A Recursive Maximum Likelihood Decoding Algorithm for Reed-Muller Codes and Related Codes*

Tadao Kasami, †, Hitoshi Tokushige, †,

Toru Fujiwara, ††, Hiroshi Yamamoto, ††, Shu Lin ‡,

† Graduate School of Information Science,
Nara Institute of Science and Technology,
8916-5 Takayama, Ikoma, Nara, 630-01, Japan
{kasami, hitosh-t}@is.aist-nara.ac.jp

†† Department of Information and Mathematical Science,
Graduate School of Engineering Science,
Osaka University
1-3 Machikaneyama, Toyonaka, Osaka, 560, Japan
{fujiwara, hiroshi}@ics.es.osaka-u.ac.jp

‡ Department of Electrical Engineering
University of Hawai'i at Mānoa
Honolulu, Hawaii, 96822 U.S.A.
slin@spectra.eng.hawaii.edu

* Revised : May 30, 1997

Abstract

This paper shows that the recursive maximum likelihood decoding algorithm (RMLD) recently proposed in [1] has the following uniform property : if binary sectionalization is used in RMLD for a class of binary block codes, called binary-shift-invariant codes, which includes Reed Muller (RM) codes, the extended permuted primitive BCH codes and their subcodes satisfying a certain condition, then the subprocedure for each subsection is the same at each recursion level except for the input values from the lower level recursion. A condition for a binary linear block code to be binary-shift-invariant is presented, and a formula for the number of additions and comparisons, denoted γ, required by RMLD based on binary sectionalization for a linear binary-shift-invariant code, is derived. As an application, it is shown how to choose a code with the smallest γ among all linear binary-shift-invariant codes spanned by a specified number of monomials of a given degree or less. The set of codewords of a specified weight in a binary-shift-invariant code is also binary-shift-invariant. A RMLD for the set of minimum weight codewords of RM code is presented in detail and a formula for the number of additions and comparison required by the RMLD is derived.

1. Introduction

For linear block codes, a trellis-based maximum likelihood decoding (MLD) algorithm that is more effective than the Viterbi algorithm has recently been proposed [1,2]. This new MLD algorithm is recursive in nature. It is shown in Sec. 3 that this recursive MLD (RMLD) algorithm has the following uniform property : if binary sectionalization is used in RMLD for a class of binary block codes, called binary-shift-invariant codes, which includes Reed Muller (RM) codes, the extended permuted primitive BCH (EBCH) codes and their subcodes satisfying a certain condition, then the subprocedure for each subsection is the same at each recursion level except for the input values from the lower level recursion. A condition for a binary linear block code of length 2^m to be binary-shift-invariant is presented, and a formula for the number of additions and comparisons, denoted γ, required by RMLD based on binary sectionalization for a linear binary-shift-invariant code, is derived. As an application, it is shown by example how to choose a code with the smallest γ among all linear binary-shift-invariant codes spanned by a specified number of monomials of a given degree or less.

In [1], the optimum sectionalizations are shown for RM codes and EBCH codes of length 64. As is shown in Table 3, however, the ratio ρ of the number of additions and comparisons with respect to binary sectionalization to that with respect to the optimum one is very close to 1 for RM codes. For EBCH(64,24), $\rho = 275839/271745 = 1.015$. Because of uniform property and almost optimum computational complexity, binary sectionalization is very good for RM codes and their related codes.

The set of codewords of a specified weight in a binary-shift-invariant code is also binary-shift-invariant. In Sec. 4, a RMLD for the set of minimum weight codewords of RM code, denoted mw-RM code, is presented in detail and a formula for the number of additions and comparison, denoted γ, required by the RMLD is derived. In real-time processing of this RMLD for mw-RM code, no reference to the minimum weight subtrellis of the target RM code is made directly. As is shown in Table 3, the values of γ required by this RMLD are considerably smaller than those required by conventional Viterbi algorithm based on 4-section minimum weight subtrellis for mw-RM codes.

2. Preliminaries

2.1 Notations

For a positive integer n, let V_n denote the vector space of all the n-tuples over GF(2). For an n-tuple $\boldsymbol{u} = (u_1, u_2, \cdots, u_n) \in V_n$, let $wt(\boldsymbol{u})$ be the Hamming weight (or simply weight) of $\boldsymbol{u}$. For a binary block code (linear or nonlinear) of length n, let T_C denote the **minimal trellis** of C and $\Sigma(C)$ the set of all states in T_C. For a nonnegative integer x with $0 \leq x \leq n$, let $\Sigma_x(C)$ denote the **state space** at time-x. $\Sigma_0(C)$ and $\Sigma_n(C)$ consist of only the **initial state** σ_I and **final state** σ_F respectively. Let x and y be two integers such that $0 \leq x < y \leq n$. For a state $\sigma_x \in \Sigma_x(C)$ and a state $\sigma_y \in \Sigma_y(C)$, let $L(\sigma_x, \sigma_y)$ denote the set of **paths** (**binary label sequences**) from σ_x to σ_y in T_C. For an n-tuple $\boldsymbol{u} = (u_1, u_2, \cdots, u_n) \in V_n$, define $p_{x,y}\boldsymbol{u} \triangleq (u_{x+1}, u_{x+2}, \cdots, u_y)$ and $p_{x,y}(C) \triangleq \{p_{x,y}\boldsymbol{u} : \boldsymbol{u} \in C\}$. Let $C_{x,y}$ denote the subcode of C which consists of those codewords whose components are all zero except for the $y - x$ components from the $(x+1)$-th bit position to the y-th bit position. For simplicity, we use the notation $C^{\mathrm{tr}}_{x,y}$, for the truncated code $p_{x,y}(C_{x,y})$. Let $p_{x,y}(T_C)$ denote the truncated subtrellis of T_C from time-x to time-y.

For $\boldsymbol{u} \in V_n$, let $wt_{x,y}(\boldsymbol{u})$ denote the weight of the truncated vector $p_{x,y}\boldsymbol{u}$. For a linear block code A and its linear subcode $B, A/B$ denotes the set of cosets of B in A, called **partition** of A with respect to B. For two binary tuples $\boldsymbol{u} = (u_1, u_2, \cdots, u_i)$ and $\boldsymbol{v} = (v_1, v_2, \cdots, v_j)$, let $\boldsymbol{u} \circ \boldsymbol{v}$ denote the concatenation of $\boldsymbol{u}$ and $\boldsymbol{v}$, $(u_1, u_2, \cdots, u_i, v_1, v_2, \cdots, v_j)$, and for block codes A and B, $A \circ B$ denotes $\{\boldsymbol{u} \circ \boldsymbol{v} : \boldsymbol{u} \in A, \boldsymbol{v} \in B\}$. Define $\mathbf{0}_j$ as the all zero sequence of length j.

2.2 A Review of RMLD Algorithm

For a binary linear block code, the detail of RMLD algorithm is presented in [1,2]. For the set of codewords of a low weight in a binary linear block code, RMLD algorithm is studied in [3]. In this subsection, we review these results briefly.

Let C be a binary block code of length n. For integers x and y such that $0 \leq x < y \leq n$, let $L_{x,y}$ denote the family of the sets of paths (binary label sequences) from states in $\Sigma_x(C)$ to states in $\Sigma_y(C)$ in $p_{x,y}(T_C)$, that is,

$$L_{x,y} \triangleq \{L(\sigma, \sigma') : \sigma \in \Sigma_x(C) \text{ and } \sigma' \in \Sigma_y(C)\}. \tag{2.1}$$

This $L_{x,y}$ takes the key role in RMLD. Note that

$$\bigcup_{\sigma \in \Sigma_x(C), \sigma' \in \Sigma_y(C)} L(\sigma, \sigma') = p_{x,y}(C). \tag{2.2}$$

If C is linear, then it is known [4] that

$$L_{x,y} = p_{x,y}(C)/C^{\mathrm{tr}}_{x,y}. \tag{2.3}$$

If C is the set of codewords with a weight $w \neq 0$ in a binary linear code B, the following condition holds [3]:

(LW) For $\sigma \in \Sigma_x(C)$ and $\sigma' \in \Sigma_y(C)$, suppose $\boldsymbol{u}$ is in $L(\sigma, \sigma')$. Then $\boldsymbol{v} \in L(\sigma, \sigma')$, if and only if

$$wt(\boldsymbol{v}) = wt(\boldsymbol{u}), \tag{2.4}$$

$$\boldsymbol{u} + \boldsymbol{v} \in B^{\mathrm{tr}}_{x,y}. \tag{2.5}$$

It follows from (LW) that for $D \in L_{x,y}$, there are a weight w' with $0 \leq w' \leq w$ and coset F in $p_{x,y}(B)/B^{\mathrm{tr}}_{x,y}$ such that

$$D = F_{w'} \triangleq \{\boldsymbol{u} \in F : wt(\boldsymbol{u}) = w'\}, \tag{2.6}$$

and different D and D' in $L_{x,y}$ are disjoint. To obtain $L_{x,y}$ for a nonlinear code C, we need to find $p_{x,y}(C)$ and its decomposition into $L_{x,y}$.

For $D \in L_{x,y}$, let $l_{x,y}(D)$ (or simply $l(D)$) and $m_{x,y}(D)$ (or simply $m(D)$) denote the path in D which has the largest (correlation) metric (from the $(x+1)$-th bit position to the y-th bit position) in D and its metric, respectively. A table consisting of $l_{x,y}(D)$ and $m_{x,y}(D)$ for each $D \in L_{x,y}$ is called $L_{x,y}$-table. Note that for $x = 0$ and $y = n$, $L_{0,n} = \{L(\sigma_I, \sigma_F)\} = \{C\}$. When $L_{0,n}$-table is constructed, it stores the most likely codeword and its correlation metric. The RMLD algorithm for decoding C is simply a **recursive procedure** for constructing $L_{x,y}$-tables for long trellis sections from short trellis sections. Assume that $y - x \geq 2$. Let z be an integer such that $x < z < y$. For each $D_y \in L_{x,y}$, define $A_z(D_y) \triangleq \{D_z \in L_{x,z} : D_z \subseteq p_{x,z}(D_y)\}$. For each $D_z \in A_z(D_y)$, since T_C is minimal, there is exactly one set in $L_{z,y}$, denoted $\mathrm{adj}(D_z, D_y)$, such that

$$D_z \circ \mathrm{adj}(D_z, D_y) \subseteq D_y. \tag{2.7}$$

Then,

$$D_y = \bigcup_{D_z \in A_z(D_y)} D_z \circ \mathrm{adj}(D_z, D_y), \tag{2.8}$$

and

$$m(D_y) = \max_{D_z \in A_z(D_y)} \{m(D_z) + m(\mathrm{adj}(D_z, D_y))\}, \tag{2.9}$$

$$l(D_y) = l(D'_z) \circ l(\mathrm{adj}(D'_z, D_y)), \tag{2.10}$$

where the maximum of the right-hand side of (2.9) is taken by $D'_z \in A_x(D_y)$. Note that $m(D_z)$ and $l(D_z)$ are stored in the $L_{x,z}$-table and $m(\mathrm{adj}(D_z, D_y))$ and $l(\mathrm{adj}(D'_z, D_y))$ are stored in $L_{z,y}$-table. Therefore, from (2.9) and (2.10), $L_{x,y}$-table can be constructed from $L_{x,z}$- and $L_{z,y}$-tables.

The $L_{0,n}$-table for C (or decoding C) can be obtained by executing the following recursive procedure RMLD-$C(x, y)$:

(1) If $y - x \geq 2$, then choose an integer z such that $x < z < y$. Execute RMLD-$C(x, z)$ and RMLD-$C(z, y)$ to form $L_{x,z}$- and $L_{z,y}$-tables. Construct the $L_{x,y}$-table from $L_{x,z}$-table, $L_{z,y}$-table and $T_{x,y;z}$ by using (2.9) and (2.10).

(2) If $y = x + 1$, then construct $L_{x,x+1}$-table directly. △△

A straightforward method for solving (2.9) is **add-compare-select**(**ACS**) which is used in conventional Viterbi algorithm. Let $\gamma(x, y; z)$ denote the number of additions and comparisons in **ACS** for solving (2.9). Then

$$\gamma(x, y; z) = \sum_{D_y \in L_{x,y}} (2|A_z(D_y)| - 1). \tag{2.11}$$

If C is linear, then $|A_z(D_y)|$ is the same values for all $D_y \in L_{x,y}$, denoted $A_{x,y:z}$, is given [1] by

$$A_{x,y;z} = |C^{\mathrm{tr}}_{x,y}|/(|C^{\mathrm{tr}}_{x,z}| \cdot |C^{\mathrm{tr}}_{z,y}|), \tag{2.12}$$

and therefore,

$$\gamma(x, y; z) = (2A_{x,y;z} - 1)|L_{x,y}|. \tag{2.13}$$

A more effective method proposed in [1,2], called sort-add-compare-select (SACS) procedure, can be used to solve (2.9), which results in much less computations. This method is suitable for implementation by software.

3. RMLD for Reed-Muller Codes and Related Codes

We first review the representation of a binary vector in V_{2^m} by a Boolean polynomial of m variables [5].

For a positive integer μ, let P_μ denote the set of polynomials over GF(2) of μ variables $x_1, x_2, \cdots, x_\mu$. For $1 \le \nu \le \mu$, let $P_{\mu,\nu}$ be the set of polynomials of degree ν or less in P_μ, and let $P_{\mu,0} = \{0, 1\}$. Let $f = f(x_1, x_2, \cdots, x_\mu)$ be in P_μ and let $v_\mu(f)$ (or simply $v(f)$) denote the binary vector $(v_1, v_2, \cdots, v_{2^\mu}) \in V_{2^\mu}$ such that $v_{1+\sum_{j=1}^{\mu} b_j 2^{j-1}} = f(b_1, b_2, \cdots, b_\mu)$ for $(b_1, b_2, \cdots, b_\mu) \in V_\mu$. For $f \in P_\mu$, $wt(v_\mu(f))$ is also expressed as $|f|_\mu$, and for $b \in \{0, 1\}$ and $1 \le i \le \mu$, $f_{x_i=b}$ denotes the polynomial in $P_{\mu-1}$ defined as $f_{x_i=b}\,(x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_\mu) \triangleq f(x_1, \cdots, x_{i-1}, b, x_{i+1}, \cdots, x_\mu)$. Similarly, for $1 \le i_1 < i_2 < \cdots < i_h \le \mu$ and binary constants $b_1, b_2, \cdots, b_h$, $f_{x_{i_1}=b_1, x_{i_2}=b_2, \cdots, x_{i_h}=b_h}$ is defined as a polynomial in $P_{\mu-h}$. For simplicity, let $f_{b_1 b_2 \cdots b_h}$ denote $f_{x_{\mu-h+1}=b_1, x_{\mu-h+2}=b_2, \cdots, x_\mu=b_h}$. For a binary sequence $\beta = b_1 b_2 \cdots b_h$, define $j \triangleq \sum_{i=1}^{h} b_i 2^{i-1}$. Then,

$$p_{j2^{\mu-h},(j+1)2^{\mu-h}}(v_\mu(f)) = v_{\mu-h}(f_\beta). \tag{3.1}$$

For a subset $X \subseteq P_\mu$, define $v_\mu(X) = \{v_\mu(f) : f \in X\}$. Let $\mathrm{RM}_{m,r}$ denote the r-th order RM code of length 2^m. Define $n \triangleq 2^m$. Then, $\mathrm{RM}_{m,r} = v_m(P_{m,r})$. For a binary block code C of length n, there exists a subset P_C of P_m such that $C = v_m(P_C)$. C is said to be **binary-shift-invariant**, if and only if for any $f(x_1, x_2, \cdots, x_m) \in P_C$ and $(a_1, a_2, \cdots, a_m) \in V_m$, $f(x_1 + a_1, x_2 + a_2, \cdots, x_m + a_m) \in P_C$. Extended permuted binary primitive BCH codes (EBCH codes) [6] and RM codes are binary-shift-invariant[7]. For a binary-shift-invariant code C and a nonnegative integer $w \le 2^m$, $C_w(\triangleq \{\mathbf{u} \in C : wt(\mathbf{u}) = w\})$ is also binary-shift-invariant.

For $f(x_1, x_2, \cdots, x_m) \in P_m$, binary sequences $\beta = b_1 b_2 \cdots b_h$ and $\beta' = b'_1 b'_2 \cdots b'_h$ with $1 \le h \le m$, define $f'(x_1, x_2, \cdots, x_m) \triangleq f(x_1, x_2, \cdots, x_{m-h}, x_{m-h+1} + b_1 + b'_1, x_{m-h+2} + b_2 + b'_2, \cdots, x_m + b_h + b'_h)$, $j \triangleq \sum_{i=1}^{h} b_i 2^{i-1}$ and $j' \triangleq \sum_{i=1}^{h} b'_i 2^{i-1}$. From (3.1),

$$\begin{aligned} p_{j2^{m-h},(j+1)2^{m-h}}(v_m(f)) &= v_{m-h}(f_\beta) = v_{m-h}(f'_{\beta'}) \\ &= p_{j'2^{m-h},(j'+1)2^{m-h}}(v_m(f')). \end{aligned} \tag{3.2}$$

If C is binary-shift-invariant, then $f' \in P_C$ if and only if $f \in P_C$. Hence, from (3.2),

$$p_{j2^{m-h},(j+1)2^{m-h}}(C) = p_{j'2^{m-h},(j'+1)2^{m-h}}(C). \tag{3.3}$$

Similarly, we have that

$$C^{\mathrm{tr}}_{j2^{m-h},(j+1)2^{m-h}} = C^{\mathrm{tr}}_{j'2^{m-h},(j'+1)2^{m-h}}. \tag{3.4}$$

From (2.1),(3.3) and (3.4), we have the following lemma.

Lemma 1 : Suppose a linear block code C of length 2^m is binary-shift-invariant. Then for $1 \le h \le m$ and $0 \le j < 2^h$, $L_{j2^{m-h},(j+1)2^{m-h}}$ for C is the same for all j, and for $0 \le w \le 2^m$, $L_{j2^{m-h},(j+1)2^{m-h}}$ for C_w is the same for all j.

$\triangle\triangle$

Suppose that z is always chosen as $(x+y)/2$ in RMLD-$C(x,y)$ with $y-x \ge 2$. This is called binary sectionalization. The value of h in (3.2) and (3.3) denotes the level of current recursion, that is, $h = 0$ means $x = 0$ and $y = 2^m$, the 0-th level, and $h = 1$ means $(x,y) = (0, 2^{m-1})$ or $(2^{m-1}, 2^m)$, the first level. Lemma 1 implies that if C is linear and binary-shift-invariant or C is the set of codewords with weight w of a linear binary-shift-invariant code, and the binary sectionalization is adopted in RMLD-C, then the subprocedure RMLD-$C(x, y)$'s at the same level are all the same except for the return values from lower levels. This reduces the complexity of hardware implementation by sharing the same circuit to construct different tables.

For $f(x_1, x_2, \cdots, x_m) \in P_m$, define $\triangle f$ as follows :

$$\partial f \triangleq \{f(x_1+a_1, x_2+a_2, \cdots, x_m+a_m) - f(x_1, x_2, \cdots, x_m) : (a_1, a_2, \cdots, a_m) \in V_m\}. \tag{3.5}$$

If C is a binary linear code of length 2^m, then there is a subset Q_C of P_m such that C is spanned by $v_m(Q_C)$. C is binary shift-invariant if and only if for $f \in Q_C$, δf is a linear sum of polynomials in Q_C.

For two monomials $t_1 = x_{i_1} x_{i_2} \cdots x_{i_p}$ and $t_2 = x_{j_1} x_{j_2} \cdots x_{j_q}$, t_1 is said to be a subterm of t_2 or t_2 is said to be a superterm of t_1, if and only if $\{i_1, i_2, \cdots, i_p\} \subseteq \{j_1, j_2, \cdots, j_q\}$. Then, for a monomial t in P_m, ∂t is the set of subterms of t other than t itself.

Suppose that a linear code C is spanned by $v_m(M_B)$(or simply M_B), where M_B is a set of monomials in P_m. Then, C is binary-shift-invariant, if and only if the following closure condition on subterms holds:

(CS) For $t \in M_B$, all subterms of t are in M_B.

For integers h and j such that $1 \leq h \leq m$ and $0 \leq j < 2^h$, define

$$\begin{cases} x \triangleq j 2^{m-h}, \\ y \triangleq (j+1) 2^{m-h}. \end{cases} \tag{3.6}$$

Then it is easy to show that

(1) $p_{x,y}(C)$ is spanned by $M_{B,\mathbf{0}_h} \triangleq \{x_{i_1}x_{i_2}\cdots x_{i_p} \in M_B : 1 \le i_1 < i_2 < \cdots < i_p \le m-h\}$ and

(2) $C^{\mathrm{tr}}_{x,y}$ is spanned by $M_B/x_{m-h+1}x_{m-h+2}\cdots x_m \triangleq \{x_{i_1}x_{i_2}\cdots x_{i_p} : 1 \le i_1 < i_2 < \cdots < i_p \le m-h$ and $x_{i_1}x_{i_2}\cdots x_{i_p}x_{m-h+1}x_{m-h+2}\cdots x_m \in M_B\}$.

Then, it follows from (2.3) and (2.11) that for C,

$$\log_2 |L_{x,y}| = |M_{B,\mathbf{0}_h}| - |M_B/x_{m-h+1}x_{m-h+2}\cdots x_m|, \tag{3.7}$$

$$\log_2 A_{x,y;z} = |M_B/x_{m-h+1}x_{m-h+2}\cdots x_m| - 2|M_B/x_{m-h}x_{m-h+1}\cdots x_m|. \tag{3.8}$$

For $1 \le \nu \le \mu$, let $M_{\mu,\nu}$ be the set of all monomials in $P_{\mu,\nu}$. Since $\mathrm{RM}_{\mu,\nu}$ is spanned by $v_\mu(M_{\mu,\nu})$, $M_{m,r,\mathbf{0}_h} = M_{m-h,\min\{r,m-h\}}$ and $M_{m,h}/x_{m-h+1}x_{m-h+2}\cdots x_m = M_{m-r,r-h}$ (empty for $r < h$), we have the following lemma.

Lemma 2 : For integers x and y defined by (3.6) respectively,

(1)

$$p_{x,y}(\mathrm{RM}_{m,r}) = \mathrm{RM}_{m-h,\min\{r,m-h\}}, \tag{3.9}$$

(2)

$$[\mathrm{RM}_{m,r}]^{\mathrm{tr}}_{x,y} = \begin{cases} \mathrm{RM}_{m-h,r-h}, & \text{if } r \ge h, \\ \{\mathbf{0}_{2^{m-h}}\}, & \text{otherwise.} \end{cases} \tag{3.10}$$

(3) $L_{x,y}$ for $\mathrm{RM}_{m,r}$ is the same for all j such that $0 \le j < 2^h$, and

$$L_{x,y} = \begin{cases} \mathrm{RM}_{m-h,\min\{r,m-h\}}/\mathrm{RM}_{m-h,r-h}, & \text{for } r \ge h, \\ \mathrm{RM}_{m-h,\min\{r,m-h\}}/\{\mathbf{0}\}, & \text{otherwise.} \end{cases} \tag{3.11}$$

$\triangle\triangle$

If follows from (2.11),(3.10) and a well known formula [4] that for $\mathrm{RM}_{m,r}$ and $r \neq h$,

$$\log_2 A_{x,y;z} = \left(\begin{array}{c} m-h-1 \\ r-h \end{array} \right). \tag{3.12}$$

Example 1: A (64,40) subcode of $\mathrm{RM}_{6,3}$ is being considered as an inner code in a concatenated coding system for NASA's high-speed satellite communications[8]. The overall decoder for the RM subcode consists of 32 identical maximum likelihood decoding (MLD) decoders, each such decoder processes a (64,35) subcode C or its coset in parallel. In this example, we assume that a RMLD decoder based on binary sectionalization is used, and consider how to choose the (64,35) subcode C of $\mathrm{RM}_{6,3}$ to minimize the total number γ of additions and comparisons in ACS. We assume that a subset M of $M_{m,r}$ satisfies the closure condition (CS). Let $C(M)$ denote the subcode of $\mathrm{RM}_{m,r}$ spanned by M, and let t be a monomial in M whose superterms are not in M. For $1 \leq h < m$, then it follows from (3.7) and (3.8) that the difference between the values of $\log_2 |L_{0,2^{m-h}}|$ (or $\log_2 A_{0,2^{m-h};2^{m-h-1}}$) for $C(M)$ and $C(M \backslash \{t\})$ is independent of M. Let $\triangle L^{(h)}(t)$ (or $\triangle A^{(h)}(t)$) denote the difference. Table 1 shows $\triangle L^{(h)}(t)$ with $1 \leq h \leq 5$ and $\triangle A^{(h)}(t)$ with $0 \leq h \leq 5$ for $t \in M_{6,3} \backslash M_{6,1}$. For a subset $\triangle M$ of M consisting of different monomials $t_1, t_2, \ldots, t_k$ such that there is no superterm of t_i in $M \backslash \{t_1, t_2, \ldots, t_{i-1}\}$, the difference between the values of $\log_2 |L_{0,2^{m-h}}|$ (or $\log_2 A_{0,2^{m-h};2^{m-h-1}}$) for $C(M)$ and $C(M \backslash \triangle M)$ is given by $\sum_{i=1}^{k} \triangle L^{(h)}(t_i)$, denoted $\triangle L^{(h)}(\triangle M)$ (or $\sum_{i=1}^{k} \triangle A^{(h)}(t_i)$, denoted $\triangle A^{(h)}(\triangle M)$). The total number γ for $C(M_{6,3} \backslash \triangle M)$

with $|\triangle M| = 7$ is given by

$$\begin{aligned}
\gamma &= (2 \cdot 2^{10-\triangle A^{(0)}(\triangle M)} - 1) \cdot 2^{0-\triangle L^{(0)}(\triangle M)} \cdot 2^0 \\
&+ (2 \cdot 2^{6-\triangle A^{(1)}(\triangle M)} - 1) \cdot 2^{10-\triangle L^{(1)}(\triangle M)} \cdot 2^1 \\
&+ (2 \cdot 2^{3-\triangle A^{(2)}(\triangle M)} - 1) \cdot 2^{10-\triangle L^{(2)}(\triangle M)} \cdot 2^2 \\
&+ (2 \cdot 2^{1-\triangle A^{(3)}(\triangle M)} - 1) \cdot 2^{7-\triangle L^{(3)}(\triangle M)} \cdot 2^3 \\
&+ (2 \cdot 2^{0-0} - 1) \cdot 2^{4-\triangle L^{(4)}(\triangle M)} \cdot 2^4 \\
&+ (2 \cdot 2^{0-0} - 1) \cdot 2^{2-0} \cdot 2^5 \qquad (3.13)
\end{aligned}$$

There are two cases to be considered.

(i) From the closure condition (CS) and $|\triangle M| = 7$, $\triangle M$ can contain at most one monomial in $M_{6,2}$. Consider the case where $\triangle M$ contains one monomial t in $M_{6,2}$. By considering the effect to γ by t and its all superterms based on Table 1, we see that we should choose x_1x_2, $x_1x_2x_i$ with $3 \leq i \leq 6$, and then $x_1x_3x_4$ and $x_2x_3x_4$. This choice of $\triangle M$ yields the smallest $\gamma = 7039$. If we choose one monomial of form $x_{i_1}x_{i_2}x_5$ other than $x_1x_2x_5$ instead of one of $x_1x_3x_4$ and $x_2x_3x_4$, then we have $\gamma = 8959$, which is the third smallest value of γ.

(ii) $\triangle M \subseteq M_{6,3} \backslash M_{6,2}$: Since each monomial can be chosen independently without considering the closure condition (CS), from Table 1 and (3.13) we see that we should choose $x_1x_2x_3$ first, $\{x_{i_1}x_{i_2}x_4 : 1 \leq i_1 < i_2 < 4\}$ next and any three monomials in $\{x_{i_1}x_{i_2}x_5 : 1 \leq i_1 < i_2 < 5\}$. For this choice of $\triangle M$, $\gamma = 7871$, which is the second smallest value of γ.

Table 2 summarize these choices of $\triangle M$. A RMLD decoder for $C(M_{6,3} \backslash \triangle$

M_2) is designed with VLSI[9], and a decoder for $C(M_{6,3} \backslash \triangle M_1)$ is being designed.

$\triangle\triangle$

In general, binary sectionalization is not optimum in terms of the number of additions and comparisons. The following sectionalization, called hybrid binary sectionalizataion, is shown to be optimum for $\mathrm{RM}_{6,2}$, $\mathrm{RM}_{6,3}$, $\mathrm{RM}_{7,2}$ and the EBCH code of length 64 with 24 information bits, denoted EBCH(64,24), when ACS procedure is used for solving (2.8). Let the code length be denoted by 2^m.

(Hy 1) For $x = 0$ and $y = 2^m$,

$$z = 2^m - 1.$$

(Hy 2) For $x = 0$ and $y = 2^m - 2^{m-i}$ with $2 \leq i \leq m$,

$$z = 2^m - 2^{m-i+1}.$$

(Hy 3) For $y - x = 2^{m-i}$ with $1 \leq i < m$,

$$z = (x + y)/2.$$

Neither binary sectionalization nor hybrid one is not optimum for EBCH(64,45) and $\mathrm{RM}_{6,4}$. As is shown in Table 3, hoeever, the ratio ρ of the number of additions and comparisons by using ACS procedure with respect to binary sectionalization to that with respect to the optimum one is very close to 1 for RM codes. For EBCH(64,24), $\rho = 275839/271745 = 1.015$.

4. RMLD for the Set of Minimum Weight Codewords of a RM Code

We first review a minimum weight subtrellis search used in suboptimum decoding algorithms proposed in [10-12]. For a binary linear block code C, let mw-C denote the set of codewords of the minimum weight w_1. For a codeword $\boldsymbol{u}$ of C, define mw-$C[\boldsymbol{u}]$ as

$$\text{mw-}C[\boldsymbol{u}] \triangleq \{\boldsymbol{u}+\boldsymbol{v} : \boldsymbol{v} \in C \text{ and } wt(\boldsymbol{u}+\boldsymbol{v}) = w_1\}.$$

Suppose C is used for error control over an AWGN channel using BPSK signaling. Given a received sequence r at the output of the matched filter and a codeword $\boldsymbol{u}$, the process to find the codeword in mw-$C[\boldsymbol{u}]$ that has the **largest correlation metric** with the received sequence r is called a **minimum-weight subtrellis search**. In [3], an outline of a recursive algorithm (RMLD) for a minimum-weight subtrellis search for $\text{RM}_{m,r}$ is presented. Except for modifying input metrics relative to the center codeword $\boldsymbol{u}$, the search algorithm is independent of $\boldsymbol{u}$. For simplicity, the center codeword is considered to be the all zero codeword $\mathbf{0}$. In this section, a formula for the number of additions and comparisons required by a RMLD is derived based on the details of the RMLD for mw-$\text{RM}_{m,r}$.

To derive RMLD for mw-$\text{RM}_{m,r}$, we need to find $p_{x,y}[\text{mw-RM}_{m,r}]$ for properly chosen x and y and its decomposition derived from conditions (2.4), (2.5) and (2.6) in which $B = \text{RM}_{m,r}$.

4.1 $p_{x,y}[\text{mw-RM}_{m,r}]$

Polynomials of degree 1 are said to be linear. For $1 \le \nu \le \mu$, linear polynomials $f_i \triangleq a_{i,0} + \sum_{j=1}^{\mu} a_{i,j}x_j$ with $1 \leq i \leq \nu$ are said to be c-independent if and only if the rank of the coefficient matrix $(a_{i,j} : 1 \leq i \leq \nu, 1 \leq j \leq \mu)$ is ν. Let $\mathrm{LP}_{\mu,\nu}$ denote the set of all products of ν linearly independent linear polynomials in $P_{\mu,1}$. The size of $\mathrm{LP}_{\mu,\nu}$ is given [5] by

$$|\mathrm{LP}_{\mu,\nu}| = 2^{\nu} \begin{bmatrix} \mu \\ \nu \end{bmatrix}, \tag{4.1}$$

where $\begin{bmatrix} \mu \\ \nu \end{bmatrix}$ is defined as

$$\begin{bmatrix} i \\ j \end{bmatrix} = \begin{cases} \prod_{p=0}^{j-1} \frac{2^{i-p}-1}{2^{j-p}-1}, & \text{for } j \geq 1, \\ 1, & \text{for } j = 0. \end{cases} \tag{4.2}$$

For convenience, define $\mathrm{LP}_{\mu,0} \triangleq \{1\}$. It is known [5] that for $f \in P_{m,r}$, $wt(v_m(f)) = |f|_m = 2^{m-r}$ if and only if $f \in \mathrm{LP}_{m,r}$. That is,

$$\text{mw-RM}_{m,r} = \{v_m(f) : f \in \mathrm{LP}_{m,r}\}. \tag{4.3}$$

For $f(x_1, x_2, \cdots, x_\mu)$ and $f'(x_1, x_2, \cdots, x_\mu) \in P_\mu$ with $\mu \geq 1$, define an equivalence relation $f \overset{s}{\approx} f'$, if and only if there is a binary μ-tuple $(a_1, a_2, \cdots, a_\mu)$ such that

$$f'(x_1, x_2, \cdots, x_\mu) = f(x_1 + a_1, x_2 + a_2, \cdots, x_\mu + a_\mu). \tag{4.4}$$

Let $E(f)$ denote the equivalent class (with respect to $\overset{s}{\approx}$) that contains f, and let $\mathrm{ELP}_{\mu,\nu}$ denote the set of equivalent classes (with respect to $\overset{s}{\approx}$) in $\mathrm{LP}_{\mu,\nu}$. For $f = f_1 f_2 \cdots f_\nu \in \mathrm{LP}_{\mu,\nu}$ where $f_1, f_2, \cdots, f_\nu$ are c-independent linear polynomials in $P_{\mu,1}$ and $\alpha = (a_1, a_2, \cdots, a_\nu) \in V_\nu$, define $f^{(\alpha)} \triangleq (f_1 + a_1)(f_2 + a_2) \cdots (f_\nu + a_\nu)$. Then, we have that for $f \in \mathrm{LP}_{\mu,\nu}$,

$$E(f) = \{f^{(\alpha)} : \alpha \in V_\nu\}, \tag{4.5}$$

$$|E(f)| = 2^{\nu}. \tag{4.6}$$

It follows from (4.1) and (4.6) that

$$|\mathrm{ELP}_{\mu,\nu}| = \begin{bmatrix} \mu \\ \nu \end{bmatrix}. \tag{4.7}$$

Then the following lemma holds.

Lemma 3 : If a polynomial f is in $\mathrm{LP}_{\mu,\nu}$ with $1 \leq \nu \leq \mu$, then either

(1) $\mu > \nu$, and $f_0(\triangleq f_{x_\nu=0})$ and $f_1(\triangleq f_{x_\nu=1})$ are in $\mathrm{LP}_{\mu-1,\nu}$, and $f_0 \overset{s}{\approx} f_1$, or

(2) $f_0 \in \mathrm{LP}_{\mu-1,\nu-1}$ and $f_1 = 0$, or

(3) $f_0 = 0$ and $f_1 \in \mathrm{LP}_{\mu-1,\nu-1}$. $\triangle\triangle$

This lemma is a special case of Sec. 3 in [13]. A proof is given in Appendix A. The inverse also holds:

Lemma 4 : Suppose there are polynomials g and g' such that either

(1') g and g' are in $\mathrm{LP}_{\mu-1,\nu}$ and $g \overset{s}{\approx} g'$, or

(2') $g \in \mathrm{LP}_{\mu-1,\nu-1}$ and $g' = 0$, or

(3') $g = 0$ and $g' \in \mathrm{LP}_{\mu-1,\nu-1}$.

Then there is $f \in \mathrm{LP}_{\mu,\nu}$ such that

$$f_0 = g, \text{ and } f_1 = g'. \tag{4.8}$$

(Proof) In case of (1'), g can be expressed as $g_1 g_2 \cdots g_\nu$ where $g_1, g_2, \cdots, g_\nu$ are linearly independent polynomials in $P_{\nu-1,1}$. From (4.5), there is $\alpha = (a_1, a_2, \cdots, a_\nu) \in V_\nu$ such that $g' = g^{(\alpha)}$. Define

$$f(x_1, x_2, \cdots, x_\mu) = (g_1 + a_1 x_\mu)(g_2 + a_2 x_\mu) \cdots (g_\nu + a_\nu x_\mu). \tag{4.9}$$

In case of (2'), define

$$f(x_1, x_2, \cdots, x_\mu) \triangleq (x_\mu + 1) g(x_1, x_2, \cdots, x_{\mu-1}), \tag{4.10}$$

and in case of (3'), define

$$f(x_1, x_2, \cdots, x_\mu) = x_\mu g'(x_1, x_2, \cdots, x_{\mu-1}). \tag{4.11}$$

Then, f meets the condition (4.8). △△

From (3.1), (4.3) and Lemmas 3 and 4, we have Corollary 2:

Corollary 2 : For $1 \leq h \leq m$ and $0 \leq j < 2^h$, define

$$\begin{cases} x & \triangleq & j 2^{m-h}, \\ y & \triangleq & (j+1) 2^{m-h}. \end{cases} \tag{4.12}$$

Then

$$p_{x,y}(\text{mw-RM}_{m,r}) = \bigcup_{\max\{0, r-h\} \leq \nu \leq \min\{m-h, r\}} v_{m-h}(\text{LP}_{m-h,\nu}) \cup \{\mathbf{0}_{y-x}\}. \tag{4.13}$$

△△

For i $\in \{1, 2, 3\}$, let $\text{LP}^{(\text{i})}_{\mu,\nu}$ denote the subset of $\text{LP}_{\mu,\nu}$ which consists of all polynomials in $\text{LP}_{\mu,\nu}$ that satisfy the condition (i) in Lemma 3. Each equivalent class in $\text{ELP}_{\mu,\nu}$ is a subset of $\text{LP}^{(1)}_{\mu,\nu}$ or $\text{LP}^{(2)}_{\mu,\nu} \cup \text{LP}^{(3)}_{\mu,\nu}$. Let $\text{ELP}^{(1)}_{\mu,\nu}$

and $\mathrm{ELP}_{\mu,\nu}^{(2,3)}$ denote the set of equivalent classes in $\mathrm{LP}_{\mu,\nu}^{(1)}$ and $\mathrm{LP}_{\mu,\nu}^{(2)} \cup \mathrm{LP}_{\mu,\nu}^{(3)}$, respectively. For $E \in \mathrm{ELP}_{\mu,\nu}^{(1)}$ and $b \in \{0,1\}$, define

$$E_b \triangleq \{f_b : f \in E\}, \tag{4.14}$$

and for $E \in \mathrm{ELP}_{\mu,\nu}^{(2,3)}$,

$$E_b \triangleq \{f_b : f_b \neq 0 \text{ and } f \in E\}. \tag{4.15}$$

Then it is easy to show that E_b is an equivalent class in $\mathrm{LP}_{\mu-1,\nu}$ and $E_0 = E_1$.

In the above proof of the case (1') of Lemma 4, let $g * g'$ denote f defined by (4.9), and let E denote the equivalence class in $\mathrm{LP}_{\mu-1,\nu}$ which contains g and g'. Then, for $\alpha \in V_\nu$ with $1 \leq \nu \leq \mu - 1$, $\{g * g^{(\alpha)} : g \in E\}$ is an equivalence class in $\mathrm{ELP}_{\mu,\nu}^{(1)}$. Let $E \overset{\alpha}{\circ} E$ denote the equivalence class. Then

$$\mathrm{ELP}_{\mu,\nu}^{(1)} = \{E \overset{\alpha}{\circ} E : E \in \mathrm{ELP}_{\mu-1,\nu} \text{ and } \alpha \in V_\nu\}. \tag{4.16}$$

In case of (2') (or (3')), let $g * 0$ (or $0 * g'$) denote f defined by (4.10) (or (4.11)), and let E denote the equivalence class containing g (or g'). Then, let E^{up} denote the equivalence class containing $g * 0$ and $0 * g'$ in $\mathrm{LP}_{\mu,\nu}$.

4.2 Decomposition of $p_{x,y}(\text{mw-RM}_{m,r})$ into $L_{x,y}$

From the recursive property of RM codes as shown in Lemmas 1 to 4, we will choose z as $(x+y)/2$ in (1) of RMLD-[mw-RM$_{m,r}$](x, y), that is, we adopt binary sectionalization. The following lemma shows how $p_{x,y}(\text{mw-RM}_{m,r})$ is decomposed into $L_{x,y}$.

Lemma 5 : For $0 \leq h < m$ and $0 \leq j < 2^h$, define

$$\begin{cases} x & \triangleq & j2^{m-h}, \\ y & \triangleq & (j+1)2^{m-h}. \end{cases} \tag{4.17}$$

Then, for $\boldsymbol{u} = v_{m-h}(f)$ and $\boldsymbol{v} = v_{m-h}(f')$ in $p_{x,y}(\text{mw-RM}_{m,r})$ with f and f' in $P_{m-h,r}$, there are states $\sigma \in \Sigma_x(\text{mw-RM}_{m,r})$ and $\sigma' \in \Sigma_y(\text{mw-RM}_{m,r})$ in $T_{\text{mw-RM}_{m,r}}$ such that $\boldsymbol{u}$ and $\boldsymbol{v}$ are in $L(\sigma, \sigma')$, if and only if one of the following condition holds:

(i) $r < h$, and $f = f'$, that is, there is only one f,

(ii) $m > r \geq h > 0$, and if $f \neq f'$, then f and f' are in $\text{LP}_{m-h,r-h+1}$, and $f \overset{s}{\approx} f'$,

(iii) $r \geq h$, and f and f' are in $\text{LP}_{m-h,r-h}$. △△

(A proof is given in Appendix B.)

Hereafter, $v_\mu(f)$ with $f \in P_\mu$ and $v_\mu(X)$ with $X \subseteq P_\mu$ are abreviated as f and X, respectively, if there is no confusion. Corollary 3 follows from (2.1), (3.1), (4.13) and Lemma 5.

Corollary 3 :

(1) If $r < h < m$, then $L_{x,y}$ is the set of singletons $\{f\}$ for $f \in \bigcup_{0 \leq \nu \leq \min\{m-h,r\}} \text{LP}_{m-h,\nu} \cup \{0\}$, and

$$|L_{x,y}| = 1 + \sum_{\nu=0}^{\min\{m-h,r\}} 2^\nu \begin{bmatrix} m-h \\ \nu \end{bmatrix}. \tag{4.18}$$

(2) If $1 \leq h \leq r$, then $L_{x,y}$ is the family of sets of the following three types:

$$\begin{aligned} L_{x,y} \quad = \quad & \{\text{LP}_{m-h,r-h}, E \in \text{ELP}_{m-h,r-h+1}, \\ & \{f\} \text{ for } f \in \bigcup_{r-h+2 \leq \nu \leq \min\{m-h,r\}} \text{LP}_{m-h,\nu} \cup \{0\}\}. \end{aligned} \tag{4.19}$$

and

$$|L_{x,y}| = 2 + \begin{bmatrix} m-h \\ r-h+1 \end{bmatrix} + \sum_{\nu=r-h+2}^{\min\{m-h,r\}} 2^r \begin{bmatrix} m-h \\ \nu \end{bmatrix}. \tag{4.20}$$

(3) For $h = 0$,

$$L_{0,2^m} = \{\mathrm{LP}_{m,r}\}. \tag{4.21}$$

$\triangle\triangle$

It follows from (2.6) that for $D \in L_{x,y}$, there is a weight w and a coset F in $p_{x,y}(\mathrm{RM}_{m,r})/[\mathrm{RM}]^{\mathrm{tr}}_{x,y} = \mathrm{RM}_{m-h,\min\{m-h,r\}}/\mathrm{RM}_{m-h,r-h}$ ($\mathrm{RM}_{m-h,\min\{m-h,r\}}$ if $r < h$) such that $D = F_w$. If $r < h$, D is a singleton. If $1 \le h \le r$, then

$$\mathrm{LP}_{m-h,r-h} = [\mathrm{RM}_{m-h,r-h}]_{2^{m-h}},$$

$$E = F_{2^{m-r-1}}, \text{ for } E \in \mathrm{ELP}_{m-h,r-h+1},$$

$$\{f\} = F_{2^\nu}, \text{ for } f \in \mathrm{LP}_{m-h,\nu} \text{ with } r-h+2 \le \nu \le \min\{m-h,r\},$$

$$\{0\} = [\mathrm{RM}_{m-h,r-h}]_0.$$

Example 2 : From (2.1), $|L_{0,y}| = |\Sigma_y(C)|$. Hence, for $1 \le h \le r$, $|\Sigma_{2^{m-h}}(\text{mw-}\mathrm{RM}_{m,r})|$ is given by the right-hand side of (4.20). For $h = 1$,

$$|\Sigma_{2^{m-1}}(\text{mw-}\mathrm{RM}_{m,r})| = 2 + \begin{bmatrix} m-1 \\ r \end{bmatrix},$$

and for $h = 2$ and $2 \le r \le m-2$,

$$\begin{aligned} |\Sigma_{2^{m-2}}(\text{mw-}\mathrm{RM}_{m,r})| &= 2 + \begin{bmatrix} m-2 \\ r-1 \end{bmatrix} + 2^r \begin{bmatrix} m-2 \\ r \end{bmatrix} \\ &= 2 + \begin{bmatrix} m-1 \\ r \end{bmatrix} \\ &= |\Sigma_{2^{m-1}}(\text{mw-}\mathrm{RM}_{m,r})|. \end{aligned}$$

In the above derivation, identity $\left[\begin{smallmatrix} i \\ j \end{smallmatrix}\right] = \left[\begin{smallmatrix} i-1 \\ j-1 \end{smallmatrix}\right] + 2^j \left[\begin{smallmatrix} i-1 \\ j \end{smallmatrix}\right]$ [M-S] is used. $\triangle\triangle$

From the definition of $A_z(D)$, Lemmas 3 and 4, (3.1), (4.14) to (4.16) and Corollary 3, we have Corollary 4. A proof is given in Appendix C.

Corollary 4 : For $0 \leq h < m$ and $0 \leq j < 2^h$, define

$$\begin{cases} x &= j2^{m-h} = (2j)2^{m-h-1}, \\ z &= (2j+1)2^{m-h-1}, \\ y &= (j+1)2^{m-h}. \end{cases} \tag{4.22}$$

Define $\underline{\nu} \triangleq 0$, if $r < h$, and $\underline{\nu} \triangleq r - h + 2$, otherwise. Then, for $D_y \in L_{x,y}$, $A_z(D_y)$ and $\mathrm{adj}(D_z, D_y)$ with $D_z \in A_z(D_y)$ are given as follows:

(i) For $\{f\} \in L_{x,y}$ where $f \in \mathrm{LP}_{m-h,\nu} \cup \{0\}$ with $\underline{\nu} \leq \nu \leq \min\{m-h, r\}$,

$$A_z(\{f\}) = \{f_0\}, \quad \mathrm{adj}(\{f_0\}, \{f\}) = \{f_1\}. \tag{4.23}$$

(ii) For $\mathrm{LP}_{m-h,r-h} \in L_{x,y}$ with $r \geq h$, if $r = h$, then $\mathrm{LP}_{m-h,r-h} = \{1\}$ and

$$A_z(\{1\}) = \{1\}, \quad \mathrm{adj}(\{1\}, \{1\}) = \{1\}, \tag{4.24}$$

and if $r > h$, then

$$A_z(\mathrm{LP}_{m-h,r-h}) = \{\{0\}, \mathrm{LP}_{m-h-1,r-h-1}, E \in \mathrm{ELP}_{m-h-1,r-h}\} \tag{4.25}$$

$$\mathrm{adj}(E, \mathrm{LP}_{m-h,r-h}) = E, \ \text{ for } E \in \mathrm{ELP}_{m-h-1,r-h}, \tag{4.26}$$

$$\mathrm{adj}(\mathrm{LP}_{m-h-1,r-h-1}, \mathrm{LP}_{m-h,r-h}) = \{0\}, \tag{4.27}$$

$$\mathrm{adj}(\{0\}, \mathrm{LP}_{m-h,r-h}) = \mathrm{LP}_{m-h-1,r-h-1} = \text{mw-RM}_{m-h-1,r-h-1}. \tag{4.28}$$

(iii) For $E \in L_{x,y}$ with $E \in \mathrm{ELP}_{m-h,r-h+1}$ and $m > r \geq h \geq 1$, if $E \in \mathrm{ELP}^{(1)}_{m-h,r-h+1}$, then

$$A_z(E) = \{\{f_0\} : f \in E\}, \tag{4.29}$$

$$\mathrm{adj}(\{f_0\}, E) = \{f_1\}, \text{ for } f \in E, \tag{4.30}$$

and if $E \in \mathrm{ELP}^{(2,3)}_{m-h,r-h+1}$, then

$$A_z(E) = \{E_0, \{0\}\}, \tag{4.31}$$

$$\mathrm{adj}(E_0, E) = \{0\}, \tag{4.32}$$

$$\mathrm{adj}(\{0\}, E) = E_1. \tag{4.33}$$

$\triangle\triangle$

Example 3 : Let $1 \leq r < m$ and $h = 1$ in (4.17). Then, $x = 0$ and $y = 2^{m-1}$ or $x = 2^{m-1}$ and $y = 2^m$. From (4.19)

$$L_{0,2^{m-1}} = L_{2^{m-1},2^m} = \{\{0\}, \mathrm{LP}_{m-1,r-1}, E \in \mathrm{ELP}_{m-1,r}\}.$$

Next let $h = 0$, that is, $x = 0$ and $y = 2^m$. From (4.21), $L_{0,2^m} = \{\mathrm{LP}_{m,r}\}$. From Corollary 4.

$$A_{2^{m-1}}(\mathrm{LP}_{m,r}) = \{\{0\}, \mathrm{LP}_{m-1,r-1}, E \in \mathrm{ELP}_{m-1,r}\},$$

$$\mathrm{adj}(\{0\}, \mathrm{LP}_{m,r}) = \mathrm{LP}_{m-1,r-1},$$

$$\mathrm{adj}(\mathrm{LP}_{m-1,r-1}, \mathrm{LP}_{m,r}) = \{0\},$$

$$\mathrm{adj}(E, \mathrm{LP}_{m,r}) = E, \text{ for } E \in \mathrm{ELP}_{m-1,r}.$$

$\triangle\triangle$

4.3 RMLD for mw-RM$_{m,r}$

By using the binary sectionalization, we present RMLD(x, y) for mw-RM$_{m,r}$ with $1 \leq r < m$ and evaluate the complexity based on Corollaries 3 and 4.

Let $\gamma(x,y)$ denote the number of those additions and comparisons with respect to metrics which are required for RMLD(x,y). Assume that metrics relative to that of $\mathbf{0}_{y-x}$ ($p_{x,y}$ (the center codeword)) are used.

(1) RMLD$(0,2^m)$

Assume that $L_{0,2^{m-1}}$-table and $L_{2^{m-1},2^m}$-table have been obtained. Then, by referring to Examples 2 and 3, RMLD$(0,2^m)$ is defined as follows:

(L) $m^{(1)}_{0,2^m} \leftarrow \max\{m_{0,2^{m-1}}(E) + m_{2^{m-1},2^m}(E) : E \in \mathrm{ELP}_{m-1,r}\}$.
(The number of additions is $\begin{bmatrix} m-1 \\ r \end{bmatrix}$ and that of comparisons is $\begin{bmatrix} m-1 \\ r \end{bmatrix} - 1$.) Choose E such that $m^{(1)}_{0,2^m} = m_{0,2^{m-1}}(E) + m_{2^{m-1},2^m}(E)$.

$$\begin{aligned} m_{0,2^m}(\mathrm{LP}_{m,r}) \quad \leftarrow \quad & \max\{m^{(1)}_{0,2^m}, m^{(2)}_{0,2^m} \triangleq m_{0,2^{m-1}}(\mathrm{LP}_{m-1,r-1}), \\ & m^{(3)}_{0,2^m} \triangleq m_{2^{m-1},2^m}(\mathrm{LP}_{m-1,r-1})\}. \end{aligned}$$

(The number of comparisons is 2.)

$$l_{0,2^m}(\mathrm{LP}_{m,r}) \leftarrow \begin{cases} l_{0,2^{m-1}}(E) \circ l_{2^{m-1},2^m}(E), & \text{if } m^{(1)}_{0,2^m} \geq \max\{m^{(2)}_{0,2^m}, m^{(3)}_{0,2^m}\}, \\ l_{0,2^{m-1}}(\mathrm{LP}_{m-1,r-1}) \circ \mathbf{0}_{2^{m-1}}, & \text{otherwise and if } m^{(2)}_{0,2^m} \geq m^{(3)}_{0,2^m}, \\ \mathbf{0}_{2^{m-1}} \circ l_{2^{m-1},2^m}(\mathrm{LP}_{m-1,r-1}), & \text{otherwise .} \end{cases}$$

Output $l_{0,2^m}(\mathrm{LP}_{m,r})+$ (the center codeword) with the largest metric.

△△

By the symmetric property of $\mathrm{RM}_{m,r}$ (Lemma 1),

$$\gamma(0,2^m) = 2\begin{bmatrix} m-1 \\ r \end{bmatrix} + 1 + 2\gamma(0,2^{m-1}). \tag{4.34}$$

(2) For $1 \leq h < m-1$ and $0 \leq j < 2^{h-1}$, define

$$\begin{aligned} x &= j2^{m-h} = (2j)2^{m-h-1}, \\ z &= (2j+1)2^{m-h-1}, \\ y &= (j+1)2^{m-h}. \end{aligned}$$

(By the symmetric property, it can be assumed that $y \leq 2^{m-1}$.) Assume that $L_{x,z}$-table and $L_{z,y}$-table have been obtained. Then, RMLD(x, y) is defined as follows:

Let ν be an integer such that if $r < h$, then $0 \leq \nu \leq \min\{m-h, r\}$ and otherwise, $r-h+2 \leq \nu \leq \min\{m-h, r\}$.

(2-1) For g and g' in $\mathrm{LP}_{m-h-1,\nu}$ with $\nu \leq m-h-1$ such that $g \overset{s}{\approx} g'$,

$$\begin{aligned} m_{x,y}(g * g') &\leftarrow m_{x,z}(g) + m_{z,y}(g'), \\ l_{x,y}(g * g') &\leftarrow l_{x,z}(g) + l_{z,y}(g'), \end{aligned}$$

where $g * g' \in \mathrm{LP}_{m-h,\nu}$, and for $g \in \mathrm{LP}_{m-h-1,\nu-1}$ with $\nu \geq 1$,

$$\begin{aligned} m_{x,y}(g * 0) &\leftarrow m_{x,z}(g), \\ m_{x,y}(0 * g) &\leftarrow m_{z,y}(g), \\ l_{x,y}(g * 0) &\leftarrow l_{x,z}(g) \circ \mathbf{0}_{y-z}, \\ l_{x,y}(0 * g) &\leftarrow \mathbf{0}_{z-x} \circ l_{z,y}(g), \end{aligned}$$

where $g * 0$ and $0 * g$ are in $\mathrm{LP}_{m-h,\nu}$.

(The number of additions is $2^{2\nu} \left[{m-h-1 \atop \nu}\right]$.)

(2-2) Suppose that $r \geq h$. Then,

(L-i) if $r = h$,

$$\begin{aligned} m_{x,y}(\mathrm{LP}_{m-h,r-h}) &\leftarrow m_{x,z}(1) + m_{z,y}(1), \\ l_{x,y}(\mathrm{LP}_{m-h,r-h}) &\leftarrow l_{x,z}(1) \circ l_{z,y}(1), \end{aligned}$$

(The number of additions is 1.)

(L-ii) if $r > h$,

$$m^{(1)}_{x,y} \leftarrow \max\{m_{x,z}(E) + m_{z,y}(E) : E \in \mathrm{ELP}_{m-h-1,r-h}\}.$$

(The number of additions is $\left[{m-h-1 \atop r-h}\right]$ and that of comarisons is $\left[{m-h-1 \atop r-h}\right] - 1$.)

Choose E such that $m^{(1)}_{x,y} = m_{x,z}(E) + m_{z,y}(E)$, and

$$\begin{aligned} m_{x,y}(\mathrm{LP}_{m-h,r-h}) \leftarrow \max\{ \;& m^{(1)}_{x,y}, \\ & m^{(2)}_{x,y} \triangleq m_{x,z}(\mathrm{LP}_{m-h-1,r-h-1}), \\ & m^{(3)}_{z,y} \triangleq m_{z,y}(\mathrm{LP}_{m-h-1,r-h-1}) \;\}, \end{aligned}$$

(The number of comparisons is 2.)

$$l_{x,y}(\mathrm{LP}_{m-h,r-h}) \leftarrow \begin{cases} l_{x,z}(E) \circ l_{z,y}(E), \text{ if } m^{(1)}_{x,y} \geq \max\{m^{(2)}_{x,y}, m^{(3)}_{x,y}\}, \\ l_{x,z}(\mathrm{LP}_{m-h-1,r-h-1}) \circ \mathbf{0}_{y-z}, \\ \qquad\qquad \text{otherwise and if } m^{(2)}_{x,z} \geq m^{(3)}_{z,y}, \\ \mathbf{0}_{z-x} \circ l_{z,y}(\mathrm{LP}_{m-h-1,r-h-1}), \text{ otherwise.} \end{cases}$$

(E-1) For $E \in \mathrm{ELP}_{m-h-1,r-h+1}$ and $\alpha \in V_{r-h+1}$,

$$m_{x,y}(E \overset{\alpha}{\circ} E) \leftarrow \max\{m_{x,z}(g) + m_{z,y}(g^{(\alpha)}) : g \in E\}, \quad (4.35)$$

let $g_{\max,\alpha}$ denote $g \in E$ which takes the maximum of the above right-hand side, and

$$l_{x,y}(E \overset{\alpha}{\circ} E) \leftarrow l_{x,z}(g_{\max,\alpha}) \circ l_{z,y}(g^{(\alpha)}_{\max,\alpha}),$$

where $E \overset{\alpha}{\circ} E \in \mathrm{ELP}^{(1)}_{m-h,r-h+1}$. If (4.35) is computed by add-compare-selet (ACS) procedure, then the number of additions is $2^{2(r-h+1)} \begin{bmatrix} m-h-1 \\ r-h+1 \end{bmatrix}$ and that of comparisons is $2^{r-h+1}(2^{r-h+1} - 1) \begin{bmatrix} m-h-1 \\ r-h+1 \end{bmatrix}$. If (4.35) is computed by sort-add-compare-select (SACS) procedure, the number of additions and comparisons is reduced (see Appendix D).

(E-2) For $E \in \mathrm{ELP}_{m-h-1,r-h}$,

$$m_{x,y}(E^{\mathrm{up}}) \leftarrow \max\{m_{x,z}(E), m_{z,y}(E)\}$$

(the number of comparisons is $\begin{bmatrix} m-h-1 \\ r-h \end{bmatrix}$),

$$l_{x,y}(E^{\mathrm{up}}) \leftarrow \begin{cases} l_{x,y}(E) \circ \mathbf{0}_{y-z}, & \text{if } m_{x,z}(E) \geq m_{z,y}(E), \\ \mathbf{0}_{z-x} \circ l_{z,y}(E), & \text{otherwise }, \end{cases}$$

where $E^{\mathrm{up}} \in \mathrm{ELP}^{(2,3)}_{m-h,r-h+1}$.

△△

From the above description,

$$\begin{aligned} &\gamma(x,y) \\ &\quad = \gamma(x,z) + \gamma(z,y) + \begin{cases} \displaystyle\sum_{\nu=0}^{\min\{m-h-1,r\}} 2^{2\nu} \begin{bmatrix} m-h-1 \\ \nu \end{bmatrix}, & \text{for } r < h < m-1, \\ \left.\begin{array}{c} \displaystyle\sum_{\nu=2}^{\min\{m-h-1,r\}} 2^{2\nu} \begin{bmatrix} m-h-1 \\ \nu \end{bmatrix} \\ +6 \begin{bmatrix} m-h-1 \\ 1 \end{bmatrix} + 2, \end{array}\right\} & \text{for } r = h, \\ \left.\begin{array}{c} \displaystyle\sum_{\nu=r-h+2}^{\min\{m-h-1,r\}} 2^{2\nu} \begin{bmatrix} m-h-1 \\ \nu \end{bmatrix} \\ +3 \begin{bmatrix} m-h-1 \\ r-h \end{bmatrix} \\ +2^{r-h+1}(2^{r-h+2} - 1) \begin{bmatrix} m-h-1 \\ r-h+1 \end{bmatrix} + 1. \end{array}\right\} & \text{for } 1 \leq h < r. \end{cases} \end{aligned} \tag{4.36}$$

(3) For $h = m-1$, note that $\mathrm{LP}_{1,0} = \{1\}$, $\mathrm{LP}_{1,1} = \{x_1, 1 + x_1\}$ and $\mathrm{ELP}_{1,1}$

consists of a single class $\{x_1, 1+x_1\}$. For an integer $j < 2^{m-1}$,

$$\gamma(2j, 2j+2) = \begin{cases} 1, & \text{if} \quad m-1 > r, \\ 2, & \text{if} \quad m-1 = r. \end{cases}$$

△△

Note that in real-time processing of the above algorithm, no reference to the minimum weight subtrellis of the target RM code is made directly.

The above approach presented in this subsection can be applied to some EBCH codes that consist of relatively small number of cosets with respect to RM codes as their subcodes.

Table 3 shows the numbers of additions and comparisons required by the above RMLD algorithm for several mw-RM$_{m,r}$ codes and RM$_{m,r}$ codes with respect to binary sectionalization. The second and third columns for mw-RM$_{m,r}$ and the fifth and seventh columns for RM$_{m,r}$ show the numbers for two cases where the subprocedure (4.35) is computed by ACS (add-compare-select) procedure and SACS (sort-add-compare-select) procedure, respectively. For SACS, the number of operation depends on the received sequence. The values of the third column are evaluated by using the upper bound (D.2) shown in Appendix D, and the values of the seventh column are evaluated by an upper bound in [1, 2]. The fourth column shows the comparisons with those required by conventional Viterbi algorithm based on 4-section minimum weight subtrellises where the branch metrics are computed by generating the metrics of length 4 or 8 in the order of Gray code [14] and combining them. The sixth and last columns show the values of ρ, the ratio of the number of additions and comparisons by using ACS procedure and SACS procedure with respect to binary sectionalization to that with respect to the optimum one respectively.

Table 1:$\triangle L^{(h)}(t)$ and $\triangle A^{(h-1)}(t)$ with $1 \leq h \leq 5$ for $t \in M_{6,3} - M_{6,1}$

	$\triangle L^{(1)}(t)$	$\triangle L^{(2)}(t)$	$\triangle L^{(3)}(t)$	$\triangle L^{(4)}(t)$	$\triangle A^{(0)}(t)$	$\triangle A^{(1)}(t)$	$\triangle A^{(2)}(t)$	$\triangle A^{(3)}(t)$
$t = x_4x_5x_6$	-1	-1	-1	0	-1	-1	-1	1
$t = x_ix_5x_6$ $(1 \leq i < 4)$	-1	-1	0	0	-1	-1	1	0
$t = x_{i_1}x_{i_2}x_6$ $(1 \leq i_1 < i_2 < 5)$	-1	0	0	0	-1	1	0	0
$t = x_{i_1}x_{i_2}x_5$ $(1 \leq i_1 < i_2 < 5)$	1	0	0	0	1	0	0	0
$t = x_{i_1}x_{i_2}x_4$ $(1 \leq i_1 < i_2 < 4)$	1	1	0	0	1	0	0	0
$t = x_1x_2x_3$	1	1	1	0	1	0	0	0
$t = x_5x_6$	-1	-1	0	0	-1	-1	1	0
$t = x_ix_6$ $(1 \leq i < 5)$	-1	0	0	0	-1	1	0	0
$t = x_ix_5$ $(1 \leq i < 5)$	1	0	0	0	1	0	0	0
$t = x_ix_4$ $(1 \leq i < 4)$	1	1	0	0	1	0	0	0
$t = x_ix_3$ $(1 \leq i < 3)$	1	1	1	0	1	0	0	0
$t = x_1x_2$	1	1	1	1	1	0	0	0

$\triangle L^{(5)}(t) = \triangle A^{(4)}(t) = \triangle A^{(5)}(t) = 0$ for $t \in M_{6,3} - M_{6,1}$.

Table 2:$\triangle L^{(h)}(\triangle M)$ and $\triangle A^{(h-1)}(\triangle M)$ with $1 \leq h \leq 5$ and γ for $C(M_{6,3} \setminus \triangle M)$.

$\triangle M$	$\triangle L^{(1)}(t)$	$\triangle L^{(2)}(t)$	$\triangle L^{(3)}(t)$	$\triangle L^{(4)}(t)$	$\triangle A^{(0)}(t)$	$\triangle A^{(1)}(t)$	γ
$x_{i_1}x_{i_2}x_5(3)$ $x_{i_1}x_{i_2}x_4(3)$ $x_1x_2x_3$	7	4	1	0	7	0	7807
x_1x_2 $x_1x_2x_i(4)$ $(3 \leq i \leq 6)$ $x_{i_1}x_{i_2}x_5(1)$ $x_{i_1}x_{i_2}x_4(1)$	5	4	2	1	5	1	8959
x_1x_2 $x_1x_2x_i(4)$ $(3 \leq i \leq 6)$ $x_1x_3x_4$ $x_2x_3x_4$	5	5	2	1	5	1	7039

$\triangle L^{(5)}(t) = \triangle A^{(2)}(t) = \triangle A^{(3)}(t) = \triangle A^{(4)}(t) = \triangle A^{(5)}(t) = 0$ for $C(M_{6,3} \setminus \triangle M)$.

In the column of $\triangle M$, notation "$t(k)$", where t is a form of monomial, indicates that $\triangle M$ contains exactly k monomials of the form t.

Table 3: The Numbers of Additions and Comparisons Required by RMLD with respect to binary sectionalization and the values of ρ for mw-RM$_{m,r}$ and RM$_{m,r}$

Code		RMLD for mw-RM$_{m,r}$		Conventional Viterbi Decoding for mw-RM$_{m,r}$	RMLD for RM$_{m,r}$			
m	r	ACS	SACS		ACS	ρ	SACS	ρ
6	2	3331	3191	5127	79231	1.013	67711	1.009
	3	5651	5535	9299	327039	1.003	212159	1.000
	4	2179	2031		5375	1.018	4375	1.002
7	2	14595	13975	26967	4606719	1.007	3463935	1.005
	3	51315	50415		4425388799	1.001	2293107455	1.000
	4	44691	40615		151116543	1.456	78223999	1.048
	5	8579	7287		22783	1.032	168871	1.000
8	2	61443	58839		562599423	1.004	372676095	1.003
	3	437299	430479					
	4	808627	726103					

References

[1] T.Fujiwara, H.Yamamoto, T.Kasami and S.Lin, "A Trellis-Based Recursive Maximum Likelihood Decoding Algorithm for Linear Block Codes," *Proc. of the 33rd Annual Allerton Conference on Communication, Control and Computing*, pp.700–709, Oct. 1995, also submmitted to *IEEE Trans. Inf. Theory*, 1996 (revised Apr. 1997).

[2] H.Yamamoto, H.Nagano, T.Fujiwara, T.Kasami and S.Lin, "Recursive MLD Algorithm Using the Detail Trellis Structure for a Linear Block Codes and Its Average Complexity Analysis," *Proc. of the International Symposium on Information Theory and Its Applications*, Victoria, Canada, pp.704–708, Sep. 1996.

[3] T.Kasami, T.Koumoto, T.Fujiwara, H.Yamamoto, Y.Desaki and S.Lin, "Low Weight Subtrellises for Binary Linear Block Codes and Their Applications," *IEICE Trans. Fundamentals*, Vol.E80-A, No.11, Nov. 1997.

[4] G.D.Forney Jr., "Dimension/Length Profiles and Trellis Complexity of Linear Block Codes," *IEEE Trans. Inf. Theory*, Vol.40, No.6, pp.1741–1752, Nov. 1994.

[5] F.J.MacWillams and N.J.A.Sloane, "The Theory of Error-Correcting Codes," *Amsterdam, The Netherlands: North-Holland*, 1977.

[6] W.W.Peterson and E.J.Weldon Jr., "Error Correcting Codes," 2nd ed., *MIT Press, Cambridge*, 1972.

[7] T.Kasami, T.Takata, T.Fujiwara and S.Lin, "On Complexity of Trellis Structure of Linear Block Codes," *IEEE Trans. Inf. Theory*, Vol.39, No.3, pp.1057–1064, 1993.

[8] H.T.Moorthy, S.Lin and G.T.Uehara, "Good Trellises for IC Implementation of Viterbi Decoders for Linear Block Codes," *IEEE Trans. Commun.*, Vol.45, No.1, pp.52–63, Jan. 1997.

[9] T.Fujiwara, T.Komura, T.Onoye, Y.Kaji, T.Kasami and S.Lin, "IC Implementation of a Recursive Maximum Likelihood Decoding Algorithm for Reed-Muller and Related Codes," *4th International Symposium on Communication Theory and Applications*, Lake District, U.K., 13-18 July, 1997.

[10] H.T.Moorthy, S.Lin and T.Kasami, "Soft-Decision Decoding of Binary Linear Block Codes Based on an Iterative Search Algorithm," *IEEE Trans. Inf. Theory*, Vol.43, No.3, pp.1030–1040, May 1997.

[11] T.Koumoto, H.Nagano, T.Takata, T.Fujiwara, T.Kasami and S.Lin, "A New Iterative Soft-Decision Decoding Algorithm," *Technical Report of IEICE*, IT95-28, Jul. 1995, and *Proc. of the 18th Symposium on Information Theory and Its Applications*, pp.557–560, Oct. 1995.

[12] T.Koumoto, T.Takata, T.Kasami and S.Lin, "An Iterative Soft-Decision Decoding Algorithm," *Proc. of the International Symposium on Information Theory and Its Applications, Victoria, Canada*, pp.806–810, Sep. 1996.

[13] T.Kasami, T.Sugita and T.Fujiwara, "The Split Weight (w_L, w_R) Enumeration of Reed-Muller Codes for $w_L + w_R < 2d_{\min}$," to appear in *Proc. of AAECC 12*.

[14] A.Vardy and Y.Be'ery, "Maximum Likelihood-Soft-Decision Decoding of BCH Codes," *IEEE Trans. Inf. Theory*, Vo.40, pp.546–554, Mar. 1994.

Appendix A : **Proof of Lemma 2**

Let $f_1 f_2 \cdots f_\nu$ be a polynomial in $L_{\mu,\nu}$, where $f_i = a_{i,0} + \sum_{j=1}^{\mu} a_{i,j} x_j \in P_{\mu,1}$ with $1 \leq i \leq \nu$ and the rank of matrix $A \triangleq (a_{i,j} : 1 \leq i \leq \nu, 1 \leq j \leq \mu)$ is ν. If $\mu \geq \nu$ and the rank of the submatrix obtained from A by deleting the last column is ν, then the condition (1) holds. Otherwise, by repeating row operation $f_i \leftarrow f_i + f_{i'} + 1$, we can assume that $a_{1,j} = 0$ for $1 \leq j \leq \mu - 1$. Since the rank of A is ν, $a_{1,\mu} = 1$. If $a_{1,0} = 1$, then the condition (2) holds, and otherwise, the condition (3) holds.

$\triangle\triangle$

Appendix B : Proof of Lemma 4

From (2.2), there are states $\sigma \in \Sigma_x(\text{mw-RM}_{m,r})$ and $\sigma' \in \Sigma_y(\text{mw-RM}_{m,r})$ in $T_{\text{mw}-\text{RM}_{m,r}}$ such that $\boldsymbol{u} \in L(\sigma, \sigma')$. From (2.4) and (2.5), $\boldsymbol{v} \in L(\sigma, \sigma')$ if and only if

$$wt(\boldsymbol{v}) = wt(\boldsymbol{u}), \tag{B.1}$$

$$\boldsymbol{u} + \boldsymbol{v} \in [\text{RM}_{m,r}]^{\text{tr}}_{x,y}. \tag{B.2}$$

If $r < h$, then from Lemma 1, $[\text{RM}_{m,r}]^{\text{tr}}_{x,y} = \{\mathbf{0}_{2^{m-h}}\}$. Hence, the condition (i) holds. Next consider the remaining case of $r \geq h$.

(IF-part) : If the condition (ii) holds, then it follows from (4.5) and "$f \overset{s}{\approx} f'$" that $wt(v_{m-h}(f)) = wt(v_{m-h}(f')) = 2^{m-r-1}$ and $f + f' \in P_{m-h,r-h}$. From (??), "$f + f' \in P_{m-h,r-h}$" implies $\boldsymbol{u} + \boldsymbol{v} \in \text{RM}_{m-h,r-h} = [\text{RM}_{m,r}]^{\text{tr}}_{x,y}$. If the condition (iii) holds, then from (??) and (4.3), $wt(v_{m-h}(f)) = wt(v_{m-h}(f')) = 2^{m-r}$ and $v_{m-h}(f) + v_{m-h}(f') \in \text{RM}_{m-h,r-h} = [\text{RM}_{m,r}]^{\text{tr}}_{x,y}$.

(Only-if-part) : Suppose that conditions (B.1) and (B.2) hold. Since $r \geq h$, from Lemma 1, the condition (B.2) becomes $\boldsymbol{u} + \boldsymbol{v} \in \text{RM}_{m-h,r-h}$, that is,

$$f + f' \in P_{m-h,r-h}. \tag{B.3}$$

From (4.3) and Corollary 2, the condition (B.1) implies that f and f' are in the same $\text{LP}_{m-h,\nu}$, where $r - h \leq \nu \leq \min\{m - h, r\}$. If $\nu = r - h$, then the condition (iii) holds. Suppose that

$$\nu > r - h. \tag{B.4}$$

Then, $m > r$. By an invertible affine transformation, f and f' can be expressed as

$$f = \prod_{i=1}^{\nu}(a_{i,0} + \sum_{j=1}^{m-h} a_{i,j}x_j), \tag{B.5}$$

$$f' = x_1 x_2 \cdots x_\nu, \tag{B.6}$$

where the rank of matrix $A \triangleq (a_{i,j} : 1 \le i \le \nu, 1 \le j \le m-h)$ is ν. For $1 \le j_1 < j_2 < \cdots < j_\nu \le m-h$, let $\alpha_{i_1,j_2,\cdots,j_\nu}$ denote the determinant of the $(\nu \times \nu)$ submatrix which consists of the j_1-th, j_2-th, $\cdots, j_\nu$-th columns of A. Then, $f + \sum_{1 \le j_1 < j_2 < \cdots < j_\nu \le m-h} \alpha_{j_1,j_2,\cdots,j_\nu} x_{j_1} x_{j_2} \cdots x_{j_\nu} \in P_{m-h,\nu-1}$. From (B.3) and (B.4), $f + f' \in P_{m-h,\nu-1}$. Then, $f + f' \in P_{m-h,\nu-1}$ if and only if

$$\alpha_{j_1,j_2,\cdots,j_\nu} = 1 \Leftrightarrow j_1 = 1, j_2 = 2, \cdots, j_\nu = \nu.$$

Therefore, $a_{i,j} = 0$, for $1 \le i \le \nu$ and $\nu < j \le m-h$. This condition implies that $f \overset{s}{\approx} f'$, that is,

$$f = \prod_{i=1}^{\nu} (x_i + b_i), \quad b_i \in \{0, 1\}. \tag{B.7}$$

If $f \ne f'$, then the degree of $f + f'$ is exactly $\nu - 1$. From (B.3), $\nu - 1 \le r - h$. From (B.4), $\nu = r - h + 1$. $\triangle\triangle$

Appendix C : Proof of Corollary 4

It follows from Eq.(3.1) that for $f \in \mathrm{LP}_{m-h,\nu}$,

$$p_{x,z}(v_{m-h}(f)) = v_{m-h-1}(f_0), \tag{C.1}$$

$$p_{z,y}(v_{m-h}(f)) = v_{m-h-1}(f_1). \tag{C.2}$$

From Corollary 3, there are three cases (i), (ii) and (iii) stated in Corollary 4.

Case (i) and (ii) with $r = h$ **:** It follows from (C.1) and (C.2) that for $f \in \mathrm{LP}_{m-h,\nu} \cup \{0\}$, $\{f_0\} \in L_{x,z}$ and $\{f_1\} \in L_{z,y}$. That is, Eq.(4.23) and Eq.(4.24) hold.

Case (ii) with $r > h$ **:** It follows from Lemmas 3 and 4 that for $f \in \mathrm{LP}_{m-h,r-h}$, there are the following three cases:

(1) f_0 and f_1 are in $\mathrm{LP}_{m-h-1,r-h}$, and f_0 and f_1 are in an equivalence class $E \in \mathrm{ELP}_{m-h-1,r-h}$, and from Corollary 3, E is in $L_{x,z}$ and $L_{z,y}$. Conversely, for any g and g' in E, $g * g' \in \mathrm{LP}_{m-h,r-h}$. That is, for $E \in \mathrm{ELP}_{m-h-1,r-h}$, $E \in A_z(\mathrm{LP}_{m-h,r-h})$ and Eq.(4.26) holds.

(2) $f_0 \in \mathrm{LP}_{m-h-1,r-h-1}$ and $f_1 = 0$, and from Corollary 3, $\mathrm{LP}_{m-h-1,r-h-1} \in L_{x,z}$ and $\{0\} \in L_{z,y}$. Conversely, for any $g \in LP_{m-h-1,r-h-1}$, $g * 0 \in \mathrm{LP}_{m-h,r-h}$. That is, $\mathrm{LP}_{m-h-1,r-h-1} \in A_z(\mathrm{LP}_{m-h,r-h})$ and Eq.(4.27) holds.

(3) $f_0 = 0$ and $f_1 \in \mathrm{LP}_{m-h-1,r-h-1}$, and from Corollary 3, $\{0\} \in L_{x,z}$ and $\mathrm{LP}_{m-h-1,r-h-1} \in L_{z,y}$. Conversely, for any $g \in \mathrm{LP}_{m-h-1,r-h-1}$, $0 * g \in \mathrm{LP}_{m-h,r-h}$. That is, $\{0\} \in A_z(\mathrm{LP}_{m-h,r-h})$ and Eq.(4.28) holds. Thus, Eq.(4.25) holds.

Case (iii) : For $f \in E \in L_{x,y}$ with $E \in \mathrm{ELP}_{m-h,r-h+1}$, if $E \in \mathrm{ELP}^{(1)}_{m-h,r-h+1}$, then f_0 and f_1 are in $\mathrm{LP}_{m-h-1,r-h+1}$, and from Corollary 3, $\{f_0\} \in L_{x,z}$ and $\{f_1\} \in L_{z,y}$. That is, Eq.(4.29) and Eq.(4.30) hold. If $E \in \mathrm{ELP}^{(2,3)}_{m-h,r-h+1}$, then either $p_{x,z}(v_{m-h}(E)) = v_{m-h-1}(E_0)$ with $E_0 \in \mathrm{ELP}_{m-h-1,r-h} \in L_{x,z}$ and $p_{z,y}(v_{m-h}(E)) = \{v_{m-h-1}(0)\} \in L_{z,y}$ or $p_{x,z}(v_{m-h}(E)) = \{v_{m-h-1}(0)\} \in L_{x,z}$ and $p_{z,y}(v_{m-h}(E)) = v_{m-h-1}(E_1)$ with $E_1 \in \mathrm{ELP}_{m-h-1,r-h} \in L_{z,y}$. That is, Eq.(4.31) to Eq.(4.33) hold.

△△

Appendix D :

For each $E \in \mathrm{ELP}_{m-h-1,r-h+1}$, consider a complete bipartite graph (trellis), denoted T_E, which has the set of states at time z, $S_z \triangleq \{\{g\} : g \in E\}$, the set of states at time y, $S_y \triangleq \{E \overset{\alpha}{\circ} E$ with $\alpha \in V_{r-h+1}\}$, and branches from each state $\{g\}$ in S_z to each state $E \overset{\alpha}{\circ} E$ in S_y with label $g^{(\alpha)}$. The number of states is given by

$$|S_z| = |S_y| = 2^{r-h+1}.$$

From (4.16), we see that such trellises for all the $E \in \mathrm{ELP}_{m-h-1,r-h+1}$ are a graphical representation of (4.29) and (4.30).

From the definition of $\mathrm{ELP}_{m-h-1,r-h+1}$, $\{g^{(\alpha)} : \alpha \in V_{r-h+1}\}$ is identical for every $g \in E$. Therefore, we can use a SACS procedure, which is called CombCBT-U procedure in [1, 2], to find $m_{x,y}(E \overset{\alpha}{\circ} E)$ and $l_{x,y}(E \overset{\alpha}{\circ} E)$ with $\alpha \in V_{r-h+1}$. The number of additions and comparisons is upper bounded by [1, 2],

$$2\eta(2^{r-h+1}) + 2^{2(r-h+1)}, \tag{D.1}$$

where $\eta(2^\nu)$ is an upper bound on the number of comparisons to sort 2^ν elements, which is given by [1, 2],

$$\eta(2^\nu) = \begin{cases} \nu, & \text{for } \nu = 0, 1, \\ 2(\nu-1)(2^\nu - 1) - 1, & \text{otherwise.} \end{cases} \tag{D.2}$$

It is also shown in [1, 2], the value of (D.2) is always smaller than $2^{r-h+1}(2^{r-h+2} - 1)$, which is the number of operations for the ACS procedure, and they are equal only for the trivial case where $r - h + 1 = 1$. When $r - h + 2$ becomes larger, the ratio of the computational complexity for the SACS procedure to that for the ACS procedure decreases rapidly.