Chapter 6

Introduction to binary block codes

In this chapter we begin to study binary signal constellations, which are the Euclidean-space images of binary block codes. Such constellations have bit rate (nominal spectral efficiency) $\rho \leq 2$ b/2D, and are thus suitable only for the power-limited regime.

We will focus mainly on binary linear block codes, which have a certain useful algebraic structure. Specifically, they are vector spaces over the binary field $\mathbb{F}_2$. A useful infinite family of such codes is the set of Reed-Muller codes.

We discuss the penalty incurred by making hard decisions and then performing classical error-correction, and show how the penalty may be partially mitigated by using erasures, or rather completely by using generalized minimum distance (GMD) decoding.

6.1 Binary signal constellations

In this chapter we will consider constellations that are the Euclidean-space images of binary codes via a coordinatewise 2-PAM map. Such constellations will be called *binary signal constellations*.

A *binary block code* of length n is any subset $\mathcal{C} \subseteq \{0,1\}^n$ of the set of all binary n-tuples of length n. We will usually identify the binary alphabet $\{0,1\}$ with the finite field $\mathbb{F}_2$ with two elements, whose arithmetic rules are those of mod-2 arithmetic. Moreover, we will usually impose the requirement that $\mathcal{C}$ be *linear*; *i.e.,* that $\mathcal{C}$ be a subspace of the n-dimensional vector space $(\mathbb{F}_2)^n$ of all binary n-tuples. We will shortly begin to discuss such algebraic properties.

Each component $x_k \in \{0,1\}$ of a codeword $\mathbf{x} \in \mathcal{C}$ will be mapped to one of the two points $\pm\alpha$ of a 2-PAM signal set $\mathcal{A} = \{\pm\alpha\} \subset \mathbb{R}$ according to a 2-PAM map $s\colon \{0,1\} \to \mathcal{A}$. Explicitly, two standard ways of specifying such a 2-PAM map are

$$\begin{aligned} s(x) &= \alpha(-1)^x; \\ s(x) &= \alpha(1-2x). \end{aligned}$$

The first map is more algebraic in that, ignoring scaling, it is an isomorphism from the additive binary group $\mathbb{Z}_2 = \{0,1\}$ to the multiplicative binary group $\{\pm 1\}$, since $s(x) \cdot s(x') = (-1)^{x+x'} = s(x+x')$. The second map is more geometric, in that it is the composition of a map from $\{0,1\} \in \mathbb{F}_2$ to $\{0,1\} \in \mathbb{R}$, followed by a linear transformation and a translation. However, ultimately both formulas specify the same map:

$$\{s(0) = \alpha, s(1) = -\alpha\}.$$

Under the 2-PAM map, the set $(\mathbb{F}_2)^n$ of all binary n-tuples maps to the set of all real n-tuples of the form $(\pm\alpha, \pm\alpha, \ldots, \pm\alpha)$. Geometrically, this is the set of all 2^n vertices of an n-cube of side 2α centered on the origin. It follows that a binary signal constellation $\mathcal{A}' = s(\mathcal{C})$ based on a binary code $\mathcal{C} \subseteq (\mathbb{F}_2)^n$ maps to a subset of the vertices of this n-cube.

The size of an N-dimensional binary constellation $\mathcal{A}'$ is thus bounded by $|\mathcal{A}'| \leq 2^n$, and its bit rate $\rho = (2/n)\log_2|\mathcal{A}'|$ is bounded by $\rho \leq 2$ b/2D. Thus binary constellations can be used only in the power-limited regime.

Since the n-cube constellation $\mathcal{A}^n = s((\mathbb{F}_2)^n) = (s(\mathbb{F}_2))^n$ is simply the n-fold Cartesian product $\mathcal{A}^n$ of the 2-PAM constellation $\mathcal{A} = s(\mathbb{F}_2) = \{\pm\alpha\}$, its normalized parameters are the same as those of 2-PAM, and it achieves no coding gain. Our hope is that by restricting to a subset $\mathcal{A}' \subset \mathcal{A}^n$, a distance gain can be achieved that will more than offset the rate loss, thus yielding a coding gain.

Example 1. Consider the binary code $\mathcal{C} = \{000, 011, 110, 101\}$, whose four codewords are binary 3-tuples. The bit rate of $\mathcal{C}$ is thus $\rho = 4/3$ b/2D. Its Euclidean-space image $s(\mathcal{C})$ is a set of four vertices of a 3-cube that form a regular tetrahedron, as shown in Figure 1. The minimum squared Euclidean distance of $s(\mathcal{C})$ is $d^2_{\min}(s(\mathcal{C})) = 8\alpha^2$, and every signal point in $s(\mathcal{C})$ has 3 nearest neighbors. The average energy of $s(\mathcal{C})$ is $E(s(\mathcal{C})) = 3\alpha^2$, so its average energy per bit is $E_b = (3/2)\alpha^2$, and its nominal coding gain is

$$\gamma_c(s(\mathcal{C})) = \frac{d^2_{\min}(s(\mathcal{C}))}{4E_b} = \frac{4}{3} \quad (1.25 \text{ dB}).$$

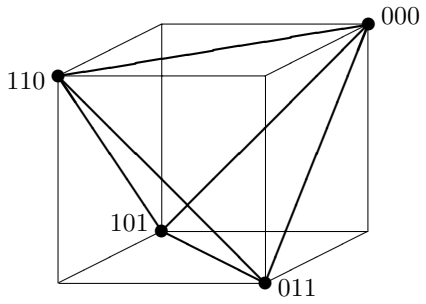


Figure 1. The Euclidean image of the binary code $\mathcal{C} = \{000, 011, 110, 101\}$ is a regular tetrahedron in $\mathbb{R}^3$.

It might at first appear that the restriction of constellation points to vertices of an n-cube might force binary signal constellations to be seriously suboptimal. However, it turns out that when ρ is small, this apparently drastic restriction does not hurt potential performance very much. A capacity calculation using a random code ensemble with binary alphabet $\mathcal{A} = \{\pm\alpha\}$ rather than $\mathbb{R}$ shows that the Shannon limit on E_b/N_0 at $\rho = 1$ b/2D is 0.2 dB rather than 0 dB; *i.e.,* the loss is only 0.2 dB. As $\rho \to 0$, the loss becomes negligible. Therefore at spectral efficiencies $\rho \leq 1$ b/2D, binary signal constellations are good enough.

6.2 Binary linear block codes as binary vector spaces

Practically all of the binary block codes that we consider will be linear. A *binary linear block code* is a set of n-tuples of elements of the binary finite field $\mathbb{F}_2 = \{0, 1\}$ that form a vector space over the field $\mathbb{F}_2$. As we will see in a moment, this means simply that $\mathcal{C}$ must have the group property under n-tuple addition.

We therefore begin by studying the algebraic structure of the binary finite field $\mathbb{F}_2 = \{0, 1\}$ and of vector spaces over $\mathbb{F}_2$. In later chapters we will study codes over general finite fields.

In general, a field $\mathbb{F}$ is a set of elements with two operations, addition and multiplication, which satisfy the usual rules of ordinary arithmetic (*i.e.,* commutativity, associativity, distributivity). A field contains an additive identity 0 such that $a + 0 = a$ for all field elements $a \in \mathbb{F}$, and every field element a has an additive inverse $-a$ such that $a + (-a) = 0$. A field contains a multiplicative identity 1 such that $a \cdot 1 = a$ for all field elements $a \in \mathbb{F}$, and every nonzero field element a has a multiplicative inverse a^{-1} such that $a \cdot a^{-1} = 1$.

The binary field $\mathbb{F}_2$ (sometimes called a Galois field, and denoted by GF(2)) is the finite field with only two elements, namely 0 and 1, which may be thought of as representatives of the even and odd integers, modulo 2. Its addition and multiplication tables are given by the rules of mod 2 (even/odd) arithmetic, with 0 acting as the additive identity and 1 as the multiplicative identity:

$$\begin{array}{ll} 0+0=0 & 0\cdot 0=0 \\ 0+1=1 & 0\cdot 1=0 \\ 1+0=1 & 1\cdot 0=0 \\ 1+1=0 & 1\cdot 1=1 \end{array}$$

In fact these rules are determined by the general properties of 0 and 1 in any field. Notice that the additive inverse of 1 is 1, so $-a = a$ for both field elements.

In general, a vector space V over a field $\mathbb{F}$ is a set of vectors v including 0 such that addition of vectors and multiplication by scalars in $\mathbb{F}$ is well defined, and such that various other vector space axioms are satisfied.

For a vector space over $\mathbb{F}_2$, multiplication by scalars is trivially well defined, since $0v = 0$ and $1v = v$ are automatically in V. Therefore all that really needs to be checked is additive closure, or the *group property* of V under vector addition; *i.e.,* for all $v, v' \in V$, $v + v'$ is in V. Finally, every vector is its own additive inverse, $-v = v$, since

$$v + v = 1v + 1v = (1 + 1)v = 0v = 0.$$

In summary, over a binary field, *subtraction is the same as addition.*

A vector space over $\mathbb{F}_2$ is called a *binary vector space.* The set $(\mathbb{F}_2)^n$ of all binary n-tuples $\mathbf{v} = (v_1, \ldots, v_n)$ under componentwise binary addition is an elementary example of a binary vector space. Here we consider only binary vector spaces which are subspaces $\mathcal{C} \subseteq (\mathbb{F}_2)^n$, which are called *binary linear block codes* of length n.

If $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_k\}$ is a set of vectors in a binary vector space V, then the set $C(G)$ of all binary linear combinations

$$C(G) = \{\sum_j a_j \mathbf{g}_j, a_j \in \mathbb{F}_2, 1 \leq j \leq k\}$$

is a subspace of V, since $C(G)$ evidently has the group property. The set G is called linearly independent if these 2^k binary linear combinations are all distinct, so that the size of $C(G)$ is $|C(G)| = 2^k$. A set G of linearly independent vectors such that $C(G) = V$ is called a *basis* for V, and the elements $\{\mathbf{g}_j, 1 \le j \le k\}$ of the basis are called *generators*. The set $G = \{\mathbf{g}_1, \ldots, \mathbf{g}_k\}$ may be arranged as a $k \times n$ matrix over $\mathbb{F}_2$, called a *generator matrix* for $C(G)$.

The dimension of a binary vector space V is the number k of generators in any basis for V. As with any vector space, the dimension k and a basis G for V may be found by the following greedy algorithm:

Initialization:	set $k = 0$ and $G = \emptyset$ (the empty set);
Do loop:	if $C(G) = V$ we are done, and $\dim V = k$;
	otherwise, increase k by 1 and take any $\mathbf{v} \in V - C(G)$ as $\mathbf{g}_k$.

Thus the size of V is always $|V| = 2^k$ for some integer $k = \dim V$; conversely, $\dim V = \log_2 |V|$.

An *(n, k) binary linear code* $\mathcal{C}$ is any subspace of the vector space $(\mathbb{F}_2)^n$ with dimension k, or equivalently size 2^k. In other words, an (n, k) binary linear code is any set of 2^k binary n-tuples including $\mathbf{0}$ that has the group property under componentwise binary addition.

Example 2 (simple binary linear codes). The (n, n) binary linear code is the set $(\mathbb{F}_2)^n$ of all binary n-tuples, sometimes called the *universe code* of length n. The $(n, 0)$ binary linear code is $\{\mathbf{0}\}$, the set containing only the all-zero n-tuple, sometimes called the *trivial code* of length n. The code consisting of $\mathbf{0}$ and the all-one n-tuple $\mathbf{1}$ is an $(n, 1)$ binary linear code, called the *repetition code* of length n. The code consisting of all n-tuples with an even number of ones is an $(n, n-1)$ binary linear code, called the even-weight or *single-parity-check* (SPC) code of length n. □

6.2.1 The Hamming metric

The geometry of $(\mathbb{F}_2)^n$ is defined by the *Hamming metric*:

$$w_H(\mathbf{x}) = \text{number of ones in } \mathbf{x}.$$

The Hamming metric satisfies the axioms of a metric:

(a) Strict positivity: $w_H(\mathbf{x}) \ge 0$, with equality if and only if $\mathbf{x} = \mathbf{0}$;

(b) Symmetry: $w_H(-\mathbf{x}) = w_H(\mathbf{x})$ (since $-\mathbf{x} = \mathbf{x}$);

(c) Triangle inequality: $w_H(\mathbf{x} + \mathbf{y}) \le w_H(\mathbf{x}) + w_H(\mathbf{y})$.

Therefore the *Hamming distance*,

$$d_H(\mathbf{x}, \mathbf{y}) = w_H(\mathbf{x} - \mathbf{y}) = w_H(\mathbf{x} + \mathbf{y}),$$

may be used to define $(\mathbb{F}_2)^n$ as a metric space, called a *Hamming space*.

We now show that the group property of a binary linear block code $\mathcal{C}$ leads to a remarkable symmetry in the distance distributions from each of the codewords of $\mathcal{C}$ to all other codewords.

Let $\mathbf{x} \in \mathcal{C}$ be a given codeword of $\mathcal{C}$, and consider the set $\{\mathbf{x} + \mathbf{y} \mid \mathbf{y} \in \mathcal{C}\} = \mathbf{x} + \mathcal{C}$ as $\mathbf{y}$ runs through the codewords in $\mathcal{C}$. By the group property of $\mathcal{C}$, $\mathbf{x} + \mathbf{y}$ must be a codeword in $\mathcal{C}$.

Moreover, since $\mathbf{x}+\mathbf{y} = \mathbf{x}+\mathbf{y}'$ if and only if $\mathbf{y} = \mathbf{y}'$, all of these codewords must be distinct. But since the size of the set $\mathbf{x}+\mathcal{C}$ is $|\mathcal{C}|$, this implies that $\mathbf{x}+\mathcal{C} = \mathcal{C}$; *i.e.,* $\mathbf{x}+\mathbf{y}$ runs through all codewords in $\mathcal{C}$ as $\mathbf{y}$ runs through $\mathcal{C}$. Since $d_H(\mathbf{x},\mathbf{y}) = w_H(\mathbf{x}+\mathbf{y})$, this implies the following symmetry:

Theorem 6.1 (Distance invariance) *The set of Hamming distances $d_H(\mathbf{x},\mathbf{y})$ from any codeword $\mathbf{x} \in \mathcal{C}$ to all codewords $\mathbf{y} \in \mathcal{C}$ is independent of $\mathbf{x}$, and is equal to the set of distances from $\mathbf{0} \in \mathcal{C}$, namely the set of Hamming weights $w_H(\mathbf{y})$ of all codewords $\mathbf{y} \in \mathcal{C}$.* □

An (n,k) binary linear block code $\mathcal{C}$ is said to have *minimum Hamming distance* d, and is denoted as an (n,k,d) code, if

$$d = \min_{\mathbf{x}\neq\mathbf{y}\in\mathcal{C}} d_H(\mathbf{x},\mathbf{y}).$$

Theorem 6.1 then has the immediate corollary:

Corollary 6.2 (Minimum distance = minimum nonzero weight) *The minimum Hamming distance of $\mathcal{C}$ is equal to the minimum Hamming weight of any nonzero codeword of $\mathcal{C}$. More generally, the number of codewords $\mathbf{y} \in \mathcal{C}$ at distance d from any codeword $\mathbf{x} \in \mathcal{C}$ is equal to the number N_d of weight-d codewords in $\mathcal{C}$, independent of $\mathbf{x}$.* □

Example 2 (cont.) The (n,n) universe code has minimum Hamming distance $d = 1$, and the number of words at distance 1 from any codeword is $N_1 = n$. The $(n,n-1)$ SPC code has minimum weight and distance $d = 2$, and $N_2 = n(n-1)/2$. The $(n,1)$ repetition code has $d = n$ and $N_n = 1$. By convention, the trivial $(n,0)$ code $\{\mathbf{0}\}$ is said to have $d = \infty$. □

6.2.2 Inner products and orthogonality

A symmetric, bilinear *inner product* on the vector space $(\mathbb{F}_2)^n$ is defined by the $\mathbb{F}_2$-valued dot product

$$\langle \mathbf{x},\mathbf{y}\rangle = \mathbf{x}\cdot\mathbf{y} = \mathbf{x}\mathbf{y}^T = \sum_i x_i y_i,$$

where n-tuples are regarded as row vectors and "T" denotes "transpose." Two vectors are said to be *orthogonal* if $\langle \mathbf{x},\mathbf{y}\rangle = 0$.

However, this $\mathbb{F}_2$ inner product does not have a property analogous to strict positivity: $\langle \mathbf{x},\mathbf{x}\rangle = 0$ does not imply that $\mathbf{x} = \mathbf{0}$, but only that $\mathbf{x}$ has an even number of ones. Thus it is perfectly possible for a nonzero vector to be orthogonal to itself. Hence $\langle \mathbf{x},\mathbf{x}\rangle$ does not have a key property of the Euclidean squared norm and cannot be used to define a metric space analogous to Euclidean space. The Hamming geometry of $(\mathbb{F}_2)^n$ is very different from Euclidean geometry.

In particular, the projection theorem does not hold, and it is therefore not possible in general to find an orthogonal basis G for a binary vector space $\mathcal{C}$.

Example 3. The $(3,2)$ SPC code consists of the four 3-tuples $\mathcal{C} = \{000, 011, 101, 110\}$. Any two nonzero codewords form a basis for $\mathcal{C}$, but no two such codewords are orthogonal. □

The orthogonal code (*dual code*) $\mathcal{C}^\perp$ to an (n,k) code $\mathcal{C}$ is defined as the set of all n-tuples that are orthogonal to all elements of $\mathcal{C}$:

$$\mathcal{C}^\perp = \{\mathbf{y} \in (\mathbb{F}_2)^n \mid \langle \mathbf{x},\mathbf{y}\rangle = 0 \text{ for all } \mathbf{x} \in \mathcal{C}\}.$$

Here are some elementary facts about $\mathcal{C}^\perp$:

(a) $\mathcal{C}^\perp$ is an $(n, n-k)$ binary linear code, and thus has a basis H of size $n-k$.[1]

(b) If G is a basis for $\mathcal{C}$, then a set H of $n-k$ linearly independent n-tuples in $\mathcal{C}^\perp$ is a basis for $\mathcal{C}^\perp$ if and only if every vector in H is orthogonal to every vector in G.

(c) $(\mathcal{C}^\perp)^\perp = \mathcal{C}$.

A basis G for $\mathcal{C}$ consists of k linearly independent n-tuples in $\mathcal{C}$, and is usually written as a $k \times n$ *generator matrix* G of rank k. The code $\mathcal{C}$ then consists of all binary linear combinations $\mathcal{C} = \{\mathbf{a}G, \mathbf{a} \in (\mathbb{F}_2)^k\}$. A basis H for $\mathcal{C}^\perp$ consists of $n-k$ linearly independent n-tuples in $\mathcal{C}^\perp$, and is usually written as an $(n-k) \times n$ matrix H; then $\mathcal{C}^\perp = \{\mathbf{b}H, \mathbf{b} \in (\mathbb{F}_2)^{n-k}\}$. According to property (b) above, $\mathcal{C}$ and $\mathcal{C}^\perp$ are dual codes if and only if their generator matrices satisfy $GH^T = 0$. The transpose H^T of a generator matrix H for $\mathcal{C}^\perp$ is called a *parity-check matrix* for $\mathcal{C}$; it has the property that a vector $\mathbf{x} \in (\mathbb{F}_2)^n$ is in $\mathcal{C}$ if and only if $\mathbf{x}H^T = \mathbf{0}$, since $\mathbf{x}$ is in the dual code to $\mathcal{C}^\perp$ if and only if it is orthogonal to all generators of $\mathcal{C}^\perp$.

Example 2 (cont.; duals of simple codes). In general, the (n, n) universe code and the $(n, 0)$ trivial code are dual codes. The $(n, 1)$ repetition code and the $(n, n-1)$ SPC code are dual codes. Note that the $(2, 1)$ code $\{00, 11\}$ is both a repetition code and an SPC code, and is its own dual; such a code is called *self-dual*. (Self-duality cannot occur in real or complex vector spaces.) □

6.3 Euclidean-space images of binary linear block codes

In this section we derive the principal parameters of a binary signal constellation $s(\mathcal{C})$ from the parameters of the binary linear block code $\mathcal{C}$ on which it is based, namely the parameters (n, k, d) and the number N_d of weight-d codewords in $\mathcal{C}$.

The dimension of $s(\mathcal{C})$ is $N = n$, and its size is $M = 2^k$. It thus supports k bits per block. The bit rate (nominal spectral efficiency) is $\rho = 2k/n$ b/2D. Since $k \leq n$, $\rho \leq 2$ b/2D, and we are in the power-limited regime.

Every point in $s(\mathcal{C})$ is of the form $(\pm\alpha, \pm\alpha, \ldots, \pm\alpha)$, and therefore every point has energy $n\alpha^2$; *i.e.,* the signal points all lie on an n-sphere of squared radius $n\alpha^2$. The average energy per block is thus $E(s(C)) = n\alpha^2$, and the average energy per bit is $E_b = n\alpha^2/k$.

If two codewords $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ have Hamming distance $d_H(\mathbf{x}, \mathbf{y})$, then their Euclidean images $s(\mathbf{x}), s(\mathbf{y})$ will be the same in $n - d_H(\mathbf{x}, \mathbf{y})$ places, and will differ by 2α in $d_H(\mathbf{x}, \mathbf{y})$ places, so

[1] The standard proof of this fact involves finding a *systematic* generator matrix $G = [I_k \mid P]$ for $\mathcal{C}$, where I_k is the $k \times k$ identity matrix and P is a $k \times (n-k)$ check matrix. Then $\mathcal{C} = \{(\mathbf{u}, \mathbf{u}P), \mathbf{u} \in (\mathbb{F}_2)^k\}$, where $\mathbf{u}$ is a free information k-tuple and $\mathbf{u}P$ is a check $(n-k)$-tuple. The dual code $\mathcal{C}^\perp$ is then evidently the code generated by $H = [-P^T \mid I_{n-k}]$, where P^T is the transpose of P; *i.e.,* $\mathcal{C}^\perp = \{(-\mathbf{v}P^T, \mathbf{v}), \mathbf{v} \in (\mathbb{F}_2)^{n-k}\}$, whose dimension is $n-k$.

A more elegant proof based on the fundamental theorem of group homomorphisms (which the reader is not expected to know at this point) is as follows. Let M be the $|\mathcal{C}^\perp| \times n$ matrix whose rows are the codewords of $\mathcal{C}^\perp$. Consider the homomorphism $M^T : (\mathbb{F}_2)^n \to (\mathbb{F}_2)^{|\mathcal{C}^\perp|}$ defined by $\mathbf{y} \mapsto \mathbf{y}M^T$; *i.e.,* $\mathbf{y}M^T$ is the set of inner products of an n-tuple $\mathbf{y} \in (\mathbb{F}_2)^n$ with all codewords $\mathbf{x} \in \mathcal{C}^\perp$. The kernel of this homomorphism is evidently $\mathcal{C}$. By the fundamental theorem of homomorphisms, the image of M^T (the row space of M^T) is isomorphic to the quotient space $(\mathbb{F}_2)^n/\mathcal{C}$, which is isomorphic to $(\mathbb{F}_2)^{n-k}$. Thus the column rank of M is $n-k$. But the column rank is equal to the row rank, which is the dimension of the row space $\mathcal{C}^\perp$ of M. □

their squared Euclidean distance will be[2]

$$\|s(\mathbf{x}) - s(\mathbf{y})\|^2 = 4\alpha^2 d_H(\mathbf{x}, \mathbf{y}).$$

Therefore

$$d^2_{\min}(s(\mathcal{C})) = 4\alpha^2 d_H(\mathcal{C}) = 4\alpha^2 d,$$

where $d = d_H(\mathcal{C})$ is the minimum Hamming distance of $\mathcal{C}$.

It follows that the nominal coding gain of $s(\mathcal{C})$ is

$$\gamma_c(s(C)) = \frac{d^2_{\min}(s(C))}{4E_b} = \frac{kd}{n}. \tag{6.1}$$

Thus the parameters (n, k, d) directly determine $\gamma_c(s(C))$ in this very simple way. (This gives another reason to prefer E_b/N_0 to $\mathrm{SNR}_{\mathrm{norm}}$ in the power-limited regime.)

Moreover, every vector $s(\mathbf{x}) \in s(\mathcal{C})$ has the same number of nearest neighbors $K_{\min}(s(\mathbf{x}))$, namely the number N_d of nearest neighbors to $\mathbf{x} \in \mathcal{C}$. Thus $K_{\min}(s(C)) = N_d$, and $K_b(s(C)) = N_d/k$.

Consequently the union bound estimate of $P_b(E)$ is

$$\begin{aligned} P_b(E) &\approx K_b(s(\mathcal{C}))Q^{\surd}(\gamma_c(s(\mathcal{C}))(2E_b/N_0)) \\ &= \frac{N_d}{k} Q^{\surd}\left(\frac{dk}{n} 2E_b/N_0\right). \end{aligned} \tag{6.2}$$

In summary, the parameters and performance of the binary signal constellation $s(\mathcal{C})$ may be simply determined from the parameters (n, k, d) and N_d of $\mathcal{C}$.

Exercise 1. Let $\mathcal{C}$ be an (n, k, d) binary linear code with d odd. Show that if we append an overall parity check $p = \sum_i x_i$ to each codeword $\mathbf{x}$, then we obtain an $(n + 1, k, d + 1)$ binary linear code $\mathcal{C}'$ with d even. Show that the nominal coding gain $\gamma_c(\mathcal{C}')$ is always greater than $\gamma_c(\mathcal{C})$ if $k > 1$. Conclude that we can focus primarily on linear codes with d even. □

Exercise 2. Show that if $\mathcal{C}$ is a binary linear block code, then in every coordinate position either all codeword components are 0 or half are 0 and half are 1. Show that a coordinate in which all codeword components are 0 may be deleted ("punctured") without any loss in performance, but with savings in energy and in dimension. Show that if $\mathcal{C}$ has no such all-zero coordinates, then $s(\mathcal{C})$ has zero mean: $\mathbf{m}(s(\mathcal{C})) = \mathbf{0}$. □

6.4 Reed-Muller codes

The Reed-Muller (RM) codes are an infinite family of binary linear codes that were among the first to be discovered (1954). For block lengths $n \leq 32$, they are the best codes known with minimum distances d equal to powers of 2. For greater block lengths, they are not in general the best codes known, but in terms of performance *vs.* decoding complexity they are still quite good, since they admit relatively simple ML decoding algorithms.

[2]Moreover, the Eudlidean-space inner product of $s(\mathbf{x})$ and $s(\mathbf{y})$ is

$$\langle s(\mathbf{x}), s(\mathbf{y})\rangle = (n - d_H(\mathbf{x}, \mathbf{y}))\alpha^2 + d_H(\mathbf{x}, \mathbf{y})(-\alpha^2) = (n - 2d_H(\mathbf{x}, \mathbf{y}))\alpha^2.$$

Therefore $s(\mathbf{x})$ and $s(\mathbf{y})$ are orthogonal if and only if $d_H(\mathbf{x}, \mathbf{y}) = n/2$. Also, $s(\mathbf{x})$ and $s(\mathbf{y})$ are antipodal ($s(\mathbf{x}) = -s(\mathbf{y})$) if and only if $d_H(\mathbf{x}, \mathbf{y}) = n$.

For any integers $m \geq 0$ and $0 \leq r \leq m$, there exists an RM code, denoted by $\mathrm{RM}(r, m)$, that has length $n = 2^m$ and minimum Hamming distance $d = 2^{m-r}$, $0 \leq r \leq m$.

For $r = m$, $\mathrm{RM}(m, m)$ is defined as the universe $(2^m, 2^m, 1)$ code. It is helpful also to define RM codes for $r = -1$ by $\mathrm{RM}(-1, m) = (2^m, 0, \infty)$, the trivial code of length 2^m. Thus for $m = 0$, the two RM codes of length 1 are the $(1, 1, 1)$ universe code $\mathrm{RM}(0, 0)$ and the $(1, 0, \infty)$ trivial code $\mathrm{RM}(-1, 0)$.

The remaining RM codes for $m \geq 1$ and $0 \leq r < m$ may be constructed from these elementary codes by the following length-doubling construction, called the $|u|u+v|$ construction (originally due to Plotkin). $\mathrm{RM}(r, m)$ is constructed from $\mathrm{RM}(r-1, m-1)$ and $\mathrm{RM}(r, m-1)$ as

$$\mathrm{RM}(r, m) = \{(\mathbf{u}, \mathbf{u}+\mathbf{v}) \mid \mathbf{u} \in \mathrm{RM}(r, m-1), \mathbf{v} \in \mathrm{RM}(r-1, m-1)\}. \tag{6.3}$$

From this construction, it is easy to prove the following facts by recursion:

(a) $\mathrm{RM}(r, m)$ is a binary linear block code with length $n = 2^m$ and dimension

$$k(r, m) = k(r, m-1) + k(r-1, m-1).$$

(b) The codes are nested, in the sense that $\mathrm{RM}(r-1, m) \subseteq \mathrm{RM}(r, m)$.

(c) The minimum distance of $\mathrm{RM}(r, m)$ is $d = 2^{m-r}$ if $r \geq 0$ (if $r = -1$, then $d = \infty$).

We verify that these assertions hold for $\mathrm{RM}(0, 0)$ and $\mathrm{RM}(-1, 0)$.

For $m \geq 1$, the linearity and length of $\mathrm{RM}(r, m)$ are obvious from the construction. The dimension (size) follows from the fact that $(\mathbf{u}, \mathbf{u}+\mathbf{v}) = \mathbf{0}$ if and only if $\mathbf{u} = \mathbf{v} = \mathbf{0}$.

Exercise 5 below shows that the recursion for $k(r, m)$ leads to the explicit formula

$$k(r, m) = \sum_{0 \leq j \leq r} \binom{m}{j}, \tag{6.4}$$

where $\binom{m}{j}$ denotes the combinatorial coefficient $\frac{m!}{j!(m-j)!}$.

The nesting property for m follows from the nesting property for $m-1$.

Finally, we verify that the minimum nonzero weight of $\mathrm{RM}(r, m)$ is 2^{m-r} as follows:

(a) if $\mathbf{u} = \mathbf{0}$, then $w_H((\mathbf{0}, \mathbf{v})) = w_H(\mathbf{v}) \geq 2^{m-r}$ if $\mathbf{v} \neq \mathbf{0}$, since $\mathbf{v} \in \mathrm{RM}(r-1, m-1)$.

(b) if $\mathbf{u} + \mathbf{v} = \mathbf{0}$, then $\mathbf{u} = \mathbf{v} \in \mathrm{RM}(r-1, m-1)$ and $w_H((\mathbf{v}, \mathbf{0})) \geq 2^{m-r}$ if $\mathbf{v} \neq \mathbf{0}$.

(c) if $\mathbf{u} \neq \mathbf{0}$ and $\mathbf{u}+\mathbf{v} \neq \mathbf{0}$, then both $\mathbf{u}$ and $\mathbf{u}+\mathbf{v}$ are in $\mathrm{RM}(r, m-1)$ (since $\mathrm{RM}(r-1, m-1)$ is a subcode of $\mathrm{RM}(r, m-1)$), so

$$w_H((\mathbf{u}, \mathbf{u}+\mathbf{v})) = w_H(\mathbf{u}) + w_H(\mathbf{u}+\mathbf{v}) \geq 2 \cdot 2^{m-r-1} = 2^{m-r}.$$

Equality clearly holds for $(\mathbf{0}, \mathbf{v})$, $(\mathbf{v}, \mathbf{0})$ or $(\mathbf{u}, \mathbf{u})$ if we choose $\mathbf{v}$ or $\mathbf{u}$ as a minimum-weight codeword from their respective codes.

The $|u|u+v|$ construction suggests the following tableau of RM codes:

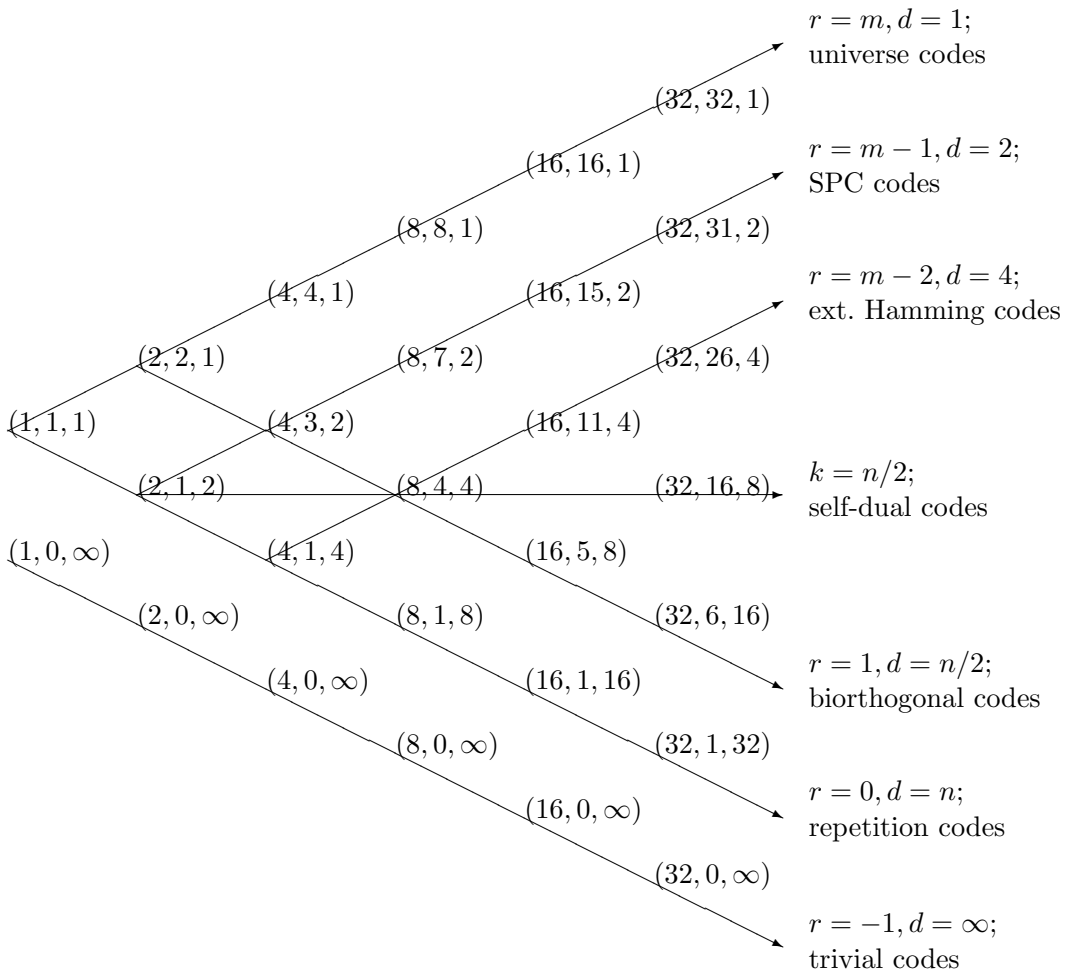


Figure 2. Tableau of Reed-Muller codes.

In this tableau each RM code lies halfway between the two codes of half the length that are used to construct it in the $|u|u+v|$ construction, from which we can immediately deduce its dimension k.

Exercise 3. Compute the parameters (k, d) of the RM codes of lengths $n = 64$ and 128. □

There is a known closed-form formula for the number N_d of codewords of minimum weight $d = 2^{m-r}$ in $\mathrm{RM}(r, m)$:

$$N_d = 2^r \prod_{0 \le i \le m-r-1} \frac{2^{m-i} - 1}{2^{m-r-i} - 1}. \tag{6.5}$$

Example 4. The number of weight-8 words in the $(32, 16, 8)$ code $\mathrm{RM}(2, 5)$ is

$$N_8 = 4\frac{31 \cdot 15 \cdot 7}{7 \cdot 3 \cdot 1} = 620.$$

The nominal coding gain of $\mathrm{RM}(2, 5)$ is $\gamma_c(\mathcal{C}) = 4$ (6.02 dB); however, since $K_b = N_8/k = 38.75$, the effective coding gain by our rule of thumb is only about $\gamma_{\mathrm{eff}}(\mathcal{C}) \approx 5.0$ dB. □

The codes with $r = m - 1$ are *single-parity-check (SPC) codes* with $d = 2$. These codes have nominal coding gain $2(k/n)$, which goes to 2 (3.01 dB) as $n \to \infty$; however, since $N_d = 2^m(2^m - 1)/2$, we have $K_b = 2^{m-1} \to \infty$, which ultimately limits the effective coding gain.

The codes with $r = m - 2$ are *extended Hamming (EH) codes* with $d = 4$. These codes have nominal coding gain $4(k/n)$, which goes to 4 (6.02 dB) as $n \to \infty$; however, since $N_d = 2^m(2^m - 1)(2^m - 2)/24$, we again have $K_b \to \infty$.

Exercise 4 (optimizing SPC and EH codes). Using the rule of thumb that a factor of two increase in K_b costs 0.2 dB in effective coding gain, find the value of n for which an $(n, n-1, 2)$ SPC code has maximum effective coding gain, and compute this maximum in dB. Similarly, find m such that a $(2^m, 2^m - m - 1, 4)$ extended Hamming code has maximum effective coding gain, using $N_d = 2^m(2^m - 1)(2^m - 2)/24$, and compute this maximum in dB. □

The codes with $r = 1$ (*first-order Reed-Muller codes*) are interesting, because as shown in Exercise 5 they generate biorthogonal signal sets of dimension $n = 2^m$ and size 2^{m+1}, with nominal coding gain $(m + 1)/2 \to \infty$. It is known that as $n \to \infty$ this sequence of codes can achieve arbitrarily small $\Pr(E)$ for any E_b/N_0 greater than the ultimate Shannon limit, namely $E_b/N_0 > \ln 2$ (-1.59 dB).

Exercise 5 (biorthogonal codes). We have shown that the first-order Reed-Muller codes $\mathrm{RM}(1, m)$ have parameters $(2^m, m+1, 2^{m-1})$, and that the $(2^m, 1, 2^m)$ repetition code $\mathrm{RM}(0, m)$ is a subcode.

(a) Show that $\mathrm{RM}(1, m)$ has one word of weight 0, one word of weight 2^m, and $2^{m+1} - 2$ words of weight 2^{m-1}. [Hint: first show that the $\mathrm{RM}(1, m)$ code consists of 2^m complementary codeword pairs $\{\mathbf{x}, \mathbf{x} + \mathbf{1}\}$.]

(b) Show that the Euclidean image of an $\mathrm{RM}(1, m)$ code is an $M = 2^{m+1}$ biorthogonal signal set. [Hint: compute all inner products between code vectors.]

(c) Show that the code $\mathcal{C}'$ consisting of all words in $\mathrm{RM}(1, m)$ with a 0 in any given coordinate position is a $(2^m, m, 2^{m-1})$ binary linear code, and that its Euclidean image is an $M = 2^m$ orthogonal signal set. [Same hint as in part (a).]

(d) Show that the code $\mathcal{C}''$ consisting of the code words of $\mathcal{C}'$ with the given coordinate deleted ("punctured") is a binary linear $(2^m - 1, m, 2^{m-1})$ code, and that its Euclidean image is an $M = 2^m$ simplex signal set. [Hint: use Exercise 7 of Chapter 5.] □

In Exercise 2 of Chapter 1, it was shown how a 2^m-orthogonal signal set $\mathcal{A}$ can be constructed as the image of a $2^m \times 2^m$ binary Hadamard matrix. The corresponding 2^{m+1}-biorthogonal signal set $\pm\mathcal{A}$ is identical to that constructed above from the $(2^m, m + 1, 2^{m-1})$ first-order RM code.

The code dual to $\mathrm{RM}(r, m)$ is $\mathrm{RM}(m - r - 1, m)$; this can be shown by recursion from the facts that the $(1, 1)$ and $(1, 0)$ codes are duals and that by bilinearity

$$\langle(\mathbf{u}, \mathbf{u} + \mathbf{v}), (\mathbf{u}', \mathbf{u}' + \mathbf{v}')\rangle = \langle \mathbf{u}, \mathbf{u}'\rangle + \langle \mathbf{u} + \mathbf{v}, \mathbf{u}' + \mathbf{v}'\rangle = \langle \mathbf{u}, \mathbf{v}'\rangle + \langle \mathbf{v}, \mathbf{u}'\rangle + \langle \mathbf{v}, \mathbf{v}'\rangle,$$

since $\langle \mathbf{u}, \mathbf{u}'\rangle + \langle \mathbf{u}, \mathbf{u}'\rangle = 0$. In particular, this confirms that the repetition and SPC codes are duals, and shows that the biorthogonal and extended Hamming codes are duals.

This also shows that RM codes with $k/n = 1/2$ are self-dual. The nominal coding gain of a rate-1/2 RM code of length 2^m (m odd) is $2^{(m-1)/2}$, which goes to infinity as $m \to \infty$. It seems likely that as $n \to \infty$ this sequence of codes can achieve arbitrarily small $\Pr(E)$ for any E_b/N_0 greater than the Shannon limit for $\rho = 1$ b/2D, namely $E_b/N_0 > 1$ (0 dB).

Exercise 6 (generator matrices for RM codes). Let square $2^m \times 2^m$ matrices U_m, $m \geq 1$, be specified recursively as follows. The matrix U_1 is the 2×2 matrix

$$U_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

The matrix U_m is the $2^m \times 2^m$ matrix

$$U_m = \begin{bmatrix} U_{m-1} & 0 \\ U_{m-1} & U_{m-1} \end{bmatrix}.$$

(In other words, U_m is the m-fold tensor product of U_1 with itself.)

(a) Show that RM(r, m) is generated by the rows of U_m of Hamming weight 2^{m-r} or greater. [Hint: observe that this holds for $m = 1$, and prove by recursion using the $|u|u+v|$ construction.] For example, give a generator matrix for the $(8, 4, 4)$ RM code.

(b) Show that the number of rows of U_m of weight 2^{m-r} is $\binom{m}{r}$. [Hint: use the fact that $\binom{m}{r}$ is the coefficient of z^{m-r} in the integer polynomial $(1 + z)^m$.]

(c) Conclude that the dimension of RM(r, m) is $k(r, m) = \sum_{0 \leq j \leq r} \binom{m}{j}$. □

6.4.1 Effective coding gains of RM codes

We provide below a table of the nominal spectral efficiency ρ, nominal coding gain γ_c, number of nearest neighbors N_d, error coefficient per bit K_b, and estimated effective coding gain γ_{eff} at $P_b(E) \approx 10^{-5}$ for various Reed-Muller codes, so that the student can consider these codes as components in system design exercises.

In later lectures, we will consider trellis representations and trellis decoding of RM codes. We give here two complexity parameters of the minimal trellises for these codes: the state complexity s (the binary logarithm of the maximum number of states in a minimal trellis), and the branch complexity t (the binary logarithm of the maximum number of branches per section in a minimal trellis). The latter parameter gives a more accurate estimate of decoding complexity.

code	ρ	γ_c	(dB)	N_d	K_b	γ_{eff} (dB)	s	t
(8,7,2)	1.75	7/4	2.43	28	4	2.0	1	2
(8,4,4)	1.00	2	3.01	14	4	2.6	2	3
(16,15,2)	1.88	15/8	2.73	120	8	2.1	1	2
(16,11,4)	1.38	11/4	4.39	140	13	3.7	3	5
(16, 5,8)	0.63	5/2	3.98	30	6	3.5	3	4
(32,31, 2)	1.94	31/16	2.87	496	16	2.1	1	2
(32,26, 4)	1.63	13/4	5.12	1240	48	4.0	4	7
(32,16, 8)	1.00	4	6.02	620	39	4.9	6	9
(32, 6,16)	0.37	3	4.77	62	10	4.2	4	5
(64,63, 2)	1.97	63/32	2.94	2016	32	1.9	1	2
(64,57, 4)	1.78	57/16	5.52	10416	183	4.0	5	9
(64,42, 8)	1.31	21/4	7.20	11160	266	5.6	10	16
(64,22,16)	0.69	11/2	7.40	2604	118	6.0	10	14
(64, 7,32)	0.22	7/2	5.44	126	18	4.6	5	6

Table 1. Parameters of RM codes with lengths $n \leq 64$.

6.5 Decoding of binary block codes

In this section we will first show that with binary codes MD decoding reduces to "maximum-reliability decoding." We will then discuss the penalty incurred by making hard decisions and then performing classical error-correction. We show how the penalty may be partially mitigated by using erasures, or rather completely by using generalized minimum distance (GMD) decoding.

6.5.1 Maximum-reliability decoding

All of our performance estimates assume minimum-distance (MD) decoding. In other words, given a received sequence $\mathbf{r} \in \mathbb{R}^n$, the receiver must find the signal $s(\mathbf{x})$ for $\mathbf{x} \in \mathcal{C}$ such that the squared distance $\|\mathbf{r} - s(\mathbf{x})\|^2$ is minimum. We will show that in the case of binary codes, MD decoding reduces to maximum-reliability (MR) decoding.

Since $\|s(\mathbf{x})\|^2 = n\alpha^2$ is independent of $\mathbf{x}$ with binary constellations $s(\mathcal{C})$, MD decoding is equivalent to *maximum-inner-product decoding*: find the signal $s(\mathbf{x})$ for $\mathbf{x} \in \mathcal{C}$ such that the inner product

$$\langle \mathbf{r}, s(\mathbf{x}) \rangle = \sum_k r_k s(x_k)$$

is maximum. Since $s(x_k) = (-1)^{x_k}\alpha$, the inner product may be expressed as

$$\langle \mathbf{r}, s(\mathbf{x}) \rangle = \alpha \sum_k r_k (-1)^{x_k} = \alpha \sum_k |r_k| \operatorname{sgn}(r_k)(-1)^{x_k}$$

The sign $\operatorname{sgn}(r_k) \in \{\pm 1\}$ is often regarded as a "hard decision" based on r_k, indicating which of the two possible signals $\{\pm\alpha\}$ is more likely in that coordinate without taking into account the remaining coordinates. The magnitude $|r_k|$ may be viewed as the reliability of the hard decision. This rule may thus be expressed as: find the codeword $\mathbf{x} \in \mathcal{C}$ that maximizes the reliability

$$r(\mathbf{x} \mid \mathbf{r}) = \sum_k |r_k| (-1)^{e(x_k, r_k)},$$

where the "error" $e(x_k, r_k)$ is 0 if the signs of $s(x_k)$ and r_k agree, or 1 if they disagree. We call this rule *maximum-reliability decoding*.

Any of these optimum decision rules is easy to implement for small constellations $s(\mathcal{C})$. However, without special tricks they require at least one computation for every codeword $\mathbf{x} \in \mathcal{C}$, and therefore become impractical when the number 2^k of codewords becomes large. Finding simpler decoding algorithms that give a good tradeoff of performance *vs.* complexity, perhaps only for special classes of codes, has therefore been the major theme of practical coding research.

For example, the Wagner decoding rule, the earliest "soft-decision" decoding algorithm (*circa* 1955), is an optimum decoding rule for the special class of $(n, n-1, 2)$ SPC codes that requires many fewer than 2^{n-1} computations.

Exercise 7 ("Wagner decoding"). Let $\mathcal{C}$ be an $(n, n-1, 2)$ SPC code. The Wagner decoding rule is as follows. Make hard decisions on every symbol r_k, and check whether the resulting binary word is in $\mathcal{C}$. If so, accept it. If not, change the hard decision in the symbol r_k for which the reliability metric $|r_k|$ is minimum. Show that the Wagner decoding rule is an optimum decoding rule for SPC codes. [Hint: show that the Wagner rule finds the codeword $\mathbf{x} \in \mathcal{C}$ that maximizes $r(\mathbf{x} \mid \mathbf{r})$.] □

6.5.2 Hard decisions and error-correction

Early work on decoding of binary block codes assumed hard decisions on every symbol, yielding a hard-decision n-tuple $\mathbf{y} \in (\mathbb{F}_2)^n$. The main decoding step is then to find the codeword $\mathbf{x} \in \mathcal{C}$ that is closest to $\mathbf{y}$ in Hamming space. This is called *error-correction*.

If $\mathcal{C}$ is a linear (n, k, d) code, then, since the Hamming metric is a true metric, no error can occur when a codeword $\mathbf{x}$ is sent unless the number of hard decision errors $t = d_H(\mathbf{x}, \mathbf{y})$ is at least as great as half the minimum Hamming distance, $t \geq d/2$. For many classes of binary block codes, efficient algebraic error-correction algorithms exist that are guaranteed to decode correctly provided that $2t < d$. This is called *bounded-distance error-correction*.

Example 5 (Hamming codes). The first binary error-correction codes were the Hamming codes (mentioned in Shannon's original paper). A Hamming code $\mathcal{C}$ is a $(2^m - 1, 2^m - m - 1, 3)$ code that may be found by puncturing a $(2^m, 2^m - m - 1, 4)$ extended Hamming RM$(m - 2, m)$ code in any coordinate. Its dual $\mathcal{C}^\perp$ is a $(2^m - 1, m, 2^{m-1})$ code whose Euclidean image is a 2^m-simplex constellation. For example, the simplest Hamming code is the $(3, 1, 3)$ repetition code; its dual is the $(3, 2, 2)$ SPC code, whose image is the 4-simplex constellation of Figure 1.

The generator matrix of $\mathcal{C}^\perp$ is an $m \times (2^m - 1)$ matrix H whose $2^m - 1$ columns must run through the set of all nonzero binary m-tuples in some order (else $\mathcal{C}$ would not be guaranteed to correct any single error; see next paragraph).

Since $d = 3$, a Hamming code should be able to correct any single error. A simple method for doing so is to compute the "syndrome"

$$\mathbf{y}H^T = (\mathbf{x} + \mathbf{e})H^T = \mathbf{e}H^T,$$

where $\mathbf{e} = \mathbf{x} + \mathbf{y}$. If $\mathbf{y}H^T = \mathbf{0}$, then $\mathbf{y} \in \mathcal{C}$ and $\mathbf{y}$ is assumed to be correct. If $\mathbf{y}H^T \neq \mathbf{0}$, then the syndrome $\mathbf{y}H^T$ is equal to one of the rows in H^T, and a single error is assumed to have occurred in the corresponding position. Thus it is always possible to change any $\mathbf{y} \in (\mathbb{F}_2)^n$ into a codeword by changing at most one bit.

This implies that the 2^{n-m} "Hamming spheres" of radius 1 and size 2^m centered on the 2^{n-m} codewords $\mathbf{x}$, which consist of $\mathbf{x}$ and the $n = 2^m - 1$ n-tuples $\mathbf{y}$ within Hamming distance 1 of $\mathbf{x}$, form an exhaustive partition of the set of 2^n n-tuples that comprise Hamming n-space $(\mathbb{F}_2)^n$.

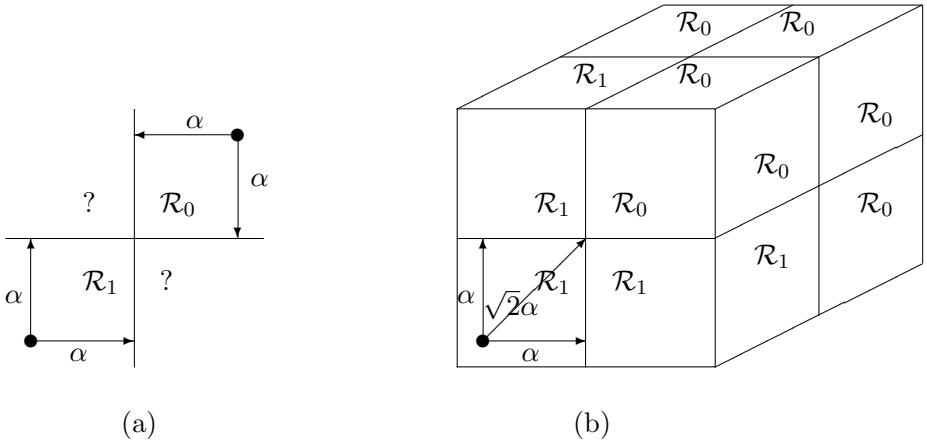
In summary, Hamming codes form a "perfect" Hamming sphere-packing of $(\mathbb{F}_2)^n$, and have a simple single-error-correction algorithm. □

We now show that even if an error-correcting decoder does optimal MD decoding in Hamming space, there is a loss in coding gain of the order of 3 dB relative to MD Euclidean-space decoding.

Assume an (n, k, d) binary linear code $\mathcal{C}$ with d odd (the situation is worse when d is even). Let $\mathbf{x}$ be the transmitted codeword; then there is at least one codeword at Hamming distance d from $\mathbf{x}$, and thus at least one real n-tuple in $s(\mathcal{C})$ at Euclidean distance $4\alpha^2 d$ from $s(\mathbf{x})$. For any $\varepsilon > 0$, a hard-decision decoding error will occur if the noise exceeds $\alpha + \varepsilon$ in any $(d + 1)/2$ of the places in which that word differs from $\mathbf{x}$. Thus with hard decisions the minimum squared distance to the decision boundary in Euclidean space is $\alpha^2(d + 1)/2$. (For d even, it is $\alpha^2 d/2$.)

On the other hand, with "soft decisions" (reliability weights) and MD decoding, the minimum squared distance to any decision boundary in Euclidean space is $\alpha^2 d$. To the accuracy of the union bound estimate, the argument of the $Q\sqrt{}$ function thus decreases with hard-decision decoding by a factor of $(d + 1)/2d$, or approximately $1/2$ (-3 dB) when d is large. (When d is even, this factor is exactly $1/2$.)

Example 6 (Hard and soft decoding of antipodal codes). Let $\mathcal{C}$ be the $(2, 1, 2)$ binary code; then the two signal points in $s(\mathcal{C})$ are antipodal, as shown in Figure 3(a) below. With hard decisions, real 2-space $\mathbb{R}^2$ is partitioned into four quadrants, which must then be assigned to one or the other of the two signal points. Of course, two of the quadrants are assigned to the signal points that they contain. However, no matter how the other two quadrants are assigned, there will be at least one decision boundary at squared distance α^2 from a signal point, whereas with MD decoding the decision boundary is at distance $2\alpha^2$ from both signal points. The loss in the error exponent of $P_b(E)$ is therefore a factor of 2 (3 dB).



(a) (b)

Figure 3. Decision regions in $\mathbb{R}^n$ with hard decisions. (a) $(2, 1, 2)$ code; (b) $(3, 1, 3)$ code.

Similarly, if $\mathcal{C}$ is the $(3, 1, 3)$ code, then $\mathbb{R}^3$ is partitioned by hard decisions into 8 octants, as shown in Figure 3(b). In this case (the simplest example of a Hamming code), it is clear how best to assign four octants to each signal point. The squared distance from each signal point to the nearest decision boundary is now $2\alpha^2$, compared to $3\alpha^2$ with "soft decisions" and MD decoding in Euclidean space, for a loss of $2/3$ (1.76 dB) in the error exponent. □

6.5.3 Erasure-and-error-correction

A decoding method halfway between hard-decision and "soft-decision" (reliability-based) techniques involves the use of "erasures." With this method, the first step of the receiver is to map each received signal r_k into one of three values, say $\{0, 1, ?\}$, where for some threshold T,

$$\begin{aligned} r_k \to \ & 0 \quad \text{if} \quad r_k > T; \\ r_k \to \ & 1 \quad \text{if} \quad r_k < -T; \\ r_k \to \ & ? \quad \text{if} \quad -T \le r_k \le T. \end{aligned}$$

The decoder subsequently tries to map the ternary-valued n-tuple into the closest codeword $\mathbf{x} \in \mathcal{C}$ in Hamming space, where the erased positions are ignored in measuring Hamming distance.

If there are s erased positions, then the minimum distance between codewords is at least $d - s$ in the unerased positions, so correct decoding is guaranteed if the number t of errors in the unerased positions satisfies $t < (d-s)/2$, or equivalently if $2t + s < d$. For many classes of binary block codes, efficient algebraic erasure-and-error-correcting algorithms exist that are guaranteed to decode correctly if $2t + s < d$. This is called *bounded-distance erasure-and-error-correction.*

Erasure-and-error-correction may be viewed as a form of MR decoding in which all reliabilities $|r_k|$ are made equal in the unerased positions, and are set to 0 in the erased positions.

The ternary-valued output allows a closer approximation to the optimum decision regions in Euclidean space than with hard decisions, and therefore reduces the loss. With an optimized threshold T, the loss is typically only about half as much (in dB).

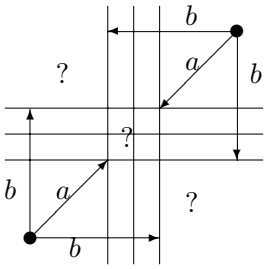


Figure 4. Decision regions with hard decisions and erasures for the $(2, 1, 2)$ code.

Example 6 (cont.). Figure 4 shows the 9 decision regions for the $(2, 1, 2)$ code that result from hard decisions and/or erasures on each symbol. Three of the resulting regions are ambiguous. The minimum squared distances to these regions are

$$\begin{aligned} a^2 &= 2(\alpha - T)^2 \\ b^2 &= (\alpha + T)^2. \end{aligned}$$

To maximize the minimum of a^2 and b^2, we make $a^2 = b^2$ by choosing $T = \frac{\sqrt{2}-1}{\sqrt{2}+1}\alpha$, which yields

$$a^2 = b^2 \frac{8}{(\sqrt{2}+1)^2}\alpha^2 = 1.372\alpha^2.$$

This is about 1.38 dB better than the squared Euclidean distance α^2 achieved with hard decisions only, but is still 1.63 dB worse than the $2\alpha^2$ achieved with MD decoding. □

Exercise 8 (Optimum threshold T). Let $\mathcal{C}$ be a binary code with minimum distance d, and let received symbols be mapped into hard decisions or erasures as above. Show that:

(a) For any integers t and s such that $2t + s \geq d$ and for any decoding rule, there exists some pattern of t errors and s erasures that will cause a decoding error;

(b) The minimum squared distance from any signal point to its decoding decision boundary is equal to at least $\min_{2t+s\geq d} \{s(\alpha - T)^2 + t(\alpha + T)^2\}$;

(c) The value of T that maximizes this minimum squared distance is $T = \frac{\sqrt{2}-1}{\sqrt{2}+1}\alpha$, in which case the minimum squared distance is equal to $\frac{4}{(\sqrt{2}+1)^2}\alpha^2 d = 0.686\ \alpha^2 d$. Again, this is a loss of 1.63 dB relative to the squared distance $\alpha^2 d$ that is achieved with MD decoding. □

6.5.4 Generalized minimum distance decoding

A further step in this direction that achieves almost the same performance as MD decoding, to the accuracy of the union bound estimate, yet still permits algebraic decoding algorithms, is generalized minimum distance (GMD) decoding.

In GMD decoding, the decoder keeps both the hard decision $\mathrm{sgn}(r_k)$ and the reliability $|r_k|$ of each received symbol, and orders them in order of their reliability.

The GMD decoder then performs a series of erasure-and-error decoding trials in which the $s = d-1, d-3, \ldots$ least reliable symbols are erased. (The intermediate trials are not necessary because if $d-s$ is even and $2t < d-s$, then also $2t < d-s-1$, so the trial with one additional erasure will find the same codeword.) The number of such trials is $d/2$ if d is even, or $(d+1)/2$ if d is odd; *i.e.,* the number of trials needed is $\lceil d/2 \rceil$.

Each trial may produce a candidate codeword. The set of $\lceil d/2 \rceil$ trials may thus produce up to $\lceil d/2 \rceil$ distinct candidate codewords. These words may finally be compared according to their reliability $\mathbf{r}(\mathbf{x} \mid \mathbf{r})$ (or any equivalent optimum metric), and the best candidate chosen.

Example 7. For an $(n, n-1, 2)$ SPC code, GMD decoding performs just one trial with the least reliable symbol erased; the resulting candidate codeword is the unique codeword that agrees with all unerased symbols. Therefore in this case the GMD decoding rule is equivalent to the Wagner decoding rule (Exercise 7), which implies that it is optimum. □

It can be shown that no error can occur with a GMD decoder provided that the squared norm $||\mathbf{n}||^2$ of the noise vector is less than $\alpha^2 d$; *i.e.,* the squared distance from any signal point to its decision boundary is $\alpha^2 d$, just as for MD decoding. Thus there is no loss in coding gain or error exponent compared to MD decoding.

It has been shown that for the most important classes of algebraic block codes, GMD decoding can be performed with little more complexity than ordinary hard-decision or erasures-and-errors decoding. Furthermore, it has been shown that not only is the error exponent of GMD decoding equal to that of optimum MD decoding, but also the error coefficient and thus the union bound estimate are the same, provided that GMD decoding is augmented to include a d-erasure-correction trial (a purely algebraic solution of the $n-k$ linear parity-check equations for the d unknown erased symbols).

However, GMD decoding is a bounded-distance decoding algorithm, so its decision regions are like spheres of squared radius $\alpha^2 d$ that lie within the MD decision regions $\mathcal{R}_j$. For this reason GMD decoding is inferior to MD decoding, typically improving over erasure-and-error-correction by 1 dB or less. GMD decoding has rarely been used in practice.

6.5.5 Summary

In conclusion, hard decisions allow the use of efficient algebraic decoding algorithms, but incur a significant SNR penalty, of the order of 3 dB. By using erasures, about half of this penalty can be avoided. With GMD decoding, efficient algebraic decoding algorithms can in principle be used with no loss in performance, at least as estimated by the the union bound estimate.