Low-Density Parity-Check Codes

Robert G. Gallager

1963

Preface

The Noisy Channel Coding Theorem discovered by C. E. Shannon in 1948 offered communication engineers the possibility of reducing error rates on noisy channels to negligible levels without sacrificing data rates. The primary obstacle to the practical use of this theorem has been the equipment complexity and the computation time required to decode the noisy received data.

This monograph presents a technique for achieving high data rates and negligible error probabilities on noisy channels with a reasonable amount of equipment. The advantages and disadvantages of this technique over other techniques for the same purpose are neither simple nor clear-cut, and depend primarily upon the channel and the type of service required. More important than the particular technique, however, is the hope that the concepts here will lead to new and better coding procedures.

The chapters of the monograph are arranged in such a way that with the exception of Chapter 5 each chapter can be read independently of the others. Chapter 1 sets the background of the study, summarizes the results, and briefly compares low-density coding with other coding schemes. Chapter 2 analyzes the distances between code words in low-density codes and Chapter 3 applies these results to the problem of bounding the probability of decoding error that can be achieved for these codes on a broad class of binary-input channels. The results of Chapter 3 can be immediately applied to any code or class of codes for which the distance properties can be bounded. Chapter 4 presents a simple decoding algorithm for these codes and analyzes the resulting error probability. Chapter 5 briefly extends all the previous results to multi-input channels, and Chapter 6 presents the results of computer simulation of the low-density decoding algorithm.

The work reported here is an expanded and revised version of my doctoral dissertation, completed in 1960 in the Department of Electrical Engineering, M.I.T. I am grateful to my thesis supervisor, Professor Peter Elias, and to my thesis readers, Professors Robert M. Fano and John M. Wozencraft, for assistance and encouragement both during the course of the thesis and later.

This research was made possible in part by support extended by the Research Laboratory of Electronics of the Massachusetts Institute of Technology, which is supported in part by the U.S. Army, the Air Force Office of Scientific Research, and the Office of Naval Research; additional support was received through the National Science Foundation (Grant G-16526) and the National Institute of Health (Grant MH-04737-03).

Much of Chapter 4 is reprinted with permission of the editors from an article by the author in the Transactions of the I.R.E., IT-9, pages 21 to 28.

The experimental results in Chapter 6 were obtained in part through the support of the Rome Air Development Center and in part through the support of the M.I.T. Computation Center.

Cambridge, Mass.
July, 1963

Robert G. Gallager

Contents

1 Introduction

1.1 Coding for Digital Data Transmission

The need for efficient and reliable digital data communication systems has been rising rapidly in recent years. This need has been brought on by a variety of reasons, among them being the increase in automatic data processing equipment and the increased need for long range communication. Attempts to develop data systems through the use of conventional modulation and voice transmission techniques have generally resulted in systems with relatively low date rates and high error probabilities.

A more fundamental approach to the problems of efficiency and reliability in communication systems is contained in the Noisy Channel Coding theorem developed by C. E. Shannon [15, 4] in 1948. In order to understand the meaning of this theorem, consider Figure 1.1. The source produces binary digits, or binits, at some fixed time rate R_t. The encoder is a device that performs data processing, modulation, and anything else that might be necessary to prepare the data for transmission over the channel. We shall assume, however, that the encoder separates the source sequence into blocks of ν binits and operates on only one block at a time. The encoder output is then transmitted over the channel and changed by some sort of random disturbance or noise. The decoder processes the channel output and produces a delayed replica of the source binits. The coding theorem states that for a large variety of channel models, encoders and decoders exist such that the probability of the decoder reproducing a source binit in error P_e is bounded by

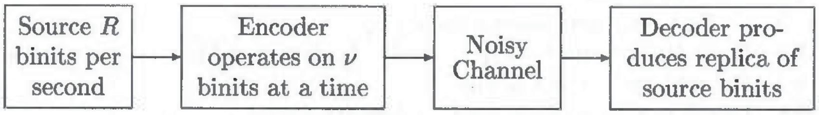

Source R binits per second
Encoder operates on ν binits at a time
Noisy Channel
Decoder produces replica of source binits

Figure 1.1: Block diagram of a communication system.

$$e^{-\nu[E_L(R_t)+0(\nu)]} \le P_e \le e^{-\nu E(R_t)}$$

The functions $E(R_t)$ and $E_L(R_t)$ depend upon the channel but not upon ν; they are positive when $R_t = 0$, and decrease with R_t until they become 0 at some time rate C_t known as the channel capacity. The exact nature of these functions and the particular class of channels for which this theorem has been proved need not concern us here. The important result is that the coding constraint length ν is a fundamental parameter of a communication system. If a channel is to be used efficiently, that is with R_t close to C_t, then ν must be made correspondingly large to achieve a satisfactory error probability.

The obvious response of an engineer to such a theorem is: "Splendid, but how does one build encoders and decoders that behave in this way when ν is large?" It is rather sobering to observe that if an encoder stores a waveform or code word for each possible block of ν binits, then the storage requirement must be proportional to 2^ν, which is obviously impractical when ν is large. Fortunately, Elias [3] and Reiffen [14] have proved that for a wide variety of channel models, the results of the Noisy Channel Coding theorem can be achieved with little equipment complexity at the encoder by the use of parity-check coding. This will be described in more detail later.

Unfortunately, the problem of decoding simply but effectively when ν is large appears to be much more difficult than the problem of encoding. Enough approaches to this problem have been developed to assure one that the Coding theorem has engineering importance. On the other hand these approaches have not been carried far enough for the design of an efficient, reliable data communication system to become a matter of routine engineering.

This monograph contains a detailed study of one of the three or four most promising approaches to simple decoding for long constraint length codes. The purpose of publishing this work is primarily to show how such a coding and decoding scheme would work and where it might be useful. Also, naturally, it is hoped that this will stimulate further research on the subject. Further mathematical analysis will probably be fruitless, but there are many interesting modifications of the scheme that might be made and much experimental work that should be done.

In order the prove mathematically some results about low-density parity-check codes, we shall assume that the codes are to be used on a somewhat restricted and idealized class of channels. It is obvious that results using such channel models can be applied only to channels that closely approximate the model. However, when studying the probability of decoding error, we are interested primarily in the extremely atypical events that cause errors. It is not easy to find models that approximate both these atypical events and typical events. Consequently the analysis of codes on idealized channels can provide only limited insight about real channels, and such insight should be used with caution.

The channel models to be considered here are called symmetric binary-input channels. By this we mean a time-discrete channel for which the input is a sequence of the binary digits 0 and 1 and the output is a corresponding sequence of letters from a discrete or continuous alphabet. The channel is memoryless in the sense that given the input at a given time, the output at the corresponding time is statistically independent of all other inputs and outputs. The symmetry requirement will be defined precisely in Chapter 3, but roughly it means that the outputs can be paired in such a way that the probability of one output given an input is the same as that of the other output of the pair given the other input. The binary symmetric channel, abbreviated BSC, is a member of this class of channels in which there are only two output symbols, one corresponding to each input. The BSC can be entirely specified by the probability of a crossover from one input to the other output.

If a symmetric binary-input channel were to be used without coding, a sequence of binary digits would be transmitted through the channel and the receiver would guess the transmitted symbols one at a time from the received symbols. If coding were to be used, however, the coder would first take sequences of binary digits carrying the information from the source and would map these sequences into longer redundant sequences, called code words, for transmission over the channel. We define the rate R of such codes to be the ratio of the length of the information sequence to the length of the code word sequence. If the code words are of length n, then there are 2^{nR} possible sequences from the source that are mapped into n-length code words. Thus only a fraction $2^{-n(1-R)}$ of the 2^n different n-length sequences can be used as code words.

At the receiver, the decoder, with its knowledge of which sequences are code words, can separate the transmitted n-length code word from the channel noise. Thus the code word is mapped back into the nR information digits. Many decoding schemes find the transmitted code word by first making a decision on each received digit and then using a knowledge of the code words to correct the errors. This intermediate decision, however, destroys a considerable amount of information about the transmitted message, as discussed in detail for several channels in Reference [1]. The decoding scheme to be described here avoids this intermediate decision and operates directly with the *a posteriori* probabilities of the input symbols conditional on the corresponding received symbols.

The codes to be discussed here are special examples of parity-check codes[1]. The code words of a parity-check code are formed by combining a block of binary-information digits with a block of check digits. Each check digit is the modulo 2 sum[2] of a pre-specified set of information digits. These formation rules for the check digits can be represented conveniently by a parity-check matrix, as in Figure 1.2. This matrix represents a set of linear homogeneous modulo 2 equations called parity-check equations, and the set of code words is the set of solutions of these equations. We call the set of digits contained in a parity-check equation a parity-check set. For example, the first parity-check set in Figure 1.2 is the set of digits $(1, 2, 3, 5)$.

$$\begin{array}{c|ccccccc|}
 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ \hline
 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
n(1-R) & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\
 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ \hline
\end{array}
\qquad
\begin{array}{l}
x_5 = x_1 + x_2 + x_3 \\
x_6 = x_1 + x_2 + x_4 \\
x_7 = x_1 + x_3 + x_4
\end{array}$$

Figure 1.2: Example of a parity-check matrix.

The use of-parity check codes makes coding (as distinguished from decoding) relatively simple to implement. Also, as Elias [3] has shown, if a typical parity-check code of long block length is used on a BSC, and if the code rate is between critical rate and channel capacity, then the probability of decoding error will be almost as small as that for the best possible code of that rate and block length.

[1] For a more detailed discussion of parity-check codes, see Peterson [12].

[2] The modulo 2 sum is 1 if the ordinary sum is odd and 0 if the ordinary sum is even.

Unfortunately, the decoding of parity-check codes is not inherently simple to implement; thus we must look for special classes of parity-check codes, such as described in Section 1.2, for which reasonable decoding procedures exist.

1.2 Low-Density Parity-Check Codes

Low-density parity-check codes are codes specified by a matrix containing mostly 0's and relatively few 1's. In particular, an (n, j, k) low-density code is a code of block length n with a matrix like that of Figure 2.1, where each column contains a small fixed number j of 1's and each row contains a small fixed number k of 1's. Note that this type of matrix does not have the check digits appearing in diagonal form as do those in Figure 1.2. However, for coding purposes, the equations represented by these matrices can always be solved to give the check digits as explicit sums of information digits.

Low density codes are not optimum in the somewhat artificial sense of minimizing the probability of decoding error for a given block length, and it can be shown that the maximum rate at which they can be used is bounded below channel capacity. However, the existence of a simple decoding scheme more than compensates for these disadvantages.

1.3 Summary of Results

An ensemble of (n, j, k) codes will be formed in Chapter 2, and this ensemble will be used to analyze the distance properties of (n, j, k) codes. The distance between two words in a code is simply the number of digits in which they differ. Clearly an important parameter in a code is the set of distances separating one code word from all the other code words. In a parity-check code, it can be shown that all code words have the same set of distances to the other code words [12]. Thus the distance properties for the ensemble can be summarized by the typical number of code words at each distance from the all-zero code word. It is found that the typical (n, j, k) code for $j \geq 3$ has a minimum distance that increases linearly with the block length for j and k constant. Figure 2.4 plots the ratio of minimum distance to block length for several values of j and k and compares the ratio with the same ratio for ordinary parity-check codes. The (n, j, k) codes with $j = 2$ exhibit markedly different behavior, and it is shown that the minimum distance of an $(n, 2, k)$ code can increase at most logarithmically with the block length.

In Chapter 3, a general upper bound on the probability of decoding error for symmetric binary-input channels with maximum-likelihood decoding is derived for both individual codes and for arbitrary ensembles of codes. The bound is a function of the code only through its distance properties. The assumption of maximum-likelihood decoding is made partly for analytic convenience and partly so as to be able to evaluate codes independently of their decoding algorithms. Any practical decoding algorithm, such as that described in Chapter 4, involves a trade-off between error probability and simplicity; the maximum-likelihood decoder minimizes the error probability but is totally impractical if the block length is large.

It is shown in Chapter 3 that if the distance properties of the code are exponentially related to the block length, and if the code rate is sufficiently low, then the bound to $P(e)$ is an exponentially decreasing function of the block length. For the appropriate ensemble of codes, these bounds reduce to the usual random coding bounds [3, 4].

For the special case of the binary symmetric channel, a particularly simple bound to $P(e)$ is found; this is used to show that over a range of channel crossover probabilities, a typical low-density code has the same error behavior as the optimum code of a slightly higher rate. Figure 3.5 illustrates this loss of effective rate associated with low-density codes.

In Chapter 4, two decoding schemes are described. In the first, which is particularly simple, the decoder first makes a decision on each digit, then computes the parity checks and changes any digit that is contained in more than some fixed number of unsatisfied parity-check equations. The process is repeated, each time using the changed digits, until the sequence is decoded. The second decoding scheme is based on a procedure for computing the conditional probability that an input symbol is 1; this is conditioned on all the received symbols that are in any of the parity-check sets containing the digit in question. Once again, the procedure is iterated until the sequence is decoded. The computation per digit per iteration in each scheme is independent of the code length. The probabilistic, or second scheme, entails slightly more computation than the first scheme, but decodes with a lower error probability.

A mathematical analysis of the probability of decoding error using probabilistic decoding is difficult because of statistical dependencies. However, for a BSC with sufficiently small cross-over probabilities and for codes with $j \geq 4$, a very weak upper bound the probability of error is derived that decreases exponentially with a root of the code length. Figure 3.5 plots cross-over probabilities for which the probability of decoding error is guaranteed to approach 0 with increasing code length. It is hypothesized that the probability of decoding error actually decreases exponentially with block length, while the number of iterations necessary to decode increases logarithmically.

Chapter 5 extends all the major results of Chapters 2, 3, and 4 to nonbinary low-density parity-check codes. Although the theory generalizes in a very natural way, the expressions for minimum distance, error probability, and probabilistic decoding performance error are sufficiently complicated that little insight is gained into the advantages or disadvantages of a multilevel system over a binary system. Some experimental work would be helpful here in evaluating these codes.

Some experimental results for binary low-density codes are presented in Chapter 6. An IBM 7090 computer was used to simulate both probabilistic decoding and the noise generated by several different types of channels. Due to limitation on computer time, the only situations investigated were those in which the channel was sufficiently noisy to yield a probability of decoding error greater than 10^{-4}. The most spectacular data from these experiments are given in Figure 6.8, which emphasizes the advantages of a decoding scheme that operates from a likelihood receiver instead of a decision receiver.

1.4 Comparison with Other Schemes

Some other coding and decoding schemes that appear extremely promising for achieving low error probabilities and high data rates at reasonable cost are the following: first, convolutional codes [3] with sequential decoding as developed by Wozencraft [17], Fano [5], and Reiffen [14]; second, convolutional codes with Massey's threshold decoding [10]; and third, the Bose-Chaudhuri codes [2] with the decoding schemes developed by Peterson [12] and Zierler and Gorenstein [18].

It has been shown by Fano [5] that for arbitrary discrete memoryless channels, sequential decoding has a probability of decoding error that is upper bounded by a function of the form $e^{-\alpha n}$. Here n is the constraint length of the code and α is a function of both the channel and the code; α is positive for rates below channel capacity C. Fano also shows that for rates below a certain quantity called R_{comp}, where $R_{\text{comp}} < C$, the average amount of computation in decoding a digit is bounded by a quantity independent of constraint length.

An experimental sequential decoder has been built at Lincoln Laboratories, Lexington, Massachusetts [11]. By using this decoder in a system with a feedback link and an appropriately designed modulator and demodulator, reliable transmission has been achieved experimentally [9] over a telephone circuit at about 7500 bits per second rather than the 1200 or 2400 bits per second possible without coding.

The two principal weaknesses of sequential decoding are as follows: First, the amount of computation required per digit is a random variable, and this creates a waiting line problem at the decoder; second, if the decoder once makes an error, a large block of errors can be made before the decoder gets back on the proper track. If a feedback link is available, these problems are not serious, but considerably more study is required for cases in which no feedback exists.

Threshold decoding is the simplest scheme to implement that is discussed here; it involves only shift registers, a few binary adders, and a threshold device. It is most effective at relatively short constraint lengths, and has a somewhat higher error probability and less flexibility than sequential decoding.

The computation per digit associated with the Bose-Chaudhuri codes on the BSC increases roughly as the cube of the block length but does not fluctuate widely. The decoding scheme guarantees correction of all combinations of up to some fixed number of errors and corrects nothing beyond. For moderately long block lengths, this restriction in the decoding procedure causes a large increase in P_e. No way is known to make use of the *a posteriori* probabilities at the output of more general binary input channels. This inability to make use of *a posteriori* probabilities appears to be a characteristic limitation of algebraic as opposed to probabilistic decoding techniques.

The computation per digit associated with low-density parity-check codes appears to increase at most logarithmically with block length and not to fluctuate widely with the noise. The probability of decoding error is unknown, but is believed to decrease exponentially with block length at a reasonable rate. The ability to decode the digits of a block in parallel makes it possible to handle higher data rates than is possible with other schemes.

For many channels with memory, retaining the *a posteriori* probabilities from the channel makes it practically unnecessary to take account of the memory in any other way. For instance, on a fading channel when the fade persists for several baud lengths, the *a posteriori* probabilities will indicate the presence of a fade. If this channel were used as a BSC however, it would be necessary for the decoder to account for the fact that bursts of errors are more probable than isolated errors. Then, using *a posteriori* probabilities gives low-density decoding and sequential decoding a great flexibility in handling channels with dependent noise. For channels in which the noise is rigidly constrained to occur in short, severe bursts, on the other hand, there is a particularly simple procedure for decoding the Bose-Chaudhuri codes [12].

When transmitting over channels subject to long fades or long noise bursts, it is often impractical to correct errors in these noisy periods. In such cases it is advantageous to use a combination of error correction and error detection with feedback and retransmission [16]. All of the coding and decoding schemes being considered here fit naturally into such a system, but in cases where little or no error correction is attempted, low-density codes appear at a disadvantage.

In conclusion, all these schemes have their own advantages, and clearly no scheme is optimum for all communication situations. It appears that enough coding and decoding alternatives now exist for serious consideration of the use of coding on particular channels.

2 Distance Functions

The distance function of a parity check-code $N(\ell)$ is defined as the number of code words in the code of weight ℓ. From the group properties of a parity-check code, it easily follows [12] that $N(\ell)$ is also the number of code words at distance ℓ from any given code word. The minimum distance D of a code is then defined as the smallest value of $\ell > 0$ for which $N(\ell) \neq 0$. Clearly, in a code of given block length n and rate R it is desirable to make D as large as possible and to make $N(\ell)$ as small as possible for those ℓ just larger than D. However, the next chapter, which discusses bounding the probability of decoding error for symmetric binary-input channels, will make the exact effect of $N(\ell)$ on error-correcting capability clearer.

For a parity-check code of long block length it is usually impractical to calculate exactly the distance function or even the minimum distance because of the enormous number of code words involved. It is often simpler to analyze the average distance function of an ensemble of codes; the statistics of an ensemble permit one to average over quantities that are not tractable in individual codes. From the ensemble average, one can then make statistical statements about the member codes.

2.1 Equiprobable Ensemble of Parity-Check Codes

This chapter will be concerned primarily with the distance functions of low-density parity-check codes, but for comparison purposes, the average distance function of another ensemble of parity-check codes will be derived first. Since a parity-check code is completely specified by a parity check matrix, an ensemble of parity-check codes may be defined in terms of an ensemble of parity-check matrices. The equiprobable ensemble of parity-check codes of rate R and block length n will be defined as the ensemble in which the $n(1-R)$ by n parity-check matrix is filled with statistically independent equiprobable binary digits. This is essentially the same ensemble as that considered by Elias [3] in his random coding bounds for parity-check codes; the minor difference is that codes in this ensemble may have a rate slightly higher than R, since the rows of a matrix in this ensemble are not necessarily independent over the modulo 2 field.

Theorem 2.1. *Let $\overline{N(\ell)}$ be the average number of code words of weight ℓ in a code averaged over the equiprobable ensemble of parity-check codes of length n and rate R. Then for $\ell > 0$,*

$$\overline{N(\ell)}\binom{n}{\ell}2^{-n(1-R)} \leq \left[2\pi n\lambda(1-\lambda)\right]^{-\frac{1}{2}} \exp n\left[H(\lambda) - (1-R)\ln 2\right] \tag{2.1}$$

where

$$\lambda = \frac{\ell}{n}$$

$$H(\lambda) = \lambda \ln\frac{1}{\lambda} + (1-\lambda)\ln\frac{1}{1-\lambda}$$

Proof. Let $P(\ell)$ be the probability of the set of codes for which some particular sequence of weight ℓ is a code word. Stated differently, $P(\ell)$ is the probability that a particular sequence of weight ℓ will be a code word in a code chosen at random from the ensemble. Since the all-zero code word is a code word in any parity-check code, $P(\ell) = 1$ for $\ell = 0$. For $\ell \neq 0$, a particular parity-check will check with probability $\frac{1}{2}$ on the last position in which the ℓ weight sequence has a one. This makes the probability $\frac{1}{2}$ that a parity-check is satisfied regardless whether the first $\ell - 1$ ones were checked an even or an odd number of times. A sequence will be a code if and only if it satisfies all the $n(1-R)$ parity checks, so that

$$P(\ell) = 2^{-n(1-R)}; \quad \text{for } \ell \neq 0$$

The probability $P(\ell)$ can also be interpreted as an expectation of a random variable that is 1 if the sequence is a code word and 0 otherwise. Now we observe that there are $\binom{n}{\ell}$ sequences of weight ℓ and that the expected number of code words among these sequences is the sum of the expectations that the individual sequences are code words. Thus

$$\overline{N(\ell)} = \binom{n}{\ell} 2^{-n(1-R)} \tag{2.2}$$

We now bound $\binom{n}{\ell}$ by the Stirling approximation:

$$\frac{1}{\sqrt{2\pi n}} n^n \exp\Big(-n + \frac{1}{12n} - \frac{1}{360n^3}\Big) \leq \binom{n}{\ell} \leq \frac{1}{\sqrt{2\pi n}} n^n \exp\Big(-n + \frac{1}{12n}\Big) \tag{2.3}$$

It follows after some manipulation that for $\lambda n = \ell$

$$\frac{1}{\sqrt{2\pi n \lambda(1-\lambda)}} \exp\Big(nH(\lambda) - \frac{1}{12n\lambda(1-\lambda)}\Big) < \binom{n}{n\lambda} < \frac{1}{\sqrt{2\pi n\lambda(1-\lambda)}} \exp nH(\lambda) \tag{2.4}$$

where

$$H(\lambda) = -\lambda \ln \lambda - (1-\lambda)\ln(1-\lambda)$$

Combining Equations (2.4) and (2.2), we get the statement of the theorem. □

Next we observe that over the equiprobable ensemble of parity-check codes, the minimum distance of the individual codes is a random variable whose distribution function can be bounded by the following theorem.

Theorem 2.2. *Over the equiprobable ensemble of parity-check codes of length* n *and rate* R, *the minimum-distance distribution function* $\Pr(D \leq \delta n)$ *is bounded by both the following inequalities for* $\delta < \frac{1}{2}$ *and* δn *and integer:*

$$\begin{aligned} &\Pr(D \leq \delta n) \leq \frac{1}{1-2\delta}\sqrt{\frac{1-\delta}{2\pi n\delta}} \exp n\big[H(\delta) - (1-R)\ln 2\big] \\ &\Pr(D \leq \delta n) \leq 1 \end{aligned} \tag{2.5}$$

Proof. It was shown in Theorem 2.1 that the probability that a nonzero sequence is a code word over the ensemble of codes is $2^{-n(1-R)}$. The probability that any sequence of weight $n\delta$ or less is a code word is certainly less than the sum of the probabilities that the individual sequences are code words. Thus,

$$\Pr(D \leq n\delta) \leq \sum_{\ell=1}^{n\delta} \binom{n}{\ell} 2^{-n(1-R)} \tag{2.6}$$

$$\sum_{\ell=1}^{n\delta} \binom{n}{\ell} = \binom{n}{n\delta} \left[1 + \frac{n\delta}{n - n\delta + 1} + \frac{n\delta(n\delta - 1)}{(n - n\delta + 1)(n - n\delta + 2)} + \cdots \right]$$

Bounding this by a geometric series, we get

$$\sum_{\ell=1}^{n\delta} \binom{n}{\ell} \leq \binom{n}{n\delta} \frac{1-\delta}{1-2\delta} \tag{2.7}$$

Bounding Equation (2.7) by (2.4) and substituting into Equation (2.6), we get the statement of the theorem. □

As n gets larger, this bound to $\Pr(D \leq \delta n)$ as a function of δ approaches a step function with the step at that $\delta_0 < \frac{1}{2}$ for which $H(\delta_0) = (1 - R) \ln 2$. Figure 2.4 plots δ_0 as a function of rate. This result is closely related to the Gilbert bound on minimum distance [6]. The asymptotic form of the Gilbert bound for large n states that there exists *a code* for which $D \geq n\delta_0$. Theorem 2.2 states that for any $\epsilon > 0$, the probability of the set of parity-check codes for which $D < n(\delta_0 - \epsilon)$ approaches 0 exponentially with n.

2.2 Distance Properties of Low-Density Codes

In this section an ensemble of low-density parity-check codes will be defined, and theorems similar to Theorems 2.1 and 2.2 will be proved. Then a new ensemble will be formed by expurgating those codes that have small minimum distances. This expurgated ensemble will be used in the next chapter to derive bounds on the probability of decoding error for various channels.

Define an (n, j, k) parity-check matrix as a matrix of n columns that has j ones in each column, k ones in each row, and zeros elsewhere. It follows from this definition that an (n, j, k) parity-check matrix has nj/k rows and thus a rate $R \geq 1 - j/k$. In order to construct an ensemble of (n, j, k) matrices, consider first the special (n, j, k) matrix in Figure 2.1, for which $n = 20$, $j = 3$, and $k = 4$.

This matrix is divided into j submatrices, each containing a single 1 in each column. The first of these submatrices contains all its 1's in descending order; that is, the i^{th} row contains 1's in columns $(i - 1)k + 1$ to ik. The other submatrices are merely column permutations of the first. We define the ensemble of (n, j, k) codes as the ensemble resulting from random permutations of the columns of each of the bottom $j - 1$ submatrices of a matrix such as in

1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1

Figure 2.1: Example of a low-density code matrix for $n = 20$, $j = 3$, and $k = 4$.

Figure 2.1 with equal probability assigned to each permutation. This definition is somewhat arbitrary and is made for mathematical convenience. In fact such an ensemble does not include all (n, j, k) codes as just defined. Also, at least $(j-1)$ rows in each matrix of the ensemble are linearly dependent. This simply means that the codes have a slightly higher information rate than the matrix indicates.

Before finding the average distance function and the minimum-distance distribution function for these ensembles of codes, we need the following theorem.

Theorem 2.3. *For each code in an (n, j, k) ensemble, the number $N_1(\ell)$ of sequences of weight ℓ that satisfies any one of j blocks of n/k parity-checks is bounded by*

$$N_1\left[\frac{n}{k}\mu'(s)\right] \le \exp\frac{n}{k}\left[\mu(s) - s\mu'(s) + (k-1)\ln 2\right] \tag{2.8}$$

where s is an arbitrary parameter, $\mu(s)$ is defined by

$$\mu(s) = \ln 2^{-k}\Big[(1+e^s)^k + (1-e^s)^k\Big] \tag{2.9}$$

and

$$\mu'(s) = \frac{d\mu(s)}{ds}$$

Discussion. This theorem relates ℓ and $N_1(\ell)$ by expression both as functions of the parameter s. Figure 2.2 sketches ℓ/n and $[\ln N_1(\ell)]/n$ as functions of s.

Proof. For any code in the ensemble, and for any one of the j blocks of n/k parity checks, the n/k parity-check *sets* within a block are mutually exclusive and exhaust all the digits. Consider the set of all sequences of k binits that contain an even number of ones, and construct an ensemble from these sequences

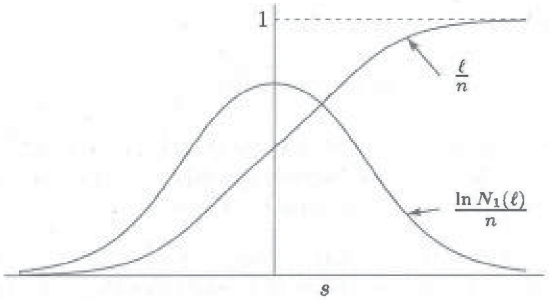


Figure 2.2: The functions ℓ/n and $[\ln N_1(\ell)]/n$ as parametric functions of s.

by assigning the same probability to each. The total number of sequences in the ensemble is 2^{k-1}, and the probability of a sequence containing i ones (i even) is $\binom{k}{i}2^{-k+1}$. The moment-generating function for the number of ones in a sequence is thus

$$g(s) = \sum_{i \text{ even}} \binom{k}{i} 2^{-k+1} e^{si} \tag{2.10}$$

or

$$g(s) = 2^{-k}\left[(1+e^s)^k + (1-e^s)^k\right] \tag{2.11}$$

To show that Equations (2.10) and (2.11) are equivalent, use the binomial expansion on Equation (2.11) and observe that odd terms cancel.

For each of the n/k parity-check sets, independently choose a sequence from the previous ensemble and use that sequence as the binits in that parity-check set. This procedure defines an ensemble of equiprobable events in which the events are the n-length sequences satisfying the n/k parity checks. The number of ones in an n-length sequence is the sum of the number of ones in the individual parity-check sets, and thus the sum of n/k independent random variables each having the moment-generating function $g(s)$ in Equation (2.11). Consequently, the moment generating function for the number of ones in an n-length sequence is $[g(s)]^{n/k}$. This is now used to bound the probability $Q(\ell)$ in this ensemble that a sequence has ℓ ones. By definition,

$$[g(s)]^{\frac{n}{k}} = \sum_{\ell=0}^{n} \exp(s\ell)Q(\ell) \tag{2.12}$$

$$\geq \exp(s\ell)Q(\ell); \quad \text{for any } s \text{ and } \ell \tag{2.13}$$

From Equation (2.9) and Equation (2.11), $\mu(s) = \ln g(s)$, so that

$$Q(\ell) \leq \exp\left[\frac{n}{k}\mu(s) - s\ell\right]$$

Finally, $N_1(\ell)$ equals $Q(\ell)$ times the number of sequences in the ensemble. Since there are 2^{k-1} sequences in the k-length ensemble, there are $2^{n(k-1)/k}$ sequences in the n-length ensemble, so that

$$N_1(\ell) \leq \exp\Big[\frac{n}{k}\mu(s) + \frac{n}{k}(k-1)\ln 2 - s\ell\Big] \tag{2.14}$$

When we set the derivative of the exponent in Equation (2.14) equal to 0, we get $\ell = (n/k)\mu'(s)$, and when we substitute this value of ℓ in Equation (2.14), Equation (2.8) results, thereby proving the theorem. □

It is shown in Reference [4] that setting $\ell = (n/k)\mu'(s)$ actually minimizes the exponent, thereby yielding the best bound; however, the theorem is true regardless of the minimal character of the exponent. Although it is not necessary in the proof, it can be shown, using "tilted" probabilities [4] and a central limit theorem [7], that asymptotically for large n,

$$N_1\Big[\frac{n}{k}\mu'(s)\Big] \rightarrow \frac{2}{\sqrt{2\pi n\mu'(s)}} \exp\frac{n}{k}\big[\mu(s) - s\mu'(s) + (k-1)\ln 2\big] \tag{2.15}$$

Theorem 2.3 can now be used to find the probability $P(\ell)$ of the set of codes for which some particular sequence of weight ℓ is a code word. Since all permutations of a code are equally likely, $P(\ell)$ is clearly independent of the particular ℓ-weight sequence chosen. If we choose an ℓ-weight sequence at random, then for any code in the ensemble the probability is $N_1(\ell)/\binom{n}{\ell}$ that the ℓ-weight sequence chosen will satisfy any particular block of n/k parity checks. Since each of the j blocks of parity checks is chosen independently,

$$P(\ell) = \left[\frac{N_1(\ell)}{\binom{n}{\ell}}\right]^j \tag{2.16}$$

The distance properties and the minimum-distance distribution function can now be derived in terms of $P(\ell)$ in the same way as they were derived for the ensemble of all parity-check codes in Equations (2.1) and (2.5).

$$\overline{N_{jk}(\ell)} \leq \binom{n}{\ell}P(\ell) = \binom{n}{\ell}^{-j+1} N_1(\ell)^j \tag{2.17}$$

$$\Pr(D \leq n\delta) \leq \sum_{\ell=2}^{n\delta}\binom{n}{\ell}P(\ell) = \sum_{\ell=2}^{n}\binom{n}{\ell}^{-j+1} N_1(\ell)^j \tag{2.18}$$

Note that in the low-density ensemble only sequences of even weight may be code words. Using Equations (2.4) and (2.14), we get

$$\overline{N_{jk}(\ell)} \leq C(\lambda,n)\exp -nB_{jk}(\lambda); \quad \text{where } \lambda = \frac{\ell}{n} \tag{2.19}$$

$$B_{jk}(\lambda) = (j-1)H(\lambda) - \frac{j}{k}\big[\mu(s) + (k-1)\ln 2\big] + js\lambda \tag{2.20}$$

$$C(\lambda,n) = \big[2\pi n\lambda(1-\lambda)\big]^{\frac{j-1}{2}} \exp\frac{j-1}{12n\lambda(1-\lambda)}; \quad \text{where } \lambda = \frac{\mu'(s)}{k} \tag{2.21}$$

Substituting Equation (2.19) into Equation (2.18), we get

$$\Pr(D \leq n\delta) \leq \sum_{\ell=2}^{n\delta} C(\lambda, n) \exp -nB_{jk}(\lambda) \tag{2.22}$$

For n large, the summations in Equations (2.19) and (2.22) are governed principally by the behavior of $B_{jk}(\lambda)$; $B_{jk}(\lambda)$ also appears in the bounds for probability of decoding error in the next chapter. Unfortunately, it is not easy to analyze $B_{jk}(\lambda)$ since it is given in terms of s, which is in turn an implicit function of λ. It is shown in Appendix A that for $j \geq 3$, $B_{jk}(\lambda)$ has the behavior shown in Figure 2.3. It is 0 at $\lambda = 0$; rises with an initial infinite slope; has a maximum; and then decreases, crossing the axis at some $\lambda = \delta_{jk}$, and remains negative for $\lambda > \delta_{jk}$.

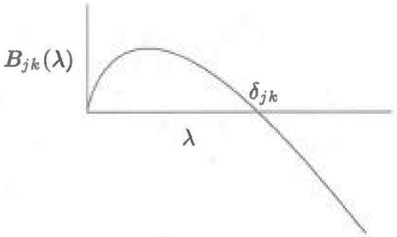


Figure 2.3: Sketch of the function $B_{jk}(\lambda)$.

It is clear that for any $\delta > \delta_{jk}$ the summation in Equation (2.22) becomes unbounded, but the minimum-distance distribution function is still bounded by 1. For $\delta < \delta_{jk}$, the biggest terms in the summation are for λ close to 0 and λ close to δ_{jk}. The following theorem, which is proved in Appendix A, states this precisely.

Theorem 2.4. *For an (n, j, k) ensemble of codes, the minimum-distance distribution function is bounded by both*

$$\Pr(D \leq n\delta) \leq \frac{k-1}{2n^{j-2}} + 0\Big(\frac{1}{n^{j-2}}\Big) + nC(n\delta, n) \exp -nB_{jk}(\delta) \tag{2.23}$$

and

$$\Pr(D \leq n\delta) \leq 1$$

where C and B are defined in Equations (2.20) *and* (2.21) *and $0(1/n^{j-2})$ is a function approaching zero with n faster than $1/n^{j-2}$.*

The first term in Equation (2.23) comes from code words of weight 2; the next term comes from words of small weights greater than 2; and the last term comes from words of large weight. As n gets larger, this bound to the minimum-distance distribution function tends toward small step at $\delta = 2/n$, and a large step at $\delta = \delta_{jk}$, with the amplitude of the small step decreasing as n^{-j+2}.

The expression δ_{jk} will be called the typical minimum-distance ratio of an (n, j, k) ensemble. For large n, most codes in the ensemble have a minimum distance either close to or greater than $n\delta_{jk}$; since δ_{jk} is independent of block length, the minimum distance typical of most codes in the ensemble increases linearly with the block length. Figure 2.4 plots δ_{jk} as a function of rate for several values of j and k and compares them with the typical minimum distance ratio of the equiprobable ensemble of codes. It can be seen that as j and k increases, δ_{jk} for the (n, j, k) codes quickly approaches δ_0 for the equiprobable ensemble of codes. This is proved in Theorem A.3 of Appendix A.

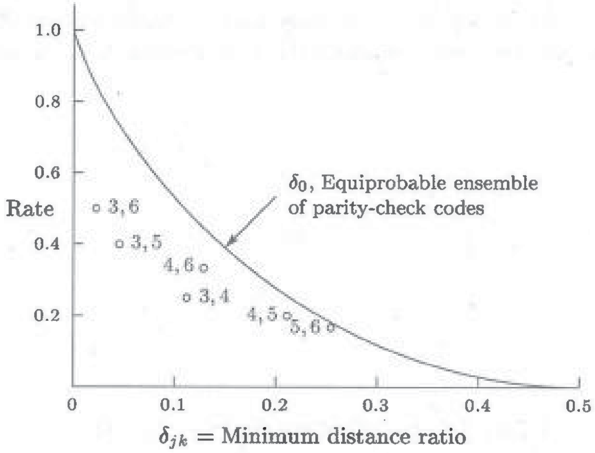


Figure 2.4: Ratio of minimum distance to block length for typical long (n, j, k) codes.

Here we see why a minimum-distance distribution function was derived before any results were obtained about probability of decoding error. If two words in a group code differ only in two digits, then the probability of a decoding error is lower bounded by the probability of receiving those two digits incorrectly; this is independent of code length. Thus, over the whole ensemble, the probability of decoding error as $n \to \infty$ is proportional to $1/n^{j-2}$, the probability of codes of minimum distance 2. Thus a very small number of poor codes dominates the probability of decoding error over the ensemble.

In order to determine the probability of error behavior of typical (n, j, k) codes with minimum distances in the order of $n\delta_{jk}$, we shall modify the (n, j, k) ensemble. Remove the half of the codes with smallest minimum distances from an (n, j, k) ensemble and double the probability of each code in the remaining half. The resulting ensemble will be called an expurgated (n, j, k) ensemble and will be used in Chapter 3 to derive bounds on the probability of decoding error for (n, j, k) codes.

Let δ_{njk} be the minimum distance of the expurgated ensemble. Then δ_{njk} is lower bounded by that value of δ for which the right hand side of Equation (2.23) is one-half. With increasing n, the bound of Equation (2.23) approaches a step function at δ_{jk}, so that δ_{njk} is asymptotically bounded by δ_{jk}. For the expurgated low-density ensemble, we now have

$$\overline{N_{jk}(\ell)} \left\{ \begin{array}{ll} \leq 2C(\lambda, n) \exp -nB_{jk}(\lambda); & \lambda \geq \delta_{njk} \\ = 0; \quad \lambda < \delta_{njk} & \end{array} \right\} \tag{2.24}$$

Similarly, we can expurgate the random ensemble of parity-check codes to get, from Equations (2.1) and (2.5)

$$\overline{N(\ell)} \left\{ \begin{array}{ll} \leq 2\big[2\pi\lambda(1-\lambda)\big]^{-\frac{1}{2}} \exp n\big[H(\lambda) - (1-R)\ln 2; & \lambda \geq \delta_0 \\ = 0; \quad \lambda \leq \delta_0 & \end{array} \right\} \tag{2.25}$$

where δ_0 satisfies $H(\delta_0) = (1-R)\ln 2$, and n is large enough so that

$$\frac{1}{1-2\delta_0}\sqrt{\frac{1-\delta_0}{2\pi n \delta_0}} \leq \tfrac{1}{2}$$

Before using this modified (n, j, k) ensemble to derive bounds to the probability of decoding error, we shall consider the special case of $j = 2$, which corresponds to ensembles in which each digit is contained in exactly two parity-check sets.

Theorem 2.5. *Let a parity-check code have block length n with each digit contained in exactly two parity-check sets, at let each parity-check set contain k digits. Then the minimum distance D of this code must be bounded by*

$$D \leq 2 + \frac{2\ln\frac{n}{2}}{\ln(k-1)} \tag{2.26}$$

Proof. The theorem will be proved by representing the code in the form of a tree as in Figure 2.5. Let the first digit in the code be represented by the node at the base of the tree. This digit is contained in two parity-check sets, which are denoted by the two branches rising from the base node. The other digits in these two parity-check sets are represented by the nodes in the first tier of the tree. In like manner, each digit in the first tier is contained in another parity-check set depicted by a branch rising from that digit.

Successive tiers in the base may be similarly constructed until, for some integer m a loop is formed by the branches rising from the m^{th} tier. Such a loop may occur either if two parity-check sets rising from the m^{th} tier contain a digit on tier $m+1$, as in Figure 2.5, or if a single parity-check set rising from the m^{th} tier contains more than one digit in the m^{th} tier.

We next bound m in terms of the block length n. The first of the tree contains $2(k-1)$ nodes; the second contains $2(k-1)^2$ nodes; and similarly the m^{th} tier contains $2(k-1)^m$ nodes, since by assumption no loop occurs in branches below the m^{th} tier. Since each node corresponds to a distinct digit,

$$\begin{aligned} 2(k-1)^m &\leq n \\ m &\leq \frac{\ln\frac{n}{2}}{\ln(k-1)} \end{aligned} \tag{2.27}$$

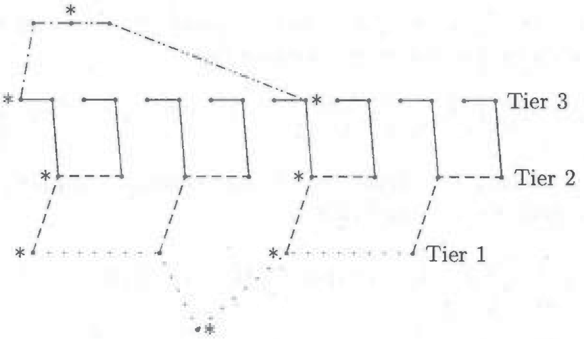


Figure 2.5: Parity-check tree

For a given loop in the tree, consider the set of nodes that comprise the intersections of the branches in the loop. Such a set of nodes is represented by asterisks in Figure 2.5. Each branch in the loop must contain exactly two of these nodes, and no other branch in the tree contains any of these nodes. Consequently, an n-length sequence that contains ones in positions corresponding to the nodes of this set and zeros elsewhere must be a code word, since all parity-check sets contain an even number of ones. Finally, the weight D of the code word corresponding to the first loop that occurs must be bounded by

$$D \leq 2m + 2 \tag{2.28}$$

since the loop is formed by a single descent and ascent in the tree. Combining Equations (2.27) and (2.28), we get the statement of the theorem, Equation (2.26). □

3 Probability of Decoding Error

A technique for upper bounding the probability of decoding error for arbitrary binary block codes will be developed in this chapter. It will be assumed that the decoding is maximum likelihood and that the channel has a binary-input alphabet, an arbitrary output alphabet, and is symmetrical in a sense to be defined later.

The reason for developing this technique is threefold: First, it is needed to demonstrate the capabilities of low-density codes; second, it provides a tool both for comparing codes and for gaining insight into the relation between a code's distance properties and its probability of decoding error; third, it yields a conceptually simpler, although analytically more complicated, technique for analyzing random ensembles of codes. The conceptual simplification here lies in a separation of the analysis of the channel from the analysis of the ensemble of codes (which is used to derive the distance properties of the ensemble).

3.1 Symmetric Binary Input Channels

A symmetric binary-input channel is defined as a time-discrete channel with the following properties:

1. The input alphabet X consists of two letters, denoted by 0 and 1.
2. The output alphabet Y can be represented either as a discrete or a continuous set of real numbers.
3. The output y at a given discrete time is statistically dependent only on the current input x.
4. The symmetry condition given by Equation (3.1) holds for all outputs y.

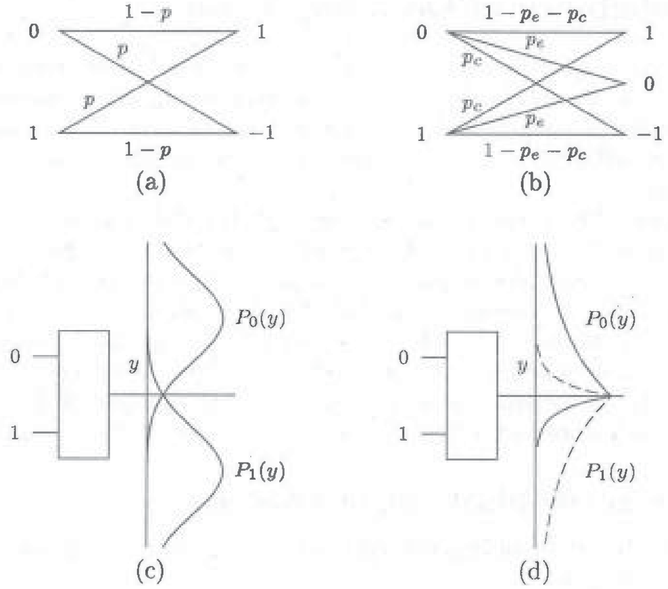
$$P_0(y) = P_1(-y) \tag{3.1}$$

In this equation and throughout this chapter, $P_x(y)$ is a conditional probability *density* if Y is a continuous set, and is a conditional probability if Y is a discrete set.

Some examples of such channels are given in Figure 3.1. The lack of symmetry between the labeling of input and output is regrettable, but a change in output labeling would greatly complicate the symmetry condition, Equation (3.1), and a change in input labeling would make parity-check codes seem less familiar to symbol-oriented readers.

3.2 Distance Properties

Assume that in a particular code of block length n, an arbitrary code word u_0 is transmitted, and assume that the number of other code words $N(\ell)$ at each distance ℓ from u_0 is known. In the following section, the probability of error



a. Binary symmetric channel.

b. Binary symmetric threshold channel.

c. Additive white Gaussian noise channel; log-likelihood output (see Section 6.3).

$$\begin{aligned} P_0(y) &= \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{\left(y-\sigma^2/2\right)^2}{2\sigma^2} \\ P_1(y) &= \frac{1}{\sqrt{2\pi}\sigma} \exp -\frac{\left(y+\sigma^2/2\right)^2}{2\sigma^2} \\ \sigma^2 &= \frac{4E_c(1-\rho)}{N_0} \end{aligned}$$

d. Rayleigh fading channel; log-likelihood output (see Section 6.4).

$$\begin{aligned} P_0(y) &= \frac{1+A}{A(2+A)} \exp -\frac{y}{A}; \quad y \geq 0 \\ &= \frac{1+A}{A(2+A)} \exp \frac{y(1+A)}{A}; \quad y < 0 \\ P_1(y) &= P_0(-y) \\ A &= \frac{E_c}{N_0} \end{aligned}$$

Figure 3.1: Symmetric binary-input channels.

using maximum-likelihood decoding will be upper bounded in terms of $N(\ell)$ when u_0 is transmitted over a symmetric binary-input channel. This bound will then easily be extended to apply to an entire code or to an ensemble of codes.

3.3 Upper Bound on Probability of Decoding Error

Let $u_0 = x_{10}, x_{20}, \ldots, x_{n0}$ be the transmitted code word, and let $v = y_1, y_2, \ldots, y_n$ be the received sequence. Let the other code words be numbered $u_1, \ldots, u_j, \ldots, u_{M-1}$ where $u_j = x_{1j}, \ldots, x_{nj}$. Using maximum likelihood decoding, a decoding error will be made if $P(v|u_j) > P(v|u_0)$ for any j, $1 \leq j \leq M-1$. Also, a decoding error might be made if $P(v|u_j) = P(v|u_0)$, and in upper bounding the probability of decoding error, we can assume that such errors are always made. Using the assumption of statistical independence between the n uses of the channel, this condition for decoding errors becomes

$$\prod_{i=1}^{n} P_{x_{ij}}(y_i) \geq \prod_{i=1}^{n} P_{x_{i0}}(y_i), \quad \text{for some } j,\ 1 \leq j \leq M-1 \tag{3.2}$$

Thus, the probability of decoding error is upper bounded by the probability that Equation (3.2) is satisfied. Equation (3.2) becomes easier to work with if we take the logarithm of both sides, yielding the following inequality between sums of random variables:

$$\sum_{i=1}^{n} \ln P_{x_{ij}}(y_i) \geq \sum_{i=1}^{n} \ln P_{x_{i0}}(y_i) \tag{3.3}$$

Finally, for reasons to be discussed later, we subtract an arbitrary function of the output sequence, $\sum_{i=1}^{n} f(y_i)$ from both sides of Equation (3.3) and multiply by -1, yielding the following condition for decoding errors:

$$\sum_{i=1}^{n} \ln \frac{f(y_i)}{P_{x_{ij}}(y_i)} \leq \sum_{i=1}^{n} \ln \frac{f(y_i)}{P_{x_{i0}}(y_i)} \tag{3.4}$$

for some j, $1 \leq j \leq M-1$. We place the following restriction on $f(y)$: $f(y)$ is positive if either $P_0(y)$ or $P_1(y)$ is positive, and

$$f(y) = f(-y); \quad \text{for all } y \tag{3.5}$$

Next we define the discrepancy $\delta(x_i y_i)$ between an input x_i and an output y_i as

$$\delta(x_i y_i) = \ln \frac{f(y_i)}{P_{x_i}(y_i)} \tag{3.6}$$

Further, define the discrepancy $D(uv)$ between u and v as

$$D(uv) = \sum_{i=1}^{n} \delta(x_i y_i) \tag{3.7}$$

From Equations (3.4), (3.6), and (3.7), we see that a decoding error is made only when $D(u_jv) \leq D(u_0v)$ for any code word u_j other than u_0. More formally, the probability of decoding error P_e is bounded by

$$P_e \leq \Pr\left\{ \bigcup_{j=1}^{M-1} [\text{event that } D(u_jv) \leq D(u_0v)] \right\} \tag{3.8}$$

The most obvious procedure for simplifying Equation (3.8) would be to upper bound the probability of the union of events by the sum of the probabilities of the individual events. This does not yield a good upper bound, however; when $D(u_0v)$ is very large, say greater than a suitable constant nd, it is likely to be larger than many of $D(u_jv)$, thereby causing one decoding error to be counted many times in the bound. To avoid this difficulty, we shall use separate bounding techniques on those events for which $D(u_0v) \geq nd$. The parameter d is arbitrary and will be optimized later. Thus, if we split Equation (3.8), we get

$$P_e \leq P_1 + P_2 \tag{3.9}$$

where

$$P_1 = \Pr\left\{ \bigcup_{j=1}^{M-1} [\text{event that } D(u_0v) > nd;\ D(u_jv) \leq D(u_0v)] \right\}$$

$$P_2 = \Pr\left\{ \bigcup_{j=1}^{M-1} [\text{event that } D(u_0v) \leq nd;\ D(u_jv) \leq D(u_0v)] \right\}$$

Now we can bound P_1 and P_2 separately by

$$P_1 \leq \Pr[D(u_0v) > nd] \tag{3.10}$$

$$P_2 \leq \sum_{j=1}^{M-1} \Pr[D(u_0v) \leq nd; \quad D(u_jv) \leq D(u_0v)] \tag{3.11}$$

Observe that Equation (3.9) is an exact expression for P_e except for the bounding involved in assuming that ambiguities (that is, cases where $D(u_0v) = D(u_jv)$) always cause errors. Thus the arbitrary function $f(y)$ can have no effect on Equation (3.9) since it has no effect on which word is decoded when u_0 is transmitted. The function $f(y)$ does have an effect on Equations (3.10) and (3.11), however, since the function helps determine the set of output sequences for which $D(u_0v) \geq nd$.

Finally, observe from Equation (3.7) that $D(u_0v)$ and $D(u_jv)$ are both defined as sums of random variables, and thus, using Equations (3.10) and (3.11), the problem of bounding P_e has been reduced to the problem of bounding the tails of the distributions of sums of random variables. This is best done by the Chernov bound technique, briefly described in Appendix B. For a more detailed exposition, see Fano [4, Chapter 8].

3.4 Chernov Bounds

In order to bound P_1 in Equation (3.10), we need the following theorem which is proved in Appendix B.

Theorem 3.1. *Let $Z = \sum_{i=1}^{n} z_i$ be the sum of n independent random variables, let $P_i(z_i)$ be the probability density of the i^{th} variable, and let $g_i(s) = \int_{-\infty}^{\infty} \exp(sz_i)P_i(z_i)\,dz_i$ be the moment generating function for the i^{th} variable. Then*

$$Pr(Z \geq nz_0) \leq \exp(-nsz_0)\prod_{i=1}^{n} g_i(s) \tag{3.12}$$

for all $s \geq 0$ such that the $g_i(s)$ exist. If the z_i are discrete, then the same statement holds except that the $P_i(z_i)$ are probabilities and the integral defining $g_i(s)$ is replaced by a sum.

To apply this theorem to $D(u_0v) = \sum_{i=1}^{n} \delta(x_{i0}y_i)$, we consider $\delta(x_{i0}y_i)$ as a random variable where x_i is given and y_i is determined according to $P_{x_i}(y_i)$. The moment generating function of δ is then

$$g_i(s) = \int_{-\infty}^{\infty} \exp\bigl[s\delta(x_{i0}y_i)\bigr]P_{x_{i0}}(y_i)\,dy_i \tag{3.13}$$

Using Equation (3.6), this becomes

$$g_i(s) = \int_{-\infty}^{\infty} \bigl[P_{x_{i0}}(y_i)\bigr]^{1-s}\bigl[f(y_i)\bigr]^{s}\,dy_i \tag{3.14}$$

For $x_{i0} = 0$, Equation (3.14) becomes

$$g_i(s) = \int_{-\infty}^{\infty} P_0(y)^{1-s} f(y)^{s}\,dy \tag{3.15}$$

For $x_{i0} = 1$, using the symmetry conditions Equations (3.1) and (3.5), Equation (3.14) becomes

$$g_i(s) = \int_{-\infty}^{\infty} P_0(-y)^{1-s} f(-y)^{s}\,dy \tag{3.16}$$

Substituting $-y$ for y as the variable of integration, we see that Equations (3.15) and (3.16) are identical, and thus $g_i(s)$ is independent of x_{i0} and i:

$$g_i(s) = g(s) = \int_{-\infty}^{\infty} P_0(y)^{1-s} f(y)^{s}\,dy \tag{3.17}$$

It can be shown that Equation (3.17) is equivalent to the distribution function of the discrepancy between the channel input and output and is independent of the input as is reasonable from the channel symmetry.

Finally, using Theorem 3.1, we get

$$P_1 \leq \Pr\bigl[D(u_0v) \geq nd\bigr] \leq g(s)^n \exp(-nsd) \tag{3.18}$$

for any $s \geq 0$ such that $g(s)$ in Equation (3.17) exists.

To complete our bound on error probability, P_2, as given by Equation (3.11) must be bounded. This requires the following theorem which is proven in Appendix B.

Theorem 3.2. *Let z_i and w_i, $1 \leq i \leq n$, be n pairs of random variables with probability density functions $P_i(z_i, w_i)$. Let the joint moment generating function of z_i, w_i, be*

$$h_i(r,t) = \iint \exp(rz_i + tw_i)P_i(z_i, w_i)\, dz_i\, dw_i \tag{3.19}$$

Let each pair of random variables be statistically independent of each other pair and define Z and W by

$$\begin{aligned} Z &= \sum_{i=1}^{n} z_i \\ W &= \sum_{i=1}^{\ell} w_i \\ \ell &\leq n \end{aligned} \tag{3.20}$$

Then, for any arbitrary numbers z_0 and w_0,

$$P(Z \leq nz_0;\, W \leq nw_0) \leq \prod_{i=1}^{\ell}[h_i(r,t)] \prod_{i=\ell+1}^{n} [h_i(r,0)] \exp -n(rz_0 + tw_0) \tag{3.21}$$

for any $r \leq 0$, $t \leq 0$ for which $h_i(r,t)$ exists. If z and w are discrete, Equation (3.21) still holds with integrals in Equation (3.19) replaced by sums, and the probability density replaced by a probability.

This theorem will be used to bound $\Pr[D(u_0v) \leq nd;\, D(u_jv) - D(u_0v) \leq 0]$ for each code word u_j. Assume first that u_j differs from u_0 in the first ℓ digits and is identical to u_0 in the last $n - \ell$ digits. Then

$$D(u_j, v) - D(u_0, v) = \sum_{i=1}^{\ell} \delta(x_{ij}, y_i) - \delta(x_{i0}, y_i)$$

From the symmetry conditions, Equations (3.1) and (3.5), and from the definition of δ in Equation (3.6), we note that $\delta(x_{ij}, y_i)$ equals $\delta(x_0, -y_i)$ for $i \leq \ell$ since we have assumed that $x_{ij} \neq x_{i0}$ for $i \leq \ell$. Now let

$$z_i = \delta(x_{i0}, y_i); \quad w_i = \delta(x_{i0}, -y_i) - \delta(x_{i0}, y_i)$$

For a given x_i, both z_i and w_i are functions of y_i, and we can write $h_i(r,t)$ in Equation (3.19) as

$$h_i(r,t) = \int_{-\infty}^{\infty} \exp\big[r\delta(x_{i0},y_i) + t\delta(x_{i0},-y_i) - t\delta(x_{i0},y_i)\big] P_{x_{i0}}(y_i)\,dy_i \tag{3.22}$$

Writing out $h_i(r,t)$ in the same way as $g_i(s)$ in Equation (3.13), we see that $h_i(r,t)$ is independent of x_{i0} and i. Thus

$$h_i(r,t) = h(r,t) = \int_{-\infty}^{\infty} P_0(y)^{1-r+t} P_0(-y)^{-t} f(y)^r\,dy \tag{3.23}$$

Now, applying Theorem 3.2, we get

$$\Pr\big[D(u_0v) \leq nd;\ D(u_jv) - D(u_0v) \leq 0\big] \leq \big[h(r,t)\big]^{\ell}\big[h(r,0)\big]^{n-\ell} e^{-nrd} \tag{3.24}$$

for any $r \leq 0$, $t \leq 0$ if u_j and u_0 differ in the first ℓ digits.

Finally, by renumbering the n digits in a block, we see that the bound in Equation (3.24) applies to each of the $N(\ell)$ code words that are distance ℓ from u_0. Thus

$$P_2 \leq \sum_{\ell=0}^{n} N(\ell)\big[h(r,t)\big]^{\ell}\big[h(r,0)\big]^{n-\ell} e^{-nrd} \tag{3.25}$$

for any $r \leq 0$, $t \leq 0$ where $h(r,t)$ is given in Equation (3.23).

The term in Equation (3.25) for $\ell = 0$ accounts for the pathological possibility that one of the other code words is identical to u_0; $N(0)$ is the number of code words other than u_0 that are identical to u_0.

Equations (3.25) and (3.18) give bounds for P_1 and P_2; from Equation (3.9) their sum bounds the maximum-likelihood probability of decoding error when a given code word is transmitted. This bound is in terms of the code distances $N(\ell)$, the channel transition probabilities $P_0(y)$, and a number of stray parameters that must be optimized; namely, s, r, t, d, and $f(y)$. Thus the combinatorial and probabilistic aspects of the problem have been solved, and given $N(\ell)$, Equations (3.25) and (3.18) are essentially as simple to evaluate when the block length is large as when it is small. However, the optimization problem is by no means trivial since the equations are transcendental and involve constraints on s, r, t and $f(y)$. One simplification is to eliminate the parameter t. Equation (3.25) is minimized with respect to t by minimizing $h(r,t)$, which can be accomplished by setting $\partial h(r,t)/\partial t = 0$. From Equation (3.23), this gives us

$$\int_{-\infty}^{\infty} \left(\ln \frac{P_0(y)}{P_0(-y)}\right) P_0(y)^{1-r+t} P_0(-y)^{-t} f(y)^r\,dy = 0 \tag{3.26}$$

If $1 - r + t = -t$, then using the symmetry of $f(y)$ we see that the integrand of Equation 3.26 is antisymmetrical in y and thus the integral is 0. To ensure that this is a minimum,

$$\frac{\partial^2 h(r,t)}{\partial t^2} = \int_{-\infty}^{\infty} \left(\ln \frac{P_0(y)}{P_0(-y)}\right)^2 P_0(y)^{1-r+t} P_0(-y)^{-t} f(y)^r\,dy \geq 0$$

Finally we observe that the solution

$$t = \frac{r-1}{2} \tag{3.27}$$

automatically satisfies the constraint

$$t \le 0; \quad \text{for } r \le 0$$

With this simplification, Equation 3.25 can be rewritten

$$P_2 \le \sum_{\ell=0}^{n} N(\ell)\left[h(r)\right]^{\ell}\left[g(r)\right]^{n-\ell} e^{-nrd} \tag{3.28}$$

$$h(r) = \int_{-\infty}^{\infty} \left[P_0(y)P_0(-y)\right]^{(1-r)/2} f(y)^r \, dy \tag{3.29}$$

$$g(r) = \int_{-\infty}^{\infty} \left[P_0(y)\right]^{1-r} f(y)^r \, dy \tag{3.30}$$

These equations use that fact that $h(r,0) = g(r)$, as can be seen from Equations (3.17) and (3.23).

3.5 $\bar{P}_e$ for Codes and Ensembles of Codes

Next consider the probability of decoding error for a complete code. Let $N_j(\ell)$ be the number of code words at distance ℓ from the code word u_j; $0 \le j \le M-1$. Then, from Equations (3.18) and (3.28), the probability of decoding error for the code assuming equiprobable use of the code words is

$$\bar{P}_e \le \sum_{j=0}^{M-1} \frac{1}{M}\Big\{g(s)^n e^{-nsd} + \sum_{\ell=0}^{n} N_j(\ell) h(r)^{\ell} g(r)^{n-\ell} e^{-nrd}\Big\}; \quad \text{for any } s \ge 0,\ r \le 0 \tag{3.31}$$

Now define

$$\overline{N(\ell)} = \frac{1}{M}\sum_{j=0}^{M-1} N_j(\ell)$$

as the average over j of the number of code words at distance ℓ from code word u_j. Equation (3.31) then becomes

$$\bar{P}_e \le g(s)^n e^{-nsd} + \sum_{\ell=0}^{n} \overline{N(\ell)} h(r)^{\ell} g(r)^{n-\ell} e^{-nrd}; \quad \text{for any } s \ge 0,\ r \le 0 \tag{3.32}$$

where

$$g(s) = \int_{-\infty}^{\infty} P_0(y)^{1-s} f(y)^s \, dy \tag{3.33}$$

$$h(r) = \int_{-\infty}^{\infty} \left[P_0(y)P_0(-y)\right]^{(1-r)/2} f(y)^r \, dy \tag{3.34}$$

Finally consider an ensemble of codes such as those considered in Chapter 2. Letting $\overline{N(\ell)}$ be the average over the ensemble of codes of the $N(\ell)$ defined in the first paragraph of this section for a particular code, Equation (3.32) again holds, and $\bar{P}_e$ is now the ensemble average probability of decoding error. Note that at least one code in the ensemble must have a probability of error as small as the average and that at least a fraction $1-\alpha$ of the codes must have a probability of error at most $\bar{P}_e/\alpha$. This last result follows from noting that if more than a fraction α of the codes had a probability of error greater than $\bar{P}_e/\alpha$ then these codes alone would contribute more than $\bar{P}_e$ to the average error probability.

The bound in Equation (3.32) is somewhat difficult to work with, first because it involves a sum over n terms where n might be large, and second because there are a number of stray parameters, r, s, d, $f(y)$, over which the bound should be optimized. Unfortunately, in general, virtually nothing can be done to simplify this bound without weakening it. Before proceeding, however, some motivation on the direction to be followed in this simplification and weakening will be helpful. It will be shown later that Equation (3.32) is approximately an exponentially decreasing function of the block length n, both for low-density and equiprobable ensembles of parity-check codes. Thus, to study $\bar{P}_e$ for very long block lengths, and to study the variation of $\bar{P}_e$ with block length, the coefficient of n in this exponential function will be of primary importance. Our aim in what follows will be to find values of d, $f(y)$, r, and s to optimize this exponential coefficient. Consequently, other parts of the expression for $\bar{P}_e$ will be ignored for purposes of optimization. Having obtained such a bound, it is of course possible to go back in any particular case and get a tighter result for Equation (3.32), but to attempt this in general would only confuse and already complicated situation.

Now assume that the distance function $\overline{N(\ell)}$ for a particular code or ensemble of codes can be bounded by an expression such as

$$\overline{N(\ell)} \le C(\lambda, n)e^{nB(\lambda)}; \quad \lambda = \frac{\ell}{n} \tag{3.35}$$

where $C(\lambda, n)$ must be a relatively small quantity for the following approach to be useful. Equations (2.1) and (2.19) give such bounds for random and low-density ensembles of parity-check codes. Now let

$$C_n = \max_{\lambda} C(\lambda, n) \tag{3.36}$$

Using C_n for $C(\lambda, n)$ in Equation (3.35), substituting this into Equation (3.32), rearranging a little, and bounding the summation by n times its maximum term, we get

$$\bar{P}_e \le \exp n\big[\ln g(s) - sd\big] + nC_n \max_{\lambda} \exp n\big[B(\lambda) + \lambda \ln h(r) + (1-\lambda)\ln g(r) - rd\big]$$

$$\text{for any } s \ge 0,\ r \le 0 \tag{3.37}$$

The functions g and h are still given by Equations (3.33) and (3.34). Equation (3.37) has two terms, each of which are essentially exponential in n. The first term decreases with d if $s > 0$, and the second increases with d if $r < 0$. Thus if we chose d to make the exponents equal, any change in d would increase one of the two exponents. Thus this choice of d minimizes the coefficient of n in the largest exponent. Eliminating d in this way, we get

$$\bar{P}_e \leq (1 + nC_n) \exp\bigl[-n \min_{\lambda} E(s, r, \lambda)\bigr]; \quad \text{for } s \geq 0,\ r \leq 0 \tag{3.38}$$

$$E(s, r, \lambda) = \frac{r}{s-r} \ln g(s) - \frac{s}{s-r}\bigl[B(\lambda) + \lambda \ln h(r) + (1-\lambda) \ln g(r)\bigr] \tag{3.39}$$

The result in Equations (3.38) and (3.39) still depends on the function $f(y)$ through the definition of the functions g and h. In Appendix B, it is shown that $E(s, r, \lambda)$ is maximized over $f(y)$ by

$$f(y) = k\left\{\frac{\left[P_0(y)^{\frac{1-r}{2}} + P_1(y)^{\frac{1-r}{2}}\right]^2 + \frac{\alpha-\lambda}{\lambda(1-\alpha)}\left[P_0(y)^{1-r} + P_1(y)^{1-r}\right]}{P_0(y)^{1-s} + P_1(y)^{1-s}}\right\}^{\frac{1}{s-r}}$$

where

$$\alpha = \frac{g(r)}{g(r) + h(r)} \tag{3.40}$$

The constant k in Equation (3.40) is arbitrary and cancels out in the bound for $\bar{P}_e$. Unfortunately, this is only an implicit solution since α itself is a function of $f(y)$. For any value of s, r, and λ, the $f(y)$ satisfying Equation (3.40) can be found only by a series of approximations for α. This makes optimizing Equation (3.39) difficult even with a computer. As a result, we choose $f(y)$ more simply to be

$$f(y) = k\left\{\frac{\left[P_0(y)^{\frac{1-r}{2}} + P_1(y)^{\frac{1-r}{2}}\right]^2}{P_0(y)^{1-s} + P_1(y)^{1-s}}\right\}^{\frac{1}{s-r}} \tag{3.41}$$

For the equiprobable ensemble of codes, maximizing $\bar{P}_e$ over λ will later be shown to yield $\lambda = \alpha$, and in this case, Equations (3.40) and (3.41) give identical results. For other ensembles, the small change in $f(y)$ caused by using Equation (3.41) instead of Equation (3.40) will cause only a second-order change in the exponent of $\bar{P}_e$.

Writing out the moment-generating functions explicitly in Equation (3.39) and using Equation (3.41), we get (see Appendix B)

$$\bar{P}_e \leq (1 + nC_n) \exp -nE(s, r) \tag{3.42}$$

$$E(s, r) = \frac{s}{s-r}\beta(\alpha) - \ln \int_0^\infty \left(P_0^{1-s} + P_1^{1-s}\right)^{-\frac{r}{s-r}} \left(P_0^{\frac{1-r}{2}} + P_1^{\frac{1-r}{2}}\right)^{\frac{2s}{s-r}} \tag{3.43}$$

$$\alpha = \frac{\displaystyle\int_0^\infty \left(P_0^{1-s} + P_1^{1-s}\right)^{-\frac{r}{s-r}} \left(P_0^{\frac{1-r}{2}} + P_1^{\frac{1-r}{2}}\right)^{\frac{2r}{s-r}} 2(P_0 P_1)^{\frac{1-r}{2}}}{\displaystyle\int_0^\infty \left(P_0^{1-s} + P_1^{1-s}\right)^{-\frac{r}{s-r}} \left(P_0^{\frac{1-r}{2}} + P_1^{\frac{1-r}{2}}\right)^{\frac{2s}{s-r}}} \tag{3.44}$$

$$\beta(\alpha) = \min_{\lambda}\left[-B(\lambda) - \lambda \ln \alpha - (1-\lambda)\ln(1-\alpha)\right] \tag{3.45}$$

Equations (3.42) to (3.45) give a general bound for $\bar{P}_e$ in terms of three parameters: s, r and λ. Equation (3.45) can be used to eliminate λ for any given $s \geq 0$ and $r \leq 0$, but maximizing $E(s,r)$ over s and r is not simple and may even involve several local maxima. However, this maximization can be performed with a computer.

3.6 Error Probability for Equiprobable Code Ensemble

As an example of the use of Equations (3.42) to (3.45) consider the special case of the equiprobable ensemble of parity-check codes, for which from Equation 2.1

$$B(\lambda) = -(1-R)\ln 2 - \lambda \ln \lambda - (1-\lambda)\ln(1-\lambda) \tag{3.46}$$

where $R = (\log_2 M)/n$ is the code rate.

Substituting Equation (3.46) into Equation (3.45) and minimizing, we see that the minimum is at $\lambda = \alpha$ and has a constant value independent of α,

$$\beta(\alpha) = (1-R)\ln 2 \tag{3.47}$$

This makes Equation (3.43) independent of α, and makes it possible to simplify Equations (3.42) and (3.43) to (see Appendix B):

$$\begin{aligned} \bar{P}_e &\leq (1 + nC_n)e^{-nE(s)} \\ E(s) &= \frac{s}{1-s}(1-R)\ln 2 - \ln \int_0^{\infty} \left[P_0(y)^{1-s} + P_1(y)^{1-s}\right]^{1/(1-s)} dy \\ &\quad \text{for any } s \text{ in the range } 0 \leq s \leq \tfrac{1}{2} \end{aligned} \tag{3.48}$$

Thus, for any given value of s, $E(s)$ is linearly related to R with a slope of $-s\ln 2/(1-s)$. Figure 3.2 illustrates the relation of $E(s)$ to R with s as a parameter. The upper envelope of this family of curves gives the desired exponent of $\bar{P}_e$ as a function of R.

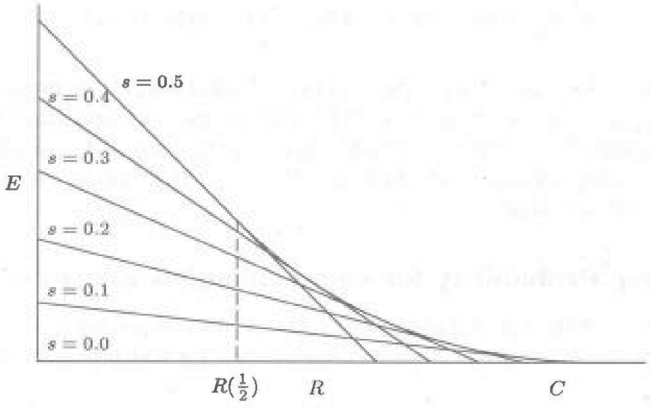
A parametric pair of equations for this envelope can be found by setting the partial derivative of $E(s)$ with respect to s equal to 0. This yields the relationship

$$R(s) = 1 - \frac{(1-s)^2\gamma'(s)}{\ln 2}; \quad 0 \leq s \leq \tfrac{1}{2} \tag{3.49}$$

$$E(s) = s(1-s)\gamma'(s) - \gamma(s) \tag{3.50}$$

where

$$\gamma(s) = \ln \int_0^{\infty} \left[P_0(y)^{1-s} + P_1(y)^{1-s}\right]^{1/(1-s)} dy \tag{3.51}$$



$$E(s,R) = \frac{s}{1-s}(1-R)\ln 2 - \gamma(s)$$
$$\gamma(s) = \ln \int_0^\infty \left[P_0(y)^{1-s} + P_1(y)^{1-s}\right]^{\frac{1}{1-s}} dy$$

Figure 3.2: Family of curves relating exponent to rate for equiprobable ensemble of codes.

It can be shown that: $R(s)$ decreases with s, $E(s)$ increases with s, the slope of $E(s)$ as a function of $R(s)$ is $(-s\ln 2)/(1-s)$, and $\lim_{s\to 0} R(s)$ is equal to channel capacity. For those values of R less than $R(\frac{1}{2})$, the E vs. R curve is given by Equation (3.48) with $s = \frac{1}{2}$

$$E = (1-R)\ln 2 - \ln \int_0^\infty \left[\sqrt{P_0(y)} + \sqrt{P_1(y)}\right]^2 dy; \quad \text{for } R < R(\tfrac{1}{2}) \qquad (3.52)$$

The E vs. R curve yielded by Equations (3.49) to (3.52) is the same as that found by Fano [4] except for some small changes in terminology. These equations simplify even further in the special case of the binary symmetric channel (See Figure 3.1). In this case,

$$\gamma(s) = \frac{1}{1-s} \ln\left[(1-p)^{1-s} + p^{1-s}\right]$$

and after some straightforward manipulation we get the familiar results,

$$R(s) = 1 - \frac{H(p_s)}{\ln 2} \qquad (3.53)$$
$$E(s) = p_s \ln\frac{1}{p} + (1-p_s)\ln\frac{1}{1-p} - H(p_s) \qquad (3.54)$$

where

$$p_s = \frac{p^{1-s}}{p^{1-s} + (1-p)^{1-s}}; \quad 0 \le s \le \tfrac{1}{2}$$

$$E = (1-R)\ln 2 - 2\ln\bigl(\sqrt{p} + \sqrt{1-p}\bigr); \quad \text{for } R \le R(\tfrac{1}{2}) \tag{3.55}$$

We have seen that for this equiprobable ensemble of codes the value of λ that yields the largest contribution to $\bar{P}_e$ is equal to α, which is given in Equation (3.44) and simplifies for the equiprobable ensemble (see Appendix B) to:

$$\alpha(s) = \frac{\displaystyle\int_0^\infty \bigl(P_0^{1-s} + P_1^{1-s}\bigr)^{1/(1-s)} \left[\frac{2(P_0P_1)^{1-s}}{\bigl(P_0^{1-s} + P_1^{1-s}\bigr)^2}\right] dy}{\displaystyle\int_0^\infty \bigl(P_0^{1-s} + P_1^{1-s}\bigr)^{1/(1-s)}\, dy} \tag{3.56}$$

One curious consequence of this is as follows: Suppose we had a way of increasing the minimum distance of a typical randomly chosen code. This would have negligible effect on the probability of decoding error for the code on a particular channel unless the minimum distance could be made larger than $n\alpha(s)$ since that is the distance at which most of the decoding errors occur.

On the other hand, if the code rate is low enough, the minimum distance can be made sufficiently large to change the exponent of $\bar{P}_e$. In Chapter 2, it was shown that the ensemble of random parity-check codes could be expurgated to include only codes with minimum distance at least $n\lambda_0$ where

$$H(\lambda_0) = (1-R)\ln 2$$

Minimizing $\beta(\alpha)$ in Equation (3.45) for this expurgated ensemble, we get

$$\beta(\alpha) = (1-R)\ln 2 = H(\lambda_0); \quad \alpha > \lambda_0 \tag{3.57}$$

$$\beta(\alpha) = -\lambda_0 \ln\alpha - (1-\lambda_0)\ln(1-\alpha); \quad \alpha \le \lambda_0 \tag{3.58}$$

Now observe that with this expurgated ensemble we can still use the same values of s and r for a given rate as we did for the unexpurgated ensemble, and certainly get a valid exponential bound on $\bar{P}_e$. If we do this, then the exponent E will be unchanged from the unexpurgated case for rates such that $\alpha > \lambda_0$ and the exponent E will be increased when $\alpha \le \lambda_0$. It can be shown that this exponent is in fact the maximum exponent over s and r. Further, it can be shown that $\alpha = \lambda_0$ at some R_0 satisfying $0 < R_0 < R(\frac{1}{2})$ and that $\alpha < \lambda_0$ for $R < R_0$. For $R < R_0$, substituting Equation (3.58) into Equation (3.43) with $s = \frac{1}{2}$, $r = 0$ and simplifying yields

$$E = -\lambda_0 \ln \int_0^\infty 2\sqrt{P_0(y)P_1(y)}\, dy \tag{3.59}$$

where λ_0 satisfies $H(\lambda_0) = (1-R)\ln 2$. Figure 3.3 shows a sketch of the E–R curve for this expurgated case. This bound was also independently derived earlier for the binary symmetric channel by Elias in unpublished work.

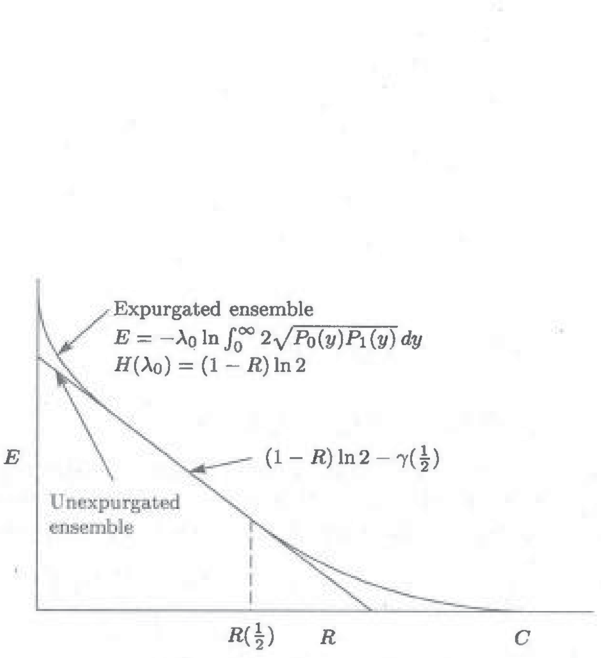


Figure 3.3: Expurgated and unexpurgated equiprobable ensemble of codes.

3.7 Binary Symmetric Channel

In order to obtain some insight into the behavior of Equations (3.42) to (3.45) for arbitrary code ensembles and, in particular, low-density code ensembles, we shall consider the binary symmetric channel (BSC) with transition probability p as shown in Figure 3.1. For this channel, the integrals in Equations (3.43) and (3.44) reduce to single terms, and we get

$$\bar{P}_e \leq (1+nC_n)\exp -nE(s,r) \tag{3.60}$$

$$E(s,r) = \frac{s}{s-r}\beta(\alpha) + \frac{r}{s-r}\ln\Big[(1-p)^{1-s}+p^{1-s}\Big] - \frac{2s}{s-r}\ln\Big[(1-p)^{\frac{1-r}{2}}+p^{\frac{1-r}{2}}\Big] \tag{3.61}$$

$$\alpha = \frac{2\big[(1-p)p\big]^{\frac{1-r}{2}}}{(1-p)^{\frac{1-r}{2}}+p^{\frac{1-r}{2}}} \tag{3.62}$$

$$\beta(\alpha) = \min_{\lambda}\big[-B(\lambda) - \lambda\ln\alpha - (1-\lambda)\ln(1-\alpha)\big] \tag{3.63}$$

In Appendix B it is shown that if $E(s,r)$ has a maximum in the region $0 < s < \infty$; $-\infty < r < 0$, then this maximum is given by

$$E = \max_{s,r} E(s,r) = p_s \ln\frac{1}{p} + (1-p_s)\ln\frac{1}{1-p} - H(p_s) \tag{3.64}$$

where p_s is the solution to the following two equations involving the unknowns p_s and p_r:

$$p_s = \frac{\lambda_0}{2} + (1-\lambda_0)p_r \tag{3.65}$$

$$H(p_s) = B(\lambda_0) + \lambda_0\ln 2 + (1-\lambda_0)H(p_r) \tag{3.66}$$

In Equations (3.65) and (3.66), λ_0 is the value of λ that maximizes

$$B(\lambda) + \frac{\lambda}{2} \ln 4p_r(1-p_r) \tag{3.67}$$

The values of s and r at which the maximum in Equation(3.64) occurs are given implicitly by

$$\begin{aligned} p_s &= \frac{p^{1-s}}{p^{1-s} + (1-p)^{1-s}} \\ p_r &= \frac{p^{1-r}}{p^{1-r} + (1-p)^{1-r}} \end{aligned} \tag{3.68}$$

The solution of Equations (3.65), (3.66), and (3.67) still involves the simultaneous solution of three equations of which two are transcendental. The advantage of these equations, however, is that they do not involve the channel transition probability p. Thus, if a solution exists to these equations, it is valid for all transition probabilities in the range

$$p_r \leq p \leq p_s \tag{3.69}$$

From Equation (3.68), this is the range of p over which $s \geq 0$ and $r \leq 0$. Figure 3.4 gives a geometrical interpretation of the exponent E in Equation (3.64) as a function of p_s and p. It is interesting to observe, also, that Equation (3.64) is identical to Equation (3.54), the exponent derived for the equiprobable ensemble, except, of course, that the value of p_s might be different. A lower bound to $\bar{P}_e$ for the best possible code of rate R can also be derived, and it has been shown [4] that Equations (3.53) and (3.54) also relate the exponent of $\bar{P}_e$ and the rate for the best possible code. Thus it is meaningful to compare codes for the BSC in terms of the parameter p_s. Equations (3.65), (3.66), and (3.67) have been solved for several expurgated ensembles of low-density parity-check codes

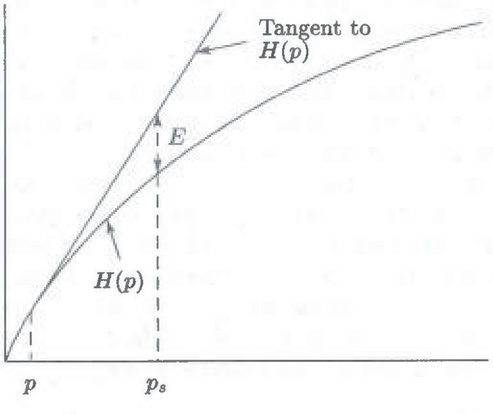


Figure 3.4: Geometric interpretation of exponent for binary symmetric channel.

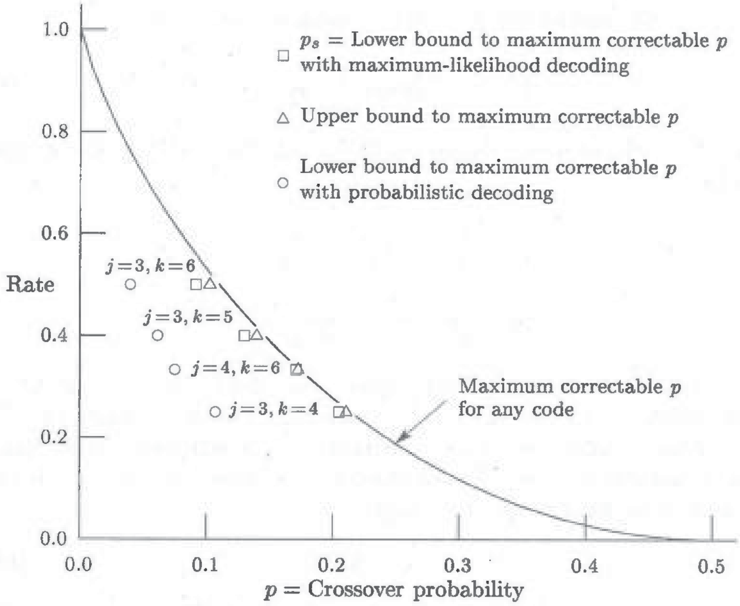


Figure 3.5: Error-correcting properties of (n, j, k) codes on BSC as function of rate for large n.

for which the function $B(\lambda)$ is bounded in Equation (2.20). Figure 3.5 presents a comparison between code rates of low-density codes and the rate of an optimum code that yields the same value of p_s and, therefore, the same exponent to $\bar{P}_e$ in the range $p_r \leq p \leq p_s$. It is interesting to observe from the comparison of Figures 2.4 and 3.5 that these codes can achieve an error probability going down exponentially with block length even when the expected number of transitions in a block is considerably more than the minimum distance. Thus although it is *possible* to make decoding errors when the number of channel transitions is half the minimum distance, decoding errors are unlikely until the number of transitions is much greater than the minimum distance. It is also interesting to observe that $\lambda_0 n$ appears to give the most likely distance between transmitted and decoded code words when decoding errors occur. More precisely, it gives the distance at which the bound to error probability is largest. It is curious that this quantity does not change as p varies between p_r and p_s.

For $p < p_r$, $E(s,r)$ is maximized by $r = 0$. This is not surprising, since from Equation (3.68), $r = 0$ for $p = p_r$. When we substitute $r = 0$ into Equation (3.61), some algebraic manipulation yields

$$\max_{s,r} E(s,r) = \min_{\lambda} -B(\lambda) + \frac{\lambda}{2} \ln \frac{1}{4p(1-p)}; \quad \text{for } p \leq p_r \tag{3.70}$$

For $p \leq p_r$, the λ that minimizes Equation (3.70) typically will decrease with p down to the code minimum distance ratio.

The preceding results have considered the case of codes and code ensembles in which Equations (3.65), (3.66), and (3.67) have a solution. Unfortunately, these equations do not have solutions for all codes. No solution corresponds to the situation in which $E(s,r)$ is maximized for $r = -\infty$. Here, Equation (3.64) is still valid for $p \leq p_s$, but p_s is now given by $\lambda_0/2$, and λ_0 is now the ratio of minimum distance to block length for the code. Physically, this means that there are so many code words at the minimum distance that error correction is unlikely when more than $n\lambda_0/2$ errors occur. One example of this is the code with only two code words, one of which is the complement of the other.

3.8 Upper Bound on Rate for Low-Density Codes

The results on error probabilities up to this point have all been upper bounds on $\bar{P}_e$. We have shown that low-density parity-check codes are at least as good on the BSC as the optimum code of a somewhat higher rate. However, there is no direct way of showing that some low-density codes are not a great deal better than the average. One small result in this direction, however, is the following theorem which shows that low-density codes cannot be used effectively on a BSC for which channel capacity is arbitrarily close to the code rate.

Theorem 3.3. *Let a parity-check code of length n and rate R containing k digits in each parity-check set be used on a BSC with crossover probability p, and let the code words be used with equal probability. Let*

$$H(p) = -p\ln p - (1-p)\ln(1-p)$$
$$p_k = \frac{1+(1-2p)^k}{2}$$

Then,

$$R > \frac{H(p_k) - H(p)}{H(p_k)} \tag{3.71}$$

implies that for a fixed k, the probability of decoding error is bounded away from 0 by an amount independent of n.

Discussion. The channel capacity of a BSC in bits per symbol is $1-[H(p)/\ln 2]$. Since $H(p_k) < \ln 2$, this theorem states that the source rate must be bounded away from the channel capacity for reliable transmission. Figure 3.5 illustrates the amount by which the capacity must exceed the source rate for several values of j and k.

Proof. Let u be a transmitted code word and let v be a received sequence. The average mutual information in bits per symbol is

$$\begin{aligned}\frac{1}{n}\overline{I(u,v)} &= -\frac{1}{n}\overline{\log_2 p(u)} + \frac{1}{n}\overline{\log_2 p_v(u)} \\ &= -\frac{1}{n}\overline{\log_2 p(v)} + \frac{1}{n}\overline{\log_2 p_u(v)}\end{aligned} \tag{3.72}$$

If the per digit equivocation satisfies the equation

$$\frac{1}{n}\overline{\log_2 p_v(u)} \geq \epsilon > 0 \tag{3.73}$$

for some ϵ independent of n, then the probability of decoding error must also remain bounded away from 0. Equation (3.73) will be established by evaluating the other terms in Equation (3.72).

Since there are 2^{nR} messages in the code set,

$$-\frac{1}{n}\overline{\log_2 p(u)} = R \tag{3.74}$$

Given the sequence u, each digit in the sequence v has probability p of being different from the corresponding digit in u, so that

$$\frac{1}{n}\overline{\log_2 p_u(v)} = \frac{-H(p)}{\ln 2} \tag{3.75}$$

Consider specifying the received sequence v by first specifying the parities of the $n(1-R)$ parity checks and then specifying the received digits in some set of nR linearly independent positions in the code. This specification is equivalent to specifying v since specifying one will make is possible to compute the other. The probability that a parity-check is satisfied is the probability that an even number of errors have occurred within the parity-check set, which is

$$\sum_{i \text{ even}} \binom{k}{i} p^i (1-p)^{k-i} = \frac{1+(1-2p)^k}{2} \tag{3.76}$$

To verify Equation (3.76), rewrite the right hand side as

$$\frac{(1-p+p)^k + (1-p-p)^k}{2}$$

and expand it in a binomial series.

The uncertainty associated with each parity-check is thus $H(p_k)/\ln 2$ bits where $p_k = [1+(1-2p)^k]/2$. Since the uncertainty associated with each information digit is at most 1 bit and dependencies can only reduce the over-all entropy, we have

$$-\frac{1}{n}\overline{\log_2 p(v)} \leq \frac{(1-R)H(p_k)}{\ln 2} + R \tag{3.77}$$

The substitution of Equations (3.74), (3.75) and (3.77) into Equation (3.72) produces

$$-\frac{1}{n}\overline{\log_2 p_v(u)} \geq \frac{H(p)}{\ln 2} - \frac{(1-R)H(p_k)}{\ln 2} \tag{3.78}$$

From the hypothesis of the theorem, there is an $\epsilon > 0$ that satisfies

$$R = \frac{H(p_k) - H(p) + \epsilon \ln 2}{H(p_k)} \tag{3.79}$$

Substituting Equation (3.79) in Equation (3.78) we obtain Equation (3.73), proving the theorem. □

4 Decoding

4.1 Introduction

Chapter 3 analyzed the probability of decoding error for (n, j, k) codes on various binary-input channels using maximum-likelihood decoding. Maximum-likelihood decoding is a convenient concept since it minimizes the probability of decoding error and thus measure the effectiveness of a code apart from any particular decoding scheme. However, implementing a maximum-likelihood decoder that actually compares the received sequence with all possible code words is a most unattractive possibility; this is particularly true for long block lengths, since the size of the code set grows exponentially with block length. A decoder that is relatively simple in terms of equipment, storage, and computation is more desirable even if it moderately increases the probability of error. If the lower probability of error is required, one can simply increase the block length of the code.

Two decoding schemes will be described here that appear to achieve a reasonable balance between complexity and probability of decoding error. The first is particularly simple but applicable only to the BSC at rates far below capacity. The second scheme, which decodes directly from the *a posteriori* probabilities at the channel output is more promising but can be understood more easily after the first scheme is described.

In the first decoding scheme, the decoder computes all the parity-checks and then changes any digit that is contained in more than some fixed number of unsatisfied parity-check equations. Using these new values, the parity checks are recomputed, and the process is repeated until the parity checks are all satisfied.

If the parity-check sets are small, this decoding procedure is reasonable, since most of the parity-check sets will contain either one transmission error or no transmission errors. Thus when most of the parity-check equations checking on a digit are unsatisfied, there is a strong indication that that digit is in error. For example, suppose a transmission error occurred in the first digit of the code in Figure 2.1. Then parity checks 1, 6, and 11 would be violated, and all three parity-check equations checking digit 1 would be violated. On the other hand, at most, one of the three equations checking on any other digit in the block would be violated.

To see how an arbitrary digit d can be corrected even if its parity-check sets contain more than one transmission error, consider the tree structure in Figure 4.1. Digit d is represented by the node at the base of the tree, and each line rising from this node represents one of the parity-check sets containing digit d. The other digits in these parity-check sets are represented by the nodes on the first tier of the tree. The lines rising from tier 1 to tier 2 of the tree represent the other parity-check sets containing the digits on tier 1, and the nodes on tier 2 represent the other digits in those parity-check sets. Notice that if such a tree is extended to many tiers, the same digit might appear in more than one place, but this will be discussed in Section 4.2

Assume now that both digit d and several of the digits in the first tier are

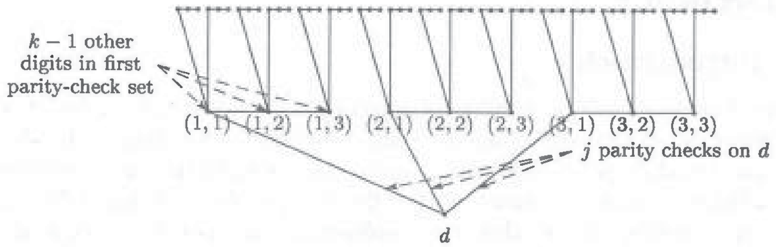


Figure 4.1: Parity-check set tree

transmission errors. Then on the first decoding attempt, the error-free digits in the second tier and their parity-check constraints will allow correction of errors in the first tier. This in turn will allow correction of digit d on the second decoding attempt. Thus digits and parity-check equations can aid in decoding a digit seemingly unconnected with them. The probabilistic decoding scheme to be described next utilizes these extra digits and extra parity-check equations more systematically.

4.2 Probabilistic Decoding

Assume that the code words from an (n, j, k) code are used with equal probability on an arbitrary binary-input channel. For any digit d, using the notation of Figure 4.1, an iteration process will be derived that on the m^{th} iteration computes the probability that the transmitted digit in position d is a 1 conditional on the received symbols out to and including the m^{th} tier. For the first iteration, we can consider digit d and the digits in the first tier to form a subcode in which all sets of these digits that satisfy the j parity-check equations in the tree have equal probability of transmission[3].

Consider the ensemble of events in which the transmitted digits in the positions of d and the first tier are independent equiprobable binary digits, and the probabilities of the received symbols in these positions are determined by the channel transition probabilities $P_x(y)$. In this ensemble the probability of any event conditional on the event that the transmitted digits satisfy the j parity-check equations is the same as the probability of an event in the subcode described above. Thus, *within this ensemble* we want to find the probability that the transmitted digit in position d is a 1 conditional on the set of received symbols $\{y\}$ and on the event S that the transmitted digits satisfy the j parity-check equations on digit d. We write this as

$$\Pr[x_d = 1 \mid \{y\}, S]$$

Using this ensemble and notation, we can prove the following theorem:

[3] An exception to this statement occurs if some linear combination of those parity-checks equations not containing d produces a parity-check equation containing only digits in the first tier. This will be discussed later but is not a serious restriction.

Theorem 4.1. *Let P_d be the probability that the transmitted digit in position d is a 1 conditional on the received digit in position d, and let $P_{i\ell}$ be the same probability for the ℓ^{th} digit in the i^{th} parity-check set of the first tier in Figure 4.1. Let the digits be statistically independent of each other, and let S be the event that the transmitted digits satisfy the j parity-check constraints on digit d. Then*

$$\frac{\Pr[x_d = 0|\{y\}, S]}{\Pr[x_d = 1|\{y\}, S]} = \frac{1 - P_d}{P_d} \prod_{i=1}^{j} \left[\frac{1 + \prod_{\ell=1}^{k-1}(1 - 2P_{i\ell})}{1 - \prod_{\ell=1}^{k-1}(1 - 2P_{i\ell})} \right] \tag{4.1}$$

In order to prove this theorem, we need the following lemma:

Lemma 4.1. *Consider a sequence of m independent binary digits in which the ℓ^{th} digit is a 1 with probability P_ℓ. Then the probability that an even number of digits are 1 is*

$$\frac{1 + \prod_{\ell=1}^{m}(1 - 2P_\ell)}{2}$$

Proof of the Lemma. Consider the function

$$\prod_{\ell=1}^{m}(1 - P_\ell + P_\ell t)$$

Observe that if this is expanded into a polynomial in t, the coefficient of t^i is the probability of i 1's. The function $\prod_{\ell=1}^{m}(1 - P_\ell - P_\ell t)$ is identical except that all the odd powers of t are negative. Adding these two functions, all the even powers of t are doubled, and the odd terms cancel out. Finally letting $t = 1$ and dividing by 2, the result is that the probability of an even number of ones. But

$$\frac{\prod_{\ell=1}^{m}(1 - P_\ell + P_\ell) + \prod_{\ell=1}^{m}(1 - P_\ell - P_\ell)}{2} = \frac{1 + \prod_{\ell=1}^{m}(1 - 2P_\ell)}{2}$$

thus proving the lemma. □

Proof of the Theorem. By the definition of conditional probabilities

$$\frac{\Pr[x_d = 0|\{y\}, S]}{\Pr[x_d = 1|\{y\}, S]} = \frac{1 - P_d}{P_d} \prod_{i=1}^{j} \frac{\Pr[S|x_d = 0, \{y\}]}{\Pr[S|x_d = 1, \{y\}]} \tag{4.2}$$

Given that $x_d = 0$, a parity check on d is satisfied if the other $(k-1)$ positions in the parity-check set contain an even number of 1's. Since all digits in the ensemble are statistically independent, the probability that all j parity-checks are satisfied is the product of the probabilities of the individual checks being satisfied. Using Lemma 4.1 this is

$$\Pr[S|x_d = 0, \{y\}] = \prod_{i=1}^{j} \left[\frac{1 + \prod_{\ell=1}^{j}(1 - 2P_{i\ell})}{2} \right] \tag{4.3}$$

Similarly,

$$\Pr[S|x_d = 1, \{y\}] = \prod_{i=1}^{j} \left[\frac{1 - \prod_{\ell=1}^{j}(1 - 2P_{i\ell})}{2} \right] \tag{4.4}$$

Substituting Equations (4.3) and (4.4) into Equation (4.2) we get the statement of the theorem. □

Judging from the complexity of this result, it would appear difficult to compute the probability that the transmitted digit in position d is a 1 conditional on the received digits in two or more tiers of the tree in Figure 4.1. Fortunately, however, the many-tier case can be solved from the 1-tier case by a simple iterative technique.

Consider first the 2-tier case. We can use Theorem 4.1 to find the probability that each of the transmitted digits in the first tier of the tree is a 1 conditional on the received digits in the second tier. The only modification of the theorem is that the first product is taken over only $j - 1$ terms, since the parity-check set containing digit d is not included. Now these probabilities can be used in Equation (4.1) to find the probability that the transmitted digit in position d is 1. The validity of the procedure follows immediately from the independence of the new values of $P_{i\ell}$ in the ensemble used in Theorem 4.1. By induction, this iteration process can be used to find the probability that the transmitted digit d is 1, given any number of tiers of distinct digits in the tree.

The general decoding procedure for the entire code may now be stated. For each digit and each combination of $j-1$ parity-check sets containing that digit, use Equation (4.1) to calculate the probability of a transmitted 1 conditional on the received symbols in the $j - 1$ parity-check sets. Thus there are j different probabilities associated with each digit, each one omitting 1 parity-check set. Next these probabilities are used in Equation (4.1) to compute a second-order set of probabilities. The probability to be associated with one digit in the computation of another digit d is the probability found in the first iteration, omitting the parity-check set containing d. If the decoding is successful, then the probabilities associated with each digit approach 0 or 1 (depending on the transmitted digit) as the number of iterations is increased. The procedure is valid only for as many iterations as meet the independence assumption in Theorem 4.1. This assumption breaks down when the tree closes upon itself. Since each tier of the tree contains $(j - 1)(k - 1)$ times more nodes than the previous tier, the independence assumption must break down while m is quite small for any code of reasonable block length. This lack of independence can be ignored, however, on the reasonable assumption that the dependencies have a relatively minor effect and tend to cancel each other out somewhat. Also, even if dependencies occur in the m^{th} iteration, the first $m - 1$ iterations have reduced the equivocation in each digit. Then we can consider the probabilities after the $m - 1$ iterations to be a new received sequence that should be easier to decode than the original received sequence.

The most significant feature of this decoding scheme is that the computation per digit per iteration is independent of the block length. Furthermore, it can be shown that the average number of iterations required to decode is bounded by a quantity proportional to the log of the log of the block length.

For the actual computation of the probabilities in Theorem 4.1, it appears to be more convenient to use Equation (4.1) in terms of log-likelihood ratios. Let

$$\begin{aligned} \ln \frac{1-P_d}{P_d} &= \alpha_d \beta_d \\ \ln \frac{1-P_{i\ell}}{P_{i\ell}} &= \alpha_{i\ell} \beta_{i\ell} \\ \ln \frac{\Pr[x_d = 0 | \{y\}, S]}{\Pr[x_d = 1 | \{y\}, S]} &= \alpha'_d \beta'_d \end{aligned} \tag{4.5}$$

where α is the sign and β is the magnitude of the log-likelihood ratio. After some manipulation, Equation (4.1) becomes

$$\alpha'_d \beta'_d = \alpha_d \beta_d + \sum_{i=1}^{j} \left(\prod_{\ell=1}^{k-1} \alpha_{i\ell} \right) f \left[\sum_{\ell=1}^{k-1} f(\beta_{i\ell}) \right] \tag{4.6}$$

where

$$f(\beta) = \ln \frac{e^\beta + 1}{e^\beta - 1}$$

The calculation of the log-likelihood ratios in Equation (4.6) for each digit can be performed either serially in time or by parallel computations. The serial computation can be programmed for a general-purpose computer, and the experimental data in Chapter 6 was obtained in this manner. For fast decoding, parallel computing is more promising, and Figure 4.2 sketches a simplified block diagram showing how this can be done.

If the input to the decoder is in the form of a log-likelihood ratio, the first row of boxes in Figure 4.2 computes $f(\beta)$ for each digit, corresponding to the rightmost operation in Equation (4.6). The output from the adders on the next row is $\sum_{\ell=1}^{k-1} f(\beta_{i\ell})$, corresponding to the two rightmost operations in Equation (4.6). Likewise, successive rows in Figure 4.2 correspond to operations in Equation (4.6) working to the left. Clearly, Figure 4.2 omits some details, such as the operations on the signs of the log-likelihood ratios with each digit, but these create no essential difficulty.

We see from Figure 4.2 that a parallel computer can be simply instrumented requiring principally a number proportional to n of analogue adders, modulo 2 adders, amplifiers and nonlinear circuits to approximate the function $f(\beta)$. How closely this function must be approximated is a subject for further study, but there are indications that it is not critical[4].

[4] Some recent experimental work indicates that if computation is strictly digital, 6 significant bits are sufficient to represent $f(\beta)$ without appreciable effect on the probability of decoding error.

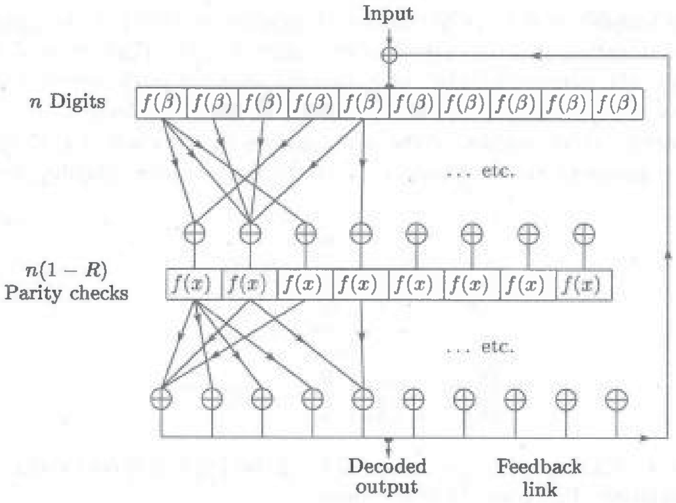


Figure 4.2: Decoding apparatus

4.3 Probability of Error Using Probabilistic Decoding

A mathematical analysis of probabilistic decoding is difficult, but a very weak bound on the probability of error can be derived easily.

Assume a BSC with crossover probability p_0 and assume first an (n, j, k) code with $j = 3$ parity-check sets containing each digit. Consider a parity-check set tree, as in Figure 4.1, containing m independent tiers, but let the tiers be numbered from top to bottom so that the uppermost tier is the 0 tier and the digit to be decoded is tier m.

Modify the decoding procedure as follow: If both parity checks corresponding to the branches rising from a digit in the first tier are unsatisfied change the digit; using these changed digits in the first tier, perform the same operation on the second tier, and continue this procedure down to digit d.

The probability of decoding error for digit d after this procedure is an upper bound to that resulting form making a decision after the m^{th} iteration of the probabilistic decoding scheme. Both procedures base their decision only on the received symbols in the m-tier tree, but the probabilistic scheme makes the most likely decision from this information.

We now determine the probability that a digit in the first tier is in error after we apply the modified decoding procedure described above. If the digit is received in error (an event of probability p_0) then a parity check constraining that digit will be unsatisfied if and only if an even number (including zero) of errors among the other $k - 1$ digits in the parity-check set. From Lemma 4.1, the probability of an even number of errors among $k - 1$ digits is

$$\frac{1 + (1 - 2p_0)^{k-1}}{2} \tag{4.7}$$

Since an error will be corrected only if both parity checks rising from the digit are unsatisfied, the following expression gives the probability that a digit in the first tier is received in error and then corrected.

$$p_0 \left[\frac{1 + (1 - 2p_0)^{k-1}}{2} \right]^2 \tag{4.8}$$

By the same reasoning, Equation (4.9) gives the probability that a digit in the first tier is received correctly but then changed because of unsatisfied parity checks.

$$(1 - p_0) \left[\frac{1 - (1 - 2p_0)^{k-1}}{2} \right]^2 \tag{4.9}$$

If we combine Equations (4.8) and (4.9), the probability of error of a digit in the first tier after applying this decoding process is

$$p_1 = p_0 - p_0 \left[\frac{1 + (1 - 2p_0)^{k-1}}{2} \right]^2 + (1 - p_0) \left[\frac{1 - (1 - 2p_0)^{k-1}}{2} \right]^2 \tag{4.10}$$

By induction it easily follows that if p_i is the probability of error after processing of a digit in the i^{th} tier, then

$$p_{i+1} = p_0 - p_0 \left[\frac{1 + (1 - 2p_i)^{k-1}}{2} \right]^2 + (1 - p_0) \left[\frac{1 - (1 - 2p_i)^{k-1}}{2} \right]^2 \tag{4.11}$$

We now show that for sufficiently small p_0, the sequence $[p_i]$ converges to 0. Consider Figure 4.3, which is a sketch of p_{i+1} as a function of p_i. Since the ordinate for one value of i is the abscissa for the next, the dotted zigzag line illustrates a convenient graphical method of finding p_i for successive values of i.

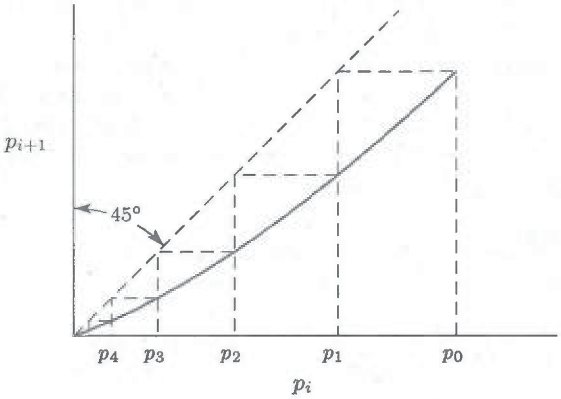


Figure 4.3: Sketch of p_{i+1} as a function of p_i.

It can be seen from Figure 4.3 that if

$$\begin{aligned} 0 < p_{i+1} < p_i; &\quad \text{for } 0 < p_i \le p_0 \\ p_{i+1} = p_i; &\quad \text{for } p_i = 0 \end{aligned} \tag{4.12}$$

then the sequence $[p_i] \to 0$. It can be seen from Equation (4.11) that for p_0 sufficiently small, the inequality (Equation (4.12)) is satisfied. Figure 4.4 gives the maximum p_0 for several values of k.

j	k	Rate	p_0
3	6	0.5	0.0395
3	5	0.4	0.0612
4	6	0.333	0.0748
3	5	0.25	0.1069

Figure 4.4: Maximum p_0 for weak bound decoding convergence.

The rate at which $[p_i] \to 0$ may be determined by noting from Equation (4.11) that for small p_i

$$p_{i+1} \approx p_i 2(k-1)p_0 \tag{4.13}$$

From this it is easy to show that for sufficiently large i,

$$p_i \approx c[2(k-1)p_0]^i \tag{4.14}$$

where c is a constant independent of i. Since the number of independent tiers in the tree increases logarithmically with block length, this bound to the probability of decoding error approaches zero with some small negative power of the block length. This slow approach to zero appears to be a consequence of the modification of the decoding scheme and of the strict independence requirement, rather than of probabilistic decoding as a whole.

This same argument can be applied to codes with more than 3 parity-check sets per digit. Stronger results will be achieved if for some integer b, to be determined later, a digit is changed whenever b or more parity-check constraints rising from the digit are violated. Using this criterion and following the reasoning leading to Equation (4.11), we obtain

$$\begin{aligned} p_{i+1} = p_0 - p_0 \sum_{\ell=b}^{j-1} \binom{j-1}{\ell} \left[\frac{1+(1-2p_i)^{k-1}}{2}\right]^{\ell} \left[\frac{1-(1-2p_i)^{k-1}}{2}\right]^{j-1-\ell} \\ + (1-p_0) \sum_{\ell=b}^{j-1} \binom{j-1}{\ell} \left[\frac{1-(1-2p_i)^{k-1}}{2}\right]^{\ell} \left[\frac{1+(1-2p_i)^{k-1}}{2}\right]^{j-1-\ell} \end{aligned} \tag{4.15}$$

The integer b can now be chosen to minimize p_{i+1}. The solution to this minimization is the smallest integer b for which

$$\frac{1-p_0}{p_0} \le \left[\frac{1+(1-2p_i)^{k-1}}{1-(1-2p_i)^{k-1}}\right]^{2b-j+1} \tag{4.16}$$

From this equation, is it seen that as p_i decreases, b also decreases. Figure 4.5 sketches p_{i+1} as a function of p_i when b is changed according to Equation (4.16). The breakpoints in the figure represent changes in b.

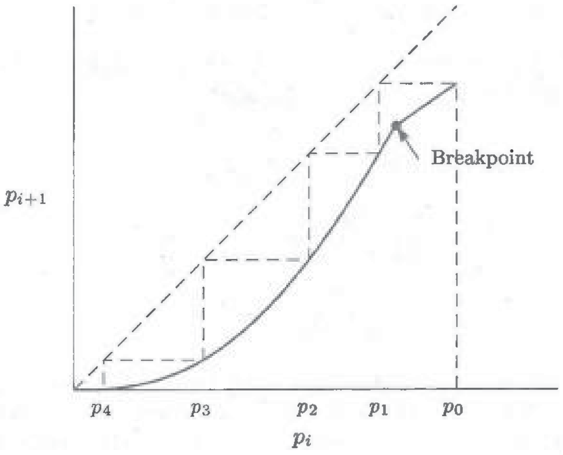


Figure 4.5: Behavior of decoding iterations for $j > 3$.

The proof that the probability of decoding error approaches 0 with an increasing number of iterations for sufficiently small p_0 is the same as before. The asymptotic approach of the sequence $[p_i]$ to 0 is different, however. From Equation (4.16), if p_i is sufficiently small, b takes the value $j/2$ for j even and $(j+1)/2$ for j odd. Using these values of b and expanding Equation (4.15) in a power series in p_i,

$$p_{i+1} = p_0 \binom{j-1}{\frac{j-1}{2}} (k-1)^{\frac{j-1}{2}} p_i^{\frac{j-1}{2}} + \text{higher order terms}; \quad j \text{ odd} \tag{4.17}$$

$$p_{i+1} = p_0 \binom{j-1}{\frac{j}{2}} (k-1)^{\frac{j}{2}} p_i^{\frac{j}{2}} + \text{higher order terms}; \quad j \text{ even} \tag{4.18}$$

Using this, it can be shown that for a suitably chosen positive constant c_{jk} and sufficiently large i

$$\begin{aligned} p_i &\leq \exp\Big[-c_{jk}\Big(\frac{j-1}{2}\Big)^i\Big]; \quad j \text{ odd} \\ p_i &\leq \exp\Big[-c_{jk}\Big(\frac{j}{2}\Big)^i\Big]; \quad j \text{ even} \end{aligned} \tag{4.19}$$

It is interesting to relate this result to the block length of the code. Since there are $(j-1)^m(k-1)^m$ digits in the m^{th} tier of a tree n must be at least this big, giving the left side of Equation (4.20). On the other hand, a specific procedure is described in Appendix C for constructing codes satisfying the right side of

Equation (4.20).

$$\frac{\ln n}{\ln(j-1)(k-1)} \geq m \geq \frac{\ln\left[\frac{n}{2k} - \frac{n}{2j(k-1)}\right]}{2\ln(j-1)(k-1)} \tag{4.20}$$

When we combine Equations (4.19) and (4.20), the probability of decoding error for a code satisfying Equation (4.20) is bounded by

$$\begin{aligned} P_m &\leq \exp -c_{jk}\left[\frac{n}{2k} - \frac{n}{2j(k-1)}\right]^{\alpha} \\ \alpha &= \frac{\ln\frac{j-1}{2}}{2\ln(j-1)(k-1)} \quad j \text{ odd} \\ \alpha &= \frac{\ln\frac{j}{2}}{2\ln(j-1)(k-1)} \quad j \text{ even} \end{aligned} \tag{4.21}$$

For $j > 3$, this probability of decoding error bound decreases exponentially with a root on n. Observe that if the number of iterations m which can be made without dependencies were $[2\ln(k-1)(j-1)]/(\ln j/2)$ times larger, then the probability of decoding error would decrease exponentially with n. It is hypothesized that using the probabilistic decoding scheme and continuing to iterate after dependencies occur will produce this exponential dependence.

A second way to evaluate the probabilistic decoding scheme is to calculate the probability distributions of log-likelihood ratios in Equation (4.6) for a number of iterations. This approach makes it possible to find whether a code of given j and k is capable of achieving arbitrarily small error probability on any given channel. With the aid of the IBM 7090 computer, it was found that a code with $j = 3$, $k = 6$ is capable of handling transition probabilities up to 0.07 and with $j = 3$, $k = 4$, transition probabilities up to 0.144 can be handled. These figures are particularly interesting since they disprove the common conjecture that the computational cutoff rate of sequential decoding [17] bounds the rate at which any simple decoding scheme can operate.

5 Low-Density Codes with Arbitrary Alphabet Sizes

The results of Chapters 2, 3, and 4 concerning binary low-density parity-check codes will be extended in this chapter to codes with an arbitrary alphabet size. The letters in the alphabet will be A'nary digits, where A is the alphabet size and the A'nary digits are numbers from 0 to $A-1$ inclusive. The definitions of (n,j,k) parity-check matrices and ensembles of matrices are the same here as in Chapter 2. The code words going with such a matrix will be sequences of A'nary digits such that the sum of the digits within any parity-check set is zero modulo A.

5.1 Distance Functions

Define the distance between two sequences in a code of alphabet size A as the number of positions in which the sequences differ. The weight of a sequence is the number of nonzero digits or the distance from the all-zero sequence. The distance function $N(\ell)$ of a code is again defined as the number of code words of weight ℓ. It follows from the group properties of such a code [12] that $N(\ell)$ is the number of words at distance ℓ from any given code word. In order to upper bound $N(\ell)$ for these codes, we need the following theorem, which is a direct extension of Theorem 2.3

Theorem 5.1. *For each code in an (n,j,k) ensemble with alphabet size A, the number $N_1(\ell)$ of sequences of weight ℓ that satisfies any one of the j blocks of n/k checks is bounded by*

$$N_1\left[\frac{n}{k}\mu_A'(s)\right] \le \exp\frac{n}{k}\left[\mu_A(s) - s\mu_A'(s) + (k-1)\ln A\right] \tag{5.1}$$

where s is an arbitrary parameter and $\mu_A(s)$ is defined by

$$\begin{aligned}\mu_A(s) &= \ln A^{-k}\Big\{\left[1+(A-1)e^s\right]^k + (A-1)(1-e^s)^k\Big\}\\ \mu_A'(s) &= \frac{d\mu_A(s)}{ds}\end{aligned} \tag{5.2}$$

Proof. Consider a particular check set of k digits. Let $m(\ell)$ be the number of different sequences of k A'nary digits of weight ℓ that sum to 0 modulo A. We shall first show that for arbitrary t

$$\sum_{\ell=0}^{k} m(\ell)t^\ell = \frac{1}{A}\left[1+(A-1)t\right]^k + \frac{A-1}{A}(1-t)^k \tag{5.3}$$

Consider the double enumerating function

$$B(t,r) = \left(1+tr+tr^2+\cdots+tr^{A-1}\right)^k \tag{5.4}$$

$$= \sum_{\ell,j} b_{\ell j}t^\ell r^j \tag{5.5}$$

Clearly $b_{\ell j}$ is the number of sequences of length k containing ℓ nonzero A'nary digits that sum to j. Now consider the expression

$$\frac{1}{A}\sum_{a=0}^{A-1} B\Big(t, e^{\frac{i2\pi a}{A}}\Big) = \sum_{\ell,j} b_{\ell j} t^{\ell} \Big(\frac{1}{A}\sum_{a=0}^{A-1} \exp\frac{ij2\pi a}{A}\Big) \tag{5.6}$$

The term in parenthesis in Equation (5.6) sums to 0 for all j that are not multiples of A due to the uniform spacing of the terms around the unit circle of the complex plane. If j is a multiple of A, the bracketed term sums to 1. Thus

$$\frac{1}{A}\sum_{a=0}^{A-1} B\Big(t, e^{\frac{i2\pi a}{A}}\Big) = \sum_{\ell=0}^{k} m(\ell) t^{\ell} \tag{5.7}$$

Finally for $r \neq 1$, from Equation (5.4) we get

$$B(t,r) = \Big[1 + t\Big(\frac{r - r^A}{1-r}\Big)\Big]^k \tag{5.8}$$

$$B\Big(t, e^{\frac{i2\pi a}{A}}\Big) = \begin{cases} (1-t)^k; & a \neq 0 \\ \big[1 + (A-1)t\big]^k; & a = 0 \end{cases} \tag{5.9}$$

Combining Equations (5.9) and (5.7), we get Equation (5.3).

Now consider an ensemble in which all n-length A'nary sequences that satisfy the given n/k parity checks are equally likely. Then, over any k digits in a check set, each of the A^{k-1} k-length sequences satisfying the check are equally likely, and from Equation (5.3), the moment-generating function for the weights of these sequences is

$$g(s) = A^{-k}\Big\{\big[1 + (A-1)e^s\big]^k + (A-1)\big(1 - e^s\big)^k\Big\} \tag{5.10}$$

Now the theorem follows in exactly the same way as in Theorem 2.3. □

There are altogether $\binom{n}{\ell}(A-1)^{\ell}$ A'nary n-length sequences of weight ℓ, so that the probability that a randomly chosen sequence of weight ℓ will satisfy the block of n/k parity checks is

$$\frac{N_1(\ell)}{\binom{n}{\ell}(A-1)^{\ell}}$$

Since over the ensemble of codes, each of the j blocks of parity checks is independent, the probability $P(\ell)$ that a sequence of weight ℓ is a code word is

$$P(\ell) = \left[\frac{N_1(\ell)}{\binom{n}{\ell}(A-1)^{\ell}}\right]^j$$

Thus, following Chapter 2, the distance function $\overline{N_{jk}(\ell)}$ and the minimum distance distribution function can be bounded by

$$\overline{N_{jk}(n\lambda)} \le C(\lambda, n)\exp -nB_{jkA}(\lambda) \tag{5.11}$$

$$\Pr(D \le n\delta) \le \sum_{\ell=2}^{n\delta} C(\lambda, n)\exp -nB_{jkA}\Big(\frac{\ell}{n}\Big) \tag{5.12}$$

where

$$B_{jkA}(\lambda) = (j-1)\big[H(\lambda) + \lambda\ln(A-1)\big] - \frac{j}{k}\big[\mu_A(s) + (k-1)\ln A\big] + js\lambda \tag{5.13}$$

$$C(\lambda, n) = \big[2\pi n\lambda(1-\lambda)\big]^{\frac{j-1}{2}}\exp\frac{j-1}{12n\lambda(1-\lambda)} \tag{5.14}$$

$$\lambda = \frac{\mu'_A(s)}{k} \tag{5.15}$$

and $\mu_A(s)$ is given by Equation (5.2).

It can be shown by methods similar to those of Appendix A that the function $B_{jkA}(\lambda)$ is 0 at $\lambda = 0$; it rises with an initial infinite slope, has one zero crossing at the typical minimum distance, and then remains negative.

5.2 Probability of Decoding Error

Consider a channel with an input alphabet of A letters which for convenience we take to be A'nary digits. Let the output be y and, as in Chapter 3, let $f(y)$ be an arbitrary function of the output. Let $u_0, u_1, \ldots, u_j, \ldots, u_{M-1}$, where $u_j = x_{1j}, x_{2j}, \ldots, x_{nj}$ be the M code words of an A'nary block code of length n. Define the discrepancy between an input word $u = (x_1, \ldots, x_n)$ and an output $v = (y_1, \ldots, y_n)$ as

$$D(u, v) = \sum_{i=1}^{n} \delta(x_i, y_i) \tag{5.16}$$

where

$$\delta(x, y) = \ln\frac{P(y|x)}{f(y)} \tag{5.17}$$

Define

$$g_i(s) = \overline{\exp s\delta_i} \tag{5.18}$$

$$h_{ij}(r, t) = \overline{\exp r\delta_i + t(\delta_j - \delta_i)} \tag{5.19}$$

The averaging in Equations (5.18) and (5.19) is according to the distribution of channel outputs y conditioned on x_i being transmitted. Now we restrict our attention to channels which are symmetrical in the sense that $g_i(s)$ and $h_{ij}(r, t)$

are independent of i and j for an appropriate choice of $f(y)$. Further, we restrict our attention to $f(y)$ functions for which this symmetry is achieved.

One example of such a channel is a channel with A'nary digits for both input and output and a probability $1-p$ of receiving the transmitted digit and $p/(A-1)$ of receiving any other digit. Another example is that of A orthogonal equal energy signals on either a white Gaussian noise channel or a Rayleigh fading channel similar to Figure 3.1.

Maximum-likelihood decoding on this channel is equivalent to choosing u_i that minimizes $D(u_i, v)$ when v is the received word. Thus, when u_0 is transmitted, we can bound the probability of maximum-likelihood decoding error by

$$P(e) \le P_1 + P_2 \tag{5.20}$$

$$P_1 \le \Pr\left[\sum_{i=1}^{n} \delta(x_{i0}, y_i) \ge nd\right] \tag{5.21}$$

$$P_2 \le \sum_{j=1}^{M-1} \Pr\left[\sum_{i=1}^{n} \delta(x_{i0}, y_i) < nd; \quad \sum_{i=1}^{n} \delta(x_{ij}, y_i) - \delta(x_{i0}, y_i) < 0\right] \tag{5.22}$$

Theorems 3.1 and 3.2 can now be used directly to bound Equations (5.21) and (5.22).

$$P_1 \le \left[g(s)\right]^n \exp(-nsd); \quad s \ge 0 \tag{5.23}$$

$$P_2 \le \sum_{\ell=0}^{n} N(\ell)\left[h(r,t)\right]^{\ell}\left[h(r,0)\right]^{n-\ell} \exp(-nrd) \quad r \le 0, t \le 0 \tag{5.24}$$

where $g(s)$ and $h(r,t)$ are given by Equations (5.18) and (5.19) and $N(\ell)$ is the distance function of the code. For an ensemble of parity-check codes, Equations (5.20), (5.23), and (5.24) bound the average probability of decoding error over the ensemble in terms of the arbitrary parameters d, $f(y)$, $s \ge 0$, $r \le 0$, $t \le 0$. As in Chapter 3, $t = (r-1)/2$ optimizes the bound over t, but no other simplification has been found. Equations (5.20), (5.23), and (5.24) are sufficient, however, in conjunction with Equation (5.11) to demonstrate the exponential decrease of probability of error with block length for an expurgated ensemble of (n, j, k) codes at sufficiently low rates.

5.3 Probabilistic Decoding

Consider an (n, j, k) code of A'nary digits, and assume that the code words have equal probability. As in Chapter 4, using the notation of Figure 4.1, we wish to find $P_m(x_d = a)$, the probability that the transmitted digit in position d was an a, $0 \le a \le A-1$, given the received symbols in the m tiers of the parity-check set tree on digit x_d. First we shall find $P_1(x_d = a)$.

Consider the ensemble in which the transmitted digits in position d and the first tier are independent equiprobable A'nary digits, and the received digits are determined according to the channel. Within this ensemble, the probability of any event conditional on the j parity checks of the first tier being satisfied is the same as the probability of the event in the actual code. Thus, using our previous notation,

$$P_1(x_d = a) = \Pr(x_d = a|\{y\}, S) \tag{5.25}$$

Theorem 5.2. *Let $P_0(x_{i\ell} = a)$ be the probability that the ℓ^{th} transmitted A'nary digit in the i^{th} parity-check set on d is a, given the received symbol in that position. Assume that all combinations of x_d and the $x_{i\ell}$ that satisfy the j parity checks on x_d are equally likely. Then*

$$P_1(x_d = a) = \frac{P_0(x_d = a)\prod_{i=1}^{j} g_i(-a)}{\sum_{a=0}^{A-1} P_0(x_d = a)\prod_{i=1}^{j} g_i(-a)} \tag{5.26}$$

where

$$G_i(t) = \sum_{a=0}^{A-1} g_i(a)t^a = \prod_{\ell=1}^{k-1}\sum_{a=0}^{A-1} P_0(x_{i\ell} = a)t^a \tag{5.27}$$

In Equation (5.26), $-a$ is taken modulo A, and the multiplication in Equation (5.27) is taken modulo t^A.

Equation (5.27) yields an explicit solution for $g_i(a)$ for each i, but computationally, $g_i(a)$ is found for all a, $0 \le a \le A-1$ simultaneously. Before proving this theorem, the following lemma is needed.

Lemma 5.1. *Consider a sequence of L statistically independent A'nary digits in which the ℓ^{th} letter assumes the value a with probability $P_\ell(a)$. Then the probability that the modulo A sum of the digits has the value a is given by $g(a)$ in the expansion*

$$G(t) = \sum_{a=0}^{A-1} g(a)t^a = \prod_{\ell=1}^{L}\sum_{a=0}^{A-1} P_\ell(a)t^a \tag{5.28}$$

where the product in Equation (5.28) is taken modulo t^A.

Proof of Lemma. Note that the right side of Equation (5.28) using ordinary multiplication is simply the z transform for the sum of the ℓ letters. In other words, the coefficient of t^a in the expanded form of Equation (5.28) is the probability that the sum of the digits is a. Taking the product modulo t^A simply adds all coefficients for which a has the same value modulo A, thus proving the lemma. □

Proof of Theorem. Using Equation (5.25), with some manipulation of conditional probabilities, we get

$$\begin{aligned} P_1(x_i = a) &= \frac{\Pr(S|x_d = a,\ \{y\})P_0(x_d = a)}{\Pr(S|\{y\})} \\ &= \frac{\Pr(S|x_d = a,\ \{y\})P_0(x_d = a)}{\sum_{a'=0}^{A-1}\Pr(S|x_d = a',\ \{y\})P_0(x_d = a')} \end{aligned} \tag{5.29}$$

Now we observe that the term $\Pr(S|x_d = a, \{y\})$ is the probability that each set of $k-1$ digits other than d in the parity-check sets add to $-a$. From Lemma 5.1,

$$\Pr(S|x_d = a, \{y\}) = \prod_{i=1}^{j} g_i(-a \bmod A) \tag{5.30}$$

where $g_i(-a \bmod A)$ is given by Equation (5.27). Substituting Equation (5.30) into Equation (5.29), we get the statement of the theorem. □

Equation 5.26 can be extended immediately to an iterative decoding procedure by the same arguments as used in Chapter 4. In successive iterations, $P_0(x_{i\ell} = a)$ becomes $P_m(x_{i\ell} = a)$, and j different probabilities must be calculated for each digit, each probability leaving one of the j parity checks out of consideration.

5.4 Probability of Error Using Probabilistic Decoding

Consider a channel with A inputs and A outputs both labeled from 0 to $A-1$. The channel transition probabilities are given by

$$P(y_a|x_a) = 1 - p_0; \quad P(y_a|x_b) = \frac{p_0}{A-1}; \quad \text{for any } a, b \text{ such that } a \neq b$$

Consider a parity-check set tree as in Figure 4.1 with m independent tiers numbered from top to bottom with $j = 3$. Modify the decoding procedure as follows: If both parity checks rising from a digit are unsatisfied and both have the same value, change the digit so as to satisfy both checks; otherwise leave the digit unchanged. The probability of error in this procedure overbounds that for probabilistic decoding. The probability that a digit in the first tier is received incorrectly and then corrected is P_0Q^2 where Q is the probability of either no errors or of errors adding to 0 mod A in one of the sets of $k-1$ digits. We define the error in a digit as $(y-x) \bmod A$. Next it will be shown that

$$Q = \frac{1 + (A-1)\left(1 - \frac{Ap_0}{A-1}\right)^{k-1}}{A} \tag{5.31}$$

The z transform for the ordinary sum of the errors in $k-1$ digits is

$$G(z) = \left(1 - p_0 + \frac{p_0}{A-1}\sum_{a=1}^{A-1} z^a\right)^{k-1} \tag{5.32}$$

$$G(z) = \left(1 - p_0 + \frac{p_0}{A-1}\frac{z - z^A}{1-z}\right)^{k-1}; \quad \text{for } z \neq 1 \tag{5.33}$$

$$= 1; \quad \text{for } z = 1 \tag{5.34}$$

Now consider the quantity

$$\frac{1}{A}\sum_{a=0}^{A-1} G\left(e^{\frac{j2\pi a}{A}}\right)$$

All powers of z in this expression that are not multiples of A cancel out due to their uniform spacing around the unit circle in the complex plane. The coefficients of powers that are multiples of A add, thus giving Q.

$$Q = \frac{1}{A} \sum_{a=0}^{A-1} G(e^{\frac{j2\pi a}{A}}) \tag{5.35}$$

Now from Equation (5.33)

$$G(e^{\frac{j2\pi a}{A}}) = \left(1 - p_0 + \frac{p_0}{A-1}\right)^{k-1}; \quad \text{for } a \neq 0 \tag{5.36}$$

Finally, combining Equations (5.36) and (5.34), we get Equation (5.31). Thus, the probability that a digit in the first tier is received incorrectly and then corrected is

$$p_0 \left\{ \frac{1 + (A-1)\left(1 - \frac{Ap_0}{A-1}\right)^{k-1}}{A} \right\}^2 \tag{5.37}$$

The probability of receiving a digit in the first tier correctly and then changing it due to two identically violated parity checks is

$$(1 - p_0)(1 - Q)\frac{1 - Q}{A - 1} \tag{5.38}$$

The term $(1 - Q)$ in Equation (5.38) is the probability that the errors in one set of $k - 1$ digits will not satisfy the parity, and the term $(1 - Q)/(1 - A)$ is the probability that the other set will have the same value modulo A as the first set.

Combining Equations (5.37), (5.38), and (5.31), we get the probability of error for a digit in the first tier after the first iteration of the decoding process

$$\begin{aligned} p_1 = p_0 - p_0 &\left\{ \frac{1 + (A-1)\left(1 - \frac{Ap_0}{A-1}\right)^{k-1}}{A} \right\}^2 \\ &+ (1 - p_0)(A - 1)\left\{ \frac{1 - (A-1)\left(1 - \frac{Ap_0}{A-1}\right)^{k-1}}{A} \right\}^2 \end{aligned} \tag{5.39}$$

Similarly, for successive tiers,

$$\begin{aligned} p_1 = p_0 - p_0 &\left\{ \frac{1 + (A-1)\left(1 - \frac{Ap_i}{A-1}\right)^{k-1}}{A} \right\}^2 \\ &+ (1 - p_0)(A - 1)\left\{ \frac{1 - (A-1)\left(1 - \frac{Ap_i}{A-1}\right)^{k-1}}{A} \right\}^2 \end{aligned} \tag{5.40}$$

The rate at which $[p_i] \to 0$ can be determined from Equation (5.40). For p_i small,

$$p_{i+1} \approx p_i 2(k-1)p_0 \tag{5.41}$$

It is interesting to observe that Equation (5.41) is identical to Equation (4.14), although, of course, the maximum value of p_0 for which p_i from Equation (5.40) converge to 0 is different. This value increases with A up to $1/2(k-1)$.

A bound on decoding error with $j > 3$ is considerably more difficult. The decoding scheme will be to change a digit whenever a number b to be determined later, or more of the parity checks rising from a digit all have the same value. The digit will be changed in such a way as to satisfy the b parity checks. If $b > (j-1)/2$, it can be shown in the same way as in Section 4.3 that

$$\begin{aligned} p_{i+1} = p_0 - p_0 \sum_{\ell=b}^{j-1} \binom{j-1}{\ell} Q_i^{\ell}(1-Q_i)^{j-1-\ell} \\ + (1-p_0) \sum_{\ell=b}^{j-1} \binom{j-1}{\ell} \left(\frac{1-Q_i}{A-1}\right)^{\ell} \left(1 - \frac{1-Q_i}{A-1}\right)^{j-1-\ell} (A-1) \end{aligned} \tag{5.42}$$

where

$$Q_i = \frac{1 + (A-1)\left(1 - \frac{Ap_i}{A-1}\right)^{k-1}}{A} \tag{5.43}$$

The integer b can now be chosen to minimize p_{i+1} subject to the restriction $b > (j-1)/2$. The solution to this minimization is the smallest integer $b > (j-1)/2$ for which

$$\frac{1-p_0}{p_0} \leq \frac{Q_i^b (A-1)^{j-2}}{(1-Q_i)^{2b+1-j}(A-2-Q_i)^{j-1-b}} \tag{5.44}$$

As p_i approaches 0, $b = j/2$ for j even and $(j+1)/2$ for j odd. Then expanding Equation (5.42) in a power series in p_i, we obtain

$$p_{i+1} = p_0 \binom{j-1}{\frac{j-1}{2}} [p_i(k-1)]^{\frac{j-1}{2}} + \cdots; \quad j \text{ odd} \tag{5.45}$$

$$p_{i+1} = \left[p_0 + \frac{1-p_0}{(A-1)^b}\right] \binom{j-1}{\frac{j}{2}} [p_i(k-1)]^{\frac{j}{2}} + \cdots; \quad j \text{ even} \tag{5.46}$$

Observe that Equation (5.45) is identical to Equation (4.17) and Equation (5.46) is identical to Equation (4.18) except for the coefficient.

From Equation (4.18) on, the derivation of error probability in Chapter 4 does not use the restriction of a binary binary alphabet, and therefore the bound on error probability in Equation (4.21) is valid for codes of arbitrary alphabet size. The coefficient c_{jk} appearing in Equation (4.21) is a function of the alphabet size A, however.

6 Experimental Results

The probability of decoding error $P(e)$ associated with a coding and decoding scheme can be directly measured by simulating both the scheme and the channel of interest on a computer. Unfortunately, the experiment must be repeated until there are many decoding failures if $P(e)$ is to be evaluated with any accuracy, and thus many times $1/P(e)$ trials are necessary. For block lengths of about 500, an IBM 7090 computer requires about 0.1 second per iteration to decode a block by the probabilistic decoding scheme. Consequently, many hours of computation time are necessary to evaluate even a $P(e)$ in the order of 10^{-4}.

Because of limitations on available computer time, all of the results presented will be for situations in which $P(e)$ is large. Certainly it would be more interesting to have results for small $P(e)$. However, the data presented can probably be extrapolated with some degree of confidence to situations in which $P(e)$ is 10^{-5} or 10^{-6}. Furthermore, even the limited data presented here give some indication of the variability of $P(e)$ with such parameters as block length, code rate, and type of channel.

6.1 Code Generation

All of the results in this chapter were obtained with low-density parity-check codes generated on an IBM 7090 computer by a pseudorandom procedure. More specifically, the parity-check matrices were chosen in the same way as the ensemble of low-density matrices was generated in Chapter 2. The first submatrix of n/k parity-check sets contained successive sets of k digits, and each succeeding submatrix was a random column permutation of the first. The random permutation was performed with a pseudorandom number routine and then modified so that no two parity-check sets would contain more than one digit in common. This modification guaranteed the validity of the first iteration in the decoding process and also excluded the remote possibility of choosing a code with a minimum distance of 2.

The codes generated in this way were stored in the computer and used in decoding the noise sequences generated by simulated binary symmetric channels, white Gaussian noise channels, and Rayleigh fading channels. In order to reduce computer time, however, the code word to be transmitted was always the all-zero sequence. This is valid since, as explained in Chapter 3, the probability of decoding error on a symmetric binary-input channel is independent of the transmitted code word. This simplification, of course, requires extreme care to ensure that the actual simulation of decoding maintains complete symmetry between positive and negative outputs.

6.2 Binary Symmetric Channel

A true simulation of a binary symmetric channel (BSC) would involve choosing random error sequences in which crossovers (that is, channel errors, represented by the crossed transition lines in Figure 3.1a) occur independently with a given

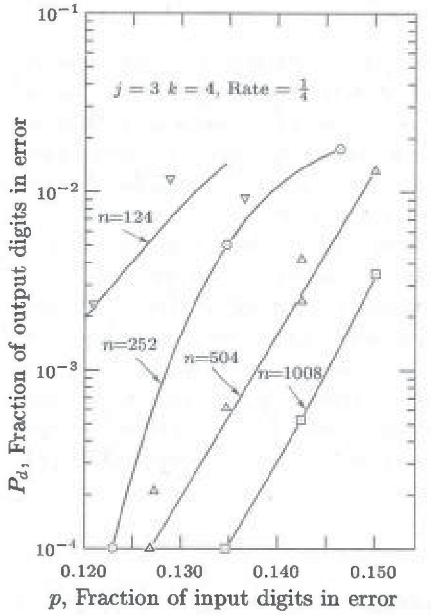


Figure 6.1: Experimental results on BSC.

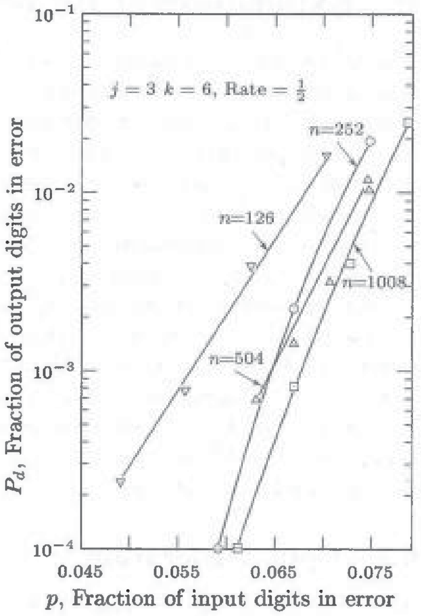


Figure 6.2: Experimental results on BSC.

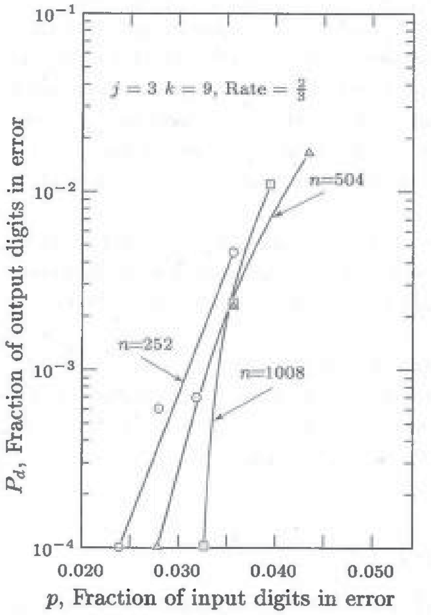


Figure 6.3: Experimental results on BSC.

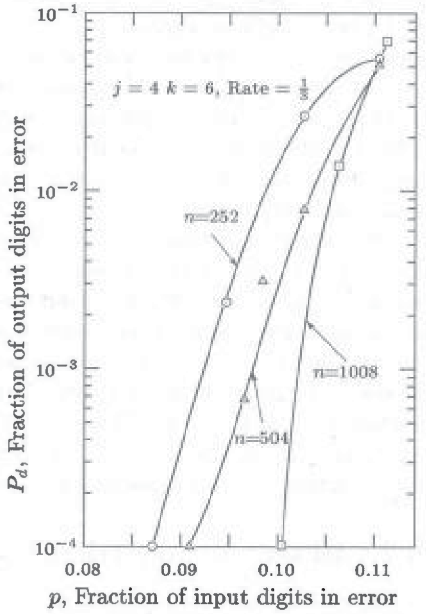


Figure 6.4: Experimental results on BSC.

probability p. Whether probabilistic decoding with a particular code can decode such a sequence depends very strongly on the number of crossovers c generated in such a process. Since c has a well known (that is, binomial) distribution, it is possible to evaluate experimentally the probability of decoding error given c crossovers and then calculate $P(e)$ for a BSC from this data. This latter procedure has the advantage of giving additional insight into the operation of the decoding scheme and also of facilitating comparison with other coding schemes that are oriented toward correcting a fixed number of crossovers.

Figures 6.1 to 6.4 present the actual data gathered this way. The abscissa on each graph is the ratio of number of crossovers to block length c/n and the ordinate is the digit error probability after decoding. In all these experiments, except for one code with a rate $\frac{1}{2}$ and block length 126, the decoder failed to decode rather than decoding to an incorrect message. In other words, the *a posteriori* probabilities computed by the decoder failed to converge to either 1 or 0. This is an important point in any communication system in which a feedback link is available since undecoded blocks of information can be retransmitted. It is important to note that $P(e)$ as shown in Figures 6.1 to 6.4 is the decoding error probability per digit that ensues when the best guess is made about each digit in blocks that can't be decoded. The probability of failure to decode a block is typically about 10 times larger than $P(e)$.

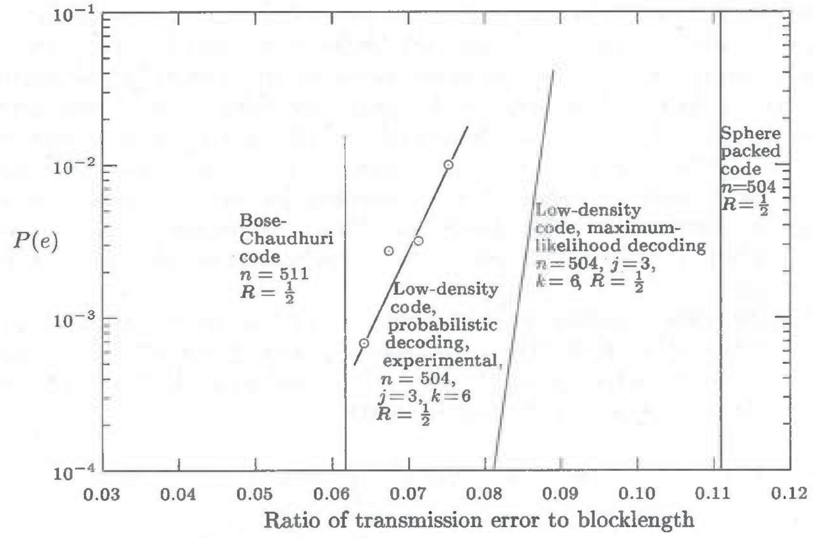


Figure 6.5: Comparison of experimental results using probabilistic decoding to theoretical results with maximum-likelihood decoding.

The median number of blocks with decoding failures per point plotted on Figures 6.1 to 6.4 is 8; many points, particularly where $P(e)$ is small, were evaluated from data containing decoding failures in only 1 or 2 blocks. Thus the position of individual points on these curves would probably change appreciably with more data.

Figure 6.5 compares the experimental data using probabilistic decoding on a code with $n = 504$, $j = 3$, and $k = 6$ to the theoretical probability of error that would result for the same code if maximum-likelihood decoding were used. For comparison purposes, a Bose-Chaudhuri code of approximately the same block length and rate is included. The value of $P(e)$ for this code assumes the use of one of the known algorithms for decoding such as Peterson's [12]. These algorithms correct only numbers of crossovers less than half the minimum distance. It appears from the curve that the Bose-Chaudhuri code would perform better at low crossover probabilities and the low-density code would perform better at high crossover probabilities.

6.3 White Gaussian Noise Channel

In the following two sections each of the channels under consideration will consist of a binary data transmitter, a physical channel, and a likelihood receiver. The output from the likelihood receiver is assumed to be the log-likelihood ratio,

$$y = \ln \frac{\Pr[x = 0|r(t)]}{\Pr[x = 1|r(t)]}$$

where x is a transmitted digit, $r(t)$ is the received waveform corresponding to that digit, and y is the output from the likelihood receiver for that digit. Of course, this output could be converted into a binary digit before attempting to decode a block of data, but this conversion would destroy some information about the transmitted sequence. Since probabilistic decoding operates naturally with log-likelihood ratios, it is natural to ask how much can be gained in terms of error probability, signal power or transmission rate by using the output of a likelihood receiver directly with the decoder rather than making binary decisions first. For both the channels considered here, this gain turns out to be of central importance.

For the white Gaussian noise channel, assume that one of two waveforms is transmitted every T seconds. These signals appear at the receiver, suitably attenuated and delayed, as two functions $x_0(t)$ and $x_1(t)$, both nonzero only from $t = 0$ to $t = T$, and both of equal energy,

$$E_c = \int_0^T x_0^2(t)\,dt = \int_0^T x_1^2(t)\,dt$$

Let $n(t)$ be a sample of white Gaussian noise of power density N_0 per unit bandwidth that is added to the signal at the receiver. Then the log-likelihood ratio y computed by an ideal receiver can easily be shown [8] to be

$$y = \frac{2}{N_0} \int_0^T [x_0(t) - x_1(t)]\,r(t)\,dt$$

where $r(t)$ is the received waveform. When $x = 0$ is the transmitted digit, then

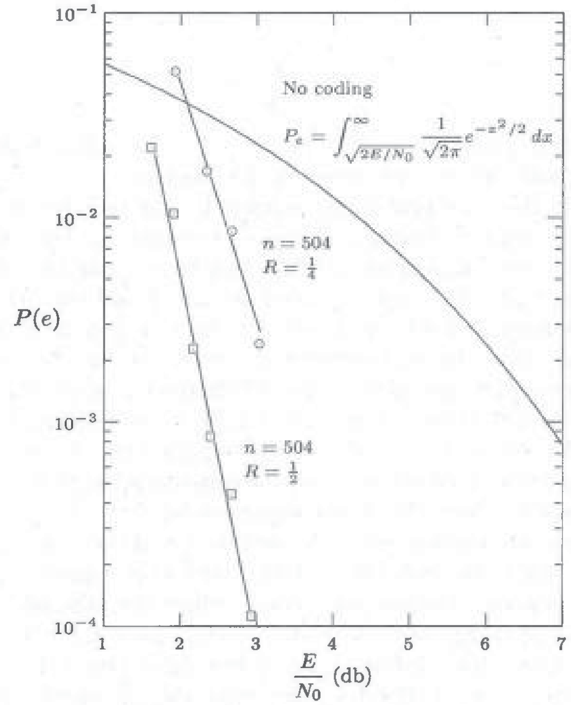


Figure 6.6: Comparison between low-density codes and no coding, white Gaussian noise.

$r(t) = x_0(t) + n(t)$, and y is easily shown to be Gaussian with probability density

$$P(y|x=0) = \frac{1}{\sqrt{2\pi}\sigma} \exp - \frac{\left(y - \frac{\sigma^2}{2}\right)^2}{2\sigma^2} \tag{6.1}$$

$$\sigma^2 = \frac{4E_c(1-\rho)}{N_0}; \qquad \rho = \frac{1}{E_c}\int_0^T x_0(t)x_1(t)\, dt \tag{6.2}$$

Likewise,

$$P(y|x=1) = \frac{1}{\sqrt{2\pi}\sigma} \exp - \frac{\left(y + \frac{\sigma^2}{2}\right)^2}{2\sigma^2}$$

Figure 3.1c contains a sketch of these probabilities.

A number of experiments were performed on the 7090 computer for codes of various block lengths and rates in which the channel outputs were chosen by a pseudorandom number generator according to the probability density in Equation (6.1), which corresponds to the all-zero code word. The simulated decoder stored these received words in the computer and then attempted to decode them by probabilistic decoding. The results of these experiments for a block length of 504 and rates of $\frac{1}{4}$ and $\frac{1}{2}$ are shown in Figure 6.6 The signal energy E appearing on the abscissa is the available energy per information digit so that

$$E = \frac{E_c}{R} \tag{6.3}$$

These data assume antipodal signals, or $\rho = -1$ in Equation (6.2). For uncorrelated signals, add 3 db to each value on the abscissa.

The fact that the error probability is lower for the rate $\frac{1}{2}$ code than for the rate $\frac{1}{4}$ code needs some explanation. Consider two systems, both with the same available signal power, noise power, block length, and number of information digits per second. If one system is coded at rate $\frac{1}{2}$ and the other at rate $\frac{1}{4}$, then the time duration of a block length for the rate $\frac{1}{2}$ code is twice that for the rate $\frac{1}{4}$ code. Thus the improvement at rate $\frac{1}{2}$ can be explained primarily by the longer constraint time of the code. While there is great theoretical merit in using the constraint time or constraint length in information bits as a basis of comparison for different rates, the cost of implementing a low-density parity check decoder is determined primarily by the constraint length in channel digits; thus we have used the latter basis of comparison here.

Consider now two systems, one coded at rate $\frac{1}{2}$ and the other uncoded, both having a final digit error probability of 10^{-3} and both transmitting the same number of information symbols per second. Since the abscissa of Figure 6.6 is given in terms of energy per information digit, Figure 6.6 indicates that the coded system requires $6.8 - 2.4$ db or 4.4 db less signal power than the uncoded system. The rate $\frac{1}{4}$ code is less favorable since the increased error-correcting power does not quite offset the loss in signal energy per channel digit. (See Figure 6.7.) Although no experimental data using likelihood receivers exist for the rate $\frac{1}{3}$ and $\frac{2}{3}$ codes, it appears unlikely from the poor performance of these codes on the BSC that they would have any advantages over the rate $\frac{1}{2}$ code.

Finally, to illustrate the advantage of likelihood receivers over decision receivers for decoding, consider Figure 6.8. This compares the experimental results for a low-density code, using a likelihood receiver and probabilistic decoding to a lower bound, to $P(e)$ for any code of the same block length and rate, using a decision receiver and maximum likelihood decoding. The abscissa, p, in Figure 6.8 is the probability of crossover that would exist if a decision were made. In other words

$$p = \int_{\sqrt{\frac{2E_c}{N_0}}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}\, dx$$

It is significant in Figure 6.8 to observe the importance of a likelihood receiver in terms of the error-correcting power of a code. It suggests that the concept of "optimum code" is not as relevant to communication as its name indicates, and that the simplicity and flexibility of coding schemes deserve much greater attention than their being "optimum."

6.4 Rayleigh Fading Channel

Assume that one of two equiprobable, equal energy, uncorrelated narrow band signals is transmitted every T seconds, and let $x_0(t)$ and $x_1(t)$ be the complex

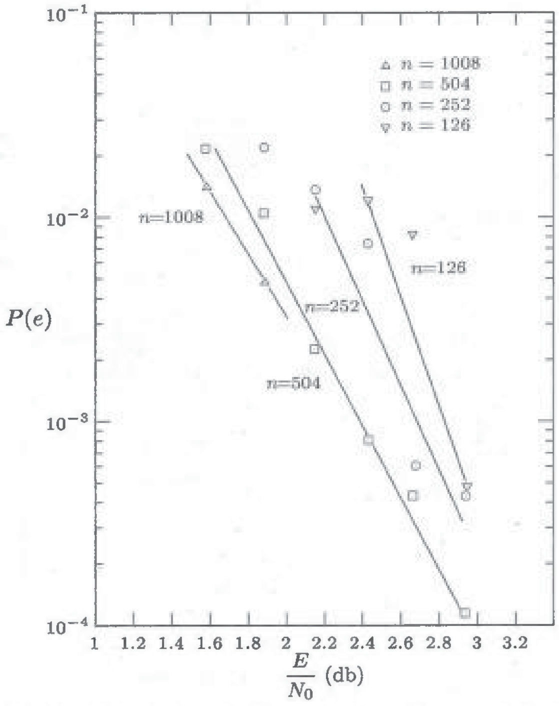


Figure 6.7: Effect of block length for rate $\frac{1}{2}$ code on white Gaussian noise channel.

positive frequency representations of these signals. Assume that the complex representation of the received signal is

$$r(t) = \alpha e^{j\beta} x_0(t) + n(t); \qquad x_0(t) \text{ transmitted}$$
$$r(t) = \alpha e^{j\beta} x_1(t) + n(t); \qquad x_1(t) \text{ transmitted}$$

where α is Rayleigh distributed and β is a random phase,

$$\Pr(\alpha) = \alpha e^{-\alpha^2/2}; \quad \alpha \geq 0$$
$$\Pr(\beta) = \frac{1}{2\pi}; \quad 0 \leq \beta \leq 2\pi$$

and $n(t)$ represents white Gaussian noise of power density N_0.

In the absence of any information about α or β before the transmission interval, it can be shown that all the information whether $x_0(t)$ or $x_1(t)$ was transmitted lies in the sampled envelopes z_0 and z_1 of the outputs of filters

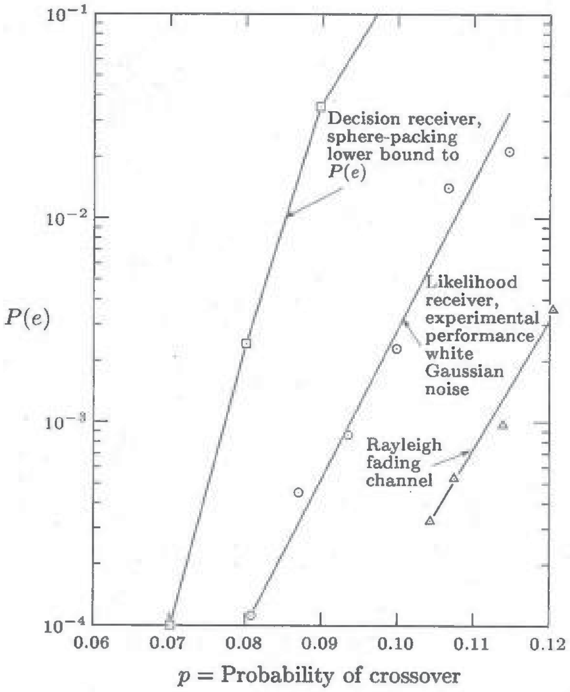


Figure 6.8: Comparison between decision receiver and likelihood receiver; $n = 504$, $R = \frac{1}{2}$.

matched to $x_0(t)$ and $x_1(t)$,

$$z_0 = \left| \int_0^T x_0^*(t) r(t)\, dt \right| E_c^{-\frac{1}{2}}$$

$$z_1 = \left| \int_0^T x_1^*(t) r(t)\, dt \right| E_c^{-\frac{1}{2}}$$

Pierce [13] shows that z_0 and z_1 are positive Rayleigh distributed random variables with variance $N_0 + E_c$ and N_0 depending on whether $x_0(t)$ or $x_1(t)$ was transmitted,

$$\Pr(z_i) = \frac{z_i}{N_0 + E_c} \exp - \frac{z_i^2}{2(N_0 + E_c)}; \quad \text{for } i = 0 \text{ of } 1, \text{ if } x_i(t) \text{ is transmitted} \tag{6.4}$$

$$\Pr(z_i) = \frac{z_i}{N_0} \exp - \frac{z_i^2}{2N_0}; \quad \text{if signal other than } x_i(t) \text{ is transmitted} \tag{6.5}$$

where

$$E_c = \int_0^T x_0(t) x_0^*(t)\, dt \tag{6.6}$$

It follows immediately from Equations (6.4) and (6.5) and the independence of z_0 and z_1 that the log-likelihood ratio for the receiver output is given by

$$y = \ln \frac{\Pr(x = 0|z_0, z_1)}{\Pr(x = 1|z_0, z_1)} = (z_0^2 - z_1^2)\left(\frac{1}{2N_0} - \frac{1}{2(N_0 + E_c)}\right) \tag{6.7}$$

Finally, it follows from Equations (6.4), (6.5) and (6.7) that

$$\Pr(y|x = 0) = \begin{cases} \dfrac{1+A}{A(2+A)} e^{-\frac{y}{A}}; & y \geq 0 \\ \dfrac{1+A}{A(2+A)} e^{\frac{y(1+A)}{A}}; & y \leq 0 \end{cases} \tag{6.8}$$

where

$$A = \frac{E_c}{N_0}$$

A Rayleigh fading channel was simulated on the computer by using a pseudo-random number generator to produce outputs y according to the probability distribution of Equation (6.8). Successive values of y were chosen independently, which appears somewhat unrealistic since we assumed the path strength was constant over the baud length, T. This should be a reasonable assumption, however, when the fading rate is comparable to the baud length, and a good assumption when scrambling is employed between the digits of successive blocks of a code.

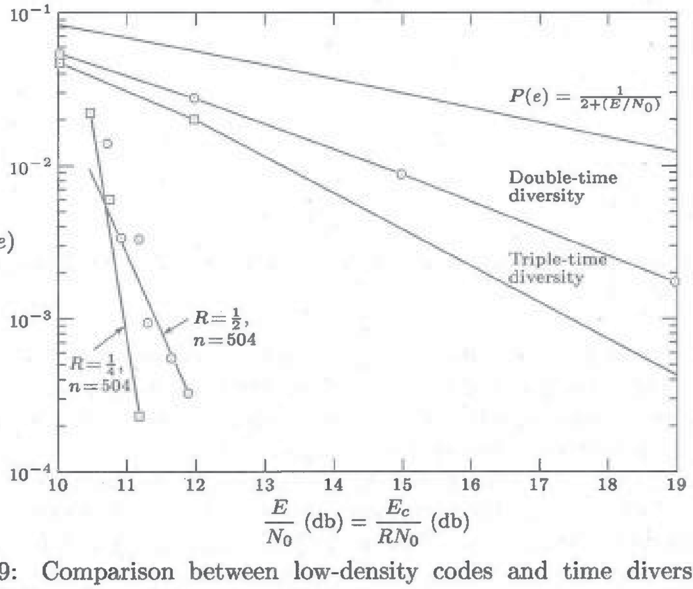


Figure 6.9: Comparison between low-density codes and time diversity for Rayleigh fading channel.

Figure 6.9 shows the results of such a simulation. Figure 6.9 shows a much more marked difference between coding and no coding than Figure 6.6, and this is, of course, due to the slow decrease of bit error rates with signal power on Rayleigh fading channels. Figure 6.9 also indicates that the rate $\frac{1}{4}$ code is somewhat better than the rate $\frac{1}{2}$ code, but there are not enough data here to be convincing. Also, the rate $\frac{1}{2}$ code contains twice as many information digits per block as the rate $\frac{1}{4}$ code, so that a block lasts twice as long for the same information rate in bits per second. This is advantageous when the fades are longer than a baud length.

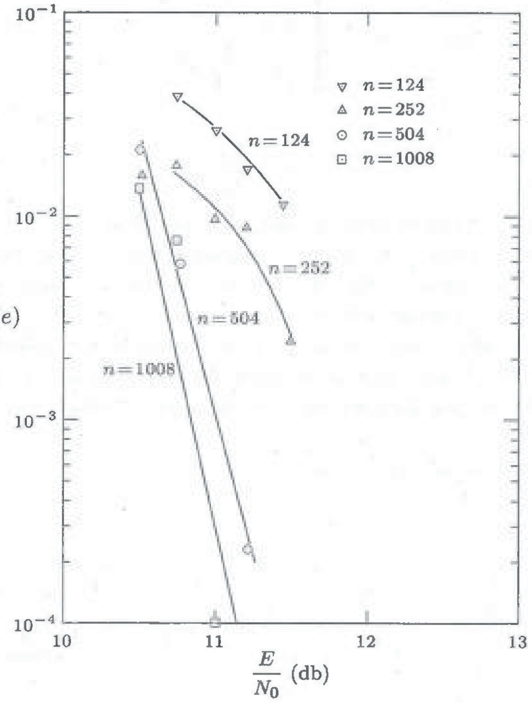


Figure 6.10: Effect of block length on error probability for Rayleigh fading channel; $R = \frac{1}{4}$.

Figure 6.10 shows the effect of block length on error probability for the rate $\frac{1}{4}$ code. The error probabilities for the smaller block length codes appear to decrease much more slowly with increasing signal power than the long block length codes, but more data would be helpful here.

Finally, Figure 6.8 again shows the advantage of a likelihood receiver over a decision receiver for the Rayleigh fading channel. The Rayleigh fading channel and Gaussian channel are so different in their characteristics that it is conjectured that this type of gain holds for most symmetric binary-input channels (with the obvious exception of the BSC).

A Properties of the Function $B(\lambda)$

In Chapter 2, the following bound was derived for the minimum-distance distribution function of an (n, j, k) ensemble of codes:

$$\begin{aligned} \Pr(D \le n\delta) &\le \sum_{\ell=2}^{n\delta} C(\lambda, n) \exp -nB(\lambda) \\ \Pr(D \le n\delta) &\le 1 \end{aligned} \tag{A.1}$$

where

$$\lambda = \frac{\ell}{n}$$

$$B(\lambda) = (j-1)H(\lambda) - \frac{j}{k}\left[\mu(s) + (k-1)\ln 2\right] + js\lambda \tag{A.2}$$

$$C(\lambda, n) = \left[2\pi n\lambda(1-\lambda)\right]^{\frac{j-1}{2}} \exp \frac{j-1}{12n\lambda(1-\lambda)} \tag{A.3}$$

$$\mu(s) + (k-1)\ln 2 = \ln \tfrac{1}{2}\left[(1+e^s)^k + (1-e^s)^k\right] \tag{A.4}$$

$$\frac{\mu'(s)}{k} = \lambda \quad \text{for optimum bound} \tag{A.5}$$

In this appendix three theorems will be proved concerning Equation (A.1). The first theorem will analyze the behavior of $B(\lambda)$, the second will bound the summation in Equation (A.1) in terms of the first and last terms, the third will show that as j and k increase, Equation (A.1) approaches the minimum-distance distribution function derived for the equiprobable ensemble of codes in Equation (2.5).

Theorem A.1. *Assume $k > j \ge 3$, and let $B(\lambda)$ be defined in Equations* (A.2), (A.4) *and* (A.5). *Then*

1. $\lim_{\lambda \to 0} B(\lambda) = 0$,

2. $\lim_{\lambda \to 0} \frac{dB}{d\lambda} = \infty$,

3. $B(\lambda)$ has only one zero in the range $0 < \lambda < \frac{1}{2}$,

4. $B(\lambda)$ has no local minimum within the range where $B(\lambda) > 0$.

Proof. 1. We show that $\lim_{\lambda \to 0} B(\lambda) = 0$ by showing that each of the three terms on the right of Equation (A.2) approaches 0. The term $H(\lambda)$ is given by $-\lambda \ln \lambda - (1-\lambda)\ln(1-\lambda)$ and clearly approaches 0. Differentiating Equation (A.4), we get

$$\lambda = \frac{\mu'(s)}{k} = \frac{e^s\left[(1+e^s)^{k-1} - (1-e^s)^{k-1}\right]}{(1+e^s)^k + (1-e^s)^k} \tag{A.6}$$

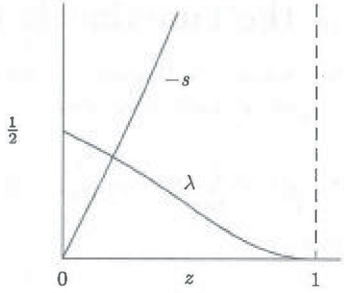


Figure A.1: Sketch of s and λ as functions of z.

and from this, $s \to -\infty$ as $\lambda \to 0$. But from Equation (A.4), $\lim_{s\to-\infty} \mu(s) + (k-1)\ln 2 = 0$. Finally,

$$js\lambda = \frac{jse^s\left[(1+e^s)^{k-1} - (1-e^s)^{k-1}\right]}{(1+e^s)^k + (1-e^s)^k}$$

which also approaches 0 as $s \to -\infty$.

2. From Equation (A.2),

$$\frac{dB}{d\lambda} = \frac{\partial B(\lambda)}{\partial \lambda} + \frac{\partial B(\lambda)}{\partial s}\left(\frac{\partial \lambda}{\partial s}\right)^{-1} = (j-1)\ln\frac{1-\lambda}{\lambda} + js$$

Making the substitution

$$\begin{aligned} z &= \frac{1-e^s}{1+e^s} \\ s &= \ln\frac{1-z}{1+z} \end{aligned} \tag{A.7}$$

and performing some manipulation on Equation (A.6), we get

$$\lambda = \frac{1-z}{2}\frac{1-z^{k-1}}{1+z^k} \tag{A.8}$$

In Figure A.1, s and λ are sketched as functions of z.

$$\begin{aligned} \lim_{\lambda\to 0}\frac{dB}{d\lambda} &= \lim_{z\to 1}(j-1)\ln\left(\frac{1+z}{1-z}\right)\left(\frac{1+z^{k-1}}{1-z^{k-1}}\right) + j\ln\frac{1+z}{1-z} \\ \lim_{\lambda\to 0}\frac{dB}{d\lambda} &= \lim_{z\to 1}\ln\left(\frac{1+z}{1-z}\right)\left(\frac{1+z^{k-1}}{1-z^{k-1}}\right)^{j-1} \\ \lim_{\lambda\to 0}\frac{dB}{d\lambda} &= \lim_{z\to 1}\ln\frac{(1+z^{k-1})^{j-1}}{(1-z^{k-1})^{j-2}(1+z)(1+z+\cdots+z^{k-2})} \end{aligned} \tag{A.9}$$

$\lim_{\lambda\to 0}\frac{dB}{d\lambda} = \infty$ for $j-2>0$, or in other words, for $j \geq 3$.

3. Before proving parts 3 and 4 of the theorem, we must show that $dB/d\lambda$ has only one extremum. Using Equation (A.9), we obtain the derivative of $dB/d\lambda$ with respect to z.

$$\frac{d}{dz}\Big(\frac{dB}{d\lambda}\Big) = \frac{2}{1-z^2} + \frac{2(j-1)(k-1)z^{k-2}}{1-z^{2(k-1)}}$$

Setting this equal to 0, we have

$$(j-1)(k-1) = \frac{1-z^{2k-2}}{(1-z^2)z^{k-2}} = \frac{1+z^2+z^4+\cdots+z^{2k-4}}{z^{k-2}}$$

$$(j-1)(k-1) = 1 + \sum_{i=1}^{\frac{k-2}{2}} \Big(z^{2i} + \frac{1}{z^{2i}}\Big); \quad \text{for } k \text{ even} \tag{A.10}$$

$$(j-1)(k-1) = \sum_{i=1}^{\frac{k-1}{2}} \Big(z^{2i-1} + \frac{1}{z^{2i-1}}\Big); \quad \text{for } k \text{ odd} \tag{A.11}$$

The functions on the right in Equations (A.10) and (A.11) are decreasing in z for $0 < z < 1$. Hence each equation can have at most one solution in this range. Thus, $dB/d\lambda$ has at most one extremum and at most two zeros for $0 < \lambda < \frac{1}{2}$. Then B has at most two zeros besides $B(0) = 0$. But since B goes positive as λ increases from 0, two zero crossings for $0 < \lambda < \frac{1}{2}$ would imply $B(\frac{1}{2}) > 0$. However, from Equation (A.4), using $s = 0$ at $\lambda = \frac{1}{2}$,

$$B(\tfrac{1}{2}) = \Big[(j-1)\ln 2 - \frac{j}{k}(k-1)\ln 2 - \Big(1 - \frac{j}{k}\Big)\ln 2\Big] < 0$$

Therefore, $B(\lambda)$ has exactly one zero for $0 < \lambda < \frac{1}{2}$.

4. If $B(\lambda)$ has a minimum within the range for which $B(\lambda) > 0$, then it would require a maximum on the either side of the minimum to satisfy $B(0) = 0$ and $B(\frac{1}{2}) < 0$. But $B(\lambda)$ has at most two extrema, so this is impossible.

□

Theorem A.2. *For an (n, j, k) ensemble of codes, the minimum-distance distribution function may be bounded by*[5]

$$\Pr(D \leq n\delta) \leq \frac{k-1}{2n^{j-2}} + 0(n^{-j+2}) + nC(\delta, n)\exp -nB(\delta) \tag{A.12}$$

Proof. From Equation (2.18), we have

$$\Pr(D \leq n\delta) \leq \sum_{\ell=2}^{n\delta} \binom{n}{\ell}^{-j+1} [N_1(\ell)]^j$$

[5] By $0(n^{-j+2})$ we mean a function that goes to zero with increasing n faster than n^{-j+2}; that is, a function $f(n)$ such that $\lim_{n\to\infty} n^{j-2}f(n) = 0$.

We can evaluate the term for $\ell = 2$ directly. Recall that $N_1(2)$ is the number of sequences of weight 2 which satisfy the first n/k parity-checks of any particular code. There are $\binom{k}{2}$ ways of arranging 2 ones in a single parity check set; multiplying by the n/k parity-check sets, we have

$$\begin{aligned} N_1(2) &= \frac{n}{k}\binom{k}{2} \\ \binom{n}{2}^{-j+1} N_1(2)^j &= \frac{n(k-1)^j}{2(n-1)^{j-1}} = \frac{k-1}{2n^{j-2}} + 0(n^{-j+2}) \\ \Pr(D \le n\delta) &\le \frac{k-1}{2n^{j-2}} + 0(n^{-j+2}) + \sum_{\ell=4}^{n\delta} C(\lambda, n) \exp -nB(\lambda) \qquad \text{(A.13)} \end{aligned}$$

where $C(\lambda, n)$ and $B(\lambda)$ are given in Equations (A.2) and (A.3). In order to bound the terms for which ℓ is small in Equation (A.13), we note from Equation (A.6) that as $\lambda \to 0$, $s \to \frac{1}{2}\ln[\lambda/(k-1)]$. Using this value of s instead of $\mu'(s)/k = \lambda$ in Equation (A.2), $B(\lambda)$ must be underbounded.

$$\begin{aligned} B(\lambda) &\ge (j-1)\Big[\lambda \ln \frac{1}{\lambda} + (1-\lambda)\ln\frac{1}{1-\lambda}\Big] - \frac{j}{k}\ln \sum_{i \text{ even}} \binom{k}{i} e^{si} + \frac{j}{2}\lambda \ln \frac{\lambda}{k-1} \\ B(\lambda) &\ge \Big(\frac{j}{2} - 1\Big)\lambda \ln \frac{1}{\lambda} - \frac{j}{k}\ln\frac{1}{1 - \binom{k}{2}e^{2s}} - \frac{j}{2}\lambda \ln(k-1) \qquad \text{(A.14)} \end{aligned}$$

Substituting ℓ/n for λ and using some inequalities, we have

$$\exp -nB(\lambda) \le n^{-\ell\left(\frac{j}{2}-1\right)} \ell^{\ell\left(\frac{j}{2}-1\right)} (k-1)^{\frac{j\ell}{2}} \exp\Big(\frac{\ell j}{2}\Big)\Big(\frac{1}{1 - \frac{k\ell}{2n}}\Big) \qquad \text{(A.15)}$$

From Equation (A.3) we get

$$C(\lambda, n) \le (2\pi\ell)^{\frac{j-1}{2}} \exp \frac{j-1}{6\ell} \qquad \text{(A.16)}$$

From Equations (A.15) and (A.16), we see that the terms for $\ell = 4$ and $\ell = 6$ in Equation (A.13) approach zero faster than n^{-j+2}. From Theorem A.1, if $B(\delta) > 0$, then for every term between $\ell = 8$ and $\ell = n\delta$, $B(\lambda)$ is lower bounded by either $B(8/n)$ or $B(\delta)$. (If $B(\delta) < 0$, the right side of Equation (A.12) is larger than 1 and the trivial bound of 1 applies.) Thus, the summation between $\ell = 8$ and δn is bounded by

$$nC_{\max}\Big[\exp -nB\Big(\frac{8}{n}\Big) + \exp -nB(\delta)\Big] \qquad \text{(A.17)}$$

The first term of Equation (A.17) has an n dependence given by

$$n^{\left[1 + \frac{j-1}{2} + 8(-\frac{j}{2}+1)\right]} = 0(n^{-j+2}); \quad \text{for } j \ge 3$$

The second term of Equation (A.17) is the last expression appearing in the statement of the theorem, Equation (A.12), proving the theorem. □

Theorem A.3. *Let δ_{jk} be the nonzero solution of $B(\lambda) = 0$ for an (n, j, k) ensemble, and let $R = 1 - j/k$ be fixed. Let $\delta_0 < \frac{1}{2}$ be the solution of $H(\delta_0) = (1 - R)\ln 2$. Then $\lim_{k\to\infty} \delta_{jk} = \delta_0$.*

From Theorem 2.2, δ_0 is the typical minimum-distance for the equiprobable ensemble of parity-check codes, so the theorem asserts that the typical minimum distance of (n, j, k) codes approach that of the equiprobable ensemble as k increases.

Proof. Using Equation (A.2), $B(\lambda)$ can be rewritten in the form

$$B(\lambda) = \Big\{-H(\lambda) + \frac{j}{k}\ln 2\Big\} + \Big\{j[H(\lambda) + s\lambda] - \frac{j}{k}\ln\big[(1 - e^s)^k + (1 + e^s)^k\big]\Big\} \tag{A.18}$$

We shall show that for $\lambda \neq 0$, the last brace in Equation (A.18) approaches 0 with increasing k. This is sufficient to prove the theorem, since $j/k = 1 - R$ and thus the first brace is zero only for $\lambda = \delta_0$.

$$H(\lambda) + s\lambda = \lambda\Big[\ln\Big(\frac{1-\lambda}{\lambda}\Big) + s\Big] - \ln(1 - \lambda)$$

Making the substitution $z = (1 - e^s)/(1 + e^s)$ of Equations (A.7) and (A.8),

$$H(\lambda) + s\lambda = \frac{1-z}{2}\frac{1-z^{k-1}}{1+z^k}\ln\frac{1+z^{k-1}}{1-z^{k-1}} - \ln\Big(\frac{1+z}{2}\Big)\Big(\frac{1+z^{k-1}}{1+z^k}\Big) \tag{A.19}$$

Also

$$\begin{aligned}\frac{1}{k}\ln\big[(1 + e^s)^k + (1 - e^s)^k\big] &= \ln(1 + e^s) + \frac{1}{k}\ln(1 + z^k)\\ &= \ln\frac{2}{1+z} + \frac{1}{k}\ln(1 + z^k)\end{aligned} \tag{A.20}$$

Combining Equations (A.19) and (A.20), the second brace in Equation (A.18) becomes

$$-j\Big(\frac{1-z}{2}\Big)\Big(\frac{1-z^{k-1}}{1+z^k}\Big)\ln\frac{1+z^{k-1}}{1-z^{k-1}} + j\ln\frac{1+z^{k-1}}{1+z^k} + \frac{j}{k}\ln(1 + z^k)$$

As k increases, for any $z < 1$, (where $\lambda > 0$), z^k and z^{k-1} approach 0. Expanding the logarithms we have

$$-j\Big(\frac{1-z}{2}\Big)\Big(\frac{1-z^{k-1}}{1+z^k}\Big)2z^{k-1} + jz^{k-1}(1 - z) + \frac{jz^k}{k}$$

In this expression, $j \to \infty$ linearly with k, but $z^{k-1} \to 0$ exponentially. Thus, the second brace in Equation (A.18) approaches 0. □

B Miscellaneous Mathematical Derivations for Chapter 3

B.1 Chernov Bounds

Theorem 3.1. *Let $Z = \sum_{i=1}^{n} z_i$ be the sum of n independent random variables, let $P_i(z_i)$ be the probability density of the i^{th} variable, and let $g_i(s) = \int_{-\infty}^{\infty} \exp(sz_i)P_i(z_i)\,dz_i$ be the moment generating function for the i^{th} variable. Then*

$$Pr(Z \geq nz_0) \leq \exp(-nsz_0) \prod_{i=1}^{n} g_i(s) \tag{B.1}$$

for all $s \geq 0$ such that the $g_i(s)$ exist. If the z_i are discrete, then the same statement holds except that the $P_i(z_i)$ are probabilities and the integral defining $g_i(s)$ is replaced by a sum.

Proof. The sum Z is itself a random variable, and has a probability distribution function $F(Z)$ and a moment-generating function,

$$G(s) = \int_{-\infty}^{\infty} \exp(sZ)\,dF(Z) = \overline{\exp(sZ)} \tag{B.2}$$

From the definition of Z, we get

$$G(s)\overline{\exp\Big(\sum_{i=1}^{n} z_i\Big)} = \overline{\prod_{i=1}^{n} \exp sz_i}$$

Since the variables are independent,

$$G(s) = \prod_{i=1}^{n} \overline{\exp sz_i} = \prod_{i=1}^{n} g_i(s) \tag{B.3}$$

Now from Equations (B.2) and (B.3), we get

$$\prod_{i=1}^{n} g_i(s) = \int_{-\infty}^{\infty} \exp(sZ)\,dF(Z) \geq \int_{nz_0}^{\infty} \exp(sZ)\,dF(Z) \tag{B.4}$$

For $s \geq 0$ and $Z \geq nz_0$, $sZ \geq sz_0$. Thus,

$$\prod_{i=1}^{n} g_i(s) \geq \exp(snz_0) \int_{nz_0}^{\infty} dF(Z) = \exp(snz_0)\,\mathrm{Pr}(Z \geq nz_0) \tag{B.5}$$

Rearranging terms, we get the statement of the theorem, Equation (B.1). The theorem is proven in exactly the same way if the z_i are discrete. □

It would appear from the rather gross inequalities in Equations (B.4) and (B.5) that the bound in Equation (B.1) is rather poor. However, this is not so if the parameter s is correctly chosen and if nz_0 is greater than the mean value of Z. To see this, consider the product $F(Z)e^{sZ}$. For larger n, $F(Z)$ increases sharply around $\bar{Z}$. However, e^{sZ} can be considered as a weighting factor that weights large Z very heavily. Thus, the product $F(Z)e^{sZ}$ will have a sharp rise for some Z larger than $\bar{Z}$. The trick is to pick s to that this rise occurs at $Z = nz_0$. Analytically, this can be done by taking the partial derivatives with respect to s of the right side of Equation (B.1) and setting it equal to 0, giving

$$nz_0 = \sum_{i=1}^{n} \frac{1}{g_i(s)} \frac{\partial g_i(s)}{\partial s} \tag{B.6}$$

With the choice of s satisfying Equation (B.6), it can be shown that the bound in Equation (B.1), known as the Chernov bound, at least has the correct exponential dependence on n.

Theorem 3.2. *Let z_i and w_i, $1 \leq i \leq n$, be n pairs of random variables with probability density functions $P_i(z_i, w_i)$. Let the joint moment generating function of z_i, w_i, be*

$$h_i(r,t) = \iint \exp(rz_i + tw_i) P_i(z_i, w_i)\, dz_i\, dw_i \tag{B.7}$$

Let each pair of random variables be statistically independent of each other pair and define Z and W by

$$\begin{aligned} Z &= \sum_{i=1}^{n} z_i \\ W &= \sum_{i=1}^{\ell} w_i \\ \ell &\leq n \end{aligned} \tag{B.8}$$

Then, for any arbitrary numbers z_0 and w_0,

$$P(Z \leq nz_0;\, W \leq nw_0) \leq \prod_{i=1}^{\ell} [h_i(r,t)] \prod_{i=\ell+1}^{n} [h_i(r,0)] \exp -n(rz_0 + tw_0) \tag{B.9}$$

for any $r \leq 0$, $t \leq 0$ for which $h_i(r,t)$ exists. If z and w are discrete, Equation (B.8) *still holds with integrals in Equation* (B.7) *replaced by sums, and the probability density replaced by a probability.*

Proof. Let $F(Z, W)$ be the distribution function of Z, W and let the moment-generating function of Z, W be

$$\begin{aligned} H(r,t) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp(rZ + tW)\, dF(Z,W) \\ &= \overline{\exp(rZ + tW)} \end{aligned} \tag{B.10}$$

Using Equation (B.8) and the independence of samples, we get

$$\begin{aligned}
H(r,t) &= \overline{\exp\Big[\sum_{i=1}^{\ell}(rz_i + tw_i) + \sum_{i=\ell+1}^{n} rz_i\Big]} \\
&= \Big[\prod_{i=1}^{\ell} \overline{\exp(rz_i + tw_i)}\Big]\Big[\prod_{i=\ell+1}^{n} \overline{\exp rz_i}\Big] \\
&= \prod_{i=1}^{\ell} h_i(r,t) \prod_{i=\ell+1}^{n} h_i(r,0) \qquad \text{(B.11)}
\end{aligned}$$

Combining Equations (B.10) and (B.11), we get

$$\begin{aligned}
\prod_{i=1}^{\ell} h_i(r,t) \prod_{i=\ell+1}^{n} h_i(r,0) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(rZ + tW)\, dF(Z,W) \\
&\geq \int_{Z=-\infty}^{nz_0}\int_{W=-\infty}^{nw_0} \exp(rZ + tW)\, dF(Z,W)
\end{aligned}$$

For $r \leq 0$, $t \leq 0$, $Z \leq nz_0$, and $W \leq nw_0$ we have

$$\exp(rZ + tW) \geq \exp(rnz_0 + tnw_0)$$

$$\prod_{i=1}^{\ell} h_i(r,t) \prod_{i=\ell+1}^{n} h_i(r,0) \geq \exp(rnz_0 + tnw_0)\Pr(Z \leq nz_0;\, W \leq nw_0)$$

Rearranging terms, we get the statement of the theorem, Equation (B.9). □

B.2 Optimum Value of $f(y)$

We wish to find an expression for $f(y) = f(-y)$ to maximize the expression

$$E(s,r,\lambda) = \frac{r}{s-r}\ln g(s) - \frac{s}{s-r}\big[B(\lambda) + \lambda \ln h(r) + (1-\lambda)\ln g(r)\big] \qquad \text{(B.12)}$$

where

$$g(s) = \int_{-\infty}^{\infty} P_0(y)^{1-s} f(y)^s\, dy \qquad \text{(B.13)}$$

$$h(r) = \int_{-\infty}^{\infty} P_0(y)^{\frac{1}{2}(1-r)} P_1(y)^{\frac{1}{2}(1-r)} f(y)^r\, dy \qquad \text{(B.14)}$$

If we write $f(y)$ in the form $f(y) = f_0(y) + \epsilon f_\epsilon(y)$, $f_0(y)$ will maximize $E(s,r,\lambda)$ if $E(s,r,\lambda)$ is maximized with respect to ϵ at $\epsilon = 0$ independent of $f_\epsilon(y)$. We can automatically satisfy the constraint $f(y) = f(-y)$ is we rewrite the integrals in Equations (B.13) and (B.14) as integrals from 0 to ∞. Thus,

$$g(s) = \int_0^\infty \Big[P_0(y)^{1-s} + P_1(y)^{1-s}\Big][f_0(y) + \epsilon f_\epsilon(y)]^s \, dy \tag{B.15}$$

$$h(r) = \int_0^\infty 2P_0(y)^{\frac{1}{2}(1-r)} P_1(y)^{\frac{1}{2}(1-r)} [f_0(y) + \epsilon f_\epsilon(y)]^r \, dy \tag{B.16}$$

Using Equations (B.15) and (B.16) we get

$$\begin{aligned}\frac{\partial E(s,r,\lambda)}{\partial \epsilon} &= \frac{rs}{(s-r)g(s)} \int_0^\infty \Big(P_0^{1-s} + P_1^{1-s}\Big)(f_0 + \epsilon f_\epsilon)^{s-1} f_\epsilon \, dy \\ &\quad - \frac{s\lambda r}{(s-r)h(r)} \int_0^\infty 2P_0^{\frac{1}{2}(1-r)} P_1^{\frac{1}{2}(1-r)} (f_0 + \epsilon f_\epsilon)^{r-1} f_\epsilon \, dy \\ &\quad - \frac{s(1-\lambda)r}{(s-r)g(r)} \int_0^\infty \Big(P_0^{1-r} + P_1^{1-r}\Big)(f_0 + \epsilon f_\epsilon)^{r-1} f_\epsilon \, dy \end{aligned} \tag{B.17}$$

If Equation B.17 is written out as one integral, it is clear that it will be 0 at $\epsilon = 0$ independent of $f_\epsilon(y)$ only if the integrand is identically 0. Thus,

$$\begin{aligned}\frac{1}{g(s)}\Big(P_0^{1-s} + P_1^{1-s}\Big) f_0^{s-1} - \frac{\lambda}{h(r)} 2P_0^{\frac{1}{2}(1-r)} P_1^{\frac{1}{2}(1-r)} f_0^{r-1} & \\ - \frac{1-\lambda}{g(r)}\Big(P_0^{1-r} + P_1^{1-r}\Big) f_0^{r-1} &= 0 \end{aligned} \tag{B.18}$$

$$f_0(y)^{s-r} = \frac{\frac{\lambda}{h(r)}\Big[2P_0(y)^{\frac{1}{2}(1-r)} P_1(y)^{\frac{1}{2}(1-r)}\Big] + \frac{1-\lambda}{g(r)}\Big[P_0(y)^{1-r} + P_1(y)^{1-r}\Big]}{\frac{1}{g(s)}\Big[P_0(y)^{1-s} + P_1(y)^{1-s}\Big]} \tag{B.19}$$

Finally, we observe that if an $f_0(y)$ satisfying Equation (B.19) is multiplied by an arbitrary constant, it will still satisfy Equation (B.19) due to the compensating changes in $g(s)$, $h(r)$ and $g(r)$. Thus, with a little manipulation we get Equation (3.40).

Next we show that this value of $f_0(y)$ yields a local maximum of $E(s,r,\lambda)$ with respect to ϵ.

$$\begin{aligned}\frac{\partial^2 E(s,r,\lambda)}{\partial \epsilon^2} &= \frac{r}{(s-r)g(s)^2}\left\{g(s)\frac{\partial^2 g(s)}{\partial \epsilon^2} - \left[\frac{\partial g(s)}{\partial \epsilon}\right]^2\right\} \\ &\quad - \frac{s\lambda}{(s-r)h(r)^2}\left\{h(r)\frac{\partial^2 h(r)}{\partial \epsilon^2} - \left[\frac{\partial h(r)}{\partial \epsilon}\right]^2\right\} \\ &\quad - \frac{s(1-\lambda)}{(s-r)g(r)^2}\left\{g(r)\frac{\partial^2 g(r)}{\partial \epsilon^2} - \left[\frac{\partial g(r)}{\partial \epsilon}\right]^2\right\} \end{aligned} \tag{B.20}$$

Consider the first brace in Equation (B.20):

$$\begin{aligned} g(s)\frac{\partial^2 g(s)}{\partial \epsilon^2} - \left[\frac{\partial g(s)}{\partial \epsilon}\right]^2\Bigg|_{\epsilon=0} &= s(s-1)\left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^s \, dy\right] \\ &\quad \times \left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^{s-2} f_\epsilon^2 \, dy\right] \\ &\quad - s^2 \left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^{s-1} f_\epsilon \, dy\right]^2 \end{aligned}$$

From the Schwartz inequality,

$$\begin{aligned} \left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^{s-1} f_\epsilon \, dy\right]^2 &\leq \left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^s \, dy\right] \\ &\quad \times \left[\int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^{s-2} f_\epsilon^2 \, dy\right] \end{aligned}$$

$$g(s)\frac{\partial^2 g(s)}{\partial \epsilon^2} - \left[\frac{\partial g(s)}{\partial \epsilon}\right]^2\Bigg|_{\epsilon=0} \geq -s g(s) \int_0^\infty (P_0^{1-s} + P_1^{1-s}) f_0^{s-2} f_\epsilon^2 \, dy$$

In the same way, it can be shown that at $\epsilon = 0$,

$$h(r)\frac{\partial^2 h(r)}{\partial \epsilon^2} - \left[\frac{\partial h(r)}{\partial \epsilon}\right]^2 \geq -r h(r)\, 2 \int_0^\infty P_0^{\frac{1}{2}(1-r)} P_1^{\frac{1}{2}(1-r)} f_0^{r-2} f_\epsilon^2 \, dy$$

$$g(r)\frac{\partial^2 g(r)}{\partial \epsilon^2} - \left[\frac{\partial g(s)}{\partial \epsilon}\right]^2 \geq -r g(r) \int_0^\infty (P_0^{1-r} + P_1^{1-r}) f_0^{r-2} f_\epsilon^2 \, dy$$

Combining these results and using the fact that $s \geq 0$, $r \leq 0$, we find

$$\begin{aligned} \frac{\partial^2 E(s,r,\lambda)}{\partial \epsilon^2}\Bigg|_{\epsilon=0} \leq \frac{rs}{s-r} \int_0^\infty \Bigg[&\frac{1}{g(s)} (P_0^{1-s} + P_1^{1-s}) f_0^{s-2} f_\epsilon \\ &- \frac{\lambda}{h(r)} 2 P_0^{\frac{1}{2}(1-r)} P_1^{\frac{1}{2}(1-r)} f_0^{r-2} f_\epsilon^2 \\ &- \frac{1-\lambda}{g(r)} (P_0^{1-r} + P_1^{1-r}) f_0^{r-2} f_\epsilon^2 \Bigg] dy \end{aligned} \qquad \text{(B.21)}$$

Finally, comparing the integrand of Equation (B.21) with Equation (B.18), we see that the integrand is identically 0. Thus,

$$\frac{\partial^2 E(s,r,\lambda)}{\partial \epsilon^2}\Bigg|_{\epsilon=0} \leq 0$$

and we have established that Equation (3.40) yields a local maximum of $E(s,r,\lambda)$ with respect to $f(y)$.

B.3 Elimination of $f(y)$ from Exponent

In this section, we simplify the expression for $E(s, r, \lambda)$ given in Equations (B.12), (B.13) and (B.14) by eliminating $f(y)$ from Equations (B.13) and (B.14) by using Equation (3.41) (repeated here as Equation (B.22) for convenience).

$$f(y) = \left[P_0(y)^{\frac{1}{2}(1-r)} + P_1(y)^{\frac{1}{2}(1-r)}\right]^{\frac{2}{s-r}} \left[P_0(y)^{1-s} + P_1(y)^{1-s}\right]^{-\frac{1}{s-r}} \tag{B.22}$$

First we add and subtract $s/(s-r)\ln[g(r)+h(r)]$ from Equation (B.12). This gives us

$$\begin{aligned} E(s,r,\lambda) &= \frac{r}{s-r}\ln g(s) \\ &\quad - \frac{s}{s-r}\{B(\lambda) + \lambda \ln \alpha + (1-\lambda)\ln(1-\alpha) + \ln[g(r)+h(r)]\} \end{aligned} \tag{B.23}$$

$$\alpha = \ln \frac{h(r)}{g(r)+h(r)} \tag{B.24}$$

Writing out $g(s)$ and $g(r)+h(r)$ by using Equation (B.22), we get

$$\begin{aligned} g(s) &= g(r) + h(r) \\ &= \int_0^\infty \left[P_0(y)^{1-s} + P_1(y)^{1-s}\right]^{-\frac{r}{s-r}} \left[P_0(y)^{\frac{1}{2}(1-r)} + P_1(y)^{\frac{1}{2}(1-r)}\right]^{\frac{2s}{s-r}} \end{aligned} \tag{B.25}$$

Substituting Equation (B.25) into Equation (B.23), and writing out the expression for α, we get Equations (3.43), (3.44), and (3.45).

B.4 Simplification of Exponent for Random Ensemble of Parity-Check Codes

Equation (3.47) shows that $\beta(\alpha) = (1-R)\ln 2$, which is independent of α in this case. Thus the expression for $E(s,r)$, Equation (3.43), is independent of α and can be written

$$\begin{aligned} E(s,r) &= \frac{s}{s-r}(1-R)\ln 2 \\ &\quad - \ln \int \left(P_0^{1-s} + P_1^{1-s}\right)^{-\frac{r}{s-r}} \left(P_0^{\frac{1}{2}(1-r)} + P_1^{\frac{1}{2}(1-r)}\right)^{\frac{2s}{s-r}} dy \end{aligned} \tag{B.26}$$

If we now make the substitutions $\rho = s/(s-r)$, $\sigma_1 = 1-s$, $\sigma_2 = \frac{1}{2}(1-r)$, we get

$$E_1(\sigma_1,\rho) = \rho(1-R)\ln 2 - \ln \int \left(P_0^{\sigma_1} + P_1^{\sigma_1}\right)^{1-\rho} \left(P_0^{\sigma_2} + P_1^{\sigma_2}\right)^{2\rho} dy \tag{B.27}$$

where

$$\sigma_2 = \frac{1 - \sigma_1(1-\rho)}{2\rho}$$

Next we find the maximum of $E_1(\sigma_1, \rho)$ over σ_1. Define

$$z(\sigma_1, y) = [P_0(y)]^{\sigma_1} + [P_1(y)]^{\sigma_1}$$

$$E_1(\sigma_1, \rho) = \rho(1-R)\ln 2 - \ln \int z(\sigma_1, y)^{1-\rho} z(\sigma_2, y)^{2\rho}\, dy$$

$$\frac{\partial E_1(\sigma_1, \rho)}{\partial \sigma_1} = \frac{-\int_0^\infty z(\sigma_1, y)^{1-\rho} z(\sigma_2, y)^{2\rho} \left[\frac{(1-\rho) z_1'(\sigma_1, y)}{z(\sigma_1, y)} - \frac{2\rho z_1'(\sigma_2, y)(1-\rho)}{z(\sigma_2, y)^{2\rho}}\right] dy}{\int_0^\infty z(\sigma_1, y)^{1-\rho} z(\sigma_2, y)^{2\rho}\, dy} \tag{B.28}$$

The partial derivate in Equation (B.28) was taken with ρ constant but σ_2 varying with σ_1 according to Equation (B.27). It is clear from the bracketed term in Equation (B.28) that $E_1(\sigma_1, \rho)$ has a stationary point at $\sigma_1 = \sigma_2$, or $1 - s = \frac{1}{2}(1-r)$.

In order to show that $\sigma_1 = \sigma_2$ actually maximizes $E_1(\sigma_1, \rho)$, it is sufficient to show that Equation (B.28) is nonnegative for $\sigma_1 < \sigma_2$ and nonpositive for $\sigma_1 > \sigma_2$. Since the sign of Equation (B.28) is determined only by the bracketed term, however, is it sufficient to show that

$$\frac{\partial}{\partial \sigma_1}\left[\frac{-z_1'(\sigma_1, y)}{z(\sigma_1, y)} + \frac{z_1'(\sigma_2, y)}{z(\sigma_2, y)}\right] \leq 0 \tag{B.29}$$

Or,

$$-\frac{z_1''(\sigma_1, y) z(\sigma_1, y) - [z_1'(\sigma_1, y)]^2}{[z(\sigma_1, y)]^2} + \frac{z_1''(\sigma_2, y) z(\sigma_2, y) - [z_1'(\sigma_2, y)]^2}{[z(\sigma_2, y)]^2}\left[-\frac{(1-\rho)}{2\rho}\right] \leq 0$$

Writing out the first term, we get

$$-\frac{\left[P_0^{\sigma_1}(\ln P_0)^2 + P_1^{\sigma_1}(\ln P_1)^2\right]\left[P_0^{\sigma_1} + P_1^{\sigma_1}\right] - \left[P_0^{\sigma_1}\ln P_0 + P_1^{\sigma_1}\ln P_1\right]^2}{[z(\sigma_1, y)]^2}$$

From the Schwartz inequality, the second part of this expression is less than or equal to the first, so the whole term is negative. In the same way, the second term is negative, establishing that $1 - s = (1-r)/2$ yields a maximum of $E(s, r)$. Substituting this into Equation (B.26), we get

$$E(s) = \frac{s}{1-s}(1-R)\ln 2 - \ln \int_0^\infty \left[P_0^{1-s} + P_1^{1-s}\right]^{1/(1-s)} dy \tag{B.30}$$

B.5 General BSC

In order to maximize Equation (3.61) over s and r, and thereby minimize our upper bound to $\bar{P}_e$ for the BSC, we could simply combine Equations (3.61), (3.62), and (3.63), and then set the partial derivatives with respect to s, r, and λ equal to 0. This procedure is tedious, and it is difficult to demonstrate that the stationary point so found is indeed the maximum over s, r of the minimum over λ. However, we recall that Equations (3.61) to (3.64) were derived by eliminating $f(y)$ from Equation (3.37).

For the BSC, it makes no difference what $f(y)$ is. Due to the symmetry condition, Equation (3.5), $f(+1) = f(-1)$, and thus $f(y)$ is specified by one value. However, we showed that multiplying $f(y)$ by a constant does not change E, so that $f(y)$ can be chosen as 1 for the BSC. At this point, we can return to Equation (3.37) and minimize this directly. We have, letting $p = P_0(-1)$

$$\begin{aligned}\bar{P}_e \le \max_{\lambda} \min_{s,r,d} \Big\{ &\exp n[\ln g(s) - sd] \\ &+ nC_n \exp n\big[B(\lambda) + \lambda \ln h(r) + (1-\lambda)\ln g(r) - rd\big]\Big\} \end{aligned} \tag{B.31}$$

$$\begin{aligned} g(s) &= p^{1-s} + (1-p)^{1-s} \\ h(r) &= 2p^{\frac{1-r}{2}}(1-p)^{\frac{1-r}{2}} \end{aligned} \tag{B.32}$$

To minimize Equation (B.31) over s, we can simply minimize $[\ln g(s) - sd]$

$$d = \frac{p^{1-s}\ln(1/p) + (1-p)^{1-s}\ln[1/(1-p)]}{p^{1-s} + (1-p)^{1-s}} \tag{B.33}$$

if d is in the proper range to make $0 \le s < \infty$.

To see that Equation (B.33) actually minimizes $\bar{P}_e$, we can show that

$$\frac{\partial^2[\ln g(s) - sd]}{\partial s^2} \ge 0 \tag{B.34}$$

This can be done either by straightforward but tedious differentiation or by recalling that the second derivative of a semi-invariant generating function $[\ln g(s)]$ is always positive [4]. Likewise, minimizing over r gives us

$$d = \frac{-\lambda}{2} p(1-p) + (1-\lambda)\frac{p^{1-r}\ln(1/p) + (1-p)^{1-r}\ln[1/(1-p)]}{p^{1-r} + (1-p)^{1-r}} \tag{B.35}$$

if d is in the proper range to make $-\infty < r \le 0$. Finally we can minimize over d by making the two exponents equal.

$$\begin{aligned}&\ln\big(p^{1-s} + (1-p)^{1-s}\big) - sd \\ &= B(\lambda) + \lambda \ln 2 + \frac{\lambda(1-r)}{2}\ln p(1-p) + (1-\lambda)\ln\big(p^{1-r} + (1-p)^{1-r}\big) - rd \end{aligned} \tag{B.36}$$

Equations (B.34), (B.35) and (B.36) can be used in principle to solve for s, r, and d in terms of λ if a solution exists with $0 \leq s < \infty$; $-\infty < r \leq 0$. To simplify these equations, first combine Equations (B.33) and (B.35) to eliminate d.

$$-p_s \ln p - (1-p_s)\ln(1-p) = -\frac{\lambda}{2}\ln p(1-p) - (1-\lambda)\big[p_r \ln p + (1-p_r)\ln(1-p)\big] \tag{B.37}$$

where

$$\begin{aligned} p_s &= \frac{p^{1-s}}{p^{1-s} + (1-p)^{1-s}} \\ p_r &= \frac{p^{1-r}}{p^{1-r} + (1-p)^{1-r}} \\ p_r &\leq p \leq p_s \end{aligned} \tag{B.38}$$

The third condition in Equation (B.38) is required by the condition $s \geq 0, r \leq 0$.

Rearranging Equation (B.37), we get

$$\begin{aligned} \Big(-p_s + \frac{\lambda}{2} + (1-\lambda)p_r\Big)\ln p &= \Big[(1-p_s) - \frac{\lambda}{2} - (1-\lambda)(1-p_r)\Big]\ln(1-p) \\ &= \Big[-p_s + \frac{\lambda}{2} + (1-\lambda)p_r\Big]\ln(1-p) \\ p_s &= \frac{\lambda}{2} + (1-\lambda)p_r \end{aligned} \tag{B.39}$$

Equation (B.36) can also be simplified if we add d to each side, then substitute Equation (B.34) in the left side of Equation (B.36), and substitute Equation (B.35) in the right side of Equation (B.36). After some simplification, this yields

$$H(p_s) = B(\lambda) + \lambda \ln 2 + (1-\lambda)H(p_r) \tag{B.40}$$

Also, since we have set the exponents in Equation (B.31) equal, we can simplify the expression for $\bar{P}_e$. Proceeding in the way used to get Equation (B.40), we obtain

$$\bar{P}_e \leq \max_{\lambda}(1 + nC_n)\exp -n\Big[-H(p_s) + p_s \ln\frac{1}{p} + (1-p_s)\ln\frac{1}{1-p}\Big] \tag{B.41}$$

where p_s satisfies Equations (B.39) and (B.40) and $p_r \leq p \leq p_s$.

Note from Figure 3.4 that the $\bar{P}_e$ in Equation (B.41) is decreasing with p_s, so that maximizing λ means to find the λ for which the p_s satisfying Equations (B.39) and (B.40) is minimized. A simpler formulation for this λ can be found from Equation (B.31), from which λ is chosen to maximize

$$\Big[B(\lambda) + \lambda \ln\frac{h(r)}{g(r)}\Big] = B(\lambda) + \frac{\lambda}{2}\ln 4p_r(1-p_r) \tag{B.42}$$

C Analysis of Number of Independent Decoding Iterations

An asymptotic bound on the probability of decoding error using probabilistic decoding was developed in Chapter 4, Equation (4.19). This bound was given as a function of the number of decoding iterations. In this appendix, upper and lower bounds will be derived on the maximum number of decoding iterations m that can be achieved with an (n, j, k) code before the independence assumption of Theorem 4.1 becomes invalid. We shall show first that for any (n, j, k) code m must be upper bounded by

$$m < \frac{\log n}{\log(k-1)(j-1)} \tag{C.1}$$

Second, and more important, a construction procedure will be described by which it is always possible to find an (n, j, k) code satisfying

$$m + 1 > \frac{\log n + \log \frac{kj-k-j}{2k}}{2\log(k-1)(j-1)} \geq m \tag{C.2}$$

Note that for large n, the m given by Equation (C.2) is approximately half that given by Equation (C.1).

Theorem C.1. *Let m be the largest number of independent decoding iterations possible for any code of block length n with k digits per parity-check and j parity-check sets per digit. Then*

$$m < \frac{\log n}{\log(k-1)(j-1)}$$

Proof. Consider an m-tier parity-check set tree for any digit in any (n, j, k) code. To achieve m independent decoding iterations, each node of this tree must correspond to a separate digit in this code. Thus, the number of nodes in the m-tier tree must be at most equal to the block length n. The first tier contains $(k-1)$ nodes for each of the j branches rising from the base node. Thus, the first tier contains $j(k-1)$ digits. Each of these digits gives rise to $(j-1)(k-1)$ digits on the second tier since only $(j-1)$ branches rise from each node on the first tier. Thus, there are $j(j-1)(k-1)^2$ digits on the second tier. Similarly, there are $j(j-1)^{i-1}(k-1)^i$ digits on the i^{th} tier. Thus,

$$1 + j(k-1) + j(j-1)(k-1)^2 + \cdots + j(j-1)^{i-1}(k-1)^i + j(j-1)^{m-1}(k-1)^m \leq n \tag{C.3}$$

The expression on the left of Equation (C.3) is lower bounded by its last term, and that in turn is lower bounded by $(j-1)^m(k-1)^m$, hence

$$(j-1)^m(k-1)^m < n \tag{C.4}$$

Taking the logarithm of both sides of Equation (C.4), we get Equation (C.1), proving the theorem. □

Equation (C.3) can also be summed exactly to get

$$1 + j(j-1)^{m-1}(k-1)^m \left[\frac{1-(j-1)^{-m}(k-1)^{-m}}{1-(j-1)^{-1}(k-1)^{-1}}\right] \leq n \tag{C.5}$$

The complexity of Equation (C.5), however, makes it less useful than Equation (C.1).

Before describing a construction procedure to satisfy Equation (C.2), a relationship will be established between m and the relative locations of the 1's in the parity-check matrix. Define a *closed path* in parity-check matrix to be a sequence of connected alternating horizontal and vertical lines with the following properties: First, the last line in the sequence terminates at the beginning of the first line; second, each vertex is at a point where the parity-check matrix contains a 1. A vertex is here defined as a connection point between successive lines in the sequence, including that between the last and first lines (see Figure C.1). Define the length of a closed path as the number of lines in the sequence. For example, the closed path in Figure C.1 has length 6. Note that a horizontal or vertical line can pass through other lines and other ones in the matrix and is still counted as one line. We allow the sequence of lines associated with a closed path to start with any line in the path and go in either direction.

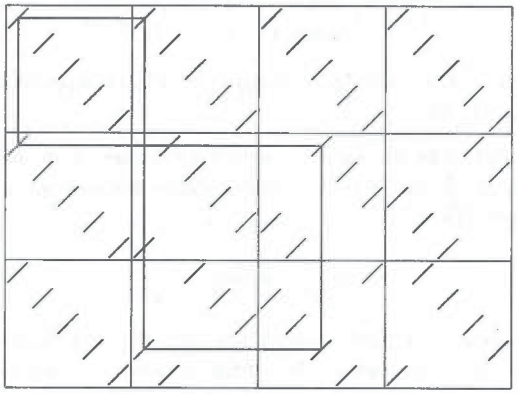


Figure C.1: Example of closed path of length 6. Blanks = 0's; slashes = 1's.

Lemma C.1. *If one or more closed path of length L exists in a parity-check matrix and no closed path of length less than L exists, then m, the number of independent decoding iterations, satisfies*

$$m < \frac{L}{4} \leq m + 1 \tag{C.6}$$

Proof of First Half of Inequality. Consider a particular closed path of length L. There are $L/2$ vertical lines in this path, each corresponding to a digit in the code. Call these digits, in order of their appearance along the closed path a_1,

$a_2, \ldots, a_{L/2}$. For the closed path shown in Figure C.1, we could have $a_1 = 1$, $a_2 = 6$, and $a_3 = 13$. Consider the parity-check set tree associated with digit $a_{L/2}$ (see Figure C.2), and consider the two paths in this tree formed by $a_{L/2}$, $a_{(L/2)-1}, \ldots, a_{L/4}$ (or $a_{(L/4)+1/2}$) and $a_{L/2}, a_1, a_2, \ldots, a_{L/4}$ (or $a_{(L/4)+1/2}$). Note that $a_{L/2}$ appears on tier 0; a_1 and $a_{(L/2)-1}$ on tier 1, and in general a_i and $a_{(L/2)-i}$ on tier i. If $L/4$ is an integer, then $a_{L/4}$ must appear twice on the $L/4^{\text{th}}$ tier and thus $m < L/4$. Alternatively, if $(L/4) + 1/2$ is an integer, then $a_{(L/4)+1/2}$ appears once on the $(L/4) - \frac{1}{2}^{\text{th}}$ tier and once on the $(L/4) + \frac{1}{2}^{\text{th}}$ tier. In this case, $m \leq L/4 - \frac{1}{2} < L/4$, completing the proof that $m < L/4$. □

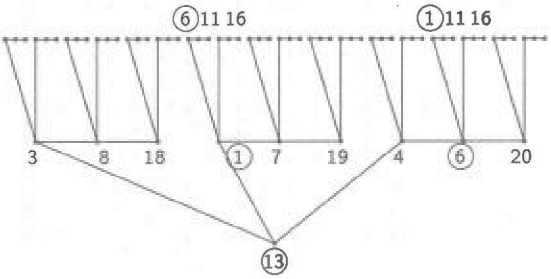
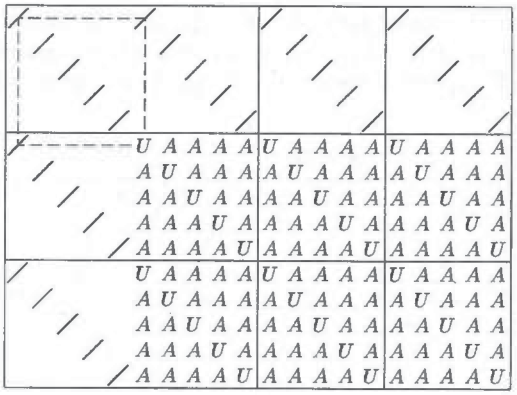


Figure C.2: Closed path of Figure C.1 in a parity-check set tree.

Proof of Second Half of Inequality. If a code has only m independent decoding iterations, then for some digit in the code, say d, the parity-check set tree contains a digit on tier $m+1$, say a_0, that has appeared elsewhere either on tier $m+1$ or on a lower tier. Now let a_1 and b_1 be the digits immediately below the two appearances of a_0 on the parity-check set tree; let a_2 and b_2 be the digits underneath them, and so forth down to digit d. The number of digits in the set $a_0, d, a_1, \ldots, b_1, \ldots$; is at most $2(m+1)$. Finally, consider drawing a closed path in the parity-check matrix starting with a_0 and the parity check set containing a_1 and a_2, and so on down to digit d and back up to a_0 via the b's. This closed path contains twice as many lines as digits, so that $L \leq 4(m+1)$ proving the theorem. □

A procedure will now be described for constructing parity-check matrices with no closed paths of length $L = 4m$ or less. The procedure will be followed by a proof the that construction can be carried out whenever Equation (C.2) is satisfied. Consider and nj/k by n matrix such as in Figure C.3. The matrix has been divided into jk square submatrices, each with n/k rows and columns. The first row of submatrices and the first column of submatrices are all identity matrices. The other submatrices contain the letter U in each position on the main diagonal and the letter A in each position off the main diagonal. Our object is to replace each submatrix containing A's and U's with permutations of the n/k by n/k identity matrix in such a way as to form no closed paths of length $4m$ or less. The letter A is used to denote an acceptable position



U A A A A U A A A A U A A A A
A U A A A A U A A A A U A A A
A A U A A A A U A A A A U A A
A A A U A A A A U A A A A U A
A A A A U A A A A U A A A A U
U A A A A U A A A A U A A A A
A U A A A A U A A A A U A A A
A A U A A A A U A A A A U A A
A A A U A A A A U A A A A U A
A A A A U A A A A U A A A A U

Figure C.3: Initial stage of construction procedure.

in which to place a 1 without forming any closed paths of length $4m$ or less. The letter U denotes an unacceptable position; these are positions in which a 1 would create a closed path of length $4m$ or less. Note that even for $m = 1$, the main diagonals contain U's because of the closed paths of length 4 such as that shown by the dotted line in Figure C.3.

Next pick a submatrix containing A's and U's, and in the first row, select some position containing an A and replace that A with a 1. Also place a 0 in front of each letter in the submatrix that is in the same row or column as that 1. Finally, for those positions in the matrix in which 1's can no longer be placed without creating a closed path of length $4m$ or less, replace the A with a U. This yields a matrix such as that in Figure C.4

Continue with row 2 of the submatrix, replacing some position containing an A (not $0A$) with a 1, filling in that row and column by 0's and changing A's to U's when necessary. Continue in this way until each row of the submatrix contains a 1 and go through each submatrix in this way. If, at some point in this process, a row is encountered, say the ℓ^{th}, in which no position contains an A without an accompanying 0, go through the following "emergency" procedure.

Let c_ℓ be a column in which row ℓ contains a U. Denote this by $P(\ell, c_\ell) = U$, where $P(i, j)$ is defined as the symbol appearing in the i^{th} row of the jth column of the submatrix. For each $i < \ell$, define c_i as the column for which $P(i, c_i) = 1$. Now find an i for which $P(i, c_\ell) = 0A$ and $P(\ell, c_i) = 0A$ (see the circled entries in Figure C.5.) For this i, change $P(i, c_\ell)$ to 1, $P(\ell, c_i)$ to 1, $P(i, c_i)$ to $0A$ and modify the A's U's and $0A$'s throughout the matrix to correspond to this new set of 1's.

For this emergency procedure to work, it is necessary first to prove that making both $P(i, c_i) = 1$ and $P(\ell, c_\ell) = 1$ simultaneously does not form any closed path of length $4m$ or less. Also, it is necessary to prove that if Equation (C.2) is satisfied, then an i always exists such that $P(i, c_\ell) = 0A$ and $P(\ell, c_i) = 0A$.

$$
\left[\begin{array}{cccccccccc|ccccc|ccccc}
/ & & & & & / & & & & & / & & & & & / & & & & \\
 & / & & & & & / & & & & & / & & & & & / & & & \\
 & & / & & & & & / & & & & & / & & & & & / & & \\
 & & & / & & & & & / & & & & & / & & & & & / & \\
 & & & & / & & & & & / & & & & & / & & & & & / \\
\hline
/ & & & & & 0U & / & 0A & 0A & 0A & U & A & A & A & A & U & A & A & A & A \\
 & / & & & & A & 0U & A & A & A & A & U & A & A & A & A & U & A & A & A \\
 & & / & & & A & 0A & U & A & A & A & A & U & A & A & A & A & U & A & A \\
 & & & / & & A & 0A & A & U & A & A & A & A & U & A & A & A & A & U & A \\
 & & & & / & A & 0A & A & A & U & A & A & A & A & U & A & A & A & A & U \\
\hline
/ & & & & & U & A & A & A & A & U & A & A & A & A & U & A & A & A & A \\
 & / & & & & A & U & A & A & A & A & U & A & A & A & A & U & A & A & A \\
 & & / & & & A & A & U & A & A & A & A & U & A & A & A & A & U & A & A \\
 & & & / & & A & A & A & U & A & A & A & A & U & A & A & A & A & U & A \\
 & & & & / & A & A & A & A & U & A & A & A & A & U & A & A & A & A & U
\end{array}\right]
$$

Figure C.4: Second step in matrix construction.

The first point will be proved by contradiction. Assume that setting $P(i, c_i) = 1$, $P(\ell, c_\ell) = 1$, and $P(i, c_i) = 0$ forms a closed path of length $4m$ or less. This path must contain both points i, c_ℓ and ℓ, c_i as vertices, since the $0A$'s formerly in these positions indicated that no closed path of length $4m$ or less existed through either point alone. Consider tracing round this closed path starting at ℓ, c_i along the horizontal line. There are two cases to be considered: First, the path comes to i, c_ℓ along a horizontal line as in Figure C.6; second, the path comes to i, c_ℓ along a vertical line as in Figure C.7.

For case 1, set $P(i, c_\ell) = 0$, $P(i, c_i) = 1$, and terminate the horizontal line coming into i, c_ℓ on point i, c_i, as in Figure C.6. Then close the path by moving vertically to ℓ, c_i. This path has a length less than $4m$ since it is shorter than the original path. However, this contradicts the assumption that ℓ, c_i was an acceptable point when $P(i, c_i)$ was equal to 1.

For case 2, set $P(i, c_\ell) = 0$, $P(\ell, c_i) = 0$, and $P(i, c_i) = 1$. Now make the vertical line previously terminating on ℓ, c_i terminate on i, c_i, and the horizontal line previously originating on i, c_ℓ originate on i, c_i (see Figure C.7). This forms a closed path of length less than $4m$ involving neither i, c_ℓ nor ℓ, c_i. This is also a contradiction since no 1's are placed in the matrix in such a way as to form a closed path of length $4m$ or less. This completes the proof that $P(\ell, c_i)$ and $P(i, c_\ell)$ may simultaneously be set equal to 1 if they are both labelled $0A$.

To complete the proof, we must show that if Equation (C.2) is satisfied, it is always possible in this emergency condition to find an i such that $P(\ell, c_i) = 0A$ and $P(i, c_\ell) = 0A$. First, we shall show that Equation (C.2) implies that there are no more than $n/2k$ values of i for which $P(\ell, c_i) = 0A$. Second, we shall

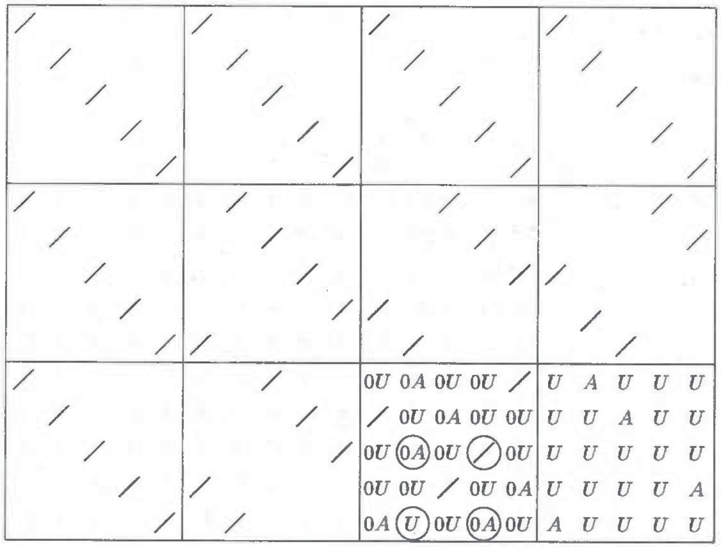


Figure C.5: Example of emergency procedure in matrix construction.

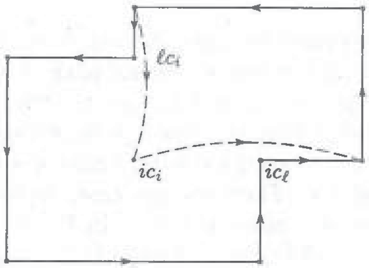


Figure C.6: Case 1: Closed path through ℓc_i and ic_ℓ.

show that more than $n/2k$ elements in column c_ℓ of the submatrix contain $0A$'s. These two relations will complete the proof since if $P(i, c_\ell) \neq 0A$ for all i for which $P(\ell, c_i) = 0A$, then column c_ℓ will contain more than $n/2k$ non-$0A$'s and more than $n/2k$ $0A$'s. But this is impossible since column ℓ of the submatrix contains only n/k elements. Thus, there must be an i for which $P(i, c_\ell) = 0A$ and $P(\ell, c_i) = 0A$.

We now bound the number of points in row ℓ that can be labelled U. If a point in the ℓ^{th} row of the submatrix is unacceptable, then a 1 placed at that point would cause a closed path of length $4m$ or less. We shall consider the first line of that closed path to be the horizontal line on row ℓ originating at the unacceptable point. The last line will then be the vertical line terminating at the unacceptable point. We first ask how many closed paths of length 4 can exist starting at an unacceptable point in row ℓ of the submatrix. There are

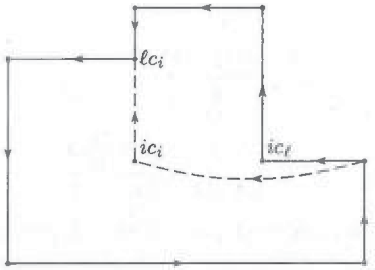


Figure C.7: Case 2: Closed path through ℓc_i and ic_ℓ.

at most $(k-1)$ points where this first horizontal line can terminate, namely, those positions in row ℓ of the complete matrix in which ones have already been placed. The vertical line from any one of these $(k-1)$ points can terminate in at most $(j-1)$ points, namely the other positions in the column in which ones have been placed. Remember that in this construction we never placed more than k 1's in a row or over j 1's in a column. Finally, if a closed path of length 4 is to be constructed, the next horizontal line from any of these $(j-1)$ points must terminate in a column contained in the submatrix under consideration, and there is at most a single 1 in any of those columns on which that horizontal like can terminate. Thus there are at most $(k-1)(j-1)$ different closed paths of length 4 that can have an unacceptable point in row ℓ of the given submatrix as a vertex. Consequently, at most $(k-1)(j-1)$ points in row ℓ of the given submatrix are unacceptable because of closed paths of length 4. The same argument can be used on closed paths of length 6. Here, for any of the $(k-1)(j-1)$ paths of length 2, there are at most $k-1$ points on which the third line can terminate, and for each of these, at most $(j-1)$ points on which the fourth line can terminate. The fifth line is now determined since it must terminate in a column of the given submatrix. Hence at most $(k-1)^2(j-1)^2$ points in row ℓ of the given submatrix are unacceptable because of closed paths of length 6. Similarly, closed paths of length $2i$ can make at most $(k-1)^{i-1}(j-1)^{i-1}$ points unacceptable. Thus, the total number of unacceptable points in row ℓ of the submatrix N_u is bounded by

$$\begin{aligned} N_u &\le \sum_{i=2}^{2m}(k-1)^{i-1}(j-1)^{i-1} \\ &= (k-1)^{2m-1}(j-1)^{2m-1}\left\{\frac{1-[(k-1)(j-1)]^{-(2m-1)}}{1-[(k-1)(j-1)]^{-1}}\right\} \\ &< \frac{(k-1)^{2m-1}(j-1)^{2m-1}}{1-[(k-1)(j-1)]^{-1}} \\ &= \frac{[(k-1)(j-1)]^{2m}}{kj-k-j} \end{aligned} \tag{C.7}$$

Thus $N_u \leq n/2k$ if

$$\frac{[(k-1)(j-1)]^{2m}}{kj-k-j} \leq \frac{n}{2k}$$

or

$$m \leq \frac{\log n + \log \frac{kj-k-j}{2k}}{2\log(k-1)(j-1)}$$

Since all the elements in row ℓ of the given submatrix are either $0A$ or U, Equation (C.2) implies that more than $n/2k$ of the elements in row ℓ are $0A$'s.

Finally, we must show that more than $n/2k$ elements in column c_ℓ of the submatrix are labelled $0A$. The argument is identical to that last argument with the exception that instead of constructing paths starting with horizontal lines from the unacceptable digit, we start with a vertical line. Equation (C.7) still gives a bound on the number of unacceptable points, and Equation (C.2) still guarantees that over $n/2k$ points are labelled $0A$ since all the elements in column c_ℓ are $0U$'s or $0A$'s. Thus, we have demonstrated a constructive procedure for generating codes in which m independent decoding iterations can be performed where m satisfies Equation (C.2).

References

[1] F. J. Bloom, S. S. L. Chang, et al. Improvements of binary transmission by null-zone reception. *Proceedings of the IRE*, 45:963–975, 1957.

[2] R. C. Bose and D. K. Ray-Chaudhuri. On a class of error-correcting binary group codes. *Information and Control*, 3:68–79, 1960.

[3] P. Elias. Coding for two noisy channels. In Colin Cherry, editor, *Information Theory*, Third London Symposium, London, England, September 1955. Butterworth's Scientific Publications.

[4] R. M. Fano. *Transmission of Information.* The M.I.T. Press and John Wiley & Sons, Inc., New York, 1961.

[5] R. M. Fano. A heuristic discussion of probabilistic decoding. *IEEE Transactions on Information Theory*, IT–9:62–71, 1963.

[6] E. N. Gilbert. A comparison of signaling alphabets. *Bell System Technical Journal*, 31:504–522, 1952.

[7] B. V. Gnedenko and A. N. Kolmogorov. *Limit Distributions for Sums of Independent Random Variables.* Addison Wesley Publishing Company, Cambridge, Massachusetts, 1954.

[8] C. W. Helstrom. Resolution of signals in white Gaussian noise. *Proceedings of the IRE*, 43:1111–1118, September 1955.

[9] I. L. Lebow, P. G. McHugh, A. C. Parker, P. Rosen, and J. M. Wozencraft. Application of sequential decoding to high rate data communication on a telephone line. *IEEE Transactions on Information Theory*, IT–9, April 1963.

[10] J. L. Massey. *Threshold Decoding.* The M.I.T. Press, Cambridge, Massachusetts, 1963.

[11] K. M. Perry and J. M. Wozencraft. SECO: A self regulating error correcting coder-decoder. *IRE Transactions on Information Theory*, IT–8(5):129–135, September 1962.

[12] W. W. Peterson. *Error-Correcting Codes.* The M.I.T. Press and John Wiley & Sons, Inc., New York, 1961.

[13] J. R. Pierce. Theoretical diversity improvement in frequency shift keying. *Proceeding of the IRE*, 46:903–910, 1958.

[14] B. Reiffen. Sequential decoding for discrete input memoryless channels. *IRE Transactions on Information Theory*, IT–8(3):208–220, April 1962.

[15] C. E. Shannon. Certain results in coding theory for noisy channels. *Information and Control*, 1:6–25, 1957.

[16] J. M. Wozencraft and M. Horstein. Coding for two way channels. In *Fourth London Symposium on Information Theory*, September 1960.

[17] J. M. Wozencraft and B. Reiffen. *Sequential Decoding.* The M.I.T. Press and John Wiley & Sons, Inc., New York, 1961.

[18] N. Zierler. A class of cyclic linear error-correcting codes in p^m symbols. Group Report 55–19, M.I.T. Lincoln Laboratory, Lexington, Massachusetts, January 1960.

Index