Distance-Preserving Codes with Multi-Level Access for Frequently-Updated Storage

Siyi Yang[1], *Student Member, IEEE*, Clayton Schoeny[1], *Student Member, IEEE*,
Laura Conde-Canencia[2], and Lara Dolecek[1], *Senior Member, IEEE*
[1] Department of Electrical and Computer Engineering, University of California, Los Angeles, Los Angeles, CA 90095 USA
[2] Lab-STICC, CNRS UMR 6285, Université de Bretagne Sud, Lorient, France

***Abstract*—Multi-level codes have been studied since they offer a good trade-off between the reliability and the speed of the reading process in modern storage systems. In our paper, we focus on another property called distance-preserving, which is crucial for trimming down the rewriting process in frequently updated storage and has not been discussed before in related literatures. We first provide a construction of a general prototype of codes that is both multi-level accessible and distance-preserving, based on so-called totally-invertible matrices. Based on the prototype code, we then present a special class of codes constructed from Cauchy matrix, a representative class from totally-invertible matrices, which are of particular interests because of their being efficiently decodable. We present the local and the global decoding algorithms with polynomial complexity for these codes. Our paper makes the first attempt on enabling multi-level reading and efficient rewriting simultaneously.**

I. Introduction

When it comes to modern storage systems, data reliability and reading efficiency are among the most concerned features. In conventional block codes, each block corrects up to a prescribed number of errors, thus reliability is ensured as long as the errors are evenly distributed in each block. However, even a single error exceeding the maximum error-correction capability will result in a failure, which makes the system extremely vulnerable to bursting errors. Extending the block length alleviates this problem notwithstanding, increases the reading complexity equally for all the symbols and is far from being efficient provided that bursting errors are rare events. Codes simultaneously presenting local and global error-correction capabilities are therefore of great interests since they maintain a good balance between data reliability and reading efficiency. We call these codes *multi-level codes* due to their offering multi-level decoding.

A major application of multi-level codes is in distributed storage system, where data are stored and managed distributively among various nodes that are either physically or virtually separated [1]. Multi-level codes not only effectively speed up the reading process in distributed storage systems, but also helps on data retrieval of defective nodes. Redundant Arrays of Independent Disks (RAID) architectures also accommodate multi-level codes well, especially when the devices are Solid State Drives like flash memories, where the basic data units are called pages. The so-called Sector-Disk (SD) codes are designed for resolving the problem incurred from page failures [2].

Various codes offering multi-level decoding have been proposed in recent literatures. Product codes, of course, are the most simple constructions that allow both local and global access. However, product codes are far from being efficient provided that the local parities and global parities are added separately. Reed-Solomon (RS) code has been a major prototype for constructions of double-level codes with efficient rates. The integrated-interleaving (I-I) codes [3] presented by Hassner *et al.* are the first constructions that offer an double-level error-correction and achieves the singleton bound. Nevertheless, the I-I codes can hardly serve as an adequate candidate since there is no clear mapping between the local information symbols and the corresponding sub-block in the codeword, that is to say, they are not multi-level accessible albeit multi-level decodable. To resolve this issue, Cassuto *et al.* proposed the multi-block interleaved codes [4] where each information symbol can be locally inferred from its corresponding sub-block. Locally recoverable code (LRC) [5] is also a related coding scheme that has drawn significant interests, focusing on local repair of individual code symbols, which is not aligned with our model inasmuch as global error-correction has not been considered.

While the locality of the reading process has been comprehensively studied, that of the rewriting process has hardly been considered. In spite of it, it is of great importance to introduce locality in rewriting process in frequently-updated storage, from both economical and computational perspective. In DNA storage [6]–[8], for example, rewriting the stored data requires synthesis of new DNA molecules, which is not only sophisticated but also costly. Therefore it is reasonable to expect a direct proportion between the number of DNA molecules to be replaced and the amount of information symbols being modified. This requirement also applies to frequently updated distributed storage systems, such as cloud storage, where the communication cost and time delay are among the major factors that affect the performance and thus local editing is always preferred.

If the number of codeword symbols need to be rewritten is upper bounded by a linear function of the amount of information symbols being modified, we say the code is *Distance-preserving*. The previously mentioned constructions, albeit offering good trade-off between local and global distances, are not distance-preserving. Observe that codes that are systematic both locally and globally are naturally distance-preserving, we

propose a construction of systematic multi-level codes in the remainder of this paper. Unlike previous coding schemes using RS codes as the prototype, we utilize a new class of auxiliary matrices called *totally invertible matrices* as an alternative to Vandermonde matrices. Not only our construction achieves the singleton bound of multi-level codes in a certain region, but it is also the first method that enable multi-level access and local rewriting at the same time. We also pay special attention to the case where the auxiliary matrices are Cauchy matrices, where the codes are able to be efficiently decoded and algorithms of polynomial complexity are proposed.

In the rest of this paper, we first introduce preliminaries of locally recoverable codes in Section II. In Section III, we first present the construction of a general class of multi-level codes that are also distance-preserving, based on so-called totally invertible matrices. We then provide a representative example of such codes that is constructed with Cauchy matrices, a special class of totally invertible matrices, which is important because they can be efficiently decoded. In Section IV, we describe an efficient decoding algorithm for this code that estimate the codewords in polynomial time. Finally, we conclude in Section V.

II. Preliminaries

Throughout the rest of this paper, $[N]$ represents $\{1, 2, \cdots, N\}$. The alphabet field, denoted by $GF(q)$, is a Galois field of size q. Messages and codewords are represented by $\mathbf{m}$ and $\mathbf{c}$, respectively. We say $\mathcal{C}$ is an $(\mathbf{m}, \mathbf{k}, \mathbf{d})_{\mathbf{q}}$**-code** if $\mathcal{C} \subset GF(q)^n$, $\dim(\mathcal{C}) = k$, and $\min\limits_{\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}, \mathbf{c}_1 \neq \mathbf{c}_2} d_H(\mathbf{c}_1, \mathbf{c}_2) = d$, where d_H refers to the Hamming distance. In a locally recoverable code $\mathcal{C}$, every consecutive p messages $\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_p$ are jointly mapped into consecutive x codewords $\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_p$, where $\mathbf{c}_i \in GF(q)^n$ for all $1 \leq i \leq p$. If there exists (n, k, d_1)-codes $\{\mathcal{C}_i\}_{i=1}^p$, such that $\mathbf{c}_i \in \mathcal{C}_i$ for all $1 \leq i \leq p$, and $\mathcal{C}$ is an (pn, pk, d_2)-code, then we call $\mathcal{C}$ an $(\mathbf{p}, \mathbf{n}, \mathbf{k}, \mathbf{d_1}, \mathbf{d_2})_{\mathbf{q}}$**-code**. Obviously any $(p, n, k, d_1, d_2)_q$-code is a multi-level code.

Lemma 1 is a known result from e.g., [9], which provides an upper bound of the global minimum distance d_2 of a code for a fixed local minimum distance d_1, and is known as the singleton bound for multi-level codes.

Lemma 1. *For an $(p, n, k, d_1, d_2)_q$-code, let $r = n - k$. For $\delta \in \mathbb{N}$, $\delta < r$, if $d_2 \leq n + 1$, $d_1 = r - \delta + 1$, then $d_2 \leq r + (p-1)\delta + 1$.*

Before we introduce constructions for multi-level codes, we first introduce an useful matrix, the so-called totally-invertible matrix in the remainder of this section.

Definition 1. *A matrix $X \in GF(q)^{u \times v}$ is called a* ***totally invertible*** *matrix if every $k \times k$ sub-matrix of X, $1 \leq k \leq \min\{u, v\}$, is nonsingular.*

Remark 1. *Let $a_1, \cdots, a_u, b_1, \cdots, b_v$ be pairwise distinct elements in $GF(q)$, then the following matrix is totally invertible and is known as* ***Cauchy matrix****,*

$$\begin{bmatrix} \frac{1}{a_1-b_1} & \frac{1}{a_1-b_2} & \cdots & \frac{1}{a_1-b_v} \\ \frac{1}{a_2-b_1} & \frac{1}{a_2-b_2} & \cdots & \frac{1}{a_2-b_v} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_u-b_1} & \frac{1}{a_u-b_2} & \cdots & \frac{1}{a_u-b_v} \end{bmatrix}.$$

For any matrix $\mathbf{A}$ in $GF(q)^{n \times v}$, we call $\mathbf{A}$ an v**-parity matrix** if any v rows of $\mathbf{A}$ are linearly independent. We know that any v-parity matrix is the parity check matrix of an $(n, n-v, v+1)_q$-code. Lemma 2 presents some v-parity matrices constructed based on totally invertible matrices.

Lemma 2. *Let $u, v, r \in \mathbb{N}$, $v \geq r$, $\mathbf{A} \in GF(q)^{(u+r) \times v}$. If $\mathbf{A}$ is a totally invertible matrix, then the following matrix $\mathbf{D}$ is an v-parity matrix:*

$$\mathbf{D} = \begin{bmatrix} \mathbf{A} \\ -\mathbf{I}_r \ \mathbf{0}_{v-r} \end{bmatrix}.$$

Proof. Suppose there exist v rows from $\mathbf{D}$ that are linearly dependent. Suppose a of these linearly dependent rows $r_1, \cdots, r_a$ are from $\mathbf{A}$, and the other $v-a$ rows $r_{a+1}, \cdots, r_v$ are from $[-\mathbf{I}_r \ \mathbf{0}_{v-r}]$, where $v - r \leq a \leq v$. Suppose the entries of -1 in $r_{a+1}, \cdots, r_v$ are located in the $j_1, \cdots, j_{v-a}$-th columns of $\mathbf{D}$, then $j_k \leq r$ for all $1 \leq k \leq v - a$. Suppose $[v] \backslash \{j_1, \cdots, j_{v-a}\} = \{c_1, \cdots, c_a\}$. Then the $a \times a$ sub-matrix generated by the intersection of the rows $r_1, \cdots, r_a$ and the $c_1, \cdots, c_a$-th columns of $\mathbf{A}$ is singular. A contradiction! ■

III. Encoding Scheme

In this section, we present constructions of distance-preserving multi-level codes that achieve the singleton bound in Lemma 1 in a certain region. In Theorem 1, we propose a general construction of $(p, n, k, r-\delta+1, \min\{r+(p-1)\delta+1, 2(r-\delta+1)\})_q$-codes using totally-invertible matrices, which achieve the singleton bound for $\delta \leq \frac{r+1}{p+1}$. Using them as the prototype, we then focus on a special case in **??** where the totally-invertible matrices are chosen to be Cauchy matrices. The nice structure of Cauchy matrices results in the existence of an efficient decoding algorithm for our proposed codes, thus is of particular interests to us.

Theorem 1. *Let $p, n, k, r, \delta \in \mathbb{N}$, $r = n - k$, $0 < \delta < \min\{k, r\}$. For $\mathbf{A}_{i,j} \in GF(q)^{k \times n}$ for all $1 \leq i, j \leq p$, define $\mathbf{G} \in GF(q)^{pk \times pn}$ as follows,*

$$G = \left[\begin{array}{c|c|c|c|c|c|c} \mathbf{I}_k & \mathbf{A}_{1,1} & \mathbf{0}_k & \mathbf{A}_{1,2} & \cdots & \mathbf{0}_k & \mathbf{A}_{1,p} \\ \hline \mathbf{0}_k & \mathbf{A}_{2,1} & \mathbf{I}_k & \mathbf{A}_{2,2} & \cdots & \mathbf{0}_k & \mathbf{A}_{2,p} \\ \hline \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \hline \mathbf{0}_k & \mathbf{A}_{p,1} & \mathbf{0}_k & \mathbf{A}_{p,2} & \cdots & \mathbf{I}_k & \mathbf{A}_{p,p} \end{array}\right]. \quad (1)$$

Suppose $\mathbf{A}_{i,j} = \mathbf{B}_{i,j}\mathbf{X}_{i,j}\mathbf{U}_j$, $1 \leq i, j \leq p$, $i \neq j$, for some $\mathbf{B}_{i,j} \in GF(q)^{k \times \delta}$, $\mathbf{X}_{i,j} \in GF(q)^{\delta \times \delta}$, $\mathbf{U}_j \in GF(q)^{\delta \times r}$, such that the following conditions are satisfied:

1) *$rank(\mathbf{B}_{i,j}) = rank(\mathbf{U}_j) = rank(\mathbf{X}_{i,j}) = \delta$, where $rank(\cdot)$ refers to the rank of the matrix in $GF(q)$;*
2) *$[\mathbf{A}_{i,i}, \mathbf{B}_{i,1}, \cdots, \mathbf{B}_{i,p}]$, $1 \leq i \leq p$, are totally invertible matrices;*

3) $\left[\mathbf{A}_{i,i}^T, \mathbf{U}_i^T\right]$, $1 \le i \le p$, *are totally invertible matrices. Then,* $\mathbf{G}$ *is the generator matrix of an* $(p, n, k, r-\delta+1, d)_q$*-code, where* $d \ge \min\{r+(p-1)\delta+1, 2(r-\delta+1)\}$.

Proof. Define the global parity check matrices $\mathbf{H}_i^{\mathrm{G}}$ and local parity check matrices $\mathbf{H}_i^{\mathrm{L}}$ for $1 \le i \le p$ as follows,

$$\mathbf{H}_i^{\mathrm{G}} = \begin{bmatrix} \mathbf{A}_{j,j} & \mathbf{B}_{1,j} & \cdots & \mathbf{B}_{j-1,j} & \mathbf{B}_{j+1,j} & \cdots & \mathbf{B}_{p,j} \\ -\mathbf{I}_r & & & \mathbf{0}_{(p-1)\delta} & & & \end{bmatrix}, \tag{2}$$

$$\mathbf{H}_i^{\mathrm{L}} = \begin{bmatrix} \mathbf{A}_{i,i} \\ \mathbf{U}_i \\ -\mathbf{I}_r \end{bmatrix}. \tag{3}$$

From conditions 2), 3), and Lemma 2, we know that all $\mathbf{H}_i^{\mathrm{G}}$, $1 \le i \le p$, are $r+(p-1)\delta$-parity matrices, and all $\mathbf{H}_i^{\mathrm{L}}$, $1 \le i \le p$, are r-parity matrices.

First we prove that $d_1 = r-\delta+1$, i.e., every sub-block has minimum Hamming distance $d_1 = r-\delta+1$. Suppose $\mathbf{c} = (\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_p)$ is a codeword, then $\mathbf{c} = \mathbf{G}$ for some message $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_p)$. Therefore for $1 \le i \le p$, where $\mathbf{c}_i = [\mathbf{m}_i, \mathbf{u}_i]$ and,

$$\begin{aligned} \mathbf{u}_i =& \mathbf{m}_i\mathbf{A}_{i,i} + \sum_{j\neq i} \mathbf{m}_j\mathbf{A}_{j,i} \\ =& \mathbf{m}_i\mathbf{A}_{i,i} + \sum_{j\neq i} \mathbf{m}_j\mathbf{B}_{j,i}X_{j,i}\mathbf{U}_i \\ =& \mathbf{m}_i\mathbf{A}_{i,i} + \mathbf{y}_i\mathbf{U}_i, \end{aligned} \tag{4}$$

where $\mathbf{y}_i = \sum_{j\neq i} \mathbf{m}_j\mathbf{B}_{j,i}\mathbf{X}_{j,i}$. For each $1 \le i \le p$, let $\tilde{\mathbf{c}}_i = [\mathbf{m}_i, \mathbf{y}_i, \mathbf{u}_i]$, then

$$\tilde{\mathbf{c}}_i\mathbf{H}_i^{\mathrm{L}} = \mathbf{m}_i\mathbf{A}_{i,i} + \mathbf{y}_i\mathbf{U}_i - \mathbf{u}_i = 0. \tag{5}$$

Since $\mathbf{H}_i^{\mathrm{L}}$ is an r-parity matrix, $w_H(\tilde{\mathbf{c}}_i) \ge r+1$ for any nonzero codeword $\mathbf{c}_i \in \mathcal{C}$. Moreover, $\mathbf{y}_i$ has length δ, which implies that $w_H(\mathbf{y}_i) \le \delta$. Therefore for any nonzero $\mathbf{c}_i$,

$$w_H(\mathbf{c}_i) = w_H(\tilde{\mathbf{c}}_i) - w_H(\mathbf{y}_i) \ge r+1-\delta. \tag{6}$$

Therefore $d_1 \ge r-\delta+1$. We next prove that $d_1 = r-\delta+1$.

Let the submatrix consisting of the first $r-\delta+1$ rows from $\mathbf{A}_{i,i}$ be $\tilde{\mathbf{A}}_{i,i}$. Then $\left[\tilde{\mathbf{A}}_{i,i}^T, \mathbf{U}_i^T\right]$ is a totally invertible matrix, thus there exists an vector $\mathbf{v}$ from its nullspace such that all entries of $\mathbf{v}$ are nonzero. Suppose $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2]$, where $\mathbf{v}_1 \in GF(q)^{r-\delta+1}$, $\mathbf{v}_2 \in GF(q)^{\delta}$, then $\mathbf{v}_1\tilde{\mathbf{A}}_{1,1} + \mathbf{v}_2\mathbf{U}_i = \mathbf{0}_r$. Choose an arbitrary j such that $j \neq i$, denote the matrix consisting of the first δ rows from $\mathbf{U}_i$ by $\mathbf{V}_i$, then $\mathbf{V}_i$ is invertible since $\mathbf{U}_i$ is totally invertible. Let $\mathbf{m}_j = \left[\mathbf{v}_2\mathbf{X}_{j,i}^{-1}\mathbf{V}_{j,i}^{-1}, \mathbf{0}_{k-r+\delta-1}\right]$, $\mathbf{y}_i = \mathbf{v}_2$, $\mathbf{m}_i = [\mathbf{v}_1, \mathbf{0}_{k-r+\delta-1}]$, and $\mathbf{m}_l = \mathbf{0}_k$ for all $1 \le l \le p$. Then $\mathbf{y}_i = \mathbf{v}_2$, $\mathbf{u}_i = \mathbf{m}_i\mathbf{A}_{i,i} + \mathbf{v}_2\mathbf{U}_i = \mathbf{v}_1\tilde{\mathbf{A}}_{i,i} + \mathbf{v}_2\mathbf{U}_i = \mathbf{0}_r$, thus $\tilde{\mathbf{c}}_i$ satisfies (5), $w_H(\tilde{\mathbf{c}}_i) = r+1$ and $w_H(\mathbf{y}_i) = \delta$. Therefore $w_H(\mathbf{c}_i) = w_H(\tilde{\mathbf{c}}_i) - w_H(\mathbf{y}_i) = \delta = r+1-\delta$ and $d_1 \le r-\delta+1$.

From the previous discussion, we know that $d_1 = r-\delta+1$.

Secondly, we prove that $d \ge \min\{r+(p-1)\delta+1, 2(r-\delta+1)\}$. Suppose $\mathbf{c} = (\mathbf{c}_1, \cdots, \mathbf{c}_p)$ is a nonzero codeword, for some $\mathbf{c} \in GF(q)^{pn}$, $\mathbf{c}_j \in GF(q)^n$, and for all $1 \le j \le p$.

TABLE I
POLYNOMIAL AND NORMAL FORMS OF $GF(2^4)$

0	0000	β^4	1100	β^8	1010	β^{12}	1111
β	0100	β^5	0110	β^9	0101	β^{13}	1011
β^2	0010	β^6	0011	β^{10}	1110	β^{14}	1001
β^3	0001	β^7	1101	β^{11}	0111	β^{15}	1000

Suppose $\exists j$, $1 \le j \le p$, such that $\mathbf{c}_j \neq 0$, and $\mathbf{c}_i = 0$ for all $1 \le i \le p$, $i \neq j$. Then $\mathbf{c}_j$ satisfies that:

$$\begin{aligned} &\mathbf{c}_j \begin{bmatrix} \mathbf{A}_{j,j} \\ -\mathbf{I}_r \end{bmatrix} = 0, \\ &\mathbf{c}_j \begin{bmatrix} \mathbf{A}_{i,j} \\ \mathbf{0}_r \end{bmatrix} = 0 \Longleftrightarrow \mathbf{c}_j \begin{bmatrix} \mathbf{B}_{i,j} \\ \mathbf{0}_\delta \end{bmatrix} = 0, \forall i \neq j. \end{aligned}$$

Therefore,

$$\mathbf{c}_j\mathbf{H}_j^{\mathrm{R}} = 0.$$

Since $\mathbf{H}_j^{\mathrm{R}}$ is an $r+(p-1)\delta$-parity matrix, $w_H(\mathbf{c}_j) \ge r+(p-1)\delta+1$. Therefore $w_H(\mathbf{c}) = w_H(\mathbf{c}_j) \ge r+(p-1)\delta+1$.

Suppose $\exists i, j$, $1 \le i < j \le p$, such that $\mathbf{c}_i \neq 0$, and $\mathbf{c}_j \neq 0$. Then $w_H(\mathbf{c}) \ge w_H(\mathbf{c}_i) + w_H(\mathbf{c}_j) \ge 2d_1 = 2(r-\delta+1)$. Therefore, $d = \min w_H(\mathbf{c}) \ge \min\{r+(p-1)\delta+1, 2(r-\delta+1)\}$. ∎

Construction 1. *Let* $a_1, \cdots, a_{mk+\delta}, b_1, \cdots, b_{mr}$ *be pairwise distinct elements in* $GF(q)$. *Define* $\mathbf{A}_{j,j}$, $\mathbf{B}_{i,j}$, $\mathbf{U}_j$, $\mathbf{X}_{i,j}$ *as follows, for all* $1 \le i, j \le p$, $i \neq j$, *in Theorem 1, then* G *is the generator matrix of an* $(p, n, k, r-\delta+1, d)_q$*-code, where* $d \ge \min\{r+(p-1)\delta+1, 2(r-\delta+1)\}$*:*

$$\begin{aligned} \mathbf{A}_{j,j} &= \mathbf{Y}((j-1)k, jk, (j-1)r, jr), \\ \mathbf{B}_{i,j} &= \mathbf{Y}((i-1)k, ik, (j-1)r, (j-1)r+\delta), \\ \mathbf{U}_j &= \mathbf{Y}(pk, pk+\delta, (j-1)r, jr), \\ \mathbf{X}_{i,j} &= \mathbf{I}_\delta, \end{aligned}$$

where the matrices $\mathbf{Y}(i_1, i_2, j_1, j_2)$ *for* $0 \le i_1 < i_2 \le pk+\delta$, $0 \le j_1 < j_2 \le pr$ *are defined as below,*

$$\begin{aligned} &\mathbf{Y}(i_1, i_2, j_1, j_2) = \\ &= \begin{bmatrix} \frac{1}{a_{i_1+1}-b_{j_1+1}} & \frac{1}{a_{i_1+1}-b_{j_1+2}} & \cdots & \frac{1}{a_{i_1+1}-b_{j_2}} \\ \frac{1}{a_{i_1+2}-b_{j_1+1}} & \frac{1}{a_{i_1+2}-b_{j_1+2}} & \cdots & \frac{1}{a_{i_1+2}-b_{j_2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{i_2}-b_{j_1+1}} & \frac{1}{a_{i_2}-b_{j_1+2}} & \cdots & \frac{1}{a_{i_2}-b_{j_2}} \end{bmatrix}. \end{aligned}$$

Proof. It follows immediately from Theorem 1 and the fact that any Cauchy matrix is totally invertible. ∎

Example 1. *Let* $q = 2^4$, $p = 2$, $r = 3$, $\delta = 1$, $k = 3$, $n = k+r = 6$. *Then* $r-\delta+1 = 3-1+1 = 3$, $r+(p-1)\delta+1 = 3+1\cdot1+1 = 5$, $pk+\delta = 2\cdot3+1 = 7$, *and* $pr = 2\cdot3 = 6$. *The extension field is* $GF(2^4)$, *choose a primitive polynomial over* $GF(2)$*:* $g(X) = X^4+X+1$. *Let* β *be the root of* $g(X)$, *then* β *is a primitive element of* $GF(2^4)$. *Let* $a_i = \beta^i$, $1 \le i \le 7$, $b_i = \beta^{7+i}$, $1 \le i \le 6$.

Then,

$$\mathbf{A}_{1,1}$$
$$= \begin{bmatrix} \frac{1}{\beta-\beta^8} & \frac{1}{\beta-\beta^9} & \frac{1}{\beta-\beta^{10}} \\ \frac{1}{\beta^2-\beta^8} & \frac{1}{\beta^2-\beta^9} & \frac{1}{\beta^2-\beta^{10}} \\ \frac{1}{\beta^3-\beta^8} & \frac{1}{\beta^3-\beta^9} & \frac{1}{\beta^3-\beta^{10}} \end{bmatrix} = \begin{bmatrix} \beta^5 & \beta^{12} & \beta^7 \\ 1 & \beta^4 & \beta^{11} \\ \beta^2 & \beta^{14} & \beta^3 \end{bmatrix},$$

$$\mathbf{A}_{2,2}$$
$$= \begin{bmatrix} \frac{1}{\beta^4-\beta^{11}} & \frac{1}{\beta^4-\beta^{12}} & \frac{1}{\beta^4-\beta^{13}} \\ \frac{1}{\beta^5-\beta^{11}} & \frac{1}{\beta^5-\beta^{12}} & \frac{1}{\beta^5-\beta^{13}} \\ \frac{1}{\beta^6-\beta^{11}} & \frac{1}{\beta^6-\beta^{12}} & \frac{1}{\beta^6-\beta^{13}} \end{bmatrix} = \begin{bmatrix} \beta^2 & \beta^9 & \beta^4 \\ \beta^{12} & \beta & \beta^8 \\ \beta^{14} & \beta^{11} & 1 \end{bmatrix},$$

$$\mathbf{B}_{1,2} = \begin{bmatrix} \frac{1}{\beta^1-\beta^{11}} \\ \frac{1}{\beta^2-\beta^{11}} \\ \frac{1}{\beta^3-\beta^{11}} \end{bmatrix} = \begin{bmatrix} \beta^9 \\ \beta^6 \\ \beta^{10} \end{bmatrix},$$

$$\mathbf{U}_2 = \begin{bmatrix} \frac{1}{\beta^7-\beta^{11}} & \frac{1}{\beta^7-\beta^{12}} & \frac{1}{\beta^7-\beta^{13}} \end{bmatrix} = \begin{bmatrix} \beta^7 & \beta^{13} & \beta^{10} \end{bmatrix},$$

$$\mathbf{B}_{2,1} = \begin{bmatrix} \frac{1}{\beta^4-\beta^8} \\ \frac{1}{\beta^5-\beta^8} \\ \frac{1}{\beta^6-\beta^8} \end{bmatrix} = \begin{bmatrix} \beta^{10} \\ \beta^{11} \\ \beta \end{bmatrix},$$

$$\mathbf{U}_1 = \begin{bmatrix} \frac{1}{\beta^7-\beta^8} & \frac{1}{\beta^7-\beta^9} & \frac{1}{\beta^7-\beta^{10}} \end{bmatrix} = \begin{bmatrix} \beta^4 & 1 & \beta^9 \end{bmatrix},$$

$$\mathbf{A}_{1,2} = \mathbf{B}_{1,2}\mathbf{U}_2 = \begin{bmatrix} \beta & \beta^7 & \beta^4 \\ \beta^{13} & \beta^4 & \beta \\ \beta^2 & \beta^8 & \beta^5 \end{bmatrix},$$

$$\mathbf{A}_{2,1} = \mathbf{B}_{2,1}\mathbf{U}_1 = \begin{bmatrix} \beta^{14} & \beta^{10} & \beta^4 \\ 1 & \beta^{11} & \beta^5 \\ \beta^5 & \beta & \beta^{10} \end{bmatrix}.$$

Then the generating matrix $\mathbf{G}$ *is defined as follows:*

$$\left[\begin{array}{ccc|ccc|ccc|ccc} 1 & 0 & 0 & \beta^5 & \beta^{12} & \beta^7 & 0 & 0 & 0 & \beta & \beta^7 & \beta^4 \\ 0 & 1 & 0 & 1 & \beta^4 & \beta^{11} & 0 & 0 & 0 & \beta^{13} & \beta^4 & \beta \\ 0 & 0 & 1 & \beta^2 & \beta^{14} & \beta^3 & 0 & 0 & 0 & \beta^2 & \beta^8 & \beta^5 \\ \hline 0 & 0 & 0 & \beta^{14} & \beta^{10} & \beta^4 & 1 & 0 & 0 & \beta^2 & \beta^9 & \beta^4 \\ 0 & 0 & 0 & 1 & \beta^{11} & \beta^5 & 0 & 1 & 0 & \beta^{12} & \beta & \beta^8 \\ 0 & 0 & 0 & \beta^5 & \beta & \beta^{10} & 0 & 0 & 1 & \beta^{14} & \beta^{11} & 1 \end{array}\right].$$

According to Construction 1, code with generator matrix being $\mathbf{G}$ *is a* $(2,6,3,3,5)_1 6$*-code.*

IV. Decoding Scheme

In Section III, a distance-preserving multi-level code based on so-called totally invertible matrices is proposed. In Construction 1, particularly, by setting the invertible matrices to be Cauchy matrices, we provide an explicit construction of such a code on $GF(q)$. In this section, we focus on an efficient decoding algorithm for the code in Construction 1.

Let $n,k,s,t,r \in \mathbb{N}$, $v = 2s+t$, $0 < r \leq v$, $n = k + r$. Suppose $\mathbf{A} \in GF(q)^{k\times v}$ is a Cauchy matrix defined in Remark 1. Matrix $\mathbf{H}$ is defined as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} \\ \mathbf{I}_r \ \mathbf{0}_{v-r} \end{bmatrix}. \tag{7}$$

According to Lemma 2, $\mathbf{H}$ is the parity check matrix of an $(n,k,2s+t+1)$-code denoted by $\mathcal{C}(\mathbf{H})$, which means that $\mathcal{C}(\mathbf{H})$ is able to correct up to s substitutions and t erasures.

In (3) and (2), we defined for the i-th block the global parity check matrices $\mathbf{H}_i^{\mathrm{G}}$ and local parity check matrices $\mathbf{H}_i^{\mathrm{L}}$, where $1 \leq i \leq p$. According to the proof of Theorem 1, all the blocks containing at most $\lfloor \frac{r-\delta}{2} \rfloor$ errors can be decoded by their local parity check matrices, otherwise they can still be decoded by their global parity check matrices if only one of them contains errors and the number of errors does not exceed $\lfloor \frac{r+(p-1)\delta}{2} \rfloor$. Notice that $\{\mathbf{H}_i^{\mathrm{G}}\}_{i=1}^p$, $\{\mathbf{H}_i^{\mathrm{L}}\}_{i=1}^p$ are all special cases of (7). We therefore first find the general decoding algorithm for $\mathcal{C}(\mathbf{H})$ in Section IV-A, based on which the local decoding algorithm and the global decoding algorithm are provided in Section IV-B and Section IV-C, respectively.

A. Decoding Algorithm for Codes Based on Cauchy Matrices

In this subsection, we focus on the general decoding algorithm of $\mathcal{C}(\mathbf{H})$ for $\mathbf{H}$ defined in (7). The main algorithm is presented in Algorithm 1 based on Theorem 2, where we define the **error location polynomial** $e(X)$ and prove that any separable polynomial that satisfies (9) provides the full information of errors. Theorem 2 is proved based on Lemma 3 and Lemma 4. In this section, $*$ refers to erasures.

Theorem 2. *Let* $\mathbf{c} \in \mathcal{C}(\mathbf{H})$ *and* $\mathbf{c}' \in GF(q)^n$ *be the transmitted codeword and the codeword received after an* s*-deletion* t*-erasure channel, respectively. Denote the set of locations where there is an erasure by* E*, i.e.,* $E = \{i : \mathbf{c}'_i = *\}$*, then* $E \subset [k]$ *and* $|E| \leq t$*. Suppose* $|E| = t$ *without loss of generality. Let* D *be the set of positions where there is a substitution, i.e.,* $D = \{i : \mathbf{c}'_i \neq \mathbf{c}_i, \mathbf{c}'_i \neq *\}$*, then* $D \subset [n] \setminus E$*,* $|D| \leq s$*. Let* $I = D \cap [k]$*,* $J = D \setminus I$*. Define the* ***error location polynomial*** $e(X)$ *of* $\mathbf{c}$ *and* $\mathbf{c}'$*,* $e(X) \in GF(q)[X]$*, as follows:*

$$e(X) \triangleq \prod_{i\in I}(X - a_i) \prod_{j\in J}(X - b_{j-k}). \tag{8}$$

Let $W_0 \subset [2s+t]$*,* $|W_0| = s+t$*. Suppose* $W_i = W_0 \cup \{w_i\}$*,* $1 \leq i \leq s$*,* $\{w_i\}_{i=1}^s = [v] \setminus W_0$*. Then for any separable* $f(X) \in GF(q)[X]$ *that has degree* s*,* $f(X)$ *is a solution to the following equations for all* $\{W_i\}_{i=1}^s$ *if and only if* $e(X) | f(X)$*.*

$$\sum_{w\in W} S_w \frac{\prod_{i\in E}(b_w - a_i)}{\prod_{i\in W\setminus\{w\}}(b_w - b_i)} f(b_w) = 0. \tag{9}$$

Proof. ($\Longrightarrow$) If $e(X)|f(X)$, let $h(X) = \prod_{i\in E}(X - a_i) \in GF(q)[X]$, $e'(X) = h(X)e(X)$, $f'(X) = h(X)f(X)$. Then $f'(b_w) = \prod_{i\in E}(b_w - a_i) f(b_w)$ and $e'(X)|f'(X)$. Therefore from Lemma 3, (9) is satisfied.

($\Longleftarrow$) If (9) is satisfied, denote the set of roots of $f(X)$ in the algebraic closure K of $GF(q)$ by R. Let $B = \{b_i\}_{i=1}^{2s+t}$. Then $R \cap B = \{b_j, j \in J'\}$, $R \setminus B = \{x_i, 1 \leq i \leq l\} \subset K$ for some $J' \subset [r]$, $l + |J'| = s$. Suppose $\{\mathbf{r}_i\}_{i\in I}$ are defined as follows:

$$\mathbf{r}_i = \begin{bmatrix} \frac{1}{a_i - b_1} & \frac{1}{a_i - b_2} & \cdots & \frac{1}{a_i - b_{2s+t}} \end{bmatrix}. \tag{10}$$

Then we know that $\mathbf{S} = \sum_{i\in I\cup E} e_i\mathbf{r}_i + \sum_{j\in J} e_{j+k}\mathbf{e}_j$ for some nonzero $e_i \in GF(q)$. Suppose $v = 2s+t$, $\{\mathbf{h}_i\}_{i=1}^{l}$ are defined as they are in Lemma 4, then we know that $\mathbf{S} \in \mathrm{Span}\{\mathbf{r}_i, i \in E, \mathbf{h}_i, i \in [l], \mathbf{e}_j, j \in J'\}$. Let $T = \{\mathbf{r}_i, i \in I \cup E, \mathbf{h}_i, i \in [l], \mathbf{e}_j, j \in J' \cup J\}$. Since $|T| = |I \cup E| + l + |J' \cup J| \leq |I| + |E| + l + |J| + |J'| = |E| + (|I| + |J|) + (|J'| + l) \leq t + s + s = 2s + t$, the elements of T are linearly independent from Lemma 2. Therefore $T = \{\mathbf{0}_{2s+t}\}$, which means that the representation of $\mathbf{S}$ as linear combination of elements in T is unique. Therefore $\sum_{i\in I\cup E} e_i\mathbf{r}_i + \sum_{j\in J} e_{j+k}\mathbf{e}_j = \mathbf{S} \in \mathrm{Span}\{\mathbf{r}_i, i \in E, \mathbf{h}_i, i \in [l], \mathbf{e}_j, j \in J'\}$ implies that $\{a_i, i \in I\} \subseteq \{x_i, i \in [l]\}$, and $J \subseteq J'$, thus $e(X)|f(X)$. ■

Algorithm 1 Decoding Algorithm for Code $\mathcal{C}(\mathbf{H})$

Input:
$\mathbf{c}'$: received word;
s: substitution correction limit;
t: erasure correction limit;
$\mathbf{H}$: parity-check matrix in (7);

Output:
$\hat{\mathbf{c}}$: estimation of the transmitted codeword;

Find the set E of positions where erasure happens in $\mathbf{c}'$;
for $i \in E$ **do** $(\mathbf{c}')_i \leftarrow 0$;
end for
$\mathbf{S} \leftarrow \mathbf{c}'\mathbf{H}$;
Find $W_0 \subset [2s+t]$, $|W_0| = s + |E|$. Let $\{w_i\}_{i=1}^{s+t-|E|} = [v] \setminus W_0$;
$\mathbf{F} \leftarrow \mathbf{0}^{(s+t-|E|)\times s}$, $\mathbf{u} \leftarrow \mathbf{0}^{s+t-|E|}$;
for $1 \leq i \leq s+t-|E|$ **do** $W \leftarrow W_0 \cup \{w_i\}$;
  for $w \in W$ **do** $z_w \leftarrow S_w \frac{\prod_{i\in E}(b_w - a_i)}{\prod_{i\in W\setminus\{w\}}(b_w - b_i)}$;
  end for
  $(\mathbf{u})_i \leftarrow -\sum_{w\in W} z_w b_w^s$;
  for $1 \leq j \leq s$ **do** $(\mathbf{F})_{i,j} \leftarrow \sum_{w\in W} z_w b_w^{s-j}$;
  end for
end for
if $\mathbf{F}\boldsymbol{\sigma} = \mathbf{u}$ has an unique solution $\boldsymbol{\sigma}$ **then**
  Run Algorithm 2 for $\boldsymbol{\sigma}$;
  return $\hat{\mathbf{c}}$;
else if $\mathbf{F}\boldsymbol{\sigma} = \mathbf{u}$ has at least two solutions $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2$ **then**
  Let $K \subset GF(q) \setminus \{0\}$, $|K| = 2s+1$;
  for $k \in K$ **do** $\boldsymbol{\sigma} \leftarrow \boldsymbol{\sigma}_1 + k(\boldsymbol{\sigma}_2 - \boldsymbol{\sigma}_1)$;
    Run Algorithm 2 for $\boldsymbol{\sigma}$, obtain p, $\hat{\mathbf{c}}$;
    if $p = True$ **then**
      return $\hat{\mathbf{c}}$;
    end if
  end for
end if

Lemma 3. *Let $n, u, v \in \mathbb{N}$, $0 < u < \min\{n, v\}$. Define $\{\mathbf{h}_i\}_{i=1}^{n}$, $\{\mathbf{e}_j\}_{j=1}^{v}$ on $GF(q)$ as follows:*

$$\mathbf{h}_i = \left[\begin{array}{cccc} \frac{1}{a_i - b_1} & \frac{1}{a_i - b_2} & \cdots & \frac{1}{a_i - b_v} \end{array}\right]. \tag{11}$$

$$(\mathbf{e}_j)_i = \begin{cases} 1, & i = j, \\ 0, & otherwise. \end{cases} \tag{12}$$

Suppose $\mathbf{S} \in GF(q)$, $\mathbf{S} = (S_1, \cdots, S_v)$. If for some $I \subset [n]$, $J \subset [v]$, $|I| + |J| \leq u$, $\mathbf{S} \in \mathrm{Span}\{\mathbf{h}_i, i \in I, \mathbf{e}_j, j \in J\}$, define $f(X) \in GF(q)[X]$ as follows:

$$f(X) \triangleq \prod_{i\in I}(X - a_i) \prod_{j\in J}(X - b_j). \tag{13}$$

Then the following equations are satisfied for any $W \subset [v]$, $g(X) \in GF(q)$ such that $|W| = u + 1$, $\deg(g) = u$ and $f(X)|g(X)$:

$$\sum_{w\in W} S_w \frac{g(b_w)}{\prod_{i\in W\setminus\{w\}}(b_w - b_i)} = 0. \tag{14}$$

Proof. Let $k_1 = |I|$, $k_2 = |J \setminus W|$, $k_3 = u - |I| - |J|$. Suppose $I = \{i_1, \cdots, i_{k_1}\}$, $J \setminus W = \{j_1, \cdots, j_{k_2}\}$. Let $k = k_1 + k_2 + k_3 + 1$. Then $k = |I| + |J \setminus W| + u + 1 - |I| - |J| = |I| + |J \setminus W| + |W| - |I| - |J| = |W \setminus J|$. Suppose $W \setminus J = \{w_1, \cdots, w_k\}$.

Denote the algebraic closure of $GF(q)$ by K, then $g(X)$ splits in K, given that $g(X)|f(X)$ on $GF(q)$, there exists $x_1, \cdots, x_{k_3} \in K$ such that:

$$g(X) = \prod_{i\in I}(X - a_i) \prod_{j\in J}(X - b_j) \prod_{l=1}^{k_3}(X - x_l). \tag{15}$$

Define $\{\tilde{\mathbf{h}}_{W,i}\}_{i=1}^{n}$, $\tilde{\mathbf{S}}_W$ as follows:

$$\tilde{\mathbf{h}}_{W,i} = \left[\begin{array}{cccc} \frac{1}{a_i - b_{w_1}} & \frac{1}{a_i - b_{w_2}} & \cdots & \frac{1}{a_i - b_{w_k}} \end{array}\right]. \tag{16}$$

$$\tilde{\mathbf{S}}_W = \left[\begin{array}{cccc} S_{w_1} & S_{w_2} & \cdots & S_{w_k} \end{array}\right]. \tag{17}$$

The condition $\mathbf{S} \in \mathrm{Span}\{\mathbf{h}_i, i \in I, \mathbf{e}_j, j \in J\}$ implies that $\tilde{\mathbf{S}}_W \in \mathrm{Span}\{\tilde{\mathbf{h}}_{W,i}, i \in I\}$. Let $\{X_1, \cdots, X_{k-1}\} = \{a_i, i \in I, b_j, j \in J \setminus W, x_l, l \in [k_3]\}$. Define $\mathbf{H}_W$ as follows:

$$\mathbf{H}_W = \left[\begin{array}{cccc} \frac{1}{X_1 - b_{w_1}} & \frac{1}{X_1 - b_{w_2}} & \cdots & \frac{1}{X_1 - b_{w_k}} \\ \frac{2}{X_2 - b_{w_1}} & \frac{2}{X_2 - b_{w_2}} & \cdots & \frac{1}{X_2 - b_{w_k}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{X_{k-1} - b_{w_1}} & \frac{1}{X_{k-1} - b_{w_2}} & \cdots & \frac{1}{X_{k-1} - b_{w_k}} \\ S_{w_1} & S_{w_2} & \cdots & S_{w_k} \end{array}\right]. \tag{18}$$

Then $\mathbf{H}_W$ is singular, i.e., $\det(\mathbf{H}_W)=0$. Expand $\det(\mathbf{H}_W)$ along the last row, we obtain:

$$\begin{aligned}
&0\\
=&\sum_{l=1}^{k}(-1)^l S_{w_l}\frac{\prod\limits_{1\le i<j<k}(X_i-X_j)\prod\limits_{1\le i<j\le k,i,j\ne l}(b_{w_j}-b_{w_i})}{\prod\limits_{1\le i<k}\prod\limits_{1\le j\le k,j\ne l}(X_i-b_{w_j})}\\
=&\frac{\prod\limits_{1\le i<j<k}(X_i-X_j)\prod\limits_{1\le i<j\le k}(b_{w_j}-b_{w_i})}{\prod\limits_{1\le i<k}\prod\limits_{1\le j\le k}(X_i-b_{w_j})}\\
&\sum_{l=1}^{k}S_{w_l}\frac{\prod\limits_{1\le i<k}(b_{w_l}-X_i)}{\prod\limits_{1\le i\le k,i\ne l}(b_{w_l}-b_{w_i})}.
\end{aligned}\tag{19}$$

Therefore

$$\begin{aligned}
&0\\
=&\sum_{l=1}^{k}S_{w_l}\frac{\prod\limits_{1\le i<k}(b_{w_l}-X_i)}{\prod\limits_{1\le i\le k,i\ne l}(b_{w_l}-b_{w_i})}\\
=&\sum_{w\in W\setminus J}S_w\frac{g(b_w)}{\prod\limits_{w'\in W\setminus J,w'\ne w}(b_w-b_{w'})\prod\limits_{w'\in W\cap J}(b_w-b_{w'})}\\
=&\sum_{w\in W}S_w\frac{g(b_w)}{\prod\limits_{w'\in W,w'\ne w}(b_w-b_{w'})}.
\end{aligned}\tag{20}$$

The theorem is proved. ■

Lemma 4. *Let K be the algebraic closure of $GF(q)$. In Lemma 3, let $W_0\subset[v]$, $|W_0|=u$. Suppose $[v]\setminus W_0=\{l_1,\cdots,l_{v-u}\}$, let $W_i=W_0\cup\{l_i\}$, $1\le i\le v-u$. Suppose $g(X)\in GF(q)[X]$, $\deg(g)=u$, is separable and has roots $x_1,\cdots,x_k\in K$, and b_j, $j\in J$ for some $0\le k\le u$, $x_1,\cdots,x_k\notin\{b_i,i\in[v]\}$, $J\subseteq[v]$, $k+|J|=u$. Define $\{\mathbf{h}_i\}_{i=1}^k$ as follows:*

$$\mathbf{h}_i=\left[\begin{array}{cccc}\frac{1}{x_i-b_1} & \frac{1}{x_i-b_2} & \cdots & \frac{1}{x_i-b_v}\end{array}\right].\tag{21}$$

If (14) is true for all $W=W_i$, $1\le i\le v-u$, for f, then $\mathbf{S}\in\mathrm{Span}\{\mathbf{h}_i,i\in[k],\mathbf{e}_j,j\in J\}$ over K.

Proof. Let $|J\setminus W_0|=m$, suppose $J\setminus W_0=\{j_1,j_2,\cdots,j_m\}$. Then $|W_0\setminus J|=|J\setminus W_0|+|W_0|-|J|=m+u-|J|=k+m$, suppose $W_0\setminus J=\{w_1,\cdots,w_{k+m}\}$. Since $x_1,\cdots,x_k$ are pairwise distinct, (14)$\Longleftrightarrow$(20) $\Longrightarrow$(19)$\Longleftrightarrow\mathbf{H}_{W_i}$ defined in (18) are all singular. We know that for all $1\le i\le v-u$ such that $l_i\notin J$, $W_i\setminus J=\{W_0\setminus J\}\cup\{l_i\}$, $J\setminus W_i=J\setminus W_0$. Therefore for $1\le i\le v-u$ such that $l_i\notin J$ the following matrices are singular:

$$\mathbf{H}_{W_i}=\left[\begin{array}{cccc}\frac{1}{x_1-b_{w_1}} & \cdots & \frac{1}{x_1-b_{w_{k+m}}} & \frac{1}{x_1-b_{l_i}}\\ \vdots & \ddots & \vdots & \vdots\\ \frac{1}{x_k-b_{w_1}} & \cdots & \frac{1}{x_k-b_{w_{k+m}}} & \frac{1}{x_k-b_{l_i}}\\ \frac{1}{b_{j_1}-b_{w_1}} & \cdots & \frac{1}{b_{j_1}-b_{w_{k+m}}} & \frac{1}{b_{j_1}-b_{l_i}}\\ \vdots & \ddots & \vdots & \vdots\\ \frac{1}{b_{j_p}-b_{w_1}} & \cdots & \frac{1}{b_{j_p}-b_{w_{k+m}}} & \frac{1}{b_{j_p}-b_{l_i}}\\ S_{w_1} & \cdots & S_{w_{k+m}} & S_{l_i}\end{array}\right].\tag{22}$$

Notice that the first $k+m$ columns of $\mathbf{H}_{W_i}$, $1\le v-u$ are the same, denote the matrix consisting of the these columns by $\mathbf{H}_0$.

Since for each $1\le i\le u$, if $l_i\in J$, then $W_i\setminus J=\{w_1,\cdots,w_{k+m}\}$, $J\setminus W_i=\{j_1,\cdots,j_m\}\setminus\{l_i\}$. Therefore for $1\le i\le u$ such that $l_i\in J$, if $l_i=j_s$, $\mathbf{H}_{W_i}$ is obtained from deleting the $(k+s)$-th row from $\mathbf{H}_0$. Denote the i-th row of $\mathbf{H}_0$ by $\mathbf{r}_i$, $1\le i\le k+m$, and the last row by $\mathbf{S}_0$. We know from the nonsigularity of $\mathbf{H}_{W_i}$, $1\le i\le v-u$ that $\mathbf{S}_0\in\mathrm{Span}\{\mathbf{r}_i,1\le i\le k+m\}$, and $\mathbf{S}_0\in\mathrm{Span}\{\mathbf{r}_i,1\le i\le k+m,i\ne k+j\}$, for all $1\le j\le m$. Since $\mathbf{r}_i$, $1\le i\le k+m$, are linearly independent over K, the representation of $\mathbf{S}_0$ as linear combination of them is unique. Therefore $\mathbf{S}_0\in\bigcap\limits_{1\le i\le q}\mathrm{Span}\{\mathbf{r}_i,1\le i\le k+m,i\ne k+j\}=\mathrm{Span}\{\mathbf{r}_i,1\le i\le k\}$, and we suppose $\mathbf{S}_0=\sum\limits_{i=1}^{k}e_i\mathbf{r}_i$ is the unique representation, for some $\{e_i\}_{i=1}^k\subset K$. We know that $\mathbf{e}=(e_1,\cdots,e_k,\mathbf{0}_m,-1)$ satisfies that $\mathbf{e}\mathbf{H}_{W_i}=\mathbf{0}$, for all $1\le i\le v-u$ such that $l_i\notin J$, which implies that for all $j\in[v]\setminus J$,

$$\sum_{i=1}^{k}e_i\frac{1}{x_i-b_j}=S_j.\tag{23}$$

Therefore,

$$\mathbf{S}=\sum_{i=1}^{k}e_i\mathbf{h}_i+\sum_{j\in J}\left(S_j-\sum_{i=1}^{k}e_i\frac{1}{x_i-b_j}\right)\mathbf{e}_j.\tag{24}$$

The lemma is proved. ■

Suppose $f(X)=X^n+\sigma_1X^{n-1}+\cdots+\sigma_n$ in Theorem 2. Then the equations in (9) become s linear equations of $\{\sigma_i\}_{i=1}^s$ if we expand $f(b_w)$ by the coefficients of $f(X)$. Theorem 2 affirms that even though we are not able to ensure the uniqueness of $f(X)$ that satisfies (9), the roots of $f(X)$ that are in $\{a_i\}_{i=1}$ and $\{b_j\}_{j=1}$ always contain all the error locations as long as $f(X)$ is separable.

Algorithm 2 computes the transmitted codeword from the received codeword for any given polynomial that satisfies (9) and report error if the polynomial is an invalid one. According to Theorem 2, Algorithm 2 decodes the transmitted codeword from any separable solution. The remaining issue is to find a separable solution to (9) in polynomial complexity. The following Lemma 5 provides such a method based on the fact

Algorithm 2 Decoding Algorithm for Code $\mathcal{C}(\mathbf{H})$ Given $\boldsymbol{\sigma}$

Input:

$\mathbf{c}'$: received word with erasures replaced by 0;
E: the set of erasure positions;
$\mathbf{S}$: the syndrome;
$\boldsymbol{\sigma}$: coefficients of a polynomial satisfying (9);
$\mathbf{H}$: parity-check matrix in (7);

Output:

p: indicator whether the $\boldsymbol{\sigma}$ is correct;
$\hat{\mathbf{c}}$: estimation of the transmitted codeword;

$\hat{\mathbf{c}} \leftarrow \mathbf{c}'$, $I \leftarrow E$, $J \leftarrow \emptyset$, $f(X) \leftarrow X^s + \sum_{i=1}^{s} (\boldsymbol{\sigma})_i X^{n-i}$;
for $1 \le i \le mk + \delta$ **do**
  if $f(a_i) = 0$ **then** $I \leftarrow I \cup \{i\}$;
  end if
end for
for $1 \le i \le mr$ **do**
  if $f(b_j) = 0$ **then** $J \leftarrow J \cup \{j\}$;
  end if
end for
for $1 \le i \le I \cup E$ **do** $\mathbf{h}_i \leftarrow \left[\frac{1}{a_i - b_1}, \frac{1}{a_i - b_2}, \cdots, \frac{1}{a_i - b_v}\right]$;
end for
if Find e_i, $i \in I \cup E$, and e_{k+j}, $j \in J$, such that $\sum_{i \in I \cup E} e_i \mathbf{h}_i + \sum_{j \in J} e_{k+j} \mathbf{e}_j = \mathbf{S}$ **then** $p \leftarrow True$;
  for $i \in I \cup E$ **do** $(\hat{\mathbf{c}})_i \leftarrow (\hat{\mathbf{c}})_i - e_i$;
  end for
  for $j \in J$ **do** $(\hat{\mathbf{c}})_{k+j} \leftarrow (\hat{\mathbf{c}})_i - e_{k+j}$;
  end for
else $p \leftarrow False$;
end if
return p, $\hat{\mathbf{c}}$;

that a polynomial on a field F is separable if and only if its discriminant is nonzero.

Lemma 5. *Let $n \in \mathbb{N}$, $\mathbf{u} = (u_1, \cdots, u_n), \mathbf{v} = (v_1, \cdots, v_n) \in GF(q)^n$. Suppose $k_1, k_2, \cdots, k_{2n+1}$ are pairwise distinct elements in $GF(q)$. Define polynomial $f(X; \boldsymbol{\sigma})$ for any $\boldsymbol{\sigma} = (\sigma_1, \cdots, \sigma_n) \in GF(q)^n$ as follows:*

$$f(X; \boldsymbol{\sigma}) \triangleq X^n + \sum\nolimits_{j=1}^{n} \sigma_i X^{n-i}. \tag{25}$$

Let $\mathbf{w}_i = \mathbf{u} + k_i \mathbf{v}$, for all $1 \le i \le 2n+1$. Then at least one element of $\{f(X; \mathbf{w}_i)\}_{i=1}^{2n+1}$ is separable.

Proof. We know that the discriminant of $f(X; \boldsymbol{\sigma})$ is a function of $\sigma_1, \cdots, \sigma_n$, thus we can denote it by $\Delta(\boldsymbol{\sigma})$. Define $g(K) \in GF(q)[K]$ as follows:

$$g(K) \triangleq \Delta(\mathbf{u} + K\mathbf{v}). \tag{26}$$

Then $\deg(g) \le 2n$, which means that $g(K)$ has at most $2n$ roots. Therefore there exists $i \in [2n+1]$ such that $g(k_i) \neq 0$, which is equivalent to $\Delta(\mathbf{w}_i) \neq 0$. Therefore $f(X; \mathbf{w}_i)$ is separable. ■

B. Local Decoding Algorithm

Based on the decoding algorithm for s-substitution t-erasure codes with parity check matrix specified by (7), we provide the local decoding algorithm in Algorithm 3.

Algorithm 3 Local Decoding Algorithm for Construction 1

Input:

$\mathbf{c}'_i$: received word in the i-th block;
$\mathbf{H}$: parity-check matrix in (7);

Output:

$\hat{\mathbf{m}}_i$: estimation of the message in the i-th block;

$s \leftarrow \lfloor \frac{r-\delta}{2} \rfloor$, $t \leftarrow \delta$;
$\mathbf{m}'_i \leftarrow (\mathbf{c}'_i)[1:k]$, $\mathbf{u}'_i \leftarrow (\mathbf{c}'_i)[k+1:n]$;
$\mathbf{c}' \leftarrow \left[\mathbf{m}'_i, *^{\delta}, \mathbf{u}'_i\right]$, where $*$ refers to erasure;
Run Algorithm 1 for $\mathcal{C}(\mathbf{H}_i^{\mathrm{L}})$ and obtain $\hat{\mathbf{c}}$;
$\hat{\mathbf{m}}_i \leftarrow \hat{\mathbf{c}}[1:k]$;
return $\hat{\mathbf{m}}_i$;

Example 2. *(Local Decoding) Use the code constructed in Example 1. Let $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2)$, where $\mathbf{m}_1 = (\beta, 0, \beta^4)$, $\mathbf{m}_2 = (0, 1, 0)$. Therefore $\mathbf{c}_1 = (\beta, 0, \beta^4, 1, \beta^7, \beta^3)$. Suppose $\mathbf{c}'_1 = (\beta, \beta^2, \beta^4, 1, \beta^7, \beta^3)$, then $d_H(\mathbf{c}_1, \mathbf{c}'_1) = 1$ and $\mathbf{c}_1$ can be locally decoded. We let $\tilde{\mathbf{c}}'_1 = (\beta, \beta^2, \beta^4, 0, 1, \beta^7, \beta^3)$.*

We know that $(S_1, S_2, S_3) = \mathbf{c}'_1 \mathbf{H}_1^{\mathrm{L}} = (\beta, \beta^2, \beta^4)\mathbf{A}_{1,1} + 0 \cdot \mathbf{U}_1 + (1, \beta^7, \beta^3) = (\beta^8, \beta, \beta^7)$. Our objective is to find $X \in \{\beta, \beta^2, \beta^3\}$ such that the following equation is satisfied,

$$\begin{aligned} 0 =& \beta^8 \frac{(\beta^8 - \beta^7)(\beta^8 - X)}{(\beta^8 - \beta^9)(\beta^8 - \beta^{10})} + \beta \frac{(\beta^9 - \beta^7)(\beta^9 - X)}{(\beta^9 - \beta^8)(\beta^9 - \beta^{10})} \\ &+ \beta^7 \frac{(\beta^{10} - \beta^7)(\beta^{10} - X)}{(\beta^{10} - \beta^8)(\beta^{10} - \beta^8)}, \end{aligned}$$

which is equivalent to the following equation when multiplying with $(\beta^8 - \beta^9)(\beta^9 - \beta^{10})(\beta^{10} - \beta^8)$:

$$0 = \beta^2(\beta^8 - X) + \beta^2(\beta^9 - X) + \beta^{10}(\beta^{10} - X).$$

Therefore $X = \frac{\beta^{10} + \beta^{11} + \beta^5}{\beta^2 + \beta^2 + \beta^{10}} = \frac{\beta^{12}}{\beta^{10}} = \beta^2$. Then the error vector $\tilde{\mathbf{e}} = (0, \tilde{e}_2, 0, \tilde{e}_4, 0, 0, 0)$, and

$$\begin{bmatrix} \tilde{e}_2 & \tilde{e}_4 \end{bmatrix} \begin{bmatrix} 1 & \beta^4 & \beta^{11} \\ \beta^4 & 1 & \beta^9 \end{bmatrix} = \begin{bmatrix} \beta^8 & \beta & \beta^7 \end{bmatrix}.$$

And we obtain $(\tilde{e}_2, \tilde{e}_4) = (\beta^2, \beta^{11})$. Therefore $\hat{\mathbf{c}}_1 = (0, \beta^2, 0, 0, 0, 0) + \mathbf{c}'_1 = (\beta, 0, \beta^4, 1, \beta^7, \beta^3)$.

C. Global Decoding Algorithm

In this subsection, we provide the global decoding algorithm in Algorithm 4 for the code specified in Construction 1. The parity check matrix $\mathbf{H}$ in Algorithm 4 is defined as follows:

$$\mathbf{H} = \left[\begin{array}{c|c|c|c} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} & \cdots & \mathbf{A}_{1,m} \\ -\mathbf{I}_r & \mathbf{0}_r & \cdots & \mathbf{0}_r \\ \hline \mathbf{A}_{2,1} & \mathbf{A}_{2,2} & \cdots & \mathbf{A}_{2,m} \\ \mathbf{0}_r & -\mathbf{I}_r & \cdots & \mathbf{0}_r \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline \mathbf{A}_{m,1} & \mathbf{A}_{m,2} & \cdots & \mathbf{A}_{m,m} \\ \mathbf{0}_r & \mathbf{0}_r & \cdots & -\mathbf{I}_r \end{array}\right]. \tag{27}$$

Algorithm 4 Global Decoding Algorithm for Construction 1

Input:

$\mathbf{c}'$: received word;

i: the index of the message needs to be decoded;

$\mathbf{H}$: parity-check matrix in (27);

Output:

$\hat{\mathbf{m}}_i$: estimation of the message in the i-th block;

$s \leftarrow \lfloor \frac{r+(m-1)\delta}{2} \rfloor$, $t \leftarrow 0$;

$\tilde{\mathbf{S}} \leftarrow \mathbf{c}'\mathbf{H}$, $\mathbf{S}_i \leftarrow \tilde{\mathbf{S}}[1:r]$;

for $j \in [m] \setminus \{i\}$ **do**

  Solve $\mathbf{S}_j\mathbf{U}_j = \tilde{\mathbf{S}}[(j-1)r+1 : jr]$;

end for

$\mathbf{S} \leftarrow [\mathbf{S}_i, \mathbf{S}_1, \cdots, \mathbf{S}_{i-1}, \mathbf{S}_{i+1}, \cdots, \mathbf{S}_{\mathbf{m}}]$, $\mathbf{c}' \leftarrow \mathbf{c}'_i$;

Run Algorithm 1 for $\mathcal{C}(\mathbf{H}_i^{\mathrm{G}})$ and obtain $\hat{\mathbf{c}}$;

$\hat{\mathbf{m}}_i \leftarrow \hat{\mathbf{c}}[1:k]$;

return $\hat{\mathbf{m}}_i$;

Example 3. *(Global Decoding) Use the code constructed in Example 1. Let* $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2)$*, where* $\mathbf{m}_1 = (\beta, 0, \beta^4)$*,* $\mathbf{m}_2 = (0, 1, 0)$*. Therefore* $\mathbf{c} = (\mathbf{c}_1, \mathbf{c}_2)$*, where* $\mathbf{c}_1 = (\beta, 0, \beta^4, 1, \beta^7, \beta^3)$*,* $\mathbf{c}_2 = (0, 1, 0, \beta^{10}, \beta^3, \beta^{14})$*. Suppose* $\mathbf{c}' = (\mathbf{c}'_1, \mathbf{c}_2)$*, where* $\mathbf{c}'_1 = (\beta, 1, \beta^4, 1, \beta^9, \beta^3)$*. Suppose* $\mathbf{e} = \mathbf{c}' - \mathbf{c} = (\mathbf{e}_1, \mathbf{0}_6)$*. We know that* $\tilde{\mathbf{S}} = \mathbf{e}\mathbf{H} = \mathbf{c}'\mathbf{H} = (1, \beta, \beta^{11}, \beta^{13}, \beta^4, \beta)$*. Since* $(\beta^{13}, \beta^4, \beta) = \beta^6(\beta^7, \beta^{13}, \beta^{10}) = \beta^6\mathbf{U}_2$*,* $\mathbf{S} = (1, \beta, \beta^{11}, \beta^6)$*. And we know that* $\mathbf{S} = \mathbf{e}_1\mathbf{H}_1^{\mathrm{G}}$.

Our objective is to find $X_1, X_2 \in \{\beta, \beta^2, \beta^3, \beta^9, \beta^{10}, \beta^{11}\}$ *such that the following equation is satisfied,*

$$\begin{aligned} 0 =& 1\frac{(\beta^8 - X_1)(\beta^8 - X_2)}{(\beta^8 - \beta^9)(\beta^8 - \beta^{10})} + \beta\frac{(\beta^9 - X_1)(\beta^9 - X_2)}{(\beta^9 - \beta^8)(\beta^9 - \beta^{10})} \\ &+ \beta^{11}\frac{(\beta^{10} - X_1)(\beta^{10} - X_2)}{(\beta^{10} - \beta^8)(\beta^{10} - \beta^8)}, \\ 0 =& \beta\frac{(\beta^9 - X_1)(\beta^9 - X_2)}{(\beta^9 - \beta^{10})(\beta^9 - \beta^{11})} + \beta^{11}\frac{(\beta^{10} - X_1)(\beta^{10} - X_2)}{(\beta^{10} - \beta^9)(\beta^{10} - \beta^{11})} \\ &+ \beta^6\frac{(\beta^{11} - X_1)(\beta^{11} - X_2)}{(\beta^{11} - \beta^9)(\beta^{11} - \beta^9)}. \end{aligned}$$

Let $f(X) = (X - X_1)(X - X_2) = X^2 + \sigma_1 X + \sigma_2$*. Therefore we have,*

$$\begin{bmatrix} \beta^9 & \beta^6 \\ \beta^{11} & \beta^{12} \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \end{bmatrix} = \begin{bmatrix} \beta \\ \beta^{11} \end{bmatrix}. \tag{28}$$

The solution is $(\sigma_1, \sigma_2) = (\beta^{11}, \beta^{11})$*, thus* $f(X) = X^2 + \beta^{11}X + \beta^{11} = (X - \beta^2)(X - \beta^9)$*. Therefore* $\{X_1, X_2\} = \{\beta^2, \beta^9\} = \{a_2, b_2\}$*,* $\mathbf{e}_1 = (0, e_1, 0, 0, e_2, 0)$*, and* $e_1(1, \beta^4, \beta^{11}, \beta^6) + e_2(0, 1, 0, 0) = (1, \beta, \beta^{11}, \beta^6)$*. We know that* $e_1 = 1$*,* $e_2 = \beta - \beta^4 = 1$*. Therefore* $\mathbf{e} = (0, 1, 0, 0, 1, 0)$*,* $\hat{\mathbf{c}}_1 = \mathbf{c}'_1 - \mathbf{e} = (\beta, 0, \beta^4, 1, \beta^7, \beta^3)$.

V. Conclusion

Multi-level codes has been intensively studied for their capability of providing multi-level access, which is useful in distributed storage. In this paper, we studied multi-level codes that also possess distance-preserving property, which is essential for efficient rewriting in frequently updated distributed storage systems and has never been discussed before. We provided a general construction of a code that meet this requirement, based on totally invertible matrices. Using this construction as a prototype, we paid special attention to the class of codes based on Cauchy matrices due to their nice structure and their resultant potentials to be efficiently decoded. Finally, we presented an efficient decoding algorithm with a polynomial complexity that is independent of the alphabet size.

Acknowledgment

This work has received funding from DYF, NSF under the grant CCF-BSF 1718389, and from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement n. PCOFUND-GA-2013-609102.

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, Sept 2010.

[2] J. S. Plank and M. Blaum, "Sector-disk (SD) erasure codes for mixed failure modes in RAID systems," *ACM Transactions on Storage (TOS)*, vol. 10, no. 1, p. 4, 2014.

[3] M. Hassner, K. Abdel-Ghaffar, A. Patel, R. Koetter, and B. Trager, "Integrated interleaving-a novel ECC architecture," *IEEE Transactions on Magnetics*, vol. 37, no. 2, pp. 773–775, 2001.

[4] Y. Cassuto, E. Hemo, S. Puchinger, and M. Bossert, "Multi-block interleaved codes for local and global read access," in *Proc. IEEE Int. Symp. Inf. Theory*, 2017, pp. 1758–1762.

[5] I. Tamo and A. Barg, "A family of optimal locally recoverable codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4661–4676, 2014.

[6] G. M. Church, Y. Gao, and S. Kosuri, "Next-generation digital information storage in DNA," *Science*, no. 6012, 2012.

[7] L. Organick, S. D. Ang, Y.-J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, B. Nguyen *et al.*, "Random access in large-scale DNA data storage," *Nature biotechnology*, vol. 36, no. 3, p. 242, Mar. 2018.

[8] S. H. T. Yazdi, R. Gabrys, and O. Milenkovic, "Portable and error-free DNA-based data storage," *Scientific reports*, vol. 7, no. 1, p. 5011, 2017.

[9] J. Han and L. A. Lastras-Montano, "Reliable memories with subline accesses," in *Proc. IEEE Int. Symp. Inf. Theory*, June 2007, pp. 2531–2535.