Chapter 2

Sphere Packing and Shannon's Theorem

In the first section we discuss the basics of block coding on the m-ary symmetric channel. In the second section we see how the geometry of the codespace can be used to make coding judgements. This leads to the third section where we present some information theory and Shannon's basic Channel Coding Theorem.

2.1 Basics of block coding on the mSC

block code

Let A be any finite set. A *block code* or *code*, for short, will be any nonempty subset of the set A^n of n-tuples of elements from A. The number $n = n(C)$ is the *length* of the code, and the set A^n is the *codespace*. The number of members in C is the *size* and is denoted $|C|$. If C has length n and size $|C|$, we say that C is an $(n, |C|)$ *code*.

length
codespace
size
$(n, |C|)$ code

The members of the codespace will be referred to as *words*, those belonging to C being *codewords*. The set A is then the *alphabet*.

words
codewords
alphabet

If the alphabet A has m elements, then C is said to be an m-*ary code*. In the special case $|A|$=2 we say C is a *binary* code and usually take $A = \{0, 1\}$ or $A = \{-1, +1\}$. When $|A|$=3 we say C is a *ternary* code and usually take $A = \{0, 1, 2\}$ or $A = \{-1, 0, +1\}$. Examples of both binary and ternary codes appeared in Section 1.3.

m-ary code
binary
ternary

For a discrete memoryless channel, the Reasonable Assumption says that a pattern of errors that involves a small number of symbol errors should be more likely than any particular pattern that involves a large number of symbol errors. As mentioned, the assumption is really a statement about design.

On an mSC(p) the probability $p(\mathbf{y}|\mathbf{x})$ that $\mathbf{x}$ is transmitted and $\mathbf{y}$ is received is equal to $p^d q^{n-d}$, where d is the number of places in which $\mathbf{x}$ and $\mathbf{y}$ differ. Therefore

$$\text{Prob}(\mathbf{y} \,|\, \mathbf{x}) = q^n (p/q)^d \,,$$

a decreasing function of d provided $q > p$. Therefore the Reasonable Assumption is realized by the mSC(p) subject to

$$q = 1 - (m-1)p > p$$

or, equivalently,

$$1/m > p\ .$$

We interpret this restriction as the sensible design criterion that after a symbol is transmitted it should be more likely for it to be received as the correct symbol than to be received as any particular incorrect symbol.

EXAMPLES.

(i) Assume we are transmitting using the the binary Hamming code of Section 1.3.3 on BSC(.01). Comparing the received word 0011111 with the two codewords 0001111 and 1011010 we see that

$$p(0011111|0001111) = q^6p^1 \approx .009414801\,,$$

while

$$p(0011111|1011010) = q^4p^3 \approx .000000961\,;$$

therefore we prefer to decode 0011111 to 0001111. Even this event is highly unlikely, compared to

$$p(0001111|0001111) = q^7 \approx .932065348\,.$$

(ii) If $m = 5$ with $A = \{0,1,2,3,4\}^6$ and $p = .05 < 1/5 = .2$, then $q = 1 - 4(.05) = .8$; and we have

$$p(011234|011234) = q^6 = .262144$$

and

$$p(011222|011234) = q^4p^2 = .001024\,.$$

For $\mathbf{x}, \mathbf{y} \in A^n$, we define

$$\mathrm{d_H}(\mathbf{x},\mathbf{y}) = \text{the number of places in which } \mathbf{x} \text{ and } \mathbf{y} \text{ differ.}$$

Hamming distance

This number is the *Hamming distance* between $\mathbf{x}$ and $\mathbf{y}$. The Hamming distance is a genuine metric on the codespace A^n. It is clear that it is symmetric and that $\mathrm{d_H}(\mathbf{x},\mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$. The Hamming distance $\mathrm{d_H}(\mathbf{x},\mathbf{y})$ should be thought of as the number of errors required to change $\mathbf{x}$ into $\mathbf{y}$ (or, equally well, to change $\mathbf{y}$ into $\mathbf{x}$).

EXAMPLE.

$$\mathrm{d_H}(0011111, 0001111) = 1\,;$$
$$\mathrm{d_H}(0011111, 1011010) = 3\,;$$
$$\mathrm{d_H}(011234, 011222) = 2\,.$$

(2.1.1) PROBLEM. *Prove the triangle inequality for the Hamming distance:*

$$\mathrm{d_H}(\mathbf{x},\mathbf{y}) + \mathrm{d_H}(\mathbf{y},\mathbf{z}) \geq \mathrm{d_H}(\mathbf{x},\mathbf{z})\,.$$

The arguments above show that, for an $m\mathrm{SC}(p)$ with $p < 1/m$, maximum likelihood decoding becomes:

> **Minimum Distance Decoding** — When $\mathbf{y}$ is received, we must decode to a codeword $\mathbf{x}$ that minimizes the Hamming distance $\mathrm{d_H}(\mathbf{x}, \mathbf{y})$.

We abbreviate *minimum distance decoding* as **MDD**. In this context, incomplete decoding is incomplete minimum distance decoding **IMDD**:

minimum distance decoding
MDD
IMDD

> **Incomplete Minimum Distance Decoding** — When $\mathbf{y}$ is received, we must decode either to a codeword $\mathbf{x}$ that minimizes the Hamming distance $\mathrm{d_H}(\mathbf{x}, \mathbf{y})$ or to the "error detected" symbol $\boldsymbol{\infty}$.

(2.1.2) PROBLEM. *Prove that, for an $m\mathrm{SC}(p)$ with $p = 1/m$, every complete decoding algorithm is an* **MLD** *algorithm.*

(2.1.3) PROBLEM. *Give a definition of what might be called* maximum distance decoding*,* **MxDD***; and prove that* **MxDD** *algorithms are* **MLD** *algorithms for an $m\mathrm{SC}(p)$ with $p > 1/m$.*

In A^n, the *sphere*[1] of radius ρ centered at $\mathbf{x}$ is

sphere

$$\mathrm{S}_\rho(\mathbf{x}) = \{\, \mathbf{y} \in A^n \mid \mathrm{d_H}(\mathbf{x}, \mathbf{y}) \le \rho \,\}.$$

Thus the sphere of radius ρ around $\mathbf{x}$ is composed of those $\mathbf{y}$ that might be received if at most ρ symbol errors were introduced to the transmitted codeword $\mathbf{x}$.

The volume of a sphere of radius ρ is independent of the location of its center.

(2.1.4) PROBLEM. *Prove that in A^n with $|A| = m$, a sphere of radius e contains* $\displaystyle\sum_{i=0}^{e} \binom{n}{i} (m-1)^i$ *words.*

For example, a sphere of radius 2 in $\{0,1\}^{90}$ has volume

$$1 + \binom{90}{1} + \binom{90}{2} = 1 + 90 + 4005 = 4096 = 2^{12}$$

corresponding to a center, 90 possible locations for a single error, and $\binom{90}{2}$ possibilities for a double error. A sphere of radius 2 in $\{0,1,2\}^8$ has volume

$$1 + \binom{8}{1}(3-1)^1 + \binom{8}{2}(3-1)^2 = 1 + 16 + 112 = 129\ .$$

For each nonnegative real number ρ we define a decoding algorithm $\mathbf{SS}_\rho$ for A^n called *sphere shrinking*.

$\mathbf{SS}_\rho$
sphere shrinking

[1] Mathematicians would prefer to use the term 'ball' here in place of 'sphere', but we stick with the traditional coding terminology.

Radius ρ Sphere Shrinking — If $\mathbf{y}$ is received, we decode to the codeword $\mathbf{x}$ if $\mathbf{x}$ is the unique codeword in $\mathrm{S}_\rho(\mathbf{y})$, otherwise we declare a decoding default.

Thus $\mathbf{SS}_\rho$ shrinks the sphere of radius ρ around each codeword to its center, throwing out words that lie in more than one such sphere.

The various distance determined algorithms are completely described in terms of the geometry of the codespace and the code rather than by the specific channel characteristics. In particular they no longer depend upon the transition parameter p of an $m\mathrm{SC}(p)$ being used. For **IMDD** algorithms **A** and **B**, if $\mathcal{P}_C(\mathbf{A}) \leq \mathcal{P}_C(\mathbf{B})$ for some $m\mathrm{SC}(p)$ with $p < 1/m$, then $\mathcal{P}_C(\mathbf{A}) \leq \mathcal{P}_C(\mathbf{B})$ will be true for all $m\mathrm{SC}(p)$ with $p < 1/m$. The **IMDD** algorithms are (incomplete) maximum likelihood algorithms on every $m\mathrm{SC}(p)$ with $p \leq 1/m$, but this observation now becomes largely motivational.

EXAMPLE. Consider the specific case of a binary repetition code of length 26. Since the first two possibilities are not algorithms but classes of algorithms there are choices available.

w = number of 1's	0	$1 \leq w \leq 11$	$= 12$	$= 13$	$= 14$	$15 \leq w \leq 25$	26
IMDD	$\mathbf{0/\infty}$	$\mathbf{0/\infty}$	$\mathbf{0/\infty}$	$\mathbf{0/1/\infty}$	$\mathbf{1/\infty}$	$\mathbf{1/\infty}$	$\mathbf{1/\infty}$
MDD	**0**	**0**	**0**	**0/1**	**1**	**1**	**1**
$\mathbf{SS}_{12}$	**0**	**0**	**0**	∞	**1**	**1**	**1**
$\mathbf{SS}_{11}$	**0**	**0**	∞	∞	∞	**1**	**1**
$\mathbf{SS}_0$	**0**	∞	∞	∞	∞	∞	**1**

Here **0** and **1** denote, respectively, the 26-tuple of all 0's and all 1's. In the fourth case, we have less error correcting power. On the other hand we are less likely to have a decoder error, since 15 or more symbol errors must occur before a decoder error results. The final case corrects no errors, but detects nontrivial errors except in the extreme case where all symbols are received incorrectly, thereby turning the transmitted codeword into the other codeword.

The algorithm $\mathbf{SS}_0$ used in the example is the usual error detection algorithm: when $\mathbf{y}$ is received, decode to $\mathbf{y}$ if it is a codeword and otherwise decode to $\boldsymbol{\infty}$, declaring that an error has been detected.

2.2 Sphere packing

minimum distance

The code C in A^n has *minimum distance* $\mathrm{d}_{\min}(C)$ equal to the minimum of $\mathrm{d}_{\mathrm{H}}(\mathbf{x}, \mathbf{y})$, as $\mathbf{x}$ and $\mathbf{y}$ vary over all distinct pairs of codewords from C. (This leaves some confusion over $\mathrm{d}_{\min}(C)$ for a length n code C with only one word. It may be convenient to think of it as any number larger than n.) An (n, M) code

(n, M, d) code

with minimum distance d will sometimes be referred to as an (n, M, d) *code*.

EXAMPLE. The minimum distance of the repetition code of length n is clearly n. For the parity check code any single error produces a word of

odd parity, so the minimum distance is 2. The length 27 generalized Reed-Solomon code of Example 1.3.6 was shown to have minimum distance 21.

Laborious checking reveals that the $[7, 4]$ Hamming code has minimum distance 3, and its extension has minimum distance 4. The $[4, 2]$ ternary Hamming code also has minimum distance 3. We shall see later how to find the minimum distance of these codes easily.

(2.2.1) LEMMA. *The following are equivalent for the code C in A^n for an integer $e \leq n$:*

(1) *under $\mathbf{SS}_e$ any occurrence of e or fewer symbol errors will always be successfully corrected;*

(2) *for all distinct $\mathbf{x}, \mathbf{y}$ in C, we have $\mathrm{S}_e(\mathbf{x}) \cap \mathrm{S}_e(\mathbf{y}) = \emptyset$;*

(3) *the minimum distance of C, $\mathrm{d}_{\min}(C)$, is at least $2e + 1$.*

PROOF. Assume (1), and let $\mathbf{z} \in \mathrm{S}_e(\mathbf{x})$, for some $\mathbf{x} \in C$. Then by assumption $\mathbf{z}$ is decoded to $\mathbf{x}$ by $\mathbf{SS}_e$. Therefore there is no $\mathbf{y} \in C$ with $\mathbf{y} \neq \mathbf{x}$ and $\mathbf{z} \in \mathrm{S}_e(\mathbf{y})$, giving (2).

Assume (2), and let $\mathbf{z}$ be a word that results from the introduction of at most e errors to the codeword $\mathbf{x}$. By assumption $\mathbf{z}$ is not in $\mathrm{S}_e(\mathbf{y})$ for any $\mathbf{y}$ of C other than $\mathbf{x}$. Therefore, $\mathrm{S}_e(\mathbf{z})$ contains $\mathbf{x}$ and no other codewords; so $\mathbf{z}$ is decoded to $\mathbf{x}$ by $\mathbf{SS}_e$, giving (1).

If $\mathbf{z} \in \mathrm{S}_e(\mathbf{x}) \cap \mathrm{S}_e(\mathbf{y})$, then by the triangle inequality we have $\mathrm{d_H}(\mathbf{x}, \mathbf{y}) \leq \mathrm{d_H}(\mathbf{x}, \mathbf{z}) + \mathrm{d_H}(\mathbf{z}, \mathbf{y}) \leq 2e$, so (3) implies (2).

It remains to prove that (2) implies (3). Assume $\mathrm{d}_{\min}(C) = d \leq 2e$. Choose $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ in C with $\mathrm{d_H}(\mathbf{x}, \mathbf{y}) = d$. If $d \leq e$, then $\mathbf{x} \in \mathrm{S}_e(\mathbf{x}) \cap \mathrm{S}_e(\mathbf{y})$; so we may suppose that $d > e$.

Let $i_1, \ldots, i_d \leq n$ be the coordinate positions in which $\mathbf{x}$ and $\mathbf{y}$ differ: $x_{i_j} \neq y_{i_j}$, for $j = 1, \ldots, d$. Define $\mathbf{z} = (z_1, \ldots, z_n)$ by $z_k = y_k$ if $k \notin \{i_1, \ldots, i_e\}$ and $z_k = x_k$ if $k \in \{i_1, \ldots, i_e\}$. Then $\mathrm{d_H}(\mathbf{y}, \mathbf{z}) = e$ and $\mathrm{d_H}(\mathbf{x}, \mathbf{z}) = d - e \leq e$. Thus $\mathbf{z} \in \mathrm{S}_e(\mathbf{x}) \cap \mathrm{S}_e(\mathbf{y})$. Therefore (2) implies (3). □

A code C that satisfies the three equivalent properties of Lemma 2.2.1 is called an *e-error-correcting code*. The lemma reveals one of the most pleasing aspects of coding theory by identifying concepts from three distinct and important areas. The first property is algorithmic, the second is geometric, and the third is linear algebraic. We can readily switch from one point of view to another in search of appropriate insight and methodology as the context requires.

e-error-correcting code

(2.2.2) PROBLEM. *Explain why the error detecting algorithm $\mathbf{SS}_0$ correctly detects all patterns of fewer than $\mathrm{d}_{\min}$ symbol errors.*

(2.2.3) PROBLEM. *Let $f \geq e$. Prove that the following are equivalent for the code C in A^n:*

(1) *under $\mathbf{SS}_e$ any occurrence of e or fewer symbol errors will always be successfully corrected and no occurrence of f or fewer symbol errors will cause a decoder error;*

(2) *for all distinct $\mathbf{x}, \mathbf{y}$ in C, we have $\mathrm{S}_f(\mathbf{x}) \cap \mathrm{S}_e(\mathbf{y}) = \emptyset$;*

(3) *the minimum distance of C, $\mathrm{d}_{\min}(C)$, is at least $e + f + 1$.*

*A code C that satisfies the three equivalent properties of the problem is called an e-*error-correcting, *f-*error-detecting *code.*

e-error-correcting, f-error-detecting

(2.2.4) PROBLEM. *Consider an erasure channel, that is, a channel that erases certain symbols and leaves a '?' in their place but otherwise changes nothing. Explain why, using a code with minimum distance d on this channel, we can correct all patterns of up to $d-1$ symbol erasures. (In certain computer systems this observation is used to protect against hard disk crashes.)*

By Lemma 2.2.1, if we want to construct an e-error-correcting code, we must be careful to choose as codewords the centers of radius e spheres that are pairwise disjoint. We can think of this as packing spheres of radius e into the large box that is the entire codespace. From this point of view, it is clear that we will not be able to fit in any number of spheres whose total volume exceeds the volume of the box. This proves:

(2.2.5) THEOREM. (SPHERE PACKING CONDITION.) *If C is an e-error-correcting code in A^n, then*

$$|C| \cdot |\mathrm{S}_e(*)| \leq |A^n| . \qquad \square$$

Combined with Problem 2.1.4, this gives:

(2.2.6) COROLLARY. (SPHERE PACKING BOUND; HAMMING BOUND.) *If C is a m-ary e-error-correcting code of length n, then*

$$|C| \leq m^n \Big/ \sum_{i=0}^{e} \binom{n}{i} (m-1)^i . \qquad \square$$

perfect e-error-correcting code

A code C that meets the sphere packing bound with equality is called a *perfect e-error-correcting code*. Equivalently, C is a perfect e-error-correcting code if and only if $\mathbf{SS}_e$ is a **MDD** algorithm. As examples we have the binary repetition codes of odd length. The $[7,4]$ Hamming code is a perfect 1-error-correcting code, as we shall see in Section 4.1.

(2.2.7) THEOREM. (GILBERT-VARSHAMOV BOUND.) *There exists an m-ary e-error-correcting code C of length n such that*

$$|C| \geq m^n \Big/ \sum_{i=0}^{2e} \binom{n}{i} (m-1)^i .$$

PROOF. The proof is by a "greedy algorithm" construction. Let the codespace be A^n. At Step 1 we begin with the code $C_1 = \{\mathbf{x}_1\}$, for any word $\mathbf{x}_1$. Then, for $i \geq 2$, we have:

Step i. Set $S_i = \bigcup_{j=1}^{i-1} \mathrm{S}_{d-1}(\mathbf{x}_j)$.
If $S_i = A^n$, halt.
Otherwise choose a vector $\mathbf{x}_i$ in $A^n - S_i$;
set $C_i = C_{i-1} \cup \{\mathbf{x}_i\}$;
go to Step $i+1$.

At Step i, the code C_i has cardinality i and is designed to have minimum distance at least d. (As long as $d \leq n$ we can choose $\mathbf{x}_2$ at distance d from $\mathbf{x}_1$; so each C_i, for $i \geq 1$ has minimum distance exactly d.)

How soon does the algorithm halt? We argue as we did in proving the sphere packing condition. The set $S_i = \bigcup_{j=1}^{i-1} \mathrm{S}_{d-1}(\mathbf{x}_j)$ will certainly be smaller than A^n if the spheres around the words of C_{i-1} have total volume less than the volume of the entire space A^n; that is, if

$$|C_{i-1}| \cdot |\mathrm{S}_{d-1}(*)| < |A^n| \,.$$

Therefore when the algorithm halts, this inequality must be false. Now Problem 2.1.4 gives the bound. □

A sharper version of the Gilbert-Varshamov bound exists, but the asymptotic result of the next section is unaffected.

EXAMPLES.

(i) Consider a binary 2-error-correcting code of length 90. By the Sphere Packing Bound it has size at most

$$\frac{2^{90}}{|\mathrm{S}_2(*)|} = \frac{2^{90}}{2^{12}} = 2^{78} \,.$$

If a code existed meeting this bound, it would be perfect.

By the Gilbert-Varshamov Bound, in $\{0,1\}^{90}$ there exists a code C with minimum distance 5, which therefore corrects 2 errors, and having

$$|C| \geq \frac{2^{90}}{|\mathrm{S}_4(*)|} = \frac{2^{90}}{2676766} \approx 4.62 \times 10^{20} \,.$$

As $2^{78} \approx 3.02 \times 10^{23}$, there is a factor of roughly 650 separating the lower and upper bounds.

(ii) Consider a ternary 2-error-correcting code of length 8. By the Sphere Packing Bound it has size bounded above by

$$\frac{3^8}{|\mathrm{S}_2(*)|} = \frac{6561}{129} \approx 50.86 \,.$$

Therefore it has size at most $\lfloor 50.86 \rfloor = 50$. On the other hand, the Gilbert-Varshamov Bound guarantees only a code C of size bounded below by

$$|C| \geq \frac{6561}{|\mathrm{S}_4(*)|} = \frac{6561}{1697} \approx 3.87 \,,$$

that is, of size at least $\lceil 3.87 \rceil = 4$! Later we shall construct an appropriate C of size 27. (This is in fact the largest possible.)

(2.2.8) PROBLEM. *In each of the following cases decide whether or not there exists a 1-error-correcting code C with the given size in the codespace V. If there is such a code, give an example (except in (d), where an example is not required but a justification is). If there is not such a code, prove it.*

(a) $V = \{0,1\}^5$ *and* $|C| = 6$;
(b) $V = \{0,1\}^6$ *and* $|C| = 9$;
(c) $V = \{0,1,2\}^4$ *and* $|C| = 9$.
(d) $V = \{0,1,2\}^8$ *and* $|C| = 51$.

(2.2.9) PROBLEM. *In each of the following cases decide whether or not there exists a 2-error-correcting code C with the given size in the codespace V. If there is such a code, give an example. If there is not such a code, prove it.*

(*a*) $V = \{0,1\}^8$ *and* $|C| = 4$*;*

(*b*) $V = \{0,1\}^8$ *and* $|C| = 5$.

2.3 Shannon's theorem and the code region

The present section is devoted to information theory rather than coding theory and will not contain complete proofs. The goal of coding theory is to live up to the promises of information theory. Here we shall see of what our dreams are made.

Our immediate goal is to quantify the Fundamental Problem. We need to evaluate information content and error performance.

dimension

We first consider information content. The m-ary code C has *dimension* $k(C) = \log_m(|C|)$. The integer $k = \lceil k(C) \rceil$ is the smallest such that each message for C can be assigned its own individual message k-tuple from the m-ary alphabet A. Therefore we can think of the dimension as the number of codeword symbols that are carrying message rather than redundancy. (Thus

redundancy

the number $n - k$ is sometimes called the *redundancy* of C.) A repetition code has n symbols, only one of which carries the message; so its dimension is 1. For a length n parity check code, $n - 1$ of the symbols are message symbols; and so the code has dimension $n - 1$. The $[7, 4]$ Hamming code has dimension 4 as does its $[8, 4]$ extension, since both contain $2^4 = 16$ codewords. Our definition of dimension does not apply to our real Reed-Solomon example 1.3.6 since its alphabet is infinite, but it is clear what its dimension should be. Its 27 positions are determined by 7 free parameters, so the code should have dimension 7.

The dimension of a code is a deceptive gauge of information content. For instance, a binary code C of length 4 with 4 codewords and dimension $\log_2(4) = 2$ actually contains more information than a second code D of length 8 with 8 codewords and dimension $\log_2(8) = 3$. Indeed the code C can be used to produce $16 = 4 \times 4$ different valid code sequences of length 8 (a pair of codewords) while the code D only offers 8 valid sequences of length 8. Here and elsewhere, the proper measure of information content should be the fraction of the code symbols that carries information rather than redundancy. In this example $2/4 = 1/2$ of the symbols of C carry information while for D only $3/8$ of the symbols carry information, a fraction smaller than that for C.

The fraction of a repetition codeword that is information is $1/n$, and for a parity check code the fraction is $(n-1)/n$. In general, we define the *normalized*

rate

dimension or *rate* $\kappa(C)$ of the m-ary code C of length n by

$$\kappa(C) = k(C)/n = n^{-1} \log_m(|C|) .$$

The repetition code thus has rate $1/n$, and the parity check code rate $(n-1)/n$. The $[7, 4]$ Hamming code has rate $4/7$, and its extension rate $4/8 = 1/2$. The $[4, 2]$ ternary Hamming code has rate $2/4 = 1/2$. Our definition of rate does

not apply to the real Reed-Solomon example of 1.3.6, but arguing as before we see that it has "rate" 7/27. The rate is the normalized dimension of the code, in that it indicates the fraction of each code coordinate that is information as opposed to redundancy.

The rate $\kappa(C)$ provides us with a good measure of the information content of C. Next we wish to measure the error handling ability of the code. One possible gauge is $\mathcal{P}_C$, the error expectation of C; but in general this will be hard to calculate. We can estimate $\mathcal{P}_C$, for an $m\mathrm{SC}(p)$ with small p, by making use of the obvious relationship $\mathcal{P}_C \leq \mathcal{P}_C(\mathbf{SS}_\rho)$ for any ρ. If $e = \lfloor (d-1)/2 \rfloor$, then C is an e-error-correcting code; and certainly $\mathcal{P}_C \leq \mathcal{P}_C(\mathbf{SS}_e)$, a probability that is easy to calculate. Indeed $\mathbf{SS}_e$ corrects all possible patterns of at most e symbol errors but does not correct any other errors; so

$$\mathcal{P}_C(\mathbf{SS}_e) = 1 - \sum_{i=0}^{e} \binom{n}{i} (m-1)^i p^i q^{n-i} .$$

The difference between $\mathcal{P}_C$ and $\mathcal{P}_C(\mathbf{SS}_e)$ will be given by further terms $p^j q^{n-j}$ with j larger than e. For small p, these new terms will be relatively small.

Shannon's theorem guarantees the existence of large families of codes for which $\mathcal{P}_C$ is small. The previous paragraph suggests that to prove this efficiently we might look for codes with arbitrarily small $\mathcal{P}_C(\mathbf{SS}_{(\mathrm{d}_{\min}-1)/2})$, and in a sense we do. However, it can be proven that decoding up to minimum distance alone is not good enough to prove Shannon's Theorem. (Think of the 'Birthday Paradox'.) Instead we note that a received block of large length n is most likely to contain sn symbol errors where $s = p(m-1)$ is the probability of symbol error. Therefore in proving Shannon's theorem we look at large numbers of codes, each of which we decode using $\mathbf{SS}_\rho$ for some radius ρ a little larger than sn.

A family $\mathcal{C}$ of codes over A is called a *Shannon family* if, for every $\epsilon > 0$, there is a code $C \in \mathcal{C}$ with $\mathcal{P}_C < \epsilon$. For a finite alphabet A, the family $\mathcal{C}$ must necessarily be infinite and so contain codes of unbounded length. Shannon family

(2.3.1) PROBLEM. *Prove that the set of all binary repetition codes of odd length is a Shannon family on $BSC(p)$ for $p < 1/2$.*

Although repetition codes give us a Shannon family, they do not respond to the Fundamental Problem by having good information content as well. Shannon proved that codes of the sort we need are out there somewhere.

(2.3.2) THEOREM. (SHANNON'S CHANNEL CODING THEOREM.) *Consider the m-ary symmetric channel $m\mathrm{SC}(p)$ with $p < 1/m$. There is a function $C_m(p)$ such that, for any $\kappa < C_m(p)$,*

$$\mathcal{C}_\kappa = \{ \text{ } m\text{-ary block codes of rate at least } \kappa \}$$

is a Shannon family. Conversely if $\kappa > C_m(p)$, then $\mathcal{C}_\kappa$ is not a Shannon family. □

The function $C_m(p)$ is the capacity function for the mSC(p) and will be discussed below.

Shannon's theorem tells us that we can communicate reliably at high rates; but, as R.J. McEliece has remarked, its lesson is deeper and more precise than this. It tells us that to make the best use of our channel we must transmit at rates near capacity and then filter out errors at the destination. Think about Lucy and Ethel wrapping chocolates. The company can maximize its total profit by increasing the conveyor belt rate and accepting a certain amount of wastage. The tricky part is figuring out how high the rate can be set before chaos ensues.

Shannon's theorem is robust in that bounding rate by the capacity function still allows transmission at high rate for most p. In the particular case $m = 2$, we have

$$C_2(p) = 1 + p\log_2(p) + q\log_2(q)\,,$$

where $p+q = 1$. Thus on a binary symmetric channel with transition probability $p = .02$ (a pretty bad channel), we have $C_2(.02) \approx .8586$. Similarly $C_2(.1) \approx .5310$, $C_2(.01) \approx .9192$, and $C_2(.001) \approx .9886$. So, for instance, if we expect bit errors .1 % of the time, then we may transmit messages that are nearly 99% information but still can be decoded with arbitrary precision. Many channels in use these days operate with p between 10^{-7} and 10^{-15}.

entropy

We define the general entropy and capacity functions before giving an idea of their origin. The m-ary *entropy* function is defined on $(0, (m-1)/m]$ by

$$H_m(x) = -x\log_m(x/(m-1)) - (1-x)\log_m(1-x),$$

capacity

where we additionally define $H_m(0) = 0$ for continuity. Notice $H_m(\frac{m-1}{m}) = 1$. Having defined entropy, we can now define the m-ary *capacity* function on $[0, 1/m]$ by

$$C_m(p) = 1 - H_m((m-1)p)\,.$$

We have $C_m(0) = 1$ and $C_m(1/m) = 0$.

We next see why entropy and capacity might play a role in coding problems. (The lemma is a consequence of Stirling's formula.)

(2.3.3) LEMMA. *For spheres in A^n with $|A| = m$ and any σ in $(0, (m-1)/m]$, we have*

$$\lim_{n\to\infty} n^{-1}\log_m(|S_{\sigma n}(*)|) = H_m(\sigma). \qquad \square$$

For a code C of sufficient length n on mSC(p) we expect sn symbol errors in a received word, so we would like to correct at least this many errors. Applying the Sphere Packing Condition 2.2.5 we have

$$|C| \cdot |S_{sn}(*)| \le m^n\,,$$

which, upon taking logarithms, is

$$\log_m(|C|) + \log_m(|S_{sn}(*)|) \le n\,.$$

We divide by n and move the second term across the inequality to find

$$\kappa(C) = n^{-1}\log_m(|C|) \leq 1 - n^{-1}\log_m(|S_{sn}(*)|)\,.$$

The righthand side approaches $1 - H_m(s) = C_m(p)$ as n goes to infinity; so, for C to be a contributing member of a Shannon family, it should have rate at most capacity. This suggests:

(2.3.4) PROPOSITION. *If $\mathcal{C}$ is a Shannon family for mSC(p) with $0 \leq p \leq 1/m$, then* $\liminf_{C\in\mathcal{C}} \kappa(C) \leq C_m(p)$. □

The proposition provides the converse in Shannon's Theorem, as we have stated it. (Our arguments do not actually prove this converse. We can not assume our spheres of radius sn to be pairwise disjoint, so the Sphere Packing Condition does not directly apply.)

We next suggest a proof of the direct part of Shannon's theorem, noticing along the way how our geometric interpretation of entropy and capacity is involved.

The outline for a proof of Shannon's theorem is short: for each $\epsilon > 0$ (and n) we choose a $\rho\,(= \rho_\epsilon(n) = sn + o(n)\,)$ for which

$$\mathrm{avg}_C\ \mathcal{P}_C(\mathbf{SS}_\rho) < \epsilon\,,$$

for all sufficiently large n, where the average is taken over all $C \subseteq A^n$ with $|C| = m^{\kappa n}$ (round up), codes of length n and rate κ. As the average is less than ϵ, there is certainly some particular code C with $\mathcal{P}_C$ less than ϵ, as required.

In carrying this out it is enough (by symmetry) to consider all C containing a fixed $\mathbf{x}$ and prove

$$\mathrm{avg}_C\ \mathcal{P}_{\mathbf{x}}(\mathbf{SS}_\rho) < \epsilon\,.$$

Two sources of incorrect decoding for transmitted $\mathbf{x}$ must be considered:

(i) $\mathbf{y}$ is received with $\mathbf{y} \notin \mathrm{S}_\rho(\mathbf{x})$;

(ii) $\mathbf{y}$ is received with $\mathbf{y} \in \mathrm{S}_\rho(\mathbf{x})$ but also $\mathbf{y} \in \mathrm{S}_\rho(\mathbf{z})$, for some $\mathbf{z} \in C$ with $\mathbf{z} \neq \mathbf{x}$.

For mistakes of the first type the binomial distribution guarantees a probability less than $\epsilon/2$ for a choice of ρ just slightly larger than $sn = p(m-1)n$, even without averaging. For our fixed $\mathbf{x}$, the average probability of an error of the second type is over-estimated by

$$m^{\kappa n}\,\frac{|\mathrm{S}_\rho(\mathbf{z})|}{m^n}\,,$$

the number of $\mathbf{z} \in C$ times the probability that an arbitrary $\mathbf{y}$ is in $\mathrm{S}_\rho(\mathbf{z})$. This average probability has logarithm

$$-n\Big((1 - n^{-1}\log_m(|\mathrm{S}_\rho(*)|)) - \kappa\Big)\,.$$

In the limit, the quantity in the parenthesis is

$$(1 - H_m(s)) - \kappa = \beta\,,$$

which is positive by hypothesis. The average then behaves like $m^{-n\beta}$. Therefore by increasing n we can also make the average probability in the second case less than $\epsilon/2$. This completes the proof sketch.

Shannon's theorem now guarantees us codes with arbitrarily small error expectation $\mathcal{P}_C$, but this number is still not a very good measure of error handling ability for the Fundamental Problem. Aside from being difficult to calculate, it is actually channel dependent, being typically a polynomial in p and $q = 1 - (m-1)p$. As we have discussed, one of the attractions of **IMDD** decoding on m-ary symmetric channels is the ability to drop channel specific parameters in favor of general characteristics of the code geometry. So perhaps rather than search for codes with small $\mathcal{P}_C$, we should be looking at codes with large minimum distance. This parameter is certainly channel independent; but, as with dimension and rate, we have to be careful to normalize the distance. While 100 might be considered a large minimum distance for a code of length 200, it might not be for a code of length 1,000,000. We instead consider the *normalized distance* of the length n code C defined as $\delta(C) = \mathrm{d}_{\min}(C)/n$.

normalized distance

As further motivation for study of the normalized distance, we return to the observation that, in a received word of decent length n, we expect $sn = p(m-1)n$ symbol errors. For correct decoding we would like

$$p(m-1)n \leq (\mathrm{d}_{\min} - 1)/2\,.$$

If we rewrite this as

$$0 < 2p(m-1) \leq (\mathrm{d}_{\min} - 1)/n < \mathrm{d}_{\min}/n = \delta\,,$$

then we see that for a family of codes with good error handling ability we attempt to bound the normalized distance δ away from 0.

The Fundamental Problem has now become:

> **The Fundamental Problem of Coding Theory** — Find practical m-ary codes C with reasonably large rate $\kappa(C)$ and reasonably large normalized distance $\delta(C)$.

What is viewed as practical will vary with the situation. For instance, we might wish to bound decoding complexity or storage required.

Shannon's theorem provides us with cold comfort. The codes are out there somewhere, but the proof by averaging gives no hint as to where we should look.[2] In the next chapter we begin our search in earnest. But first we discuss what sort of pairs $(\delta(C), \kappa(C))$ we might attain.

[2] In the last fifty years many good codes have been constructed, but only beginning in 1993—with the introduction of turbo codes, the rediscovery of *LDPC* codes, and the intense study of related codes and associated iterative decoding algorithms—did we start to see how Shannon's bound is approachable in practice in certain cases. The codes and algorithms discussed in these remain of importance.

We could graph in $[0,1]\times[0,1]$ all pairs $(\delta(C),\kappa(C))$ realized by some m-ary code C, but many of these correspond to codes that have no claim to being practical. For instance, the length 1 binary code $C=\{0,1\}$ has $(\delta(C),\kappa(C))=(1,1)$ but is certainly impractical by any yardstick. The problem is that in order for us to be confident that the number of symbol errors in a received n-tuple is close to $p(m-1)n$, the length n must be large. So rather than graph all attainable pairs $(\delta(C),\kappa(C))$, we adopt the other extreme and consider only those pairs that can be realized by codes of arbitrarily large length.

To be precise, the point $(\delta,\kappa)\in[0,1]\times[0,1]$ belongs to the m-ary *code region* if and only if there is a sequence $\{C_n\}$ of m-ary codes C_n with unbounded length n for which

code region

$$\delta=\lim_{n\to\infty}\delta(C_n) \text{ and } \kappa=\lim_{n\to\infty}\kappa(C_n)\ .$$

Equivalently, the code region is the set of all accumulation points in $[0,1]\times[0,1]$ of the graph of achievable pairs $(\delta(C),\kappa(C))$.

(2.3.5) THEOREM. (MANIN'S BOUND ON THE CODE REGION.) *There is a continuous, nonincreasing function $\kappa_m(\delta)$ on the interval $[0,1]$ such that the point (δ,κ) is in the m-ary code region if and only if*

$$0\le\kappa\le\kappa_m(\delta)\,. \qquad \square$$

Although the proof is elementary, we do not give it. However we can easily see why something like this should be true. If the point (δ,κ) is in the code region, then it seems reasonable that the code region should contain as well the points (δ',κ) , $\delta'<\delta$, corresponding to codes with the same rate but smaller distance and also the points (δ,κ'), $\kappa'<\kappa$, corresponding to codes with the same distance but smaller rate. Thus for any point (δ,κ) of the code region, the rectangle with corners $(0,0)$, $(\delta,0)$, $(0,\kappa)$, and (δ,κ) should be entirely contained within the code region. Any region with this property has its upper boundary function nonincreasing and continuous.

In our discussion of Proposition 2.3.4 we saw that $\kappa(C)\le 1-H_m(s)$ when correcting the expected sn symbol errors for a code of length n. Here sn is roughly $(d-1)/2$ and s is approximately $(d-1)/2n$. In the present context the argument preceding Proposition 2.3.4 leads to

(2.3.6) THEOREM. (ASYMPTOTIC HAMMING BOUND.) *We have*

$$\kappa_m(\delta)\le 1-H_m(\delta/2)\ . \qquad \square$$

Similarly, from the Gilbert-Varshamov bound 2.2.7 we derive:

(2.3.7) THEOREM. (ASYMPTOTIC GILBERT-VARSHAMOV BOUND.) *We have*

$$\kappa_m(\delta)\ge 1-H_m(\delta)\ . \qquad \square$$

Various improvements to the Hamming upper bound and its asymptotic version exist. We present two.

(2.3.8) THEOREM. (PLOTKIN BOUND.) *Let C be an m-ary code of length n with $\delta(C) > (m-1)/m$. Then*

$$|C| \leq \frac{\delta}{\delta - \frac{m-1}{m}} \,. \qquad \square$$

(2.3.9) COROLLARY. (ASYMPTOTIC PLOTKIN BOUND.)
(1) *$\kappa_m(\delta) = 0$ for $(m-1)/m < \delta \leq 1$.*
(2) *$\kappa_m(\delta) \leq 1 - \frac{m}{m-1}\delta$ for $0 \leq \delta \leq (m-1)/m$.* $\square$

For a fixed $\delta > (m-1)/m$, the Plotkin bound 2.3.8 says that code size is bounded by a constant. Thus as n goes to infinity, the rate goes to 0, hence (1) of the corollary. Part (2) is proven by applying the Plotkin bound not to the code C but to a related code C' with the same minimum distance but of shorter length. (The proof of part (2) of the corollary appears below in §6.1.3. The proof of the theorem is given as Problem 3.1.6.)

(2.3.10) PROBLEM. (SINGLETON BOUND.) *Let C be a code in A^n with minimum distance $d = \mathrm{d}_{\min}(C)$. Prove $|C| \leq |A|^{n-d+1}$. (HINT: For the word $\mathbf{y} \in A^{n-d+1}$, how many codewords of C can have a copy of $\mathbf{y}$ as their first $n-d+1$ entries?)*

(2.3.11) PROBLEM. (ASYMPTOTIC SINGLETON BOUND.) *Use Problem 2.3.10 to prove $\delta + \kappa_m(\delta) \leq 1$. (We remark that this is a weak form of the asymptotic Plotkin bound.)*

While the asymptotic Gilbert-Varshamov bound shows that the code region is large, the proof is essentially nonconstructive since the greedy algorithm must be used infinitely often. Most of the easily constructed families of codes give rise to code region points either on the δ-axis or the κ-axis.

(2.3.12) PROBLEM. *Prove that the family of repetition codes produces the point $(1,0)$ of the code region and the family of parity check codes produces the point $(0,1)$.*

The first case in which points in the interior of the code region were explicitly constructed was the following 1972 result of Justesen:

(2.3.13) THEOREM. *For $0 < \kappa < \frac{1}{2}$, there is a positive constant c and a sequence of binary codes $J_{\kappa,n}$ with rate at least κ and*

$$lim_{n\to\infty}\delta(J_{\kappa,n}) \geq c(1-2\kappa)\,.$$

Thus the line $\delta = c(1-2\kappa)$ is constructively within the binary code region. $\square$

Justesen also has a version of his construction that produces binary codes of larger rate. The constant c that appears in Theorem 2.3.13 is the unique solution to $H_2(c) = \frac{1}{2}$ in $[0, \frac{1}{2}]$ and is roughly $.110\,$.

While there are various improvements to the asymptotic Hamming upper bound on $\kappa_m(\delta)$ and the code region, such improvements to the asymptotic Gilbert-Varshamov lower bound are rare and difficult. Indeed for a long time

Figure 2.1: Bounds on the m-ary code region

Figure 2.2: The 49-ary code region

it was conjectured that the asymptotic Gilbert-Varshamov bound holds with equality,

$$\kappa_m(\delta) = 1 - H_m(\delta)\,.$$

This is now known to be false for infinitely many m, although not as yet for the important cases $m = 2, 3$. The smallest known counterexample is at $m = 49$.

(2.3.14) THEOREM. *The line*

$$\kappa + \delta = \frac{5}{6}$$

is within the 49*-ary code region but is not below the corresponding Gilbert-Varshamov curve*

$$\kappa = 1 - H_{49}(\delta)\,. \qquad \square$$

This theorem and much more was proven by Tsfasman, Vladut, and Zink in 1982 using difficult results from algebraic geometry in the context of a broad generalization of Reed-Solomon codes.

It should be emphasized that these results are of an asymptotic nature. As we proceed, we shall see various useful codes for which (δ, κ) is outside the code region and important families whose corresponding limit points lie on a coordinate axis $\kappa = 0$ or $\delta = 0$.