A(\mathcal{J})$ with $|\mathcal{J}| \leqslant J-1$, and is linearly generated by the elements b_m by the induction hypothesis. This implies that a_m is also expressible as a linear combination of the elements in B.

To prove the second claim, note that $\mathrm{Span}_{\mathbb{F}_p} A_i \supseteq \mathrm{Span}_{\mathbb{F}_p} B_i$. Therefore, to show that $\mathrm{Span}_{\mathbb{F}_p} A_i = \mathrm{Span}_{\mathbb{F}_p} B_i$, it suffices to show that for any $a = 0, \dots, n-1$ and any $\mathcal{J} \subseteq \{1, \dots, n\} \backslash \{i\}$, the set $A_i(\mathcal{J})$ can be generated linearly by the set B_i. This proof amounts essentially to the same calculation as above, and will be omitted. □

The role of the basis (b_m) is to eliminate as many terms on the right-hand side of (34) as possible. To repair the node c_i we use the dual basis (b_m^*) of (b_m), writing

$$c_i = v_i^{-1} \sum_{m=0}^{l-1} c_{i,m} b_m^*. \tag{36}$$

Below $\mathrm{tr} = \mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}$ denotes the absolute trace.

Lemmas 8 and 9 immediately imply the following.

Proposition 11. *For $i = 1, \dots, n$, there exists vector space S_i over K_{i-1} such that $\dim_{K_{i-1}} S_i = p_i$ and $S_i + S_i\alpha_i + \cdots + S_i\alpha_i^{s-1} = K_i$. Furthermore, a basis for S_i over K_{i-1} is given by*

$$E_i := \{\beta_i^u \alpha_i^{u+qs} \mid u = 0, \dots, s-1; q = 0, \dots, \tfrac{p_i - 1}{s} - 1\} \\ \bigcup \Big\{ \alpha_i^{p_i - 1} \sum_{u=0}^{s-1} \beta_i^u \Big\}.$$

We continue with the description of the repair scheme where we left in (34). As a remark, below we write the scheme over $\mathbb{F}_p$ rather than over its extensions (the latter approach was chosen in [21]). Multiplying both sides of (34) by $\prod_{i'=1}^{n} e_{i'} \prod_{j' \neq i}^{n} \alpha_{j'}^{t_{j'}}$, where $e_{i'} \in E_{i'}$ and $t_{j'} = 0, \dots, s-1$, and evaluating the trace, we obtain

$$\mathrm{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j' \neq i}^{n} \alpha_{j'}^{t_{j'}} v_i \alpha_i^t h(\alpha_i) c_i \Big)$$

$$= -\operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \sum_{j\neq i}^{n} v_j \alpha_j^t h(\alpha_j) c_j\Big)$$
$$= -\sum_{j\neq i}^{n} \operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} v_j \alpha_j^t h(\alpha_j) c_j\Big)$$
$$= -\sum_{j\in\mathcal{R}} \operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} v_j \alpha_j^t h(\alpha_j) c_j\Big). \qquad (37)$$

On account of Proposition 11 and the fact that $v_i h(\alpha_i) \neq 0$, the set

$$\Big\{\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} v_i \alpha_i^t h(\alpha_i)\Big\}, \qquad (38)$$

where $e_{i'} \in E_{i'}, i' \in [n]; t = 0, \dots, s-1; t_{j'} = 0, \dots, s-1, j' \in [n]\backslash\{i\}$, is a basis of $\mathbb{K}$ over $\mathbb{F}_p$. Therefore, we can recover c_i once we know the right-hand side of (37).

For $j \in \mathcal{R}$, from (36) we have

$$\operatorname{tr}(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} v_j \alpha_j^t h(\alpha_j) c_j)$$
$$= \operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j) \sum_{m=0}^{l-1} c_{j,m} b_m^*\Big)$$
$$= \sum_{m=0}^{l-1} \operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j) b_m^*\Big) c_{j,m}. \qquad (39)$$

From (39), we see that in order to recover c_i we need to access only those symbols $c_{j,m}$ for which

$$\operatorname{tr}(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j) b_m^*) \neq 0.$$

Now, the element $\prod_{i'\neq i}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j)$ does not include α_i, β_i, and thus it can be written as an $\mathbb{F}_p$-linear combination of the elements in the set A_i. By Lemma 10, it can further be expressed as an $\mathbb{F}_p$-linear combination of the elements in the set B_i. Therefore, the elements $\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j)$ for $e_{i'} \in E_{i'}$ and $t_{j'} = 0, \dots, s-1$ can be linearly generated over $\mathbb{F}_p$ by the set

$$\bigcup_{e_i\in E_i} e_i B_i \subseteq B.$$

Since B and B^* are dual bases,

$$\operatorname{tr}\Big(\prod_{i'=1}^{n} e_{i'} \prod_{j'\neq i}^{n} \alpha_{j'}^{t_{j'}} \alpha_j^t h(\alpha_j) b_m^*\Big) \neq 0$$

if and only if $b_m \in \bigcup_{e_i\in E_i} e_i B_i$. It follows that to calculate the left hand side of (37), we need to access $\sum_{e_i\in E_i} |e_i B_i| = p_i l/s p_i = l/s$ symbols on each helper node $j \in \mathcal{R}$, which implies that the node c_i affords optimal-access repair.

In conclusion, we note that the repair scheme of each of the nodes i relies on its own element β_i. Looking back at the construction of [21], Sec. III above, it contains one such β. Thus, these codes can be furnished with a repair scheme that has the optimal access property for any one (fixed) node in the encoding; see also the discussion at the end of Sec. II-C.

B. Error correction with optimal access

In this section we present a repair scheme of the RS codes defined in the beginning of Sec. IV-A that supports both the optimal access and optimal error correction properties. The scheme relies on a combination of ideas of Sections IV-A and III. A full presentation of the proof would require us to repeat the arguments in Sec. III-C; we shall instead confine ourselves to pointing to the similarity of the starting point and argue that once this is recognized, the remaining part is reproduced directly following the proof in Sec. III-C.

Let us modify the construction of RS codes of Sec. IV-A as follows. Let us assume that the number of helper nodes is d. We will construct our RS code over the symbol field $\mathbb{K} = \mathbb{F}_p(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n)$ (33), where as before, $\deg_{K_{i-1}}(\alpha_i) = p_i$ but $\deg_{F_i}(\beta_i) = s := d-2e-k+1$. Define the code $\mathcal{C} := RS_{\mathbb{K}}(n, k, \Omega)$, where $\Omega = \{\alpha_1, \dots, \alpha_n\}$.

Without loss of generality suppose that the failed node is the first one and let $\mathcal{R} \subseteq \{2, 3, \dots, n\}$ with $|\mathcal{R}| = d, 2e + k \leqslant d \leqslant n-1$ be the subset of helper nodes. Consider a basis of $\mathbb{K}$ over $\mathbb{F}_p$ given by $\bigcup_{t=0}^{s-1} \alpha_1^t \Lambda$, where

$$\Lambda = \Big\{\prod_{i=1}^{n} e_i \prod_{j=2}^{n} \alpha_j^{t_j} \mid e_i \in E_i, i \in [n];$$
$$t_j = 0, \dots, s-1, j \in [n]\backslash\{1\}\Big\}.$$

That this is a basis is apparent from (38).

Next, note that $(v_1\alpha_1^t, \dots, v_n\alpha_n^t) \in \mathcal{C}^\perp$ for some $v = (v_1, \dots, v_n) \in (\mathbb{K}^*)^n$ and for $t = 0, \dots, n-k-1$. Therefore, for every $\lambda \in \Lambda$ we have

$$\lambda v_1 \alpha_1^t c_1 = -\sum_{j=2}^{n} \lambda v_j \alpha_j^t c_j, \quad t = 0, \dots, n-k-1.$$

Let $G_1 := \mathbb{F}_p(\alpha_2, \alpha_3, \dots, \alpha_n)$. Evaluating the trace $\operatorname{tr}_{\mathbb{K}/G_1}$ on both sides of the above equation, we obtain

$$\operatorname{tr}_{\mathbb{K}/G_1}(\lambda v_1 \alpha_1^t c_1) = -\sum_{j=2}^{n} \alpha_j^t \operatorname{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j), \qquad (40)$$
$$t = 0, \dots, n-k-1.$$

The repair scheme for the code $\mathcal{C}$ is based on (40) in exactly the same way as the repair scheme of Proposition 4 is based on (10). Namely, suppose that there are invertible linear transformations that map the vectors $(\operatorname{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j), \lambda \in \Lambda), j = 2, 3, \dots, n$ to codevectors in an MDS code of length $n-1$ and dimension $s+k-1$. Then it is possible to correct e errors in the information collected from the helper nodes upon puncturing of this code to any d coordinates in the same way as is done in Proposition 4. Thus, the main step is to prove existence of such transformations. Here we observe that the terms involved in (40) are formed of e_1 times the remaining factors in λ. The element e_1 plays the same role as e_i in (10), and the multiplier in front of it in λ does not affect the proof. For this reason, the required proof closely follows the proof in Sec. III-C, and we do not repeat it here.

Thus, the vectors $(\operatorname{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j), \lambda \in \Lambda), j \in \mathcal{R}$ suffice to recover the value of the failed node. We argue that these values can be calculated by accessing the smallest possible number of symbols on the helper nodes, and thus support the claim

of optimal access. Let $B = (b_m)$ be the basis of $\mathbb{K}$ over $\mathbb{F}_p$ defined in Lemma 10, let $B^* = (b^*_m)$ be its dual basis, and let $B_1 = \{b_m \in b | (m_1, \bar{m}_1) = (0,0)\}$. From (36), for every $\lambda \in \Lambda$ and all $j = 2, 3, \ldots, n$ we have the equality

$$\mathrm{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j) = \mathrm{tr}_{\mathbb{K}/G_1}\Big(\lambda \sum_{m=0}^{l-1} b^*_m\Big) c_{i,m}.$$

Let Γ be a basis for G_1 over $\mathbb{F}_p$. Then from the above equation, for every $\gamma \in \Gamma$ we have

$$\mathrm{tr}_{G_1/\mathbb{F}_p}(\gamma\, \mathrm{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j)) = \mathrm{tr}_{G_1/\mathbb{F}_p}\Big(\gamma\, \mathrm{tr}_{\mathbb{K}/G_1}\Big(\lambda \sum_{m=0}^{l-1} b^*_m\Big)\Big) c_{i,m}.$$

Since $\gamma \in G_1$ and $\mathrm{tr}_{G_1/\mathbb{F}_p} \circ \mathrm{tr}_{\mathbb{K}/G_1} = \mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}$, it follows that

$$\mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}(\gamma\lambda v_j c_j) = \mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}\Big(\gamma\lambda \sum_{m=0}^{l-1} b^*_m\Big) c_{i,m}. \tag{41}$$

Note that the elements $\gamma\lambda = \gamma \prod_{i=1}^{n} e_i \prod_{j=2}^{n} \alpha_j^{t_j}$ can be written as $\mathbb{F}_p$-linear combinations of the elements in the set $\bigcup_{e_1 \in E_1} e_1 B_1 \subseteq B$. By the duality of B and B^*, the number of symbols that each helper node accesses to calculate the left hand side of (41) equals $|\bigcup_{e_1 \in E_1} e_1 B_1| = l/s$, which, as remarked in the introduction, is the smallest possible number of symbols. Further, since Γ is a basis of G_1 over $\mathbb{F}_p$, we can recover $\mathrm{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j)$ from the set $\{\mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}(\gamma\lambda v_j c_j) \mid \gamma \in \Gamma\}$.

Finally, evaluating the trace $\mathrm{tr}_{G_1/\mathbb{F}_p}$ on both sides of (40), we obtain

$$\mathrm{tr}_{\mathbb{K}/\mathbb{F}_p}(\lambda v_1 \alpha_1^t c_1) = -\sum_{j=2}^{n} \mathrm{tr}_{G_1/\mathbb{F}_p}(\alpha_j^t\, \mathrm{tr}_{\mathbb{K}/G_1}(\lambda v_j c_j)), \quad t = 0, \ldots, s-1. \tag{42}$$

Since the set $\{\lambda v_1 \alpha_1^t \mid \lambda \in \Lambda; t = 0, \ldots, s-1\}$ forms a basis for $\mathbb{K}$ over $\mathbb{F}_p$, we conclude from (42) that we can perform optimal error correction for the code $\mathcal{C}$ with optimal access. As a final remark, the locations of the entries accessed on each helper node depend only on the index of the failed node, and are independent of the index of the helpers.

V. Every scalar MSR code affords optimal-access repair

This section is devoted to establishing the claim in the title. We begin with a discussion of repair schemes with a particular property of having constant repair subspaces and use it to show that every MSR code with this property can be repaired with optimal access. In the last part of the section we remove this assumption, establishing the general result, which is stated as follows.

Theorem 12. *Let $\mathcal{C}$ be an (n,k) scalar MDS code over a finite field K of length n such that any single failed node can be optimally repaired from any subset of d helper nodes, $k+1 \leqslant d \leqslant n-1$ with optimal repair bandwidth. Then there exists an explicit procedure that supports optimal-access repair of any single node from any subset of d helpers, $k+1 \leqslant d \leqslant n-1$.*

A. Constant repair subspaces

Observe that the repair scheme presented above in Sec. IV has the property that for a given index of the failed node i, the procedure for recovering the node contents does not depend on the chosen subset of d helper nodes. Indeed, to repair node i, the scheme accesses symbols $\{c_{j,m} \mid m : b_m \in \bigcup_{e_i \in E_i} e_i B_i\}$ on the node j, i.e., the symbols $c_{j,m}$ with $m = (m_i, \bar{m}_i)$ and

$$(m_i, \bar{m}_i) \in \{(u + qs, u) \mid u = 0, \ldots, s-1; \; q = 0, \ldots, (p_i - 1)/s - 1\} \cup \{(p_i - 1, s-1)\}.$$

Clearly the values of m are independent of $j \in \mathcal{R}$. This simplifies the implementation, and therefore represents a desirable property of the scheme. In this section, we generalize this observation and give conditions for it to hold.

Let $\mathcal{C}$ be an (n,k) linear scalar MDS code of length n over finite field K, and let $r = n - k$ be the number of parity nodes. Let F be a subfield of K such that $[K : F] = l$. For a subset $M \subset K$ we write $\dim_F(M)$ to refer to the dimension of the subspace spanned by the elements of M over F. The following result is a starting point of our considerations.

Theorem 13 ([16])**.** *The code $\mathcal{C}$ has an optimal linear repair scheme over F with repair degree $d = n-1$ if and only if for every $i = 1, \ldots, n$ there exist l codewords $(c^\perp_{t,1}, \ldots, c^\perp_{t,n}) \in \mathcal{C}^\perp, t = 1, \ldots, l$ such that*

$$\dim_F(c^\perp_{1,i}, \ldots, c^\perp_{l,i}) = l,$$
$$\sum_{j \neq i}^{n} \dim_F(c^\perp_{1,j}, \ldots, c^\perp_{l,j}) = \frac{(n-1)l}{r}.$$

We go on to define the main object of this section.

Definition 2. *Let $\mathcal{C}$ be a scalar MDS code that has a linear repair scheme for repair of a single node with optimal bandwidth, based on dual codewords $c^\perp_1, \ldots, c^\perp_l$. The scheme is said to have constant repair subspaces if for every $i = 1, \ldots, n$ and every $\mathcal{R} \subset [n]\backslash\{i\}, |\mathcal{R}| = d$, the information downloaded from a helper node $c_j, j \in \mathcal{R}$ to repair the failed node c_i does not depend on the index j. Namely, the subspace $\mathcal{S}^{(i)}_j := \mathrm{Span}_F(c^\perp_{1,j}, \ldots, c^\perp_{l,j}), j \in \mathcal{R}$ is independent of the index j, i.e., $\mathcal{S}^{(i)}_j = \mathcal{S}^{(i)}$ for some linear subspace $\mathcal{S}^{(i)} \subseteq K$.*

The notion of constant repair subspaces was mentioned earlier in the literature on general MSR codes, for instance, see [11].

The algorithms below in this section rely on a proposition which we cite from [21].

Proposition 14. *Let $\mathcal{C}$ be an $(n, n-r)$ MDS code and let $[n] = J \cup J^c$, where $J, |J| = r$ is the set of parity coordinates. Let $H = (h_1, \ldots, h_n)$ be a parity-check matrix of $\mathcal{C}$, where h_i denote its columns. The code $\mathcal{C}$ has an optimal linear repair scheme over F with repair degree $d = n-1$ if and only if for each $j \in J^c$ there exist r vectors $a_u \in K^{l/r}, u = 1, \ldots, r$ such that*

$$\dim_F(Ah_j) = l, \tag{43}$$
$$\dim_F(Ah_i) = \frac{l}{r}, \quad i \in \{1, \ldots, n\}\backslash\{j\}, \tag{44}$$

where $A := \mathrm{Diag}(a_1, \ldots, a_r)$ is an $l \times r$ block-diagonal matrix with blocks formed by single columns. Furthermore for every subspace $\mathcal{A}_u = \mathrm{Span}_F(a_u), u = 1, \ldots, r$ (the F-linear span of the entries of a_u) we have

$$\dim_F(\mathcal{A}_u) = \frac{l}{r}. \tag{45}$$

Remark 2. The matrix A in Proposition 14 depends on the matrix H and the choice of J, but we suppress this dependence from the notation for simplicity.

Before presenting the algorithms for finding a basis for optimal-access repair we briefly digress to state some conditions for an optimal linear repair scheme to have constant repair subspaces. First, we rephrase their definition based Proposition 14.

Definition 3. *An optimal linear repair scheme for the code $\mathcal{C}$ is said to have constant repair subspaces if for every $j = 1, \ldots, n$ there exists a vector $h \in K^r$ such that*

$$\mathrm{Span}_F(Ah_i) = \mathrm{Span}_F(Ah)$$

for every $i \in \{1, \ldots, n\}\backslash\{j\}$. Here the matrix A is as in Proposition 14, and it depends on H and the particular choice of the information coordinates.

Proposition 15. *Suppose that $\mathcal{A}_1 = \mathcal{A}_2 = \cdots = \mathcal{A}_r$ for each $i = 1, \ldots, n$, and that for every $j \in \{1, \ldots, n\}\backslash\{i\}$ there exists $v \in \{1, \ldots, r\}$ such that $h_{v,j} \in F$, then there exists an optimal linear repair scheme for the code $\mathcal{C}$ which has constant repair subspaces.*

Proof. Let $\mathcal{V}$ denote any of the (coinciding) repair subspaces. By Proposition 14, we have $\dim_F(\mathcal{V}) = l/r$. Suppose that J is the subset of parity coordinates, and the matrix H is represented in systematic form. In this case, for every $j \in J^c$, $h_{u,j} \neq 0$ for all $u = 1 \ldots, r$, and we have $\dim_F(\mathcal{V}h_{u,j}) = l/r$. Note that

$$\mathrm{Span}_F(Ah_j) = \sum_{u=1}^{r} \mathcal{A}_u h_{u,j} = \sum_{u=1}^{r} \mathcal{V}h_{u,j}, \quad j \in \{1, \ldots, n\}\backslash\{i\}, \tag{46}$$

where the sum on the right is a sum of linear spaces. By Proposition 14, we also have $l/r = \dim_F(Ah_j) = \dim_F(\sum_{u=1}^{r} \mathcal{V}h_{u,j})$. Therefore,

$$\mathcal{V}h_{1,j} = \mathcal{V}h_{2,j} = \cdots = \mathcal{V}h_{r,j}, \quad j \in J^c\backslash\{i\}. \tag{47}$$

Since for each $j \neq i$ there exists $v \in \{1, \ldots, r\}$ such that $h_{v,j} \in F$, it follows that $\mathcal{V}h_{v,j} = \mathcal{V}$. On account of (46) and (47), we have $\mathrm{Span}_F(Ah_j) = \mathcal{V} = \mathrm{Span}_F(A \cdot \mathbf{1})$ for every $j \in \{1, \ldots, n\}\backslash\{i\}$, where $\mathbf{1}$ is the all-ones column vector of length r. By Definition 3 this completes the proof. □

The assumptions of this proposition are satisfied, for instance, for the RS subfamily of [21], which therefore have constant repair subspaces (this observation was previously not stated in published literature).

Proposition 16. *If there exists an optimal linear repair scheme for the code $\mathcal{C}$ which has constant repair subspaces, then $\mathcal{A}_1 = \mathcal{A}_2 = \cdots = \mathcal{A}_r$ for every $j = 1, \ldots, n$.*

Proof. Indeed, since H_J is the identity, for $j \in J$ we have $\mathrm{Span}_F(Ah_j) = \mathcal{A}_t$ for some $t \in \{1, \ldots, r\}$. It follows that $\mathcal{A}_1 = \mathcal{A}_2 = \cdots = \mathcal{A}_r$. □

B. Optimal access for the case of constant repair subspaces

The codes constructed in Sec. IV above form essentially the only known example of RS codes that afford repair with optimal access. For instance, the optimal-repair RS codes in [21] are not known to support optimal access, and the repair scheme in [21] is far from having this property. Prior works on the problem of access cost for RS repair [23]–[25] also do not give examples of repair schemes with optimal access. In this section we show that any family of scalar MDS codes with optimal repair can be furnished with a repair scheme with optimal access, and this includes the code family in [21]. Unfortunately, our results are not explicit; rather, we present an algorithm that produces a basis for representing nodes of the codeword that supports optimal-access repair.

As in Sec. V-A, let F be a subfield of K such that $[K : F] = l$. Let $\mathcal{C}$ be an $(n, k = n - r)$ linear scalar MDS code of length n over K equipped with a repair scheme over F that attains the bound (2) for repair of a single node. Let us represent $\mathcal{C}$ in systematic form, choosing a subset $J \subseteq \{1, \ldots, n\}, |J| = r$ for the parity symbols and J^c for the data symbols. Let H be an $r \times n$ parity-check matrix for $\mathcal{C}$ such that H_J is the $r \times r$ identity matrix,

In this section we assume that there exists an optimal repair scheme over F for $\mathcal{C}$ that has constant repair subspaces, and that the repair degree is $d = n - 1$. We will lift both assumptions and show that our result holds in general in the next section. For a given $j = 1, \ldots, n$ consider the subspaces $\mathcal{A}_i, i = 1, \ldots, r$ defined in Proposition 14. Under the assumption of constant repair subspaces, they coincide, and we use the notation $\mathcal{V}_j$ to refer to any of them.

Consider the following procedure (Algorithm 1) that interatively collects vectors to form a basis of K/F that supports optimal-access repair.

Algorithm 1: Construction of an optimal basis

Input: Subspaces $\mathcal{V}_1, \ldots, \mathcal{V}_n$.
Output: A basis B for K over F.

for $j \leftarrow 1$ **to** n **do**
  $B_j \leftarrow \varnothing$;
  $\mathcal{B}_j \leftarrow \{0\}$;
for $i \leftarrow 0$ **to** $n - 1$ **do**
  foreach $I \subseteq \{1, \ldots, n\}$ *such that* $|I| = i$ **do**
    $\bar{I} \leftarrow \{1, \ldots, n\}\backslash I$;
    $\mathcal{U}_I \leftarrow \bigcap_{j \in \bar{I}} \mathcal{V}_j$;
    for $j \leftarrow 1$ **to** n **do**
      if $j \in \bar{I}$ **then**
        $\mathcal{B}_j \leftarrow \mathcal{B}_j + \mathcal{U}_I$;
        Extend the set B_j to a basis of $\mathcal{B}_j$ over F;
$\bar{B} \leftarrow \bigcup_{j=1}^{n} B_j$;
Extend the set $\bar{B}$ to a basis B of K over F;

Proposition 17. *Upon completion of Algorithm 7 we have $\mathcal{B}_j = \mathcal{V}_j$ for $j = 1, \ldots, n$, and thus B_j is a basis for $\mathcal{V}_j$ over F.*

Proof. From Algorithm 1, we have

$$\mathcal{B}_j = \sum_{i=0}^{n-1} \sum_{\substack{|I|=i, \\ I \subseteq \{1,\ldots,n\}}} \mathbb{1}_{\{j \in \bar{I}\}} \bigcap_{t \in \bar{I}} \mathcal{V}_t, \tag{48}$$

so clearly $\mathcal{B}_j \subseteq \mathcal{V}_j$. Suppose that $v \in \mathcal{V}_j \backslash \mathcal{B}_j$, then there exists a subset $\bar{I} \subset \{1, \ldots, n\}$ with $1 \leqslant |\bar{I}| \leqslant n$ such that $j \in \bar{I}$ and that

$$v \notin \bigcap_{t \in \bar{I}} \mathcal{V}_t.$$

However, $\mathcal{B}_j \supseteq \bigcap_{t \in \bar{I}} \mathcal{V}_t$ for every $\bar{I}$ with $1 \leqslant |\bar{I}| \leqslant n$ such that $j \in \bar{I}$, which is a contradiction. Hence, $\mathcal{B}_j = \mathcal{V}_j$. □

Proposition 18. *Algorithm 7 returns a basis B for K over F.*

Proof. From Algorithm 1, for every $\bar{I} \subseteq \{1, \ldots, n\}$ with $1 \leqslant |\bar{I}| \leqslant n$ and for every $j \in \bar{I}$, the set B_j contains a basis of the subspace $\mathcal{U}_I = \bigcap_{t \in \bar{I}} \mathcal{V}_t$. It follows that for every $\bar{I} \subseteq \{1, \ldots, n\}$ with $1 \leqslant |\bar{I}| \leqslant n$, the set $\bigcap_{t \in \bar{I}} B_t$ is a basis for $\bigcap_{t \in \bar{I}} \mathcal{V}_t$.

Now by Proposition 17, B_1, B_2 are bases for $\mathcal{V}_1, \mathcal{V}_2$ over F, respectively. From the above, we have $B_1 \cap B_2$ is a basis of $\mathcal{V}_1 \cap \mathcal{V}_2$ over F. It follows that $\dim_F(\mathcal{V}_1 \cap \mathcal{V}_2) = |B_1 \cap B_2|$. Then

$$\begin{aligned} \dim_F(\mathcal{V}_1 + \mathcal{V}_2) &= \dim_F(\mathcal{V}_1) + \dim_F(\mathcal{V}_2) \\ &\quad - \dim_F(\mathcal{V}_1 \cap \mathcal{V}_2) \\ &= |B_1| + |B_2| - |B_1 \cap B_2| \\ &= |B_1 \cup B_2|. \end{aligned}$$

By definition, $\mathrm{Span}_F(B_1 \cup B_2) = \mathcal{V}_1 + \mathcal{V}_2$, and so the set $B_1 \cup B_2$ is a basis of $\mathcal{V}_1 + \mathcal{V}_2$ over F. By a straightforward induction argument, we conclude that $\bigcup_{j=1}^n B_j$ is a basis for $\sum_{j=1}^n \mathcal{V}_j$ over F.

Since $\sum_{j=1}^n \mathcal{V}_j \subseteq K$, we have $|\bigcup_{j=1}^n B_j| \leqslant [K : F] = l$. It follows that we can extend the set $\bar{B} = \bigcup_{j=1}^n B_j$ to a basis B of K over F. □

Now we are ready to present a repair scheme for the code $\mathcal{C}$ with the optimal access property. Let $B = (b_m)$ be the basis of K over F constructed above and let $B^* = (b_m^*)$ be its dual basis. Given a codeword $c = (c_1, \ldots, c_n) \in \mathcal{C}$, we expand its coordinates in the basis B^*, writing

$$c_i = \sum_{m=0}^{l-1} c_{i,m} b_m^*. \tag{49}$$

Suppose that c_i is the erased coordinate of c (the "failed node"). The starting point, as above, is Eq. (34), and our first step is to choose l dual codewords $c_t^\perp, t = 1, \ldots, l$ that support the repair. Construct the $l \times n$ matrix $C^\perp = AH$ and take the rows of C to be the needed codewords $c_t^\perp$. Since $c_t^\perp \cdot c = 0$ for all t, we have $c_{t,i}^\perp c_i = -\sum_{\substack{j=1 \\ j \neq i}}^n c_{t,j}^\perp c_j$ for all $t = 1, \ldots, l$. Computing the trace $\mathrm{tr}_{K/F}$, we obtain

$$\begin{aligned} \mathrm{tr}_{K/F}(c_{t,i}^\perp c_i) &= -\sum_{j \neq i}^n \mathrm{tr}_{K/F}(c_{t,j}^\perp c_j) \\ &= -\sum_{j \neq i}^n \mathrm{tr}_{K/F}(c_{t,j}^\perp \sum_{m=0}^{l-1} c_{j,m} b_m^*) \\ &= -\sum_{j \neq i}^n \sum_{m=0}^{l-1} \mathrm{tr}_{K/F}(c_{t,j}^\perp b_m^*) c_{j,m}. \end{aligned} \tag{50}$$

Note that for each $j \in \{1, \ldots, n\} \backslash \{i\}$, we have

$$\mathrm{Span}_F(c_{1,j}^\perp, \ldots, c_{l,j}^\perp) = \mathrm{Span}_F(Ah_j) = \mathcal{V}_i, \tag{51}$$

where the last equality follows by the assumption of constant repair subspaces. By Proposition 17, the set $B_i \subseteq B$ is a basis for $\mathcal{V}_i$ over F. Therefore, $c_{t,j}^\perp$ can be linearly generated by the set B_i for every $t = 1, \ldots, l$. More precisely, let $B_i = \{b_{i,u} \,|\, u = 1, \ldots, l/r\}$, then we have

$$c_{t,j}^\perp = \sum_{u=1}^{l/r} \gamma_{j,u} b_{i,u} \tag{52}$$

for some $\gamma_{j,u}, u = 1, \ldots, l/r$. Substituting into (50), we obtain the equality

$$\mathrm{tr}_{K/F}(c_{t,i}^\perp c_i) = -\sum_{j \neq i}^n \sum_{m=0}^{l-1} \sum_{u=1}^{l/r} \mathrm{tr}_{K/F}(b_{i,u} b_m^*) \gamma_{j,u} c_{j,m}. \tag{53}$$

It follows that to determine the left-hand side of (53), on each node $c_j, j \neq i$ the repair procedure needs to access the set of symbols $\{c_{j,m} \,|\, \mathrm{tr}_{K/F}(b_{i,u} b_m^*) = 1\}$. Since $B_i \subseteq B$ and B^* is the dual basis of B for K over F, the cardinality of this subset equals $|B_i| = l/r$, verifying that the repair can be accomplished with the minimum possible access cost.

C. Optimal-access repair for general scalar MSR codes

In this section we extend the above arguments for optimal repair schemes that do not necessarily have constant repair subspaces. This is done by a simple extension of Algorithm 1. We use the same notation as in Sec. V-B.

1) Repair degree $d = n - 1$:

Assume that the index of the failed node is $i \in \{1, \ldots, n\}$. By Proposition 14, for each $j \in \{1, \ldots, n\} \backslash \{i\}$, we have

$$\dim_F(\mathcal{A}_u) = \dim_F(Ah_j) = \frac{l}{r}, \quad u = 1, \ldots, r.$$

It follows that for $j \in J^c \backslash \{i\}$ we have

$$\mathcal{A}_1 h_{j,1} = \mathcal{A}_2 h_{j,2} = \cdots = \mathcal{A}_r h_{j,r}.$$

Let $J = (i_1, \ldots, i_r)$ be the set of parity nodes written in increasing order of their indices, and for $i_t \in J$ let $\sigma(i_t) = t$. Define

$$\mathcal{V}_i^{(j)} = \begin{cases} \mathcal{A}_1 h_{j,1} & j \in J^c \backslash \{i\}, \\ \mathcal{A}_{\sigma(j)} & j \in J. \end{cases} \tag{54}$$

Algorithm 2: Construction of an optimal basis; repair degree $d = n - 1$

Input: Subspaces $\mathcal{V}_i^{(j)}, i \in \{1, \dots, n\}, j \in \{1, \dots, n\} \backslash \{i\}$.
Output: A basis B for K over F.

for $i \leftarrow 1$ **to** n **do**
  foreach $j \in \{1, \dots, n\} \backslash \{i\}$ **do**
    $B_i^{(j)} \leftarrow \varnothing$;
    $\mathcal{B}_i^{(j)} \leftarrow \{0\}$;
$\Omega \leftarrow \{1, \dots, n\}^2 \backslash \{(i, i) \mid i = 1, \dots, n\}$;
for $u \leftarrow 0$ **to** $n^2 - n - 1$ **do**
  foreach $I \subseteq \Omega$ *such that* $|I| = u$ **do**
    $\bar{I} \leftarrow \Omega \backslash I$;
    $\mathcal{U}_I \leftarrow \bigcap_{(i,j) \in \bar{I}} \mathcal{V}_i^{(j)}$;
    for $i \leftarrow 1$ **to** n **do**
      foreach $j \in \{1, \dots, n\} \backslash \{i\}$ **do**
        if $(i, j) \in \bar{I}$ **then**
          $\mathcal{B}_i^{(j)} \leftarrow \mathcal{B}_i^{(j)} + \mathcal{U}_I$;
          Extend the set $B_i^{(j)}$ to be a basis of $\mathcal{B}_i^{(j)}$ over F;
$\bar{B} \leftarrow \bigcup_{i=1}^{n} \bigcup_{j \neq i}^{n} B_i^{(j)}$;
Extend the set $\bar{B}$ to be a basis B for K over F;

Proposition 19. *When Algorithm 2 terminates, we have $\mathcal{B}_i^{(j)} = \mathcal{V}_i^{(j)}$ for $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, n\} \backslash \{i\}$, and thus $B_i^{(j)}$ is a basis for $\mathcal{V}_i^{(j)}$ over F.*

Proposition 20. *Algorithm 2 returns a basis B for K over F.*

The proofs of Propositions 19 and 20 follow closely the proofs of Proposition 17 and 18 and will be omitted.

Now it is not difficult to see that we can repair the failed node c_i with optimal access cost relying on the basis B. Indeed, for each $j \in \{1, \dots, n\} \backslash \{i\}$, we have

$$\mathrm{Span}_F(c_{1,j}^{\perp}, \dots, c_{l,j}^{\perp}) = \mathrm{Span}_F(Ah_j) = \mathcal{V}_i^{(j)}. \tag{55}$$

By Algorithm 2 and Proposition 19, the set $B_i^{(j)} \subseteq B$ is a basis for $\mathcal{V}_i^{(j)}$ over F. Therefore, $c_{t,j}^{\perp}$ can be linearly generated by the set $B_i^{(j)}$ for every $t = 1, \dots, l$. Let $B_i^{(j)} = \{b_{i,u}^{(j)} \mid u = 1, \dots, l/r\}$. Then, similarly to (52) and (53), we have

$$c_{t,j}^{\perp} = \sum_{u=1}^{l/r} \gamma_{j,u} b_{i,u}^{(j)}, \tag{56}$$

$$\mathrm{tr}_{K/F}(c_{t,i}^{\perp} c_i) = -\sum_{j \neq i}^{n} \sum_{m=0}^{l-1} \sum_{u=1}^{l/r} \mathrm{tr}_{K/F}(b_{i,u}^{(j)} b_m^*) \gamma_{j,u} c_{j,m}. \tag{57}$$

Therefore, each node $c_j, j \neq i$ needs to access the set of symbols $\{c_{j,m} \mid \mathrm{tr}_{K/F}(b_{i,u}^{(j)} b_m^*) = 1\}$, whose cardinality is given by $|B_i^{(j)}| = l/r$. It follows that the repair scheme has the optimal access property.

2) Arbitrary repair degree:

So far we assumed that the repair relies on all the surviving nodes except for the single failed node, i.e., $|\mathcal{R}| = n - 1$. In this section we derive the most general version of the result of this section, that any scalar MDS code can be repaired with optimal access from any subset of helper nodes $\mathcal{R}$ of size $d, k + 1 \leqslant d \leqslant n - 1$. Let $s := d - k + 1$.

Let $G = [g_1 | g_2 | \dots | g_n]$ be a $k \times n$ generator matrix of $\mathcal{C}$, where g_i is a k-column over K. Let $i \in \{1, \dots, n\}$ and let $\mathcal{R} \not\ni \{i\}$ be a subset of d helper nodes. Let $\hat{\mathcal{R}} = \mathcal{R} \cup \{i\}$ and $G_{\hat{\mathcal{R}}}$ be the $k \times (d+1)$ submatrix formed by the columns $g_j, j \in \hat{\mathcal{R}}$. Clearly, $G_{\hat{\mathcal{R}}}$ defines a $(d+1, k)$ punctured code $\mathcal{C}_{\hat{\mathcal{R}}}$ of the code $\mathcal{C}$. Since $\mathcal{C}$ is MDS, the code $\mathcal{C}_{\hat{\mathcal{R}}}$ is itself MDS. Let $H^{\hat{\mathcal{R}}} = (h_i^{(\hat{\mathcal{R}})}, i = 1, \dots, d+1)$ be an the $s \times (d+1)$ parity-check matrix of the code $\mathcal{C}_{\hat{\mathcal{R}}}$. Recalling Remark 1, the code generated by $H^{\hat{\mathcal{R}}}$ is a shortened code $(\mathcal{C}^{\perp})^{\hat{\mathcal{R}}}$, i.e., a subcode of $\mathcal{C}^{\perp}$ formed of the codewords with zeros in the coordinates in $\hat{\mathcal{R}}^c$.

Suppose that the code $\mathcal{C}$ can optimally repair any single failed node i from the coordinates in $\mathcal{R} = \hat{\mathcal{R}} \backslash \{i\}$. This means that the MDS code $\mathcal{C}_{\hat{\mathcal{R}}}$ can optimally repair any single failed node i from the helper nodes $\hat{\mathcal{R}} \backslash \{i\}$. Let $J \subseteq \hat{\mathcal{R}}, |J| = s$ and $i \notin J$ and assume without loss of generality that the submatrix $H_J^{\hat{\mathcal{R}}}$ is an $s \times s$ identity matrix. Now Proposition 14 applied for the code $\mathcal{C}_{\hat{\mathcal{R}}}$ guarantees that there exist vectors $a_u \in K^{l/s}, u = 1, \dots, s$ such that the block-diagonal matrix $A = \mathrm{Diag}(a_1, \dots, a_s)$ satisfies

$$\dim_F(Ah_i^{(\hat{\mathcal{R}})}) = l, \tag{58}$$

$$\dim_F(Ah_j^{(\hat{\mathcal{R}})}) = \frac{l}{s}, \quad j \in \hat{\mathcal{R}} \backslash \{i\}, \tag{59}$$

$$\dim_F(\mathcal{A}_u) = \frac{l}{s}, \quad u = 1, \dots, s, \tag{60}$$

where $\mathcal{A}_u := \mathrm{Span}_F(a_u)$.

It follows from (59) and (60) that for $j \in \hat{\mathcal{R}} \backslash (J \cup \{i\})$, we have

$$\mathcal{A}_1 h_{j,1}^{(\hat{\mathcal{R}})} = \mathcal{A}_2 h_{j,2}^{(\hat{\mathcal{R}})} = \dots = \mathcal{A}_s h_{j,s}^{(\hat{\mathcal{R}})}.$$

Let us define

$$\mathcal{V}_{\hat{\mathcal{R}},i}^{(j)} = \begin{cases} \mathcal{A}_1 h_{j,1}^{(\hat{\mathcal{R}})} & j \in \hat{\mathcal{R}} \backslash (J \cup \{i\}), \\ \mathcal{A}_{\sigma(i)} & j \in J, \end{cases} \tag{61}$$

where σ is a bijection between J and $\{1, \dots, s\}$ defined as before (54).

The procedure to construct a basis for optimal-access repair in this case is constructed as a modification of Algorithm 2, and is given in Algorithm 3.

Similarly to the previous sections, we have the following propositions, whose proofs are analogous to the proofs of Propositions 17 and 18.

Proposition 21. *When Algorithm 3 terminates, we have $\mathcal{B}_{\hat{\mathcal{R}},i}^{(j)} = \mathcal{V}_{\hat{\mathcal{R}},i}^{(j)}$ for $\hat{\mathcal{R}} \subseteq \{1, \dots, n\}$ with $|\hat{\mathcal{R}}| = d + 1$, $i \in \hat{\mathcal{R}}$, and $j \in \hat{\mathcal{R}} \backslash \{i\}$, and thus $B_{\hat{\mathcal{R}},i}^{(j)}$ is a basis of $\mathcal{V}_{\hat{\mathcal{R}},i}^{(j)}$ over F.*

Proposition 22. *Algorithm 3 returns a basis B of K over F.*

The basis of K over F constructed in the algorithm enables us to construct an optimal-access repair scheme for the code $\mathcal{C}$.

Algorithm 3: Construction of an optimal basis; arbitrary repair degree

Input: Subspaces $\mathcal{V}^{(j)}_{\hat{\mathcal{R}},i}$ for each $\hat{\mathcal{R}} \subseteq \{1,\dots,n\}$ such that $|\hat{\mathcal{R}}| = d+1$ and $i \in \hat{\mathcal{R}}, j \in \hat{\mathcal{R}}\backslash\{i\}$.
Output: A basis B for K over F.

foreach $\hat{\mathcal{R}} \subseteq \{1,\dots,n\}$ *such that* $|\hat{\mathcal{R}}| = d+1$ **do**
  foreach $i \in \hat{\mathcal{R}}$ **do**
    foreach $j \in \hat{\mathcal{R}}\backslash\{i\}$ **do**
      $B^{(j)}_{\hat{\mathcal{R}},i} \leftarrow \varnothing$;
      $\mathcal{B}^{(j)}_{\hat{\mathcal{R}},i} \leftarrow \{0\}$;
$\Omega \leftarrow \{(\hat{\mathcal{R}}, i, j) \mid \hat{\mathcal{R}} \subseteq \{1,\dots,n\}, i \in \hat{\mathcal{R}}, j \in \hat{\mathcal{R}}\backslash\{i\}\}$;
for $u \leftarrow 0$ **to** $\binom{n}{d+1}((d+1)^2 - (d+1)) - 1$ **do**
  foreach $I \subseteq \Omega$ *such that* $|I| = u$ **do**
    $\bar{I} \leftarrow \Omega\backslash I$;
    $\mathcal{U}_I \leftarrow \bigcap_{(\hat{\mathcal{R}},i,j)\in\bar{I}} \mathcal{V}^{(j)}_i$;
    foreach $\hat{\mathcal{R}} \subseteq \{1,\dots,n\}$ *such that* $|\hat{\mathcal{R}}| = d+1$ **do**
      foreach $i \in \hat{\mathcal{R}}$ **do**
        foreach $j \in \hat{\mathcal{R}}\backslash\{i\}$ **do**
          if $(\hat{\mathcal{R}}, i, j) \in \bar{I}$ **then**
            $\mathcal{B}^{(j)}_{\hat{\mathcal{R}},i} \leftarrow \mathcal{B}^{(j)}_{\hat{\mathcal{R}},i} + \mathcal{U}_I$;
            Extend the set $B^{(j)}_{\hat{\mathcal{R}},i}$ to be a basis of $\mathcal{B}^{(j)}_{\hat{\mathcal{R}},i}$ over F;
$\bar{B} \leftarrow \bigcup_{\hat{\mathcal{R}}\subseteq\{1,\dots,n\},|\hat{\mathcal{R}}|=d+1} \bigcup_{i\in\hat{\mathcal{R}}} \bigcup_{j\neq\hat{\mathcal{R}}\backslash\{i\}} B^{(j)}_{\hat{\mathcal{R}},i}$;
Extend the set $\bar{B}$ to be a basis B of K over F;

Let $d \in \{k+1,\dots,n-1\}$ be the repair degree. Let $(c_1,\dots,c_n)$ be a codeword of the code $\mathcal{C}$ written on the storage nodes, and suppose that the failed node is i and that $\mathcal{R}$ be the set of d helper nodes. Let A be the block-diagonal matrix defined above, constructed with respect to i and $H^{\hat{\mathcal{R}}}$. Define the matrix $C^{\perp} = AH^{\hat{\mathcal{R}}}$ and note that its rows $c^{\perp}_t, t = 1,\dots,l$ form codewords of the code dual to the punctured code $\mathcal{C}_{\hat{\mathcal{R}}}$. Letting $c^{\perp}_t = (c^{\perp}_{t,i})_{i\in\mathcal{R}}$, we can write

$$c^{\perp}_{t,i}c_i = -\sum_{j\in\mathcal{R}} c^{\perp}_{t,j}c_j. \tag{62}$$

Similarly to (50), we have

$$\mathrm{tr}_{K/F}(c^{\perp}_{t,i}c_i) = -\sum_{j\in\mathcal{R}}\sum_{m=0}^{l-1} \mathrm{tr}_{K/F}(c^{\perp}_{t,j}b^*_m)c_{j,m}, \tag{63}$$

where $B^* = (b^*)$ is the dual basis of the basis B. Note that for $j \in \mathcal{R}$ we have

$$\mathrm{Span}_F(c^{\perp}_{1,j},\dots,c^{\perp}_{l,j}) = \mathrm{Span}_F(Ah_j) = \mathcal{V}^{(j)}_{\mathcal{R},i}. \tag{64}$$

By Algorithm 3 and Proposition 21, the set $B^{(j)}_{\mathcal{R},i} \subseteq B$ forms a basis for the subspace $\mathcal{V}^{(i)}_{\mathcal{R},j}$ over F. Therefore, the element $c^{\perp}_{t,j}$ can be linearly generated by the set $B^{(j)}_{\mathcal{R},i}$ for every $t = 1,\dots,l$. Let $B^{(j)}_{\mathcal{R},i} = \{b^{(j)}_{\mathcal{R},i,u} \mid u = 1,\dots,l/s\}$. Then, similarly to (52) and (53), we have

$$c^{\perp}_{t,j} = \sum_{u=1}^{l/s} \gamma_{j,u} b^{(j)}_{\mathcal{R},i,u}, \tag{65}$$

$$\mathrm{tr}_{K/F}(c^{\perp}_{t,i}c_i) = -\sum_{j\in\mathcal{R}}\sum_{m=0}^{l-1}\sum_{u=1}^{l/s} \mathrm{tr}_{K/F}(b^{(j)}_{\mathcal{R},i,u}b^*_m)\gamma_{j,u}c_{j,m}. \tag{66}$$

Therefore, each node $c_j, j \in \mathcal{R}$ needs to access the set of symbol $\{c_{j,m} \mid \mathrm{tr}_{K/F}(b^{(j)}_{\mathcal{R},i,u}b^*_m) = 1\}$, whose cardinality equals $|B^{(j)}_{\mathcal{R},i}| = l/s$. It follows that the constructed repair scheme has the optimal access property.

This completes the proof of Theorem 12.

VI. Concluding remarks

We have shown that error correction is feasible in the original code family of [21] without the increase of the extension degree of the locator field of the code (the node size). Namely, codes from [21] use extension degree $l = (d-k+1)L$, where L is the product of the first n distinct primes in an arithmetic progression,

$$L = \Bigg(\prod_{\substack{i=1 \\ p_i \equiv 1 \bmod (d-k+1)}}^{n} p_i\Bigg).$$

The lower bound on l from [21], necessary for repair of a single node, has the form $l \geqslant \prod_{i=1}^{k-1} p_i$, where p_i is the i-th smallest prime. Asymptotically for fixed $d-k$ and growing n we obtain the following bounds on the node size: $\Omega(k^k) \leqslant l \leqslant O(n^n)$. Essentially the same node size is used in this paper for repair with error correction. At the same time, the explicit RS code family with optimal access that we construct comes at the expense of larger node size, namely $l = (d-k+1)^n L$. Since there is an optimal-access repair scheme for every scalar MSR code, this leaves a gap between what is known explicitly and what is shown to be possible, which represents a remaining open question related to the task of optimal repair of RS codes.

Appendix A
Proof of Proposition 7

First we present the proof for the case $h = 0$ (strictly speaking, we do not have to isolate it, but it makes understanding the general case much easier). In this case, definition (21), (22) simplifies as follows. Let $f_0(x) = x^{p_1} - f(x)$. Write f_0 as

$$\begin{aligned} f_0(x) &= a_0 + a_1x + a_2x^2 + \cdots + a_{p_1-1}x^{p_1-1} \\ &= \sum_{q=0}^{(p_1-1)/s-1} x^{qs} f_{0,q}(x), \end{aligned}$$

where

$$\begin{aligned} f_{0,0}(x) &= a_0 + a_1x + \cdots + a_{s-1}x^{s-1} \\ f_{0,1}(x) &= a_s + a_{s+1}x + \cdots + a_{2s-1}x^{s-1} \\ &\cdots \\ f_{0,(p_1-1)/s-1}(x) &= a_{p_1-1-s} + a_{p_1-s}x + \cdots + a_{p_1-1}x^s, \end{aligned} \tag{67}$$

so that the degree of the last polynomial is $\leqslant s$ and the degrees of the remaining ones are $\leqslant s-1$. Obviously, we have

$$\alpha_1^{p_1} = f_0(\alpha_1) \tag{68}$$
$$= \sum_{q=0}^{(p_1-1)/s-1} \alpha_1^{qs} f_{0,q}(\alpha_1). \tag{69}$$

As before, we start with (10), which implies that for any polynomial $g \in F_1[x]$ of degree $\deg g \leqslant n-k-1$, we have

$$\operatorname{tr}(e_i v_1 g(\alpha_1) c_1) = -\sum_{j=2}^{n} g(\alpha_j) \operatorname{tr}(e_i v_j c_j). \tag{70}$$

Take $e_i = \alpha_1^{qs}$ and $g(x) = x^t f_{0,q}(\alpha_1)$ and sum on q on the left, then from (69) we obtain $\operatorname{tr}(v_1 \alpha_1^t f_0(\alpha_1) c_1)$. Summing on q on the right of (70) and using (68), we conclude that

$$\operatorname{tr}(v_1 \alpha_1^{p_1+t} c_1) = -\sum_{q=0}^{(p_1-1)/s-1} \sum_{j=2}^{n} \alpha_j^t f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j) \tag{71}$$

for all $t = 0, 1, \ldots, n-k-s-1$, Note that the constraint $t \leqslant n-k-s-1$ is implied by the condition $\deg(g) = \deg(x^t f_{0,q}(x)) \leqslant n-k-1$ needed in order to use (70) (and (10)). Change the variable $t \mapsto (t-1)$ to write the last equation as

$$\operatorname{tr}(v_1 \alpha_1^{p_1-1+t} c_1) = -\sum_{q=0}^{(p_1-1)/s-1} \sum_{j=2}^{n} \alpha_j^{t-1} f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j), \quad t = 1, 2, \ldots, n-k-s. \tag{72}$$

From (17) and the fact that

$$\bigcap_{u=1}^{s-1} \{u-s, u-s+1, \ldots, u-s+n-k-1\} = \{-1, 0, 1, \ldots, n-k-s\},$$

we obtain

$$\operatorname{tr}(\beta^u \alpha_1^{p_1-1+t} v_1 c_1) = -\sum_{j=2}^{n} \alpha_j^{t-u+s} \operatorname{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j), \quad -1 \leqslant t \leqslant n-k-s, \quad 1 \leqslant u \leqslant s-1.$$

Summing these equations on $u = 1, 2, \ldots, s-1$, we obtain the relation

$$\operatorname{tr}\Big(\sum_{u=1}^{s-1} \beta^u \alpha_1^{p_1-1+t} v_1 c_1\Big) = -\sum_{j=2}^{n} \sum_{u=1}^{s-1} \alpha_j^{t-u+s} \operatorname{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j), \quad -1 \leqslant t \leqslant n-k-s.$$

For each $t = 1, 2, \ldots, n-k-s$ let us add this equation and (72). This gives $n-k-s$ relations of the form

$$\operatorname{tr}\Big(\sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1+t} v_1 c_1\Big) = -\sum_{j=2}^{n} \Big(\sum_{u=1}^{s-1} \alpha_j^{t-u+s} \operatorname{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j) + \sum_{q=0}^{(p_1-1)/s-1} \alpha_j^{t-1} f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j) \Big).$$

Observe that the left-hand side of this equation is the same as the left-hand side of (20). Therefore,

$$\sum_{j=2}^{n} \alpha_j^t \operatorname{tr}\Big(\sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1} v_j c_j\Big) = \sum_{j=2}^{n} \Big(\sum_{u=1}^{s-1} \alpha_j^{t-u+s} \operatorname{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j) + \sum_{q=0}^{(p_1-1)/s-1} \alpha_j^{t-1} f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j) \Big), \quad 1 \leqslant t \leqslant n-k-s.$$

Replacing $t-1$ with t in this equation, we obtain that

$$\sum_{j=2}^{n} \alpha_j^t \Big(\sum_{u=1}^{s-1} \alpha_j^{s-u+1} \operatorname{tr}(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j) + \sum_{q=0}^{(p_1-1)/s-1} f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j) - \alpha_j \operatorname{tr}\Big(\sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1} v_j c_j \Big) \Big) = 0, \quad 0 \leqslant t \leqslant n-k-s-1.$$

By Proposition 1, the vector

$$\Big(\sum_{u=1}^{s-1} \alpha_j^{s-u+1} \operatorname{tr}\Big(\beta^u \alpha_1^{u+p_1-s-1} v_j c_j\Big) + \sum_{q=0}^{(p_1-1)/s-1} f_{0,q}(\alpha_j) \operatorname{tr}(\alpha_1^{qs} v_j c_j) - \alpha_j \operatorname{tr}\Big(\sum_{u=0}^{s-1} \beta^u \alpha_1^{p_1-1} v_j c_j \Big), j = 2, \ldots, n \Big) \tag{73}$$

is contained in a GRS code of length $n-1$ and dimension $s+k-1$. This proves the case $h=0$ of the proposition.

Now let us consider the general case $0 \leqslant h \leqslant s-1$. From (21) and (22) we obtain

$$\alpha_1^{p_1+h} = f_h(\alpha_1) = \sum_{q=0}^{(p_1-1)/s-1} \alpha_1^{qs} f_{h,q}(\alpha_1). \tag{74}$$

This relation enables us to use the argument that yielded (71) above: Take $e_i = \alpha_1^{u+qs}\beta^u$ and $g(x) = x^t f_{h,q}(x)$ in (70) and sum on $q = 0, 1, \ldots, (p_1-1)/s-1$. We obtain for $h = 0, \ldots, s-1$ and $u = 0, \ldots, s-1-h$

$$\operatorname{tr}(\alpha_1^{p_1+h+u+t} \beta^u v_1 c_1) = \sum_{q=0}^{(p_1-1)/s-1} \operatorname{tr}(\alpha_1^{qs+u+t} \beta^u f_{h,q}(\alpha_1) v_1 c_1)$$

$$= -\sum_{q=0}^{(p_1-1)/s-1}\sum_{j=2}^{n}\alpha_j^t f_{h,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j),$$
$$t = 0,1,\dots,n-k-s-1.$$

The restriction $t \leqslant n-k-s-1$ is imposed in the same way as in (71) (namely, it is necessary that $\deg(x^t f_{h,q}(x)) \leqslant n-k-1$). Replacing $h+u$ with h in the last equation, we obtain that

$$\operatorname{tr}(\alpha_1^{p_1+h+t}\beta^u v_1 c_1)$$
$$= -\sum_{q=0}^{(p_1-1)/s-1}\sum_{j=2}^{n}\alpha_j^t f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j),$$
$$0 \leqslant h \leqslant s-1, \quad 0 \leqslant u \leqslant h, \quad 0 \leqslant t \leqslant n-k-s-1.$$

Let us sum these equations on $u = 0,1,\dots,h$ to obtain

$$\operatorname{tr}(\alpha_1^{p_1+h+t}\sum_{u=0}^{h}\beta^u v_1 c_1)$$
$$= -\sum_{j=2}^{n}\sum_{q=0}^{(p_1-1)/s-1}\sum_{u=0}^{h}\alpha_j^t f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j),$$
$$0 \leqslant h \leqslant s-1,\ 0 \leqslant t \leqslant n-k-s-1.$$

Replacing t with $t-1$, we obtain that

$$\operatorname{tr}\Big(\alpha_1^{p_1-1+h+t}\sum_{u=0}^{h}\beta^u v_1 c_1\Big)$$
$$= -\sum_{j=2}^{n}\sum_{q=0}^{(p_1-1)/s-1}\sum_{u=0}^{h}\alpha_j^{t-1} f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j),$$
$$0 \leqslant h \leqslant s-1,\ 1 \leqslant t \leqslant n-k-s. \quad (75)$$

According to (17) and the fact that

$$\bigcap_{u=h+1}^{s-1}\{u-s, u-s+1,\dots,u-s+n-k-1\}$$
$$= \{-1,0,1,\dots,n-k-s+h\},$$

for $0 \leqslant h \leqslant s-1$, we have

$$\operatorname{tr}(\beta^u\alpha_1^{p_1-1+t}v_1c_1) = -\sum_{j=2}^{n}\alpha_j^{t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$-1 \leqslant t \leqslant n-k-s+h,\ h+1 \leqslant u \leqslant s-1.$$

Replacing t with $t+h$, we have

$$\operatorname{tr}(\beta^u\alpha_1^{p_1-1+h+t}v_1c_1)$$
$$= -\sum_{j=2}^{n}\alpha_j^{h+t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$-h-1 \leqslant t \leqslant n-k-s,\ h+1 \leqslant u \leqslant s-1.$$

Summing these equations on $u = h+1, h+2,\dots,s-1$, we obtain

$$\operatorname{tr}\Big(\sum_{u=h+1}^{s-1}\beta^u\alpha_1^{p_1-1+h+t}v_1c_1\Big)$$
$$= -\sum_{j=2}^{n}\sum_{u=h+1}^{s-1}\alpha_j^{h+t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$-h-1 \leqslant t \leqslant n-k-s.$$

Finally, adding together this equation and (75), we obtain that

$$\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1+h+t}v_1c_1\Big)$$
$$= -\sum_{j=2}^{n}\sum_{q=0}^{(p_1-1)/s-1}\sum_{u=0}^{h}\alpha_j^{t-1}f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j)$$
$$-\sum_{j=2}^{n}\sum_{u=h+1}^{s-1}\alpha_j^{h+t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$0 \leqslant h \leqslant s-1,\ 1 \leqslant t \leqslant n-k-s. \quad (76)$$

Going back to (20), let us perform the change $t \mapsto t+h$, then we obtain

$$\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1+h+t}v_1c_1\Big)$$
$$= -\sum_{j=2}^{n}\alpha_j^{h+t}\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big),$$
$$-h \leqslant t \leqslant n-k-h-1. \quad (77)$$

For $t = 1,2,\dots,n-k-s$ the left-hand sides of (76) and (77) coincide, and therefore,

$$\sum_{j=2}^{n}\alpha_j^{h+t}\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big)$$
$$= \sum_{j=2}^{n}\sum_{q=0}^{(p_1-1)/s-1}\sum_{u=0}^{h}\alpha_j^{t-1}f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j)$$
$$+\sum_{j=2}^{n}\sum_{u=h+1}^{s-1}\alpha_j^{h+t-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j),$$
$$1 \leqslant t \leqslant n-k-s.$$

Replacing t by $t+1$, we obtain that

$$\sum_{j=2}^{n}\alpha_j^t\Big(\sum_{q=0}^{(p_1-1)/s-1}\sum_{u=0}^{h}f_{h-u,q}(\alpha_j)\operatorname{tr}(\alpha_1^{u+qs}\beta^u v_j c_j)$$
$$+\sum_{u=h+1}^{s-1}\alpha_j^{h+1-u+s}\operatorname{tr}(\beta^u\alpha_1^{u+p_1-s-1}v_jc_j)$$
$$-\alpha_j^{h+1}\operatorname{tr}\Big(\sum_{u=0}^{s-1}\beta^u\alpha_1^{p_1-1}v_jc_j\Big)\Big) = 0,$$
$$0 \leqslant t \leqslant n-k-s-1.$$

The proof is complete.

References

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

[2] K. V. Rashmi, N. B. Shah, and P. V. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5227–5239, 2011.

[3] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: MDS array codes with optimal rebuilding," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1597–1616, 2013.

[4] M. Ye and A. Barg, "Explicit constructions of high-rate MDS array codes with optimal repair bandwidth," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2001–2014, 2017.

[5] ——, "Explicit constructions of optimal-access MDS codes with nearly optimal sub-packetization," *IEEE Trans. Inf. Theory*, vol. 63, no. 10, pp. 6307–6317, 2017.

[6] S. Goparaju, A. Fazeli, and A. Vardy, "Minimum storage regenerating codes for all parameters," *IEEE Trans. Inf. Theory*, vol. 63, no. 10, pp. 6318–6328, 2017.

[7] N. Raviv, N. Silberstein, and T. Etzion, "Constructions of high-rate minimum storage regenerating codes over small fields," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2015–2038, 2017.

[8] K. V. Rashmi, N. B. Shah, K. Ramchandran, and P. V. Kumar, "Regenerating codes for errors and erasures in distributed storage," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2012, pp. 1202–1206.

[9] S. Pawar, S. El Rouayheb, and K. Ramchandran, "Securing dynamic distributed storage systems against eavesdropping and adversarial attacks," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6734–6753, 2011.

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

[11] I. Tamo, Z. Wang, and J. Bruck, "Access versus bandwidth in codes for storage," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2028–2037, 2014.

[12] S. Balaji and P. V. Kumar, "A tight lower bound on the sub-packetization level of optimal-access MSR and MDS codes," *arXiv preprint arXiv:1710.05876*, 2017.

[13] V. R. Cadambe, C. Huang, and J. Li, "Permutation code: Optimal exact-repair of a single failed node in MDS code based distributed storage systems," in *2011 IEEE International Symposium on Information Theory (ISIT)*, 2011, pp. 1225–1229.

[14] M. Vajha *et al.*, "Clay codes: Moulding MDS codes to yield an MSR code," in *16th USENIX Conference on File and Storage Technologies, Oacland, CA*, 2018, pp. 139–154.

[15] Z. Chen and A. Barg, "Explicit constructions of MSR codes for clustered distributed storage: The rack-aware storage model," *IEEE Trans. Inf. Theory*, vol. 66, no. 2, pp. 886–899, 2020.

[16] V. Guruswami and M. Wootters, "Repairing Reed-Solomon codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 9, pp. 5684–5698, 2017.

[17] K. Shanmugam, D. S. Papailiopoulos, A. G. Dimakis, and G. Caire, "A repair framework for scalar MDS codes," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 998–1007, 2014.

[18] H. Dau, I. Duursma, H. M. Kiah, and O. Milenkovic, "Repairing Reed-Solomon codes with multiple erasures," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 6567–6582, 2018.

[19] H. Dau and O. Milenkovic, "Optimal repair schemes for some families of Reed-Solomon codes," in *2017 IEEE International Symposium on Information Theory*, 2017, pp. 346–350.

[20] J. Mardia, B. Bartan, and M. Wootters, "Repairing multiple failures for scalar MDS codes," *IEEE Trans. Inf. Theory*, vol. 65, no. 5, pp. 2661–2672, 2019.

[21] I. Tamo, M. Ye, and A. Barg, "The repair problem for Reed-Solomon codes: Optimal repair of single and multiple erasures, asymptotically optimal node size," *IEEE Trans. Inf. Theory*, vol. 65, no. 5, pp. 2673–2695, 2019.

[22] L. Jin, G. Luo, and C. Xing, "Optimal repairing schemes for Reed-Solomon cods with alphabet sizes linear in lengths under the rack-aware model," arXiv:1911.08016, Nov. 2019.

[23] H. Dau and E. Viterbo, "Repair schemes with optimal I/O costs for full-length Reed-Solomon codes with two parities," in *2018 IEEE Information Theory Workshop (ITW)*, 2018, pp. 1–5.

[24] H. Dau, I. Duursma, and H. Chu, "On the I/O costs of some repair schemes for full-length Reed-Solomon codes," in *2018 IEEE International Symposium Information Theory*, 2018, pp. 1700–1704.

[25] W. Li, H. Dau, Z. Wang, H. Jafarkhani, and E. Viterbo, "On the I/O costs in repairing short-length Reed-Solomon codes," in *2019 IEEE International Symposium Information Theory*, 2019, pp. 1087–1091.

Zitan Chen received his B.Eng. degree in Information Engineering in 2015 from the Chinese University of Hong Kong, Hong Kong. He is currently working toward his Ph.D. degree in the Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park.

Min Ye received his B.S. in Electrical Engineering from Peking University, Beijing, China in 2012, and his Ph.D. in the Department of Electrical and Computer Engineering, University of Maryland, College Park in 2017. He then spent two years as a postdoctoral researcher at Princeton University. Since 2019, he has been an assistant professor in the Data Science and Information Technology Research Center, Tsinghua-Berkeley Shenzhen Institute, Tsinghua Shenzhen International Graduate School, Shenzhen, China. He received the 2017 IEEE Data Storage Best Paper Award. His research interests include coding theory, information theory, differential privacy, and machine learning.

Alexander Barg is a professor in the Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, MD. He is broadly interested in information and coding theory, applied probability, and algebraic combinatorics, and has published about a hundred research papers. He received the 2015 Information Theory Society paper award, was a plenary speaker at the 2016 IEEE International Symposium on Information Theory (Barcelona, Spain), and in 2018-2019 served Editor-in-Chief of this TRANSACTIONS.