ALGEBRAS GROUPS AND GEOMETRIES 35 427 - 449 (2018)
DOI 10.29083/AGG.35.04.2018

UNIFORMLY INDEPENDENT I-SPOTTY-BYTE ERROR CONTROL CODES

Sapna Jain
Shanxi Normal University
P. R. China
sapnajain@gmx.com

Received February 7, 2020
Revised June 17, 2020

Abstract

In [5], the author has devised uniform i-spotty-byte error control codes which are suitable for semi-conductor memories where an I/O word is divided into irregular bytes not necessarily of the same length with those of equal length aligned together constituting a "sector" and uniform i-spotty-byte errors are defined as i-spotty-byte errors confined to a particular sector. The author also discussed the code design of uniform i-spotty-byte error control codes and their decoding. However, an important and practical situation is when uniform i-spotty-byte errors caused by the hit of high energetic particles are independent in nature in the sense that sum of the erroneous i-bytes is non-zero. Keeping this in view, in this paper, we propose a new model of uniform i-spotty-byte errors viz. "uniformly independent i-spotty byte errors" which are capable of correcting all uniformly independent i-spotty-byte errors of measure μ(or less). It has been further shown that these codes require fewer number of check bits than the earlier defined uniform i-spotty byte error control codes [5] and/or usual i-spotty-byte error control codes [4] meaning thereby that the rate of storage of these codes is higher than the other ones. Also, the decoding algorithm presented in this paper for uniformly independent i-spotty byte error control codes is greatly simplified than those for the uniform i-spotty byte error control codes.

Keywords: Uniform i-spotty-byte errors, i-spotty-byte distance, error control codes

1. Introduction

The uniform i-spotty-byte error control codes devised by the author [5] are suitable for semiconductor memories consisting of irregular RAM chips not necessarily of the same size and those of same length are aligned together and errors caused by the hit of high energetic particles are confined to adjacent i-bytes/RAM chips of the same size only. However, an important and practical situation is when uniform i-spotty-byte errors are independent in nature in the sense that sum of the erroneous i-bytes in the erroneous sector is non-zero. Keeping this in view, this paper proposes a new model of uniform i-spotty-byte error viz."uniformly independent i-spotty byte errors" and presents code for the correction of the same. It has been shown that the rate of storage of uniformly independent i-spotty byte error control codes is higher than the usual i-spotty byte error control codes and uniform i-spotty byte error codes. The paper also presents a decoding algorithm for uniformly independent i-spotty-byre error correcting code which is simplified in the sense of determining the erroneous sector in comparison to the decoding algorithm for uniform i-spotty byte error correcting codes.

2. Definitions and Notations

Let $q = p^m$ be a power of prime number p and $\mathbf{F}_q$ be the finite field with q elements. A partition, P, of a positive integer N is defined as

$$P : N = m_1 + m_2 + \cdots + m_g, 1 \leq m_1 \leq m_2 \cdots \leq m_g \quad g \geq 1.$$

and is denoted as

$$P = [m_1][m_2] \cdots [m_g] = [n_1]^{\lambda_1}[n_2]^{\lambda_2} \cdots [n_s]^{\lambda_s},$$

if

$$\begin{aligned}
&m_1 = m_2 = \cdots = m_{\lambda_1} = n_1, \\
&m_{\lambda_1+1} = m_{\lambda_1+2} = \cdots = m_{\lambda_1+\lambda_2} = n_2, \\
&\vdots \\
&\vdots \\
&\vdots \\
&m_{\lambda_1+\lambda_2+\cdots+\lambda_{s-1}+1} = m_{\lambda_1+\lambda_2\cdots\lambda_{s-1}+2} \\
&= \cdots = m_{\lambda_1+\lambda_2+\cdots+\lambda_s} = n_s.
\end{aligned}$$

Then we can write the field $\mathbf{F}_q^N$ as

$$\mathbf{F}_q^N \quad = \quad \mathbf{F}_q^{m_1} \oplus \mathbf{F}_q^{m_2} \oplus \cdots \oplus \mathbf{F}_q^{m_g}$$

$$= \bigoplus_{i=1}^{s}\Big(\bigoplus_{\lambda_i - copies} \mathbf{F}_q^{n_i}\Big).$$

Each vector $v \in \mathbf{F}_q^N = \bigoplus_{i=1}^{s}(\bigoplus_{\lambda_i - copies} \mathbf{F}_q^{n_i})$ can be uniquely written as $v = (v_1, v_2, \cdots, v_s)$ where $v_j \in (\mathbf{F}_q^{n_j})^{\lambda j}$ for all $1 \leq j \leq s$ and is represented as

$$v_j = (v_j^0, v_2^1, \cdots, v_j^{\lambda j - 1}), \quad v_j^a \in \mathbf{F}_q^{n_j} \text{ for all } 0 \leq a \leq \lambda_{j-1}, \tag{1}$$

or equivalently

$$\begin{aligned} v_j &= (v_j^{(0,1)}, v_j^{(0,2)}, \cdots, v_j^{(0,n_j)}, v_j^{(1,1)}, v_j^{(1,2)}, \cdots, v_j^{(1,n_j)}, \cdots \\ &\quad \cdots, v_j^{(\lambda j-1,1)}, v_j^{(\lambda j-1,2)}, \cdots, v_j^{(\lambda_j-1,n_j)}), \end{aligned}$$

where $v_j^a = (v_j^{(a,1)}, v_j^{(a,2)}, \cdots, v_j^{(a,n_j)}), \quad v_j^{(a,b)} \in \mathbf{F}_q$ for all $0 \leq a \leq \lambda_j - 1$ and $1 \leq b \leq n_j$. Here $v_j (1 \leq j \leq s)$ is called the "j^{th} sector of v" consisting of λ_j i-bytes viz. $v_j^0, v_j^1, \cdots, v_j^{\lambda_j - 1}$ each of length n_j. Thus the length of the j^{th} sector v_j is $\lambda_j n_j$. The partition P is named as *primary partition* or *irregular-byte partition.* Further, let $1 \leq T \leq N$ be a positive integer such that $P' : T = [t_1]^{\lambda_1}[t_2]^{\lambda_2} \cdots [t_s]^{\lambda_s}$ be a partition of T where $1 \leq t_i \leq n_i$ for all $1 \leq i \leq s$ and also $1 \leq t_1 \leq t_2 \leq \cdots \leq t_s$. Then P' is called as the *"secondary partition"* or *"error partition"*. Note that the secondary partition depends upon the primary partition. The number N is called the *primary number* and the number T is called the *secondary number.*

Clearly,

$$N = \lambda_1 n_1 + \lambda_2 n_2 + \cdots + \lambda_s n_s$$

and

$$T = \lambda_1 t_1 + \lambda_2 t_2 + \cdots + \lambda_s t_s.$$

We give below few definitions given in [5].

Definition 2.1 [5]. Let N and T be the primary and secondary numbers respectively as discussed in the preceeding paragraph corresponding to the partitions P and P' resp. given by

$$\begin{aligned} P : N &= [n_1]^{\lambda_1}[n_2]^{\lambda_2} \cdots [n_s]^{\lambda_s}, \\ &\text{and} \\ P' : T &= [t_1]^{\lambda_1}[t_2]^{\lambda_2} \cdots [t_s]^{\lambda_s}, \end{aligned}$$

where $1 \leq t_i \leq n_i$ for all $1 \leq i \leq s$.

Let $v = (v_1, v_2, \cdots, v_s)$ be a vector in $\mathbf{F}_q^N = \bigoplus_{i=1}^{s}\left(\bigoplus_{\lambda_i - copies} \mathbf{F}_q^{n_i}\right)$ as given in (1). The *irregular-spotty-byte weight* (or simply *i-spotty-byte weight*) $w_\beta^{(P,P')}(v)$ corresponding to the primary partition P and secondary partition P' is given by

$$w_\beta^{(P,P')}(v) = \sum_{i=1}^{s} \sum_{a=0}^{\lambda_i - 1} \left\lceil \frac{\sum_{b=1}^{n_i} w_H(v_i^{(a,b)})}{t_i} \right\rceil, \tag{2}$$

where $\sum_{b=1}^{n_i} w_H(v_i^{(a,b)})$ is the Hamming weight of the a^{th} i-byte in the i^{th} sector v_i and $\lceil x \rceil$ denotes the smallest integer greater than or equal to x.

Definition 2.2 [5]. The *irregular-spotty distance* (or simply *i-spotty distance*) between two vectors $u, v \in \mathbf{F}_q^N$ corresponding to the primary partition P and secondary partition P' is given by

$$\begin{aligned} d_\beta^{(P,P')}(u,v) = w_\beta^{(P,P')}(u - v) &= \sum_{i=1}^{s} \sum_{a=0}^{\lambda_i - 1} \left\lceil \frac{\sum_{b=1}^{n_i} w_H(u_i^{(a,b)} - v_i^{(a,b)})}{t_i} \right\rceil \\ &= \sum_{i=1}^{s} \sum_{a=0}^{\lambda_i - 1} \left\lceil \frac{\sum_{b=1}^{n_i} d_H(u_i^{(a,b)}, v_i^{(a,b)})}{t_i} \right\rceil, \end{aligned} \tag{3}$$

where $\sum_{b=1}^{n_i} d_H(u_i^{(a,b)}, v_i^{(a,b)})$ is the Hamming distance between the a^{th} i-bytes of the i^{th} sectors u_i and v_i of u and v respectively. Then i-spotty-byte distance is a metric function.

Note. We also call the i-spotty weight and i-spotty distance as "t_i/n_i-weight" and "t_i/n_i-distance" respectively. Moreover, we simply denote the i-spotty weight $w_\beta^{(P,P')}$ and i-spotty distance $d_\beta^{(P,P')}$ by w_β and d_β respectively when the primary partition P and secondary partition P' are clear from the context.

Definition 2.3 [5]. Let T and N be the primary and secondary numbers corresponding to the primary and secondary partitions P and P' resp. where P and P' are given by

$$P : N = [n_1]^{\lambda_1} [n_2]^{\lambda_2} \cdots [n_s]^{\lambda_s},$$

$$P' : T \;=\; [t_1]^{\lambda_1}[t_2]^{\lambda_2}\cdots[t_s]^{\lambda_s},$$

and $1 \leq t_i \leq n_i$ for all $1 \leq i \leq s$.

Let $V \subseteq \mathbf{F}_q^N = \bigoplus_{i=1}^{s}\Big(\bigoplus_{\lambda_i-copies} \mathbf{F}_q^{n_i}\Big)$ be an $\mathbf{F}_q$ subspace of $\mathbf{F}_q^N$ equipped with the i-spotty-byte metric d_β. Then V is called an *"irregular-spotty-byte"* (or simply *"i-spotty-byte"*) error control code and is denoted by $[N, k, d_\beta; P, P']$ where

$$\begin{aligned} N &= n_1\lambda_1 + n_2\lambda_2 + \cdots + n_s\lambda_s \\ &= \text{length of the code}, \\ k &= \dim_{\mathbf{F}_q}(V), \text{ and} \\ d_\beta &= \min_{\substack{x,y\in V \\ x\neq y}} d_\beta(x,y). \end{aligned}$$

3. Uniformly independent i-spotty-byte error control codes

In this section, we define uniformly independent i-spotty-byte errors and then design codes to control these types of errors. We begin with the definition of vectors of i-spotty weight or i-spotty measure $\mu(\mu \geq 1)$ in relation to Definition 2.1.

Definition 3.1. Let $v = (v_1, v_2, \cdots, v_s) \in \mathbf{F}_q^N = \bigoplus_{i=1}^{s}(\bigoplus_{\lambda_i-copies} \mathbf{F}_q^{n_i})$. If $w_\beta(v) = w_\beta^{(P,P')}(v) = \mu$, where $w_\beta^{(P,P')}(v)$ is given by (2), then we say that *i-spotty-weight* or *i-spotty measure of v is $\mu(\mu \geq 1)$* or equivalently we say that *t_i/n_i-measure of v is μ.*

Definition 3.2. A *"uniformly independent i-spotty-byte error"* of i-spotty measure μ is an error vector of i-spotty measure μ in which all the erroneous digits are confined to i-bytes of the same sector and the sum of erroneous i-bytes is non-zero.

Note. Every uniformly independent i-spotty-byte error is a uniform i-spotty-byte error but not conversely.

Example 3.3. Let $q = 2, N = 13, T = 9$ and

$$P : N = 13 = [1]^3[2]^2[3]^2,$$
$$P' : T = 9 = [1]^3[1]^2[2]^2,$$

be the primary and secondary partitions corresponding to $N = 13$ and $T = 9$ respectively. Then $u = (0\ \ 0\ \ 0\ \vdots\ 00\ \ 00\ \vdots\ 110\ \ 011) \in \mathbf{F}_2^{13}$ is a uniformly independent i-spotty-byte error of measure 2. But $v = (0\ \ 0\ \ 0\ \vdots\ 00\ \ 00\ \vdots\ 110\ \ 110) \in \mathbf{F}_2^{13}$ is not a uniformly independent i-spotty-byte error of measure 2 but it is a uniform i-spotty-byte error of measure 2.

We now define $r \times r$ companion matrix over $\mathbf{F}_q$ [1,6]:

Definition 3.4 [1,6]. Given a monic primitive polynomial $g(x)$ of degree r over $\mathbf{F}_q$, the $r \times r$ companion matrix M corresponding to $g(x)$ is defined as follows:

$$g(x) \quad = \quad g_0 + g_1x + g_2x^2 + \cdots\cdots + g_{r-2}x^{r-2} + g_{r-1}x^{r-1} + x^r,$$

$$M \quad = \quad \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 & -g_0 \\ 1 & 0 & \cdots & 0 & 0 & -g_1 \\ 0 & 1 & \cdots & 0 & 0 & -g_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & -g_{r-2} \\ 0 & 0 & \cdots & 0 & 1 & -g_{r-1} \end{pmatrix}_{r \times r}$$

Observations.

(i) Let α be a primitive element of $\mathbf{F}_q^r$ and a root of $g(x)$. Its companion matrix M has its columns $\begin{pmatrix} \vdots \\ \vdots \\ \alpha^i \\ \vdots \\ \vdots \end{pmatrix}$ for $i = 1$ to r where $\begin{pmatrix} \vdots \\ \vdots \\ \alpha^i \\ \vdots \\ \vdots \end{pmatrix}$ is the coefficient vector of $x^i (mod\ g(x))$.

The companion matrix of α^j is M^j and its column vectors are expressed as follows:

$$M^j = \begin{pmatrix} \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \alpha^j & \alpha^{j+1} & \cdots & \alpha^{j+r-1} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \end{pmatrix}_{r \times r} .$$

Let e be the exponent of $g(x)$, that is, $y = e$ is the least positive solution of $x^y \equiv$ $(mod\ g(x))$. The companion matrix M has the following properties [1,6]:

(a) M is non singular.

(b) $M^0 = M^e = I_r$.

(c) $M^i = M^j$ if and only if $i \equiv j (mod\ e)$.

Now, we present the code construction method of uniformly independent i-spotty-byte error control codes using the following definition:

Definition 3.5. Let q be a prime number or power of a prime number. Let $\mu, n_1 \leq n_2 \leq \cdots \leq n_s$ and $t_1 \leq t_2 \leq \cdots \leq t_s$ be positive integers with $1 \leq t_i \leq n_i$ for all $1 \leq i \leq s$. Define

$$t = \max_{j=1}^{s}\{t_j\} \quad \text{and} \quad n = \max_{j=1}^{s}\{n_j\}.$$

Further, for each $j = 1$ to s, let

(i) $m_j = q^{n_j} - 1$;

(ii) I_{n_j} be the $n_j \times n_j$ identity matrix and $O_{n_i} \times n_j$ be the $n_i \times n_j$ null matrix over $\mathbf{F}_q$ for all $1 \leq i \leq s$;

(iii) M'_j be the $n_j \times n_j$ companion matrix corresponding to a q-ary primitive polynomial $g_j(x)$ of degree n_j;

(iv) $(M_j)^{i_j}$ be the $n \times n_j$ extended matrix obtained from the $n_j \times n_j$ matrix $(M'_j)^{i_j}$ by adding $(n - n_j)$ all-zero rows to it for all $0 \leq i_j \leq m_j - 1$. We call M_j as the *extended companion matrix with respect to* M'_j.

Theorem 3.6. *Using the notations as given in Definitions 3.5, the null space of matrix* H *where*

$$H = \begin{pmatrix} H'_1 & H'_2 & \cdots & H'_s \\ H''_1 & H''_2 & \cdots & H''_s \end{pmatrix},$$

and each $H'_j (1 \leq j \leq s)$ *is a* $(\sum_{j=1}^{s} n_j) \times (m_j n_j)$ *submatrix and each* $H''_j (1 \leq j \leq s)$ *is a*

$((2\mu-1)n) \times (m_j n_j)$ *submatrix given by*

$$H'_j = \begin{pmatrix} O_{n_1\times n_j} & O_{n_1\times n_j} & \cdots & O_{n_1\times n_j} \\ O_{n_2\times n_j} & O_{n_2\times n_j} & \cdots & O_{n_2\times n_j} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ O_{n_{j-1}\times n_j} & O_{n_{j-1}\times n_j} & \cdots & O_{n_{j-1}\times n_j} \\ I_{n_j} & I_{n_j} & \cdots & I_{n_j} \\ O_{n_{j+1}\times n_j} & O_{n_{j+1}\times n_j} & \cdots & O_{n_{j+1}\times n_j} \\ \vdots & \vdots & \vdots & \\ \vdots & \vdots & \vdots & \\ O_{n_s\times n_j} & O_{n_s\times n_j} & \cdots & O_{n_s\times n_j} \end{pmatrix}_{(\sum_{j=1}^{s} n_j)\times(m_j n_j)},$$

$$H''_j = \begin{pmatrix} M_j^0 & M_j^1 & \cdots & (M_j)^{(m_j-1)} \\ M_j^0 & M_j^2 & \cdots & (M_j)^{2(m_j-1)} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ M_j^0 & (M_j)^{(2\mu-1)} & \cdots & (M_j)^{(2\mu-1)(m_j-1)} \end{pmatrix}_{(2\mu-1)n\times(m_j n_j)}.$$

is a uniformly independent i-spotty-byte error control code correcting all uniformly independent i-spotty-byte errors of measure μ or less and having check bit length $R = \sum_{j=1}^{s} n_j + (2\mu-1)n$ and code length $N = m_1n_1 + m_2n_2 + \cdots + m_sn_s$. The parameters of the resulting code will be

$$[N, N-R, d; P, P'],$$

where $P : N = [n_1]^{m_1}[n_2]^{m_2}\cdots[n_s]^{m_s}$, $P' : T = [t_1]^{m_1}[t_2]^{m_2}\cdots\cdots[t_s]^{m_s}$ and $d \le 2\mu+1$.

Proof. It suffices to prove that the code V which is the null space of H detects all i-spotty-byte errors of measure 2μ or less with errors occuring in the same sector and sum of erroneous i-bytes be non-zero meaning thereby that the code corrects all uniformly independent i-spotty-byte errors of measure μ or less.

Let $e \in \mathbf{F}_q^N = \bigoplus_{j=1}^{s}\Big(\bigoplus_{m_j-copies} \mathbf{F}_q^{n_j}\Big)$.
Then e is of the form

$$e = (e_1, \cdots, e_s)$$

$$= \quad (e_1^0, e_1^1, \cdots, e_1^{m_1-1}, e_2^0, e_2^1, \cdots, e_2^{m_2-1}, \cdots, e_s^0, e_s^1, \cdots, e_s^{m_s-1}),$$

where $e_j^{u_j} \in \mathbf{F}_q^{n_j}$ for all $1 \leq j \leq s$ and $0 \leq u_j \leq m_j - 1$.

Suppose $w_\beta(e) \leq 2\mu$ with erroneous i-bytes confined to a single sector and sum of erroneous i-bytes is non-zero. Let j^{th} sector e_j be the erroneous sector having erroneous i-bytes say $e_j^{u_1}, e_j^{u_2}, \cdots, e_j^{u_{j^*}}$ with

$$\sum_{k=1}^{j^*} \left\lceil \frac{w_H(e_j^{u_k})}{t_j} \right\rceil \leq 2\mu,$$

and $e_j^{u_1} + e_j^{u_2} + \cdots + e_j^{u_{j^*}} \neq 0$.

Then $eH^T = 0$ gives the following relation:

$$\begin{aligned}
& \left[e_j^{u_1}(I_{n_j})^T \quad e_j^{u_1}(M_j^{u_1})^T \quad e_j^{u_1}(M_j^{2u_1})^T \cdots\cdots e_j^{u_1}(M^{(2\mu-1)u_1})^T\right] \\
& +\left[e_j^{u_2}(I_{n_j})^T \quad e_j^{u_2}(M_j^{u_2})^T \quad e_j^{u_2}(M_j^{2u_2})^T \cdots\cdots e_j^{u_2}(M^{(2\mu-1)u_2})^T\right] \\
& + \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
& + \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
& +\left[e_j^{u_{j^*}}(I_{n_j})^T \quad e_j^{u_{j^*}}(M^{u_{j^*}})^T \quad e_j^{u_{j^*}}(M^{2u_{j^*}})^T \cdots\cdots e_j^{u_{j^*}}(M^{(2\mu-1)u_{j^*}})^T\right] \\
= \quad & [O_{n_j} \quad O_n \quad O_n \cdots\cdots O_n],
\end{aligned}$$

where O_{n_j} and O_n are the $1 \times n_j$ and $1 \times n$ null matrices respectively.

Writing the above equation in the matrix form gives

$$\begin{aligned}
& (e_j^{u_1}, e_j^{u_2}, \cdots, e_j^{u_{j^*}}) \times \\
& \times \begin{pmatrix}
I_{n_j} & (M_j^{u_1})^T & (M_j^{2u_1})^T & \cdots & (M_j^{(2\mu-1)u_1})^T \\
I_{n_j} & (M_j^{u_2})^T & (M_j^{2u_2})^T & \cdots & (M_j^{(2\mu-1)u_2})^T \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
I_{n_j} & (M_j^{u_{j^*}})^T & (M^{(2u_{j^*}})^T & \cdots & (M_j^{(2\mu-1)u_{j^*}})^T
\end{pmatrix} \\
& = (O_{n_j}, O_n, O_n, \cdots, O_n).
\end{aligned}$$

Replacing the extended companion matrix M_j by the corresponding companion matrix M_j' in the above matrix equation results in an equivalent matrix equation and is given by

$$(e_j^{u_1}, e_j^{u_2}, \cdots, e_j^{u_{j^*}}) \times$$

$$\times \begin{pmatrix} I_{n_j} & (M_j')^{u_1})^T & ((M_j')^{2u_1})^T & \cdots & ((M_j')^{(2\mu-1)u_1})^T \\ I_{n_j} & (M_j')^{u_2})^T & ((M_j')^{2u_2})^T & \cdots & ((M_j')^{(2\mu-1)u_2})^T \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ I_{n_j} & (M_j')^{u_{j^*}})^T & ((M_j')^{(2u_{j^*}})^T & \cdots & ((M_j')^{(2\mu-1)u_{j^*}})^T \end{pmatrix}$$

$$= (O_{n_j}, O_{n_j}, O_{n_j}, \cdots, O_{n_j}),$$

or equivalently

$$(e_j^{u_1}, e_j^{u_2}, \cdots, e_j^{u_{j^*}}) \times$$

$$\times \begin{pmatrix} I_{n_j} & I_{n_j} & \cdots & I_{n_j} \\ (M_j')^{u_1} & (M_j')^{u_2} & \cdots & (M_j')^{u_{j^*}} \\ \vdots & \vdots & \vdots & \vdots \\ (M_j')^{(2\mu-1)u_1} & (M_j')^{(2\mu-1)u_2} & \cdots & (M_j')^{(2\mu-1)u_{j^*}} \end{pmatrix}^T$$

$$= (O_{n_j}, O_{n_j}, O_{n_j}, \cdots, O_{n_j}).$$

Since the total numbers j^* of erroneous i-bytes is less than or equal to 2μ, therfore, writing the above matrix equation for the top j^* relations, we get

$$(e_j^{u_1}, e_j^{u_2}, \cdots, e_j^{u_{j^*}}) \times$$

$$\times \begin{pmatrix} I_{n_j} & I_{n_j} & \cdots & I_{n_j} \\ (M_j')^{u_1} & (M_j')^{u_2} & \cdots & (M_j')^{u_{j^*}} \\ \vdots & \vdots & \vdots & \vdots \\ (M_j')^{(j^*-1)u_1} & (M_j')^{(j^*-1)u_2} & \cdots & (M_j')^{(j^*-1)u_{j^*}} \end{pmatrix}^T$$

$$= (O_{n_j}, O_{n_j}, \cdots, O_{n_j}).$$

The coefficient matrix being Vandermonde's matrix is non-singular. Therefore, the above system of equations have a solution viz.

$$e_j^{u_1} = e_j^{u_2} = \cdots = e_j^{u_{j^*}} = O_{n_j}$$

which implies that $e = 0$, A contradiction. Thus $eH^T \neq 0$ and hence H is the null space of an i-spotty-byte code correcting all uniformly independent i-spotty-byte errors of measure μ or less. □

Note. We may also have a shortened version of the code discussed in Theorem 3.6 by choosing integers $\lambda_1, \lambda_2, \cdots, \lambda_s$ such that $1 \leq \lambda_j \leq m_j$ for all $j = 1$ to s and then keeping only first $\lambda_j n_j$ columns of he submatrices H'_j and H''_j. The parameters of the shortened code will be $[N, N-R, d; P, P']$ where

$$\begin{aligned} N &= \lambda_1 n_1 + \lambda_2 n_2 + \cdots + \lambda_s n_s, \\ R &= \sum_{i=1}^{s} n_i + (2\mu - 1)n, \\ P : N &= [n_1]^{\lambda_1}[n_2]^{\lambda_2} \cdots [n_s]^{\lambda_s}, \\ P' : T &= [t_1]^{\lambda_1}[t_2]^{\lambda_2} \cdots\cdots [t_s]^{\lambda_s}, \text{ and} \\ d &\leq 2\mu + 1. \end{aligned}$$

4. Decoding of uniformly independent i-spotty-byte error correcting codes

In this section, we present a decoding algorithm for the uniformly independent i-spotty-byte error correcting codes.

Let V be an i-spotty-byte code correcting all uniformly independent i-spotty-byte errors of measure μ or less. Let u, r and e be the transmitted codeword, the received word and the error vector respectively. The syndrome S of the received word r is calculated as

$$S = \begin{pmatrix} S_1 \\ S_2 \\ \vdots \\ S_{2\mu} \end{pmatrix}^T = rH^T = (u+e)H^T = eH^T,$$

where $S_1 \in \mathbf{F}_q^{n_1+n_2+\cdots+n_s}$ is a $(n_1 + n_2 + \cdots + n_s)$-bit q-ary row vector and $S_i \in \mathbf{F}_q^n$ for all $2 \leq i \leq 2\mu$.

The decoding is done as follows:

Step 1.

(i) If $S_1 = S_2 = \cdots = S_{2\mu} = 0$, then there is no error during the process of communication.

(ii) If $S_1 \neq 0$ and all of $S_2 = S_3 = \cdots = S_{2\mu} = 0$, then an uncorrectable error pattern of measure greater than μ has occured and a retransmission is sought.

(iii) If $S_1 = 0$ and either of $S_2, S_3, \cdots, S_{2\mu} \neq 0$, then we infer that sum of erroneous i-bytes in the erroneous sector is zero meaning thereby that the errors are not uniformly independent and again a retransmission is sought.

(iv) If $S_1 \neq 0$ and at least one of $S_2, S_3, \cdots S_{2\mu}$ is non-zero, then for the decoding purpose, the pattern of S_1 indicates the erroneous sector position as explained below:

We partition S_1 into different blocks of lengths $n_1, n_2, \cdots, n_s$ consecutively. Now, if the first n_1 bits in S_1 form a non-zero pattern and remaining $n_2+n_3+\cdots+n_s$ digits are all-zero, then the error is in the first sector. If the first n_1 bits in S_1 are all-zero and the next n_2 bits form a non-zero pattern and also the remaining $n_3 + n_4 + \cdots + n_s$ bits in S_1 are all-zero, then the error lies in the second sector and so on.

Step 2. From Step 1, we know the erroneous sector number say j^{th} sector is in error having i-bytes of length n_j. Now, from the patterns of $S_2, S_3, \cdots, S_{2\mu}$, we define a new syndrome vector S^* as follows:

$$S^* = \begin{pmatrix} S_1^* \\ S_2^* \\ \vdots \\ S_{2\mu}^* \end{pmatrix}^T,$$

where

$$\begin{aligned} S_1^* &= n_j\text{-bit non-zero pattern as given by } S_1, \\ S_2^* &= \text{first } n_j\text{-bits of } S_2, \\ S_3^* &= \text{first } n_j\text{-bits of } S_3 \\ &\vdots \\ &\vdots \\ S_{2\mu}^* &= \text{first } n_j\text{-bits of } S_{2\mu}. \end{aligned}$$

Step 3. After calculating the new syndrome vector S^* which is identical to the syndrome of an error vector in Reed-Solomon code with minimum distance $(2\mu+1)$ over $\mathbf{F}_q^{n_j}$, the error pattern and error location over $\mathbf{F}_q^{n_j}$ are determined by using the Berlekamp-Massey algorithm for RS codes over $\mathbf{F}_q^{n_j}$.

We illustrate the code construction method and decoding algorithm with the help of following examples:

Example 4.1. Let $q = 2$, $n_1 = 2, n_2 = 3, t_1 = 2, t_2 = 2$ and $\mu = 2$. Then

$$m_1 = 2^{n_1} - 1 = 2^2 - 1 = 3, \text{ and } m_2 = 2^{n_2} - 1 = 2^3 - 1 = 7,$$
$$t = \max_{j=1}^{2}\{t_j\} = \max\{2,2\} = 2,$$
$$n = \max_{j=1}^{2}\{n_j\} = \max\{2,3\} = 3.$$

The parameters of the code to be constructed as discussed in Theorem 3.6 are given by $R = \sum_{j=1}^{2} n_j + (2\mu - 1)n = 14$ and $N = \sum_{j=1}^{2} m_j n_j = 27.$

Let M_1' be the 2×2 binary companion matrix defined by the binary primitive polynomial $g_1(x) = x^2 + x + 1$. Then M_1' is given by

$$M_1' = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}_{2\times 2}.$$

The 3×2 extended companion matrix M_1' is obtained by adding a row of all-zeros to M_1' and is given by

$$M_1' = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}_{3\times 2}.$$

The various powers of extended companion matrix M_1 are given below:

$$M_1^0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}_{3\times 2}, M_1^1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}_{3\times 2},$$
$$M_1^2 = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{pmatrix}_{3\times 2}, M_1^3 = M_1^0.$$

Let M_2' be the 3×3 binary companion matrix defined by the binary primitive polynomial $g_2(x) = x^3 + x + 1$ and is given by

$$M_2' = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}_{3\times 3}.$$

Since here $n = n_2 = 3$, therefore, the extended companion matrix M_2 is same as the companion matrix M_2' i.e. $M_2 = M_2'$. The various powers of extended companion matrix M_2 are given below

$$M_2^0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}_{3\times3}, M_2^1 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}_{3\times3}, M_2^2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}_{3\times3},$$

$$M_2^3 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}_{3\times3}, M_2^4 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}_{3\times3}, M_2^5 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}_{3\times3},$$

$$M_2^6 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}_{3\times3}, M_2^7 = I_3.$$

The parity check matrix of an i-spotty-byte code correcting all uniformally independent errors of measure 2 or less is given by

$$H = \begin{pmatrix} I_2 & I_2 & I_2 & \vdots & O_{2\times3} & O_{2\times3} & O_{2\times3} & O_{2\times3} & O_{2\times3} & O_{2\times3} & O_{2\times3} \\ O_{3\times2} & O_{3\times2} & O_{3\times2} & \vdots & I_3 & I_3 & I_3 & I_3 & I_3 & I_3 & I_3 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ M_1^0 & M_1^1 & M_1^2 & \vdots & M_2^0 & M_2^1 & M_2^2 & M_2^3 & M_2^4 & M_2^5 & M_2^6 \\ M_1^0 & M_1^2 & M_1^4 & \vdots & M_2^0 & M_2^2 & M_2^4 & M_2^6 & M_2^8 & M_2^{10} & M_2^{12} \\ M_1^0 & M_1^3 & M_1^6 & \vdots & M_2^0 & M_2^3 & M_2^6 & M_2^9 & M_2^{12} & M_2^{15} & M_2^{18} \end{pmatrix}_{14\times27},$$

$$= \begin{pmatrix} 10 & 10 & 10 & \vdots & 000 & 000 & 000 & 000 & 000 & 000 & 000 \\ 01 & 01 & 01 & \vdots & 000 & 000 & 000 & 000 & 000 & 000 & 000 \\ 00 & 00 & 00 & \vdots & 100 & 100 & 100 & 100 & 100 & 100 & 100 \\ 00 & 00 & 00 & \vdots & 010 & 010 & 010 & 010 & 010 & 010 & 010 \\ 00 & 00 & 00 & \vdots & 001 & 001 & 001 & 001 & 001 & 001 & 001 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 10 & 01 & 11 & \vdots & 100 & 001 & 010 & 101 & 011 & 111 & 110 \\ 01 & 11 & 10 & \vdots & 010 & 101 & 011 & 111 & 110 & 100 & 001 \\ 00 & 00 & 00 & \vdots & 001 & 010 & 101 & 011 & 111 & 110 & 100 \\ 10 & 11 & 01 & \vdots & 100 & 010 & 011 & 110 & 001 & 101 & 111 \\ 01 & 10 & 11 & \vdots & 010 & 011 & 110 & 001 & 101 & 111 & 100 \\ 00 & 00 & 00 & \vdots & 001 & 101 & 111 & 100 & 010 & 011 & 110 \\ 10 & 10 & 10 & \vdots & 100 & 101 & 110 & 010 & 111 & 001 & 011 \\ 01 & 01 & 01 & \vdots & 010 & 111 & 001 & 011 & 100 & 101 & 110 \\ 00 & 00 & 00 & \vdots & 001 & 011 & 100 & 101 & 110 & 010 & 111 \end{pmatrix}_{14\times 27},$$

Let $e = (00\ \ 00\ \ 00\ \vdots\ 100\ 010\ 000\ 000\ 000\ 000\ 000)$ be the error vector that has occured during the process of communication.

We compute the syndrome vector S as

$$S = eH^T = (S_1, S_2, S_3, S_4) = (00110, 101, 010, 111).$$

We partition S_1 into $n_1 + n_2 = 2 + 3$ bits i.e. $S_1 = (00 \quad 110)$.

Since the first n_1-bits (i.e 2 bits) of S_1 are all-zero and the remaining n_2-bits (i.e. 3 bits) are non-zero, therefore, we infer that the error lies in the second sector.

We form a new syndrome vector S^*as

$$S^* = (S_1^*, S_2^*, S_3^*, S_4^*),$$

where

$$\begin{aligned} S_1^* &= n_2\text{-bit i.e. 3 bit non-zero pattern in } S_1 = 110, \\ S_2^* &= \text{first } n_2\text{-bits i.e. first 3 bits of } S_2 = 101, \\ S_3^* &= \text{first 3-bits of } S_3 = 010, \\ S_4^* &= \text{first 3-bits of } S_4 = 111. \end{aligned}$$

Thus

$$S^* = (S_1^*, S_2^*, S_3^*, S_4^*) = (110, 101, 010, 111) = (\alpha^3, \alpha^6, \alpha, \alpha^5),$$

where α is a primitive element of F_2^3 defined by the polynomial $x^3 + x + 1$.

The error locator polynomial σ^* for the syndrome S^* using the Berlekamp-Massey algorithm is computed as

$$\sigma^*(x) = 1 + \alpha^3 x + \alpha x^2.$$

The degree of the error locator polynomial denotes the number of erroneous i-bytes. Thus, in this case, there are two erroneous i-bytes. The inverse of the roots of the error locator polynomial gives the positions of the erroneous i-bytes. Since the roots of $\sigma^*(x)$ are α^6 and $\alpha^0 = 1$, therefore, the inverse of the roots of $\sigma^*(x)$ are $\alpha^{-6} = \alpha^1$ and $\alpha^0 = 1$. Thus we deduce that zeroth and first i-bytes are the erroneous i-bytes in the erroneous second sector. To find the erroneous i-byte patterns, we compute the evaluator polynomial evaluator $Z^*(x)$ as given below:

$$\begin{aligned} Z^*(x) &= \sigma^*(x) S^*(x) \\ &= (1 + \alpha^3 x + \alpha x^2)(1 + S_1^* x + S_2^* x^2 + S_3^* x^3 + S_4^* x^4) \\ &= (1 + \alpha^3 x + \alpha x^2)(1 + \alpha^3 x + \alpha^6 x^2 + \alpha x^3 + \alpha^5 x^4) \\ &= 1 + \alpha x^2 \text{ (keeping the terms of degree less than or equal to} \\ &\qquad \text{2 since there are two erroneous i-bytes).} \end{aligned}$$

Now, the erroneous i-byte patterns are given by the formula

$$e_j^i = \beta_i \left(\frac{Z^*(\beta_i^{-1})}{\prod\limits_{k \neq i} (1 + \beta_k \beta_i^{-1})} \right),$$

where j is the erroneous sector number and $\beta_i = \alpha^i$= inverse of root of $\sigma^*(x)$ and i varies over powers occuring in α^i and gives the position of erroneous i-byte.

Here $j = 2, i = 0, 1$, $\beta_0 = \alpha^0$ and $\beta_1 = \alpha^1$.

The erroneous i-bytes e_2^0 and e_2^1 are computed as

$$\begin{aligned} e_2^0 &= \beta_0 \left(\frac{Z^*(\beta_0^{-1})}{(1 + \beta_1 \beta_0^{-1}})\right) \\ &= 1 \left(\frac{1 + \alpha . 1^2}{(1 + \alpha)} \right) \\ &= 1 \left(\frac{1 + \alpha}{1 + \alpha} \right) = 1 = (100), \end{aligned}$$

$$
\begin{aligned}
e_2^1 &= \beta_1\left(\frac{Z^*(\beta_1^{-1})}{(1+\beta_0\beta_1^{-1}})\right) \\
&= \alpha\left(\frac{1+\alpha(\alpha^{-1})^2}{(1+\alpha^{-1}.1)}\right) \\
&= \alpha\left(\frac{1+\alpha^{-1}}{1+\alpha^{-1}}\right) = \alpha = (010).
\end{aligned}
$$

Hence the error pattern is given by

$$e = (00 \;\; 00 \;\; 00 \;\vdots\; 100 \;\; 010 \;\; 000 \;\; 000 \;\; 000 \;\; 000 \;\; 000).$$

Example 4.2. Using the parameters of Example 4.1, we consider the case of double uniformly independent i-spotty-byte errors occuring in the same i-byte of the erroneous sector. For this, let the error vector be

$$e = (00 \;\; 00 \;\; 00 \;\vdots\; 000 \;\; 000 \;\; 111 \;\; 000 \;\; 000 \;\; 000 \;\; 000).$$

The syndrome vector is computed as

$$S = eH^T = (S_1, S_2, S_3, S_4) = (00111, 100, 001, 011).$$

Since the first $n_1(=2)$ bits of S_1 are all-zero and remaining $n_2(=3)$bits form a non-zero pattern, therefore, the error lies in the second sector.

Define

$$S^* = (S_1^*, S_2^*, S_3^*, S_4^*) = (111, 100, 001, 011) = (\alpha^5, 1, \alpha^2, \alpha^4),$$

where α is a root of $x^3 + x + 1 \in \mathbf{F}_2[x]$.

The error locator polynomial $\sigma^*(x)$ for the syndrome S^* using the Berlekamp Massey algorithm is given by

$$\sigma^*(x) = 1 + \alpha^2 x.$$

Since degree $(\sigma^*(x)) = 1$, therefore, the number of erroneous i-bytes in the erroneous second sector is 1. Also, α^5 is a root of $\sigma^*(x)$ meaning thereby that $\alpha^{-5} = \alpha^2$ gives the location of the erroneous i-byte i.e. error lies in the second i-byte of the second sector.

The evaluator polynomial $Z^*(x)$ is computed as

$$
\begin{aligned}
Z^*(x) &= \sigma^*(x)(x)S^*(x) \\
&= (1+\alpha^2 x)(1+\alpha^5 x + x^2 + \alpha^2 x^3 + \alpha^4 x^4) \\
&= 1 + \alpha^3 x \text{ (kleeping the terms of degree less than or} \\
&\qquad\qquad \text{equal to 1 as there is a single erroneous i-byte).}
\end{aligned}
$$

Now, the erroneous i-byte pattern is given by

$$e_2^2 = \alpha^2 \left(\frac{1 + \alpha^3(\alpha^{-2})}{1} \right) = \alpha^2(1+\alpha) = \alpha^2\alpha^3 = \alpha^5 = (111).$$

Hence the error pattern is given by

$$e = (00 \;\; 00 \;\; 00 \;\vdots\; 000 \;\; 000 \;\; 111 \;\; 000 \;\; 000 \;\; 000 \;\; 000).$$

Example 4.3. Let $q = 2$, $n_1 = 2, n_2 = 4, t_1 = 2, t_2 = 4$ and $\mu = 1$. Then

$$m_1 = 2^{n_1} - 1 = 2^2 - 1 = 3, \quad m_2 = 2^{n_2} - 1 = 2^4 - 1 = 15,$$
$$t = \max_{j=1}^{2}\{t_j\} = \max\{2,4\} = 4, \quad \text{and}$$
$$n = \max_{j=1}^{2}\{n_j\} = \max\{2,4\} = 4.$$

The parameters of the code to be constructed are given by $R = (n_1 + n_2) + (2\mu + 1)n = (2+4) + 4 = 10$ and $N = m_1 n_1 + m_2 n_2 = 3 \times 2 + 15 \times 4 = 66$.

Let M_1' be the 2×2 binary companion matrix defined by the binary primitive polynomial $g_1(x) = x^2 + x + 1$ as given in Example 4.1. The 4×2 extended companion matrix M_1 is obtained by adding two all-zero rows to M_1' and is given by

$$M_1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}_{4\times 2}.$$

The various powers of extended companion matrix M_1 are given below:

$$M_1^0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}_{4\times 2}, M_1^1 = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}_{4\times 2}, M_1^2 = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}_{4\times 2}, M_1^3 = M_1^0.$$

Let M_2' be the 4×4 binary companion matrix defined by the primitive polynomial $g_2(x) = x^4 + x + 1$ and is given by

$$M_2' = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}_{4\times 4}.$$

Since here $n = n_2 = 4$, therefore, the extended companion matrix M_2 is same as the companion matrix M_2' i.e. $M_2 = M_2'$. The various powers of extended companion matrix

M_2 are given below

$$
\begin{aligned}
M_2^0 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}_{4\times 4}, M_2^1 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}_{4\times 4}, M_2^2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}_{4\times 4}, \\
M_2^3 &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}_{4\times 4}, M_2^4 = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}_{4\times 4}, M_2^5 = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}_{4\times 4}, \\
M_2^6 &= \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}_{4\times 4}, M_2^7 = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}_{4\times 4} \quad M_2^8 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}_{4\times 4} \\
M_2^9 &= \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}_{4\times 4}, M_2^{10} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}_{4\times 4} \quad M_2^{11} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}_{4\times 4} \\
M_2^{12} &= \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}_{4\times 4}, M_2^{13} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}_{4\times 4} \quad M_2^{14} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}_{4\times 4}, \\
M_2^{15} &= I_4.
\end{aligned}
$$

The parity check matrix H of an i-spotty-byte code correcting all uniformally independent errors of measure 1 or less is given by

$$H = (A \vdots B)_{10\times 66},$$

where

$$
\begin{aligned}
A &= \begin{pmatrix} I_2 & I_2 & I_2 \\ O_{4\times 2} & O_{4\times 2} & O_{4\times 2} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots \\ M_1^0 & M_1^1 & M_1^2 \end{pmatrix}_{10\times 6}, \\
&= \begin{pmatrix} 10 & 10 & 10 \\ 01 & 01 & 01 \\ 00 & 00 & 00 \\ 00 & 00 & 00 \\ 00 & 00 & 00 \\ 00 & 00 & 00 \\ \cdots & \cdots & \cdots \\ 10 & 01 & 11 \\ 01 & 11 & 10 \\ 00 & 00 & 00 \\ 00 & 00 & 00 \end{pmatrix}_{10\times 6}, \\
B &= \begin{pmatrix} O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} & O_{2\times 4} \\ I_4 & I_4 & I_4 & I_4 & I_4 & I_4 & I_4 & I_4 & I_4 & I_4 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ M_2^0 & M_2^1 & M_2^2 & M_2^3 & M_2^4 & M_2^5 & M_2^6 & M_2^7 & M_2^7 & M_2^9 \end{pmatrix}_{10\times 60},
\end{aligned}
$$

$$
= \begin{pmatrix}
0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 \\
0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 \\
1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\
0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 & 0100 \\
0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 & 0010 \\
0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 & 0001 \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
1000 & 0001 & 0010 & 0100 & 1001 & 0011 & 0110 & 1101 & 1010 & 0101 & 1011 & 0111 & 1111 & 1110 & 1100 \\
0100 & 1001 & 0011 & 0110 & 1101 & 1010 & 0101 & 1011 & 0111 & 1111 & 1110 & 1100 & 1000 & 0001 & 0010 \\
0010 & 0100 & 1001 & 0011 & 0011 & 0110 & 1101 & 0101 & 1011 & 0111 & 1111 & 1110 & 1100 & 1000 & 0001 \\
0001 & 0010 & 0100 & 1001 & 0011 & 0110 & 1101 & 1010 & 0101 & 1011 & 0111 & 1111 & 1110 & 1100 & 1000
\end{pmatrix}_{10\times 60}
$$

Let $e = (00\ \ 00\ \ 00\ \vdots\ 0100\ 1000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000)$ be the error vector. Then the syndrome of the error vector e is given by

$$S = eH^T = (S_1, S_2) = (001100, 0000).$$

Since $S_1 \neq 0$ and $S_2 = 0$, thus an uncorrectable error pattern has occured and retransmission is sought. This is evident from the fact that the code which is the null space of H can correct only single i-byte errors whereas the error vector e contains double i-byte errors and hence the errors are only detected but not corrected by the decoder.

On the other hand, suppose e has single i-spotty-byte error occuring in the (say) second sector i.e. let

$$
\begin{aligned}
e = (0\ 0\ 00\ \vdots\ 1100\ \ &0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000 \\
&0000\ 0000\ 0000\ 0000\ 0000).
\end{aligned}
$$

Computing syndrome of e gives

$$S = eH^T = (S_1, S_2) = (00100, 1100).$$

Since the first $n_1(= 2)$ bits of S_1 are all-zero and the remaining $n_2(= 4)$-bits form a non-zero pattern, therefore, we infer that the error lies in the second sector of the error vector.

We form a new syndrome vector

$$S^* = (S_1^*, S_2^*),$$

where

$$
\begin{aligned}
S_1^* &= n_2(i.e.4)\text{-bit non-zero pattern in } S_1 = 1100 = \alpha^4, \\
S_2^* &= \text{first 4-bits of } S_2 = 1100 = \alpha^4,
\end{aligned}
$$

where α is a root of $x^4 + x + 1$.

Thus

$$S^* = (S_1^*, S_2^*) = (\alpha^4, \alpha^4)$$

The error locator polynomial $\sigma^*(x)$ for the syndrome S^* using the Berlekamp-Massey algorithm is computed as

$$\sigma^*(x) = 1 + x.$$

Since $\deg(\sigma^*(x_1) = 1$, therefore, we infer that there is only one erroneous i-byte in the erroneous second sector. Since $\alpha^0 = 1$ is a root of $\sigma^*(x)$, therefore, $\alpha_0^{-1} = 1 = \alpha^0$ gives the location of the erroneous i-byte i.e. 0^{th} i-byte in the second sector contains errors.

The evaluator polynomial $Z^*(x)$ is given by

$$\begin{aligned} Z^*(x) &= \sigma^*(x) S^*(x) \\ &= (1+x)(1+\alpha^4 x + \alpha^4 x^2) \\ &= 1 + \alpha x \text{ (as } \deg\ (Z^*(x) \leq 1). \end{aligned}$$

Now, the erroneous i-byte pattern is given by the formula

$$\begin{aligned} e_2^0 &= \beta_0 \left(\frac{Z^*(\beta_0^{-1})}{1} \right), \\ &= \alpha^0 \left(\frac{1+\alpha.1}{1} \right) = 1 + \alpha = \alpha^4 = (1100), \quad \text{where } \beta_0 = \alpha^0 = 1. \end{aligned}$$

Hence the error pattern is given by

$$e = (00\ \ 00\ \ 00\ \vdots\ 1100\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000$$
$$0000\ 0000\ 0000\ 0000\ 0000).$$

Comparative Study.

In this section, we present a comparative study of rates of transmission/storage of binary uniformly independent i-spotty byte error control codes presented in this paper and non-uniformly independent i-spotty byte codes presented earlier [4.5] for different values of the parameters $n_1, n_2, \cdots, n_s$ and $t_1, t_2 \cdots, t_s$. We are restricting ourself for $s = 2$ only.

Table 5.1

Value of parameters	measures μ *of* uniformly independent errors to be corrected	Rate of uniformly independent i-spotty byte code	Rate of non-uniformly independent i-spotty byte code
$n_1 = 2, n_2 = 3$ $t_1 = 2, t_2 = 2$	$\mu = 1$	0.703	0.600
$n_1 = 3, n_2 = 4$ $t_1 = 3, t_2 = 3$	$\mu = 1$	0.864	0.816
$n_1 = 3, n_2 = 5$ $t_1 = 2, t_2 = 4$	$\mu = 1$	0.926	0.900
$n_1 = 2, n_2 = 4$ $t_1 = 2, t_2 = 3$	$\mu = 1$	0.848	0.785
$n_1 = 2, n_2 = 3$ $t_1 = 2, t_2 = 1$	$\mu = 2$	0.481	0.333
$n_1 = 3, n_2 = 4$ $t_1 = 1, t_2 = 1$	$\mu = 2$	0.764	0.523
$n_1 = 3, n_2 = 3$ $t_1 = 1, t_2 = 1$	$\mu = 2$	0.642	0.440

We infer from the Table 5.1 that the rate of transmission/storage of uniformly independent i-spotty byte codes is higher than the non-uniformly independent i-spotty byte codes and thus the codes presented in this paper are more economical to use and provide better transmission/storage rate than the earlier considered non-uniformly independent i-spotty-byte codes [4,5].

5. Conclusion.

In this paper, we have presented a new class of i-spotty-byte codes viz. uniformly independent i-spotty-byte error control codes which are capable of correcting all uniform i-spotty-byte errors and the errors are independent in the sense that the sum of erroneous i-bytes occuring in the erroneous sector is non-zero. We have discussed the code design of these codes in terms of their parity check matrix followed by a decoding algorithm to decode the received vector which is further illustrated by several examples. Finally, a comparative study of the codes presented in this paper with the earlier known i-spotty-byte codes has been given to show that these codes are more economical in terms of their transmission/storage than the previously considered i-spotty-byte codes [4,5].

Acknowledgment. The author would like to thank her spouse Prof.(Dr.) Arihant Jain for his constant support and encouragement for pursuing research.

References

[1] E. Fujiwara, Codes for dependable systems, Wiley Interscience, 2006.

[2] S. Jain, *Irregular-spotty-byte error control codes*, World Applied Sciences Journal, 30 (2014), 1993-1999.

[3] S. Jain, *On weight enumerators of i-spotty-byte codes*, World Applied Sciences Journal, 30 (2014), 1985-1992.

[4] S. Jain, *Code design of i-spotty-byte error correcting codes*, Alexendria Journal of Mathematics, 5 (2015), 20-42.

[5] S. Jain, *Uniform i-spotty-byte error control codes*, Journal of Combinatorial Mathematics and Combinatorial Computing, 100 (2017), 77-95.

[6] K. Suzuki, T. Kashiyama and E. Fujiwara, *A General class of M-spotty Byte Error Control Codes*, IEICE Trans. Fundamentals, Vol.-E90-A, No.7, July (2007).

[7] K. Suzuki, T. Kashiyama and E. Fujiwara, *Complex m-spotty Byte Error Control Codes*, Por. IEEE Inform. Theory Workshop:211-215, 2005.

[8] K. Suzuki, M. Shimizu, T. Kashiyama and E. Fujiwara, *A Class of Error Control Codes for m-spotty Byte Errors occured in a limited number of Bytes*, Proc. IEEE Int. Symp. on Information Theory:2109-2113, 2005.