IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2023EAL2030

Publicized:2023/08/16

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

LETTER

On the First Separating Redundancy of Array LDPC Codes*

Haiyang LIU[†a)], ***Nonmember and*** **Lianrong MA**[††b)], ***Member***

SUMMARY Given an odd prime q and an integer $m \leq q$, a binary $mq \times q^2$ quasi-cyclic parity-check matrix $\boldsymbol{H}(m,q)$ can be constructed for an array low-density parity-check (LDPC) code $C(m,q)$. In this letter, we investigate the first separating redundancy of $C(m,q)$. We prove that $\boldsymbol{H}(m,q)$ is 1-separating for any pair of (m,q), from which we conclude that the first separating redundancy of $C(m,q)$ is upper bounded by mq. Then we show that our upper bound on the first separating redundancy of $C(m,q)$ is tighter than the general deterministic and constructive upper bounds in the literature. For $m = 2$, we further prove that the first separating redundancy of $C(2,q)$ is $2q$ for any odd prime q. For $m \geq 3$, we conjecture that the first separating redundancy of $C(m,q)$ is mq for any fixed m and sufficiently large q.

key words: *Array LDPC codes, separating matrix, separating redundancy*

1. Introduction

The separating redundancy is an important concept in understanding the error-and-erasure decoding of linear block codes over a communication channel [1],[2]. Despite the importance, it is intractable to calculate the separating redundancy of a linear block code in general. Even deriving the meaningful bounds on the separating redundancy is a difficult problem. If we restrict ourselves to codes with algebraic structures, however, the investigation on the separating redundancy through analytic methods becomes possible. To date, several works have studied the separating redundancy for some classes of codes or for some specific codes [3]-[9].

Array low-density parity-check (LDPC) codes, proposed by Fan [10], are an important class of algebraic LDPC codes. An array LDPC code $C(m,q)$ can be described by a binary $mq \times q^2$ quasi-cyclic (QC) parity-check matrix $\boldsymbol{H}(m,q)$, where q is an odd prime and m is an integer satisfying $m \leq q$. In recent years, array LDPC codes have received a lot of interest. It has been shown that these codes have amicable performances, which make them suitable for practical applications. In addition, thanks to their nice algebraic properties, several structural parameters of array LDPC codes have been theoretically analyzed and determined in the previous works [11]-[21], which is helpful in understanding the code performances.

In this letter, we consider the separating property of array LDPC codes. Using analytical approaches, we prove that $\boldsymbol{H}(m,q)$ is 1-separating for any pair of (m,q), which indicates that the first separating redundancy of $C(m,q)$ is upper bounded by mq, the number of rows in $\boldsymbol{H}(m,q)$. By calculation, we show that our upper bound on the first separating redundancy of $C(m,q)$ is tighter than the general deterministic and constructive upper bounds in the literature. On the other hand, it is known from the previous results [2],[5] that the first separating redundancy of $C(m,q)$ is lower bounded by mq if the minimum distance of $C^{\perp}(m,q)$, the dual of $C(m,q)$, is equal to q. (Note that the minimum distance of $C^{\perp}(m,q)$ is at most q, since each row of $\boldsymbol{H}(m,q)$ is of weight q.) For $m = 2$, we further prove that the minimum distance of $C^{\perp}(2,q)$ is q for any odd prime q, which suggests that the first separating redundancy of $C(2,q)$ is $2q$. Through numerical observation, we conjecture that the minimum distance of $C^{\perp}(m,q)$ is q and the first separating redundancy of $C(m,q)$ is mq for any fixed $m \geq 3$ and sufficiently large q.

2. Preliminaries

In this section, we introduce the concepts of separating matrix and separating redundancy. We also review the properties of array LDPC codes that will be investigated in this work.

2.1 Separating redundancy and related concepts

Let C be an $[n,k,d]$ linear code over $\mathbb{F}_q$, where $\mathbb{F}_q$ is the finite field with q elements, n, k, and d are the length, dimension, and minimum distance of C, respectively. Suppose C is described by an $m \times n$ parity-check matrix $\boldsymbol{H}$, whose rows span the dual code $C^{\perp}$. Note that $\boldsymbol{H}$ may contain redundant rows, which suggests that $m \geq \mathrm{rank}(\boldsymbol{H}) = n - k$, where $\mathrm{rank}(\boldsymbol{H})$ is the rank of $\boldsymbol{H}$ over $\mathbb{F}_q$.

Let $\mathcal{I} = \{1, \cdots, n\}$ and $\mathcal{J} = \{1, \cdots, m\}$ be the indices for the columns and rows of $\boldsymbol{H}$, respectively. Suppose $\mathcal{S} \subseteq \mathcal{I}$ and $\mathcal{T} \subseteq \mathcal{J}$. Let $\boldsymbol{H}_{\mathcal{S}}^{\mathcal{T}} = [h_{j,i}]$, where $i \in \mathcal{S}$ and $j \in \mathcal{T}$. By inspection, we know that $\boldsymbol{H}_{\mathcal{S}}^{\mathcal{T}}$ is a submatrix of $\boldsymbol{H}$ with size $|\mathcal{T}| \times |\mathcal{S}|$, where $|\mathcal{S}|$ (resp., $|\mathcal{T}|$) is the size of the set $\mathcal{S}$ (resp., $\mathcal{T}$). In the rest discussions, we always assume that $|\mathcal{S}| \leq d - 1$.

Let $\boldsymbol{x} = [x_i]$ be a vector of length n over $\mathbb{F}_q$. The *Hamming weight* (in brief, *weight*) of $\boldsymbol{x}$ is

[†]The author is with the Institute of Microelectronics, Chinese Academy of Sciences, Beijing 100029, China.

[††]The author is with the Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China.

*This work is supported by the National Natural Science Foundation of China (NSFC) under Grant 62271482.

a) E-mail: liuhaiyang@ime.ac.cn

b) E-mail: malian@tsinghua.edu.cn

$$\mathrm{wt}(\boldsymbol{x}) = |\{i \in \mathcal{I} : x_i \neq 0\}|. \quad (1)$$

For a set $\mathcal{S} \subseteq \mathcal{I}$, let $\boldsymbol{x}_{\mathcal{S}}$ be the vector of length $|\mathcal{S}|$ obtained by deleting the entries whose indices are in the set $\bar{\mathcal{S}} = \mathcal{I} \backslash \mathcal{S}$ from $\boldsymbol{x}$. Define

$$C_{\bar{\mathcal{S}}} = \{\boldsymbol{c}_{\bar{\mathcal{S}}} : \boldsymbol{c} \in C\}, \quad (2)$$

which is the code C punctured on $\mathcal{S}$. In other words, $C_{\bar{\mathcal{S}}}$ contains all the vectors obtained by deleting the entries indexed by $\mathcal{S}$ from codewords in C.

Define

$$\hat{\mathcal{S}} = \{j \in \mathcal{J} : h_{j,i} = 0, \ \forall\, i \in \mathcal{S}\}, \quad (3)$$

and

$$\boldsymbol{H}(\mathcal{S}) = \boldsymbol{H}^{\hat{\mathcal{S}}}_{\bar{\mathcal{S}}}. \quad (4)$$

We have $\mathrm{rank}(\boldsymbol{H}(\mathcal{S})) \leq n - k - |\mathcal{S}|$ [2]. If $\mathrm{rank}(\boldsymbol{H}(\mathcal{S})) = n - k - |\mathcal{S}|$, $\boldsymbol{H}(\mathcal{S})$ is a parity-check matrix of the code $C_{\bar{\mathcal{S}}}$ in (2). In this case, we say that $\boldsymbol{H}$ *separates* the set $\mathcal{S}$.

Definition 1 ([2]): Suppose $\boldsymbol{H}$ is a parity-check matrix of an $[n, k, d]$ linear code C over $\mathbb{F}_q$ and $\mathcal{I}$ is the column index set of $\boldsymbol{H}$. If $\boldsymbol{H}$ separates every subset of $\mathcal{I}$ with size $1, 2, \cdots, l$, then $\boldsymbol{H}$ is said to be l-separating, where l is a positive integer such that $l \leq d - 1$.

It can be shown that there always exists an l-separating parity-check matrix for any linear code C with minimum distance d and any positive integer $l \leq d - 1$ [1]-[3]. From the practical point of view, it is desirable to make the number of rows of a parity-check matrix as small as possible in order to maintain reasonable complexity.

Definition 2 ([2]): The *l-th separating redundancy* of the code C, denoted by $s_l(C)$, is defined as the minimum number of rows of an l-separating parity-check matrix of C.

In this letter, we consider the first separating redundancy. The following lower bound is needed in our discussion.

Lemma 1 ([2],[5]): For an $[n, k]$ linear code C,

$$s_1(C) \geq \lceil \frac{n}{n - d^{\perp}}(n - k - 1) \rceil, \quad (5)$$

where $d^{\perp}$ is the minimum distance of $C^{\perp}$.

2.2 Array LDPC Codes

Let q be an odd prime and $m \leq q$ an integer. An array LDPC code $C(m, q)$ [10] is a binary linear code specified by the following $mq \times q^2$ QC parity-check matrix:

$$\boldsymbol{H}(m, q) = \begin{bmatrix} \boldsymbol{H}_1 \\ \boldsymbol{H}_2 \\ \vdots \\ \boldsymbol{H}_m \end{bmatrix}$$

$$= \begin{bmatrix} \boldsymbol{I}_q & \boldsymbol{I}_q & \boldsymbol{I}_q & \cdots & \boldsymbol{I}_q \\ \boldsymbol{I}_q & \boldsymbol{P} & \boldsymbol{P}^2 & \cdots & \boldsymbol{P}^{q-1} \\ & \vdots & & \vdots & \\ \boldsymbol{I}_q & \boldsymbol{P}^{m-1} & \boldsymbol{P}^{2(m-1)} & \cdots & \boldsymbol{P}^{(m-1)(q-1)} \end{bmatrix}, \quad (6)$$

where $\boldsymbol{I}_q$ is a $q \times q$ identity matrix, $\boldsymbol{P} = \begin{bmatrix} \boldsymbol{0} & 1 \\ \boldsymbol{I}_{q-1} & \boldsymbol{0}^{\mathrm{T}} \end{bmatrix}$, and $\boldsymbol{0}$ is the all-zero row vector of length $q - 1$. It is evident that $\boldsymbol{H}(m, q)$ is regular, and each column (resp., row) of $\boldsymbol{H}(m, q)$ is of weight m (resp., q).

Lemma 2 ([11]): We have $\mathrm{rank}(\boldsymbol{H}(m, q)) = m(q-1) + 1$.

The row space of $\boldsymbol{H}(m, q)$ is the dual code of $C(m, q)$, denoted by $C^{\perp}(m, q)$. By Lemma 2, the dimension of $C^{\perp}(m, q)$ is $m(q - 1) + 1$.

3. Main Results

In this section, we provide our main theoretical results. First, let us prove the following lemma.

Lemma 3: Suppose $\boldsymbol{H}'(m, q)$ is the matrix obtained from $\boldsymbol{H}(m, q)$ in (6) by removing a row from each submatrix $\boldsymbol{H}_j, j = 2, \cdots, m$. It holds that $\mathrm{rank}(\boldsymbol{H}'(m, q)) = m(q - 1) + 1$.

Proof: We only need to show that each removed row is a linear combination of some rows in $\boldsymbol{H}'(m, q)$. First let us consider $j = 2$. Suppose $\boldsymbol{h}$ is the row in $\boldsymbol{H}_2$ that is removed from $\boldsymbol{H}(m, q)$ to obtain $\boldsymbol{H}'(m, q)$. By inspection, we know that the sum of the $2q$ rows in $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ is the all-zero vector. As a consequence, $\boldsymbol{h}$ is the sum of the $2q - 1$ rows, the q rows in $\boldsymbol{H}_1$ as well as the $q - 1$ rows in $\boldsymbol{H}_2$ except for $\boldsymbol{h}$, each of which is a row in $\boldsymbol{H}'(m, q)$. The same is also true for $j = 3, \cdots, m$, and the lemma is proved. □

Remark 1: By Lemma 3, we know that $\boldsymbol{H}'(m, q)$ is also a parity-check matrix of $C(m, q)$. Moreover, since $\boldsymbol{H}'(m, q)$ has $m(q - 1) + 1$ rows, it is a generator matrix of $C^{\perp}(m, q)$.

Theorem 1: The parity-check matrix $\boldsymbol{H}(m, q)$ is 1-separating.

Proof: Consider the i-th column of $\boldsymbol{H}(m, q)$ and let $\mathcal{S} = \{i\}$, where $1 \leq i \leq q^2$. By (6), we know that there are m rows of $\boldsymbol{H}(m, q)$, one row in each $\boldsymbol{H}_j (j = 1, \cdots, m)$, whose i-th column is 1. We perform a row permutation on $\boldsymbol{H}(m, q)$ such that these m rows are on the top of the obtained matrix, which is denoted by $\boldsymbol{H}$. In other words, all the first m entries of the i-th column of $\boldsymbol{H}$ are one. By Lemma 3, we conclude that the first row as well as the last $m(q - 1)$ rows of $\boldsymbol{H}$ are $m(q-1)+1$ linearly independent rows. This suggests that the last $m(q - 1)$ rows of $\boldsymbol{H}$ are linearly independent. Therefore, we have $\mathrm{rank}(\boldsymbol{H}(\mathcal{S})) = m(q-1) = \mathrm{rank}(\boldsymbol{H}(m, q)) - 1$, where $\boldsymbol{H}(\mathcal{S})$ is defined by (4). This indicates that $\boldsymbol{H}$ separates the set $\mathcal{S}$, from which we conclude that $\boldsymbol{H}$ is 1-separating. □

The following upper bound is a direct consequence of Theorem 1 and Definition 2.

Corollary 1: Suppose $s_1(C(m, q))$ is the first separating redundancy of $C(m, q)$. Then $s_1(C(m, q)) \leq mq$.

Now we compare our upper bound in Corollary 1 with known upper bounds. In the literature, several upper bounds on the separating redundancy have been provided for general linear codes or for some specific (families of) codes [1]-[9]. However, no specific results on the separating redundancy of $C(m, q)$ are available to the best of our knowledge. Therefore, we compare our bound with general upper bounds. The

Table 1 Calculation results for the upper bounds on $s_1(C(m,q))$.

(m,q)	Parameters $[n,k,d]$	Upper Bounds			
		[2, Corollary 5]	[5, Theorem 17]	[5, Theorem 10]	Corollary 1
$(3,3)$	$[9,2,6]$	28	14	20	9
$(3,5)$	$[25,12,6]$	91	65	37	15
$(3,7)$	$[49,30,6]$	190	190	54	21
$(4,5)$	$[25,8,8]$	153	68	46	20
$(5,5)$	$[25,4,10]$	231	63	55	25

upper bounds in [2, Theorem 10] and [5, Theorem 10] suggest that the value of $s_l(C)$ tends to be a polynomial function of $n-k$ for an $[n,k,d]$ code C and a given value $l \leq d-1$ under certain conditions. The proofs of the theorems indicate that we can find l-separating parity-check matrices that meet the upper bound values through probabilistic algorithms. However, these bounds are not deterministic.

Since our upper bound in Corollary 1 is deterministic and constructive, we compare our bounds with two general deterministic and constructive upper bounds provided in [2, Corollary 5] and [5, Theorem 17], respectively, in the following.

The upper bound in [2, Corollary 5] states that

$$s_1(C) \leq \sum_{i=1}^{2} \begin{pmatrix} n-k \\ i \end{pmatrix} \tag{7}$$

holds for an $[n,k,d]$ binary code C with $d \geq 3$. For $s_1(C(m,q))$, the upper bound in (7) becomes $\frac{1}{2}(mq-m+1)(mq-m+2)$. By calculation, we conclude that $\frac{1}{2}(mq-m+1)(mq-m+2) > mq$ holds for any (m,q) such that $q \geq 3$ and $m \leq q$, which means that our upper bound in Corollary 1 is better.

The upper bound in [5, Theorem 17] suggests that

$$s_1(C) \leq \min\left\{(n-k)C_1(n,\mu,1), (n-k-1)\begin{pmatrix} n \\ 1 \end{pmatrix}\right\} \tag{8}$$

holds for an $[n,k,d]$ code C, where $C_1(n,\mu,1)$ is the covering number [22] and $\mu = \min\{d, n-k\} - 1$.

For $C(m,q)$, we have $\mu = d-1$ and $C_1(n,\mu,1) = \lceil \frac{n}{d-1} \rceil$. By inspection, we know that the upper bound in (8) becomes $(n-k)\lceil \frac{n}{d-1} \rceil$. It holds that $d \leq 2q < 2q+1$, since $[\mathbf{1},\mathbf{1},\mathbf{0},\cdots,\mathbf{0}]$ is a codeword in $C(m,q)$, where $\mathbf{1}$ (resp., $\mathbf{0}$) is the all-one (resp., all-zero) vector of length q. Thus, we have $\frac{n}{d-1} > \frac{q}{2}$, which suggests that

$$\lceil \frac{n}{d-1} \rceil \geq \lceil \frac{q}{2} \rceil = \frac{q+1}{2}.$$

As a consequence,

$$(n-k)\lceil \frac{n}{d-1} \rceil \geq \frac{(mq-m+1)(q+1)}{2}$$

$$\stackrel{\text{(a)}}{\geq} \frac{(2m+1)(q+1)}{2} > mq.$$

The above inequality (a) is due to the fact that $q \geq 3$. Again, our upper bound in Corollary 1 is better.

Table 1 lists the calculation results of our upper bound in Corollary 1 as well as the upper bounds in [2, Corollary 5] and [5, Theorem 17] for some pairs of (m,q). For comparison purposes, we also calculated the upper bound in [5, Theorem 10], which is a refined probabilistic upper bound on the separating redundancy of a linear code. It is known from the table that our upper bound is better than the known upper bounds for all the calculated cases.

We know from the previous discussion that $\boldsymbol{H}'(m,q)$ is a generator matrix of $C^{\perp}(m,q)$. Since each row of $\boldsymbol{H}'(m,q)$ has weight q, we have $d(C^{\perp}(m,q)) \leq q$. By calculation, we conclude that the lower bound in Lemma 1 becomes $s_1(C(m,q)) \geq mq$ if $d(C^{\perp}(m,q)) = q$. By Corollary 1, we have $s_1(C(m,q)) = mq$ if $d(C^{\perp}(m,q)) = q$. As a consequence, it is interesting to investigate whether the upper bound $d(C^{\perp}(m,q)) \leq q$ is tight for a pair (m,q). Towards this end, we performed some computer searches. The results are listed in Table 2.

Table 2 Some values of $d(C^{\perp}(m,q))$.

m \ q	3	5	7	11	13
2	3	5	7	11	13
3	2	5	7	*11*	*13*
4	–	4	7	*11*	*13*
5	–	2	6	*11*	*13*

For $m = 2$, we performed exhaustive computer searches for all the values of q listed in Table 2. For $m \geq 3$, we performed exhaustive computer searches for $q \leq 7$. The values are the minimum weights of the codewords in $C^{\perp}(m,q)$ for these cases.

For the remaining cases, we did not perform exhaustive computer searches due to the excessive running time. Instead, we performed non-exhaustive computer searches on the parity-check matrix of $C^{\perp}(m,q)$ using the low-weight codeword search algorithm in [23] for these cases. The values are the minimum weights of the collected codewords for these cases. (Note that a generator matrix of $C(m,q)$, which is provided in [11, p. 282] or [21, p. 1480], is a parity-check matrix of $C^{\perp}(m,q)$.) We typeset all the values of these cases in *italic*.

We can see from the results in Table 2 that $d(C^{\perp}(m,q))$ can be less than q if m is close to q for a given q. Since $C^{\perp}(m,q)$ is a subcode of $C^{\perp}(m',q)$ if $m < m'$, we have $d(C^{\perp}(m,q)) \geq d(C^{\perp}(m',q))$. In fact, we have the following result.

Theorem 2: For any odd prime q, we have $d(C^{\perp}(q,q)) = 2$, where $d(C^{\perp}(q,q))$ is the minimum distance of $C^{\perp}(q,q)$.

Proof: By inspection, we know that the following $(q-$

$1) \times q^2$ matrix

$$\begin{bmatrix} \mathbf{1} & \mathbf{1} & & & \\ & \mathbf{1} & \mathbf{1} & & \\ & & \ddots & \ddots & \\ & & & \mathbf{1} & \mathbf{1} \end{bmatrix} \tag{9}$$

is a parity-check matrix of $C^{\perp}(q,q)$, where $\mathbf{1}$ is the all-one vector of length q. Thus, $C^{\perp}(q,q)$ contains no codewords of weight 1. Because the first two columns of the matrix in (9) are identical, we have $d(C^{\perp}(q,q)) = 2$. □

On the other hand, we conclude from Table 2 that $d(C^{\perp}(2,q)) = q$ holds for all the values of q in the table. In the following, we prove that $d(C^{\perp}(2,q)) = q$ holds for *any* odd prime q. Before that, we generalize the idea in the work [12] and present a representation method for $\boldsymbol{H}(2,q)$, which is useful in our analysis.

We know from (6) that each row of $\boldsymbol{H}(2,q)$ can be divided into q blocks and each block is one vector in the set $\mathcal{Z} = \{\boldsymbol{e}_0, \boldsymbol{e}_1, \cdots, \boldsymbol{e}_{q-1}\}$, where $\boldsymbol{e}_i$ is the all-zero row vector of length q except that the $(i+1)$-th entry is 1. Construct a mapping $\phi : \mathcal{Z} \to \mathbb{Z}_q$ by $\phi(\boldsymbol{e}_i) = i$. Then every row of $\boldsymbol{H}_1$ can be represented as a row vector of length q

$$[i, i, i, \cdots, i] \tag{10}$$

for some $i \in \mathbb{Z}_q$. Similarly, every row of $\boldsymbol{H}_2$ can be represented as a row vector of length q

$$[i, i+(q-1), i+2(q-1), \cdots, i+(q-1)(q-1)] \tag{11}$$

for some $i \in \mathbb{Z}_q$. (Note that the entries in (10) or (11) form a mod-q arithmetic progression.) Then we construct a $2q \times q$ matrix $\tilde{\boldsymbol{H}}(2,q) = \begin{bmatrix} \tilde{\boldsymbol{H}}_1 \\ \tilde{\boldsymbol{H}}_2 \end{bmatrix}$, where $\tilde{\boldsymbol{H}}_1$ (resp., $\tilde{\boldsymbol{H}}_2$) is obtained by representing each row of $\boldsymbol{H}_1$ (resp., $\boldsymbol{H}_2$) in form (10) (resp., (11)). As an example, the six rows of $\tilde{\boldsymbol{H}}(2,3)$ are successively given by [0 0 0], [1 1 1], [2 2 2], [0 2 1], [1 0 2], and [2 1 0].

Recall that $\boldsymbol{H}'(2,q)$ is a generator matrix of $C^{\perp}(m,q)$, where $\boldsymbol{H}'(2,q)$ is obtained from $\boldsymbol{H}(2,q)$ by removing its last row. For convenience, we let $\tilde{\boldsymbol{H}}'(2,q) = \begin{bmatrix} \tilde{\boldsymbol{H}}_1 \\ \tilde{\boldsymbol{H}}'_2 \end{bmatrix}$, where $\tilde{\boldsymbol{H}}'_2$ is obtained from $\tilde{\boldsymbol{H}}_2$ by removing its last row.

For two column vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ of length r, $\boldsymbol{b}$ is said to be a *permutation version* of $\boldsymbol{a}$ if there exists $\sigma \in \mathcal{S}_r$ such that $b_{\sigma(i)} = a_i$ for each $1 \le i \le r$, where $\mathcal{S}_r$ is the set of all permutations on the set $\{1, 2, \cdots, r\}$. For example, $[0\ 2\ 3\ 1]^{\mathrm{T}}$ is a permutation version of $[1\ 3\ 0\ 2]^{\mathrm{T}}$.

Theorem 3: The minimum distance of $C^{\perp}(2,q)$, $d(C^{\perp}(2,q))$, is q for any odd prime q.

Proof: Suppose $\boldsymbol{v} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_q]$ is a codeword in $C^{\perp}(2,q)$, where each $\boldsymbol{v}_i (1 \le i \le q)$ is of length q. We distinguish between the following two cases.

Case 1: $\boldsymbol{v}_i \neq \mathbf{0}$ for each $1 \le i \le q$, where $\mathbf{0}$ is the all-zero vector of length q. Thus, we have $\mathrm{wt}(\boldsymbol{v}_i) \ge 1$ for each $1 \le i \le q$, which indicates that $\mathrm{wt}(\boldsymbol{v}) \ge q$ in this case.

Case 2: There exists a vector $\boldsymbol{v}_i$ that is equal to $\mathbf{0}$. Without loss of generality, we can assume $\boldsymbol{v}_1 = \mathbf{0}$. Since $\boldsymbol{H}'(2,q)$ is a generator matrix of $C^{\perp}(2,q)$, $\boldsymbol{v}$ is the sum of some rows of $\boldsymbol{H}'(2,q)$. Denote the set of indices of these rows by $\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$, where $\mathcal{V}_1$ and $\mathcal{V}_2$ are subsets of $\{1, \cdots, q\}$ and $\{q+1, \cdots, 2q-1\}$, respectively. Construct a matrix

$$\tilde{\boldsymbol{H}}'' = \begin{bmatrix} \tilde{\boldsymbol{H}}''_1 \\ \tilde{\boldsymbol{H}}''_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_{1,1} & \boldsymbol{h}_{1,2} & \cdots & \boldsymbol{h}_{1,q} \\ \boldsymbol{h}_{2,1} & \boldsymbol{h}_{2,2} & \cdots & \boldsymbol{h}_{2,q} \end{bmatrix}, \tag{12}$$

where $\tilde{\boldsymbol{H}}''_1$ (resp., $\tilde{\boldsymbol{H}}''_2$) is a submatrix of $\tilde{\boldsymbol{H}}_1$ (resp., $\tilde{\boldsymbol{H}}'_2$) whose row indices are in the set $\mathcal{V}_1$ (resp., $\mathcal{V}_2$). Since $\boldsymbol{v}_1 = \mathbf{0}$, $\tilde{\boldsymbol{H}}''_1$ and $\tilde{\boldsymbol{H}}''_2$ have the same number of rows. Denote the number by r. We have $1 \le r \le q-1$. In addition, $\boldsymbol{h}_{2,1}$ is a permutation version of $\boldsymbol{h}_{1,1}$. Assume the r entries in $\boldsymbol{h}_{1,1}$ or $\boldsymbol{h}_{2,1}$ are $j_1, j_2, \cdots, j_r$.

We claim that $\boldsymbol{h}_{2,l}$ is not a permutation version of $\boldsymbol{h}_{1,l}$ for each $2 \le l \le q$. Suppose to the contrary. By (10), we know that the r entries in $\boldsymbol{h}_{1,l}$ are $j_1, j_2, \cdots, j_r$. By (11), we know that the r entries in $\boldsymbol{h}_{2,l}$ are $j_1 + l'(q-1), j_2 + l'(q-1), \cdots, j_r + l'(q-1)$, where $l' = l-1$. If $\boldsymbol{h}_{2,l}$ is a permutation version of $\boldsymbol{h}_{1,l}$, we have

$$\begin{aligned} j_1 + j_2 + \cdots + j_r = {} & j_1 + l'(q-1) + j_2 + l'(q-1) \\ & + \cdots + j_r + l'(q-1) \end{aligned}$$

After some calculations, we get

$$rl'(q-1) = 0,$$

which is a contradiction since $1 \le r \le q-1$ and $1 \le l' \le q-1$.

Because $\boldsymbol{h}_{2,l}$ is not a permutation version of $\boldsymbol{h}_{1,l}$, we conclude that there exists at least one entry in $\boldsymbol{h}_{2,l}$ (resp., $\boldsymbol{h}_{1,l}$) that is not in $\boldsymbol{h}_{1,l}$ (resp., $\boldsymbol{h}_{2,l}$) for each $2 \le l \le q$. This implies that $\mathrm{wt}(\boldsymbol{v}_l) \ge 2$ holds for each $2 \le l \le q$. Thus, $\mathrm{wt}(\boldsymbol{v}) \ge 2(q-1) > q$ in this case.

This completes the proof. □

Due to Lemma 1, Theorem 2, and Corollary 1, we have the following result.

Corollary 2: With the above notations, it holds that $s_1(C(2,q)) = 2q$.

Remark 2: We mention that the lower bound $s_1(C(2,q)) \ge 2q$ can also be obtained from [8, Theorem 1], since $C^{\perp}(2,q)$ has dimension $2q-1$.

For any fixed $m \ge 3$, we know from the results in Table 2 that $d(C^{\perp}(m,q))$ seems to be q as q increases. We present the following conjecture.

Conjecture 1: With the above notations, $d(C^{\perp}(m,q)) = q$ and $s_1(C(m,q)) = mq$ hold for any fixed $m \ge 3$ and sufficiently large q.

4. Concluding Remarks

In this letter, we investigated $s_1(C(m,q))$, the first separating redundancy of $C(m,q)$. We proved that $\boldsymbol{H}(m,q)$ is

1-separating for any pair of (m, q) and obtained the upper bound $s_1(C(m, q)) \leq mq$. Then we showed that our upper bound on $s_1(C(m, q))$ is tighter than the general deterministic and constructive upper bounds in the literature. For $m = 2$, we further proved that $s_1(C(2, q)) = 2q$ for any odd prime q. We also presented a conjecture that $s_1(C(m, q) = mq$ for any fixed $m \geq 3$ and sufficiently large q based on numerical observation.

As a future work, we will try to prove the above-mentioned conjecture. Another question for further study is to determine the values of $s_l(C(m, q))$ or provide the meaningful bounds for $l \geq 2$.

Acknowledgments

The authors greatly appreciate the reviewer's valuable comments.

References

[1] K. A. S. Abdel-Ghaffar and J. H. Weber, "Separating erasures from errors for decoding," in Proc. IEEE Int. Symp. Inf. Theory, Toronto, Canada, pp. 215-219, Jul. 2008.

[2] K. A. S. Abdel-Ghaffar and J. H. Weber, "Parity-check matrices separating erasures from errors," IEEE Trans. Inf. Theory, vol.59, no.6, pp. 3332-3346, Jun. 2013.

[3] K. A. S. Abdel-Ghaffar and J. H. Weber, "Separating redundancy of linear MDS codes," in Proc. IEEE Int. Symp. Inf. Theory, Istanbul, Turkey, pp. 7-12, Jul. 2013.

[4] H. Liu, D. Kim, Y. Li, and A. Z. Jia, "On the separating redundancy of extended Hamming codes," in Proc. IEEE Int. Symp. Inf. Theory, Hong Kong, China, pp. 2406-2410, Jun. 2015.

[5] Y. Tsunoda, Y. Fujiwara, H. Ando, and P. Vandendriessche, "Bounds on separating redundancy of linear codes and rates of X-codes," IEEE Trans. Inf. Theory, vol. 64, no. 12, pp. 7577-7593, Dec. 2018.

[6] H. Liu, Y. Li, and L. Ma, "On the second separating redundancy of LDPC codes from finite planes," IEICE Trans. Fundamentals, vol. E101-A, no. 3, pp. 617-622, Mar. 2018.

[7] H. Liu, Y. Li, and L. Ma, "On the separating redundancy of the duals of first-order generalized Reed-Muller codes," IEICE Trans. Fundamentals, vol. E102-A, no. 1, pp. 310-315, Jan. 2019.

[8] H. Liu and L. Ma, "Further results on the separating redundancy of binary linear codes," IEICE Trans. Fundamentals, vol. E102-A, no. 10, pp. 1420-1425, Oct. 2019.

[9] H. Liu, L. Ma, and H. Zhang, "On the separating redundancy of ternary Golay codes," IEICE Trans. Fundamentals, vol. E104-A, no. 3, pp. 650-655, Mar. 2021.

[10] J. L. Fan, "Array codes as low-density parity-check codes," Proc. 2nd Int. Symp. Turbo Codes, pp. 543-546, Sep. 2000.

[11] T. Mittelholzer, "Efficient encoding and minimum distance bounds of Reed-Solomon-type array codes," Proc. IEEE Int. Symp. Information Theory (ISIT), p. 282, Jun./Jul. 2002.

[12] K. Yang and T. Helleseth, "On the minimum distance of array codes as LDPC codes," IEEE Trans. Inf. Theory, vol. 49, no. 12, pp. 3268-3271, Dec. 2003.

[13] K. Sugiyama and Y. Kaji, "On the minimum weight of simple full-length array LDPC codes," IEICE Trans. Fundamentals, vol. E91-A, no. 6, pp. 1502-1508, Jun. 2008.

[14] Y. Kaji, "On the number of minimum weight codewords of SFA-LDPC codes," In Proc. IEEE Int. Symp. Information Theory (ISIT), pp. 70-74, Jun./Jul. 2009.

[15] M. Esmaeili and M. J. Amoshahy, "On the stopping distance of array code parity-check matrices," IEEE Trans. Inf. Theory, vol. 55, no. 8, pp. 3488-3493, Aug. 2009.

[16] M. Esmaeili, M. H. Tadayon, and T. A. Gulliver, "More on the stopping and minimum distances of array codes," IEEE Trans. Commun. vol. 59, no. 3, pp. 750-757, Mar. 2011.

[17] H. Liu, L. Ma and J. Chen, "On the number of minimum stopping sets and minimum codewords of array LDPC codes," IEEE Commun. Lett., vol 14, no. 7, pp. 670-672, July 2010.

[18] H. Liu, L. He and J. Chen, "Further results on the stopping distance of array LDPC matrices," IEICE Trans. Fundamentals, vol. E95-A, no. 5, pp. 918-926, May 2012.

[19] L. Dolecek, Z. Zhang, M. J. Wainwright, V. Anantharam, and B. Nikolic, "Analysis of absorbing sets and fully absorbing sets of array-based LDPC codes," IEEE Trans. Inf. Theory, vol. 56, no. 1, pp. 181-201, Jan. 2010.

[20] E. Rosnes, M. A. Ambroze, and M. Tomlinson, "On the minimum/stopping distance of array low-density parity-check codes," IEEE Trans. Inf. Theory, vol. 60, no. 9, pp. 5204-5214, Sep. 2014.

[21] H. Liu, S. Yang, G. Deng and J. Chen, "More on the minimum distance of array LDPC codes," IEEE Commun. Lett., vol 18, no. 9, pp. 1479-1482, July 2010.

[22] C. J. Colbourn and J. H. Dinitz, Eds., Handbook of Combinatorial Designs, 2nd edition. Boca Raton, FL: Chapman & Hall/CRC, 2007.

[23] X. Y. Hu, M. P. C. Fossorier, and E. Eleftheriou, "On the computation of the minimum distance of low-density parity-check codes," Proc. IEEE International Conf. Commun., pp. 767-771, 2004.