Elements of Information Theory
Thomas M. Cover, Joy A. Thomas
Copyright © 1991 John Wiley & Sons, Inc.
Print ISBN 0-471-06259-6 Online ISBN 0-471-20061-1

Chapter 8

Channel Capacity

What do we mean when we say that A communicates with B? We mean that the physical acts of A have induced a desired physical state in B. This transfer of information is a physical process and therefore is subject to the uncontrollable ambient noise and imperfections of the physical signalling process itself. The communication is successful if the receiver B and the transmitter A agree on what was sent.

In this chapter we find the maximum number of distinguishable signals for n uses of a communication channel. This number grows exponentially with n, and the exponent is known as the channel capacity. The channel capacity theorem is the central and most famous success of information theory.

The mathematical analog of a physical signalling system is shown in Fig. 8.1. Source symbols from some finite alphabet are mapped into some sequence of channel symbols, which then produces the output sequence of the channel. The output sequence is random but has a distribution that depends on the input sequence. From the output sequence, we attempt to recover the transmitted message.

Each of the possible input sequences induces a probability distribution on the output sequences. Since two different input sequences may give rise to the same output sequence, the inputs are confusable. In the next few sections, we will show that we can choose a "non-confusable" subset of input sequences so that with high probability, there is only one highly likely input that could have caused the particular output. We can then reconstruct the input sequences at the output with negligible probability of error. By mapping the source into the appropriate "widely spaced" input sequences to the channel, we can transmit a message with very low probability of error and

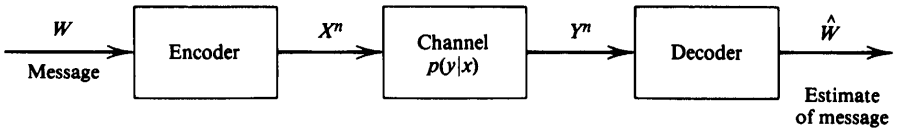


Figure 8.1. A communication system.

reconstruct the source message at the output. The maximum rate at which this can be done is called the capacity of the channel.

Definition: We define a *discrete channel* to be a system consisting of an input alphabet $\mathcal{X}$ and output alphabet $\mathcal{Y}$ and a probability transition matrix $p(y|x)$ that expresses the probability of observing the output symbol y given that we send the symbol x. The channel is said to be *memoryless* if the probability distribution of the output depends only on the input at that time and is conditionally independent of previous channel inputs or outputs.

Definition: We define the *"information" channel capacity* of a discrete memoryless channel as

$$C = \max_{p(x)} I(X; Y), \tag{8.1}$$

where the maximum is taken over all possible input distributions $p(x)$.

We shall soon give an operational definition of channel capacity as the highest rate in bits per channel use at which information can be sent with arbitrarily low probability of error. Shannon's second theorem establishes that the "information" channel capacity is equal to the "operational" channel capacity. Thus we drop the word "information" in most discussions of channel capacity.

There is a duality between the problems of data compression and data transmission. During compression, we remove all the redundancy in the data to form the most compressed version possible, whereas during data transmission, we add redundancy in a controlled fashion to combat errors in the channel. In the last section of this chapter, we show that a general communication system can be broken into two parts and that the problems of data compression and data transmission can be considered separately.

8.1 EXAMPLES OF CHANNEL CAPACITY

8.1.1 Noiseless Binary Channel

Suppose we have a channel whose the binary input is reproduced exactly at the output. This channel is illustrated in Figure 8.2. In this

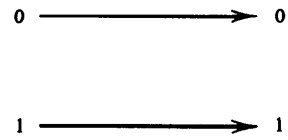


Figure 8.2. Noiseless binary channel.

case, any transmitted bit is received without error. Hence, 1 error-free bit can be transmitted per use of the channel, and the capacity is 1 bit. We can also calculate the information capacity $C = \max I(X; Y) = 1$ bit, which is achieved by using $p(x) = (\frac{1}{2}, \frac{1}{2})$.

8.1.2 Noisy Channel with Nonoverlapping Outputs

This channel has two possible outputs corresponding to each of the two inputs, as illustrated in Figure 8.3. The channel appears to be noisy, but really is not.

Even though the output of the channel is a random consequence of the input, the input can be determined from the output, and hence every transmitted bit can be recovered without error. The capacity of this channel is also 1 bit per transmission. We can also calculate the information capacity $C = \max I(X; Y) = 1$ bit, which is achieved by using $p(x) = (\frac{1}{2}, \frac{1}{2})$.

8.1.3 Noisy Typewriter

In this case, the channel input is either received unchanged at the output with probability $\frac{1}{2}$ or is transformed into the next letter with probability $\frac{1}{2}$ (Figure 8.4). If the input has 26 symbols and we use every alternate input symbol, then we can transmit 13 symbols without error with each transmission. Hence the capacity of this channel is log 13 bits per transmission. We can also calculate the information capacity $C =$

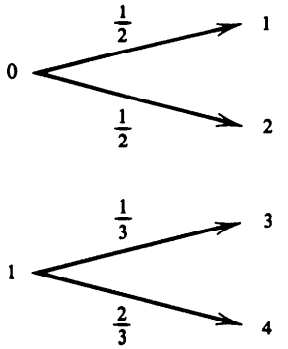


Figure 8.3. Noisy channel with nonoverlapping outputs.

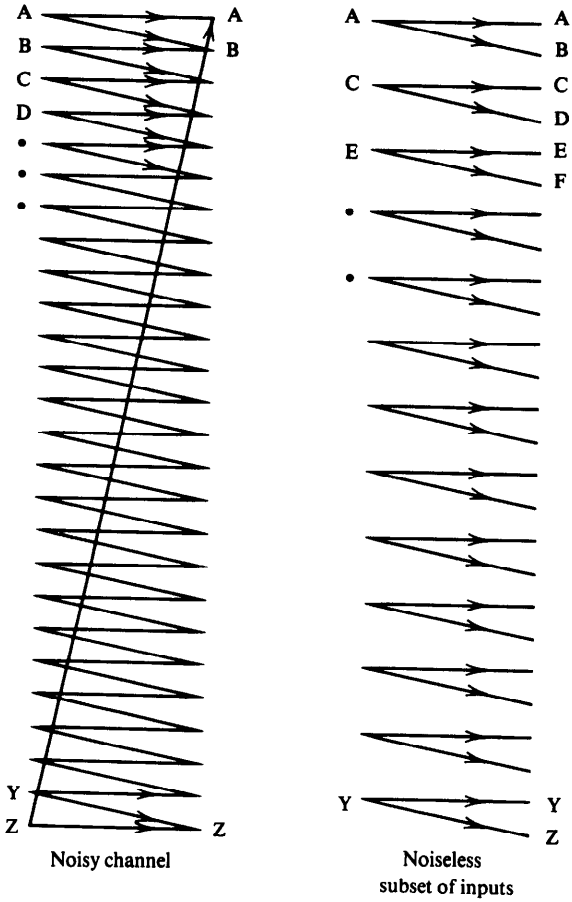


Figure 8.4. Noisy typewriter.

$\max I(X; Y) = \max[H(Y) - H(Y|X)] = \max H(Y) - 1 = \log 26 - 1 = \log 13$, achieved by using $p(x)$ uniformly distributed over all the inputs.

8.1.4 Binary Symmetric Channel

Consider the binary symmetric channel (BSC), which is shown in Figure 8.5. This is a binary channel in which the input symbols are complemented with probability p. This is the simplest model of a channel with errors; yet it captures most of the complexity of the general problem.

When an error occurs, a 0 is received as a 1 and vice versa. The received bits do not reveal where the errors have occurred. In a sense, all the received bits are unreliable. Later we show that we can still use such a communication channel to send information at a non-zero rate with an arbitrarily small probability of error.

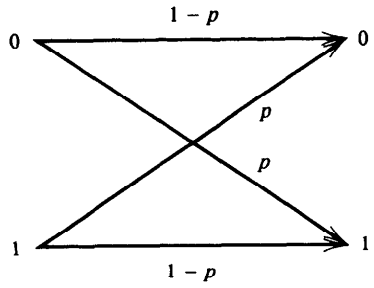


Figure 8.5. Binary symmetric channel.

We bound the mutual information by

$$I(X; Y) = H(Y) - H(Y|X) \tag{8.2}$$

$$= H(Y) - \sum p(x)H(Y|X = x) \tag{8.3}$$

$$= H(Y) - \sum p(x)H(p) \tag{8.4}$$

$$= H(Y) - H(p) \tag{8.5}$$

$$\leq 1 - H(p), \tag{8.6}$$

where the last inequality follows because Y is a binary random variable. Equality is achieved when the input distribution is uniform. Hence the information capacity of a binary symmetric channel with parameter p is

$$C = 1 - H(p) \text{ bits}. \tag{8.7}$$

8.1.5 Binary Erasure Channel

The analog of the binary symmetric channel in which some bits are lost (rather than corrupted) is called the binary erasure channel. In the binary erasure channel, a fraction α of the bits are erased. The receiver knows which bits have been erased. The binary erasure channel has two inputs and three outputs as shown in Figure 8.6.

We calculate the capacity of the binary erasure channel as follows:

$$C = \max_{p(x)} I(X; Y) \tag{8.8}$$

$$= \max_{p(x)} (H(Y) - H(Y|X)) \tag{8.9}$$

$$= \max_{p(x)} H(Y) - H(\alpha). \tag{8.10}$$

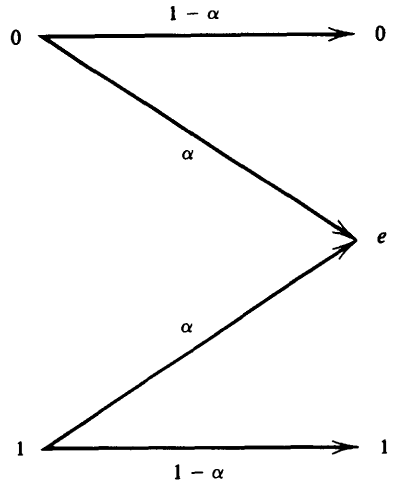


Figure 8.6. Binary erasure channel.

The first guess for the maximum of $H(Y)$ would be $\log 3$, but we cannot achieve this by any choice of input distribution $p(x)$. Letting E be the event $\{Y = e\}$, using the expansion

$$H(Y) = H(Y, E) = H(E) + H(Y|E), \tag{8.11}$$

and letting $\Pr(X = 1) = \pi$, we have

$$H(Y) = H((1-\pi)(1-\alpha), \alpha, \pi(1-\alpha)) = H(\alpha) + (1-\alpha)H(\pi). \tag{8.12}$$

Hence

$$C = \max_{p(x)} H(Y) - H(\alpha) \tag{8.13}$$

$$= \max_{\pi} (1-\alpha)H(\pi) + H(\alpha) - H(\alpha) \tag{8.14}$$

$$= \max_{\pi} (1-\alpha)H(\pi) \tag{8.15}$$

$$= 1 - \alpha, \tag{8.16}$$

where capacity is achieved by $\pi = \frac{1}{2}$.

The expression for the capacity has some intuitive meaning: since a proportion α of the bits are lost in the channel, we can recover (at most) a proportion $1 - \alpha$ of the bits. Hence the capacity is at most $1 - \alpha$. It is not immediately obvious that it is possible to achieve this rate. This will follow from Shannon's second theorem.

In many practical channels, the sender receives some feedback from the receiver. If feedback is available for the binary erasure channel, it is very clear what to do: if a bit is lost, retransmit it until it gets through. Since the bits get through with probability $1 - \alpha$, the effective rate of transmission is $1 - \alpha$. In this way we are easily able to achieve a capacity of $1 - \alpha$ with feedback.

Later in the chapter, we will prove that the rate $1 - \alpha$ is the best that can be achieved both with and without feedback. This is one of the consequences of the surprising fact that feedback does not increase the capacity of discrete memoryless channels.

8.2 SYMMETRIC CHANNELS

The capacity of the binary symmetric channel is $C = 1 - H(p)$ bits per transmission and the capacity of the binary erasure channel is $C = 1 - \alpha$ bits per transmission.

Now consider the channel with transmission matrix:

$$p(y|x) = \begin{bmatrix} 0.3 & 0.2 & 0.5 \\ 0.5 & 0.3 & 0.2 \\ 0.2 & 0.5 & 0.3 \end{bmatrix}. \tag{8.17}$$

Here the entry in the xth row and the yth column denotes the conditional probability $p(y|x)$ that y is received when x is sent. In this channel, all the rows of the probability transition matrix are permutations of each other and so are the columns. Such a channel is said to be *symmetric*. Another example of a symmetric channel is one of the form

$$Y = X + Z \pmod{c}, \tag{8.18}$$

where Z has some distribution on the integers $\{0, 1, 2, \ldots, c-1\}$, and X has the same alphabet as Z, and Z is independent of X.

In both these cases, we can easily find an explicit expression for the capacity of the channel. Letting $\mathbf{r}$ be a row of the transition matrix, we have

$$I(X; Y) = H(Y) - H(Y|X) \tag{8.19}$$

$$= H(Y) - H(\mathbf{r}) \tag{8.20}$$

$$\leq \log|\mathcal{Y}| - H(\mathbf{r}) \tag{8.21}$$

with equality if the output distribution is uniform. But $p(x) = 1/|\mathcal{X}|$ achieves a uniform distribution on Y, as seen from

$$p(y) = \sum_{x \in \mathcal{X}} p(y|x)p(x) = \frac{1}{|\mathcal{X}|} \sum p(y|x) = c \frac{1}{|\mathcal{X}|} = \frac{1}{|\mathcal{Y}|}, \qquad (8.22)$$

where c is the sum of the entries in one column of the probability transition matrix.

Thus the channel in (8.17) has capacity

$$C = \max_{p(x)} I(X; Y) = \log 3 - H(0.5, 0.3, 0.2), \qquad (8.23)$$

and C is achieved by a uniform distribution on the input.

The transition matrix of the symmetric channel defined above is doubly stochastic. In the computation of the capacity, we used the facts that the rows were permutations of one another and that all the column sums were equal.

Considering these properties, we can define a generalization of the concept of a symmetric channel as follows:

Definition: A channel is said to be *symmetric* if the rows of the channel transition matrix $p(y|x)$ are permutations of each other, and the columns are permutations of each other. A channel is said to be *weakly symmetric* if every row of the transition matrix $p(\cdot|x)$ is a permutation of every other row, and all the column sums $\Sigma_x p(y|x)$ are equal.

For example, the channel with transition matrix

$$p(y|x) = \begin{pmatrix} \frac{1}{3} & \frac{1}{6} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{6} \end{pmatrix} \qquad (8.24)$$

is weakly symmetric but not symmetric.

The above derivation for symmetric channels carries over to weakly symmetric channels as well. We have the following theorem for weakly symmetric channels:

Theorem 8.2.1: *For a weakly symmetric channel,*

$$C = \log|\mathcal{Y}| - H(\text{row of transition matrix}), \qquad (8.25)$$

and this is achieved by a uniform distribution on the input alphabet.

8.3 PROPERTIES OF CHANNEL CAPACITY

1. $C \geq 0$, since $I(X; Y) \geq 0$.
2. $C \leq \log|\mathcal{X}|$ since $C = \max I(X; Y) \leq \max H(X) = \log|\mathcal{X}|$.

3. $C \leq \log|\mathcal{Y}|$ for the same reason.
4. $I(X; Y)$ is a continuous function of $p(x)$.
5. $I(X; Y)$ is a concave function of $p(x)$ (Theorem 2.7.4).

Since $I(X; Y)$ is a concave function over a closed convex set, a local maximum is a global maximum. From properties (2) and (3), the maximum is finite, and we are justified in using the term maximum, rather than supremum in the definition of capacity.

The maximum can then be found by standard nonlinear optimization techniques like gradient search. Some of the methods that can be used include the following:

- Constrained maximization using calculus and the Kuhn-Tucker conditions.
- The Frank-Wolfe gradient search algorithm.
- An iterative algorithm developed by Arimoto [11] and Blahut [37]. We will describe the algorithm in Section 13.8.

In general, there is no closed form solution for the capacity. But for many simple channels it is possible to calculate the capacity using properties like symmetry. Some of the examples considered earlier are of this form.

8.4 PREVIEW OF THE CHANNEL CODING THEOREM

So far, we have defined the information capacity of a discrete memoryless channel. In the next section, we prove Shannon's second theorem, which gives an operational meaning to the definition of capacity as the number of bits we can transmit reliably over the channel.

But first we will try to give an intuitive idea as to why we can transmit C bits of information over a channel. The basic idea is that, for large block lengths, every channel looks like the noisy typewriter channel (Figure 8.4) and the channel has a subset of inputs that produce essentially disjoint sequences at the output.

For each (typical) input n-sequence, there are approximately $2^{nH(Y|X)}$ possible Y sequences, all of them equally likely (Figure 8.7). We wish to ensure that no two X sequences produce the same Y output sequence. Otherwise, we will not be able to decide which X sequence was sent.

The total number of possible (typical) Y sequences is $\approx 2^{nH(Y)}$. This set has to be divided into sets of size $2^{nH(Y|X)}$ corresponding to the different input X sequences. The total number of disjoint sets is less than or equal to $2^{n(H(Y)-H(Y|X))} = 2^{nI(X;Y)}$. Hence we can send at most $\approx 2^{nI(X;Y)}$ distinguishable sequences of length n.

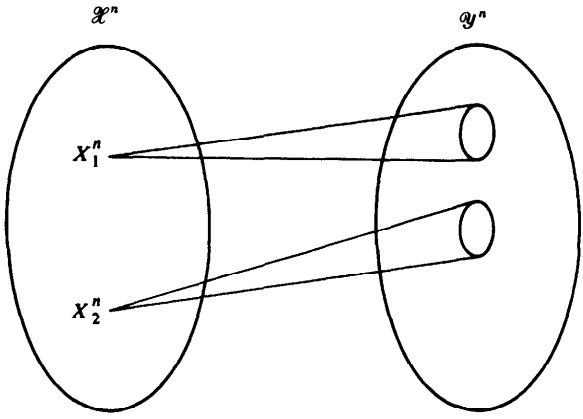


Figure 8.7. Channels after n uses.

Although the above derivation outlines an upper bound on the capacity, a stronger version of the above argument will be used in the next section to prove that this rate I is achievable, with an arbitrarily low probability of error.

Before we proceed to the proof of Shannon's second theorem, we need a few definitions.

8.5 DEFINITIONS

We analyze a communication system as shown in Figure 8.8.

A message W, drawn from the index set $\{1, 2, \ldots, M\}$, results in the signal $X^n(W)$, which is received as a random sequence $Y^n \sim p(y^n|x^n)$ by the receiver. The receiver then guesses the index W by an appropriate decoding rule $\hat{W} = g(Y^n)$. The receiver makes an error if $\hat{W}$ is not the same as the index W that was transmitted.

We now define these ideas formally.

Definition: A *discrete channel*, denoted by $(\mathcal{X}, p(y|x), \mathcal{Y})$, consists of two finite sets $\mathcal{X}$ and $\mathcal{Y}$ and a collection of probability mass functions $p(y|x)$, one for each $x \in \mathcal{X}$, such that for every x and y, $p(y|x) \geq 0$, and for every x, $\sum_y p(y|x) = 1$, with the interpretation that X is the input and Y is the output of the channel.

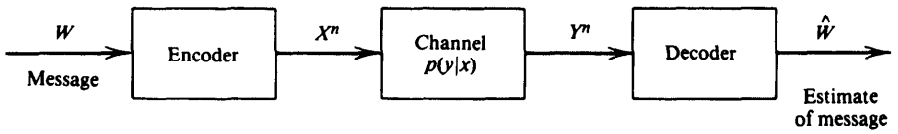


Figure 8.8. A communication channel.

Definition: The *nth extension of the discrete memoryless channel* (DMC) is the channel $(\mathcal{X}^n, p(y^n|x^n), \mathcal{Y}^n)$, where

$$p(y_k|x^k, y^{k-1}) = p(y_k|x_k), \qquad k = 1, 2, \ldots, n . \tag{8.26}$$

Remark: If the channel is used *without feedback*, i.e., if the input symbols do not depend on the past output symbols, namely, $p(x_k|x^{k-1}, y^{k-1}) = p(x_k|x^{k-1})$, then the channel transition function for the n-th extension of the discrete memoryless channel reduces to

$$p(y^n|x^n) = \prod_{i=1}^{n} p(y_i|x_i) . \tag{8.27}$$

When we refer to the discrete memoryless channel, we shall mean the discrete memoryless channel without feedback, unless we explicitly state otherwise.

Definition: An (M, n) code for the channel $(\mathcal{X}, p(y|x), \mathcal{Y})$ consists of the following:

1. An index set $\{1, 2, \ldots, M\}$.
2. An encoding function $X^n : \{1, 2, \ldots, M\} \to \mathcal{X}^n$, yielding codewords $X^n(1), X^n(2), \ldots, X^n(M)$. The set of codewords is called the codebook.
3. A decoding function

$$g: \mathcal{Y}^n \to \{1, 2, \ldots, M\} , \tag{8.28}$$

which is a deterministic rule which assigns a guess to each possible received vector.

Definition (*Probability of error*): Let

$$\lambda_i = \Pr(g(Y^n) \neq i | X^n = X^n(i)) = \sum_{y^n} p(y^n|x^n(i)) I(g(y^n) \neq i) \tag{8.29}$$

be the conditional probability of error given that index i was sent, where $I(\cdot)$ is the indicator function.

Definition: The *maximal probability of error* $\lambda^{(n)}$ for an (M, n) code is defined as

$$\lambda^{(n)} = \max_{i \in \{1, 2, \ldots, M\}} \lambda_i . \tag{8.30}$$

Definition: The (arithmetic) *average probability of error* $P_e^{(n)}$ for an (M, n) code is defined as

$$P_e^{(n)} = \frac{1}{M} \sum_{i=1}^{M} \lambda_i . \tag{8.31}$$

Note that

$$P_e^{(n)} = \Pr(I \neq g(Y^n)) \tag{8.32}$$

if the index I is chosen uniformly on the set $\{1, 2, \ldots, M\}$. Also obviously

$$P_e^{(n)} \leq \lambda^{(n)} . \tag{8.33}$$

One would expect the maximal probability of error to behave quite differently from the average probability. But in the next section, we will prove that a small average probability of error implies a small maximal probability of error at essentially the same rate.

Definition: The *rate* R of an (M, n) code is

$$R = \frac{\log M}{n} \text{ bits per transmission} . \tag{8.34}$$

Definition: A rate R is said to be *achievable* if there exists a sequence of $(\lceil 2^{nR} \rceil, n)$ codes such that the maximal probability of error $\lambda^{(n)}$ tends to 0 as $n \to \infty$.

Later, we will write $(2^{nR}, n)$ codes to mean $(\lceil 2^{nR} \rceil, n)$ codes. This will simplify the notation.

Definition: The *capacity* of a discrete memoryless channel is the supremum of all achievable rates.

Thus rates less than capacity yield arbitrarily small probability of error for sufficiently large block lengths.

8.6 JOINTLY TYPICAL SEQUENCES

Roughly speaking, we will decode a channel output Y^n as the ith index if the codeword $X^n(i)$ is "jointly typical" with the received signal Y^n. We now define the important idea of joint typicality and find the probability of joint typicality when $X^n(i)$ is the true cause of Y^n and when it is not.

Definition: The set $A_\epsilon^{(n)}$ of *jointly typical* sequences $\{(x^n, y^n)\}$ with respect to the distribution $p(x, y)$ is the set of n-sequences with empirical entropies ϵ-close to the true entropies, i.e.,

$$A_\epsilon^{(n)} = \Big\{(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \tag{8.35}$$

$$\left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon , \tag{8.36}$$

$$\left| -\frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon , \tag{8.37}$$

$$\left| -\frac{1}{n} \log p(x^n, y^n) - H(X, Y) \right| < \epsilon \Big\} , \tag{8.38}$$

where

$$p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i) . \tag{8.39}$$

Theorem 8.6.1 (*Joint AEP*): *Let (X^n, Y^n) be sequences of length n drawn i.i.d. according to $p(x^n, y^n) = \prod_{i=1}^n p(x_i, y_i)$. Then*

1. $\Pr((X^n, Y^n) \in A_\epsilon^{(n)}) \to 1$ *as* $n \to \infty$.
2. $|A_\epsilon^{(n)}| \leq 2^{n(H(X,Y)+\epsilon)}$.
3. *If $(\tilde{X}^n, \tilde{Y}^n) \sim p(x^n)p(y^n)$, i.e., $\tilde{X}^n$ and $\tilde{Y}^n$ are independent with the same marginals as $p(x^n, y^n)$, then*

$$\Pr((\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)}) \leq 2^{-n(I(X;Y)-3\epsilon)} . \tag{8.40}$$

Also, for sufficiently large n,

$$\Pr((\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)}) \geq (1-\epsilon)2^{-n(I(X;Y)+3\epsilon)} . \tag{8.41}$$

Proof: By the weak law of large numbers,

$$-\frac{1}{n} \log p(X^n) \to -E[\log p(X)] = H(X) \quad \text{in probability} . \tag{8.42}$$

Hence, given $\epsilon > 0$, there exists n_1, such that for all $n > n_1$,

$$\Pr\left(\left| -\frac{1}{n} \log p(X^n) - H(X) \right| > \epsilon \right) < \frac{\epsilon}{3} . \tag{8.43}$$

Similarly, by the weak law,

$$-\frac{1}{n} \log p(Y^n) \to -E[\log p(Y)] = H(Y) \quad \text{in probability} , \tag{8.44}$$

and

$$-\frac{1}{n}\log p(X^n, Y^n) \to -E[\log p(X, Y)] = H(X, Y) \quad \text{in probability}, \tag{8.45}$$

and there exist n_2 and n_3 such that for all $n \geq n_2$,

$$\Pr\left(\left|-\frac{1}{n}\log p(Y^n) - H(Y)\right| > \epsilon\right) < \frac{\epsilon}{3} \tag{8.46}$$

and for all $n \geq n_3$,

$$\Pr\left(\left|-\frac{1}{n}\log p(X^n, Y^n) - H(X, Y)\right| > \epsilon\right) < \frac{\epsilon}{3}. \tag{8.47}$$

Choosing $n > \max\{n_1, n_2, n_3\}$, the probability of the union of the sets in (8.43), (8.46) and (8.47) must be less than ϵ. Hence for n sufficiently large, the probability of the set $A_\epsilon^{(n)}$ is greater than $1 - \epsilon$, establishing the first part of the theorem.

To prove the second part of the theorem, we have

$$1 = \sum p(x^n, y^n) \tag{8.48}$$

$$\geq \sum_{A_\epsilon^{(n)}} p(x^n, y^n) \tag{8.49}$$

$$\geq |A_\epsilon^{(n)}| 2^{-n(H(X, Y)+\epsilon)}, \tag{8.50}$$

and hence

$$|A_\epsilon^{(n)}| \leq 2^{n(H(X, Y)+\epsilon)}. \tag{8.51}$$

For sufficiently large n, $\Pr(A_\epsilon^{(n)}) \geq 1 - \epsilon$, and therefore

$$1 - \epsilon \leq \sum_{(x^n, y^n) \in A_\epsilon^{(n)}} p(x^n, y^n) \tag{8.52}$$

$$\leq |A_\epsilon^{(n)}| 2^{-n(H(X, Y)-\epsilon)}, \tag{8.53}$$

and

$$|A_\epsilon^{(n)}| \geq (1 - \epsilon) 2^{n(H(X, Y)-\epsilon)}. \tag{8.54}$$

Now if $\tilde{X}^n$ and $\tilde{Y}^n$ are independent but have the same marginals as X^n and Y^n, then

$$\Pr((\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)}) = \sum_{(x^n, y^n) \in A_\epsilon^{(n)}} p(x^n)p(y^n) \tag{8.55}$$

$$\leq 2^{n(H(X,Y)+\epsilon)} 2^{-n(H(X)-\epsilon)} 2^{-n(H(Y)-\epsilon)} \tag{8.56}$$

$$= 2^{-n(I(X;Y)-3\epsilon)}. \tag{8.57}$$

By similar arguments, we can also show that

$$\Pr((\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)}) = \sum_{A_\epsilon^{(n)}} p(x^n)p(y^n) \tag{8.58}$$

$$\geq (1-\epsilon)2^{n(H(X,Y)-\epsilon)} 2^{-n(H(X)+\epsilon)} 2^{-n(H(Y)+\epsilon)} \tag{8.59}$$

$$= (1-\epsilon)2^{-n(I(X;Y)+3\epsilon)}. \tag{8.60}$$

This completes the proof of the theorem. □

The jointly typical set is illustrated in Figure 8.9. There are about $2^{nH(X)}$ typical X sequences, and about $2^{nH(Y)}$ typical Y sequences. However, since there are only $2^{nH(X,Y)}$ jointly typical sequences, not all pairs of typical X^n and typical Y^n are also jointly typical. The probability that any randomly chosen pair is jointly typical is about $2^{-nI(X;Y)}$. Hence, for a fixed Y^n, we can consider about $2^{nI(X;Y)}$ such pairs before we are likely to come across a jointly typical pair. This suggests that there are about $2^{nI(X;Y)}$ distinguishable signals X^n.

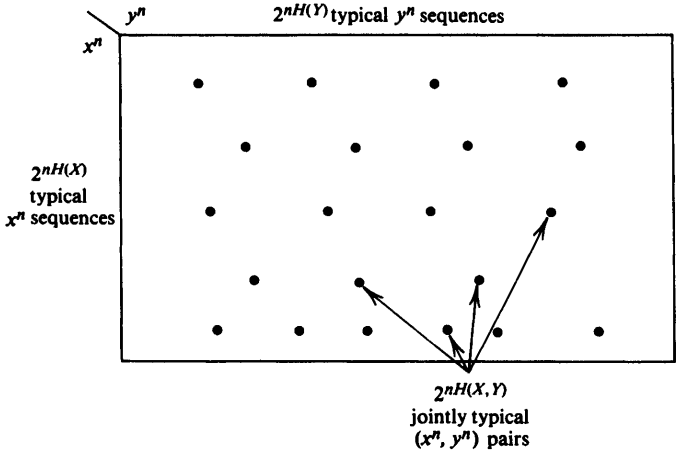


Figure 8.9. Jointly typical sequences.

8.7 THE CHANNEL CODING THEOREM

We now prove what is perhaps the basic theorem of information theory, the achievability of channel capacity. The basic argument was first stated by Shannon in his original 1948 paper. The result is rather counterintuitive; if the channel introduces errors, how can one correct them all? Any correction process is also subject to error, ad infinitum.

Shannon used a number of new ideas in order to prove that information can be sent reliably over a channel at all rates up to the channel capacity. These ideas include

- Allowing an arbitrarily small but non-zero probability of error,
- Using the channel many times in succession, so that the law of large numbers comes into effect, and
- Calculating the average of the probability of error over a random choice of codebooks, which symmetrizes the probability, and which can then be used to show the existence of at least one good code.

Shannon's outline of the proof was based on the idea of typical sequences, but the proof was not made rigorous until much later. The proof given below makes use of the properties of typical sequences and is probably the simplest of the proofs developed so far. As in all the proofs, we use the same essential ideas—random code selection, calculation of the average probability of error for a random choice of codewords, etc. The main difference is in the decoding rule. In the proof, we decode by joint typicality; we look for a codeword that is jointly typical with the received sequence. If we find a unique codeword satisfying this property, we declare that word to be the transmitted codeword. By the properties of joint typicality stated previously, with high probability the transmitted codeword and the received sequence are jointly typical, since they are probabilistically related. Also, the probability that any other codeword looks jointly typical with the received sequence is 2^{-nI}. Hence, if we have fewer then 2^{nI} codewords, then with high probability, there will be no other codewords that can be confused with the transmitted codeword, and the probability of error is small.

Although jointly typical decoding is suboptimal, it is simple to analyze and still achieves all rates below capacity.

We shall now give the complete statement and proof of Shannon's second theorem:

Theorem 8.7.1 (*The channel coding theorem*): *All rates below capacity* C *are achievable. Specifically, for every rate* $R < C$, *there exists a sequence of* $(2^{nR}, n)$ *codes with maximum probability of error* $\lambda^{(n)} \to 0$.

Conversely, any sequence of $(2^{nR}, n)$ *codes with* $\lambda^{(n)} \to 0$ *must have* $R \leq C$.

Proof: We prove that rates $R < C$ are achievable and postpone the proof of the converse to Section 8.9.

Achievability: Fix $p(x)$. Generate a $(2^{nR}, n)$ code at random according to the distribution $p(x)$. Specifically, we independently generate 2^{nR} codewords according to the distribution,

$$p(x^n) = \prod_{i=1}^{n} p(x_i) . \tag{8.61}$$

We exhibit the 2^{nR} codewords as the rows of a matrix:

$$\mathscr{C} = \begin{bmatrix} x_1(1) & x_2(1) & \cdots & x_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(2^{nR}) & x_2(2^{nR}) & \cdots & x_n(2^{nR}) \end{bmatrix} . \tag{8.62}$$

Each entry in this matrix is generated i.i.d. according to $p(x)$. Thus the probability that we generate a particular code $\mathscr{C}$ is

$$\Pr(\mathscr{C}) = \prod_{w=1}^{2^{nR}} \prod_{i=1}^{n} p(x_i(w)) . \tag{8.63}$$

Consider the following sequence of events:

1. A random code $\mathscr{C}$ is generated as described in (8.63) according to $p(x)$.
2. The code $\mathscr{C}$ is then revealed to both sender and receiver. Both sender and receiver are also assumed to know the channel transition matrix $p(y|x)$ for the channel.
3. A message W is chosen according to a uniform distribution

$$\Pr(W = w) = 2^{-nR}, \qquad w = 1, 2, \ldots, 2^{nR} . \tag{8.64}$$

4. The wth codeword $X^n(w)$, corresponding to the wth row of $\mathscr{C}$, is sent over the channel.
5. The receiver receives a sequence Y^n according to the distribution:

$$P(y^n|x^n(w)) = \prod_{i=1}^{n} p(y_i|x_i(w)) . \tag{8.65}$$

6. The receiver guesses which message was sent. (The optimum procedure to minimize probability of error is maximum likelihood decoding, i.e., the receiver should choose the *a posteriori* most likely message. But this procedure is difficult to analyze. Instead, we will use *typical set decoding*, which is described below. Typical

set decoding is easier to analyze and is asymptotically optimal.) The receiver declares that the index $\hat{W}$ was sent if the following conditions are satisfied:

- $(X^n(\hat{W}), Y^n)$ is jointly typical.
- There is no other index k, such that $(X^n(k), Y^n) \in A_\epsilon^{(n)}$.

If no such $\hat{W}$ exists or if there is more than one such, then an error is declared. (We may assume that the receiver outputs a dummy index such as 0 in this case.)

7. There is a decoding error if $\hat{W} \neq W$. Let $\mathscr{E}$ be the event $\{\hat{W} \neq W\}$.

Analysis of the probability of error

Outline: We first outline the analysis.

Instead of calculating the probability of error for a single code, we calculate the average over all codes generated at random according to the distribution (8.63). By the symmetry of the code construction, the average probability of error does not depend on the particular index that was sent. For a typical codeword, there are two different sources of error when we use typical set decoding: either the output Y^n is not jointly typical with the transmitted codeword or there is some other codeword that is jointly typical with Y^n. The probability that the transmitted codeword and the received sequence are jointly typical goes to one as shown by the joint AEP. For any rival codeword, the probability that it is jointly typical with the received sequence is approximately 2^{-nI}, and hence we can use about 2^{nI} codewords and still have low probability of error. We will later extend the argument to find a code with low maximal probability of error.

Detailed calculation of the probability of error: We will calculate the average probability of error, averaged over all codewords in the codebook, and averaged over all codebooks, i.e., we calculate

$$\Pr(\mathscr{E}) = \sum_{\mathscr{C}} P(\mathscr{C}) P_e^{(n)}(\mathscr{C}) \tag{8.66}$$

$$= \sum_{\mathscr{C}} P(\mathscr{C}) \frac{1}{2^{nR}} \sum_{w=1}^{2^{nR}} \lambda_w(\mathscr{C}) \tag{8.67}$$

$$= \frac{1}{2^{nR}} \sum_{w=1}^{2^{nR}} \sum_{\mathscr{C}} P(\mathscr{C}) \lambda_w(\mathscr{C}), \tag{8.68}$$

where $P_e^{(n)}(\mathscr{C})$ is defined for typical set decoding.

By the symmetry of the code construction, the average probability of error averaged over all codes does not depend on the particular index that was sent, i.e., $\Sigma_{\mathscr{C}} P(\mathscr{C})\lambda_w(\mathscr{C})$ does not depend on w. Thus we can assume without loss of generality that the message $W=1$ was sent, since

$$\Pr(\mathscr{E}) = \frac{1}{2^{nR}} \sum_{w=1}^{2^{nR}} \sum_{\mathscr{C}} P(\mathscr{C})\lambda_w(\mathscr{C}) \tag{8.69}$$

$$= \sum_{\mathscr{C}} P(\mathscr{C})\lambda_1(\mathscr{C}) \tag{8.70}$$

$$= \Pr(\mathscr{E}|W=1)\,. \tag{8.71}$$

Define the following events:

$$E_i = \{(X^n(i), Y^n) \text{ is in } A_\epsilon^{(n)}\}, \quad i \in \{1, 2, \ldots, 2^{nR}\}\,, \tag{8.72}$$

where E_i is the event that the ith codeword and Y^n are jointly typical. Recall that Y^n is the result of sending the first codeword $X^n(1)$ over the channel.

Then an error occurs in the decoding scheme if either E_1^c occurs (when the transmitted codeword and the received sequence are not jointly typical) or $E_2 \cup E_3 \cup \cdots \cup E_{2^{nR}}$ occurs (when a wrong codeword is jointly typical with the received sequence). Hence, letting $P(\mathscr{E})$ denote $\Pr\{\mathscr{E}|W=1\}$, we have

$$\Pr(\mathscr{E}|W=1) = P(E_1^c \cup E_2 \cup E_3 \cup \cdots \cup E_{2^{nR}}) \tag{8.73}$$

$$\leq P(E_1^c) + \sum_{i=2}^{2^{nR}} P(E_i)\,, \tag{8.74}$$

by the union of events bound for probabilities. Now, by the joint AEP, $P(E_1^c) \to 0$, and hence

$$P(E_1^c) \leq \epsilon\,, \qquad \text{for } n \text{ sufficiently large}\,. \tag{8.75}$$

Since by the code generation process, $X^n(1)$ and $X^n(i)$ are independent, so are Y^n and $X^n(i)$, $i \neq 1$. Hence, the probability that $X^n(i)$ and Y^n are jointly typical is $\leq 2^{-n(I(X;Y)-3\epsilon)}$ by the joint AEP. Consequently,

$$P(\mathscr{E}) = P(\mathscr{E}|W=1) \leq P(E_1^c) + \sum_{i=2}^{2^{nR}} P(E_i) \tag{8.76}$$

$$\leq \epsilon + \sum_{i=2}^{2^{nR}} 2^{-n(I(X;Y)-3\epsilon)} \tag{8.77}$$

$$= \epsilon + (2^{nR} - 1)2^{-n(I(X;Y)-3\epsilon)} \tag{8.78}$$

$$\leq \epsilon + 2^{3n\epsilon}2^{-n(I(X;Y)-R)} \tag{8.79}$$

$$\leq 2\epsilon \tag{8.80}$$

if n is sufficiently large and $R < I(X;Y) - 3\epsilon$.

Hence, if $R < I(X;Y)$, we can choose ϵ and n so that the average probability of error, averaged over codebooks and codewords, is less than 2ϵ.

To finish the proof, we will strengthen this conclusion by a series of code selections.

1. Choose $p(x)$ in the proof to be $p^*(x)$, the distribution on X that achieves capacity. Then the condition $R < I(X;Y)$ can be replaced by the achievability condition $R < C$.
2. Get rid of the average over codebooks. Since the average probability of error over codebooks is small ($\leq 2\epsilon$), there exists at least one codebook $\mathscr{C}^*$ with a small average probability of error. Thus $P_e^n(\mathscr{C}^*) \leq 2\epsilon$. Determination of $\mathscr{C}^*$ can be achieved by an exhaustive search over all $(2^{nR}, n)$ codes.
3. Throw away the worst half of the codewords in the best codebook $\mathscr{C}^*$. Since the average probability of error for this code is less then 2ϵ, we have

$$2\epsilon \geq \frac{1}{2^{nR}} \sum \lambda_i(\mathscr{C}^*), \tag{8.81}$$

which implies that at least half the indices i and their associated codewords $X^n(i)$ must have conditional probability of error λ_i less than 4ϵ (otherwise, these codewords themselves would contribute more than 2ϵ to the sum). Hence the best half of the codewords have a maximal probability of error less than 4ϵ. If we reindex these codewords, we have 2^{nR-1} codewords. Throwing out half the codewords has changed the rate from R to $R - \frac{1}{n}$, which is negligible for large n.

Combining all these improvements, we have constructed a code of rate $R' = R - \frac{1}{n}$, with maximal probability of error $\lambda^{(n)} \leq 4\epsilon$. This proves the achievability of any rate below capacity. □

Random coding is the method of proof for the above theorem, not the method of signalling. Codes are selected at random in the proof merely to symmetrize the mathematics and to show the existence of a good deterministic code. We proved that the average over all codes of block

length n has small probability of error. We can find the best code within this set by an exhaustive search. Incidentally, this shows that the Kolmogorov complexity of the best code is a small constant. This means that the revelation (in step 2) to the sender and receiver of the best code $\mathscr{C}^*$ requires no channel. The sender and receiver merely agree to use the best $(2^{nR}, n)$ code for the channel.

Although the theorem shows that there exist good codes with exponentially small probability of error for long block lengths, it does not provide a way of constructing the best codes. If we used the scheme suggested by the proof and generate a code at random with the appropriate distribution, the code constructed is likely to be good for long block lengths. However, without some structure in the code, it is very difficult to decode (the simple scheme of table lookup requires an exponentially large table). Hence the theorem does not provide a practical coding scheme. Ever since Shannon's original paper on information theory, researchers have tried to develop structured codes that are easy to encode and decode. So far, they have developed many codes with interesting and useful structures, but the asymptotic rates of these codes are not yet near capacity.

8.8 ZERO-ERROR CODES

The outline of the proof of the converse is most clearly motivated by going through the argument when absolutely no errors are allowed. We will now prove that $P_e^{(n)} = 0$ implies $R \leq C$.

Assume that we have a $(2^{nR}, n)$ code with zero probability of error, i.e., the decoder output $g(Y^n)$ is equal to the input index W with probability 1. Then the input index W is determined by the output sequence, i.e., $H(W|Y^n) = 0$. Now, to obtain a strong bound, we arbitrarily assume that W is uniformly distributed over $\{1, 2, \ldots, 2^{nR}\}$. Thus $H(W) = nR$. We can now write the string of inequalities:

$$nR = H(W) = \underbrace{H(W|Y^n)}_{=0} + I(W; Y^n) \tag{8.82}$$

$$= I(W; Y^n) \tag{8.83}$$

$$\overset{(a)}{\leq} I(X^n; Y^n) \tag{8.84}$$

$$\overset{(b)}{\leq} \sum_{i=1}^{n} I(X_i; Y_i) \tag{8.85}$$

$$\overset{(c)}{\leq} nC, \tag{8.86}$$

where (a) follows from the data processing inequality (since $W \to X^n(W) \to Y^n$ forms a Markov chain), (b) will be proved in Lemma 8.9.2 using the discrete memoryless assumption, and (c) follows from the definition of (information) capacity.

Hence, for any zero-error $(2^{nR}, n)$ code, for all n,

$$R \leq C . \tag{8.87}$$

8.9 FANO'S INEQUALITY AND THE CONVERSE TO THE CODING THEOREM

We now extend the proof that was derived for zero-error codes to the case of codes with very small probabilities of error. The new ingredient will be Fano's inequality, which gives a lower bound on the probability of error in terms of the conditional entropy. Recall the proof of Fano's inequality, which is repeated here in a new context for reference.

Let us define the setup under consideration. The index W is uniformly distributed on the set $\mathcal{W} = \{1, 2, \ldots, 2^{nR}\}$, and the sequence Y^n is probabilistically related to W. From Y^n, we estimate the index W that was sent. Let the estimate be $\hat{W} = g(Y^n)$. Define the probability of error

$$P_e^{(n)} = \Pr(\hat{W} \neq W) . \tag{8.88}$$

Define

$$E = \begin{cases} 1, & \text{if } \hat{W} \neq W , \\ 0, & \text{if } \hat{W} = W . \end{cases} \tag{8.89}$$

Then using the chain rule for entropies to expand $H(E, W|Y^n)$ in two different ways, we have

$$H(E, W|Y^n) = H(W|Y^n) + H(E|W, Y^n) \tag{8.90}$$

$$= H(E|Y^n) + H(W|E, Y^n) . \tag{8.91}$$

Now since E is a function of W and $g(Y^n)$, it follows that $H(E|W, Y^n) = 0$. Also, $H(E) \leq 1$, since E is a binary valued random variable. The remaining term, $H(W|E, Y^n)$, can be bounded as follows:

$$H(W|E, Y^n) = P(E = 0)H(W|Y^n, E = 0) + P(E = 1)H(W|Y^n, E = 1) \tag{8.92}$$

$$\leq (1 - P_e^{(n)})0 + P_e^{(n)} \log(|\mathcal{W}| - 1) \tag{8.93}$$

$$\leq P_e^{(n)} nR , \tag{8.94}$$

since given $E = 0$, $W = g(Y^n)$, and when $E = 1$, we can upper bound the conditional entropy by the logarithm of the number of outcomes. Combining these results, we obtain Fano's inequality:

$$H(W|Y^n) \leq 1 + P_e^{(n)} nR . \tag{8.95}$$

Since for a fixed code $X^n(W)$ is a function of W,

$$H(X^n(W)|Y^n) \leq H(W|Y^n) . \tag{8.96}$$

Then we have the following lemma.

Lemma 8.9.1 (*Fano's inequality*): *For a discrete memoryless channel with a codebook $\mathscr{C}$ and the input messages uniformly distributed, let $P_e^{(n)} = \Pr(W \neq g(Y^n))$. Then*

$$H(X^n|Y^n) \leq 1 + P_e^{(n)} nR . \tag{8.97}$$

We will now prove a lemma which shows that the capacity per transmission is not increased if we use a discrete memoryless channel many times.

Lemma 8.9.2: *Let Y^n be the result of passing X^n through a discrete memoryless channel. Then*

$$I(X^n; Y^n) \leq nC, \qquad \text{for all } p(x^n) . \tag{8.98}$$

Proof:

$$I(X^n; Y^n) = H(Y^n) - H(Y^n|X^n) \tag{8.99}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|Y_1, \ldots, Y_{i-1}, X^n) \tag{8.100}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i) , \tag{8.101}$$

since by the definition of a discrete memoryless channel, Y_i depends only on X_i and is conditionally independent of everything else. Continuing the series of inequalities, we have

$$I(X^n; Y^n) = H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.102}$$

$$\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.103}$$

$$= \sum_{i=1}^{n} I(X_i; Y_i) \tag{8.104}$$

$$\leq nC, \tag{8.105}$$

where (8.103) follows from the fact that the entropy of a collection of random variables is less than the sum of their individual entropies, and (8.105) follows from the definition of capacity. Thus we have proved that using the channel many times does not increase the information capacity in bits per transmission. □

We are now in a position to prove the converse to the channel coding theorem.

Proof: *Converse to Theorem 8.7.1, (the channel coding theorem)*: We have to show that any sequence of $(2^{nR}, n)$ codes with $\lambda^{(n)} \to 0$ must have $R \leq C$.

If the maximal probability of error tends to zero, then the average probability of error for the sequence of codes also goes to zero, i.e., $\lambda^{(n)} \to 0$ implies $P_e^{(n)} \to 0$, where $P_e^{(n)}$ is defined in (8.31). For each n, let W be drawn according to a uniform distribution over $\{1, 2, \ldots, 2^{nR}\}$. Since W has a uniform distribution, $P_e^{(n)} = \Pr(\hat{W} \neq W)$. Hence

$$nR = H(W) = H(W|Y^n) + I(W; Y^n) \tag{8.106}$$

$$\leq H(W|Y^n) + I(X^n(W); Y^n) \tag{8.107}$$

$$\leq 1 + P_e^{(n)} nR + I(X^n(W); Y^n) \tag{8.108}$$

$$\leq 1 + P_e^{(n)} nR + nC \tag{8.109}$$

by Lemma 8.9.1 and Lemma 8.9.2. Dividing by n, we obtain

$$R \leq P_e^{(n)} R + \frac{1}{n} + C. \tag{8.110}$$

Now letting $n \to \infty$, we see that the first two terms on the right hand side tend to 0, and hence

$$R \leq C. \tag{8.111}$$

We can rewrite (8.110) as

$$P_e^{(n)} \geq 1 - \frac{C}{R} - \frac{1}{nR}. \tag{8.112}$$

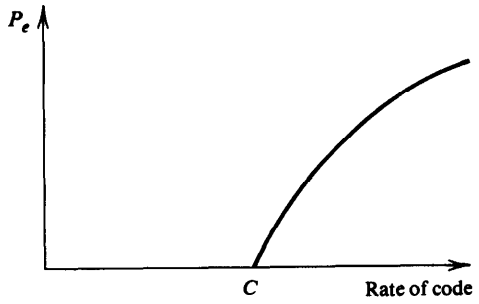


Figure 8.10. Lower bound on the probability of error.

This equation shows that if $R > C$, the probability of error is bounded away from 0 for sufficiently large n (and hence for all n, since if $P_e^{(n)} = 0$ for small n, we can construct codes for large n with $P_e^{(n)} = 0$ by concatenating these codes). Hence we cannot achieve an arbitrarily low probability of error at rates above capacity. This inequality is illustrated graphically in Figure 8.10. □

This converse is sometimes called the weak converse to the channel coding theorem. It is also possible to prove a strong converse, which states that for rates above capacity, the probability of error goes exponentially to 1. Hence, the capacity is a very clear dividing point—at rates below capacity, $P_e^{(n)} \to 0$ exponentially, and at rates above capacity, $P_e^{(n)} \to 1$ exponentially.

8.10 EQUALITY IN THE CONVERSE TO THE CHANNEL CODING THEOREM

We have proved the channel coding theorem and its converse. In essence, these theorems state that when $R < C$, it is possible to send information with an arbitrarily low probability of error, and when $R > C$, the probability of error is bounded away from zero.

It is interesting and rewarding to examine the consequences of equality in the converse; hopefully, it will give some ideas as to the kind of codes that achieve capacity. Repeating the steps of the converse in the case when $P_e = 0$, we have

$$nR = H(W) \tag{8.113}$$

$$\overset{(a)}{=} H(X^n(W)) \tag{8.114}$$

$$= \underbrace{H(X^n|Y^n)}_{=0,\ \text{since} P_e=0} + I(X^n; Y^n) \tag{8.115}$$

$$= I(X^n; Y^n) \tag{8.116}$$

$$= H(Y^n) - H(Y^n|X^n) \tag{8.117}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i) \quad \text{(since channel is a DMC)} \tag{8.118}$$

$$\overset{(b)}{\leq} \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.119}$$

$$= \sum_{i=1}^{n} I(X_i; Y_i) \tag{8.120}$$

$$\overset{(c)}{\leq} nC\,. \tag{8.121}$$

We have equality in (a) only if all the codewords are distinct. We have equality in (b) only if the Y_i's are independent, and equality in (c) only if the distribution of X_i is $p^*(x)$, the distribution on X that achieves capacity. We have equality in the converse only if these conditions are satisfied. This indicates that for an efficient code that achieves capacity, the codewords are distinct and the distribution of the Y_i's looks i.i.d. according to

$$p^*(y) = \sum_x p^*(x)p(y|x)\,, \tag{8.122}$$

the distribution on Y induced by the optimum distribution on X. The distribution referred to in the converse is the empirical distribution on X and Y induced by a uniform distribution over codewords, i.e.,

$$p(x_i, y_i) = \frac{1}{2^{nR}} \sum_{w=1}^{2^{nR}} I(X_i(w) = x_i)p(y_i|x_i)\,. \tag{8.123}$$

We can check this result in examples of codes which achieve capacity:

1. *Noisy typewriter.* In this case, we have an input alphabet of 26 letters, and each letter is either printed out correctly or changed to the next letter with probability $\frac{1}{2}$. A simple code that achieves capacity (log 13) for this channel is to use every alternate input letter so that no two letters can be confused. In this case, there are 13 codewords of blocklength 1. If we choose the codewords i.i.d. according to a uniform distribution on $\{1, 3, 5, 7, \ldots, 25\}$, then the output of the channel is also i.i.d. and uniformly distributed on $\{1, 2, \ldots, 26\}$, as expected.

2. *Binary symmetric channel.* Since given any input sequence, every possible output sequence has some positive probability, it will not be possible to distinguish even two codewords with zero probability of error. Hence the zero-error capacity of the BSC is zero.

   However, even in this case, we can draw some useful conclusions. The efficient codes will still induce a distribution on Y that looks i.i.d. $\sim$Bernoulli($\frac{1}{2}$). Also, from the arguments that lead up to the converse, we can see that at rates close to capacity, we have almost entirely covered the set of possible output sequences with decoding sets corresponding to the codewords. At rates above capacity, the decoding sets begin to overlap, and the probability of error can no longer be made arbitrarily small.

8.11 HAMMING CODES

The channel coding theorem promises the existence of block codes that will allow us to transmit information at rates below capacity with an arbitrarily small probability of error if the block length is large enough. Ever since the appearance of Shannon's original paper, people have searched for such codes. In addition to achieving low probabilities of error, useful codes should be "simple" so that they can be encoded and decoded efficiently.

The search for simple good codes has come a long way since the publication of Shannon's original paper in 1948. The entire field of coding theory has been developed during this search. We will not be able to describe the many elegant and intricate coding schemes that have been developed since 1948. We will only describe the simplest such scheme developed by Hamming [129]. It illustrates some of the basic ideas underlying most codes.

The object of coding is to introduce redundancy so that even if some of the information is lost or corrupted, it will still be possible to recover the message at the receiver. The most obvious coding scheme is to repeat information. For example, to send a 1, we send 11111, and to send a 0, we send 00000. This scheme uses 5 symbols to sent 1 bit, and therefore has a *rate* of $\frac{1}{5}$ bits per symbol. If this code is used on a binary symmetric channel, the optimum decoding scheme is to take the majority vote of each block of 5 received bits. If 3 or more bits are 1, we decode the block as a 1, otherwise we decode it as 0. An error occurs if and only if more than 3 of the bits are changed. By using longer repetition codes, we can achieve an arbitrarily low probability of error. But the rate of the code also goes to zero with block length, and so even though the code is "simple," it is really not a very useful code.

Instead of simply repeating the bits, we can combine the bits in some intelligent fashion so that each extra bit checks whether there is an

error in some subset of the information bits. A simple example of this is a parity check code. Starting with a block of $n-1$ information bits, we choose the n-th bit so that the parity of the entire block is 0 (the number of 1's in the block is even). Then if there is an odd number of errors during the transmission, the receiver will notice that the parity has changed and detect the error. This is the simplest example of an *error detecting code*. The code does not detect an even number of errors and does not give any information about how to correct the errors that occur.

We can extend the idea of parity checks to allow for more than one parity check bit and to allow the parity checks to depend on various subsets of the information bits. The Hamming code that we describe below is an example of a parity check code. We describe it using some simple ideas from linear algebra.

To illustrate the principles of Hamming codes, we consider a binary code of block length 7. All operations will be done modulo 2. Consider the set of all non-zero binary vectors of length 3. Arrange them in columns to form a matrix,

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}. \tag{8.124}$$

Consider the set of vectors of length 7 in the null space of H (the vectors which when multiplied by H give 000). From the theory of linear spaces, since H has rank 3, we expect the null space of H to have dimension 4. We list these 2^4 codewords in Table 8.1.

Since the set of codewords is the null-space of a matrix, it is *linear* in the sense that the sum of any two codewords is also a codeword. The set of codewords therefore forms a linear subspace of dimension 4 in the vector space of dimension 7.

Looking at the codewords, we notice that other than the all 0 codeword, the minimum number of 1's in any codeword is 3. This is called the *minimum weight* of the code. We can see that the minimum weight of a code has to be at least 3 since all the columns of H are different and so no two columns can add to 000. The fact that the minimum distance is exactly 3 can be seen from the fact that the sum of any two columns must be one of the columns of the matrix.

Table 8.1. The Hamming (7,4) Code

0000000	0100101	1000011	1100110
0001111	0101010	1001100	1101001
0010110	0110011	1010101	1110000
0011001	0111100	1011010	1111111

Since the code is linear, the difference between any two codewords is also a codeword, and hence any two codewords differ in at least 3 places. The minimum number of places in which two codewords differ is called the *minimum distance* of the code. The minimum distance of the code is a measure of how far apart the codewords are and will determine how distinguishable the codewords will be at the output of the channel. The minimum distance is equal to the minimum weight for a linear code. We aim to develop codes that have a large minimum distance.

For the code described above, the minimum distance is 3. Hence if a codeword $\mathbf{c}$ is corrupted in only one place, it will differ from any other codeword in at least two places, and therefore be closer to $\mathbf{c}$ than to any other codeword. But can we discover which is the closest codeword without searching over all the codewords?

The answer is yes. We can use the structure of the matrix H for decoding. The matrix H is called the *parity check matrix* and has the property that for every codeword $\mathbf{c}$, $H\mathbf{c} = \mathbf{0}$. Let $\mathbf{e}_i$ be a vector with a 1 in the ith position and 0's elsewhere. If the codeword is corrupted at position i, then the received vector $\mathbf{r} = \mathbf{c} + \mathbf{e}_i$. If we multiply this received vector by the matrix H, we obtain

$$H\mathbf{r} = H(\mathbf{c} + \mathbf{e}_i) = H\mathbf{c} + H\mathbf{e}_i = H\mathbf{e}_i , \tag{8.125}$$

which is the vector corresponding to the ith column of H. Hence looking at $H\mathbf{r}$, we can find which position of the received vector was corrupted. Reversing this bit will give us a codeword.

This yields a simple procedure for correcting one error in the received sequence. We have constructed a codebook with 16 codewords of block length 7, which can correct up to one error. This code is called a *Hamming code*.

We have not yet identified a simple encoding procedure; we could use any mapping from a set of 16 messages into the codewords. But if we examine the first 4 bits of the codewords in the table, we observe that they cycle through all 2^4 combinations of 4 bits. Thus we could use these 4 bits to be the 4 bits of the message we want to send; the other 3 bits are then determined by the code. In general, it is possible to modify a linear code so that the mapping is explicit, so that the first k bits in each codeword represent the message, and the last $n - k$ bits are parity check bits. Such a code is called a *systematic code*. The code is often identified by its block length n, the number of information bits k and the minimum distance d. For example, the above code is called a $(7, 4, 3)$ Hamming code, i.e., $n = 7$, $k = 4$ and $d = 3$.

We can easily generalize this procedure to construct larger matrices H. In general, if we use l rows in H, then the code that we obtain will have block length $n = 2^l - 1$, $k = 2^l - l - 1$, and minimum distance 3. All these codes are called Hamming codes and can correct one error.

Hamming codes are the simplest examples of linear parity check codes. They demonstrate the principle that underlies the construction of other linear codes. But with large block lengths it is likely that there will be more than one error in the block. In the early 1950's, Reed and Solomon found a class of multiple error correcting codes for non-binary channels. In the late 1950's, Bose and Chaudhuri [42] and Hocquenghem [134] generalized the ideas of Hamming codes using Galois field theory to construct t-error correcting codes (called BCH codes) for any t. Since then various authors have developed other codes and also developed efficient decoding algorithms for these codes. With the advent of integrated circuits, it has become feasible to implement fairly complex codes in hardware and realize some of the error correcting performance promised by Shannon's channel capacity theorem. For example, all compact disc players include error correction circuitry based on two interleaved (32, 28, 5) and (28, 24, 5) Reed-Solomon codes that allow the decoder to correct bursts of up to 4000 errors.

All the codes described above are block codes—they map a block of information bits onto a channel codeword and there is no dependence on past information bits. It is also possible to design codes where each output block depends not only on the current input block, but also on some of the past inputs as well. A highly structured form of such a code is called a *convolutional code*. The theory of convolutional codes has developed considerably over the last 25 years. We will not go into the details, but refer the interested reader to textbooks on coding theory [41], [179].

Although there has been much progress in the design of good codes for the binary symmetric channel, it is still not possible to design codes that meet the bounds suggested by Shannon's channel capacity theorem. For a binary symmetric channel with crossover probability p, we would need a code that could correct up to np errors in a block of length n and have $n(1 - H(p))$ information bits. None of the codes known so far achieve this performance. For example, the repetition code suggested earlier corrects up to $n/2$ errors in a block of length n, but its rate goes to 0 with n. Until 1972, all known codes that could correct $n\alpha$ errors for block length n had asymptotic rate 0. In 1972, Justesen [147] described a class of codes with positive asymptotic rate and positive asymptotic minimum distance as a fraction of the block length. However, these codes are good only for long block lengths.

8.12 FEEDBACK CAPACITY

The channel with feedback is illustrated in Figure 8.11. We assume that all the received symbols are sent back immediately and noiselessly to the transmitter, which can then use them to decide which symbol to send next.

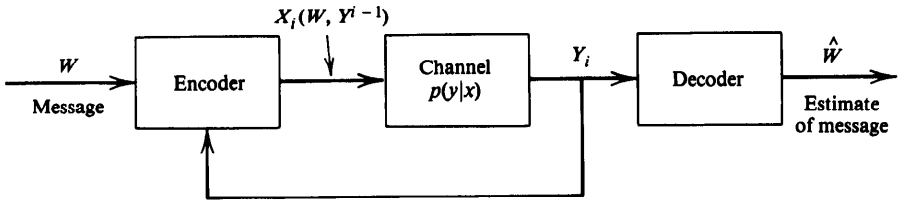


Figure 8.11. Discrete memoryless channel with feedback.

Can we do better with feedback? The surprising answer is no, which we shall now prove. We define a $(2^{nR}, n)$ *feedback code* as a sequence of mappings $x_i(W, Y^{i-1})$, where each x_i is a function only of W and the previous received values, $Y_1, Y_2, \ldots, Y_{i-1}$, and a sequence of decoding functions $g: \mathcal{Y}^n \to \{1, 2, \ldots, 2^{nR}\}$. Thus

$$P_e^{(n)} = \Pr\{g(Y^n) \neq W\}, \tag{8.126}$$

when W is uniformly distributed over $\{1, 2, \ldots, 2^{nR}\}$.

Definition: The *capacity with feedback*, C_{FB}, of a discrete memoryless channel is the supremum of all rates achievable by feedback codes.

Theorem 8.12.1 (*Feedback capacity*):

$$C_{FB} = C = \max_{p(x)} I(X; Y). \tag{8.127}$$

Proof: Since a non-feedback code is a special case of a feedback code, any rate that can be achieved without feedback can be achieved with feedback, and hence

$$C_{FB} \geq C. \tag{8.128}$$

Proving the inequality the other way is slightly more tricky. We cannot use the same proof that we used for the converse to the coding theorem without feedback. Lemma 8.9.2 is no longer true, since X_i depends on the past received symbols, and it is no longer true that Y_i depends only on X_i and is conditionally independent of the future X's in (8.101).

There is a simple change that will make the method work; instead of using X^n, we will use the index W and prove a similar series of inequalities. Let W be uniformly distributed over $\{1, 2, \ldots, 2^{nR}\}$. Then

$$nR = H(W) = H(W|Y^n) + I(W; Y^n) \tag{8.129}$$

$$\leq 1 + P_e^{(n)} nR + I(W; Y^n) \tag{8.130}$$

by Fano's inequality.

Now we can bound $I(W; Y^n)$ as follows:

$$I(W; Y^n) = H(Y^n) - H(Y^n|W) \tag{8.131}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|Y_1, Y_2, \ldots, Y_{i-1}, W) \tag{8.132}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|Y_1, Y_2, \ldots, Y_{i-1}, W, X_i) \tag{8.133}$$

$$= H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.134}$$

since X_i is a function of $Y_1, \ldots, Y_{i-1}$ and W; and conditional on X_i, Y_i is independent of W and past samples of Y. Then using the entropy bound, we have

$$I(W; Y^n) = H(Y^n) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.135}$$

$$\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|X_i) \tag{8.136}$$

$$= \sum_{i=1}^{n} I(X_i; Y_i) \tag{8.137}$$

$$\leq nC \tag{8.138}$$

from the definition of capacity for a discrete memoryless channel.

Putting these together, we obtain

$$nR \leq P_e^{(n)} nR + 1 + nC\,, \tag{8.139}$$

and dividing by n and letting $n \to \infty$, we conclude

$$R \leq C\,. \tag{8.140}$$

Thus we cannot achieve any higher rates with feedback than we can without feedback, and

$$C_{FB} = C\,. \quad \square \tag{8.141}$$

As we have seen in the example of the binary erasure channel, feedback can help enormously in simplifying encoding and decoding. However, it cannot increase the capacity of the channel.

8.13 THE JOINT SOURCE CHANNEL CODING THEOREM

It is now time to combine the two main results that we have proved so far: data compression ($R > H$: Theorem 5.4.2) and data transmission ($R < C$: Theorem 8.7.1). Is the condition $H < C$ necessary and sufficient for sending a source over a channel?

For example, consider sending digitized speech or music over a discrete memoryless channel. We could design a code to map the sequence of speech samples directly into the input of the channel, or we could compress the speech into its most efficient representation, then use the appropriate channel code to send it over the channel. It is not immediately clear that we are not losing something by using the two-stage method, since the data compression does not depend on the channel and the channel coding does not depend on the source distribution.

We will prove in this section that the two-stage method is as good as any other method of transmitting information over a noisy channel. This result has some important practical implications. It implies that we can consider the design of a communication system as a combination of two parts, source coding and channel coding. We can design source codes for the most efficient representation of the data. We can separately and independently design channel codes appropriate for the channel. The combination will be as efficient as anything we could design by considering both problems together.

The common representation for random data uses a binary alphabet. Most modern communication systems are digital, and data is reduced to a binary representation for transmission over the common channel. This offers an enormous reduction in complexity. A system like ISDN (Integrated Services Digital Network) uses the common binary representation to allow speech and digital data to use the same communication channel.

The result that a two-stage process is as good as any one stage process seems so obvious that it may be appropriate to point out that it is not always true. There are examples of multiuser channels where the decomposition breaks down.

We will also consider two simple situations where the theorem appears to be misleading. A simple example is that of sending English text over an erasure channel. We can look for the most efficient binary representation of the text and send it over the channel. But the errors will be very difficult to decode. If however we send the English text directly over the channel, we can lose up to about half the letters and yet be able to make sense out of the message. Similarly, the human ear has some unusual properties that enable it to distinguish speech under very high noise levels if the noise is white. In such cases, it may be appropriate to send the uncompressed speech over the noisy channel

rather than the compressed version. Apparently the redundancy in the source is suited to the channel.

Let us define the setup under consideration. We have a source V, that generates symbols from an alphabet $\mathcal{V}$. We will not make any assumptions about the kind of stochastic process produced by V other than that it is from a finite alphabet and satisfies the AEP. Examples of such processes include a sequence of i.i.d. random variables and the sequence of states of a stationary irreducible Markov chain. Any stationary ergodic source satisfies the AEP, as will be shown in Section 15.7.

We want to send the sequence of symbols $V^n = V_1, V_2, \ldots, V_n$ over the channel so that the receiver can reconstruct the sequence. To do this, we map the sequence onto a codeword $X^n(V^n)$ and send the codeword over the channel. The receiver looks at his received sequence Y^n and makes an estimate $\hat{V}^n$ of the sequence V^n that was sent. The receiver makes an error if $V^n \neq \hat{V}^n$. We define the probability of error $P_e^{(n)}$ as

$$P_e^{(n)} = \Pr(V^n \neq \hat{V}^n) = \sum_{y^n} \sum_{v^n} p(v^n)p(y^n|x^n(v^n))I(g(y^n) \neq v^n), \tag{8.142}$$

where I is the indicator function and $g(y^n)$ is the decoding function. The system is illustrated in Figure 8.12.

We can now state the joint source channel coding theorem:

Theorem 8.13.1 (*Source-channel coding theorem*): *If $V_1, V_2, \ldots, V_n$ is a finite alphabet stochastic process that satisfies the AEP, then there exists a source channel code with $P_e^{(n)} \to 0$ if $H(\mathcal{V}) < C$.*

Conversely, for any stationary stochastic process, if $H(\mathcal{V}) > C$, the probability of error is bounded away from zero, and it is not possible to send the process over the channel with arbitrarily low probability of error.

Proof:

Achievability: The essence of the forward part of the proof is the two-stage encoding described earlier. Since we have assumed that the stochastic process satisfies the AEP, it implies that there exists a typical set $A_\epsilon^{(n)}$ of size $\leq 2^{n(H(\mathcal{V})+\epsilon)}$ which contains most of the probability. We

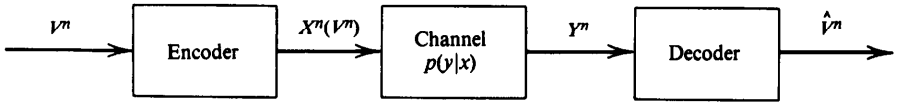


Figure 8.12. Joint source and channel coding.

will encode only the source sequences belonging to the typical set; all other sequences will result in an error. This will contribute at most ϵ to the probability of error.

We index all the sequences belonging to $A_\epsilon^{(n)}$. Since there are at most $2^{n(H+\epsilon)}$ such sequences, $n(H+\epsilon)$ bits suffice to index them. We can transmit the desired index to the receiver with probability of error less than ϵ if

$$H(\mathcal{V}) + \epsilon = R < C . \tag{8.143}$$

The receiver can reconstruct V^n by enumerating the typical set $A_\epsilon^{(n)}$ and choosing the sequence corresponding to the estimated index. This sequence will agree with the transmitted sequence with high probability. To be precise,

$$P_e^{(n)} = P(V^n \neq \hat{V}^n) \tag{8.144}$$

$$\leq P(V^n \notin A_\epsilon^{(n)}) + P(g(Y^n) \neq V^n | V^n \in A_\epsilon^{(n)}) \tag{8.145}$$

$$\leq \epsilon + \epsilon = 2\epsilon \tag{8.146}$$

for n sufficiently large. Hence, we can reconstruct the sequence with low probability of error for n sufficiently large if

$$H(\mathcal{V}) < C . \tag{8.147}$$

Converse: We wish to show that $P_e^{(n)} \to 0$ implies that $H(\mathcal{V}) \leq C$ for any sequence of source-channel codes

$$X^n(V^n): \mathcal{V}^n \to \mathcal{X}^n , \tag{8.148}$$

$$g_n(Y^n): \mathcal{Y}^n \to \mathcal{V}^n . \tag{8.149}$$

By Fano's inequality, we must have

$$H(V^n | \hat{V}^n) \leq 1 + P_e^{(n)} \log |\mathcal{V}^n| = 1 + P_e^{(n)} n \log |\mathcal{V}| . \tag{8.150}$$

Hence for the code,

$$H(\mathcal{V}) \overset{(a)}{\leq} \frac{H(V_1, V_2, \ldots, V_n)}{n} \tag{8.151}$$

$$= \frac{H(V^n)}{n} \tag{8.152}$$

$$= \frac{1}{n} H(V^n | \hat{V}^n) + \frac{1}{n} I(V^n; \hat{V}^n) \tag{8.153}$$

$$\overset{(b)}{\leq} \frac{1}{n}(1 + P_e^{(n)} n \log|\mathcal{V}|) + \frac{1}{n} I(V^n; \hat{V}^n) \tag{8.154}$$

$$\overset{(c)}{\leq} \frac{1}{n}(1 + P_e^{(n)} n \log|\mathcal{V}|) + \frac{1}{n} I(X^n; Y^n) \tag{8.155}$$

$$\overset{(d)}{\leq} \frac{1}{n} + P_e^{(n)} \log|\mathcal{V}| + C, \tag{8.156}$$

where (a) follows from the definition of entropy rate of a stationary process, (b) follows from Fano's inequality, (c) from the data processing inequality (since $V^n \to X^n \to Y^n \to \hat{V}^n$ forms a Markov chain) and (d) from the memorylessness of the channel. Now letting $n \to \infty$, we have $P_e^{(n)} \to 0$ and hence

$$H(\mathcal{V}) \leq C. \quad \square \tag{8.157}$$

Hence we can transmit a stationary ergodic source over a channel if and only if its entropy rate is less than the capacity of the channel.

With this result, we have tied together the two basic theorems of information theory: data compression and data transmission. We will try to summarize the proofs of the two results in a few words. The data compression theorem is a consequence of the AEP, which shows that there exists a "small" subset (of size 2^{nH}) of all possible source sequences that contain most of the probability and that we can therefore represent the source with a small probability of error using H bits per symbol. The data transmission theorem is based on the joint AEP; it uses the fact that for long block lengths, the output sequence of the channel is very likely to be jointly typical with the input codeword, while any other codeword is jointly typical with probability $\approx 2^{-nI}$. Hence we can use about 2^{nI} codewords and still have negligible probability of error. The source channel separation theorem shows that we can design the source code and the channel code separately and combine the results to achieve optimal performance.

SUMMARY OF CHAPTER 8

Information capacity: $C = \max_{p(x)} I(X; Y)$.

Examples:

- Binary symmetric channel: $C = 1 - H(p)$.
- Binary erasure channel: $C = 1 - \alpha$.
- Symmetric channel: $C = \log|\mathcal{Y}| - H$(row of transition matrix).

Properties of C:

1. $0 \le C \le \min\{\log|\mathcal{X}|, \log|\mathcal{Y}|\}$.
2. $I(X; Y)$ is a continuous concave function of $p(x)$.

Definition: The set $A_\epsilon^{(n)}$ of *jointly typical* sequences $\{(x^n, y^n)\}$ with respect to the distribution $p(x, y)$ is given by

$$A_\epsilon^{(n)} = \Big\{(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n: \tag{8.158}$$

$$\left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon, \tag{8.159}$$

$$\left| -\frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon, \tag{8.160}$$

$$\left| -\frac{1}{n} \log p(x^n, y^n) - H(X, Y) \right| < \epsilon \Big\}, \tag{8.161}$$

where $p(x^n, y^n) = \prod_{i=1}^n p(x_i, y_i)$.

Joint AEP: Let (X^n, Y^n) be sequences of length n drawn i.i.d. according to $p(x^n, y^n) = \prod_{i=1}^n p(x_i, y_i)$. Then

1. $\Pr((X^n, Y^n) \in A_\epsilon^{(n)}) \to 1$ as $n \to \infty$.
2. $|A_\epsilon^{(n)}| \le 2^{n(H(X,Y)+\epsilon)}$.
3. If $(\tilde{X}^n, \tilde{Y}^n) \sim p(x^n)p(y^n)$, then $\Pr((\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)}) \le 2^{-n(I(X;Y)-3\epsilon)}$.

The channel coding theorem: All rates below capacity C are achievable, that is, for every $\epsilon > 0$ and rate $R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error

$$\lambda^{(n)} \le \epsilon, \tag{8.162}$$

for n sufficiently large. Conversely, if $\lambda^{(n)} \to 0$, then $R \le C$.

Feedback capacity: Feedback does not increase capacity for discrete memoryless channels, i.e., $C_{FB} = C$.

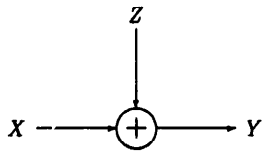
Source channel theorem: A stochastic process with entropy rate $H(\mathcal{V})$ cannot be sent reliably over a discrete memoryless channel if $H(\mathcal{V}) > C$. Conversely, if the process satisfies the AEP, then the source can be transmitted reliably if $H(\mathcal{V}) < C$.

PROBLEMS FOR CHAPTER 8

1. *Preprocessing the output.* One is given a communication channel with transition probabilities $p(y|x)$ and channel capacity $C = \max_{p(x)} I(X; Y)$. A helpful statistician preprocesses the output by forming $\tilde{Y} = g(Y)$. He claims that this will strictly improve the capacity.
   (a) Show that he is wrong.
   (b) Under what conditions does he not strictly decrease the capacity?

2. *Maximum likelihood decoding.* A source produces independent, equally probable symbols from an alphabet (a_1, a_2) at a rate of one symbol every 3 seconds. These symbols are transmitted over a binary symmetric channel which is used once each second by encoding the source symbol a_1 as 000 and the source symbol a_2 as 111. If in the corresponding 3 second interval of the channel output, any of the sequences 000,001,010,100 is received, a_1 is decoded; otherwise, a_2 is decoded. Let $\epsilon < \frac{1}{2}$ be the channel crossover probability.
   (a) For each possible received 3-bit sequence in the interval corresponding to a given source letter, find the probability that a_1 came out of the source given that received sequence.
   (b) Using part (a), show that the above decoding rule minimizes the probability of an incorrect decision.
   (c) Find the probability of an incorrect decision (using part (a) is not the easy way here).
   (d) If the source is slowed down to produce one letter every $2n + 1$ seconds, a_1 being encoded by $2n + 1$ 0's and a_2 being encoded by $2n + 1$ 1's. What decision rule minimizes the probability of error at the decoder? Find the probability of error as $n \to \infty$.

3. *An additive noise channel.* Find the channel capacity of the following discrete memoryless channel:



   where $\Pr\{Z = 0\} = \Pr\{Z = a\} = \frac{1}{2}$. The alphabet for x is $\mathcal{X} = \{0, 1\}$. Assume that Z is independent of X.
   Observe that the channel capacity depends on the value of a.

4. *Channels with memory have higher capacity.* Consider a binary symmetric channel with $Y_i = X_i \oplus Z_i$, where $\oplus$ is mod 2 addition, and $X_i, Y_i \in \{0, 1\}$.
   Suppose that $\{Z_i\}$ has constant marginal probabilities $\Pr\{Z_i = 1\} = p = 1 - \Pr\{Z_i = 0\}$, but that $Z_1, Z_2, \ldots, Z_n$ are not necessarily in-

dependent. Assume that Z^n is independent of the input X^n. Let $C = 1 - H(p, 1-p)$. Show that $\max_{p(x_1, x_2, \ldots, x_n)} I(X_1, X_2, \ldots, X_n; Y_1, Y_2, \ldots, Y_n) \geq nC$.

5. *Channel capacity.* Consider the discrete memoryless channel $Y = X + Z \pmod{11}$, where

$$Z = \begin{pmatrix} 1, & 2, & 3 \\ 1/3, & 1/3, & 1/3 \end{pmatrix}$$

and $X \in \{0, 1, \ldots, 10\}$. Assume that Z is independent of X.

(a) Find the capacity.

(b) What is the maximizing $p^*(x)$?

6. *Using two channels at once.* Consider two discrete memoryless channels $(\mathcal{X}_1, p(y_1|x_1), \mathcal{Y}_1)$ and $(\mathcal{X}_2, p(y_2|x_2), \mathcal{Y}_2)$ with capacities C_1 and C_2 respectively. A new channel $(\mathcal{X}_1 \times \mathcal{X}_2, p(y_1|x_1) \times p(y_2|x_2), \mathcal{Y}_1 \times \mathcal{Y}_2)$ is formed in which $x_1 \in \mathcal{X}_1$ and $x_2 \in \mathcal{X}_2$, are simultaneously sent, resulting in y_1, y_2. Find the capacity of this channel.

7. *Noisy typewriter.* Consider a 26-key typewriter.

(a) If pushing a key results in printing the associated letter, what is the capacity C in bits?

(b) Now suppose that pushing a key results in printing that letter or the next (with equal probability). Thus $A \to A$ or $B, \ldots, Z \to Z$ or A. What is the capacity?

(c) What is the highest rate code with block length one that you can find that achieves *zero* probability of error for the channel in part (b).

8. *Cascade of binary symmetric channels.* Show that a cascade of n identical binary symmetric channels,

$$X_0 \to \boxed{\text{BSC\#1}} \to X_1 \to \cdots \to X_{n-1} \to \boxed{\text{BSC\#}n} \to X_n$$

each with raw error probability p, is equivalent to a single BSC with error probability $\frac{1}{2}(1 - (1-2p)^n)$ and hence that $\lim_{n\to\infty} I(X_0; X_n) = 0$ if $p \neq 0, 1$. No encoding or decoding takes place at the intermediate terminals $X_1, \ldots, X_{n-1}$. Thus the capacity of the cascade tends to zero.

9. *The Z channel.* The Z channel has binary input and output alphabets and transition probabilities $p(y|x)$ given by the following matrix:

$$Q = \begin{bmatrix} 1 & 0 \\ 1/2 & 1/2 \end{bmatrix} \qquad x, y \in \{0, 1\}$$

Find the capacity of the Z channel and the maximizing input probability distribution.

10. *Suboptimal codes.* For the Z channel of the previous problem, assume that we choose a $(2^{nR}, n)$ code at random, where each codeword is a

sequence of *fair* coin tosses. This will not achieve capacity. Find the maximum rate R such that the probability of error $P_e^{(n)}$, averaged over the randomly generated codes, tends to zero as the block length n tends to infinity.

11. *Zero-error capacity.* A channel with alphabet $\{0, 1, 2, 3, 4\}$ has transition probabilities of the form

$$p(y|x) = \begin{cases} 1/2 & \text{if } y = x \pm 1 \bmod 5 \\ 0 & \text{otherwise.} \end{cases}$$

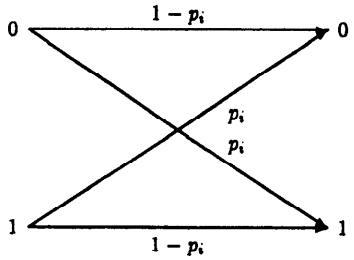
(a) Compute the capacity of this channel in bits.

(b) The zero-error capacity of a channel is the number of bits per channel use that can be transmitted with zero probability of error. Clearly, the zero-error capacity of this pentagonal channel is at least 1 bit (transmit 0 or 1 with probability 1/2). Find a block code that shows that the zero-error capacity is greater than 1 bit. Can you estimate the exact value of the zero-error capacity?

(*Hint*: Consider codes of length 2 for this channel.)

The zero-error capacity of this channel was found by Lovasz [182].

12. *Time-varying channels.* Consider a time-varying discrete *memoryless* channel. Let $Y_1, Y_2, \ldots, Y_n$ be conditionally independent given $X_1, X_2, \ldots, X_n$, with conditional distribution given by $p(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{n} p_i(y_i|x_i)$.



Let $\mathbf{X} = (X_1, X_2, \ldots, X_n)$, $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)$. Find $\max_{p(\mathbf{x})} I(\mathbf{X}; \mathbf{Y})$.

HISTORICAL NOTES

The idea of mutual information and its relationship to channel capacity was first developed by Shannon in his original paper [238]. In this paper, he stated the channel capacity theorem and outlined the proof using typical sequences in an argument similar to the one described here. The first rigorous proof was due to Feinstein [107], who used a painstaking "cookie-cutting" argument to find the number of codewords that can be sent with a low probability of error. A simpler proof using a random coding exponent was developed by Gallager [118]. Our proof is based on Cover [62] and on Forney's unpublished course notes [115].

The converse was proved by Fano [105], who used the inequality bearing his name. The strong converse was first proved by Wolfowitz [276], using techniques that are closely related to typical sequences. An iterative algorithm to calculate the channel capacity was developed independently by Arimoto [11] and Blahut [37].

The idea of the zero-error capacity was developed by Shannon [239]; in the same paper, he also proved that feedback does not increase the capacity of a discrete memoryless channel. The problem of finding the zero-error capacity is essentially combinatorial; the first important result in this area is due to Lovasz [182].