Local Weight Distribution of the (256, 93) Third-Order Binary Reed-Muller Code

Kenji Yasunaga† Toru Fujiwara† Tadao Kasami‡

† 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
‡ Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0101, Japan

Abstract Local weight distribution is the weight distribution of minimal codewords in linear codes. We give the local weight distribution of the $(256, 93)$ third-order binary Reed-Muller code. We modify a coset partitioning algorithm by using a binary shift invariance property for Reed-Muller codes. This property reduces by about 1/256 the time complexity of the previous method.

Key words Local weight distribution, minimal codeword, Reed-Muller code, binary shift.

1 Introduction

The studies of minimal codewords in a linear code are crucial for the performance analysis of the code under maximum-likelihood (ML) decoding. The weight distribution of minimal codewords, called *local weight distribution*, is also important for ML decoding 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 [8]. An improvement for the Seguin lower bound using local weight distributions is also presented [18]. The number of minimal codewords in codes determines the complexity of gradient-like decoding of the code [12].

The local weight distributions are completely known only for certain classes of codes, including q-ary Hamming codes, the second-order binary Reed-Muller codes, and maximum-distance-separable (MDS) codes [4]. Partial results on binary Reed-Muller codes are reported in [5]. Although an efficient method to examine minimality of codeword is presented in [1], the computation for obtaining the local weight distribution is still very time-consuming for codes with large number of codewords.

The automorphism group of codes can help reduce the complexity for computing the local weight distribution [1]. To determine the local weight distribution of cyclic codes, checking minimality only for the representative codewords of cyclic permutations is sufficient to determine the local weight distribution. Using this idea, Mohri et al. obtained the local weight distributions of the binary primitive BCH codes of length 63 [14, 15].

For cyclic permutations, an efficient method for generating the cyclic representative codewords is known [3, 14]. However, for other automorphism groups of codes, say the affine group of extended binary primitive BCH codes or the general affine group of Reed-Muller codes, no efficient method for generating representatives is known. For effective use of these large automorphism groups in computing the local weight distributions of extended primitive BCH codes and Reed-Muller codes, we considered using coset partition technique [21], which was used for efficient computation of weight distribution of extended primitive BCH codes in [9].

In the algorithm in [21], a code is considered a set of cosets of a subcode, and the set of cosets are partitioned into equivalence classes with an invariance property. For any coset in the same class, the weight distribution of minimal codewords is the same. Thus, computing the weight distributions of minimal codewords only for the representative cosets is sufficient to obtain the local weight distribution of the target code. Thereby the computational complexity is reduced. Using this algorithm, we obtained the local weight distributions of the $(128, k)$ extended primitive BCH codes for $k \leq 50$ and the $(128, 64)$ third-order binary Reed-Muller code [20, 21].

In this paper, we target the $(256, 93)$ third-order binary Reed-Muller code. When we choose the $(256, 37)$ second-order Reed-Muller code as a subcode, the set of cosets of the second-order Reed-Muller code is partitioned into 32 equivalence classes [11]. We need to compute the weight distribution of minimal codewords for the 32 representative cosets. To compute the local weight distribution of this code in practical time, we modified our proposed method [21]. For each representative coset, first, we see the coset as the set of *subcosets* of a subcode. We choose the first-order Reed-Muller code as a subcode. Then we will partition the set of subcosets into equivalence classes. In partitioning the set of subcosets, *binary shifts* in the general affine group work effectively. By this modification, we reduce the time complexity for most representative cosets to 1/256. For

some remaining representative cosets for which binary shift technique is not very effective, the weight distributions of minimal codewords is determined by finding the invariant affine permutations. We can also determine them by using a method presented in [6].

2 Local weight distribution of linear codes

Let C be an (n, k) linear code over the field $\mathbf{F}_q$. A support set of a vector $\boldsymbol{v}$, which is the set of indices of nonzero elements in $\boldsymbol{v}$, is defined as $\mathrm{Supp}(\boldsymbol{v}) = \{i : v_i \neq 0\}$. If $\mathrm{Supp}(\boldsymbol{v}) \subset \mathrm{Supp}(\boldsymbol{v}')$ (respectively, $\subseteq$), we write $\boldsymbol{v} \prec \boldsymbol{v}'$ (respectively, $\preceq$).

Definition 1 (Minimal Codeword [4])**.** *A codeword $\boldsymbol{v}(\neq \mathbf{0})$ is called* minimal *in C if $\boldsymbol{v}' \preceq \boldsymbol{v}$ implies $\boldsymbol{v}' = c\boldsymbol{v}$, where $\boldsymbol{v}' \in C \setminus \{\mathbf{0}, \boldsymbol{v}\}$ and c is a nonzero constant in $\mathbf{F}_q$.*

In some papers, a minimal codeword is called a *zero neighbor* [1, 14, 21].

Henceforth we only consider the case $q = 2$, i.e. C is a binary code. The local weight distribution is defined as follows:

Definition 2 (Local weight distribution)**.** *The local weight distribution of C is the $(n + 1)$-tuple $(L_0(C), L_1(C), \ldots, L_n(C))$, where $L_w(C)$ is the number of minimal codewords with weight w in C.*

On the local weight distribution, we have the following lemma [2, 4].

Lemma 1. *Let $A_w(C)$ be the number of codewords with weight w in C and d be the minimum distance of C.*

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

When the (global) weight distribution $(A_0(C), A_1(C), \ldots, A_n(C))$ is known, only $L_w(C)$ with $2d \leq w \leq n - k + 1$ need to be known to determine the local weight distribution.

3 Coset partitioning algorithm for computing LWD

We review a coset partitioning algorithm for computing local weight distribution proposed in [21]. This algorithm works effectively for codes which have large automorphism group. A detail description of time/space complexity and effectiveness of this algorithm is presented in [21].

For a permutation π and a set of vectors D, we define the set of the permuted vectors $\pi[D]$ as

$$\pi[D] = \{\pi\boldsymbol{v} : \boldsymbol{v} \in D\}. \tag{2}$$

The automorphism group of a code C is the set of all permutations by which C is permuted into C, and denoted by $\mathrm{Aut}(C)$, i.e.,

$$\mathrm{Aut}(C) = \{\pi : \pi[C] = C\}. \tag{3}$$

Minimality of codewords is invariant under the automorphism group of the code as shown in the following theorem.

Theorem 1. For $\pi \in \mathrm{Aut}(C)$ and $\boldsymbol{v} \in C$, $\pi\boldsymbol{v}$ is a minimal codeword if $\boldsymbol{v}$ is a minimal codeword.

For a binary (n, k) linear code C and its linear subcode with dimension k', let C/C' denote the set of cosets of C' in C, that is,

$$C/C' = \{\boldsymbol{v} + C' : \boldsymbol{v} \in C\}. \tag{4}$$

Then,

$$|C/C'| = 2^{k-k'}, \quad \text{and} \quad C = \bigcup_{D \in C/C'} D. \tag{5}$$

For a coset $\boldsymbol{v} + C' \in C/C'$, the codeword $\boldsymbol{v}$ is called a *representative* codeword of the coset.

We introduce the notion of *local weight subdistribution* for cosets.

Definition 3 (Local weight subdistribution for cosets)**.** *The local weight subdistribution for a coset $D \in C/C'$ is the $(n+1)$-tuple $(LS_0(D), LS_1(D), \ldots, LS_n(D))$, where $LS_w(D)$ is the number of minimal codewords of C in D with weight w.*

From (5), the local weight distribution of C is given as the sum of the local weight subdistributions for the cosets in C/C', that is, $L_w(C) = \sum_{D \in C/C'} LS_w(D)$.

The following theorem gives an invariance property under permutations for cosets.

Theorem 2. *For $D_1, D_2 \in C/C'$, the local weight subdistribution for D_1 and that for D_2 are the same if there exists $\pi \in \mathrm{Aut}(C)$ such that $\pi[D_1] = D_2$.*

The following lemma gives the set of all permutations by which every coset in C/C' is permuted into one in C/C'.

Lemma 2. *For a linear code C and its linear subcode C', the set $\{\pi : \pi[D] \in C/C'$ for any $D \in C/C'\}$ is equal to $\mathrm{Aut}(C) \cap \mathrm{Aut}(C')$.*

$\mathrm{Aut}(C) \cap \mathrm{Aut}(C')$ (or even $\mathrm{Aut}(C)$) is generally not known. However, if a subgroup of $\mathrm{Aut}(C) \cap \mathrm{Aut}(C')$ is known, we can use the subgroup.

Definition 4. *Let Π be a subset of $\mathrm{Aut}(C) \cap \mathrm{Aut}(C')$. For $D_1, D_2 \in C/C'$, we represent $D_1 \sim_\Pi D_2$ if and only if there exists $\pi \in \Pi$ such that $\pi[D_1] = D_2$.*

Lemma 3. *The relation "$\sim_\Pi$" is an equivalence relation on C/C' if Π forms a group.*

When the set of cosets are partitioned into the equivalence classes by the relation "$\sim_\Pi$", the local weight subdistributions for cosets which belong to the same equivalence class are the same.

Based on the above discussion, we formulate the coset partitioning algorithm.

Coset partitioning algorithm for computing LWD [21]:

1) Choose a subcode C' and a subgroup Π of permutations of $\mathrm{Aut}(C) \cap \mathrm{Aut}(C')$.

2) Partition C/C' into equivalence classes with permutations in Π, and obtain the number of cosets in each equivalence class.

3) Compute the local weight subdistributions for the representative cosets in each equivalence class.

4) Sum up all the local weight subdistributions.

Computational complexity for the local weight subdistributions for representative cosets can be reduced by considering the trellis structure of C' [21].

4 LWD of the $(256, 93)$ Reed-Muller code

The r-th order binary Reed-Muller code of length 2^m, denoted by $RM(r, m)$, is the set of vectors obtained by all Boolean polynomials of degree at most r. We intend to compute the local weight distribution of $RM(3, 8)$. The general affine group $GA(m)$ is an automorphism group of $RM(r, m)$, and is the full automorphism group when $1 \le r \le m-2$ [13]. When we use the coset partitioning algorithm described in Section 3 for computing it, we can choose $RM(2, 8)$ as C'. Then a permutation set Π is $GA(8)$. The result of step 2) is known [10, 11]. $RM(3, 8)/RM(2, 8)$ is classified into 32 equivalence classes [17]. All we have to do is to compute the local weight subdistributions for 32 representative cosets. However, the expected computing time for each coset is not in practical (about 3000 hours for each coset with 2GHz Pentium4 processor by an algorithm in [21] without using the trellis structure of the code and about 1000 hours when using the trellis structure of $RM(2, 8)$). The binary shift invariance property helps reduce the time complexity, as we describe below.

The coset partitioning algorithm can be also used for computing local weight subdistribution for cosets. That is, for a coset $\boldsymbol{v} + C'$, when we choose a linear subcode C'' of C', we can see $\boldsymbol{v} + C'$ as the set of cosets $(\boldsymbol{v} + C')/C''$. If one partition $(\boldsymbol{v} + C')/C''$ into equivalence classes, the time complexity for computing the local weight subdistribution for $\boldsymbol{v} + C'$ can be reduced. However, in this case, the set of permutation Π we can use is not $\mathrm{Aut}(C) \cap \mathrm{Aut}(C') \cap \mathrm{Aut}(C'')$ but $\{\rho : \rho\boldsymbol{v} \in \boldsymbol{v} + C',\ \rho \in \mathrm{Aut}(C) \cap \mathrm{Aut}(C') \cap \mathrm{Aut}(C'')\}$.

Theorem 3. *For $E_1, E_2 \in (\boldsymbol{v}+C')/C''$, the local weight subdistribution for E_1 and that for E_2 are the same if there exists $\pi \in \{\rho : \rho\boldsymbol{v} \in \boldsymbol{v}+C', \rho \in \mathrm{Aut}(C) \cap \mathrm{Aut}(C')\}$ such that $\pi[E_1] = E_2$.*

For computing the local weight subdistribution for a coset $\boldsymbol{v} + RM(2, 8)$, we can choose $RM(1, 8)$ as a linear subcode of $RM(2, 8)$. Then we need to find a permutation set $\{\rho : \rho\boldsymbol{v} \in \boldsymbol{v} + C',\ \rho \in GA(8)\}$.

The general affine group $GA(m)$ is the set of permutations for m-variable polynomials $f(x_1, x_2, ..., x_m)$ that replace

$$f(x_1, ..., x_m) \ \text{ by } \ f\left(\sum a_{1j}x_j + b_j, ..., \sum a_{mj}x_j + b_m\right)$$

where $A = (a_{ij})$ is an invertible $m \times m$ binary matrix and $(b_1, ..., b_m)$ is a binary m-tuple. Affine permutation is called a *binary shift* when A is the identity matrix E. Let $BS(m)$ denote $GA(m)$ with $A = E$.

A set of binary shifts is a candidate for Π because a binary shift π satisfies $\pi\boldsymbol{v} \in \boldsymbol{v} + C'$ clearly. Let $C_{BS}(\boldsymbol{v})$ be a set of codewords permuted by binary shifts, that is, $C_{BS}(\boldsymbol{v}) = \{\pi\boldsymbol{v} : \pi \in BS(m)\}$.

Theorem 4. *[7, 11] Let f be an r-th order Boolean polynomial. For a coset $f + RM(r-1, m)$, $C_{BS}(f)$ is a linear subspace of $f + RM(r-1, m)$.*

Lemma 4. *[7, 11] Let f be an r-th order Boolean polynomial, and $\beta_i \in BS(m)$ be the permutation that only replaces x_i by $x_i + 1$. For a coset $f + RM(r-1, m)$, $\beta_i f$ for $1 \le i \le m$ are bases of $C_{BS}(f)$.*

Lemma 5. *For $\boldsymbol{v} + RM(r-1, m) \in RM(r, m)/RM(r-1, m)$, let $\boldsymbol{v} + \boldsymbol{v}_1 + RM(r-1, m)$ be one of cosets in $(\boldsymbol{v} + RM(r-1, m))/RM(r-2, m)$. The local weight subdistribution of $\boldsymbol{v} + \boldsymbol{v}_1 + RM(r-1, m)$ and that of $\boldsymbol{v} + \boldsymbol{v}_1 + \boldsymbol{u} + RM(r-1, m)$ for any $\boldsymbol{u} \in C_{BS}(\boldsymbol{v}_1)$ are the same.*

For a coset $\boldsymbol{v} + RM(r-1, m) \in RM(r, m)/RM(r-1, m)$, the number of cosets in $(\boldsymbol{v} + RM(r-1, m))/RM(r-2, m)$ is $2^{\frac{k_1}{k_2}}$ where $k_1 = \dim(RM(r-1, m))$ and $k_2 = \dim(RM(r-2, m))$. From Lemma 5, each coset in $(\boldsymbol{v} + RM(r-1, m))/RM(r-2, m)$ has $|C_{BS}(\boldsymbol{v})| = 2^{\dim(C_{BS}(\boldsymbol{v}))}$ equivalent cosets. Therefore, for each coset $\boldsymbol{v} + RM(r-1, m) \in RM(r, m)/RM(r-1, m)$, the number of cosets in $(\boldsymbol{v} + RM(r-1, m))/RM(r-2, m)$ we have to compute the local weight subdistribution will be reduced by $1/|C_{BS}(\boldsymbol{v})|$.

For 32 representative coset $f_i + RM(2, 8) \in RM(3, 8)/RM(2, 8)$ for $1 \le i \le 32$, we computed the dimension of $C_{BS}(f_i)$. The computation is just investigating the number of independent vectors in candidate bases, which are presented in Lemma 4. The 32 representative cosets and the dimension of $C_{BS}(f_i)$ is listed in Table 1. In this table, we follow f_i as [11, 17], and $\nu_i(3, 8)$ is the number of equivalent cosets with $f_i + RM(2, 8)$ presented in [17]. For most cases, the dimension of $C_{BS}(f_i)$ is 8, and thus the time complexity for computing the local weight subdistribution for $f_i + RM(2, 8)$ is reduced by $1/256$. For the case that $i = 1, 2, 3$ ($f_1 = 0$, $f_2 = x_1x_2x_3$, $f_3 = x_1x_2x_3 + x_2x_4x_5$),

above binary shift set method is not very effective for their small $\dim(C_{BS}(f_i))$. For many of $f_i + RM(2,8)$ including those with $i \leq 3$, we can find permutations such that $\pi f_i \in f_i + RM(2,8)$ because of their simple forms of polynomials. In [6], Borissov and Manev gave another approach for determining the local weight subdistributions for four cosets (above three cosets and the coset $f_7 + RM(2,8) = x_1x_2x_7 + x_3x_4x_7 + x_5x_6x_7 + RM(2,8)$). From Theorem 7 in [6], there is no minimal codewords in $0 + RM(2,8)$, and the local weight subdistribution for $x_1x_2x_3 + RM(2,8)$ is determined immediately. For the coset $x_1x_2x_3 + x_2x_4x_5 + RM(2,m)$, the number of codewords which one should check minimality is only 2^{m+1}. Therefore the local weight subdistributions for these three cosets are determined with little computation. We give a necessary and sufficient condition for minimality of codewords in terms of Boolean polynomial for codes of length 2^m, including binary Reed-Muller codes, in Appendix. This condition is a generalization of the fact that is used in Theorem 7 in [6].

The local weight distribution of the $(256, 93)$ third-order Reed-Muller code is presented in Table 2. In the table, L_w denotes the number of minimal codewords with weight w and NL_w denotes the number of non-minimal codewords with weight w.

Table 1: The dimension of $C_{BS}(f_i)$ and $\nu_i(3,8)$ [17] for representative coset $f_i + RM(2,8) \in RM(3,8)/RM(2,8)$

i	$\dim(C_{BS}(f_i))$	$\nu_i(3,8)$
1	0	1
2	3	97 155
3	5	84 330 540
4	6	3 855 110 400
5	6	5 059 832 400
6	6	1 799 051 520
7	7	449 762 880
8	7	566 701 228 800
9	7	60 717 988 800
10	7	4 599 609 830 400
11	7	113 340 245 760
12	7	3 454 178 918 400
13	8	2 763 343 134 720
14	8	64 478 006 476 800
15	8	290 151 029 145 600
16	8	2 266 804 915 200
17	8	116 060 411 658 240
18	8	1 740 906 174 873 600
19	8	6 963 624 699 494 400
20	8	12 379 777 243 545 600
21	8	27 854 498 797 977 600
22	8	15 916 856 455 987 200
23	8	1 740 906 174 873 600
24	8	13 600 829 491 200
25	8	3 626 887 864 320
26	8	55 266 862 694 400
27	8	497 401 764 249 600
28	8	108 806 635 929 600
29	8	18 134 439 321 600
30	8	2 321 208 233 164 800
31	8	1 740 906 174 873 600
32	8	217 613 271 859 200

5 Conclusions

The local weight distribution of the $(256, 93)$ third-order binary Reed-Muller code was computed by the modified coset partitioning algorithm. We applied the coset partitioning technique to compute the local weight subdistributions for each representative cosets. Binary shifts in the general affine group is useful for partitioning subcosets into equivalence classes. The time complexity was reduced by 1/256 or more for almost all representative cosets.

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] P.E. Allard, S.G.S. Shiva, and S.E. Tavares, "A note on the decomposition of cyclic codes into cyclic classes," *Inform. Contr.* vol. 22, no. 1, pp. 100–106, Feb. 1973.

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

[5] Y. Borissov, N. Manev, and S. Nikova, "On the non-minimal codewords in binary Reed-Muller codes," Discrete Applied Mathematics, vol. 128, issue 1, pp. 65-74, May 2003.

[6] Y. Borissov and N. Manve, "Minimal codewords in linear codes," *Serdica*, 30 (2–3), pp. 303–324, 2004.

[7] Y. Desaki, T. Fujiwara, and T. Kasami, "The weight distributions of extended binary primitive BCH codes of length 128," *IEEE Trans. Inform. Theory*, vol. 43, no. 4, pp. 1364–1371, Jul. 1997.

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

[9] T. Fujiwara and T. Kasami, "The weight distribution of $(256, k)$ extended binary primitive BCH codes with $k \leq 63$ and $k \geq 207$," *IEICE Technical Report*, IT97-46, Sept. 1997.

Table 2: Local weight distribution of the $(256, 93)$ third-order binary Reed-Muller code.

w	L_w	NL_w
0	0	1
32	777 240	0
48	2 698 577 280	0
56	304 296 714 240	0
64	74 957 481 580 800	27 323 138 140
68	707 415 842 488 320	0
72	28 055 013 884 190 720	0
76	764 244 915 168 215 040	0
80	20 661 780 862 988 697 600	46 740 515 022 720
84	414 411 510 493 363 568 640	0
88	6 266 129 424 660 312 883 200	2 092 541 588 766 720
92	71 773 299 826 457 585 909 760	0
96	627 671 368 441 418 233 282 560	175 220 907 231 460 776
100	4 208 996 769 021 096 823 357 440	896 295 872 432 701 440
104	21 729 928 024 588 603 285 831 680	17 649 688 833 056 931 840
108	86 666 048 822 136 825 068 912 640	205 538 965 619 138 887 680
112	267 785 773 787 841 625 294 110 720	2 443 557 443 225 474 732 800
116	642 456 218 534 940 726 012 149 760	21 242 190 246 763 906 990 080
120	1 198 819 482 820 829 207 341 301 760	140 350 571 829 684 449 787 904
124	1 741 767 435 501 050 021 239 848 960	685 463 378 439 064 120 197 120
128	1 971 038 877 022 035 145 182 412 800	2 501 927 491 519 281 688 549 830
132	1 735 627 864 909 747 949 509 017 600	6 825 033 969 741 135 851 028 480
136	1 184 951 930 170 762 649 130 762 240	14 007 903 221 896 242 660 327 424
140	620 824 077 435 771 999 611 781 120	21 653 383 289 415 490 307 358 720
144	242 710 219 348 184 804 622 336 000	25 077 997 997 100 046 146 507 520
148	65 293 324 137 047 881 521 561 600	21 372 930 224 054 562 686 238 720
152	8 982 921 659 842 430 396 006 400	12 747 024 014 435 005 946 757 120
156	0	4 208 997 665 316 969 256 058 880
160	0	627 671 543 662 325 464 743 336
164	0	71 773 299 826 457 585 909 760
168	0	6 266 131 517 201 901 649 920
172	0	414 411 510 493 363 568 640
176	0	20 661 827 603 503 720 320
180	0	764 244 915 168 215 040
184	0	28 055 013 884 190 720
188	0	707 415 842 488 320
192	0	74 984 804 718 940
200	0	304 296 714 240
208	0	2 698 577 280
224	0	777 240
256	0	1

[10] X. Hou, "GL(m,2) acting on R(r,m)/R(r-1,m)," *Discr. Math.*, 149, pp. 99–122, 1996.

[11] X. Hou, "Classification of R(3,8)/R(2,8)," unpublished.

[12] T.-Y. Hwang, "Decoding linear block codes for minimizing word error rate," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 733–737, Nov. 1979.

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

[14] M. Mohri, and M. Morii, "On computing the local distance profile of binary cyclic codes," *Proc. of ISITA2002*, pp. 415–418, Oct. 2002.

[15] 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.

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

[17] T. Sugita, T. Kasami, and T. Fujiwara, "The weight distribution of the third-order Reed-Muller codes of length 512," *IEEE Trans. Inform. Theory*, vol. 42, no. 5, pp. 1622–1625, Sept. 1996.

[18] T. Yasuda, K. Yasunaga, and T. Fujiwara, "Improvement of the Seguin lower bound using the local weight distribution," *Proc. of SITA2005*, pp. 435–438, Nov. 2005 (*in Japanese*).

[19] 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.

[20] 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.

[21] K. Yasunaga and T. Fujiwara, "An algorithm for computing the local weight distribution of binary linear codes closed under a group of permutations," *Proc. of ISITA2004*, pp. 846–851, Oct. 2004.

Appendix

Minimal codewords in linear code

Any binary vector of length 2^m can be expressed in terms of Boolean polynomial of m variables. Let P_m be the set of Boolean polynomials with m variables $x_1, x_2, \ldots, x_m$. For a nonnegative integer i less than 2^m, let $(b_{i1}, b_{i2}, \ldots, b_{im})$ be the standard binary expression of i such that $i = \sum_{j=1}^{m} b_{ij} 2^{m-j}$. For $f(x_1, x_2, \ldots, x_m) \in P_m$, define a vector $\boldsymbol{v}(f) = (v_0, v_1, \ldots, v_{2^m-1})$ where $v_i = f(b_{i1}, b_{i2}, \ldots, b_{im})$. A vector $\boldsymbol{v}(f)$ is the vector representation of Boolean polynomial f. Any binary vector $\boldsymbol{u}$ of length 2^m have a Boolean polynomial f such that $\boldsymbol{u} = \boldsymbol{v}(f)$. We use both vector and polynomial for representing codewords of length 2^m.

We give a necessary and sufficient condition for non-minimality in a code of length 2^m.

Lemma 6. *For $f, g \in P_m$, if $f \preceq g$ then $gf = f$. Otherwise, $gf \prec f$.*

Theorem 5. *For a code C of length 2^m, $f \in C$ is not minimal in C if and only if there exists $g \in P_m$ such that $gf \in C \setminus \{0, f\}$.*

Proof. (If part) From Lemma 6, $gf \neq f$ means $gf \prec f$. The existence of $gf \in C$ such that $gf \prec f$ leads the non-minimality of f.

(Only if part) Non-minimality of f implies the existence of $f' \in C \setminus \{0\}$ such that $f' \prec f$. Then f' is g because $f'f = f' \neq f$ from Lemma 6. □

A necessary and sufficient condition for minimality in Reed-Muller codes is given straightforwardly from Theorem 5.

Corollary 1. *A Boolean polynomial $f \in RM(r, m)$ is minimal in $RM(r, m)$ if and only if, for any $g \in RM(r, m)$, $gf \notin RM(r, m) \setminus \{0, f\}$.*