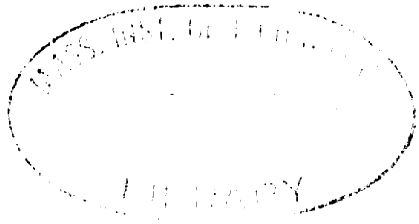

LOW DENSITY PARITY CHECK CODES

by

ROBERT GRAY GALLAGER

B.S., University of Pennsylvania
(1953)

M.S., Massachusetts Institute of Technology
(1957)

SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
September, 1960

Signature of Author _____
Department of Electrical Engineering, August 26, 1960

Certified by Peter Elias
Thesis Supervisor

Accepted by _____
Chairman, Departmental Committee on Graduate Students

LOW DENSITY PARITY CHECK CODES

by

ROBERT GRAY GALLAGER

Submitted to the Department of Electrical Engineering on August 26, 1960 in partial fulfillment of the requirements for the degree of Doctor of Science.

ABSTRACT

An ensemble of parity check codes of arbitrary block length, n is considered in which each digit of each code is checked by a small fixed number, j, of parity check equations, and each parity check set contains a small, fixed number, k, of digits. The typical minimum distance of codes in such an ensemble increases linearly with n for constant j and k if $j \geq 3$.

The probability of decoding error for this ensemble of codes on a memoryless symmetric channel with a binary input alphabet and an arbitrary output alphabet is analyzed. Using maximum likelihood decoding on a sufficiently quiet channel, the probability of error is shown to be exponentially decreasing with n; this exponent is relatively close to the theoretical optimum exponent.

A simple decoding scheme that directly uses the channel a posteriori probabilities is described in which the decoding computation per digit appears to be constant, or at most, logarithmically increasing with the code length. The probability of decoding error is shown by a weak bound to approach zero with increasing block length when this non-optimum decoding scheme is used on a Binary Symmetric Channel of sufficiently high capacity.

Although no tight bounds have been found for the probability of decoding error using this simple decoding scheme, both some experimental results and the form of the weak bound indicate the potentiality of the decoding scheme.

Thesis Supervisor: Peter Elias
Title: Professor of Electrical Engineering

ACKNOWLEDGMENT

The author gratefully acknowledges both the Research Laboratory of Electronics and the M. I. T. Computation Center for the facilities used in this research. He is also indebted to his associates in R. L. E. for their patience in listening to many half-formed ideas and proofs. He also wishes to publicly thank his wife, Ruth, for her aid in editing and the typing of this paper.

The author is especially grateful to Professor Robert M. Fano, not only for assistance in this work, but also for his consistent encouragement and excellent advice over the past four years.

Finally, the author particularly wishes to thank Professor Peter Elias for the time, quickness, and insight that aided this work in immeasurable ways.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Coding for Digital Data Transmission

Coding for error correction is one of the many tools available for achieving reliable data transmission in communication systems. For a wide variety of channels, the Noisy Channel Coding Theorem (11) of Information Theory indicates what can be achieved through coding. This remarkable theorem proves that if properly coded information is transmitted at a rate below channel capacity, then the probability of decoding error can be made to approach zero exponentially with the code length. The theorem does not, however, relate the code length to the cost in storage, computation, and equipment necessary to achieve this low error probability. Ideally, one wants an inexpensive coding and decoding scheme that is applicable to a wide variety of channels and that achieves the low error probability possible with long codes. The coding and decoding scheme involving low density parity check codes to be described in this paper is one of a number of recent schemes attempting to approach this ideal. A rough comparison of these schemes will be given in a later section.

In order to mathematically prove some results about

low density parity check codes, we will 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 only be applied to channels that are good approximations of the model. However, when the probability of decoding error for a code on an idealized channel is 10^{-20}, it is difficult to determine what constitutes a "good" approximation. The analysis of a code on an idealized channel often provides insight about the effect of various approximations, but such insight should be used with caution.

The channel model to be assumed here is a binary input channel that is memoryless and symmetric at the input. Binary input means that the channel has a transmitter that accepts a signal from a binary alphabet once each unit of time and transmits a waveform corresponding to that input over the channel. The results in this paper may be extended to multi-input channels also, but this will be done elsewhere. Memoryless means that given the input over a unit of time, the output over the corresponding unit of time is statistically independent of the inputs and outputs at all other times. A symmetric input channel will be defined precisely later, but loosely, it is a channel in which the noise affects both input symbols in a symmetric fashion. The Binary Symmetric Channel, abreviated 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 cross-over from one input to the other output.

If a binary symmetric input memoryless 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 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 informations 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 ref. (1). The decoding

scheme to be described here avoids this intermediate decision by using the probability that a transmitted digit is a 1 conditional on the received symbol in the decoding process.

A parity check code * is a code in which the code words are those sequences satisfying a set of linear homogeneous modulo 2 equations called parity check equations. This set of equations can be represented by a parity check matrix, as in Fig. 1-1, in which each row represents a parity check equation. A parity check set is defined as the set of positions in which a row contains ones, or, in other words, as the set of digits checked by a parity check equation. In matrix terminology, the set of code words is simply the null space of the parity check matrix over the modulo 2 field.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	1	1	1	0	1	0	0
n(1-R)	1	1	0	1	0	1	0
	1	0	1	1	0	0	1

n

$$x_5 = x_1 + x_2 + x_3$$
$$x_6 = x_1 + x_2 + x_4$$
$$x_7 = x_1 + x_3 + x_4$$

Figure 1-1

EXAMPLE OF PARITY CHECK MATRIX

* For a more detailed discussion of parity check codes see Slepian (13) or Peterson(9).

Restricting our attention to parity check codes is not serious as far as the probability of decoding error is concerned. Elias[3] has shown that for a BSC, a randomly chosen parity check code of long block length and rate reasonable close to channel capacity will have essentially the same probability of decoding error as could be achieved with any code.

The mapping from information sequences into code words for parity check codes can be simply accomplished in several ways. One way is to diagonalize the parity check matrix, which can be done without changing the set of code words. Then information digits can be used for the non-diagonalized digits and the diagonalized digits can be computed as the modulo 2 sum of the information digits. The decoding problem, that of separating the noise from the code word, is the difficult problem.

Low Density Parity Check Codes

In order to find a simple decoding scheme, a special class of parity check codes will be defined. Low density parity check codes are codes specified by a parity check matrix containing mostly zeros and only a small number of ones. In particular, an (n,j,k) low density parity check code is defined as a code of block length n in which each column of the parity check matrix contains j ones and each row contains k ones. These codes are not optimum in the somewhat artificial sense of minimizing probability of

decoding error for a given block length, and it will be shown in Chapter III that the rate at which these codes can be used is bounded below channel capacity. However, a very simple decoding scheme exists for these codes that compensates for their lack of optimality.

Summary of Results

An ensemble of (n,j,k) codes will be found in Chapter II 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, all code words have the same set of distances to the other code words since the code words form a group. 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. Fig. 2-4 plots the ratio of minimum distance to block length for several values of j and k and compares it with the typical minimum distance of an ordinary parity check code. (n,j,k) codes with $j = 2$ exhibit markedly different behavior and it is shown that the minimum distance of an (n,j,k) code can increase at most logarithmically with the block length.

In Chapter III, the distance properties derived in Chapter II are first used to bound the probability of decoding error for an (n,j,k) code on a BSC assuming maximum likelihood decoding. The maximum likelihood assumption permits an evaluation of the codes themselves that is independent of the particular decoding scheme. It is shown that for reasonably large channel cross-over probabilities, the probability of decoding error for a typical (n,j,k) code has the same behavior as that of optimum parity check codes of slightly higher rate. Fig. 3-1 illustrates this loss of rate associated with (n,j,k) codes.

Next the probability of decoding error for (n,j,k) codes on a general binary symmetric input memoryless channel is derived. It is shown that the probability of decoding error decreases exponentially with block length, but the exponent is given as the solution of three simultaneous transcendental equations. A relatively simple numerical procedure is given to find the exponent for any particular channel. The techniques of Chapter III are also applicable to any parity check code or ensemble of parity check codes for which the distance properties are known.

In Chapter IV, 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 other decoding scheme is based on a procedure for computing the probability that a transmitted digit is a one conditional 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 code length. The probabilistic scheme entails slightly more computation than the first scheme, but decodes much better.

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 to the probability of error is derived that decreases exponentially with a root of the code length. Fig. 3-1 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 V presents some experimental results that indicate that the decoding scheme is much better than would be expected from the bound in Chapter IV. All the experimental results are for very noisy channels on which the probability of decoding error is high; many more data are needed for less noisy channels.

Comparison With Other Schemes

The two other coding and decoding schemes that appear most capable of achieving the low error probabilities associated with long codes at a reasonable cost are convolutional codes[3] with Wozencraft's[14] Sequential Decoding Scheme and the Bose-Chaudhuri Codes[2] with Petersen's[9] decoding scheme. Both these schemes and the low density scheme presented here are still in a stage of development and it is difficult to predict the potentialities of any of them.

Sequential decoding for the BSC is hypothesized to have, first, a probability of decoding error that decreases exponentially with the constraint length, n, at the random-coding exponent, and second, an average number of computations per digit that increases as n^{β} where B is a constant less than 1. In order to rigorously prove this bound to probability of decoding error, however, one must allow a number of computations growing as n^{2+2B}.[10] The experimental evidence[8] indicates that the actual computation required is below the hypothetical n^{β} bound.

The principal drawback of sequential decoding is that the amount of computation necessary to decode varies considerably with the noise. The flexibility of the decoding scheme allows this variation to be cut down in many reasonable ways so that the problem might not be serious in practice. However, more experimental data are needed to get a close relationship between the peaks of computation required and the probability of error.

Sequential decoding can also be applied to a wide variety of memoryless channels, as shown by Reiffen[10]. In particular, sequential decoding is capable of making full use of the a posteriori probabilities at the outputs of the channels considered here.

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, the number of errors correctable is roughly half the number correctable by the equivalent sequential decoding scheme. 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 probability 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. Thus low density codes require less computation than either of the other schemes for long block length, but the asscciated probability of error is still open to question.

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 take account of 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.(9)

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.(1),(15) All three of the coding and decoding schemes being considered here fit naturally into such a system, but in cases when little or no error correction is attempted, low density codes appear at a disadvantage.

The conclusion one can draw from this is that all three schemes are sufficiently promising to merit further investigation.

CHAPTER II

DISTANCE PROPERTIES

Ensemble of all Parity Check Codes

The analysis of the distances between code words in a code is usually difficult because of the immense number of code words involved. Often it is simpler to analyze a whole ensemble of codes because the statistics of an ensemble permit one to average over quantities that are not tractable in individual codes. From the ensemble behavior, one can make statistical statements about the properties of the member codes.

This chapter will be principally concerned with the distance properties of low density codes, but for comparison some distance properties of an ensemble of all group 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. Consider the ensemble of $n(1-R)$ by n matrices of binary digits in which each element is zero or one with probability $\frac{1}{2}$ and all elements are statistically independent. This will be called the ensemble of all parity check codes of length n and rate R and is essentially the same as that considered by Elias.[3] Note that some codes in

the ensemble may have a rate greater than R since the rows of a matrix in this ensemble are not necessarily linearly independent over the modulo 2 field. Over this ensemble, the minimum distance, D, of a code is a random variable, and its distribution function may be bounded by the following theorem:

Theorem 1:

Over the ensemble of all parity check codes of length n and rate R, the minimum distance distribution function is bounded by both

$$\Pr\left[D \leq \delta n\right] \leq \frac{1}{1-2\delta}\sqrt{\frac{1-\delta}{2\pi n\delta}} \exp n\left[H(\delta) - (1-R)\ln 2\right] \qquad (2.1)$$

and

$$\Pr\left[D < \delta n\right] \leq 1$$

where

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

Proof:

Let P(L) be the probability of the set of codes for which some particular sequence of weight L is a code word. Stated differently, P(L) 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 sequence is a code word in any parity check code, P(L) = 1 for L = 0.

For $L \neq 0$, 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 the parity check is satisfied regardless of whether the first L-1 ones were checked an even or an odd number of times. A sequence will be a code word if and only if it satisfies all the $n(1-R)$ parity checks, so that

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

The minimum distance of a parity check code is less than or equal to $n\delta$ if and only if the code contains a code word of weight between 1 and $n\delta$. Since there are $\binom{n}{L}$ sequences of weight L, and since the probability of a union of events is less than or equal to the sum of probabilities of the individual events,

$$\Pr(D \leq n\delta) \leq \sum_{L=1}^{n\delta} \binom{n}{L} P(L) = \sum_{L=1}^{n\delta} \binom{n}{L} 2^{-n(1-R)} \qquad (2.2)$$

For $\delta < \frac{1}{2}$, this can be bounded by a geometric series to give

$$\Pr(D \leq n\delta) \leq \left(\frac{1-\delta}{1-2\delta}\right) \binom{n}{n\delta} 2^{-n(1-R)} \qquad (2.3)$$

Using the Sterling approximation to a factorial,

$$\frac{1}{\sqrt{2\pi n}}\, n^n \exp\left[-n + \frac{1}{12n} - \frac{1}{360n^3}\right] \leq n! \leq \frac{1}{\sqrt{2\pi n}}\, n^n \exp\left[-n + \frac{1}{12n}\right]$$

one can show after some manipulation that

$$\frac{1}{\sqrt{2\pi n\delta(1-\delta)}} \exp\left[nH(\delta) - \frac{1}{12n\delta(1-\delta)}\right] < \binom{n}{n\delta} < \frac{1}{\sqrt{12\pi n\delta(1-\delta)}} \exp nH(\delta) \tag{2.4}$$

where

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

Substituting this in Eq.(2.3), we get Eq.(2.1), proving the theorem.

Q. E. D.

As n gets larger this bound to $Pr(D \leq n\delta)$ as a function of δ approaches a step function with the step at that $\delta_o < \frac{1}{2}$ for which $H(\delta_o) = (1-R)\ln 2$. δ_o is plotted as a function of rate in Fig. 2-4. This result is closely related to the Gilbert bound on minimum distance.[5] The asymptotic form of the Gilbert bound for large n states that there exists a code for which $D \geq n\delta_o$. Theorem 1 states that for any $\epsilon > 0$, the probability of the set of parity check codes that has $D < n(\delta_o - \epsilon)$ approaches 0 exponentially with n.

Distance Properties of Low Density Codes

In this section an ensemble of low density parity check codes will be defined and a theorem similar to Theorem 1 will be proved. Then a new ensemble will be formed by expurgating those codes that have small minimum distances. This modified ensemble will be used in the next chapter to derive bounds on 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 $\frac{nj}{k}$ rows and thus a rate $R \geq 1 - \frac{j}{k}$. In order to construct an ensemble of (n,j,k) matrices, consider first the special (n,j,k) matrix in Fig. 2-1 for which $n = 24$, $k = 6$, and $j = 3$.

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

Figure 2-1

BASE MATRIX FOR AN (n,j,k) ENSEMBLE

This matrix is divided into j blocks of $\frac{n}{k}$ rows each and the i'th row of each block contains ones in positions (i-1)k + 1 to ik. Using this as a "base" matrix we can form new (n,j,k) matrices by separately taking each block of $\frac{n}{k}$ rows and permuting the columns within that block. Note that this type of permutation preserves the k ones in each row and the single one per column for each of the j blocks of $\frac{n}{k}$ rows.

Finally, define an (n,j,k) ensemble of matrices to be the set of all such multiple permutations of the "base" matrix, assigning equal probability to each permutation. Thus the ensemble contains all (n,j,k) matrices for which each of the j blocks of $\frac{n}{k}$ rows contains a single one in each column. The ensemble of (n,j,k) codes is defined by this ensemble of (n,j,k) parity check matrices. This definition is somewhat arbitrary, but will be justified by its analytical simplicity. Before finding the minimum distance distribution function for this ensemble of codes, we will need the following theorem:

Theorem 2:

For each code in an (n,j,k) ensemble, the number, $N[L]$, of sequences of weight L that satisfy any one of the j blocks of $\frac{n}{k}$ parity checks is bounded by

$$N\left[\frac{n}{k}\mu'(s)\right] \leq \exp \frac{n}{k}\left[\mu(s) - s\mu'(s) + (k-1)\ln 2\right] \qquad (2.5)$$

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

$$\mu(s) = \ln 2^{-k}\left[(1+e^{s})^{k} + (1-e^{s})^{k}\right],$$

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

Discussion:

This theorem relates L and N(L) by expressing both as functions of the parameter s. Figure 2-2 sketches L/n and $\frac{\ln N(L)}{n}$ as functions of s.

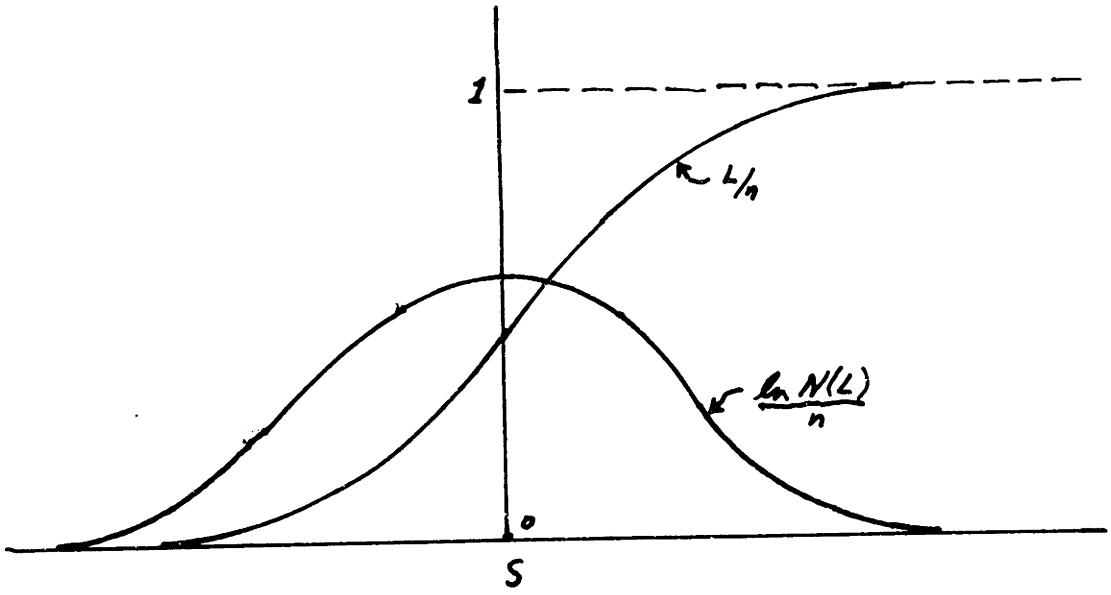

Figure 2-2

PARAMETRIC REPRESENTATION OF L/n AND $\frac{\ln N(L)}{n}$

Proof:

For any code in the ensemble, and for any one of the j blocks of $\frac{n}{k}$ parity checks, the $\frac{n}{k}$ parity check sets within a block are mutually exclusive and exhaust all the digits. Consider the set of all sequences of k binary digits that contain an even number of ones, and construct an ensemble from these sequences 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

$$g(s) = \sum_{i \text{ even}} \binom{k}{i} 2^{-k+1} e^{si} \tag{2.7}$$

or

$$g(s) = 2^{-k}\left[(1+e^{s})^{k} + (1-e^{s})^{k}\right] \tag{2.8}$$

To show that Eqs.(2.7) and (2.8) are equivalent, use the binomial expansion on Eq.(2.8) and observe that odd terms cancel.

For each of the $\frac{n}{k}$ parity check sets, independently choose a sequence from the above ensemble and use that sequence as the digits in that parity check set. This procedure defines an ensemble of equiprobable events in which the events are the n-length sequences satisfying the $\frac{n}{k}$ 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 is the sum of $\frac{n}{k}$ independent random variables each having the moment generating function, g(s), in Eq.(2.8). Consequently, the moment generating function for the number of ones in an n-length sequence is $\left[g(s)\right]^{\frac{n}{k}}$. This is now used to bound the probability, Q(L), in this ensemble, that a sequence has L ones. By definition,

$$\left[g(s)\right]^{\frac{n}{k}} = \sum_{L=0}^{n} \exp(sL)Q(L) \tag{2.9}$$

$$\geq \exp(sL)Q(L) \qquad \text{(for any s and L)} \tag{2.10}$$

From Eq.(2.6) and Eq.(2.8), $\mu(s) = \ln g(s)$, so that

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

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

$$N(L) \leq \exp\left[\frac{n}{k}\mu(s) + \frac{n}{k}(k-1)\ln 2 - sL\right] \tag{2.11}$$

Setting the derivative of the exponent in Eq.(2.11) equal to 0, we get $L = \frac{n}{k}\mu'(s)$, and substituting this value of L in Eq.(2.11), Eq.(2.5) results, proving the theorem.

It is shown in ref.(4) that setting $L = \frac{n}{k}\mu'(s)$ actually minimizes the exponent, thereby yielding the best bound, but the theorem is true regardless of the minimal character of the exponent. Although not necessary here, it can be shown, using "tilted" probabilities [4] and a central limit theorem,[6] that asymptotically for large n,

$$N\left[\frac{n}{k}\mu'(s)\right] \longrightarrow \frac{2}{\sqrt{2\pi n\mu''(s)}} \exp \frac{n}{k}\left[\mu(s) - s\,'(s) + (k-1)\ln 2\right] \tag{2.12}$$

Theorem 2 can now be used to find the probability, P(L), 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(L) 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 $\frac{N(L)}{\binom{n}{L}}$ that the L weight sequence chosen will satisfy any particular block of $\frac{n}{k}$ parity checks. Since each of the j blocks of parity checks is chosen independently,

$$P(L) = \left[\frac{N(L)}{\binom{n}{L}}\right]^j \tag{2.13}$$

The minimum distance distribution function can now be derived in terms of P(L) in the same way as it was derived for the ensemble of all parity check codes in Eq.(2.2).

$$\Pr\left[D \le n\delta\right] \le \sum_{L=2}^{n\delta} \binom{n}{L} P(L) = \sum_{L=2}^{n\delta} \binom{n}{L}^{-j+1} \left[N(L)\right]^{j} \qquad (2.14)$$

Note that in the low density ensemble only sequences of even weight may be code words. Using Eqs(2.4) and (2.11), we get

$$\binom{n}{L} P(L) \le C(\lambda,n)\exp - nE(\lambda) \qquad (2.15)$$

where $\lambda = \frac{L}{n}$

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

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

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

Substituting Eq.(2.15) into Eq.(2.14), we get

$$\Pr\left[D \le n\delta\right] \le \sum_{L=2}^{n\delta} C(\lambda,n)\exp -nE(\lambda) \qquad (2.16)$$

As n increases, the summation in Eq.(2.16) is governed principally by the behavior of $E(\lambda)$. $E(\lambda)$ also appears in the bounds for probability of decoding error in the next chapter. Unfortunately, it is not easy to analyze $E(\lambda)$ 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 $j \to 3$, $E(\lambda)$ has the behavior shown in Fig.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 remaining negative for $\lambda > \delta_{jk}$.

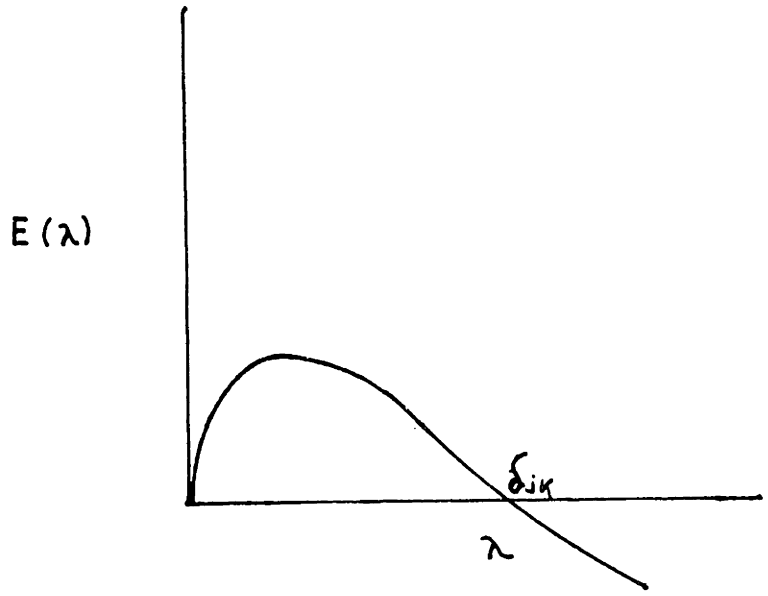

Figure 2-3

SKETCH OF THE FUNCTION $E(\lambda)$

It is clear that for any $\delta > \delta_{jk}$ the summation in Eq.(2.16) 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 A2

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

$$\Pr\left[D \le n\delta\right] \le \frac{k-1}{2n^{j-2}} + O\left(\frac{1}{n^{j-2}}\right) + nC(\delta n, n)\exp -nE(\delta) \quad (2.17)$$

and

$$\Pr\left[D \le n\delta\right] \le 1$$

where C and E are defined in Eq.(2.15) and $O\left(\frac{1}{n^{j-2}}\right)$ is a function approaching 0 with n faster than $\left(\frac{1}{n^{j-2}}\right)$.

The first term in Eq.(2.17) comes from code words of weight 2, the next term from words of small weights greater than 2, and the last term from words of large weight. As n gets larger this bound to the minimum distance distribution function tends toward a small step at $\delta = \frac{2}{n}$, and a large step at $\delta = \delta_{jk}$, with the amplitude of the small step decreasing as n^{-j+2}.

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 $n\delta_{jk}$, and since δ_{jk} is independent of block length, the minimum distance typical of most codes in the ensemble increases linearly with block length. Fig. 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 ensemble of all codes. It can be seen that as j and k increase, δ_{jk} for the (n,j,k) codes quickly approaches δ_o for the ensemble of all codes. This is proved in Theorem A3 of Appendix A.

Here one sees why a minimum distance distribution function was derived before obtaining any results about probability of decoding error. If two words in a group code differ only in two digits, then the probability of 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 $\left(\frac{1}{n^{j-2}}\right)$, the probability of codes of minimum distance 2. A very small fraction of poor codes, consequently, 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 will 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 a modified (n,j,k) ensemble and will be used in the next chapter to derive bounds on the probability of decoding error for (n,j,k) codes.

Let λ_{njk} be the minimum distance of the modified ensemble. λ_{njk} is lower bounded by that value of δ for

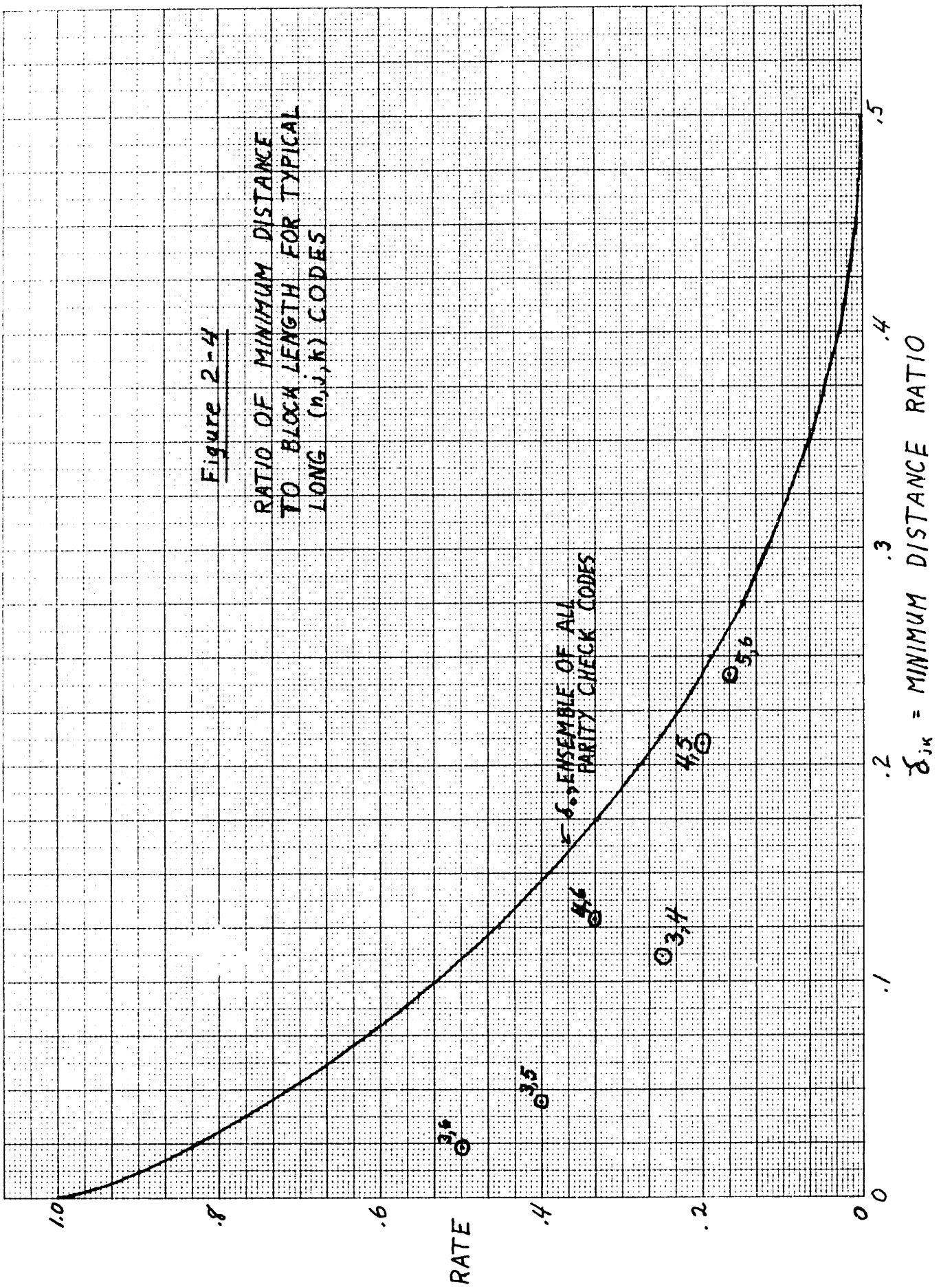

Figure 2-4

RATIO OF MINIMUM DISTANCE TO BLOCK LENGTH FOR TYPICAL LONG (n,j,k) CODES

which the right side of Eq.(2.17) is one half. With increasing n, the bound of Eq.(2.17) approaches a step function at δ_{jk}, so that λ_{njk} is asymptotically bounded by δ_{jk}.

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

Theorem 3:

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

$$D \leq 2 + \frac{2 \ln n/2}{\ln(k-1)} \qquad (2.18)$$

Proof:

The theorem will be proved by representing the code in the form of a tree as in Fig. 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 tree may be similarly constructed until, for some integer, m, a loop is formed at the m'th tier. Such a loop may occur either if two branches rising from the m'th tier contain a digit in common, as in Fig. 2-5, or if a single branch 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 tier of the tree contains 2(k-1) nodes, the second contains $2(k-1)^2$ nodes, and similarly the m'th 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,

$$2(k-1)^m \leq n$$

$$m \leq \frac{\ln n/2}{\ln(k-1)} \tag{2.19}$$

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 Fig. 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 the 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

$$D \leq 2m + 2 \qquad (2.20)$$

since the loop is formed by a single descent and ascent in the tree. Combining Eqs.(2.19) and (2.20), we get the statement of the theorem, Eq.(2.18).

Q. E. D.

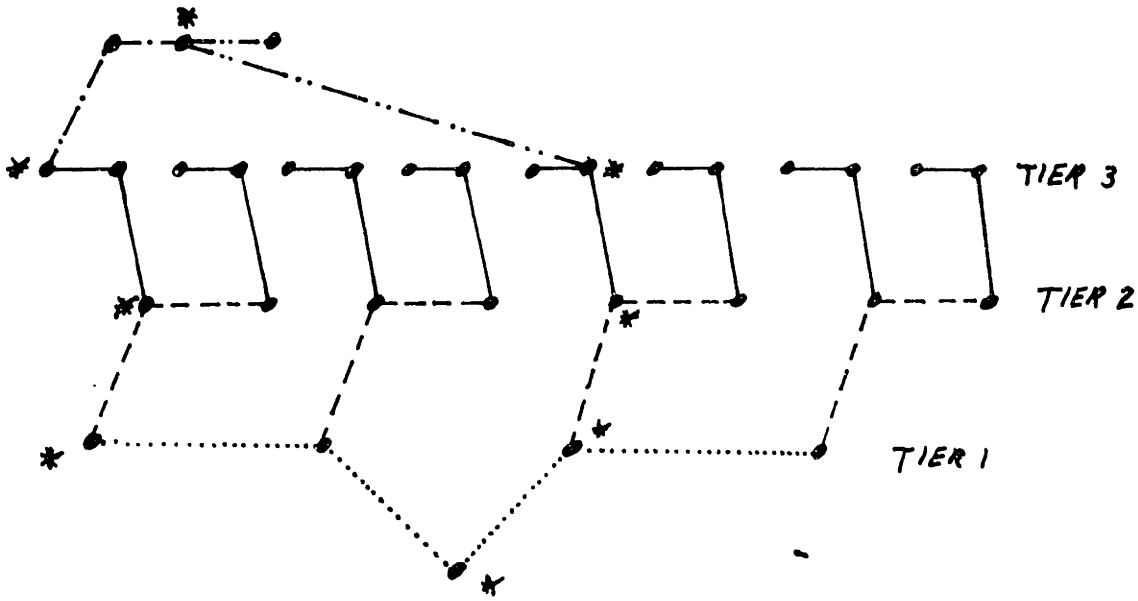

Figure 2-5

PARITY CHECK TREE

CHAPTER III

PROBABILITY OF DECODING ERROR

Upper Bound on Rate

The major portion of this chapter will analyze the effectiveness of typical codes in an (n,j,k) ensemble for correcting errors on a memoryless symmetric channel with a binary input alphabet and arbitrary output alphabet. It will be found that the probability of error behavior of the typical (n,j,k) code is similar to that of an ordinary random parity check code of somewhat higher rate. First, however, a fundamental, but not serious, limitation on any (n,j,k) code will be derived. This limitation will show that not even the best (n,j,k) code can be used to transmit information reliable over a channel whose capacity is too close to the code rate. The limitation will be proved only for the BSC, but it can be shown to be applicable to any channel. The limit on capacity suggests that even the best (n,j,k) code behaves similarly to the typical (n,j,k) code, and that a search for (n,j,k) codes that are much better than typical will be fruitless.

Theorem 3.1:

Let a parity check code on length n and rate R containing k digits in each parity check set be used on a BSC with cross-over 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,

$$R > \frac{H(p_k) - H(p)}{H(p_k)} \qquad (3.1)$$

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 $1 - \frac{H(p)}{\ln 2}$. Since $H(p_k) < \ln 2$, this theorem states that the source rate must be bounded away from the channel capacity for reliable transmission. Fig.(3-1) illustrates the amount by which the source rate must exceed the capacity 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

$$\frac{1}{n}\overline{I(u;v)} = -\frac{1}{n}\overline{\log_2 p(u)} + \frac{1}{n}\overline{\log_2 p_v(u)}$$

$$= -\frac{1}{n}\overline{\log_2 p(v)} + \frac{1}{n}\overline{\log_2 p_u(v)} \qquad (3.2)$$

If the per digit equivocation satisfies the equation

$$-\frac{1}{n}\overline{\log_2 p_v(u)} \geq \epsilon > 0 \qquad (3.3)$$

for some ϵ independent of n, then the probability of decoding error must also remain bounded away from 0. Eq.(3.3) will be established by evaluating the other terms in Eq.(3.2).

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

$$-\frac{1}{n}\overline{\log_2 p(u)} = R \qquad (3.4)$$

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

$$\frac{1}{n}\overline{\log_2 p_u(v)} = \frac{-H(p)}{\ln 2} \qquad (3.5)$$

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 v since specifying one will make it possible to compute the other. The probability that a parity check is satisfied is the probability that an even number of errors have occured within the parity check set, which is

$$\sum_{i \text{ even}} \binom{k}{i} p^i (1-p)^{k-i} = \frac{1+(1-2p)^k}{2} \tag{3.6}$$

To verify Eq.(3.6), rewrite the right side as

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

and expand in a binomial series.

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

$$-\frac{1}{n}\overline{\log_2 p(v)} \leq \frac{(1-R)H(p_k)}{\ln 2} + R \tag{3.7}$$

The substitution of Eqs.(3.4), (3.5), and (3.7) into Eq. (3.2) produces

$$-\frac{1}{n}\overline{\log_2 p_v(u)} \geq \frac{H(p)}{\ln 2} - \frac{(1-R)H(p_k)}{\ln 2} \tag{3.8}$$

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

$$R = \frac{H(p_k) - H(p) + \epsilon \ln 2}{H(p_k)} \tag{3.9}$$

Substituting Eq.(3.9) in Eq.(3.8), we obtain Eq.(3.3), proving the theorem.

Q. E. D.

Ensemble Probability of Error for Binary Symmetric Channel

A bound is derived in this section concerning the probability of decoding error when (n,j,k) codes are used on a binary symmetric channel with maximum likelihood decoding. Given a received sequence, v, a maximum likelihood decoder chooses the code word, u, that maximizes $p_u(v)$, the probability of v conditional on u. Since

$$p_v(u) = \frac{p_u(v)p(u)}{p(v)}$$

it is easily seen that if all code words are used with equal probability, then maximizing $p_u(v)$ is equivalent to maximizing $p_v(u)$, which minimizes the probability of decoding error.

It is next shown that for any parity check code, the probability of decoding error on the BSC using maximum likelihood decoding is independent of the transmitted code word. Let u_1 be the transmitted code word, let v be the received

sequence, and let e, the error sequence, be the modulo 2 sum of u_1 and v. If e is as close to some non-zero code word, u_2, as it is to the zero sequence, then v will be as close to $u_1 \oplus u_2$ as it is to u_1. Since the group property of a parity check code implies that $u_1 \oplus u_2$ is a code word, a decoding error will result. This result depends only on the error sequence, which is independent of the transmitted word. Thus the probability of error is independent of the transmitted word.

Each code in the modified (n,j,k) ensemble has its own probability of decoding error when used on a BSC, and we shall next bound the average value over the ensemble of all these probabilities. Given this average probability of decoding error, it should be easy to find a code for which the probability of error is not much greater than the average.

Since the probability of decoding error in any code is independent of the transmitted code word, the assumption that the all zero sequence is transmitted can be used in finding the ensemble average probability of decoding error. Rather than first finding the probability of error for each code and then averaging over the codes, we shall first average over the codes given a particular received sequence, and then average over the received sequences.

Assume a received sequence, v, that contains c ones and n-c zeros. The ensemble probability of decoding error for this sequence is less than or equal to the probability

of the set of codes that contain a code word differing from v in c or fewer places. Let M(L,c) be the number of sequences of weight L that differ from v in c or fewer places. An L weight sequence differs from v in c or fewer places if and only if the L weight sequence contains at least $\frac{L}{2}$ ones in positions where v contains ones. Since there are $\binom{c}{i}$ arrangements in which an L weight sequence has i ones in positions where v has ones, and since there are $\binom{n-c}{L-i}$ arrangements of the other L-i ones in positions where v has zeros,

$$M(L,c) = \sum_{i=L/2}^{L} \binom{c}{i} \binom{n-c}{L-i}$$

In the modified (n,j,k) ensemble as defined at the end of Chapter II, no sequence of weight $L < n\lambda_{njk}$ is a code word. For $L \geq n\lambda_{njk}$, the probability that an L-weight sequence is a code word is less than or equal to 2P(L), where P(L) is given in Eq.(2.13). Since the probability of a union of events is less than or equal to the sum of the probabilities of the individual events, the probability that a code word of weight L differs from v in c or fewer places is less than or equal to

$$\begin{cases} 2P(L)M(Lc) & (\text{for } L \geq n\lambda_{njk}) \\ 0 & (\text{for } L < n\lambda_{njk}). \end{cases}$$

Let $P_c(e)$ be the ensemble probability of decoding error given a received sequence of c ones. It is the probability that a code word of any weight differs from the received sequence in c or fewer places.

$$P_c(e) \leq \sum_{L=n\lambda_{njk}}^{2c} 2P(L)M(Lc) = \sum_{L=n\lambda_{njk}}^{2c} 2P(L) \sum_{i=\frac{L}{2}}^{L} \binom{c}{i}\binom{n-c}{L-i} \quad (3.10)$$

Eq.(3.10) can be simplified by noting that for $c < \frac{n}{2}$ the maximum term of the summation over i is at $i = \frac{L}{2}$.* Bounding the other terms by the terms of a geometric series we obtain

$$\sum_{i=L/2}^{L} \binom{L}{i}\binom{n-c}{L-i} \leq \frac{2n-2c-L}{2n-4c}\binom{c}{L/2}\binom{n-c}{L/2}$$

$$P_c(e) \leq \sum_{L=n\lambda_{njk}}^{2c} \frac{2n-2c-L}{n-2c}\binom{c}{L/2}\binom{n-c}{L/2}P(L) \quad (3.11)$$

The bound in Eq.(3.11) increases with the number of errors, c, until it becomes greater than 1. It will now be shown that the number of errors for which the bound equals 1 is asymptotically a fixed fraction, γ_{jk}, of the block length.

* Only even values of L are considered since P(L)=0 for L odd.

It follows from Eq.(2.4) that

$$\binom{c}{L/2}\binom{n-c}{L/2} \leq \exp\left[cH\left(\frac{L}{2c}\right) + (n-c)H\left(\frac{L}{2(n-c)}\right)\right] \qquad (3.12)$$

Applying Eqs.(3.12), (2.11), (2.13), and (2.4) to Eq.(3.11), we have

$$P_c(e) \leq \sum_{L=n\lambda_{njk}}^{2c} A(\lambda,\gamma)\exp\, nB(\lambda,\gamma) \qquad (3.13)$$

where

$$\lambda = \frac{L}{n} \quad ; \quad \gamma = \frac{c}{n}$$

and

$$A(\lambda,\gamma) = \frac{2-2\gamma-\lambda}{1-2\gamma}\left[2\pi n\lambda(1-\lambda)\right]^{\frac{1}{2}} \exp\frac{1}{12n\lambda(1-\lambda)}$$

$$B(\lambda,\gamma) = \gamma H\frac{\lambda}{2\gamma} + (1-\gamma)H\left(\frac{\lambda}{2(1-\gamma)}\right) - jH(\lambda) +$$

$$+ \frac{1}{k}\left[\mu(s) + (k-1)\ln 2\right] - js\lambda$$

The parameter, s, is implicitly a function of λ through the relation $\mu'(s) = \lambda k$ where $\mu(s) = \ln 2^{-k}\left[(1+e^s)^k + (1-e^s)^k\right]$. If, for a particular value of $\gamma = \frac{c}{n}$, $B(\lambda,\gamma)$ remains negative

for all λ then $P_c(e)$ must approach 0 with increasing n, but if $B(\lambda,\gamma)$ is positive for some λ, then the first bound in Eq.(3.12) increases without limit as n increases. Thus γ_{jk} is that γ for which $B(\lambda,\gamma)$, maximized over λ, is equal to 0. γ_{jk} is called the error correcting breakpoint of the code since the bound to $P_c(e)$ as a function of γ approaches a unit step at $\gamma = \gamma_{jk}$ as $n \to \infty$. Fig. 3-1 plots γ_{jk} for several different values of j and k. For comparison, the continuous line in Fig. 3-1 is the error correcting breakpoint for the ensemble of all codes.

In order to evaluate the overall probability of decoding error, $P_c(e)$ will be bounded by Eq.(3.11) for $c < n\gamma_{jk}$ and by 1 for $c > n\gamma_{jk}$. For a binary symmetric channel with a cross-over probability p, the probability of c cross-overs is $\binom{n}{c} p^c(1-p)^{n-c}$. Averaging $P_c(e)$ over c, we obtain the ensemble average probability of decoding error,

$$P(e) \leq P_1 + P_2 \tag{3.14}$$

where

$$P_1 = \sum_{c=b}^{n} \binom{n}{c} p^c (1-p)^{n-c}$$

$$P_2 = \sum_{c=\frac{n\lambda_{njk}}{2}}^{b-1} \sum_{L=n\lambda_{njk}}^{2c} \frac{2n-2c-L}{n-2c} \binom{c}{L/2} \binom{n-c}{L/2} P(L) \binom{n}{c} p^c (1-p)^{n-c}$$

in which b is the smallest integer greater than or equal to

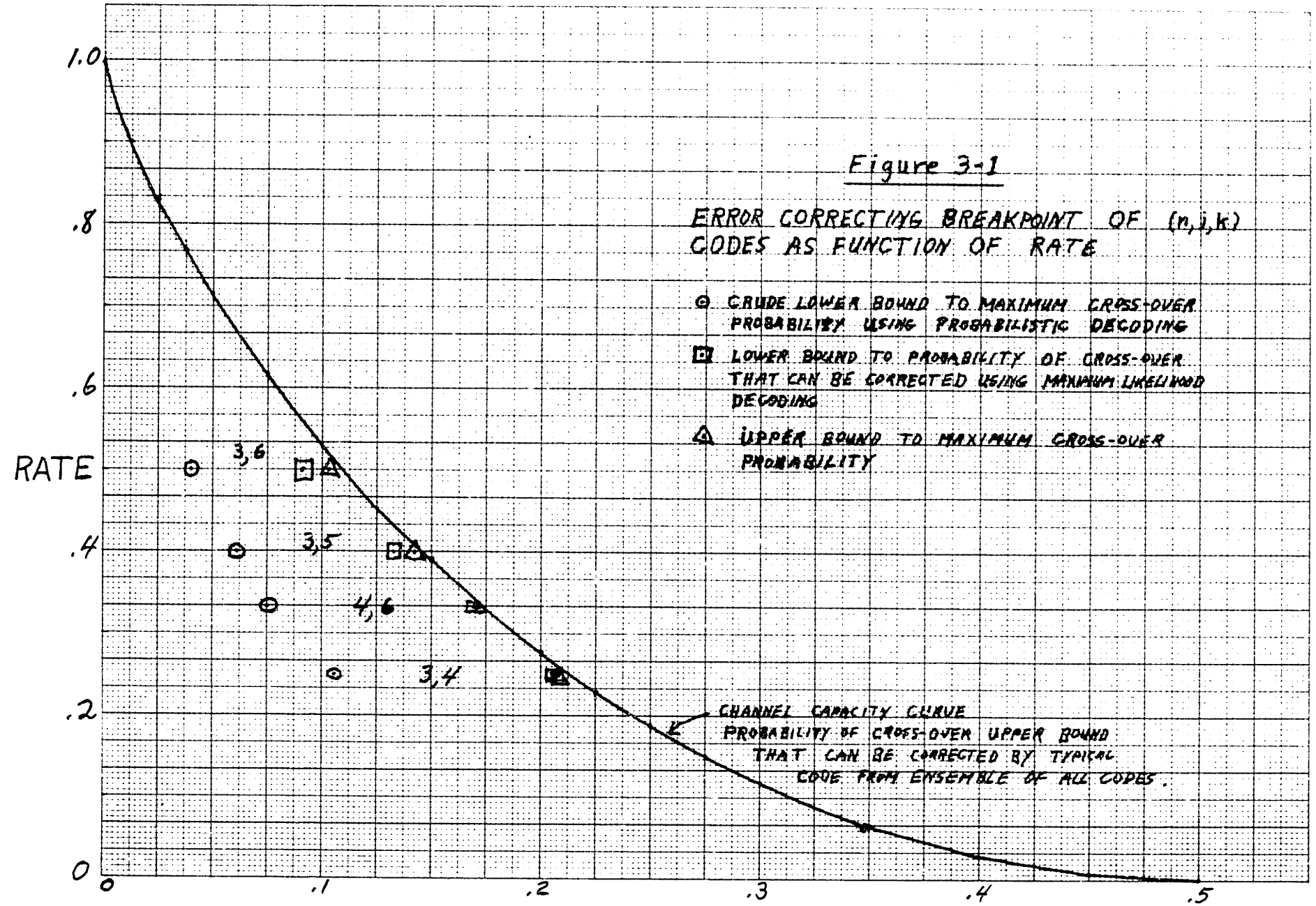

Figure 3-1

ERROR CORRECTING BREAKPOINT OF (n,j,k) CODES AS FUNCTION OF RATE

⊙ CRUDE LOWER BOUND TO MAXIMUM CROSS-OVER PROBABILITY USING PROBABILISTIC DECODING

⊡ LOWER BOUND TO PROBABILITY OF CROSS-OVER THAT CAN BE CORRECTED USING MAXIMUM LIKELIHOOD DECODING

△ UPPER BOUND TO MAXIMUM CROSS-OVER PROBABILITY

γ_{jk} = ERROR CORRECTING BREAKPOINT

$\gamma_{jk}n$.

Next it is shown that for any channel probability $p < \gamma_{jk}$, P(e) approaches 0 exponentially with increasing n. Thus the error correcting breakpoint, γ_{jk}, is also the maximum channel cross-over probability for which reliable decoding can be assured using an (n,j,k) code.

P_1, in Eq.(3.14), can be bounded by a geometric series for $p < \gamma_{jk}$, obtaining

$$P_1 \leq \frac{(1-p)b}{b-pn} \binom{n}{b} p^b (1-p)^{n-b}$$

Applying Eq.(2.4) to $\binom{n}{b}$, and recalling that $b \geq n\gamma_{jk}$,

$$P_1 \leq \frac{(1-p)\gamma_{jk}}{(\gamma_{jk}-p)\sqrt{2\pi n\gamma_{jk}(1-\gamma_{jk})}} \exp -n\left[T_p(\gamma_{jk}) - H(\gamma_{jk})\right] \tag{3.15}$$

where $T_p(\gamma_{jk}) = -\gamma_{jk}\ln p - (1-\gamma_{jk})\ln (1-p)$ is the tangent at p to the binary entropy curve as shown in Fig. 3-2.

It is next demonstrated that P_2 in Eq.(3.14) is bounded in terms of the same exponent as that given for P_1 in Eq.(3.15) if p is in the range

$$\frac{\gamma_{jk} - \frac{\lambda_{jkn}}{2}}{1-\gamma_{jk} - \frac{\lambda_{jkn}}{2}} < \frac{p}{1-p} < \frac{\gamma_{jk}}{1-\gamma_{jk}} \tag{3.16}$$

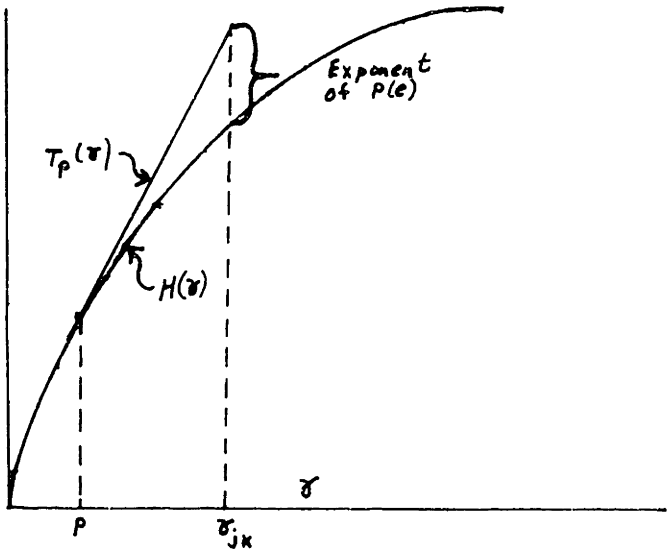

Figure 3-2

GEOMETRIC INTERPRETATION OF P(e) EXPONENT

If Eq.(3.16) is satisfied and if the order of summation in P_2 of Eq.(3.14) is interchanged, then for every value of L in the sum, the sum over c can be bounded by a geometric series with a maximum term at c = b-1.

$$P_2 \leq \sum_{L=n_{njk}}^{2(b-1)} \frac{2^{n-2b+2-L}}{1-\left(\frac{\gamma_{jk}-\lambda/2}{1-\gamma_{jk}-\lambda/2}\right)\left(\frac{1-\rho}{\rho}\right)} \binom{b-1}{L/2}\binom{n-b+1}{L/2} P(L)\binom{n}{b-1}\rho^{b-1}(1-\rho)^{n-b+1}$$

Using Eq.(3.13) and recalling that $B(\lambda,\gamma) \leq 0$ for $\gamma \leq \gamma_{jk}$, we have

$$P_2 \leq \sum_{L=n\lambda_{njk}}^{2(b-1)} \frac{p(1-\gamma_{jk}-\lambda/2}{p(1-\lambda)-\gamma_{jk}+\lambda/2} \; A(\lambda, \gamma_{jk}) \binom{n}{b-1} p^{b-1}(1-p)^{n-b+1} \tag{3.17}$$

Combining Eq.(3.15) with Eq.(3.17),

$$P(e) \leq D \exp -n\left[T_p(\gamma_{jk}) - H(\gamma_{jk})\right] \tag{3.18}$$

$$D = \frac{1}{\sqrt{2\pi n \gamma_{jk}(1-\gamma_{jk})}}\left[\frac{(1-p)}{(\gamma_{jk}-p)} + \frac{A(2\gamma_{jk},\gamma_{jk}) \quad (1-p)\gamma_{jk}(1-\gamma_{jk}-\frac{\lambda_{njk}}{2})}{(1-\gamma_{jk})(p-\gamma_{jk}-p\lambda_{njk} + \lambda_{\frac{njk}{2}})}\right]$$

for

$$\frac{\gamma_{jk} - \frac{\lambda_{jkn}}{2}}{1-\gamma_{jk} - \frac{\lambda_{jkn}}{2}} < \frac{p}{1-p} < \frac{\gamma_{jk}}{1-\gamma_{jk}}$$

The exponent in Eq.(3.18) is the same as the exponent in the probability of decoding error over the ensemble of all parity check codes at a rate $1 - \frac{H(\gamma_{jk})}{\ln 2}$ bits per

symbol[3]. It is also the exponent of the Hamming bound,[7] which is a lower bound to the probability of decoding error for any code of rate $1 - \frac{1}{\ell n 2} H(\gamma_{jk})$. Thus, for the range of channel cross-over probabilities given by Eq.(3.16), the low density code ensemble has the same exponent in the probability of error as the ensemble of all codes at a slightly higher rate. Fig. 3-1 illustrates the magnitude of this rate loss for several values of j and k.

For the rest of this section, P(e) will be evaluated when the channel cross-over probability is too small to satisfy Eq.(3.16). Conceptually the problem is simple. Eq.(3.15) is a valid bound to P_1, and P_2 can be bounded by finding the maximum term in the double summation of Eq.(3.13) and multiplying by the number of terms. The solution to this problem is involved, however, because the maximum term can appear either on a boundary of the summation or within the summation. The reader who is content with the statement that the exponent governing P(e) increases as the channel cross-over probability decreases will lose nothing by omitting the rest of this section.

First we substitute the equality

$$\binom{c}{L/2}\binom{n-c}{L/2}\binom{n}{c} = \binom{L}{L/2}\binom{n-L}{c-L/2}\binom{n}{L}$$

into Eq.(3.14), and apply Eqs.(2.4) and (2.15) to obtain

$$P_2 \leq \sum_{c=\lambda_{jkn}/2}^{b-1} \sum_{L=n\lambda_{jkn}}^{2c} \frac{n-c-L/2}{n-2c} C(\lambda,n) \exp -nF(\lambda,\gamma) \quad (3.19)$$

$$F(\lambda,\gamma) = -\lambda \ln 2 - (1-\lambda)H\left(\frac{\gamma-\lambda/2}{1-\lambda}\right) + E(\lambda) - \gamma \ln p - (1-\gamma)\ln(1-p) \quad (3.20)$$

where $\lambda = L/n$, $\gamma = \frac{c}{n}$, and $C(\lambda,n)$ and $E(\lambda)$ are defined in Eq.(2.15). To find the minimum value of $F(\lambda,\gamma)$ we can re-write Eq.(3.20) in the form

$$F(\lambda,\gamma) = \frac{-\lambda}{2} \ln 4p(1-p) + E(\lambda)$$

$$- (1-\lambda)\left[-H\left(\frac{\gamma-\lambda/2}{1-\lambda}\right) - \left(\frac{\gamma-\lambda/2}{1-\lambda}\right)\ln p - \frac{1-\gamma\lambda/2}{1-\lambda} \ln(1-p)\right]$$

is now contained entirely within the square brackets. The minimum value of this square bracket is 0 and occurs at

$$\frac{\gamma-\lambda/2}{1-\lambda} = p \quad (3.21)$$

Thus

$$F(\lambda,\gamma)_{\gamma \min} = \frac{-\lambda}{2} \ln 4p(1-p) + E(\lambda) \quad (3.22)$$

$$\frac{dF(\lambda,\gamma)_{\gamma\,min}}{d\lambda} = -\frac{1}{2}\ln 4p(1-p) + E'(\lambda) \qquad (3.23)$$

The behavior of $E(\lambda)$ is discussed in Appendix A, and $E'(\lambda)$ is sketched in Fig. 3-3.

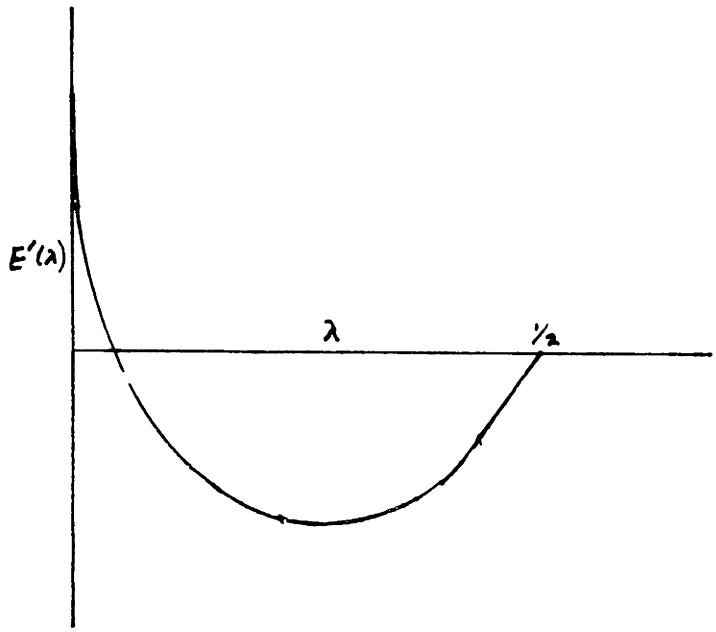

Figure 3-3

SKETCH OF $E'(\lambda) = \frac{dE(\lambda)}{d\lambda}$

From Fig. 3-3 and Eq.(3.23), it can be seen that for p very small, $\frac{dF(\lambda,\gamma)_{\gamma_{min}}}{d\lambda}$ is always positive, and the minimum value of F in the double summation of Eq.(3.19) occurs at $L = n\lambda_{njk}$. As p gets larger, $\frac{dF(\lambda,\gamma)_{\gamma\,min}}{d\lambda}$ eventually becomes negative, and has 2 zeros, the second of which represents a local minimum of $F(\lambda,\gamma)$. Both the $\lambda = \lambda_{min}$ at

which the minimum occurs and the γ satisfying Eq.(3.21) increase with p. When this γ reaches γ_{jk}, then the minimum term is once again at γ_{jk} and the exponent in Eq.(3.18) holds. Summarizing, the exponent of the probability of error is the exponent of Eq.(3.18) for $p \geq \frac{\gamma_{jk} - \lambda_{min}/2}{1-\lambda_{min}}$. For $p < \frac{\gamma_{jk} - \lambda_{min}/2}{1-\lambda_{min}}$, the exponent is the minimum of $-\frac{\lambda_{jkn}}{2} \ln[4\, p(1-p)]$ and $\frac{\lambda_{min}}{2} \ln[4p(1-p)] + E(\lambda_{min})$.

Binary Symmetric Input Multi-Output Channels

Binary symmetric channels are sometimes approximated in practice by communication systems in which binary signals are sent over essentially memoryless physical channels with a receiver at the output that chooses the most likely signal. Such a receiver throws away information by making a decision instead of retaining the a posteriori probabilities. Fortunately, the decoding scheme to be described in the next chapter operates directly from the a posteriori probabilities as easily as from a binary decision. It would be folly to use a decision-making receiver with such a decoding scheme. Thus, in this section, a bound will be formulated for the average probability of decoding error over an (n,j,k) ensemble of codes used on a channel with an arbitrary set of outputs. To make the problem tractable, however, only those channels will be considered for which the noise, in a sense to be defined later, acts symmetrically on the two possible inputs.

Consider a channel with an input alphabet of two symbols, $x = 0$ and $x = 1$; an output alphabet (discrete or continuous) of symbols y; and a set of channel transition probabilities $P_x(y)$. Define

$$I_o(y) = \ln \frac{2P_o(y)}{P_o(y)+P_1(y)} ; \quad I_1(y) = \ln \frac{2P_1(y)}{P_o(y)+P_1(y)} \tag{3.24}$$

For equiprobable inputs, these are the mutual informations associated with inputs of 0 and 1 respectively. Let $F_o(I_o, I_1)$ be the distribution function of $I_o(y)$ and $I_1(y)$ when $x = 0$ is sent, and let $F_1(I_o, I_1)$ be the distribution function of $I_1(y)$ and $I_o(y)$ when $x = 1$ is sent. Define a symmetric input channel as one satisfying

$$F_o(I_o, I_1) = F_1(I_1, I_o) \tag{3.25}$$

Thus, $F_o(I_o, I_1)$ differs from $F_1(I_o, I_1)$ only by an interchange of I_o and I_1.

If the input signals are properly chosen, both channels with independent memoryless additive noise and fading channels with additive noise in which both the fade and noise are independent and memoryless are examples of binary symmetric input channels.

As in the last section, assume a maximum likelihood decoder. If $v = y^1 y^2 \ldots y^n$ is received, the decoder

chooses the code word, $u = x^1 x^2 \ldots x^n$ that maximizes $P_u(v) = \prod_{i=1}^{n} P_{x^i}(y^i)$.

Define

$$I_u(v) = \sum_{i=1}^{n} I_{x^i}(y^i) = \sum_{i=1}^{n} \ln \frac{2P_{x^i}(y^i)}{P_o(y^i)+P_1(y^i)}$$

$$= \ln P_u(v) - \sum_{i=1}^{n} \ln \tfrac{1}{2} P_o(y^i) + P_1(y^i)$$

The u for which $P_u(v)$ is maximized is also the u for which $I_u(v)$ is maximized. Since the specification of $I_o(y)$ and $I_1(y)$ for each digit will allow the decoder to maximize $I_u(v)$, the probability of decoding error for a code can be determined from the distribution function $F_o(I_o, I_1)$ and $F_1(I_o, I_1)$ along with a knowledge of the code words. With this we can prove the following theorem:

Theorem 3.2:

For any parity check code and any binary symmetric input channel (ie., satisfying Eq.(3.25)) the probability of decoding error using a maximum likelihood decoder is independent of the transmitted code word.

Proof:

It is sufficient to show that an arbitrary code word has the same probability of decoding error as the all zero

sequence. Consider modulo 2 adding this arbitrary code word to each code word in the code. From the group properties of a parity check code,[13] this addition reproduces the same set of code words, but the arbitrary code word has become the all zero sequence. Alternatively, this may be viewed as redefining all ones to be zeros and vice versa in places in which the arbitrary code word had ones. Along with this redefinition, the distribution function of I_o and I_1 must also be changed, so that $F_o(I_o, I_1) \rightarrow F_1(I_1, I_o)$ and $F_1(I_1, I_o) \rightarrow F_o(I_o, I_1)$. But from Eq.(3.25), this change is trivial. Thus the arbitrary code word is isomorphic to the all zero code word, both with respect to the other code words and with respect to the distribution function of $I_x(y)$. But the probability of error is determined by these same quantities, proving the theorem. Q. E. D.

The average probability of decoding error can now be bounded for an (n,j,k) ensemble by bounding the probability of error when the all zero sequence is transmitted. Let

$$I_o(v) = \sum_{i=1}^{n} I_o(y^i).$$

$I_o(v)$ is the sum of n independent random variables each with the probability distribution $F_o(I_o, I_1)$. The probability of error can be bounded by

$$P(e) \leq \underbrace{\Pr\left[I_o(v) \leq nI_c\right]}_{P_1} + \underbrace{\Pr\left[I_o(v) \geq nI_c;\ \text{decoding error occurs}\right]}_{P_2} \tag{3.26}$$

where I_c is an arbitrary parameter to be determined later. First, we shall determine a bound for P_2.

Consider a code word, u, containing ones in the first L positions and zeros elsewhere. An incorrect decoding will result if $I_u(v) - I_o(v) \geq 0$. Ambiguities are regarded as errors throughout. But since **u** and the zero sequence differ only in the first L places,

$$I_u(v) - I_o(v) = \sum_{i=1}^{L} I_1(y^i) - I_o(y^i)$$

$$\Pr\left[I_o(v) \geq nI_c;\ \text{decoding error occurs to } u\right]$$

$$\leq \Pr\left[I_o(v) \geq nI_c;\ \sum_{i=1}^{L} I_1(y^i) - I_o(y^i) \geq 0\right] \tag{3.27}$$

Since the channel makes no distinction between the first L digits and any other L digits, <u>every</u> code word of weight L must satisfy Eq.(3.27). Over the modified (n,j,k) ensemble no sequence of weight $L < n\lambda_{njk}$ is a code word, and for $L \geq n\lambda_{njk}$, the probability that an L-weight sequence is a code word is less than or equal to 2P(L), with P(L) given in Eq.(2.13).

Since there are $\binom{n}{L}$ different sequences of weight L and since the probability of a union of events is less than or equal to the sum of the probabilities of the individual events,

$$P_2 \leq \sum_{L=n\lambda_{njk}}^{n} \binom{n}{L} 2P(L) \Pr\left[I_o(v) \geq nI_c ; \sum_{i=1}^{L} I_1(y^i) - I_o(y^i) \geq 0 \right] \tag{3.28}$$

The last factor in Eq.(3.28) will be evaluated by a double Chernov bound developed by Shannon[12] and Fano[4]. The only addition by the author is the summing of the two random variables over unequal numbers of terms.

Theorem 3.3:

Let $F(x,y)$ be the distribution function of the random variables x and y, and assume that $\mu(t,r) = \ln g(t,r)$ where the moment generating function,

$$g(t,r) = \int \exp(tx+ry)\, dF(x,y)$$

exists for some region of $t>0$, $r>0$. Let x_i and y_i represent the i'th sample of n independent samples of these variables, and let

$$X = \sum_{i=1}^{n} x_i \; ; \qquad Y = \sum_{i=1}^{L} y_i \qquad (L \leq n)$$

Then

$$\Pr\left[X \geq a;\ Y \geq b\right] \leq \exp\left[L\mu(t,r) + (n-L)\mu(t,0) - ta - rb\right]$$

(for any $t > 0$, $r > 0$ for which $g(t,r)$ exists) (3.29)

Proof:

The joint moment generating function of X, Y is

$$\overline{\exp\ tX+rY} = \overline{\exp\left[\sum_{i=1}^{L}(x_i t + y_i r) + \sum_{i=L+1}^{n} x_i t\right]} \tag{3.30}$$

This is the average of the product of n independent random variables, which is equal to the product of the averages.

$$\overline{\exp(tX+rY)} = \prod_{i=1}^{L} \overline{\exp(x_i t + y_i r)} \prod_{i=L+1}^{n} \overline{\exp\ x_i t}$$

$$= \left[g(t,r)\right]^{L}\left[g(t,0)\right]^{n-L} \tag{3.31}$$

where $g(t,r) = \overline{\exp(tx+ry)} = \int \exp(tx+ry)\, dF(x,y)$

Next some bounding operations are performed on the left side of Eq.(3.30). Let $F_n(X,Y)$ be the joint distribution function of X, Y.

$$\overline{\exp(tX+rY)} = \int_X \int_Y \exp(tX+rY)dF_n(X,Y) \geq \int_{X=a}^{\infty} \int_{Y=b}^{\infty} \exp(tX+rY)dF_n(X,Y)$$

For $t>0$, $r>0$, the minimum value of the integrand is at $X = a$, $Y = b$.

$$\overline{\exp(tX+rY)} \geq \exp(ta+rb) \int_{X=a}^{\infty} \int_{Y=b}^{\infty} dF_n(X,Y)$$

$$\Pr(X \geq a;\ Y \geq b) \leq \overline{\exp(tX+rY)}\ \exp(-ta-rb) \tag{3.32}$$

Substituting Eq.(3.31) into Eq.(3.32) and using $\mu(t,r) = \ln (g(t,r))$, the statement of the theorem is obtained.

Q. E. D.

Another convenient statement of the theorem is found by minimizing the exponent in Eq.(3.29), and expressing a and b in terms of the parameters t and r.

$$\Pr\left[X > L\mu_t'(t,r) + (n-L)\mu_t'(t,0);\ Y > L\mu_r'(t,r)\right] \le \exp\left[L\mu(t,r) + \right.$$

$$\left. +(n-L)\mu(t,0) - tL\mu_t'(t,r) - t(n-L)\mu_t'(t,0) - rL\mu_r'(t,r)\right]$$

where $\mu_t'(t,r)$ is $\frac{\partial \mu(t,r)}{\partial t}$ and $\mu_r'(t,r) = \frac{\partial \mu(t,r)}{\partial r}$.

For a proof that this is actually a minimum, and for a more complete discussion of the Chernov bound, see reference (4). Eq.(3.29), however, is more convenient here. Using I_o for x, $I_1 - I_o$ for y, nI_c for a, and 0 for b,

$$\Pr\left[I_o(v) \ge nI_c; \sum_{=1}^{L} I_1(y) - I_o(y) \ge 0\right]$$

$$\le \exp\left[L\mu(t,r) + (n-L)\mu(t,o) - tnI_c\right] \qquad (3.33)$$

where $\mu(t,r) = \ln \int\int \exp\left[tI_o + r(I_1-I_o)\right] dF_o(I_o,I_1)$ (3.34)

Substituting Eqs.(3.33) and (2.15) into Eq.(3.28),

$$P_2 \le \sum_{L=n\lambda_{njk}}^{n} 2C(\lambda,n)\exp + n\left[-E(\lambda) + \lambda\mu(t,r) + (1-\lambda)\mu(t,o) - tI_c\right]$$

(3.35)

The ubiquitous function $E(\lambda)$ is defined in Eq.(2.15) and is discussed in detail in Appendix A.

Eq.(3.26) bounded P(e) in terms of the two quantities P_1 and P_2. Eq.(3.35) bounds P_2 and we shall now use Eq.(3.29) to bound P_1. Let L = 0, and substitute $-I_o$ for x, $-nI_c$ for a, and -w for t. Since the moment generating function of $-I_o$ is $g(-t,0)$,

$$P_1 = \Pr\left[-I_o(v) \geq -nI_c\right] \leq \exp\left[n\mu(w,o) - wnI_c\right] \quad \text{(for any } w < 0) \tag{3.36}$$

When this bound is optimized over w, the usual Chernov bound for the sum of n independent random variables in terms of the parameter w results;[4],[11]

$$P_1 = \Pr\left[+I_o(v) \leq n\mu'_w(w,o)\right] \leq \exp n\left[\mu(w,o) - w\mu'_w(w,o)\right] \tag{3.37}$$

$$\text{(where } \mu'_w(w,o) = I_c)$$

Eqs.(3.35) and (3.37) bound the probability of decoding error for a binary symmetric input multi-output channel in terms of the parameters t,r,w, and λ. The parameter s also appears in the definition of $E(\lambda)$. t,r,w, and s must be chosen in such a way as to minimize the bound for P(e); this will lead to a set of simultaneous transcendental equations.

Only one simplification appears to be possible, and that is to express r simply in terms of t. For given values

of t, λ, and I_c, the exponent in Eq.(3.35) is minimized by setting $\mu_r'(t,r) = 0$.[4] From Eq.(3.34), we have

$$\mu_r'(t,r) = \frac{\iint (I_1-I_0)\exp\left[tI_0+r(I_1-I_0)\right]dF_0(I_0,I_1)}{\iint \exp\left[tI_0+r(I_1-I_0)\right]dF_0(I_0,I_1)} \tag{3.38}$$

We now use the channel symmetry to find an r for which the numerator in Eq.(3.38) is 0. Applying the symmetry condition, $F_0(I_0,I_1) = F_1(I_1,I_0)$, to the numerator of Eq.(3.38), and then interchanging the variables I_0 and I_1, the numerator of Eq.(3.38) is

$$\iint (I_1-I_0)\exp\left[tI_0 + r(I_1-I_0)\right]dF_0(I_0,I_1)$$
$$= \iint (I_0-I_1)\exp\left[tI_1 + r(I_0-I_1)\right]dF_1(I_0,I_1)$$

Using Eq.(3.24) to express I_0 and I_1 in terms of $P_0(y)$ and $P_1(y)$, and noting that $dF_0(I_0,I_1') = P_0(y)dy$ and $dF_1(I_0,I_1) = P_1(y)dy$, the numerator of Eq.(3.38) is

$$\int_y \ln\left[\frac{P_1(y)}{P_0(y)}\right] \frac{P_0(y)^{t-r+1}\,P_1(y)^r}{\left[\frac{1}{2}P_0(y) + \frac{1}{2}P_1(y)\right]^t}\,dy$$
$$= \int_y \ln\left[\frac{P_0(y)}{P_1(y)}\right] \frac{P_1(y)^{t-r+1}\,P_0(y)^r}{\left[\frac{1}{2}P_0(y) + \frac{1}{2}P_1(y)\right]^t}\,dy \tag{3.39}$$

If $t - r + 1 = r$, the right side of Eq.(3.39) is the negative of the left side, so that $r = (t+1)/2$ must make $\mu'_r(t,r) = 0$.

Incorporating this result in Eq.(3.35), removing the maximum value of $C(\lambda,n)$ from the summation, and bounding by n times the maximum term, we obtain

$$P_2 \leq 2nC_{max}(\lambda,n) \exp n\left[-E(\lambda) + \lambda\mu\left(t, \frac{t+1}{2}\right) + (1-\lambda)\mu(t,o) - tI_c\right] \tag{3.40}$$

evaluated at the $\lambda \geq \lambda_{jk}$ that maximizes the exponent.

Since the exponent in Eq.(3.40) is differentiable for $\lambda > \lambda_{jk}$, the maximum value for $\lambda \geq \lambda_{jk}$ must occur either for $\lambda = \lambda_{jk}$ or for a λ satisfying

$$E'(\lambda) = \mu\left(t, \frac{t+1}{2}\right) - \mu(t,0). \tag{3.41}$$

From Fig. 3-3, it can be seen that Eq.(3.41) has either two solutions or no solution. If there is no solution, the maximum value of Eq.(3.40) is at $\lambda = \lambda_{jk}$. If there are two solutions, the larger represents a local maximum of Eq.(3.40), and either this λ or λ_{jk} maximizes Eq.(3.40). Finally the minimum exponent can be obtained in the overall probability of error by setting the exponents of P_1 and P_2 equal and minimizing over t and w.

$$P(e) \leq \left[1 + 2nC_{max}(\lambda,n)\right] \exp n\left[\mu(w,o) - w\mu'_w(w,0)\right] \tag{3.42}$$

subject to the four conditions:

1) $\mu(w,0) - w\mu'_w(w,0) = -E(\lambda) + \lambda\mu(t,\frac{t+1}{2}) + (1-\lambda)\mu(t,0) - tI_c$;

2) $\mu'_w(w,0) = I_c = \lambda\mu'_t(t,\frac{t+1}{2}) + (1-\lambda)\mu'_t(t,0)$;

3) λ chosen to maximize Eq.(3.40);

4) $w \leq 0$; $t \geq 0$.

Condition 1, above, equalizes the exponent for P_1 and P_2; condition 2 minimizes over t and w; condition 3 maximizes over λ ; and condition 4 is necessary for the validity of the Chernov bounds.

Postponing the problem of whether condition 4 is consistent with the first three conditions, we first consider conceptually how to solve for t, w, and λ from the first three conditions. For any value of t, we first find the larger value of λ satisfying Eq.(3.41). Using first this value of λ and then λ_{jk}, we can find two values of I_c from condition 2, and then find which of them maximizes the right side of condition 1. This procedure given us λ, I_c, and the right side of condition 1 as functions of t. Using the fact that the second partial derivative of μ with respect to t is positive,(4) it is easy to show that I_c is increasing with t, and the right side of condition 1 is decreasing with t. Finally, condition 2 can be used to solve for w, from which

the left side of condition 1 can be solved. Both these quantities are seen to be increasing functions of t. Since the left side of condition 1 is increasing and the right side decreasing with t, it is a simple matter to increase t until the two sides are equal. Two difficulties can occur, however. First, w could become positive before equality is attained. This simply means that the channel capacity is too low for the code to be effective. The second difficulty can occur if the left side of condition 1 is greater than the right side at t = 0. If this condition holds, it can be shown that the probability of decoding error is bounded by

$$P(e) \leq 2nC_{max}(\lambda,n) \exp n\left[-E(\lambda) + \lambda\mu(0,\tfrac{1}{2})\right] \qquad (3.43)$$

evaluated at the $\lambda = \lambda_{jk}$ that maximizes the exponent. To see that Eq.(3.43) is actually a bound to the probability of decoding error, return to Eq.(3.27) and omit the cutoff level on $I_o(v)$. This gives

$$P(e) \sum_{L=n\lambda_{njk}}^{n} \binom{n}{L} 2P(L) \Pr\left[\sum_{i=1}^{L} I_1(y^i) - I_o(y^i) \geq 0\right]$$

Using this and repeating the argument from Eq.(3.27) to Eq.(3.42), the result is Eq.(3.43). The use of Eq.(3.43) for P(e) instead of Eq.(3.42) is analogous to the well known change of exponent for random coding on the BSC at rates below critical.

It is encouraging to note that this bound for probability of decoding error gives the same exponent derived in the first part of this chapter for the special case of the BSC. The proof of this is straightforward, but quite tedious. Nevertheless, the parameters t and w will be related to the parameter γ_{jk} that was used in the first derivation. The relations are

$$\frac{p^{w+1}}{p^{w+1} + q^{w+1}} = \gamma_{jk}$$

$$\frac{p^{t+1}}{p^{t+1} + q^{t+1}} = \frac{\gamma_{jk} - \lambda/2}{1 - \lambda}$$

in which λ is the solution of condition 3. The exponent goes to 0 when w = 0 which corresponds to $p = \gamma_{jk}$. The exponent changes, as discussed in the first part of the chapter, when t = 0 which corresponds to $p = \frac{\gamma_{jk} - \lambda/2}{1 - \lambda}$.

CHAPTER IV

DECODING

Introduction

The previous chapter 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 measures the effectiveness of a code apart from any particular decoding scheme. However, implementing a maximum likelihood decoder that actually compares a 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 is applicable only to the Binary Symmetric Channel at rates far below channel 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 digits, 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 cross-over or no cross-overs. Thus, when most of the parity check equations checking on a digit are unsatisfied, it is a strong indication that the digit is incorrect. An incorrect digit will not be corrected on the first try if too many of the parity check sets containing that digit contain other cross-overs also. If some of these other cross-overs are corrected on the first try, then the digit in question may be corrected on the second.

Fig. 4-1 illustrates more clearly how a digit can be corrected on the second decoding try. The branches in the figure represent parity check sets and the nodes represent digits. Some arbitrary digit, d, is at the base of the tree, and using arbitrary numbering, digit i_ℓ is the ℓ'th digit in

the i'th parity check set containing d. Assume that both digit d and several of the digits in the first tier are cross-overs. Then on the first decoding try the error-free digits in the second tier and their parity check constraints will allow correction of the cross-overs in the first tier, which in turn will allow correction of digit d on the second decoding try. 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.

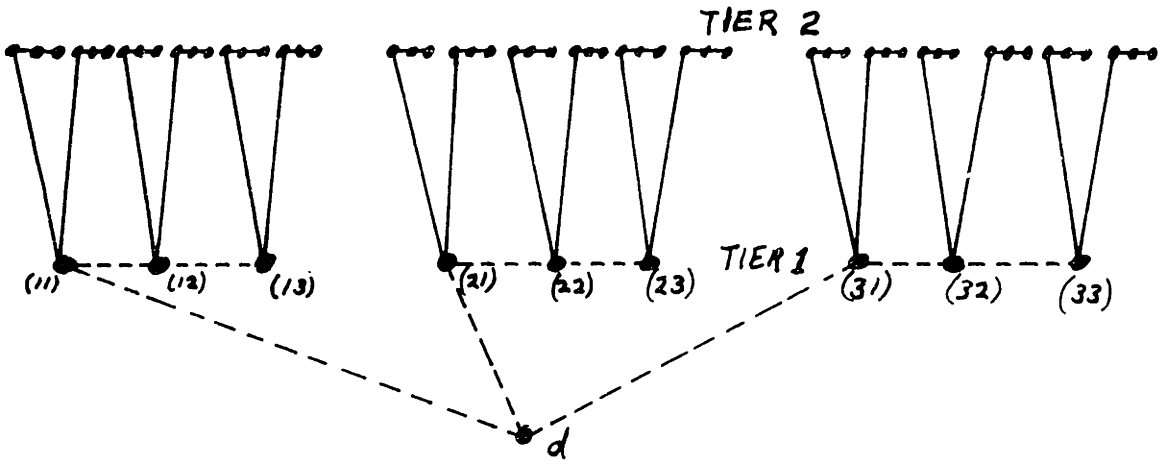

Figure 4-1

PARITY CHECK SET TREE

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 Fig. 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 one 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 sub-code in which all sets of these digits that satisfy the j parity check equations in the tree are equally likely.*

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 $P_x(y)$. The probability of any event in this ensemble 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 sub-code described above. Thus, within this ensemble, we want to find the probability that the transmitted digit in position

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

d is a one 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

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

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

Theorem 4.1:

Let P_d be the probability that the transmitted digit in position d is a one conditional on the received digit in position d, and let $P_{i\ell}$ be the probability that the transmitted digit in position $i\ell$ of the first tier in Fig. 4-1 is a one conditional on the received symbol in position $i\ell$. Let the digits be statistically independent and let S be the event that the transmitted digits satisfy the j parity check constraints on digit d. Then

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

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

Lemma 4.1:

Consider a sequence of m independent binary digits in which the ℓ'th digit is 1 with probability P_ℓ. Then the

probability than an even number of digits are 1 is

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

Proof of 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 ones. 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 the probability of an even number of ones. But

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

proving the lemma.

Proof of Theorem:

By the definition of conditional probabilities,

$$\frac{\Pr(x_d=0 \mid S,\bar{y})}{\Pr(x_d=1 \mid S,\bar{y})} = \frac{1-P_d}{P_d} \frac{\Pr(S \mid x_d=0,\bar{y})}{\Pr(S \mid x_d=1,\bar{y})} \qquad (4.2)$$

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 ones. 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

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

Similarly, if $x_d = 1$, a parity check on d is satisfied if the other (k-1) positions in the set contain an odd number of ones. The probability of an odd number of ones is 1 minus the probability given in lemma 4.1, and taking the product over the j parity checks, we have

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

Substituting Eqs.(4.3) and (4.4) into Eq.(4.2), we get the statement of the theorem.

Q. E. D.

Judging from the complexity of this result, it would appear difficult to compute the probability that the transmitted digit in position d is a one conditional on the received digits in two or more tiers of the tree in Fig. 4-1. Fortunately, however, the many tier case can be solved from the one 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 one 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 Eq.(4.1) to find the probability that the transmitted digit in position d is one. The validity of the procedure follows immediately from the independence of the new values of P_{il} 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 one given any number of tiers 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 Eq.(4.1) to calculate the probability of a transmitted one 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 Eq(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 digit 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.

If this iteration process could be continued until it computed the probability of a transmitted one conditional on the complete received sequence, this procedure would be equivalent to maximum likelihood decoding. However, the iteration process is only valid for as many iterations as meet the independence assumption in Theorem 4.1. This assumption does not hold on the m'th iteration if, first, different nodes in the m'th tier of the tree represent the same digit or if, second, some linear combination of those parity check constraints outside the m tier tree produce a parity check containing only digits in the m'th tier. Since each tier of the tree contains (j-1)(k-1) more nodes than the previous tier, the independence assumption must break down while m is quite small for any code of reasonable block length.

We can ignore this lack of independence, however, and continue using the iterative process. This is ultimately justified, of course, only by the fact that it works. It is also reasonable, however, because when dependencies such as repeated digits and extra constraints start occurring, they are far out in the tree and have a relatively minor effect. Moreover these dependencies affect the probabilities being computed in a random fashion and should cancel out to some extent. 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'th iteration 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 block length. It is an unanswered question, whether the average number of iterations required to decode is a function of the block length. It is reasonable, however, to expect that the average number of iterations is bounded by a quantity independent of code length. The number of iterations that are effective in decoding increases with the code length, but this greater number of iterations need be used only for the improbable noise configuration that would have caused errors in a shorter code. If the probability of requiring m iterations to decode decreases rapidly enough with m independent of the block length, then the average number of iterations to decode is bounded.

Probabilistic decoding can be performed using either series or parallel computing. Series decoding can be programmed for a general purpose computer, and the experimental data in Chapter V were obtained in this manner. For fast decoding, a decoder using parallel computing is more promising; such a decoder could be specially constructed. Both the amount of equipment and the data handling capacity in bits per second would be proportional to the block length. Fig. 4-2 sketches a block diagram of a parallel decoder.

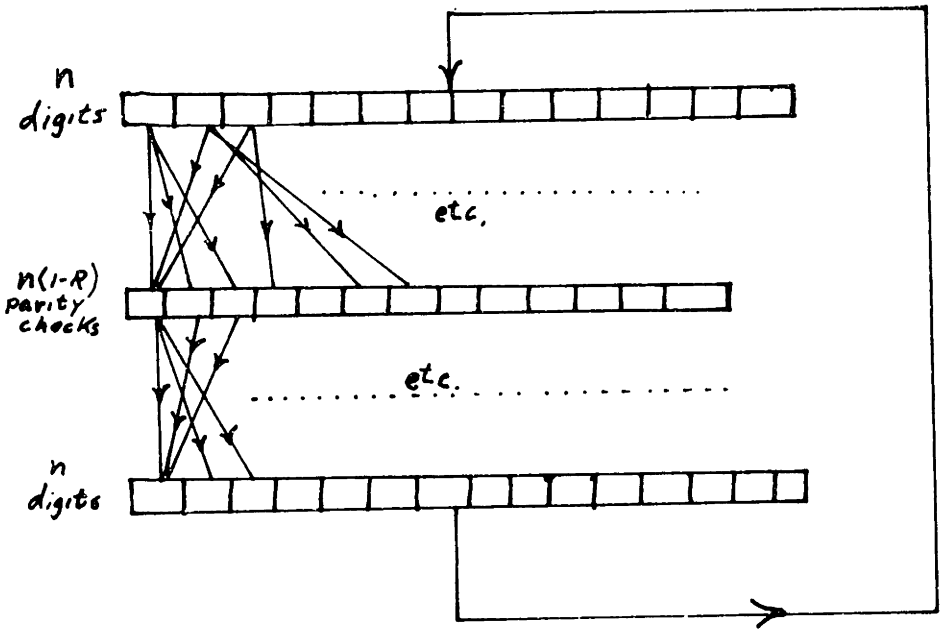

Figure 4-2

BLOCK DIAGRAM OF DECODER

For either series or parallel computing, it may be more convenient to use Eq.(4.1) in terms of log likelihood ratios. Let

$$\ln \frac{\Pr(x_d=0 \mid S,\bar{y})}{\Pr(x_d=1 \mid S,\bar{y})} = y'_d \; x'_d$$

$$\ln \frac{1-P_d}{P_d} = y_d \; x_d \quad ; \quad \ln \frac{1-P_{il}}{P_{il}} = y_{il} \; x_{il} \tag{4.5}$$

when y is ± 1 and $x \geq 0$. After some manipulation, Eq.(4.1)

becomes

$$y'_d\, x'_d = y_d\, x_d + \sum_{i=1}^{j-1} \prod_{l=1}^{k-1} (y_{il})\, f\left[\sum_{l=1}^{k-1} f(x_{il})\right] \qquad (4.6)$$

when

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

Using this formulation, a parallel computer could be simply instrumented requiring principally (4j-1)n analogue adders, (2j-1) modulo 2 adders, and 4jn circuits to compute f(x). The complexity of the non-linear circuit required to compute f(x) depends on the approximations that can be made without materially affecting the decoding process, but presumably a very simple circuit would be sufficient.

Probability of Error Using Probabilistic Decoding

The dependencies that occur after several iterations of the probabilistic decoding scheme make a mathematical analysis of the process extremely difficult. Even in the absence of dependencies, a mathematical analysis is still difficult. Conceptually, given a distribution function of channel transition probabilities, Eq.(4.1) can be iteratively used to calculate successive distribution functions for the probability that a digit is in error. This is a complicated computation that would have to be performed on a computer, however, and an analysis of the resulting approximations would be difficult. A simple procedure that avoids these

mathematical and computational problems will be used in this section to bound the probability of decoding error using probabilistic decoding. Although this bound is quite weak, it provides additional insight into the decoding process.

Assume a binary symmetric channel with cross-over probability p_o and assume an (n,j,k) code with j=3 parity check sets containing each digit. Consider a parity check set tree, as in Fig. 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 follows: 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; 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 from 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 second procedure always makes the most likely decision from this information.

We now determine the probability that a digit in the first tier is in error after applying the modified decoding

procedure described above. If the digit is received in error (an event of probability p_o) then a parity check constraining that digit will be unsatisfied if and only if an even number of errors occurs among the other k-1 digits in the parity check set. From lemma 4.1, the probability of an even number of errors amoung k-1 digits is

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

Since an error will be corrected only if both parity checks missing 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.

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

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

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

Combining Eqs.(4.8) and (4.9), the probability of error of a digit in the first tier after applying this decoding

process is

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

Similarly, using the fact that each digit in the first tier has probability p_1 of being in error after processing, the probability of error of a digit in the second tier after processing is

$$p_2 = p_o - p_o\left[\frac{1 + (1-2p_1)^{k-1}}{2}\right]^2 + (1-p_o)\left[\frac{1 - (1-2p_1)^{k-1}}{2}\right]^2$$

and the probability of error of a digit in tier $i + 1$ after processing is

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

We now show that for sufficiently small p_o, the sequence $[p_i]$ converges to 0. Consider Fig. 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 zig-zig line illustrates a convenient graphical method of finding p_i for successive values of i. It can be seen from the figure that if

$$0 < p_{i+1} < p_i \quad (\text{for } 0 < p_i \leq p_o) \tag{4.12}$$

$$p_{i+1} = p_i \quad (\text{for } p_i = 0)$$

then the sequence $[p_i] \to 0$.

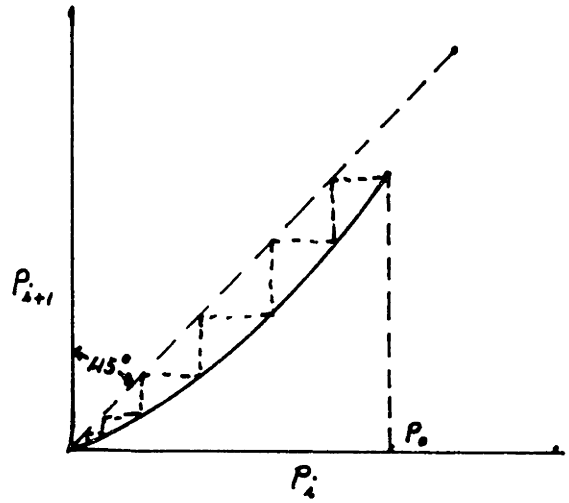

Figure 4-3

p_{i+1} AS A FUNCTION OF p_i

Mathematically, Eq.(4.12) insures that the sequence $[p_i]$ is bounded and decreasing, implying that it converges to a limit point. As a result of the continuity of Eq.(4.11), p_{i+1} must be equal p_i at the limit point. Since $p_{i+1} = p_i$ only for $p_i = 0$, the sequence converges to 0.

Next it is shown that the condition of Eq.(4.12) holds for sufficiently small p_o. Using Eq.(4.11), we find

$$\frac{\partial p_{i+1}}{\partial p_o} = \frac{1 - (1-2p_i)^{2(k-1)}}{2} \geq 0$$

Thus, decreasing p_o and holding p_i constant decreases p_{i+1}, so that if Eq.(4.12) holds for one value of p_o, it must hold for all smaller values of p_o. Also from Eq.(4.11) we note that $p_{i+1} = 0$ for $p_i = 0$ and

$$\left.\frac{\partial p_{i+1}}{\partial p_i}\right|_{p_i=0} = 2(k-1)p_o$$

For $p_o < \frac{1}{2(k-1)}$, the initial slope of p_{i+1} as a function of p_i is less than 1, and since $\frac{\partial p_{i+1}}{\partial p_i}$ is continuous,

$$0 \leq p_{i+1} < p_i$$

For sufficiently small p_i. Then making p_o sufficiently small, Eq.(4.12) must be satisfied, and $[p_i] \to 0$. Fig. 3-1 plots the largest value of p_o for which Eq.(4.12) is satisfied for several values of j and k. If p_o is less than this value, then as the block length, n, approaches infinity, the number of independent tiers in the parity check set tree approaches

infinity, and the probability of decoding error approaches zero.

The rate at which $[p_i] \to 0$ may be determined by noting that for small p_i

$$p_{i+1} \approx p_i \, 2(k-1)p_o$$

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

$$p_i \approx C \, 2(k-1)p_o^{\,i}$$

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 0 with some small negative power of block length. This slow approach to 0 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 of the parity check constraints rising from the digit are violated. Using this criterion and following the reasoning leading to Eq(4.11),

we obtain

$$p_{i+1} = p_o - p_o \sum_{\ell=b}^{j-1} \binom{j-1}{\ell} \left[\frac{1+(1-2p_i)^{k-1}}{2}\right]^{\ell} \left[\frac{1-(1-2p_i)^{k-1}}{2}\right]^{j-1-\ell}$$

$$+ (1-p_o) \sum_{=b}^{j-1} \binom{j-1}{\ell} \left[\frac{1-(1-2p_i)^{k-1}}{2}\right]^{\ell} \left[\frac{1+(1-2p_i)^{k-1}}{2}\right]^{j-1-\ell} \qquad (4.13)$$

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

$$\frac{1-p_o}{p_o} \leq \left[\frac{1 + (1-2p_i)^{k-1}}{1 - (1-2p_i)^{k-1}}\right]^{2b-j+\ell} \qquad (4.14)$$

From this equation, it is seen that as p_i decreases, b also decreases. Fig. 4-4 sketches p_{i+1} as a function of p_i when b is changed according to Eq.(4.14). The break points in the figure represent changes of b.

The proof that the probability of decoding error approaches 0 with an increasing number of iterations for sufficiently small cross-over probabilities is the same as before. The asymptotic approach of the sequence $[p_i]$ to 0 is different, however. From Eq.(4.14), as $p_i \to 0$, b takes the

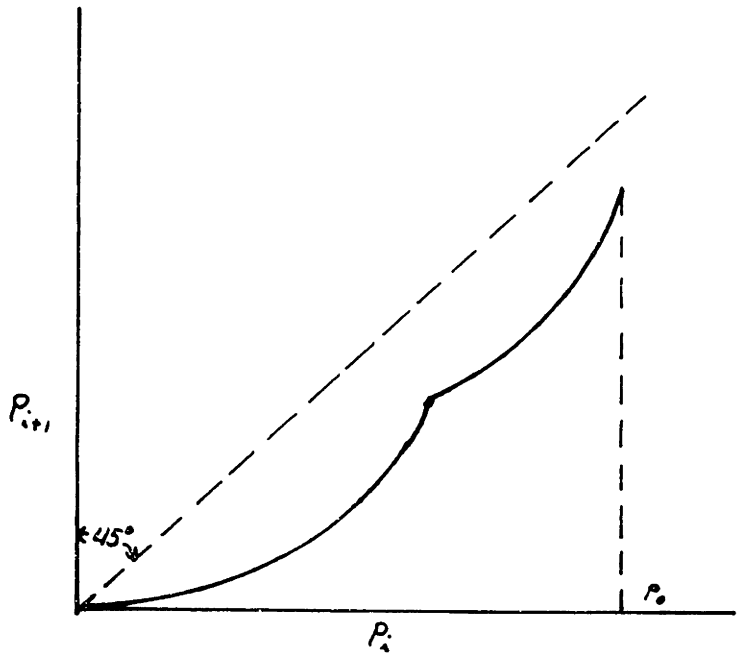

Figure 4-4

p_{i+1} AS A FUNCTION OF p_i WITH VARIABLE b

value $\frac{1}{2}$ for j even and $\frac{j+1}{2}$ for j odd. Using these values of b and expanding Eq.(4.13) in a power series in p_i

$$p_{i+1} = p\binom{j-1}{\frac{j-1}{2}}(k-1)^{\frac{j-1}{2}} p_i^{\frac{j-1}{2}} + \text{higher order terms} \quad (j \text{ odd}) \quad (4.15)$$

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

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

$$p_i \leq \exp -C_{jk}\left(\frac{j-1}{2}\right)^1 \quad (j \text{ odd}) \qquad (4.16)$$

$$p_i \leq \exp -C_{jk}\left(\frac{j}{2}\right)^1 \quad (j \text{ even})$$

It is interesting to relate this result to the block length of the code. The number, m, of independent tiers in a parity check set tree is a random variable dependeng on both the digit and the code. On the average, however, dependencies start to occur when a tree contains on the order of n digits. Since each independent tier in a parity check set tree contains (j-1)(k-1) digits for each digit in the tier below it, the number of digits in an m tier tree is proportional to

$$\left[(j-1)(k-1)\right]^m .$$

Thus assume that for some constant A,

$$n = A\left[(j-1)(k-1)\right]^{2m} \qquad (4.17)$$

Combining Eqs.(4.16) and (4.17), the probability of decoding error for a typical tier size is

$$p_m \leq \exp -C_{jk}\left[\frac{n}{A}\right]^{\frac{\ln(j-1)/2}{2\ln(j-1)(k-1)}} \quad (j \text{ odd})$$

$$p_m \leq \exp -C_{jk}\left[\frac{n}{A}\right]^{\frac{\ln\ j/2}{2\ln(j-1)(k-1)}} \quad (j \text{ even})$$

For $j > 3$, this probability of decoding error bound decreases exponentially with a root of n. Observe that if the number of iterations, m which can be made without dependencies were $\frac{2 \ln(j-1)(k-1)}{\ln j/2}$ 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. It is also reasonable to hypothesize that the maximum number of iterations necessary to decode increases logarithmically with block length. If both of these hypotheses are true, then a decoder can achieve an exponential decrease in the probability of decoding error at the expense of a linear increase in storage and an at most logarithmic increase in computation. Unfortunately, experimental proof or disproof of these hypotheses is likely to be difficult for reasons given in the next chapter.

CHAPTER V

EXPERIMENTAL RESULTS

Introduction

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 $\frac{1}{P(e)}$ trials are necessary. For block lengths of about 500, the I. B. M. 704 computer requires about 2.5 seconds per iteration to decode by the probabilistic decoding scheme of Chapter IV. Consequently, many hours of computation time are necessary to evaluate even a P(e) of the order of 10^{-3}.

Any other experimental approach to determining P(e) would involve measuring some other quantity that could be used to determine P(e). For instance, for a maximum likelihood decoding scheme, the distance properties of the code could be experimentally measured and used to compute P(e) as in Chapter III. So little is known theoretically about probabilistic decoding, however, that the only alternative to direct measurement is a measurement with some of the

channel variations eliminated. For instance, on the BSC, it is more convenient to measure $P_c(e)$, the probability of decoding error given c cross-overs. This reduces the number of trials somewhat, but does not eliminate the need for many trials where $P_c(e)$ is small.

Because of limitations on available computer time, all of the results presented will be for situations in which $P(e)$ or $P_c(e)$ is large. Certainly it would be more interesting to have results for small $P(e)$. However, the data presented are at least sufficiently convincing to justify further experimental work, for which suggestions are given in Chapter VI.

A block length of about 500 was chosen for all of the experiments since this length is long enough to provide large trees for decoding, but short enough to avoid storage problems in the computer. A value of $j = 3$ parity check sets per digit was used for all of the codes except the cyclic code since preliminary investigation showed that the decoding scheme is most effective for $j = 3$.

The first three codes to be discussed were used on the BSC and the last code on a Gaussian noise channel. The BSC was unduly emphasized for the following reasons: first, the effect of channel variations on the BSC can be eliminated by evaluating $P_c(e)$ instead of $P(e)$; next, the BSC is convenient for comparison with other coding and decoding schemes; and finally, it is likely that the operation of the

decoding scheme on one channel is typical of its operation on other channels.

(504,3,6) Code on Binary Symmetric Channel

A code of block length 504 with each digit contained in three parity check sets and each parity check set containing 6 digits was selected by the IBM 704 computer using a pseudo-random number routine. The only restriction on the code was that no two parity check sets should contain more than one digit in common. That restriction guaranteed the validity of the first order iteration in the decoding process and also excluded the remote possibility of choosing a code with a minimum distance of 2.

For various numbers, c, of channel cross-overs, an experimental determination was made of $P_c(e)$, the probability of decoding error given c cross-overs. This was done by feeding a random permutation of c ones into a simulation of the probabilistic decoding process described in the last chapter. The transmitted sequence was thus assumed to be the zero sequence since the decoding process locates errors in the same way regardless of the transmitted code word.

Eq.(4.6) was used to compute the log likelihood ratio, and decoding was considered complete when all the parity checks were satisfied. In a probabilistic formulation, the parity checks are computed by first assigning to each digit its most likely value. It follows that, in the notation of Eq.(4.5), if $\prod_{i=1}^{k} y_i = 1$, parity is satisfied and if

$\prod_{i=1}^{k} \gamma_i = -1$, parity is unsatisfied.

Fig. 5-1 plots the fraction of times the decoder was unable to decode correctly as a function of the number of cross-overs. The number in parentheses beside each point is the number of trials performed with that number of cross-overs. In all the trials on this code, the decoder never decoded to the wrong code word; it just failed to find a code word. If a feedback channel is available, this inability to decode troublesome noise patterns is not a serious limitation, since retransmission is possible.

Out of the error patterns correctly decoded, 86% were decoded in between 9 and 19 iterations. The rest were spread out between 20 and 40 iterations. There appeared to be a slight increase in the number of iterations necessary to decode as the number of cross-overs was increased from 37 to 41, but not enough to be statistically significant.

The other curve drawn in Fig. 5-1 is the theoretical bound from Eq.(3.2) for $P_c(e)$ using maximum likelihood decoding. In evaluating Eq.(3.2), an asymptotic form for P(L) was taken from Eqs.(2.12) and (2.13), and an asymptotic formula was used for summing L around its maximum term.

These results appear encouraging when we observe that no other known coding and decoding scheme of this rate is able to decode this many errors with a reasonable amount of computation. How well the decoding scheme works with smaller numbers of errors is of greater interest, though. The rate

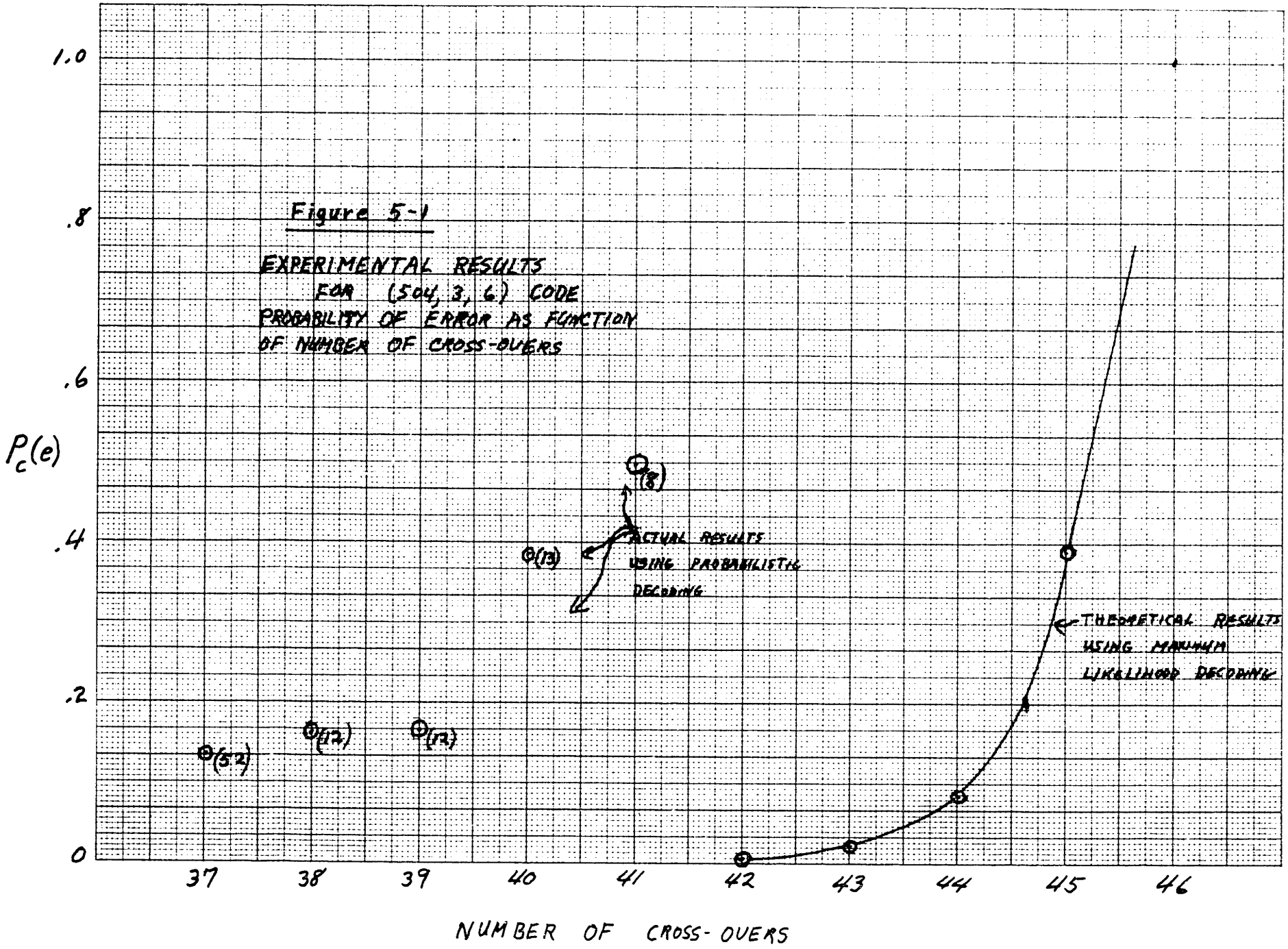

Figure 5-1

EXPERIMENTAL RESULTS
FOR (504, 3, 6) CODE
PROBABILITY OF ERROR AS FUNCTION
OF NUMBER OF CROSS-OVERS

at which the experimental $P_c(e)$ decreases as c decreases is discouraging, but there is no justification for extrapolating this curve to much smaller numbers of cross-overs. Either a great deal of additional experimental data or a new theoretical approach will be necessary to evaluate $P_c(e)$ at smaller values of c.

A (500,3,4) Code on the Binary Symmetric Channel

A (500,3,4) code, which has rate, $\frac{1}{4}$, was chosen by the IBM 704 computer in the same way as the (504,3,6) code of the last section. Sequences containing from 20 to 77 cross-overs were put in to be decoded. There were two sequences for each number of cross-overs from 65 to 69 and from 72 to 77, and one sequence for all the other numbers. The decoding was successful for all sequences except one 73 cross-over case, one 75 cross-over case, and both 77 cross-over cases. The theoretical error correcting breakpoint for the (500,3,4) ensemble from Fig. 3-1 is 103 errors, and the error correcting breakpoint for the ensemble of all codes of rate $\frac{1}{4}$ is 108 errors.

Low Density Cyclic Code

A cyclic matrix is a matrix in which each row is a single cyclic shift of the previous row. A cyclic code is described by a cyclic n by n parity check matrix, and the row rank of this matrix is the number of independent parity check equations.* A low density cyclic code is a cyclic code

* For a complete discussion of cyclic codes, see ref.(9)

in which each row of the parity check matrix contains a very small number of ones.

Low density cyclic codes have two advantages over other low density codes. First, the code words may be generated by a shift register with a complexity proportional to the block length n, whereas for an ordinary low density code, the coding complexity is proportional to n^2. Second, in the cyclic code there are n low density parity check sets available in the decoding process instead of n(1-R). That these n parity checks are dependent does not matter, since the decoding depends only on independence in the first few tiers of the parity check set trees as in Fig. 4-1. On the other hand, the theory developed here does not apply to cyclic codes. Also, no method is known to find low density cyclic codes with reasonable minimum distance properties.

The probabilistic decoding scheme was applied to a cyclic code of length 511 for which the first parity check set contained digits 1,2,8,16,64,128. This code has 127 information digits and a rate of approximately $\frac{1}{4}$. An attempt was made to decode randomly chosen sequences of 70, 74,78,82, and 86 cross-overs. The first three trials were successful, but decoding occurred to a code word of weight 28 in the 82 cross-over case. This was the only example of an incorrect decoding in all the experimental work on all the codes tested; no decoding was possible for the 86 cross-over case.

(500,3,5) Code on White Gaussian Noise Channel

Assume a channel that accepts inputs of plus or minus 1 and adds a Gaussian random variable of mean 0 and variance 1 to the input to form the output. The log likelihood ratio of the input conditional on the output is simply twice the received signal. The channel capacity of this channel can be calculated[1] to be .5 bits per symbol. However, if the receiver converts the channel into a BSC by making a decision on each symbol and throwing away the probabilities, the probability of cross-over becomes .16, and the channel capacity is reduced to .37 bits per symbol.

In this experiment a (500,3,5) code, which has rate .4 bits per symbol, was simulated on the IBM 704 computer along with the channel just described. Probabilistic decoding was performed using the log likelihood ratios at the output of the channel. Out of 13 trials, the decoding scheme decoded correctly on 11 trials and failed to decode twice.

This experiment is interesting since it suggests that the loss of rate necessitated by the non-optimum coding and decoding techniques proposed here is more than compensated for by the opportunity of using the a posteriori probabilities at the channel output.

CHAPTER VI

SUGGESTIONS FOR FUTURE WORK

The coding and decoding scheme presented in the previous chapters represents an attempt to achieve the low error probabilities possible with coding while avoiding an excessive cost in terms of equipment and computation.

The theoretical analysis of low density codes presented in Chapters II and III provides an adequate picture of the codes themselves. There is opportunity for further work in obtaining upper bounds to the minimum distance of the codes and lower bounds for the probability of decoding error. Also, it might be possible to simplify the upper bound to the probability of decoding error on binary symmetric input channels. Without better bounds on the decoding scheme, however, these problems are of greater academic than practical interest.

The most promising area for future work is further analysis of the decoding scheme, both theoretical and experimental. The bound in Chapter IV for the probability of decoding error using the probabilistic decoding scheme is weaker than necessary for two reasons: first, the decoding model

that is used for the bound not only throws away the channel a posteriori probabilities before decoding, but also throws away the probabilities at every iteration of the decoding process; second, for mathematical expediency, the decoding is stopped as soon as dependencies occur in the iteration process.

A closer bound to the probability of error using probabilistic decoding can be obtained by actually computing the distribution function of the probability of error associated with a digit after an arbitrary number of iterations, assuming no dependencies. Eq.(4.1) expresses the probability of error for a digit after the first iteration as a function of the a posteriori probabilities of each digit in the one tier tree. Thus, given the distribution function of the a posteriori probabilities connected with a received digit, Eq.(4.1) can be used to find the distribution function of the probability of error after one iteration. In a similar manner, the distribution function of the probability of error of a digit after any iteration can be computed from the distribution function of the probability of error after the previous iteration. This procedure must be performed numerically on a computer, and the resulting approximations are difficult to analyze, although they should not affect the overall behavior of the process.

Bounding the probability of error by computing these iterative distribution functions is still limited by being unable to take dependencies into account. For any particular

channel, however, the rate at which these iterative distribution functions converge to 0, if they do converge to 0, might well be combined with some experimental work to provide estimates of the probability of error as a function of code length. Because of the dependencies in the iterative process, it is possible that the decoding scheme will work on some channels for which these distribution functions do not converge to 0, although it is doubtful if very low error probabilities could be achieved in these situations.

An analysis of the probability of decoding error taking into account the dependencies in the iteration process appears to be quite difficult. Two possible approaches are, first, to analyze the effect of a single dependency in the hope of generalization, and second, to analyze the dependencies on a statistical basis.

The experimental data presented in Chapter V treat only channels that are so noisy that the probability of decoding error is large. The more important question of how well the decoding scheme works on less noisy channels remains to be investigated. To answer this question, a parallel decoder of the type described in Chapter IV would have to be constructed, since only then would very large numbers of trials be practical. Some additional experiments would be desirable on a computer before starting such a large project, however.

Various modifications of the decoding scheme itself can be tried experimentally with the dual purposes of lowering

the probability of decoding error and finding cheaper and easier ways to construct a parallel decoder. Next, enough additional data could be taken on the (504,3,6) code to extend Fig. 5-1 down to about 32 errors. This might indicate whether a parallel decoder is worth constructing. Finally, experiments should be performed at some different block lengths to determine the dependence of the probability of decoding error on block length.

APPENDIX

PROPERTIES OF THE FUNCTION $E(\lambda)$

In Chapter II, the following bound was derived for the minimum distance distribution of an (n,j,k) ensemble of codes.

$$\Pr(D \leq n\delta) \leq \sum_{L=2}^{n} C(\lambda,n) \exp -nE(\lambda) \qquad (A.1)$$

$$\leq 1$$

where

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

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

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

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

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

In this appendix three theorems will be proved concerning Eq.(A.1). The first theorem will analyze the behavior of $E(\lambda)$, the second will bound the summation in Eq.(A.1) in

terms of the first and last terms, and the third will show that as j and k get large, Eq.(A.1) approaches the minimum distance distribution function derived for the ensemble of all codes in Eq.(2.1).

Theorem A 1.

Assume $k > j \geq 3$, and let $E(\lambda)$ be defined in Eqs. (A.2), (A.4), and (A.5). Then

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

2) $\lim_{\lambda \to 0} \frac{dE}{d\lambda} = \infty$,

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

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

Proof:

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

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

and from this, $s \to -\infty$ as $\lambda \to 0$. But from Eq.(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 Eq.(A.2),

$$\frac{dE(\lambda)}{d\lambda} = \frac{\partial E(\lambda)}{\partial \lambda} + \frac{\partial E(\lambda)}{\partial s}\left[\frac{\partial \lambda}{\partial s}\right]^{-1} = (j-1)\ln\frac{1-\lambda}{\lambda} + js$$

Making the substitution

$$z = \frac{1-e^{s}}{1+e^{s}}, \qquad s = \ln\frac{1-z}{1+z}, \qquad (A.7)$$

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

$$\lambda = \frac{1-z}{2}\,\frac{1-z^{k-1}}{1+z^{k}} \qquad (A.8)$$

s and λ are sketched as a function of z in Fig. A-1.

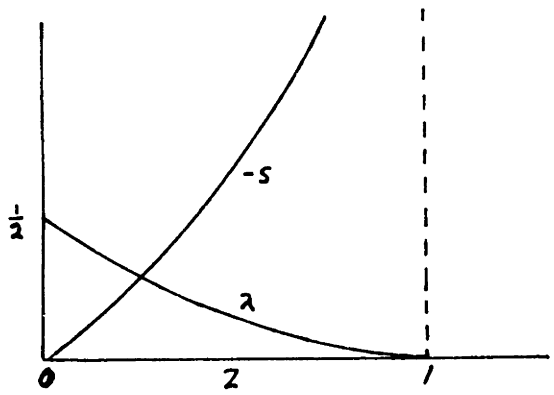

Figure A-1

s AND λ AS FUNCTIONS OF z

$$\lim_{\lambda\to 0}\frac{dE}{d\lambda} = \lim_{z\to 1}\ (j-1)\ln\left(\frac{1+z}{1-z}\right)\left(\frac{1+z^{k-1}}{1-z^{k-1}}\right) + j\ln\frac{1-z}{1+z}$$

$$= \lim_{z\to 1}\ \ln\left(\frac{1-z}{1+z}\right)\left(\frac{1+z^{k-1}}{1-z^{k-1}}\right)^{j-1}$$

$$= \lim_{z\to 1}\ \ln\frac{(1+z^{k-1})^{j-1}}{(1-z^{k-1})^{j-2}(1+z)(1+z+\ldots+z^{k-2})} \qquad (4.9)$$

$$= \infty \quad \text{for } j-2 > 0, \text{ or in other words, for } j \geq 3.$$

3) Before proving parts 3 and 4 of the theorem, we must show that $\frac{dE}{d\lambda}$ has only one extremum. Using Eq.(A.9), we obtain the derivitive of $\frac{dE}{d\lambda}$ with respect to z.

$$\frac{d}{dz}\left(\frac{dE}{d\lambda}\right) = -\frac{2}{1-z^2} + \frac{2(j-1)(k-1)z^{k-2}}{1-z^{2(k-1)}}$$

Setting this equal to 0, we have

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

$$(j-1)(k-1) = 1 + \sum_{i=1}^{\frac{k-2}{2}}\left[z^{2i} + \frac{1}{z^{2i}}\right] \quad \text{(for k even)} \qquad (A.10)$$

$$= \sum_{i=1}^{\frac{k-1}{2}}\left[z^{2i-1} + \frac{1}{z^{2i-1}}\right] \quad \text{(for k odd)} \qquad (A.11)$$

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

$$E\left(\frac{1}{2}\right) = \left[(j-1)\ln 2 - \frac{j}{k}(k-1)\ln 2 - \left(1-\frac{j}{k}\right)\ln 2\right] < 0$$

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

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

Q. E. D.

Theorem A.2

For an (n,j,k) ensemble of codes, the minimum distance distribution function may be bounded by*

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

$$\leq 1$$

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

Proof:

From Eq(2.14), we have

$$\Pr(D < n\delta) \leq \sum_{L=2}^{n\delta} \binom{n}{L}^{-j+1} \left[N(L)\right]^j$$

We can evaluate the term for L = 2 directly. N(2) is the number of sequences of weight 2 which satisfy the first $\frac{n}{k}$ 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 $\frac{n}{k}$ parity check sets, we have $\frac{n}{k}\binom{k}{2}$.

$$\binom{n}{2}^{-j+1} N(2)^j = \frac{n(k-1)^j}{2(n-1)^{j-1}} = \frac{(k-1)^j}{2n^{j-2}} + O(n^{-j+2})$$

$$\Pr(D < n\delta) \leq \frac{(k-1)^j}{2n^{j-2}} + O(n^{-j+2}) + \sum_{L=4}^{n\delta} C(\lambda, n)\exp -nE(\lambda) \qquad (A.13)$$

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

$$E(\lambda) \geq (j-1)\left[\lambda \ln\frac{1}{\lambda} + (1-\lambda)\ln\frac{1}{1-\lambda}\right] - \frac{j}{k}\ln \sum_{i \text{ even}} \binom{k}{i} e^{si} + \frac{j}{2}\lambda \ln\frac{\lambda}{k-1}$$

$$E(\lambda) \geq \left(\frac{j}{2} - 1\right)\lambda \ln \frac{1}{\lambda} - \frac{j}{k} \ln \left[\frac{1}{1 - \left(\frac{k}{2}\right)e^{2s}}\right] - \frac{j}{2}\lambda \ln (k-1)$$

Substituting $\frac{L}{n}$ for λ and using some inequalities we have

$$\exp -nE(\lambda) \leq n^{-L\left(\frac{j}{2}-1\right)} L^{L\left(\frac{j}{2}-1\right)} (k-1)^{\frac{jL}{2}} \exp \frac{Lj}{2} \frac{1}{1 - \frac{KL}{2n}} \qquad \text{(A.15)}$$

From Eq.(A.3), we get

$$C(\lambda, n) \leq (2\pi L)^{\frac{j-1}{2}} \exp \frac{j-1}{3n} \qquad \text{(A.16)}$$

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

$$nC_{max}\left[\exp -nE\left(\frac{8}{n}\right) + \exp -nE(\delta)\right]$$

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

$$n^{\left[1 + \frac{j-1}{2} + 8\left(-\frac{1}{2} + 1\right)\right]} = O(n^{-j+2}) \text{ (for } j \geq 3)$$

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

the theorem.

Q. E. D.

Theorem A.3:

Let δ_{jk} be the non-zero solution of $E(\lambda) = 0$ for an (n,j,k) ensemble, and let $R = 1 - \frac{j}{k}$ be fixed. Let $\delta_o < \frac{1}{2}$ be the solution of $H(\delta_o) = (1-R)\ln 2$. Then $\lim_{k\to\infty} \delta_{jk} = \delta_o$.

From Theorem 2.1, δ_o is the typical minimum distance for the ensemble of all random codes, so the theorem asserts that the typical minimum distance of (n,j,k) codes approaches that of all codes as k gets large.

Proof:

Using Eq.(A.2), $E(\lambda)$ can be rewritten in the form

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

We shall show that for $\lambda \neq 0$, the last bracket in Eq.(A.17) approaches 0 with increasing k. This is sufficient to prove the theorem, since $\frac{j}{k} = 1-R$ and thus the first bracket is zero only for $\lambda = \delta_o$.

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

Making the substitution $z = \frac{1-e^s}{1+e^s}$ of Eqs.(A.7) and (A.8),

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

Also,

$$\frac{1}{k}\ln\left[(1+e^{s})^{k} + (1-e^{s})^{k}\right] = \ln(1+e^{s}) + \frac{1}{k}\ln(1+z^{k})$$

$$= \ln\frac{2}{1+z} + \frac{1}{k}\ln(1+z^{k}) \tag{A.20}$$

Combining Eqs.(A.19) and (A.20), the second bracket in Eq.(A.18) becomes

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

As k gets large, for any $z < 1$ (ie. $\lambda > 0$), z^{k} and z^{k-1} approach 0. Expanding the logarithm we have

$$-j\left(\frac{1-z}{2}\right)\left(\frac{1-z^{k-1}}{1+z^{k}}\right)2z^{k-1} + jz^{k-1}(1-z) + \frac{jz^{k}}{k} + \text{higher order terms}$$

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

Q. E. D.

BIOGRAPHY

Robert Gallager was born in Philadelphia, Pa. in 1931 and graduated from the Moore School, University of Pennsylvania in 1953 with the degree of B.S. in E. E.. After working at Bell Telephone Laboratories for a year and a half, he was drafted into the army and spent the next two years at the Signal Corps Proving Grounds, Fort Huachuca, Arizona.

Mr. Gallager entered M. I. T. in September 1956 with a Sperry Gyroscope Fellowship and received his M.S. in E. E. in September 1957. Out of the next three years at M.I.T, two years were spent as a Research Assistant in the Research Laboratory of Electronics, and the other as the recipient of a Bendix Fellowship. During this time he taught an undergraduate course in Communication Theory and worked for two summers at Lincoln Laboratories on Digital Communication on Telephone Lines.

BIBLIOGRAPHY

1. Bloom, F.J., Chang, S.S.L., et al., "Improvement of Binary Transmission by Null-Zone Reception," Proc. I.R.E., Vol.45, 1957, pp.963-975.

2. Bose, R.C. and Ray-Chaudhuri, D.K., "A Class of Error-Correcting Binary Group Codes," Inf. and Control, Vol. 3, March, 1960, pp. 68-79.

3. Elias, P., "Coding for Two Noisy Channels," Information Theory, (C. Cherry, Editor), Third London Symposium, September, 1955.(Buttersworth Scientific Publications, London, England).

4. Fano, R.M., The Transmission of Information, (The Technology Press, Cambridge, Mass.) 1960.

5. Gilbert, E.N., "A Comparison of Signaling Alphabets," Bell System Tech. Jour., Vol,28, 1950, pp.193-198.

6. Gnedenko and Kolmogorov, Limit Distributions for Sums of Independent Random Variables,(Addison-Wesley Publishing Co., Cambridge, Mass.) 1954.

7. Hamming, R.W., "Error Detecting and Error Correcting Codes," Bell System Tech. Jour., Vol. 28, 1950, pp. 193-198.

8. Horstein, M., "Experimental Study of Sequential Decoding for Binary Symmetric Channel," Lincoln Lab Group Report, 1958, pp. 34-74.

9. Petersen, W.W., Unpublished Course Notes for Course 6.575, Spring, 1960 at M. I. T. (To appear in book form later).

10. Reiffen, B., "Sequential Encoding and Decoding for the Discrete Memoryless Channel," Ph. D. Thesis, M.I.T., September, 1960.

11. Shannon, C.E., "Certain Results in Coding Theory for Noisy Channels," Information and Control, Vol. 1, September, 1957, pp. 6-25.

12. Shannon, C.E., Unpublished Seminar Notes, Dept. Of Electrical Engineering, M.I.T., 1956.

13. Slepian, D., "A Class of Binary Signalling Alphabets," Bell System Tech. Jour., Vol. 35, January, 1956, pp. 203-234.

14. Wozencraft, J.M., "Sequential Decoding for Reliable Communication," M.I.T.Research Laboratory for Electronics Report No. 325, August 9, 1957.

15. Wozencraft, J.M., and Horstein, M., "Coding for Two Way Channels," Fourth London Symposium on Information Theory, September, 1960.