The 27th Symposium on Information Theory
and Its Applications (SITA2004)
Gero, Gifu, Japan, Dec. 14–17, 2004

Relations among the Local Weight Distributions of a Linear Block Code, Its Extended Code and Its Even Weight Subcode

Kenji YASUNAGA * Toru FUJIWARA *

Abstract— Relations among the local weight distributions of a binary linear code, its extended code and its even weight subcode are presented. Using the relations, the local weight distributions of the $(127, k)$ primitive BCH codes for $k \le 50$ and their even weight subcodes are obtained from the local weight distribution of their extended codes.

Keywords— local weight distribution, primitive BCH code, extended code, even weight subcode, transitive invariant code

1 Introduction

In a binary linear code, a zero neighbor is a codeword whose Voronoi region shares a facet with that of the all-zero codeword [1]. The local weight distribution [2, 7] (or local distance profile [1, 3, 4, 6]) of a binary linear code is defined as the weight distribution of zero neighbors in the code. Knowledge of the local weight distribution of a code is valuable for the error performance analysis of the code. For example, the local weight distribution gives a tighter upper bound on error probability for soft decision decoding over AWGN channel than the usual union bound [4].

Formulas for local weight distribution are only known for Hamming codes and second-order Reed-Muller codes. An algorithm for computing the local weight distribution of cyclic codes was proposed by Mohri et al. and obtained the local weight distributions of the binary primitive BCH codes of length 63 [3]. We proposed an algorithm for computing the local weight distribution of a code which is closed under a group of permutations and obtained the local weight distributions of the $(128, k)$ extended primitive BCH codes for $k \le 50$ [6, 7]. For extended primitive BCH codes, which is closed under the affine group of permutations, our proposed algorithm has considerably smaller complexity than the algorithm in [3]. However, for cyclic codes, the complexity is not reduced. Then, the local weight distributions of the $(127, k)$ primitive BCH codes for $k \ge 36$ are still not obtained although those of the corresponding $(128, k)$ extended primitive BCH codes with $k \le 50$ are obtained. A method for obtaining the local weight distribution of a code from that of its extended code should be considered.

In this paper, we derive relations among local weight distributions of a binary linear code, its extended code and its even weight subcode. A more concrete relation for transitive invariant codes is also presented. The extended binary primitive BCH codes and Reed-Muller codes are transitive invariant codes. The local weight distributions of the $(127, k)$ binary primitive BCH codes for $36 \le k \le 50$ and their even weight subcodes are obtained by using the relations from the local weight distributions of their extended codes, which are presented in [6, 7].

*Graduate School of Information Science and Technology, Osaka University, 1-5, Yamadaoka, Suita, Osaka 565-0871, Japan. E-mail: {k-yasunaga, fujiwara}@ist.osaka-u.ac.jp

2 Local Weight Distribution

Let C be a binary (n, k) linear code. Define a mapping s from $\{0, 1\}$ to $\mathbf{R}$ as $s(0) = -1$ and $s(1) = 1$. The mapping s is naturally extended to one from $\{0, 1\}^n$ to $\mathbf{R}^n$. A zero neighbor of C is defined [1] as follows:

Definition 1 (Zero neighbor). For $\boldsymbol{v} \in C$, define $\boldsymbol{m}_0 \in \mathbf{R}^n$ as $\boldsymbol{m}_0 = \frac{1}{2}(s(\mathbf{0})+s(\boldsymbol{v}))$, where $\mathbf{0} = (0, 0, \ldots, 0)$. The codeword $\boldsymbol{v}$ is a zero neighbor if and only if

$$d_E(\boldsymbol{m}_0, s(\boldsymbol{v})) = d_E(\boldsymbol{m}_0, s(\mathbf{0})) < d_E(\boldsymbol{m}_0, s(\boldsymbol{v}')), \quad \text{for any } \boldsymbol{v}' \in C \setminus \{\mathbf{0}, \boldsymbol{v}\}, \qquad (1)$$

where $d_E(\boldsymbol{x}, \boldsymbol{y})$ is the squared Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbf{R}^n$.

The following lemma is useful to check whether a given codeword is a zero neighbor or not.

Lemma 1. [1] $\boldsymbol{v} \in C$ is a zero neighbor if and only if there does not exist $\boldsymbol{v}' \in C \setminus \{\mathbf{0}\}$ such that $\mathrm{Supp}(\boldsymbol{v}') \subsetneq \mathrm{Supp}(\boldsymbol{v})$. Note that $\mathrm{Supp}(\boldsymbol{v})$ is the set of support of $\boldsymbol{v}$, which is the set of positions of nonzero elements in $\boldsymbol{v} = (v_1, v_2, \ldots, v_n)$.

If $\boldsymbol{v} \in C$ can be represented as $\boldsymbol{v} = \boldsymbol{v}_1 + \boldsymbol{v}_2$, where $\boldsymbol{v}_1, \boldsymbol{v}_2 \in C$ and $\mathrm{Supp}(\boldsymbol{v}_1) \cap \mathrm{Supp}(\boldsymbol{v}_2) = \emptyset$, $\boldsymbol{v}$ is said to be decomposable. From Lemma 1, $\boldsymbol{v}$ is not a zero neighbor if and only if $\boldsymbol{v}$ is decomposable.

The local weight distribution is defined as follows:

Definition 2 (Local weight distribution). Let $L_w(C)$ be the number of zero neighbors with weight w in C. The local weight distribution of C is defined as the $(n+1)$-tuple $(L_0(C), L_1(C), \ldots, L_n(C))$.

On the local weight distribution, we have the following lemma.

Lemma 2. [2, 5] Let $A_w(C)$ be the number of codewords with weight w in C and d be the minimum distance of C.

$$L_w(C) = \begin{cases} A_w(C), & w < 2d, \\ 0, & w > n - k + 1. \end{cases} \qquad (2)$$

To obtain the local weight distribution, if the weight distribution is known, only $L_w(C)$ with $2d \le w \le n - k + 1$ are need to be obtained. Generally, the complexity for computing the local weight distribution is larger than that for computing the weight distribution. Therefore, the above relation is useful for obtaining the local weight distributions. Moreover, when all the weights w in a code is confined in $w < 2d$ and $w > n - k + 1$, the local weight distribution can be obtained from the weight distribution straightforwardly. For example, the local weight distribution of the $(127, k)$ primitive BCH code for $k \leq 29$ can be obtained from the weight distributions of the code.

3 Relations of Local Weight Distribution

3.1 General relation

Consider a binary linear code C of length n, its extended code C_{ex}, and its even weight subcode C_{even}. For a codeword $\boldsymbol{v} \in C$, let $\mathrm{w}(\boldsymbol{v})$ be the weight of $\boldsymbol{v}$ and $\boldsymbol{v}^{(\text{ex})}$ be the corresponding codeword in C_{ex}, that is, $\boldsymbol{v}^{(\text{ex})}$ is obtained from $\boldsymbol{v}$ by adding the over-all parity bit.

First, a relation between C and C_{ex} with respect to zero neighborhood is presented. For this, we refine the notation, decomposable codeword, and introduce even-decomposable codeword and only-odd-decomposable one.

Definition 3. A decomposable codeword (i.e., not a zero neighbor) $\boldsymbol{v}$ is said to be even-decomposable if there is a decomposition $\boldsymbol{v} = \boldsymbol{v}_1 + \boldsymbol{v}_2$ such that both $\mathrm{w}(\boldsymbol{v}_1)$ and $\mathrm{w}(\boldsymbol{v}_2)$ are even. Also, a decomposable codeword $\boldsymbol{v}$ is said to be only-odd-decomposable if both $\mathrm{w}(\boldsymbol{v}_1)$ and $\mathrm{w}(\boldsymbol{v}_2)$ are odd for all the decomposition $\boldsymbol{v} = \boldsymbol{v}_1 + \boldsymbol{v}_2$.

Any decomposable codeword of even weight is even-decomposable or only-odd-decomposable. Any odd weight decomposable codeword is neither even-decomposable nor only-odd-decomposable.

A relation between C and C_{ex} with respect to zero neighborhood is given in the following theorem.

Theorem 1. (1) For $\boldsymbol{v} \in C$ with even $\mathrm{w}(\boldsymbol{v})$, the following (a) and (b) hold.

- (a) If $\boldsymbol{v}$ is a zero neighbor in C, $\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor in C_{ex}.
- (b) Suppose that $\boldsymbol{v}$ is not a zero neighbor in C.
  - (i) If $\boldsymbol{v}$ is even-decomposable, then $\boldsymbol{v}^{(\text{ex})}$ is not a zero neighbor.
  - (ii) If $\boldsymbol{v}$ is only-odd-decomposable, then $\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor.

(2) For $\boldsymbol{v} \in C$ with odd $\mathrm{w}(\boldsymbol{v})$, the following (a) and (b) hold.

- (a) If $\boldsymbol{v}$ is a zero neighbor in C, $\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor in C_{ex}.
- (b) If $\boldsymbol{v}$ is not a zero neighbor in C, $\boldsymbol{v}^{(\text{ex})}$ is not a zero neighbor in C_{ex}.

(Proof) We only give a proof for (1).

(a) For an even weight codeword $\boldsymbol{v}$ which is a zero neighbor (i.e., indecomposable) in C, if $\boldsymbol{v}^{(\text{ex})}$ is decomposable as $\boldsymbol{v}_1^{(\text{ex})} + \boldsymbol{v}_2^{(\text{ex})}$, then $\boldsymbol{v}$ is decomposable as $\boldsymbol{v}_1 + \boldsymbol{v}_2$ because the parity bits of $\boldsymbol{v}^{(\text{ex})}$, $\boldsymbol{v}_1^{(\text{ex})}$ and $\boldsymbol{v}_2^{(\text{ex})}$ are zero. This condradicts the indecomposability of $\boldsymbol{v}$. Then, $\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor in C_{ex}.

(b) For an even weight codeword $\boldsymbol{v}$ which is not a zero neighbor in C, (i) if $\boldsymbol{v}$ is even-decomposable, for any decomposition $\boldsymbol{v}_1 + \boldsymbol{v}_2 (= \boldsymbol{v})$, $\boldsymbol{v}^{(\text{ex})}$ is decomposable as $\boldsymbol{v}_1^{(\text{ex})} + \boldsymbol{v}_2^{(\text{ex})}$ because the parity bits of $\boldsymbol{v}^{(\text{ex})}$, $\boldsymbol{v}_1^{(\text{ex})}$ and $\boldsymbol{v}_2^{(\text{ex})}$ are zero. Thus, $\boldsymbol{v}$ is not a zero neighbor in C_{ex}. (ii) In the case that $\boldsymbol{v}$ is only-odd-decomposable, suppose that $\boldsymbol{v}^{(\text{ex})}$ is decomposable as $\boldsymbol{v}_1^{(\text{ex})} + \boldsymbol{v}_2^{(\text{ex})}$. Since the parity bit of $\boldsymbol{v}^{(\text{ex})}$ is zero, the parity bit of $\boldsymbol{v}_1^{(\text{ex})}$ and $\boldsymbol{v}_2^{(\text{ex})}$ must be zero, then the weights of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are both even. This contradicts the fact that $\boldsymbol{v}$ is only-odd-decomposable. Thus, $\boldsymbol{v}$ is a zero neighbor in C_{ex}. $\square$

A similar relation as above holds between the codewords in C and C_{even}. These relations are summarized in Table 1.

Suppose that no only-odd-decomposable codeword exists in C from Theorem 1. (1) $\boldsymbol{v} \in C$ is a zero neighbor in C if and only if $\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor in C_{ex}, and (2) $\boldsymbol{v} \in C$ with even weight is a zero neighbor in C if and only if $\boldsymbol{v}$ is a zero neighbor in C_{even}. Therefore, in such a case, the local weight distributions of C_{ex} and C_{even} are obtained from that of C. Next, we give a sufficient condition where no only odd-decomposable codeword exists.

Theorem 2. If all the weights of codewords in C_{ex} are multiples of four, no only-odd-decomposable codeword exists in C.

(Proof) If $\boldsymbol{v} \in C$ with even $\mathrm{w}(\boldsymbol{v})$ is decomposed into $\boldsymbol{v}_1 + \boldsymbol{v}_2$ and both $\mathrm{w}(\boldsymbol{v}_1)$ and $\mathrm{w}(\boldsymbol{v}_2)$ are odd, the weights of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ can be represented as $\mathrm{w}(\boldsymbol{v}_1) = 4i - 1$ and $\mathrm{w}(\boldsymbol{v}_2) = 4j - 1$, where i and j are integers. Then, $\mathrm{w}(\boldsymbol{v}) = \mathrm{w}(\boldsymbol{v}_1 + \boldsymbol{v}_2) = \mathrm{w}(\boldsymbol{v}_1) + \mathrm{w}(\boldsymbol{v}_2) = (4i - 1) + (4j - 1) = 4i + 4j - 2$. This contradicts the fact that $\mathrm{w}(\boldsymbol{v})$ is a multiple of four. $\square$

For example, all the weights of codewords in the $(128, k)$ extended primitive BCH code with $k \leq 57$ are multiples of four. In the case of Reed-Muller codes, the codes in which all the weights of codewords are multiples of four can be known by using Corollary 13 of Chapter 15 in [8]. The third-order Reed-Muller code of length 128, 256 and 512 are true for the case. Alghough the local weight distribution of C_{ex} for these codes can be obtained from that of C, what we need is a method for computing the local weight distribution of C from that of C_{ex}. We will show that if C_{ex} is a transitive invariant code which does not contain only-odd-decomposable codeword, the local weight distribution of C can be obtained from that of C_{ex}.

3.2 Relation for transitive invariant codes

A Transitive invariant code is the code which is invariant under a transitive group of permutation A group of permutations is said to be transitive if for any two symbols in a codeword there exists a permutation that interchange them [9]. The extended primitive BCH codes and Reed-Muller codes are transitive invariant codes. For a transitive invariant code C_{ex}, a

Table 1: Zero neighbor property of $\boldsymbol{v}$ in original code, extended code and even weight subcode.

Theorem 1	Original code C			Extended code C_{ex}	Even weight subcode C_{even}
	$\text{w}(\boldsymbol{v})$	Decomposability	$\boldsymbol{v}$ is a zero neighbor?	$\boldsymbol{v}^{(\text{ex})}$ is a zero neighbor	$\boldsymbol{v}$ is a zero neighbor?
(1)-(a)	Even	Both	Yes	Yes	Yes
(1)-(b)-(i)	Even	Even-decomposable	No	No	No
(1)-(b)-(ii)	Even	Only-odd-decomposable	No	Yes	Yes
(2)-(a)	Odd	N/A	Yes	Yes	N/A
(2)-(b)	Odd	N/A	No	No	N/A

relation on the weight distributions of C and C_{ex} is presented in Theorem 8.15 in [9]. A similar relation holds for local weight distribution. The following lemma can be proved in a similar way as the proof of Theorem 8.15.

Lemma 3. In the $L_w(C_{\text{ex}})$ zero neighbors of C_{ex} with weight w, there are $\frac{w}{n+1}L_w(C_{\text{ex}})$ zero neighbors whose parity bit is one.

It is clear that there are $\frac{n+1-w}{n+1}L_w(C_{\text{ex}})$ zero neighbors of weight w in C_{ex} whose parity bit is zero from this lemma. The following theorem [6] is obtained from Theorem 1 and Lemma 3.

Theorem 3. If C_{ex} is a transitive invariant code of length n+1,

$$L_i(C) = \frac{i+1}{n+1}L_{i+1}(C_{\text{ex}}), \quad \text{for odd } i, \tag{3}$$

$$L_i(C) \leq \frac{n+1-i}{n+1}L_i(C_{\text{ex}}), \quad \text{for even } i. \tag{4}$$

If all the weights of codewords in a transitive invariant code C_{ex} are multiples of four, the equality of (4) holds. That is the following theorem holds.

Theorem 4. If all the weights of codewords in a transitive invariant code C_{ex} are multiples of four, we have that

$$L_i(C) = \begin{cases} \dfrac{i+1}{n+1}L_{i+1}(C_{\text{ex}}), & \text{for odd } i, \\[2ex] \dfrac{n+1-i}{n+1}L_i(C_{\text{ex}}), & \text{for even } i. \end{cases} \tag{5}$$

Therefore, the local weight distribution of the $(127, k)$ primitive BCH code for $k \leq 57$ is obtained by using the local weight distribution of the corresponding $(128, k)$ extended code.

4 Obtained Local Weight Distribution

As discussed in the previous section, the local weight distributions of the $(127, k)$ primitive BCH codes for $k \leq 57$ are obtained from that of the corresponding $(128, k)$ extended primitive BCH codes. The obtained local weight distributions are presented in Tables 2 and 3. Since the local weight distribution for the $(128, 57)$ extended primitive BCH code is unknown, only the local weight distributions for $k = 36, 43, 50$ are given in the table.

5 Conclusion

In this paper, some relations among local weight distributions of a binary linear code, its extended code and its even weight subcode are presented. The local weight distributions of the $(127, k)$ primitive BCH codes with $k = 36, 43, 50$ are obtained. If the local weight distribution of the $(128, 57)$ extended primitive BCH code is obtained, we can obtain the local weight distributions of the $(127, 57)$ primitive BCH code and the $(127, 56)$ even weight subcode.

References

[1] E. Agrell, "Voronoi regions for binary linear block codes," *IEEE Trans. Inform. Theory*, vol.42, no.1, pp.310–316, Jan. 1996.

[2] E. Agrell, "On the Voronoi Neighbor Ratio for Binary Linear Block Codes," *IEEE Trans. Inform. Theory*, vol.44, no.7, pp.3064–3072, Nov. 1998.

[3] M. Mohri, Y. Honda, and M. Morii,"A method for computing the local weight distribution of binary cyclic codes," *IEICE Trans.Fundamentals (Japanese Edition)*, vol.J86-A, no.1, pp.60–74, Jan. 2003.

[4] G. D. Forney, Jr., "Geometrically uniform codes," *IEEE Trans. Inform. Theory*, vol.37, no.5, pp.1241–1260, Sept, 1991.

[5] A. Ashikhmin and A. Barg, "Minimal vectors in linear codes," *IEEE Trans. Inform. Theory*, vol.44, no.5, pp.2010–2017, Sept. 1998.

[6] K. Yasunaga and T. Fujiwara, "An algorithm for computing the local distance profile of binary linear codes closed under a group of permutations," *IEICE Technical Report*, IT2003-47, Sept. 2003.

[7] K. Yasunaga and T. Fujiwara, "The local weight distributions of the (128,50) extended binary primitive BCH code and (128,64) Reed-Muller code," *IEICE Technical Report*, IT2004-19, Jul. 2004.

[8] F.J. MacWilliams and N.J.A. Sloane, *The theory of error-correcting codes*, North-Holland, 1977.

[9] W. W. Peterson and E. J. Weldon, Jr., *Error-correcting codes, 2nd Edition*, MIT Press, 1972.

Table 2: The local weight distributions of the $(127, k)$ primitive BCH codes for $k = 36, 43,$ and 50.

$(127, 36)$ BCH code

w	L_w
31	2,667
32	8,001
35	4,572
36	11,684
39	640,080
40	1,408,176
43	12,220,956
44	23,330,916
47	132,560,568
48	220,934,280
51	823,921,644
52	1,204,193,172
55	3,157,059,472
56	4,059,076,464
59	7,022,797,740
60	7,959,170,772
63	9,742,066,368
64	9,742,066,368
67	7,959,170,772
68	7,022,797,740
71	4,059,071,892
72	3,157,055,916
75	1,204,193,172
76	823,921,644
79	217,627,200
80	130,576,320
83	23,330,916
84	12,220,956
87	1,408,176
88	640,080

$(127, 43)$ BCH code

w	L_w
31	31,115
32	93,345
35	2,478,024
36	6,332,728
39	82,356,960
40	181,185,312
43	1,554,145,736
44	2,967,005,496
47	16,837,453,752
48	28,062,422,920
51	106,485,735,720
52	155,632,998,360
55	400,716,792,672
56	515,207,304,864
59	905,612,814,120
60	1,026,361,189,336
63	1,238,334,929,472
64	1,238,334,929,472
67	1,026,345,592,720
68	905,599,052,400
71	515,097,101,376
72	400,631,078,848
75	155,191,535,184
76	106,183,681,968
79	26,980,367,680
80	16,188,220,608
83	1,617,588,840
84	847,308,440

$(127, 50)$ BCH code

w	L_w
27	40,894
28	146,050
31	4,853,051
32	14,559,153
35	310,454,802
36	793,384,494
39	10,538,703,840
40	23,185,148,448
43	199,123,183,160
44	380,144,258,760
47	2,154,195,406,104
48	3,590,325,676,840
51	13,633,106,229,288
52	19,925,309,104,344
55	51,285,782,220,204
56	65,938,862,854,548
59	115,927,157,830,260
60	131,384,112,207,628
63	158,486,906,385,472
64	158,486,906,385,472
67	131,258,388,369,668
68	115,816,225,032,060
71	64,917,266,933,304
72	50,491,207,614,792
75	15,345,182,164,032
76	10,499,335,164,864

Table 3: The local weight distributions of the even weight subcode of the $(127, k)$ primitive BCH codes for $k = 36, 43,$ and 50.

$(127, 35)$ even weight subcode

w	L_w
32	8,001
36	11,684
40	1,408,176
44	23,330,916
48	220,934,280
52	1,204,193,172
56	4,059,076,464
60	7,959,170,772
64	9,742,066,368
68	7,022,797,740
72	3,157,055,916
76	823,921,644
80	130,576,320
84	12,220,956
88	640,080

$(127, 42)$ even weight subcode

w	L_w
32	93,345
36	6,332,728
40	181,185,312
44	2,967,005,496
48	28,062,422,920
52	155,632,998,360
56	515,207,304,864
60	1,026,361,189,336
64	1,238,334,929,472
68	905,599,052,400
72	400,631,078,848
76	106,183,681,968
80	16,188,220,608
84	847,308,440

$(127, 49)$ even weight subcode

w	L_w
28	146,050
32	14,559,153
36	793,384,494
40	23,185,148,448
44	380,144,258,760
48	3,590,325,676,840
52	19,925,309,104,344
56	65,938,862,854,548
60	131,384,112,207,628
64	158,486,906,385,472
68	115,816,225,032,060
72	50,491,207,614,792
76	10,499,335,164,864