Pure and Applied Mathematics Journal
2015; 4(2-1): 36-41
Published online February 10, 2015 (http://www.sciencepublishinggroup.com/j/pamj)
doi: 10.11648/j.pamj.s.2015040201.17
ISSN: 2326-9790 (Print); ISSN: 2326-9812 (Online)

A Note on the Rank Bounded Distance and Its Algorithms for Cyclic Codes

Takayasu Kaida[1, *], Junru Zheng[2]

[1]Department Information and Computer Sciences, Faculty of Humanity-Oriented Science and Engineering, Kinki University, Iizuka, Fukuoka, Japan
[2]Department of Human Development, Faculty of Humanities, Kyushu Women's University, Kitakyushu, Fukuoka, Japan

Email address:
kaida@m.ieice.org (T. Kaida), kaida@fuk.kindai.ac.jp (T. Kaida), zheng@kwuc.ac.jp (J. Zheng)

To cite this article:
Takayasu Kaida, Junru Zheng. A Note on the Rank Bounded Distance and Its Algorithms for Cyclic Codes. *Pure and Applied Mathematics Journal.* Special Issue: Abridging over Troubled Water---Scientific Foundation of Engineering Subjects. Vol. 4, No. 2-1, 2015, pp. 36-41. doi: 10.11648/j.pamj.s.2015040201.17

Abstract: The minimum distance for linear codes is one of the important parameters. The shift bound is a good lower bound of the minimum distance for cyclic codes, Reed-Muller codes and geometric Goppa codes. It is necessary to construct the maximum value of the independent set for the calculation of the shift bound. However, its computational complexity is very large, because the construction of the independent set is not unique. The authors proposed an algorithm for calculation of the independent set using the discrete Fourier transform in 2010. In this paper we give simple modification and new recurrent algorithms to improve the original algorithm.

Keywords: Cyclic Code, Discrete Fourier Transform, Independent Set, Proposed Algorithm, Recurrent Algorithm, Shift Bound

1. Introduction

For cyclic codes, some lower bounds are suggested. The shift bound is a well-known and good lower bound for its minimum distance [15, 14 ,9]. However, its computational complexity is very large. On the other hand, it is known that the Hamming weight of codeword is calculated by the discrete Fourier transform. The authors proposed an algorithm by discrete Fourier transform [16, 17, 3, 19, 18, 5]. We showed the complexity of proposed lower bound is less than shift bound [20,23].

In this paper, we recall some concepts and notations concerned with cyclic code, discrete Fourier transform and the independent set in Section 2, the original proposed algorithm and its simple modification, and the procedure for construction of independent set by their two algorithms are shown in Section 3, respectively. We give the improvement of proposed algorithm using the recursive function and its concrete algorithm as the 2-depth version and the recursive version and their modified versions, evaluate their computational complexities in Section 4. Finally we describe conclusions in Section 5.

2. Preliminaries

In this section we recall the construction of a cyclic code from its defining set. We consider cyclic codes with length n over a finite field $F = GF(q)$ through this paper, where $q = p^m$, p is a prime and m is a positive integer. Note that α is an n-th root of unity over F, E is an extension field of F including α and Z_n means the residue class ring of integers modulo n whose elements are often identified as the representatives themselves. We use notations $\#A$ as the cardinality of a set A and

$$A \backslash B = \{ a \in A \mid a \notin B \}.$$

Definition 1. A cyclotomic set $CS(i) \subset Z_n$ of $i \in Z_n$ over F is defined by

$$CS(i) = \{ iq^r \mid 0 \le r < n \}.$$

Especially, we write $R_s = CS(i)$ if $s = \min CS(i)$ in Z_n. ◊

We consider the defining set for a cyclic code as a cyclotomic set or an union of some cyclotomic sets. Such a defining set is called *complete (defining set)* and we consider only complete defining sets through this paper without notion.

Definition 2. Let $D = \cup_{t=1}^{r} R_{s_t}$. A cyclic code $C = C(D)$ defined by its defining set D is

$$C = \{\, c \in F^n \mid c(\alpha^i) = 0, \quad {}^{\forall} i \in D \,\},$$

where

$$c(x) = \sum_{i=0}^{n-1} c_i\, x^i$$

over E from $c = (c_0, c_1, \ldots, c_{n-1})$.⋄

Note that a cyclic code $C(D)$ is corresponding to a complete defining set D as one to one.

We denote $d(C)$ the minimum distance of C defined by

$$d(C) = \min\{\, w(c) \mid c \in C\backslash\{0\} \,\},$$

where $w(\boldsymbol{c}) = \#\{0 \le i < n | c_i \ne 0\}$ is the Hamming weight of $\boldsymbol{c} = (c_0, c_1, \ldots, c_{n-1})$. We denote $d(C(D))$ into $d(D)$ because of correspondence of $C(D)$ and D.

Next we recall some well-known properties for the Blahut theorem and so on.

Definition 3. Let $\boldsymbol{a} = (a_0, a_1, \ldots, a_{n-1})$ be a vector with length n in F^n. The discrete Fourier transform (DFT) of a is defined as

$$\boldsymbol{A} = DFT(\boldsymbol{a}) = (A_0, A_1, \ldots, A_{n-1}) \in E^n,$$

where

$$A_i = \sum_{j=0}^{n-1} a_j\, \alpha^{ij}$$

for $0 \le i < n$. ⋄

Definition 4. Let $\boldsymbol{a} = (a_0, a_1, \ldots, a_{n-1}) \in F^n$. The DFT matrix $M(\boldsymbol{a})$ of a is defined by $M(\boldsymbol{a}) = [\, m_{ij} \,]$ $(0 \le i, j < n)$ with $m_{ij} = A_{i+j \bmod n}$, i.e.,

$$M(\boldsymbol{a}) = \begin{bmatrix} A_0 & A_1 & \cdots & A_{n-1} \\ A_1 & A_2 & \cdots & A_0 \\ \vdots & \vdots & / & \vdots \\ A_{n-1} & A_0 & \cdots & A_{n-2} \end{bmatrix}$$

over E with size $n \times n$, where $(A_0, A_1, \cdots, A_{n-1}) = DFT(\boldsymbol{a})$.⋄

Lemma 1. [11,1] The Hamming weight $w(\boldsymbol{a})$ of a is corresponding to the rank of the DFT matrix $M(\boldsymbol{a})$, i.e.,

$$w(\boldsymbol{a}) = \operatorname{rank} M(\boldsymbol{a}).$$

Then we have

$$d(C) = \min\{\, \operatorname{rank} M(\boldsymbol{c}) \mid \boldsymbol{c} \in C\backslash\{0\} \,\}.$$

Definition 5. [8,10,2] If there is an $i_0 \in Z_n$ such that

$$\{\, i_0 + j + ak \mid 1 \le j < \delta, 0 \le k \le s \,\} \subseteq D,$$

where a fixed $a \in Z_n$ with $\gcd(a, n) = 1$, then the Hartmann-Tzeng(HT) bound $d_{HT}(D)$ is defined by the maximum value of $\delta + s$. ⋄

Especially, in the case of $s = 0$ at Definition 5 we call the value of δ the BCH bound $d_{BCH}(D)$.

Lemma 2. [2] For a cyclic code $C(D)$

$$d_{BCH}(D) \le d_{HT}(D) \le d(D).$$

Following in this paper, we denote the complete defining set D of given code *C(D)* and any general complete defining sets are denoted by R.

Definition 6. For a defining set $R \subset Z_n$ we define an independent set S of R as follows:

The empty set is an independent set of arbitrary defining set.

For an $x \notin R$ and an independent set $S \subseteq R$ of R, $S \cup \{x\}$ is also an independent set of R.

If S is an independent set of R and $y \in Z_n$, then

$$S + y = \{x + y | x \in S\}$$

is also an independent set of R. ⋄

Next lemma is obvious from definitions of the independent set in [7,10,9].

Lemma 3. Let $I_0 = \emptyset$. There are two sequences $\boldsymbol{x} = (x_0, x_1, x_2, \ldots, x_{s-1})$ and $\boldsymbol{y} = (y_0, y_1, y_2, \ldots, y_{s-1})$ over Z_n with $I_{i+1} = (I_i + y_i) \cup \{x_i\}$ such that $x_i \notin R$ and $I_i + y_i \subseteq R$ then

$$I_i = \left\{ x_{l-1} + \sum_{j=l}^{i-1} y_j \,\middle|\, l = 1, 2, \ldots, i \right\}$$

are independent sets of R.⋄

It is obvious that the cardinality of I_i is i.

In general choice of x and y in Definition 6 and Lemma 3 is not unique, then an independent set of R is not also unique. The maximum size for all independent sets of R is denoted $n(R)$ and the shift bound is defined by following definition.

Definition 7. The shift bound $d_S(D)$ is defined as

$$d_S(D) = \min\left\{ n(R) \,\middle|\, \begin{matrix} D \subseteq \quad {}^{\forall}R \subset Z_n, \\ R \ne Z_n \text{ is complete} \end{matrix} \right\}.$$

Theorem 1. [15,14,9] The shift bound $d_S(D)$ is a lower bound of the minimum distance $d(D)$ of a cyclic code $C(D)$ and the shift bound is greater than or equal to the HT bound, i.e.,

$$d_{HT}(D) \le d_S(D) \le d(D).$$

3. Proposed Algorithm and Construction of Independent Set

In this section we will refer a few definitions and lemmata in [1,18]

Definition 8. Let Δ be a non-zero element in E, where E is an extension field of F. We define the general matrix $G(R)$ of a defining set R as

$$G(R) = [\, g_{ij} \,]_{0 \le i, j < n}$$

with size $n \times n$, where

$$g_{ij} = \begin{cases} \Delta & \text{if} \quad i + j \pmod n \notin R, \\ 0 & \text{if} \quad i + j \pmod n \in R. \end{cases}$$

Definition 9. Let two sequences $\boldsymbol{i} = (i_1, i_2, \ldots, i_s)$ and $\boldsymbol{j} = (j_1, j_2, \ldots, j_t)$ over Z_n. We define a submatrix $G(R)^{(i,j)}$ with size $s \times t$ of a general matrix $G(R)$ with size $n \times n$ by

$$G(R)^{(i,j)} = [g_{i_u j_v}]_{1 \le u \le s, 1 \le v \le t} = \begin{bmatrix} g_{i_1 j_1} & g_{i_1 j_2} & \cdots & g_{i_1 j_t} \\ g_{i_2 j_1} & g_{i_2 j_2} & \cdots & g_{i_2 j_t} \\ \vdots & \vdots & \ddots & \vdots \\ g_{i_s j_1} & g_{i_s j_2} & \cdots & g_{i_s j_t} \end{bmatrix} \qquad (1)$$

where the elements of the submatrix $G(R)^{(i,j)}$ are fixed Δ or 0 by Definition 8. ◇

Definition 10. Let set a general matrix $G(R)$ and its submatrix $G(R)^{(\boldsymbol{i},\boldsymbol{j})}$. For a $j \in \boldsymbol{j}$, a set of row indexes with non-zero element in the j-th column is called the support(-row) set of the j-th column denoted by $Sup_{col}(j)$, i.e.,

$$Sup_{col}(j) = \{\, i_u \in \boldsymbol{i} \mid g_{i_u j} = \Delta \,\}.$$

And for a $j \in \boldsymbol{j}$, the cardinality of above set is called the column weight of the j-th column denoted by $w_{col}(j)$, i.e.,

$$w_{col}(j) = \# Sup_{col}(j).$$

Lemma 4. Let $G(R)$ be a general matrix and g_j be one of its columns. If g_j is a singleton, then the corresponding row is linearly independent from the other rows. ◇

Lemma 5. Let $G(R, \boldsymbol{i}, \boldsymbol{j})$ be a submatrix of general matrix of $G(R)$ and g_j be one of its columns. Suppose g_j is a singleton and let row i be its corresponding row. Let $G\ '(R, \boldsymbol{i}, \boldsymbol{j})$ be the matrix obtained from $G(R, \boldsymbol{i}, \boldsymbol{j})$ by erasing column j row i. Then $G(R, \boldsymbol{i}, \boldsymbol{j})$ has full rank if and only if $G\ '(R, \boldsymbol{i}, \boldsymbol{j})$ has full rank. ◇

We proposed a simple method to calculate the independent set with discrete Fourier transform [19,22].

Definition 11. For a cyclic code $C(D)$, the proposed bound (left version) $d_{p\ell}(D)$ is defined by

$$d_{p\ell}(D) = \min\{\, p_\ell(R) \mid D \subseteq \ {}^{\forall}R \subset Z_n, R \text{ is complete} \,\},$$

where $p_\ell(R)$ is the final value of r in the below Algorithm A input R.

We proposed a simple method to calculate the value $p_\ell(R)$ and the independent set with discrete Fourier transform [19,22].

Let $C(R)$ be a cyclic code with a defining set R, r be a non-negative integer. The proposed algorithm for lower bound of the rank of the DFT matrix from its general matrix is as follows.

[Algorithm A (the original: left version)]

1. Give the general matrix $G(R)$ using the defining set R. And set $G^{(\boldsymbol{u},\boldsymbol{v})} = G(R)$ with $\boldsymbol{u} = \boldsymbol{v} = Z_n$ and $r = 0$.
2. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let $Z = \{\, v \in \boldsymbol{v} \mid g_{uv} = 0 \quad {}^{\forall}u \in \boldsymbol{u} \,\}$ be a set of indexes with all-zero column. Renew
$$\boldsymbol{v} \leftarrow \boldsymbol{v} \backslash Z.$$
3. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let min_{w-col} be the minimum value of the column weight, i.e.,
$$min_{w-col} = min\{w_{col}(v) | v \in \boldsymbol{v}\} \qquad (2)$$
and, give a set of their row indexes
$$Min_w = \{v \in \boldsymbol{v} | w_{col} = min_{w-col}\} \qquad (3)$$
and its minimum value be set $v_\ell = \min\ M\,in_w$. Then renew $\boldsymbol{u}$ and $\boldsymbol{v}$ by
$$\boldsymbol{u} \leftarrow \boldsymbol{u} \backslash Sup_{col}(v_r), \qquad \boldsymbol{v} \leftarrow \boldsymbol{v} \backslash \{v_r\} \qquad (4)$$
from Definition 10, and $r \leftarrow r + 1$.
4. If the submatrix $G^{(\boldsymbol{u},\boldsymbol{v})}$ has two or more rows, i.e., $\# u > 1$, then return to (Step 2).
5. If one row matrix $G^{(\boldsymbol{u},\boldsymbol{v})}$, i.e., vector is non-zero vector, then renew $r \leftarrow r + 1$.

It is clear that the last number of r is equal to the rank of a submatrix of general matrix. From the step 2 of the proposed algorithm, we can have the 1st column of $(n - k + 1) \times n$ submatrix of the general matrix $G(R)$ such as $(\Delta^+, 0, \cdots, 0)$.

Property 1. For the general matrix $G(R)$ of a defining set R, the choice of making $(\Delta^+, 0, \cdots, 0)$ column have not influence on the rank of $G(R)$.

Proof: For the general matrix $G(R)$ of a defining set R, from Definition 8, the weight of all columns and rows of $G(R)$ are as same as R. the result are same whether we fix the 1st column or the other column on step 2 of the proposed algorithm, because the columns or rows of general matrix $G(R)$ are cyclic sequentially. ◇

The number of returning our proposed algorithm is equal to the rank of submatrix of general matrix. We show the method for construction of the independent set by the next example.

Then we give another version of the Algorithm A as follows.

[Algorithm A' (right version)]

1. Same as Algorithm A.
2. Same as Algorithm A.
3. For $G^{(\boldsymbol{u},\boldsymbol{v})}$, the values min_{w-col} and Min_w are also as same as Algorithm A. And the maximum value of Min_w is set by $v_r = \max Min_w$. Then renew $\boldsymbol{u}$ and $\boldsymbol{v}$ by
$$\boldsymbol{u} \leftarrow \boldsymbol{u} \backslash Sup_{col}(v_r), \qquad \boldsymbol{v} \leftarrow \boldsymbol{v} \backslash \{v_r\}$$
from Definition 10, and $r \leftarrow r + 1$.
4. Same as Algorithm A.
5. Same as Algorithm A.

The number of returning our proposed algorithms are lower bounds of the rank of submatrix of general matrix $G(R)$. We show the method for construction of the independent set by the next example.

Example 1. [12,21,22] For a binary cyclic code C with length $n = 73$ and its defining set $R_{1,3,5,9,13} = R_1 \cup R_3 \cup R_5 \cup R_9 \cup R_{13}$, using our above proposed algorithms. By Algorithm A, the indexes of row and column are

$$\boldsymbol{i} = (0,6,1,5,4,41,36,3,31,71,8,18,69,72)$$

and

$$\boldsymbol{j} = (0,5,33,10,40,20,6,8,11,2,13,3,4,1).$$

From [9], the two sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ can calculate as

$$x_i = i_{i+1} + j_{i+1}, (i = 0,1,\cdots,s-1)$$

and

$$\begin{cases} y_0 & = \quad 0, \\ y_j & = \quad i_{j+1} - i_j, (j = 1,\cdots,s-1). \end{cases}$$

Therefore, the independent sets are

$$\begin{aligned}
&I_0 = \emptyset, \\
&I_1 = \{0\} \cup \{0\}, \\
&I_2 = (I_1 + 6) \cup \{11\}, \\
&I_3 = (I_2 + (-5)) \cup \{34\}, \\
&I_4 = (I_3 + 4) \cup \{15\}, \\
&I_5 = (I_4 + (-1)) \cup \{44\}, \\
&I_6 = (I_5 + 37) \cup \{61\}, \\
&I_7 = (I_6 + (-5)) \cup \{42\}, \\
&I_8 = (I_7 + (-33)) \cup \{11\}, \\
&I_9 = (I_8 + 28) \cup \{42\}, \\
&I_{10} = (I_9 + 40) \cup \{0\}, \\
&I_{11} = (I_{10} + (-63)) \cup \{21\}, \\
&I_{12} = (I_{11} + 10) \cup \{21\}, \\
&I_{13} = (I_{12} + 51) \cup \{0\}, \\
&I_{14} = (I_{13} + 3) \cup \{0\}.
\end{aligned}$$

Therefore, the independent sets are

$$I_{14} = \{0,1,2,3,4,5,7,9,10,12,19,32,57,62\}.$$

Hence, we have the two sequences

$$(x_0, x_1, x_2, \ldots, x_{13}) = (0,11,34,15,44,61,42,11,42,0,21,21,0,0)$$

and

$$(y_0, y_1, y_2, \ldots, y_{13}) = \\ (0,6,-5,4,-1,37,-5,-33,28,40,-63,10,51,3)$$

By the Algorithm A' (right version), the indexes of row and column are

$$\boldsymbol{i} = (71,66,63,52,25,70,38,58,49,9,2,47,24,30,14,42)$$

and

$$\boldsymbol{j} = (72,67,27,55,34,63,22,62,69,50,57,51,71,68,70,66).$$

Therefore, the independent sets are

$$\begin{aligned}
&I_0 = \emptyset, \\
&I_1 = \{0\} \cup \{70\}, \\
&I_2 = (I_1 + (-5)) \cup \{60\}, \\
&I_3 = (I_2 + (-3)) \cup \{17\}, \\
&I_4 = (I_3 + (-11)) \cup \{34\}, \\
&I_5 = (I_4 + (-27)) \cup \{59\}, \\
&I_6 = (I_5 + 45) \cup \{60\}, \\
&I_7 = (I_6 + (-32)) \cup \{60\}, \\
&I_8 = (I_7 + 20) \cup \{47\}, \\
&I_9 = (I_8 + (-9)) \cup \{45\}, \\
&I_{10} = (I_9 + (-40)) \cup \{59\}, \\
&I_{11} = (I_{10} + (-7)) \cup \{59\}, \\
&I_{12} = (I_{11} + 45) \cup \{25\}, \\
&I_{13} = (I_{12} + (-23)) \cup \{22\}, \\
&I_{14} = (I_{13} + 6) \cup \{25\}, \\
&I_{15} = (I_{14} + (-16)) \cup \{11\}, \\
&I_{16} = (I_{15} + 28) \cup \{35\}.
\end{aligned}$$

Therefore, the independent sets are

$$I_{16} = \{3,19,20,24,25,31,32,35,36,37,39,40,41,48,64,69\}.$$

Hence, we have the two sequences

$$(x_0, x_1, x_2 \ldots, x_{15}) = \\ (70,60,17,34,59,60,60,47,45,59,59,25,22,25,11,35)$$

and

$$(y_0, y_1, y_2, \ldots, y_{15}) = \\ (0,-5,-3,-11,-27,45,-32,20,-9,-40,-7,45,-23,6,-16,28)$$

The computational complexity for the construction of independent set is very large, because it is not unique. However the complexity of our proposed algorithms are very small. Running the left and right version algorithms, we are able to choose the maximum value from both of two version algorithms.

4. More Improvement of the Algorithms

In this section we consider the recursive function for more improvement of the algorithms at the previous section. Before that a simple improvement is shown in next Algorithm B. We give recurrent algorithms Algorithm C and Algorithm C' finally. Moreover we discuss computational complexity concerned with proposed algorithms in this section [23].

4.1. Recurrent Algorithms

Definition 12. For a matrix $G^{(\boldsymbol{u},\boldsymbol{v})}$ and its Min_w from (3) at the Algorithm A as

$$Min_w = \{\, j_1, j_2, \ldots, j_z \,\}$$

with $j_1 < j_2 < \cdots < j_z$ in Z_n, we set (4) at the Algorithm A into

$$\boldsymbol{u}(j_s) \leftarrow \boldsymbol{u} \setminus Sup_{col}(j_s),\ \ \boldsymbol{v}(j_s) \leftarrow \boldsymbol{v} \setminus \{j_s\} \qquad (5)$$

for $j_s \in Min_w$. Then we define the next column weight

$next - w_{col}$ of $G^{(u,v)}$ as

$$next - w_{col} = \begin{cases} 0 & if \ \# \boldsymbol{u} = 1, \\ min_{w-col} & if \ \# u > 1, \ \# Min_w = 1, \\ next - w_{col} \ of \ G^{(\boldsymbol{u}(j_S),\boldsymbol{v}(j_S))}, \forall j_S \in Min_w & \end{cases} \qquad (6)$$

from (2), (4) and (5). Note that this function by (6) is the recursive function and multi-returns in the case of 3rd condition. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ we define a set of the column index $Next - Min_w$ as

$$Next - w_{col} = \left\{ j \, \middle| \begin{array}{l} the\ minimum\ next - w_{col} \\ in\ G^{(\boldsymbol{u}(j),\boldsymbol{v}(j)),}\ \forall j \in Min_w \end{array} \right\} \qquad (7)$$

if $\#\boldsymbol{u} > 1$ and $Next - Min_w = Min_w$ if $\#\boldsymbol{u} = 1$ from (6). ◊

[Algorithm B (the 2-depth version)]

1. Give the general matrix $G(R)$ using the defining set R. And set $G^{(\boldsymbol{u},\boldsymbol{v})} = G(R)$ with $\boldsymbol{u} = \boldsymbol{v} = Z_n$ and $r = 0$.
2. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let $Z = \{ v \in \boldsymbol{v} \mid g_{uv} = 0 \quad {}^{\forall}u \in \boldsymbol{u} \}$ be a set of indexes with all-zero column. Renew by

$$\boldsymbol{v} \leftarrow \boldsymbol{v} \backslash Z.$$

3. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let min_{w-col} be the minimum value of the column weight, i.e.,

$$min_{w-col} = \min\{ w_{col}(v) \mid v \in \boldsymbol{v} \}$$

and, give a set of their row indexes

$$Min_w = \{ v \in \boldsymbol{v} \mid w_{col}(v) = min_{w-col} \}.$$

(a) If $\#Min_w = 1$ then set $v_\ell = j_1$ from Definition 12.
(b) If $\#Min_w \geq 2$ then

$$v_\ell = min\left\{ j \, \middle| \begin{array}{l} the\ minimum\ value\ in \\ min_{w-col}\ of\ G^{(\boldsymbol{u}(j),\boldsymbol{v}(j))}, j \in Min_w \end{array} \right\}$$

Then renew u and v by

$$\boldsymbol{u} \leftarrow \boldsymbol{u} \backslash Sup_{col}(v_\ell), \boldsymbol{v} \leftarrow \boldsymbol{v} \backslash \{v_\ell\}$$

from Definition 10, and $r \leftarrow r + 1$.

4. If the submatrix $G^{(\boldsymbol{u},\boldsymbol{v})}$ has two or more rows, i.e., $\#u > 1$, then return to (Step 2).
5. If one row matrix $G^{(\boldsymbol{u},\boldsymbol{v})}$, i.e., vector is non-zero vector, then renew $r \leftarrow r + 1$.

Unfortunately this Algorithm B is more computational complexity than the Algorithm A. However we expect that Algorithm B gives more good value than results of Algorithm A and Algorithm A'.

Definition 13. For a matrix $G^{(\boldsymbol{u},\boldsymbol{v})}$, the column index $next - j$ by the recurrent calculation is defined as

$$next - j = \min \min Next - Min_w \qquad (8)$$

from Definition 12 and (7). ◊

[Algorithm C (the recursive (step-by-step) version)]

1. Give the general matrix $G(R)$ using the defining set R. And set $G^{(\boldsymbol{u},\boldsymbol{v})} = G(R)$ with $\boldsymbol{u} = \boldsymbol{v} = Z_n$ and $r = 0$.
2. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let $Z = \{ v \in \boldsymbol{v} \mid g_{uv} = 0 \quad {}^{\forall}u \in \boldsymbol{u} \}$ be a set of indexes with all-zero column. Renew by

$$\boldsymbol{v} \leftarrow \boldsymbol{v} \backslash Z.$$

3. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let min_{w-col} be the minimum value of the column weight, i.e.,

$$min_{w-col} = \min\{ w_{col}(v) \mid v \in \boldsymbol{v} \}$$

and, give a set of their row indexes Min_w and the column index v_ℓ by

$$Min_w = \{ v \in \boldsymbol{v} \mid w_{col}(v) = min_{w-col} \}$$

and $v_\ell = next - j$ from (8) in definition 13. Then renew $\boldsymbol{u}$ and $\boldsymbol{v}$ by

$$\boldsymbol{u} \leftarrow \boldsymbol{u} \backslash Sup_{col}(v_\ell), \boldsymbol{v} \leftarrow \boldsymbol{v} \backslash \{v_\ell\}$$

from Definition 10, and $r \leftarrow r + 1$.

4. If the submatrix $G^{(\boldsymbol{u},\boldsymbol{v})}$ has two or more rows, i.e., $\#\boldsymbol{u} > 1$, then return to (Step 2).
5. If one row matrix $G^{(\boldsymbol{u},\boldsymbol{v})}$, i.e., vector is non-zero vector, then renew $r \leftarrow r + 1$.

4.2. Computational Complexity of Proposed Algorithms

In [22] we gave the computational complexity of the algorithm for the shift bound and the original proposed algorithm (Algorithm A). Let n be the code length and k be the dimension of code. The computational complexity of Algorithm A is $O(k^2 n)$ and the computational complexity of algorithm for the shift bound is $O(k^{d_S} n^{d_S+1})$, where d_S is its shift bound.

In Algorithm C the general matrix $G(R)$ and the first (Step 2) give $G^{(\boldsymbol{u},\boldsymbol{v})}$ with $n - k$ rows. The depth of recurrence is $d_S, d_S - 1, \ldots, 1$ at each cycle of (Step 3). Then the computational complexity of Algorithm C is $O(k^{d_S}(n - k)^{d_S^2/2})$ at most.

We show more improvement such that record of the depth r' and general submatrix $G\ (u, v)$ at bottom of recurrence are used as Algorithm C'.

[Algorithm C' (the recursive (large-step) version)]

1. Same as Algorithm C.
2. Same as Algorithm C.
3. For $G^{(\boldsymbol{u},\boldsymbol{v})}$ let min_{w-col} be the minimum value of the column weight, i.e.,

$$min_{w-col} = \min\{ w_{col}(v) \mid v \in \boldsymbol{v} \}$$

and, give a set of their row indexes Min_w and the column index v_ℓ by

$$Min_w = \{ v \in \boldsymbol{v} \mid w_{col}(v) = min_{w-col} \}$$

and $v_\ell = next - j$ from (7) in definition 13. Record and return the depth of recurrence r' and the bottom submatrix $G^{(\boldsymbol{u}(j),\boldsymbol{v}(j))}$. Then

$$G^{(\boldsymbol{u},\boldsymbol{v})} \leftarrow G^{(\boldsymbol{u}(j),\boldsymbol{v}(j))}$$

from Definition 12, and $r \leftarrow r + r'$.

4. Same as Algorithm C.
5. Same as Algorithm C.

The computational complexity of this algorithm (Algorithm C') is $O(k^{ds}(n-k)^{ds})$ at most.

5. Conclusion

In this paper, we shown the improvement algorithms for construction of the independent set and lower bound of cyclic codes. We can obtain the larger value of independent set from our two version algorithms and discuss about their computational complexities.

We know that the calculation of lower bound or minimum distance of $C(D)$ by our proposed algorithms needs for all complete defining set R with $D \subseteq R \subset Z_n$. Then we will discuss the algorithms furthermore, for example relationship between the proposed algorithms and the Roos bound[13,4], and so on.

References

[1] E.Betti, M.Sala, "A theory for distance bounding cyclic codes", BCRI-CGC-Preprint 2007, the file of BCRI-63.pdf on http://www.bcri.ucc.ie/nonPeerReviewed.html.

[2] C.R.P.Hartmann, K.K.Tzeng, "Generalization of the BCH bound", Information and Control, Vol.20, pp.489-498, 1972.

[3] T.Kaida, J.Zheng, "A decoding method up to the Hartmann-Tzeng bound using DFT for cyclic codes", Proceedings of 2007 Asia-Pacific Conference on Communications, pp.114-117, 2007.

[4] T.Kaida, J.Zheng, "A constructing approach of the Roos bound for cyclic codes", Proceedings of 2008 International Symposium on Information Theory and its Applications, pp.395-399, 2008.

[5] T.Kaida, J.Zheng, "On improved algorithms of the proposed lower bound including well-known bounds for cyclic codes", Proceedings of 2012 International Symposium on Information Theory and its Applications, pp.446-449, 2012.

[6] T.Kaida, J.Zheng, "On some algorithms on the proposed lower bound for for cyclic codes", Proceedings of 2013 IEEE Asia-Pacific Conference on Comunications, pp.526-530, 2013.

[7] F.J.MacWilliams, N.J.A.Sloane, The Theory of Error-Correcting Codes, North-Holland, 1977.

[8] J. L. Massey, "Shift-register synthesis and BCH decoding", IEEE Transaction on Information Theory, vol.IT-15, pp.122-127, Jan., 1969.

[9] R.Pellikaan, "The shift bound for cyclic, Reed-Muller and geometric Goppa codes", Arithmetic, Geometry and Coding Theory 4, pp.155-174, Walter de Gruyter & Co, Berlin, 1996.

[10] W.W.Peterson, E.J. Weldon, Jr., "Error Correcting Codes", nd ed. Cambridge, MA: MIT Press, 1972.

[11] F.Ponchio, M.Sala, "A lower bound on the distance of cyclic codes", BCRI Preprint 2003, the file of BCRI-07.ps on http://www.bcri.ucc.ie/nonPeerReviewed.html.

[12] G.Promhouse and S.E.Tavares, "The minimum distance of all binary cyclic codes of odd lengths from 69 to 99", IEEE Transaction on Information Theory, vol.IT-24, No.4, pp.438-442, July, 1978.

[13] C.Roos, "A new lower bound for the minimum distance of a cyclic code", IEEE Transaction on Information Theory, Vol.29, No.3, pp.330-332, 1983.

[14] M.van Eupen, J.H.van Lint, "On the minimum distance of ternary cyclic codes", IEEE Transaction on Information Theory, Vol.39, No.2, pp.409-422, 1993.

[15] J.H.van Lint, R.M.Wilson, "On the minimum distance of cyclic codes", IEEE Transaction on Information Theory, Vol.32, No.1, pp.23-40, 1986.

[16] J. Zheng, T. Kaida, "On linear complexity and minimum distance for cyclic codes by defining sequence with unknown elements", The Second International Workshop on Sequence Design and its Application in Communication, CD-ROM No.55, 2005.

[17] J.Zheng, T.Kaida, "On shift bound for cyclic codes by DFT with unknown elements", Proceedings of 2007 Information Workshop on Signal Design and Its Applications In Communication, pp.409-412, 2007.

[18] J.Zheng, T.Kaida, "A note on the shift bound for cyclic codes by the DFT", IEICE Transaction of Fundamentals, Vol.E93-A, No.11, pp.1918-1921, 2010.

[19] J.Zheng, T.Kaida, "An algorithm for new lower bound of minimum distance by DFT for cyclic codes", Proceedings of 2010 International Symposium on Information Theory and its Applications, pp.846-849, 2010.

[20] J.Zheng, T.Kaida, "On relationship between proposed lower bound and shift bound for cyclic codes", The Fifth Workshop on Signal Design and Its Applications In Communications, pp.13-16, 2011.

[21] J.Zheng, T.Kaida, "The designed minimum distance of medium lengths for binary cyclic codes", Proceedings of 2012 International Symposium on Information Theory and its Applications, pp.441-445, 2012.

[22] J.Zheng, T.Kaida, "Construction of Independent Set and its Application for Designed Minimum Distance", IEICE Transaction of Fundamentals, Vol.E95-A, No.12, pp.2107-2112, 2012.

[23] J.Zheng, T.Kaida, "On relationship between proposed lower bound and well-known bounds for cyclic codes", The Sixth Workshop on Signal Design and Its Applications In Communications, pp.32-35, 2013.