Chapter 9

Reed-Muller Codes: Weak Codes with Easy Decoding

We now introduce an interesting class of multiple-error-correcting binary codes whose prime importance lies in an easily implementable decoding technique: the Reed-Muller codes. Their parameters are listed in Figure 1.

Length:	$n = 2^m$
Information symbols:	$k = \sum_{i=0}^{r} \binom{n}{i}$
Minimum distance:	$d = 2^{m-r}$
Error-control capacity:	Corrects $2^{m-r-1} - 1$ errors by a technique based on majority logic which is easy to implement

Figure 1: Parameters of the Reed-Muller code $\mathcal{R}(r, m)$

There are closely related punctured Reed-Muller codes, the parameters of which are presented in Figure 2.

One of these codes, the Reed-Muller code $\mathcal{R}(1, 5)$, was used by the 1969 Mariner to transmit pictures of the Moon. The code $\mathcal{R}(1, 5)$ has

Length:	$n = 2^m - 1$
Information symbols:	$k = \sum_{i=0}^{r} \binom{n}{i}$
Minimum distance:	$d = 2^{m-r} - 1$
Error-control capacity:	Corrects errors as $\mathcal{R}(r,m)$ does, but has a better information rate and is a cyclic code

Figure 2: Parameters of the punctured Reed-Muller code

length 32 with 6 information bits and it corrects 7 errors. Each dot of the transmited picture was assigned one of $2^6 = 64$ degrees of greyness, and these 6 information bits were then encoded into a word of length 32.

We introduce Reed-Muller codes by means of Boolean polynomials, which we first discuss in some detail. To understand the decoding of Reed-Muller codes, it is more convenient to work with finite geometries, where code words become characteristic functions of flats.

9.1 Boolean Functions

Reed-Muller codes are best described by means of Boolean polynomials. We first show how binary words translate to Boolean functions and Boolean polynomials. Then we introduce the codes and present their basic properties. However, the decoding is better explained in a different, geometrical, presentation of binary words, which we introduce later.

Definition. *A* Boolean function $\mathbf{f} = f(x_0, x_1, \ldots, x_{m-1})$ *of m variables is a rule which assigns to each m-tuple $(x_0, x_1, \ldots, x_{m-1})$ of* 0*'s and* 1*'s a value $f(x_0, x_1, \ldots, x_{m-1}) = 0$ or 1. In other words,* $\mathbf{f}$ *is a function from $\mathbf{Z}_2^m$ to $\mathbf{Z}_2$.*

Truth Table. A simple way of presenting a Boolean function is to list all of its values. That is, we write down all the 2^m combinations of the values of all variables $x_0, x_1, \ldots, x_{m-1}$, and then to each combination, we assign the value $f(x_0, x_1, \ldots, x_{m-1})$. For notational convenience, we proceed in such a way that the columns form binary expansions of the numbers 0, 1, 2, ... (from the first row downward). This presentation is called the truth table of the function $\mathbf{f}$.

An example of a truth table of a Boolean function of three variables is presented in Figure 3. Observe that a truth table of a Boolean function

$\mathbf{x}_0$	0	1	0	1	0	1	0	1
$\mathbf{x}_1$	0	0	1	1	0	0	1	1
$\mathbf{x}_2$	0	0	0	0	1	1	1	1
$\mathbf{f}$	0	1	1	0	1	1	1	0

Figure 3: An example of a Boolean function of three variables

of three variables yields a binary word of length 8, and, conversely, every binary word of length 8 is the truth table of some Boolean function. We thus can (and will) identify Boolean function of three variables with binary words of length 8. For example, the word 01101110 is the same as the Boolean function in Figure 3.

More in general, every binary word $\mathbf{f}$ of length 2^m is considered as a Boolean function of m variables. If we write the indices starting with zero, then the binary word

$$\mathbf{f} = f_0 f_1 \ldots f_{2^m-1}$$

is the Boolean function with

$$\begin{aligned} \mathbf{f}(0,0,\ldots,0,0) &= f_0, \\ \mathbf{f}(0,0,\ldots,0,1) &= f_1, \\ \mathbf{f}(0,0,\ldots,1,0) &= f_2, \\ &\vdots \\ \mathbf{f}(1,1,\ldots,1,1) &= f_{2^m-1}. \end{aligned}$$

In general,

$$f_i = \mathbf{f}(i_{m-1}, \ldots, i_1, i_0),$$

where the number i has the binary expansion $i_{m-1} \ldots i_1 i_0$ (i.e., $i = \sum_{k=0}^{2^m-1} i_k 2^k$).

Examples

(1) There are two constant Boolean functions

$$\mathbf{1} = 11\ldots11 \quad \text{and} \quad \mathbf{0} = 00\ldots00.$$

(2) Every variable is a Boolean function. For example, x_0 is the Boolean function which assigns to each m-tuple $(x_0, x_1, \ldots, x_{m-1})$ the first coordinate value x_0. Thus, the value is 0 for all even numbers and 1 for all odd ones:

$$\mathbf{x}_0 = 0101010\ldots1.$$

(See Figure 3 for the case $m = 3$.) In general,

> x_i is the binary word whose kth position equals 1 precisely when the binary expansion of k has 1 in the ith position $(k = 0, 1, \ldots, 2^m - 1)$.

This follows from the way we are writing the truth tables. For example, $\mathbf{x}_1 = 00110011\ldots0011$, and

$$\mathbf{x}_{m-1} = \underbrace{000\ldots00}_{2^{m-1}}\underbrace{111\ldots11}_{2^{m-1}}.$$

For $m = 4$, the four variables are shown on Figure 4.

$\mathbf{x}_0$	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
$\mathbf{x}_1$	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
$\mathbf{x}_2$	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
$\mathbf{x}_3$	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Figure 4: The four variables as Boolean functions ($m = 4$)

9.2 Boolean Polynomials

We now introduce the basic logical operations on Boolean functions of m variables. If $\mathbf{f}$ is a Boolean function, we denote by f_i $(i = 0, 1, \ldots, 2^m - 1)$ the ith position of its truth table.

The logical sum (also called "exclusive or") of Boolean functions $\mathbf{f}$ and $\mathbf{g}$ is the function

$$\mathbf{f} + \mathbf{g},$$

whose value is 1 precisely when either $\mathbf{f}$ or $\mathbf{g}$ has value 1, but not both. That is, the ith coordinate of $\mathbf{f} + \mathbf{g}$ is $f_i + g_i$ (addition modulo 2). The logical sum is nothing new: it is just the usual addition in the linear space $\mathbf{Z}_2^n$, where $n = 2^m$.

The logical product (also called "and") of Boolean functions $\mathbf{f}$ and $\mathbf{g}$ is the function

$$\mathbf{fg},$$

whose value is 1 precisely when both $\mathbf{f}$ and $\mathbf{g}$ have value 1. That is, the ith coordinate of $\mathbf{fg}$ is $f_i g_i$ (multiplication modulo 2). This is a new operation imposed on $\mathbf{Z}_2^n$ for $n = 2^m$, viz., the coordinatewise multiplication

$$(f_{2^m-1}\ldots f_1 f_0)(g_{2^m-1}\ldots g_1 g_0) = (f_{2^m-1}g_{2^m-1})\cdots(f_1 g_1)(f_0 g_0).$$

Remarks

(1) Observe that the logical product fulfils

$$\mathbf{f}\,\mathbf{f} = \mathbf{f}.$$

Thus, no exponents higher then 1 are ever needed.

(2) There are other natural operations which, however, can be expressed by means of the logical sum and logical product. For example, the *negation* $\bar{\mathbf{f}}$ (which has value 1 precisely when $\mathbf{f}$ has value 0) is simply expressed as a sum with the constant function $\mathbf{1} = 11\ldots11$:

$$\bar{\mathbf{f}} = \mathbf{1} + \mathbf{f}.$$

The operation "or" ($\mathbf{f}$ or $\mathbf{g}$), whose value is 1 precisely when $\mathbf{f}$ or $\mathbf{g}$ (or both) have value 1, can be expressed as

$$\mathbf{f} \text{ or } \mathbf{g} = \mathbf{f} + \mathbf{g} + \mathbf{fg}.$$

(3) We have seen that $\mathbf{1}$ and each variable $\mathbf{x}_i$ are Boolean functions. Other Boolean functions can be obtained by addition and multiplication. For example: $\mathbf{x}_i\mathbf{x}_j$, $\mathbf{1} + \mathbf{x}_i + \mathbf{x}_j$, etc. These are called Boolean polynomials:

Definition. *By a* Boolean polynomial *in m indeterminates is meant a sum of (some of the) following Boolean functions: $\mathbf{1}$, $\mathbf{x}_i$, $\mathbf{x}_i\mathbf{x}_j$, $\ldots$, $\mathbf{x}_{i_1}\mathbf{x}_{i_2}\ldots\mathbf{x}_{i_k}$ (where the indices range over $0, 1, \ldots, m-1$). The zero function $\mathbf{0}$ is called a Boolean polynomial of degree -1, the function $\mathbf{1}$ is called a Boolean polynomial of degree 0, and a Boolean polynomial $\mathbf{f}$ has degree $k \geq 1$ provided that k is a maximum number of factors in a summand of $\mathbf{f}$ (i.e., $\mathbf{f}$ has a summand $\mathbf{x}_{i_1}\mathbf{x}_{i_2}\ldots\mathbf{x}_{i_k}$, and no summand has more than k factors).*

Examples

(1) The Boolean polynomial $\mathbf{1} + \mathbf{x}_1\mathbf{x}_2$ of three indeterminates has degree 2. It is the negation of the polynomial

$$\mathbf{x}_1\mathbf{x}_2 = (01010101)(00110011) = 00010001.$$

Thus,

$$1 + \mathbf{x}_1\mathbf{x}_2 = 11101110.$$

The Boolean polynomial $1 + \mathbf{x}_1\mathbf{x}_2$ considered as a function of four indeterminates is the word

$$1110111011101110.$$

(2) The polynomial $\mathbf{x}_i\mathbf{x}_j$ $(i \neq j)$ is the binary word whose kth position is 1 precisely when k has 1 in positions i and j. The number of such k's is 2^{m-2} (because we can choose the $m-2$ remaining positions of k arbitrarily). Thus, the binary word $\mathbf{x}_i\mathbf{x}_j$ has Hamming weight 2^{m-2}.

Analogously, $\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k$ has Hamming weight 2^{m-3}, and, in general,

$$\mathbf{x}_{i_1}\mathbf{x}_{i_2}\ldots\mathbf{x}_{i_s} \text{ has Hamming weight } 2^{m-s}$$

for arbitrary pairwise distinct indices $i_1, \ldots, i_s$ chosen between 0 and $m-1$.

Remark. (4) A Boolean polynomial never uses exponents, simply because $x_i^2 = x_i$. Thus, there are just four Boolean polynomial of one indeterminate $\mathbf{x}$, viz.: x, $x+1$, 0, and 1.

We will later introduce (non-Boolean) polynomials of one indeterminate—those are fundamentally different from the Boolean polynomials since all the information is checked by the exponents.

Translation between words and Boolean polynomials. Every Boolean polynomial of m-variables yields a binary word of length 2^m: for a single indeterminate, see Example (2) in 9.1, and, further, we perform the required additions and multiplications.

Conversely, every binary word $\mathbf{f} = f_0f_1\ldots f_{2^m-1}$ can be translated into a Boolean polynomial as follows. First, observe that the last indeterminate $\mathbf{x}_{m-1}$ is the word

$$\mathbf{x}_{m-1} = 00\ldots0011\ldots11,$$

whose first half is the constant function 0 and the other half is the constant function 1 (now both considered in $m-1$ indeterminates, i.e., in the length $\frac{1}{2}2^m = 2^{m-1}$). As a consequence, we see that for each Boolean function $f(x_0, x_1, \ldots, x_{m-1})$, the first half of the corresponding word $\mathbf{f}$ is $f(x_0, x_1, \ldots, x_{m-2}, 0)$ and the other half is $f(x_0, x_1, \ldots, x_{m-2}, 1)$ (both considered as functions of $m-1$ indeterminates).

The following trivial proposition, applied recursively, then yields an algorithm for converting a word into a Boolean polynomial:

Proposition. *Every Boolean function $\mathbf{f}$ of m variables can be expressed [by means of the two halves $f(x_0, \ldots, x_{m-2}, 0)$ and $f(x_0, \ldots, x_{m-2}, 1)$ of the corresponding binary word] as follows:*

$$f(x_0, \ldots, x_{m-2}, x_{m-1}) = f(x_0, \ldots, x_{m-2}, 0) + [f(x_0, \ldots, x_{m-2}, 0) + f(x_0, \ldots, x_{m-2}, 1)]x_{m-1}.$$

PROOF. Since x_{m-1} can only take one of the two possible values, 0 or 1, it is sufficient to verify that the identity holds for both of them. For $x_{m-1} = 0$, the identity is clear, and for $x_{m-1} = 1$, we get

$$f(x_0, \ldots, x_{m-2}, 1) = f(x_0, \ldots, x_{m-2}, 0) + [f(x_0, \ldots, x_{m-2}, 0) + f(x_1, \ldots, x_{m-2}, 1)].$$

□

Example. (3) Let us translate $\mathbf{f} = 01101110$ into a Boolean polynomial (of three variables). We apply the preceding proposition:

$$\begin{aligned}\mathbf{f} &= 0110 + [0110 + 1110]x_2\\ &= 0110 + 1000x_2.\end{aligned}$$

Next we apply the same proposition to the two words 0110 and 1000 (of two indeterminates):

$$\begin{aligned}\mathbf{f} &= (01 + [01 + 10]x_1) + (10 + [10 + 00]x_1)x_2\\ &= 01 + 11x_1 + 10x_2 + 10x_1x_2.\end{aligned}$$

Finally, an application of the proposition to the words of length 2 (in the indeterminate x_0) yields:

$$\begin{aligned}\mathbf{f} &= (0 + [0 + 1]x_0) + (1 + [1 + 1]x_0)x_1 + (1 + [1 + 0]x_0)x_2\\ &\quad + (1 + [1 + 0]x_0)x_1x_0\\ &= x_0 + x_1 + x_2 + x_0x_2 + x_1x_2 + x_0x_1x_2.\end{aligned}$$

Theorem. *The linear space $\mathbf{Z}_2^n$, where $n = 2^m$, has a basis formed by all one-summand Boolean polynomials, i.e., by the following polynomials*

$$\begin{aligned}&1,\\ &\mathbf{x}_i \quad (i = 0, 1, \ldots, m-1),\\ &\mathbf{x}_i\mathbf{x}_j \quad (i, j = 0, 1, \ldots, m-1 \text{ and } i \neq j),\\ &\vdots\\ &\mathbf{x}_0\mathbf{x}_1\ldots\mathbf{x}_{m-1}.\end{aligned}$$

PROOF. Every word of length $n = 2^m$ is a Boolean function of m indeterminates, and, hence, can be expressed as a Boolean polynomial. It follows that the one-summand Boolean polynomials span the whole space $\mathbf{Z}_2^n$. In order to prove that they form a basis, it is sufficient to show that their number is $n = 2^m$, the dimension of the space. In fact, we have 1 polynomial of degree 0, m polynomials of degree 1, $\binom{m}{2}$ polynomials of degree 2, etc. For each $k = 0, 1, \ldots, m-1$, there are $\binom{m}{k}$ one-summand polynomials of degree k, and, thus, the total number of the one-summand polynomials is

$$\sum_{k=0}^{m} \binom{m}{k} = 2^m. \tag{9.2.1}$$

[The last equation is easily derived from the binomial theorem applied to $(1+1)^m$.] □

9.3 Reed-Muller Codes

Definition. *By the* Reed-Muller code *of length $n = 2^m$ and degree r $(= 0, 1, \ldots, m)$ is meant the binary code $\mathcal{R}(r, m)$ of all binary words of length n, which have, as Boolean polynomials, degree at most r.*

Examples

(1) $\mathcal{R}(0, m)$ consists of the polynomials of degree at most 0, i.e., of **0** and **1**. Thus, $\mathcal{R}(0, m)$ is the repetition code of length 2^m.

(2) $\mathcal{R}(1, m)$ has basis $\mathbf{1}, \mathbf{x}_0, \ldots, \mathbf{x}_{m-1}$. In fact, each polynomial of degree at most 1 is a sum of (some of) those $m+1$ polynomials, and they are linearly independent by Theorem 9.2. Thus, $\mathcal{R}(1, m)$ is a $(2^m, m+1)$-code.

For example, $\mathcal{R}(1, 3)$ has the following generator matrix:

$$\mathbf{G} = \begin{bmatrix} \mathbf{1} \\ \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

We will see that $\mathcal{R}(1, 3)$ is the extended Hamming code (see 8.5). $\mathcal{R}(1, 4)$ is a $(16, 5)$-code and $\mathcal{R}(1, 5)$ is a $(32, 6)$-code, which was used by the 1969 Mariner, as mentioned in the introduction.

(3) $\mathcal{R}(2, m)$ has basis $\mathbf{1}, \mathbf{x}_0, \ldots, \mathbf{x}_{m-1}, \mathbf{x}_0\mathbf{x}_1, \mathbf{x}_0\mathbf{x}_2, \ldots, \mathbf{x}_{m-2}\mathbf{x}_{m-1}$. Thus, this is a $\left(2^m, \binom{m}{2} + m + 1\right)$-code. For example, $\mathcal{R}(2, 3)$ is a $(8, 7)$-code.

It is easy to see that this is just the even-parity code of length 8. $\mathcal{R}(2, 4)$ is a $(16, 11)$-code. We will see that $\mathcal{R}(2, 4)$ is the extended Hamming code.

(4) $\mathcal{R}(m-1, m)$ is the even-parity code of length 2^m. In fact, every code word of $\mathcal{R}(m-1, m)$ is a sum of the polynomials $\mathbf{x}_{i_1}\mathbf{x}_{i_2}\ldots\mathbf{x}_{i_s}$, where $s \leq m-1$. Each of these polynomials has an even Hamming weight [see Example (2) of 9.2]. Thus, $\mathcal{R}(m-1, m)$ is a subspace of the even-parity code. Moreover, $\mathcal{R}(m-1, m)$ has dimension $2^m - 1$, because it contains all of the basis polynomials in Theorem 9.2 except the last one, $\mathbf{x}_0\mathbf{x}_1\ldots\mathbf{x}_{m-1}$. Since the even-parity code also has dimension $2^m - 1$, the two codes are identical by Corollary (4) of 7.4.

Theorem. *The Reed-Muller code $\mathcal{R}(r, m)$ has*

$$k = \sum_{i=0}^{r} \binom{m}{i}$$

information symbols, and its dual code is $\mathcal{R}(m-r-1, m)$.

PROOF. I. The space $\mathcal{R}(r, m)$ is spanned by all the polynomials $\mathbf{x}_{i_1}\ldots\mathbf{x}_{i_s}$, where $0 \leq s \leq r$, and for a given s, we have $\binom{m}{s}$ such polynomials. By Theorem 9.2, all these polynomials are linearly independent, thus, they form a basis of $\mathcal{R}(r, m)$. We conclude that $\mathcal{R}(r, m)$ has dimension $\sum_{s=0}^{r} \binom{m}{s}$.

II. The dimension of $\mathcal{R}(r, m)^\perp$ is $2^m - \sum_{s=0}^{r} \binom{m}{s}$ (by Theorem 7.7). Using Equation (9.2.1) and the well-known identity

$$\binom{m}{s} = \binom{m}{m-s},$$

we see that the dimension of $\mathcal{R}(r, m)^\perp$ can be re-written as follows:

$$\begin{aligned} \dim \mathcal{R}(r, m)^\perp &= \sum_{s=0}^{m} \binom{m}{s} - \sum_{s=0}^{r} \binom{m}{s} \\ &= \sum_{s=r+1}^{m} \binom{m}{s} \\ &= \sum_{s=r+1}^{m} \binom{m}{m-s} \\ &= \sum_{i=0}^{m-s-1} \binom{m}{i}. \end{aligned}$$

We conclude that linear spaces $\mathcal{R}(r,m)^\perp$ and $\mathcal{R}(m-r-1,m)$ have the same dimension. By Corollary (4) of 7.4, it is now sufficient to show that $\mathcal{R}(m-r-1,m)$ is a subspace of the space $\mathcal{R}(r,m)^\perp$.

Thus, it is our task to verify that each Boolean polynomial $\mathbf{f}$ of degree $p \le m-r-1$ is orthogonal to all Boolean polynomials $\mathbf{g}$ of degree $q \le r$ [and, thus, $\mathbf{f}$ lies in $\mathcal{R}(r,m)^\perp$]. Since the scalar product $\mathbf{f}\cdot\mathbf{g}$ is the sum of the coordinates $f_i g_i$ of the logical product $\mathbf{fg}$, we just have to show that $\mathbf{fg}$, represented as a binary word, has an even Hamming weight. The degree of the polynomial $\mathbf{fg}$ is at most $p+q \le m-r-1+r = m-1$ (see Exercise 9A). Thus, $\mathbf{fg}$ is a code word of $\mathcal{R}(m-1,m)$, which by Example (4) above implies that $\mathbf{fg}$ has an even Hamming weight. □

Example. (5) $\mathcal{R}(m-2,m)$ is the extended Hamming code of length 2^m (see 8.5). In fact, the dual code is

$$\mathcal{R}(m-2,m)^\perp = \mathcal{R}(1,m)$$

and, thus, $\mathcal{R}(m-2,m)$ has the following parity check matrix:

$$\mathbf{H} = \begin{bmatrix} 1 \\ \mathbf{x}_0 \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_m \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & \cdots & 0 & 0 & 1 & 1 \\ & & & & \cdots & & & & \\ 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 1 & 1 \end{bmatrix}.$$

We can add the first row to all the other rows, and then interchange it with the last row. This yields another parity check matrix:

$$\mathbf{H} \sim \left[\begin{array}{ccccccccc|c} 1 & 0 & 1 & 0 & \cdots & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & \cdots & 1 & 1 & 0 & 0 \\ & & & & \cdots & & & & \\ 1 & 1 & 1 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 1 & 1 & \cdots & 1 & 1 & 1 & 1 \end{array}\right].$$

By deleting the last column and the last row from the new matrix, we obtain a 2^m-1 by m matrix $\mathbf{H}_0$ with pairwise distinct, nonzero columns. Thus, $\mathbf{H}_0$ is a parity check matrix of the Hamming code. Consequently, our matrix

$$\mathbf{H} \sim \left[\begin{array}{ccccccc|c} & & & & & & & 0 \\ & & & \mathbf{H}_0 & & & & 0 \\ & & & & & & & \vdots \\ & & & & & & & 0 \\ \hline 1 & 1 & 1 & 1 & \cdots & 1 & 1 & 1 \end{array}\right]$$

is a parity check matrix of the extended Hamming code.

Remarks

(1) Encoding of the Reed-Muller codes can be performed in the usual way by multiplying the information word by the generator matrix [see Remark (4) of 8.1]. In other words, the information bits become the coefficients of the corresponding Boolean polynomial. For example, in $\mathcal{R}(1,m)$, we encode the $m+1$ information bits as follows:

$$[u_1, u_2, \ldots, u_{m+1}] \begin{bmatrix} 1 \\ \mathbf{x}_0 \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_{m-1} \end{bmatrix} = u_1 + u_2\mathbf{x}_0 + \cdots + u_{m+1}\mathbf{x}_{m-1}.$$

(2) The minimum weight of the Reed-Muller code $\mathcal{R}(r,m)$ is

$$d = 2^{m-r}.$$

In fact, the code word $\mathbf{x}_0\mathbf{x}_1\ldots\mathbf{x}_{r-1}$ has a Hamming weight 2^{m-r} [see Example (2) of 9.2], thus, $d \le 2^{m-r}$. Since $\mathcal{R}(r,m)$ is a subspace of the even-parity code $\mathcal{R}(m-1,m)$, we know that d is even. In 9.6, we will see that $\mathcal{R}(r,m)$ can correct $2^{m-r-1}-1$ errors. Thus, by Proposition 4.6, $d > 2(2^{m-r-1}-1)$, and we conclude that

$$2^{m-r} \le d < 2^{m-r} - 2.$$

Since d is even, this proves $d = 2^{m-r}$.

(3) In some respect, it is more suitable to work with the *punctured Reed-Muller codes* $\overline{\mathcal{R}}(r,m)$ (see 8.6). This means that in $\mathcal{R}(r,m)$, we delete the last symbol from each code word. The resulting code $\overline{\mathcal{R}}(r,m)$ has length 2^m-1, and the number of information symbols can be shown to be equal to that in $\mathcal{R}(r,m)$. The minimum Hamming distance of $\overline{\mathcal{R}}(r,m)$ is $2^{m-r}-1$ and, thus, it can correct $2^{m-r-1}-1$ errors, the same number as the original code can. We list all punctured Reed-Muller codes of lengths $7, \ldots, 127$ in Appendix B.

For example, $\overline{\mathcal{R}}(0,m)$ is the repetition code of length 2^m-1 and $\overline{\mathcal{R}}(m-1,m)$ is the Hamming code of that length.

9.4 Geometric Interpretation: Three-Dimensional Case

In order to explain the decoding of Reed-Muller codes, it is convenient to introduce a new interpretation: instead of with Boolean functions, we

work with flats (or affine subspaces). We first present the case of the three-dimensional geometry, which corresponds to the codes of length 8, and then the general geometry of $\mathbf{Z}_2^m$.

Recall that the conventional three-dimensional Euclidean geometry operates within the linear space $\mathbf{R}^3$, whose points (or vectors) are triples $\mathbf{a} = a_1a_2a_3$ of real numbers. We have lines in $\mathbf{R}^3$, which can be described as follows:

$$\mathbf{a} + t\,\mathbf{b} \qquad (\mathbf{a}, \mathbf{b} \text{ in } \mathbf{R}^3, \mathbf{b} \neq \mathbf{0}).$$

Here t denotes a real parameter, thus, the line $\mathbf{a}+t\,\mathbf{b}$ is the set of all points $\{\mathbf{a}+t\,\mathbf{b} \mid t \in \mathbf{R}\}$. Further, we have planes in $\mathbf{R}^3$:

$$\mathbf{a} + t\,\mathbf{b} + s\,\mathbf{c} \qquad (\mathbf{a}, \mathbf{b}, \mathbf{c} \text{ in } \mathbf{R}^3, \mathbf{b} \text{ and } \mathbf{c} \text{ linearly independent}).$$

Here, again, t and s are real parameters.

Now, the *binary Euclidean three-dimensional geometry* can be introduced quite analogously. Its *points* are the vectors of the linear space $\mathbf{Z}_2^3$. In contrast to the real case above, there are precisely eight points. We can enumerate them by the corresponding binary expansions of their indices: $\mathbf{p}_0 = 000$, $\mathbf{p}_1 = 001$, etc., see Figure 5.

Point	Characteristic Function
$\mathbf{p}_0 = 000$	00000001
$\mathbf{p}_1 = 001$	00000010
$\mathbf{p}_2 = 010$	00000100
$\mathbf{p}_3 = 011$	00001000
$\mathbf{p}_4 = 100$	00010000
$\mathbf{p}_5 = 101$	00100000
$\mathbf{p}_6 = 110$	01000000
$\mathbf{p}_7 = 111$	10000000

Figure 5: Points of the binary three-dimensional Euclidean geometry

Lines of the Euclidean geometry can be described as follows:

$$\mathbf{a} + t\,\mathbf{b} \qquad (\mathbf{a}, \mathbf{b} \text{ in } \mathbf{Z}_2^3, \mathbf{b} \neq \mathbf{0}),$$

where t is a binary parameter, $t = 0, 1$. Thus, the line has precisely two points: $\mathbf{a}$ and $\mathbf{a} + \mathbf{b}$. Conversely, every pair $\mathbf{a}$, $\mathbf{a}'$ of points constitutes a line, viz.,

$$\mathbf{a} + t(\mathbf{a}' - \mathbf{a}).$$

It follows that lines are just all two-point subsets:

$$\{\mathbf{p}_0, \mathbf{p}_1\},\ \{\mathbf{p}_0, \mathbf{p}_2\},\ \ldots,\ \{\mathbf{p}_6, \mathbf{p}_7\}.$$

The number of lines is

$$\binom{8}{2} = 27,$$

since a line is just a choice of an (unordered) pair from among the eight points.

Finally, *planes* of the Euclidean geometry are described as

$$\mathbf{a} + t_1\mathbf{b}_1 + t_2\mathbf{b}_2 \qquad (\mathbf{a}, \mathbf{b}_1, \mathbf{b}_2 \text{ in } \mathbf{Z}_2^3;\ \mathbf{b}_1, \mathbf{b}_2 \text{ linearly independent}),$$

where t_1 and t_2 are binary parameters. The plane, then, consists of the following four points: $\mathbf{a}$, $\mathbf{a} + \mathbf{b}_1$, $\mathbf{a} + \mathbf{b}_2$, and $\mathbf{a} + \mathbf{b}_1 + \mathbf{b}_2$. All planes are listed in Figure 6.

Plane	Characteristic Function	Boolean Polynomial
$\{\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_5, \mathbf{p}_7\}$	10101010	x_0
$\{\mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_6, \mathbf{p}_7\}$	11001100	x_1
$\{\mathbf{p}_4, \mathbf{p}_5, \mathbf{p}_6, \mathbf{p}_7\}$	11110000	x_2
$\{\mathbf{p}_0, \mathbf{p}_2, \mathbf{p}_4, \mathbf{p}_6\}$	01010101	$1 + x_0$
$\{\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_4, \mathbf{p}_5\}$	00110011	$1 + x_1$
$\{\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$	00001111	$1 + x_2$
$\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_5, \mathbf{p}_6\}$	01100110	$x_0 + x_1$
$\{\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_4, \mathbf{p}_6\}$	01011010	$x_0 + x_2$
$\{\mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4, \mathbf{p}_5\}$	00111100	$x_1 + x_2$
$\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4, \mathbf{p}_7\}$	10010110	$x_0 + x_1 + x_2$
$\{\mathbf{p}_0, \mathbf{p}_3, \mathbf{p}_4, \mathbf{p}_7\}$	10011001	$1 + x_0 + x_1$
$\{\mathbf{p}_0, \mathbf{p}_2, \mathbf{p}_5, \mathbf{p}_7\}$	10100101	$1 + x_0 + x_2$
$\{\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_6, \mathbf{p}_7\}$	11000011	$1 + x_1 + x_2$
$\{\mathbf{p}_0, \mathbf{p}_3, \mathbf{p}_5, \mathbf{p}_6\}$	01101001	$1 + x_0 + x_1 + x_2$

Figure 6: Planes in the binary three-dimensional Euclidean geometry

Observe that the planes passing through the origin ($\mathbf{a} = \mathbf{0}$) are precisely those which can be described by a homogenous equation

$$h_0x_0 + h_1x_1 + h_2x_2 = 0.$$

In fact, such a plane is precisely a two-dimensional subspace of $\mathbf{Z}_2^3$, now see Remark (4) of 7.7. It follows that a general plane can always be described by a (possibly nonhomogenous) equation

$$h_0x_0 + h_1x_1 + h_2x_2 = c.$$

(Given a plane $\mathbf{a}+t_1\mathbf{b}_1+t_2\mathbf{b}_2$ and describing the parallel plane $t_1\mathbf{b}_1+t_2\mathbf{b}_2$ by the equation $h_0x_0+h_1x_1+h_2x_2=0$, put $c=h_0a_0+h_1a_1+h_2a_2$, where $\mathbf{a}=a_0a_1a_2$.)

Furthermore, every line can be described by a pair of nonhomogenous equations

$$\begin{aligned} h_0x_0 + h_1x_1 + h_2x_2 &= c, \\ h_0'x_0 + h_1'x_1 + h_2'x_2 &= c'. \end{aligned}$$

This follows from the fact that every line $\mathbf{a}+t\,\mathbf{b}$ is an intersection of two planes: choose a basis $\mathbf{b}$, $\mathbf{d}$, $\mathbf{d}'$ of the space $\mathbf{Z}_2^3$ and consider the planes $\mathbf{a}+t\,\mathbf{b}+s\,\mathbf{d}$ and $\mathbf{a}+t\,\mathbf{b}+s'\,\mathbf{d}'$.

Lines and planes are examples of *flats* (also called *affine spaces*). A *flat* in the space $\mathbf{Z}_2^3$ is a coset (6.2) of a linear subspace of the space $\mathbf{Z}_2^3$. That is, a flat has the form

$$\mathbf{a}+K=\{\,\mathbf{a}+\mathbf{b}\mid \mathbf{b} \text{ is a point of } K\,\},$$

where $\mathbf{a}$ is a point of $\mathbf{Z}_2^3$, and K is a linear subspace. If the dimension of K is s, we call the coset an s-flat. Thus, lines are preisely the 1-flats, and planes are the 2-flats. For each point $\mathbf{p}_i$, we have a 0-flat $\{\,\mathbf{p}_i\,\}$, and there is precisely one 3-flat, viz., the whole space $\mathbf{Z}_2^3$.

Every flat L can be described by the binary word $\mathbf{f}_L=f_7\ldots f_1f_0$ defined by

$$f_i=\begin{cases} 1 & \text{if the point } \mathbf{p}_i \text{ lies in } L, \\ 0 & \text{otherwise.} \end{cases}$$

The word $\mathbf{f}_L$ (or the correspondig Boolean function of three variables) is called the ***characteristic function*** of the flat L. (For 1-flats and 2-flats, see the above figures.)

Given flats L and L', their intersection $L\cap L'$ is obviously characterized by the logical product $\mathbf{f}_L\mathbf{f}_{L'}$. For example, the first two planes in Figure 6 intersect in the line $\{\,\mathbf{p}_3,\mathbf{p}_7\,\}$. The logical product of their characteristic functions is

$$\mathbf{x}_0\mathbf{x}_1 = 10001000,$$

which is indeed the characteristic function of $\{\,\mathbf{p}_3,\mathbf{p}_7\,\}$.

Remark. The Reed-Muller code $\mathcal{R}(1,3)$ is spanned by the characteristic functions of all planes. In fact, we see in Figure 6 that the characteristic functions of planes are precisely all Boolean polynomials of degree 1 in three variables.

The Reed-Muller code $\mathcal{R}(2,3)$ is spanned by the characteristic functions of all planes and all lines. In fact, every line L is an intersection of two planes. Hence, the characteristic function $\mathbf{f}_L$ is a product of two Boolean polynomials of degree 1, which is a Boolean polynomial of degree 2—a code word of $\mathcal{R}(2,3)$.

9.5 Geometric Interpretation: General Case

We now pass to the Euclidean geometry of the m-dimensional binary linear space $\mathbf{Z}_2^m$. The points, or vectors, of $\mathbf{Z}_2^m$ can, again, be ordered by the binary expansions of the numbers $0, 1, \ldots, 2^m-1$. That is, we put $\mathbf{Z}_2^m=\{\,\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\ldots,\mathbf{p}_{2^m-1}\,\}$, where

$$\begin{aligned} \mathbf{p}_0 &= 000\ldots00, \\ \mathbf{p}_1 &= 000\ldots01, \\ \mathbf{p}_2 &= 000\ldots10, \\ &\vdots \\ \mathbf{p}_{2^m-1} &= 111\ldots11. \end{aligned}$$

Definition. *Let K be an r-dimensional linear subspace of the space $\mathbf{Z}_2^m$. Every coset*

$$\mathbf{a}+K=\{\,\mathbf{a}+\mathbf{b}\mid \mathbf{b} \textit{ lies in } K\,\}$$

of K is called an r-flat in the binary m-dimensional Euclidean geometry.

An $(m-1)$-flat is also called a hyperplane.

Notation. Given a basis $\mathbf{b}_1,\ldots,\mathbf{b}_r$ of the subspace K, the r-flat $\mathbf{a}+K$ is also denoted by

$$\mathbf{a}+t_1\mathbf{b}_1+\cdots+t_r\mathbf{b}_r.$$

It has 2^r points (given by the r choices $t_i=0,1$ for $i=1,\ldots,r$).

Another notation of the r-flat $\mathbf{a}+K$ is by means of a nonhomogenous system of linear equations: whenever the space K is described by equations $\mathbf{H}\mathbf{x}^{\mathrm{tr}}=\mathbf{0}^{\mathrm{tr}}$ [see Remark (4) of 7.7], then the r-flat $\mathbf{a}+K$ is described by the following equations:

$$\mathbf{H}\mathbf{x}^{\mathrm{tr}}=\mathbf{c}^{\mathrm{tr}}, \qquad \text{where} \quad \mathbf{c}^{\mathrm{tr}}=\mathbf{H}\mathbf{a}^{\mathrm{tr}}.$$

The number of these equations is $m - r$. For example, each hyperplane is described by a single equation:

$$h_0x_0 + h_1x_1 + \cdots + h_{m-1}x_{m-1} = c.$$

Examples

(1) Every 0-flat is a one-point set. Thus, there are 2^m different 0-flats, viz., $\{\mathbf{p}_0\}, \ldots, \{\mathbf{p}_{2^m-1}\}$.

(2) Every 1-flat (or line) is a two-point set,

$$\mathbf{a} + t\,\mathbf{b} \equiv \{\mathbf{a}, \mathbf{a} + \mathbf{b}\},$$

and, conversely, every two-point set is a 1-flat. Therefore, there are $\binom{2^m}{2}$ 1-flats.

(3) Let P_i denote the flat described by the equation $x_i = 1$. That is, P_i is the set of all points $\mathbf{p}_k$ which have a 1 in the ith position. For example, $P_0 = \{\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_5, \ldots, \mathbf{p}_{2^m-1}\}$.

Each P_i is a hyperplane. In fact, the point $\mathbf{p}_{2^i}$ has precisely one nonzero position (the ith one), and, thus,

$$P_i \equiv \mathbf{p}_{2^i} + K,$$

where K is the linear space determined by the equation $k_i = 0$. It is clear that the dimension of K is $m - 1$.

The number of all hyperplanes is $2(2^m - 1)$. In fact, the space $\mathbf{Z}_2^m$ has precisely $2^m - 1$ subspaces K of dimension $m - 1$ (see Exercise 7H). Each of these subspaces K has 2^{m-1} points and, thus, by Remark 6.2, there are $2^m/2^{m-1}$ cosets modulo K.

(4) For $i \neq j$, the intersection $P_i \cap P_j$ (i.e., the set of all points with a 1 in the ith and jth positions) is an $(m-2)$-flat. In fact, for the point $\mathbf{a} = \mathbf{p}_{2^i+2^j}$ (with 1's just on the ith and jth positions), we have

$$P_i \cap P_j \equiv \mathbf{a} + K,$$

where K is determined by the equation $k_i = k_j = 0$. The dimension of K is $m - 2$.

Definition. *By the* characteristic function *of an r-flat L is meant the binary word $\mathbf{f}_L = f_{2^m-1} \ldots f_1 f_0$ defined by*

$$f_j = \begin{cases} 1 & \text{if the point } \mathbf{p}_j \text{ lies in } L, \\ 0 & \text{otherwise.} \end{cases}$$

Remark. The characteristic function $\mathbf{f}_L$ can be interpreted as a Boolean polynomial $f_L(x_0, x_1, \ldots, x_{m-1})$. It follows from 9.1 that L consists of precisely those points $\mathbf{p}_i = x_0x_1 \ldots x_{m-1}$ which satisfy $f_L(x_0, x_1, \ldots, x_{m-1}) = 1$. Shortly:

$$x_0x_1 \ldots x_{m-1} \text{ lies in } L \iff f_L(x_0, x_1, \ldots, x_{m-1}) = 1.$$

Examples

(5) The only m-flat, i.e., the space $\mathbf{Z}_2^m$, has the characteristic function $\mathbf{1} = 111 \ldots 11$. For the hyperplane P_i above $f_{P_i} = x_i$.

(6) Every hyperplane has a characteristic function which is a Boolean polynomial of degree 1: if the hyperplane L is described by the equation

$$h_0x_0 + h_1x_1 + \cdots + h_{m-1}x_{m-1} = c,$$

then

$$f_L(x_0, \ldots, x_{m-1}) = h_0x_0 + h_1x_1 + \cdots + h_{m-1}x_{m-1} + c + 1.$$

In fact, a point $x_0x_1 \ldots x_{m-1}$ lies in the hyperplane precisely when $h_0x_0 + \cdots + h_{m-1}x_{m-1} = c$, i.e., when $f_L(x_0, x_1, \ldots, x_{m-1}) = 1$.

(7) For two flats L and L', the function $\mathbf{f}_L\mathbf{f}_{L'}$ is the characteristic function of the intersection $L \cap L'$.

For example, the polynomial $\mathbf{x}_i\mathbf{x}_j$ is the characteristic function of the $(m-2)$-flat, which is the intersection of the hyperplanes P_i and P_j. More in general: the Boolean polynomial

$$\mathbf{x}_{i_1}\mathbf{x}_{i_2} \ldots \mathbf{x}_{i_s}$$

is the characteristic function of an $(m-s)$-flat.

Theorem. *The characteristic function of an r-flat is a Boolean polynomial of degree $m - r$.*

PROOF. In the notation above we have described each r-flat L by $m - r$ equations $\mathbf{H}\mathbf{x}^{\mathrm{tr}} = \mathbf{c}^{\mathrm{tr}}$, or

$$\sum_{j=0}^{m-1} h_{ij}x_j = c_i \qquad \text{for } i = 1, 2, \ldots, m - r.$$

We can rewrite the equations as follows:

$$\sum_{j=0}^{m-1} (h_{ij}x_j + c_i + 1) = 1 \qquad \text{for } i = 1, 2, \ldots, m - r.$$

Then the following Boolean polynomial of degree $m-r$

$$f(x_1,\ldots,x_{m-1}) = \prod_{i=1}^{m-r}\sum_{j=0}^{m-1}(h_{ij}x_j + c_i + 1)$$

is the characteristic function of L: the equation $f(x_0,\ldots,x_{m-1}) = 1$ holds precisely when $\sum_{j=0}^{m-1}(h_{ij}x_j + c_i + 1) = 1$ for each $i = 1, \ldots, m-r$. □

Corollary. (1) *Reed-Muller codes can be characterized geometrically as follows: $\mathcal{R}(r,m)$ is spanned by all characteristic functions of flats of dimension at least $m-r$ in the binary m-dimensional geometry over $\mathbf{Z}_2$.*

(2) *Every characteristic function of an $(r+1)$-flat lies in the dual code of $\mathcal{R}(r,m)$.*

In fact, $\mathcal{R}(r,m)$ contains all characteristic functions of s-flats, $s \geq m-r$, by the preceding theorem. That those functions span the space $\mathcal{R}(r,m)$ follows from Example (7). Thus, (1) is true, and (2) follows from Theorem 9.3. □

9.6 Decoding Reed-Muller Codes

We now present an interesting and easily implementable decoding technique for the code $\mathcal{R}(r,m)$. It is based on majority logic, and it can correct $2^{m-r-1}-1$ errors. In contrast to other decoding techniques, the present method does not use syndromes, rather it directly computes the corrupted bits from the properties of the received word. The idea is as follows. We receive a binary word $\mathbf{w} = w_{2^m-1}\ldots w_1 w_0$ of length 2^m. Assuming that less then 2^{m-r-1} bits are corrupted, we want to determine, for each $i = 0, \ldots, 2^m-1$, whether or not w_i should be corrected. This can be reformulated by asking whether the position of $\mathbf{w}$ corresponding to the 0-flat $\{\mathbf{p}_i\}$ should be corrected.

Instead of answering this question directly, we take a broader point of view: for each s-flat L, where $s = 0, 1, \ldots, r+1$, we determine whether the positions of the received word $\mathbf{w}$ corresponding to the points of L (i.e., those bits w_i such that the point $\mathbf{p}_i$ lies in L) are corrupted or not. Well, not exactly. We just distinguish between "even" and "odd" s-flats: an s-flat L is called *even* if the number of all corrupted positions w_i corresponding to points $\mathbf{p}_i$ of L is even, otherwise L is *odd*. If we are able to determine the parity of each s-flat L, the decoding is clear: we correct a bit w_i if and only if the 0-flat $\{\mathbf{p}_i\}$ has odd parity. The trick is that we start with $s = r+1$ and then proceed to the lower dimensions.

Thus, the first step of decoding the word $\mathbf{w}$ is to determine, for each $(r+1)$-flat L, whether L is odd or even. This is performed by computing the scalar product of $\mathbf{w}$ with the characteristic function $\mathbf{f}_L$:

$$L \text{ is even} \iff \mathbf{w}\cdot\mathbf{f}_L = 0.$$

In fact: if $\mathbf{w}$ is a code word, then $\mathbf{w}\cdot\mathbf{f}_L = 0$ by Corollary (2) in 9.5. Now, if precisely two of the bits of $\mathbf{w}$ corresponding to points of L are corrupted, the value $\mathbf{w}\cdot\mathbf{f}_L$ will not be changed. Analogously with 4, 6, ... bits. But if an odd number of bits of $\mathbf{w}$ corresponding to points of L are corrupted, then the received word fulfils $\mathbf{w}\cdot\mathbf{f}_L = 1$.

For s-flats L, where $s \leq r$, we proceed by majority logic: suppose we already know the parity of every $(s+1)$-flat containing L, then we say that L is odd if a majority of these $(s+1)$-flats is odd, and L is even if a majority of them is even. The reason why this procedure works is that each s-flat is contained in a large number of $(s+1)$-flats:

Theorem. *Every s-flat L in the binary m-dimensional geometry is contained in exactly $2^{m-s}-1$ different $(s+1)$-flats. Furthermore, each point outside of L lies in precisely one of these $(s+1)$-flats.*

PROOF. I. We prove first that every s-dimensional linear subspace K of $\mathbf{Z}_2^m$ is contained in precisely $2^{m-s}-1$ different subspaces of dimension $s+1$.

Every $(s+1)$-dimensional space $\overline{K}$ containing K has the form $\overline{K} = K + t\,\mathbf{b}$ for some point $\mathbf{b}$ outside of K, where

$$K + t\,\mathbf{b} \equiv \{\mathbf{a} + t\,\mathbf{b} \mid \mathbf{a} \in K \text{ and } t = 0, 1\}.$$

This follows immediately from the fact that every basis of K can be extended (by a single vector) to a basis of $\overline{K}$—see Theorem 7.4.

Next observe that for two points $\mathbf{b}, \mathbf{b}'$ outside of K, the linear subspaces $K + t\,\mathbf{b}$ and $K + t\,\mathbf{b}'$ coincide if and only if $\mathbf{b}$ and $\mathbf{b}'$ lie in the same coset modulo K (6.2). In fact, if $K + t\,\mathbf{b} = K + t\,\mathbf{b}'$, then $\mathbf{b}$ can be expressed as $\mathbf{a} + t\,\mathbf{b}'$, where $\mathbf{a}$ lies in K—thus, $t \neq 0$ and we get

$$\mathbf{b} - \mathbf{b}' = \mathbf{a} \in K.$$

By Proposition 6.2, $\mathbf{b}$ and $\mathbf{b}'$ lie in the same coset. Conversely, if $\mathbf{b} - \mathbf{b}' = \mathbf{a}$ is a point of K, then $\mathbf{b} = \mathbf{a} + \mathbf{b}'$ lies in $K + t\,\mathbf{b}'$, and $\mathbf{b}' = -\mathbf{a} + \mathbf{b}$ lies in $K + t\,\mathbf{b}$; thus, $K + t\,\mathbf{b} = K + t\,\mathbf{b}'$. By Remark 6.2, there are $2^m/2^s$ cosets modulo K. One of them is K itself, and all other cosets contain only points outside of K. Consequently, there are $2^{m-s}-1$ different spaces $K + t\,\mathbf{b}$ for $\mathbf{b}$ outside of K.

II. Every s-flat

$$L \equiv \mathbf{a} + K \qquad (\dim K = s)$$

is contained in $2^{m-s}-1$ different $(s+1)$-flats $\mathbf{a}+K'$, where K' is an $(s+1)$-dimensional space containing K (this follows from I.). It remains to prove that every $(s+1)$-flat $\mathbf{a}'+K'$ containing L has the mentioned form. That is, we want to show that $\mathbf{a}+K' = \mathbf{a}'+K'$. In fact, since $\mathbf{a}$ lies in $\mathbf{a}'+K'$, the difference $\mathbf{a}-\mathbf{a}'$ is in K' and, hence, the points $\mathbf{a}$ and $\mathbf{a}'$ lie in the same coset modulo K.

III. Every point $\mathbf{b}$ outside of the s-flat $L = \mathbf{a}+K$ lies in an $(s+1)$-flat containing L, viz., the flat $\mathbf{a}+K'$, where $K' = K + t(\mathbf{b}-\mathbf{a})$. In fact, $\mathbf{b}$ lies in $\mathbf{a}+[K+t(\mathbf{b}-\mathbf{a})]$ because by choosing $t=1$ and $0 \in K$, we have

$$\mathbf{b} = \mathbf{a} + [0 + (\mathbf{b}-\mathbf{a})].$$

To verify that K' has dimension $s+1$, it is sufficient to show that $\mathbf{b}-\mathbf{a}$ lies outside of K: in fact, if $\mathbf{b}-\mathbf{a}$ lies in K, then $\mathbf{a}+(\mathbf{b}-\mathbf{a}) = \mathbf{b}$ lies in L.

Finally, we prove that the $(s+1)$-flat is unique. Every $(s+1)$-flat containing $\mathbf{a}+K$ has the form $\mathbf{a}+K'$, where K' is an $(s+1)$-dimensional linear space containing K. If $\mathbf{b}$ lies in such a flat $\mathbf{a}+K'$, then $\mathbf{b}-\mathbf{a}$ is also a point of K'; thus, K' contains the linear space $K+(\mathbf{b}-\mathbf{a})$. The last two spaces have both dimensions $s+1$, hence, they coincide. □

Corollary. *If the number of errors in a received word is less then 2^{m-r-1}, then for each s-flat L, $0 \le s \le r$, the majority of $(s+1)$-flats containing L have the same parity of errors as L.*

In fact, let $t < 2^{m-r-1}$ bits of the received word $\mathbf{w}$ be corrupted. We know that L is contained in

$$2^{m-r} - 1 > 2t$$

$(s+1)$-flats L', and each such flat L' is determined by one point outside of L. Let us begin with all points $\mathbf{p}_i$ outside of L such that w_i is a corrupted bit. There are at most t corresponding $(s+1)$-flats L'. All the remaining flats L' have the property that they contain no point $\mathbf{p}_i$ outside of L such that w_i is incorrect. Thus, L' has the same error parity as L. The number of the latter flats is at least $(2^{m-r}-1)-t > t$; thus, they form a majority. □

Decoding Algorithm for the Reed-Muller Code $\mathcal{R}(r,m)$

First step: Receiving a word $\mathbf{w}$, call each $(r+1)$-flat L odd if the scalar product of its characteristic function $\mathbf{f}_L$ with $\mathbf{w}$ is 1, otherwise call L even. That is, for $(r+1)$-flats L:

$$L \text{ is } \begin{cases} \text{odd if } \mathbf{w}\cdot\mathbf{f}_L = 1, \\ \text{even if } \mathbf{w}\cdot\mathbf{f}_L = 0. \end{cases}$$

Recursive steps: For all $s = r, r-1, \ldots, 0$ such that each $(s+1)$-flat has already been called odd or even, call an s-flat L odd if a majority of $(s+1)$-flats containing L is odd, otherwise call L even.

Last step: Correct the ith bit of $\mathbf{w}$ if and only if the 0-flat $\{\mathbf{p}_i\}$ has been called odd.

Example. Working with $\mathcal{R}(1,3)$, we have received

11101010.

The first step is to decide which planes (see Figure 6 in 9.4) are odd and which are even. For example, the plane $L = \{\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_5, \mathbf{p}_7\}$ is even since $\mathbf{w}\cdot\mathbf{f}_L = 11101010 \cdot 10101010 = 0$. See Figure 7. Next we must decide,

Plane	Parity	Plane	Parity
$\{\mathbf{p}_1,\mathbf{p}_3,\mathbf{p}_5,\mathbf{p}_7\}$	even	$\{\mathbf{p}_1,\mathbf{p}_3,\mathbf{p}_4,\mathbf{p}_6\}$	odd
$\{\mathbf{p}_2,\mathbf{p}_3,\mathbf{p}_6,\mathbf{p}_7\}$	odd	$\{\mathbf{p}_2,\mathbf{p}_3,\mathbf{p}_4,\mathbf{p}_5\}$	even
$\{\mathbf{p}_4,\mathbf{p}_5,\mathbf{p}_6,\mathbf{p}_7\}$	odd	$\{\mathbf{p}_0,\mathbf{p}_3,\mathbf{p}_4,\mathbf{p}_7\}$	even
$\{\mathbf{p}_0,\mathbf{p}_2,\mathbf{p}_4,\mathbf{p}_6\}$	odd	$\{\mathbf{p}_0,\mathbf{p}_2,\mathbf{p}_5,\mathbf{p}_7\}$	even
$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_4,\mathbf{p}_5\}$	even	$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_6,\mathbf{p}_7\}$	odd
$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3\}$	even	$\{\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_4,\mathbf{p}_7\}$	even
$\{\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_5,\mathbf{p}_6\}$	odd	$\{\mathbf{p}_0,\mathbf{p}_3,\mathbf{p}_5,\mathbf{p}_6\}$	odd

Figure 7: First step of decoding 11101010

for each line L, whether L is odd or even. For example, the line $\{\mathbf{p}_0,\mathbf{p}_1\}$ is contained in three planes (see Figure 6 of 9.4):

$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_4,\mathbf{p}_5\}$ – even,
$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3\}$ – even,
$\{\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_6,\mathbf{p}_7\}$ – odd.

Line	Pariry	Line	Pariry
$\{p_0, p_1\}$	even	$\{p_2, p_4\}$	even
$\{p_0, p_2\}$	even	$\{p_2, p_5\}$	even
$\{p_0, p_3\}$	even	$\{p_2, p_6\}$	odd
$\{p_0, p_4\}$	even	$\{p_2, p_7\}$	even
$\{p_0, p_5\}$	even	$\{p_3, p_4\}$	even
$\{p_0, p_6\}$	odd	$\{p_3, p_5\}$	even
$\{p_0, p_7\}$	even	$\{p_3, p_6\}$	odd
$\{p_1, p_2\}$	even	$\{p_3, p_7\}$	even
$\{p_1, p_3\}$	even	$\{p_4, p_5\}$	even
$\{p_1, p_4\}$	even	$\{p_4, p_6\}$	odd
$\{p_1, p_5\}$	even	$\{p_4, p_7\}$	even
$\{p_1, p_6\}$	even	$\{p_5, p_6\}$	odd
$\{p_1, p_7\}$	even	$\{p_5, p_7\}$	even
$\{p_2, p_3\}$	even	$\{p_6, p_7\}$	odd

Figure 8: Second step of decoding 11101010

By majority vote, the line $\{p_0, p_1\}$ is even. We must go through all the lines and perform such a majority vote. The result is seen in Figure 8.

Finaly, we are prepared to correct the individual bits. The point p_0 is contained in seven lines, one odd and six even; thus w_0 will not be corrected. Also p_1 is contained in one odd and six even lines, etc. The only bit to correct is w_6 because p_6 is contained in seven odd lines. The word sent is

10101010

[which is the polynomial $1 + x_1$, a code word of $\mathcal{R}(1,3)$].

Concluding Remark. We have presented a method by which the Reed-Muller code $\mathcal{R}(r,m)$ can correct $2^{m-r-1} - 1$ errors. As explained in Remark (1) of 9.3, this proves that the minimum distance is 2^{m-r}.

For example, the code $\mathcal{R}(1,5)$ is a $(32,6)$-code [see Example (1) in 9.3], which corrects $2^{5-1-1} - 1 = 7$ errors, as mentioned in the introduction.

Exercises

9A Prove the following about degrees of Boolean polynomials:

(1) For nonzero Boolean polynomials $\mathbf{f}$ and $\mathbf{g}$, the degree of $\mathbf{fg}$ is the sum of the degrees of $\mathbf{f}$ and $\mathbf{g}$.

(2) The degree of $\mathbf{f} + \mathbf{g}$ is the maximum of the degrees of $\mathbf{f}$ and $\mathbf{g}$.

9B Find a binary $(15,5)$-code correcting triple errors. (Hint: use a punctured Reed-Muller code.)

9C When using $\mathcal{R}(1,3)$, decode 01111100. Verify the correctness of your decoding. Encode information bits 1011.

9D When using $\mathcal{R}(2,3)$, decode 0111100.

9E When using $\mathcal{R}(2,4)$, decode 1111111011111111.

9F What Boolean polynomial has the truth table

(1) 10100110?

(2) 1010011010100110?

9G Find the truth table of the Boolean polynomial $1 + x_0 + x_1x_2$

(1) as a function of three variables,

(2) as a function of four variables.

9H What is the relationship between simplex codes (8.2) and the Reed-Muller codes $\mathcal{R}(1,m)$?

9I Compute the number of 2-flats in the Euclidean geometry in $\mathbf{Z}_2^m$.

9J Prove that each hyperplane L in the Euclidean geometry in $\mathbf{Z}_2^m$ has the property that its complement $\mathbf{Z}_2^m - L$ is a hyperplane too. (Hint: see the proof of Theorem 9.6.)

9K Is every Boolean function a characteristic function of some flat? Characterize such functions! (Hint: express each flat as an intersection of hyperplanes.)