ERROR MECHANISMS FOR CONVOLUTIONAL CODES

by

EDWARD ANDREW BUCHER

S.B., Massachusetts Institute of Technology
(1965)

S.M., Massachusetts Institute of Technology
(1967)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
September, 1968

Signature of Author________________________
Department of Electrical Engineering, August 19, 1968

Certified by________________________

Accepted by________________________
Chairman, Departmental Committee on Graduate Students

ERROR MECHANISMS FOR CONVOLUTIONAL CODES

by

EDWARD ANDREW BUCHER

Submitted to the Department of Electrical Engineering on August 19, 1968 in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

ABSTRACT

This thesis presents upper and lower bounds to the probability of error for convolutional codes. The lower bound is derived for an optimum decoder with convolutional codes in which each of the V channel symbols generated per encoder shift may have a different "constraint length." This lower-bound is of the form $P(E) > \exp - K^*V[E_L(R) - o_1(K^*)]$ where K^*V is the sum of the V generator lengths and $o_1(K^*)$ is a function which approaches zero as K^* approaches infinity. An ensemble average upper-bound is derived for multiple generator length convolutional codes with optimum decoding. This upper-bound may be written as $\overline{P(E)} \leq \exp - K^*V[E_U(R) - o_2(K^*)]$ provided that the length of the second shortest generator is proportional to K^*. For $R \geq E_0(1)$, $E_L(R) = E_U(R)$ on symmetric channels.

The Fano sequential decoding algorithm is also investigated. A rather surprising result is found for systematic convolutional codes with sequential decoding. If the sequential decoder bias B equals the data rate R, then the sequential decoder has the same error exponent as optimum decoding for equal generator length convolutional codes. On the other hand, sequential decoding on systematic convolutional codes has a considerably lower error exponent then optimum decoding with the same K^*V. This non-optimality of sequential decoding may be removed by increasing the decoder bias B. Unfortunately, this increase in bias substantially increases the decoder computation.

THESIS SUPERVISOR: Robert G. Gallager
TITLE: Professor of Electrical Engineering

ACKNOWLEDGEMENT

I would like to acknowledge my thesis supervisor Professor Robert G. Gallager. Professor Gallager has provided guidance throughout this research and has suggested several simplifications which have broadened the applicability of this thesis.

I would also like to acknowledge my two thesis readers, Professor R. S. Kennedy and Doctor G. David Forney. Professor Kennedy served as a willing listener throughout my graduate work at M.I.T. Dr. Forney offered several suggestions during the writing of proofs for chapter III and performed the simulations reported in chapter IV.

I would like to thank Codex Corporation for allowing Dr. Forney to serve as a thesis reader while in their employ and for providing access to the data taken in Dr. Forney's sequential decoder simulation.

The support provided by the Research Laboratory of Electronics and the computations performed by the M.I.T. Computation Center are gratefully acknowledged.

Finally, I would like to acknowledge the support provided by National Science Foundation Fellowships.

TABLE OF CONTENTS

TABLE OF CONTENTS CONTINUED

FIGURES AND TABLES

I. INTRODUCTION

Most modern statistical work in communication theory stems from Shannon's proof of the coding theorem in 1948. Communication is essentially the process of transmitting information from one point to another through a noisy channel. A simple example of a noisy channel is the discrete memoryless channel (DMC). If symbol i, one of I possible symbols, is inserted into the DMC, one of J symbols, for example symbol j is received. The relationship between the symbol i and the symbol j is known only through a set of probabilities P(j/i). This set of IJ transition probabilities completely characterizes the channel noise. The DMC is a somewhat idealized model of a noisy channel with digital input and with quantized or digital output.

In designing communications systems, a specific signal is assigned to each of the M messages which the system might be called upon to transmit. If the transmission is to be over a DMC, these signals are sequences of channel input symbols. The selection rule which assigns a transmitted signal to each possible message is called the code. The coding theorem demonstrates the existence of codes which achieve arbitrarily low probability of erroneous communication if and only if the information transmission rate R is less than some maximum rate C called the channel capacity.

Perhaps the key words in the coding theorem are demonstrates and existence. Shannon demonstrated the coding theorem by showing that at least one code in a very large collection or ensemble of codes can achieve arbitrarily low probability of erroneous communication if the information rate R is less than the channel capacity C. Unfortunately, the coding theorem does not specify which codes give a low probability of error. The question of which codes give good performance has been addressed by many authors in the last twenty years. In 1950, R. W. Hamming presented the first error-correcting code. This Hamming code was the forerunner of many block codes presented by numerous authors. These block codes generate a block of N channel symbols when given a block of K information symbols. Much research has been done on block codes and the results have been presented in detail by Peterson (1960), Massey (1967), Berlekamp (1968) and Gallager (1968). In many applications, the information symbols to be transmitted arrive at the encoder serially rather than in large blocks. A type of code which takes advantage of the serial nature of incoming data is the convolutional code first presented by Elias (1954). Convolutional codes have not been studied as much as block codes. This thesis presents several significant results about convolutional codes.

Convolutional codes can be most easily explained by describing the encoder. Moreover, this description will allow us to define a set of convolutional code parameters which will be used throughout this thesis. A convolutional encoder is shown schematically in Fig. 1.1. Information symbols from a q-letter alphabet are shifted serially into a (K+1)-stage shift register. We have taken the length of the shift register, often called the constraint length of the code, to be K+1 instead of K; this notational change simplifies the later algebra. In order to make each information symbol a member of the finite field GF(q), q is restricted to be an integer power of a prime. After each information register shift, V channel symbols (phase 1 through phase V) are generated in parallel. These parallel channel symbols are commutated, added to a randomly selected sequence $\underline{r}$ and transmitted through a discrete memoryless channel. This random sequence can be omitted in most circumstances but

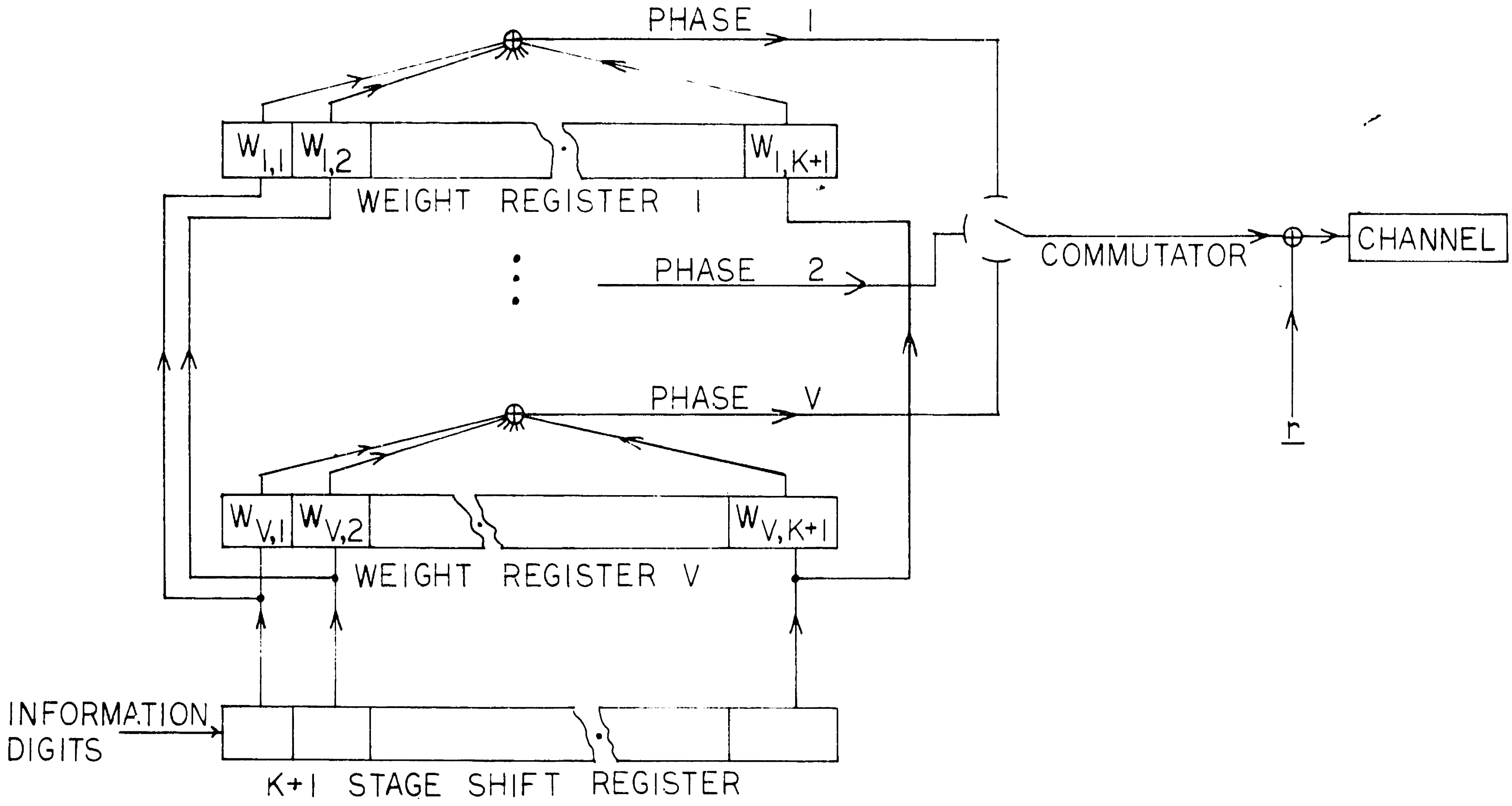

FIG. 1.1 CONVOLUTICNAL ENCODER

simplifies the analysis. Each of the V channel symbols is a weighted sum of the K+1 information symbols stored in the shift register plus the appropriate member of the sequence $\underline{r}$. All weights and elements of $\underline{r}$ are selected from GF(q) and the mathematical operations in the encoder are performed in GF(q). After the V channel symbols are generated, the information register is shifted to bring in the next information symbol and another V channel symbols are generated. Let $t_{v,d}$ be the phase v channel symbol generated immediately after the d^{th} information symbol i_d enters the encoder. Then,

$$t_{v,d} = \sum_{b=1}^{K+1} w_{v,b} i_{d+1-b} + r_{v,d} \qquad 1 \leq v \leq V \qquad (1.1)$$

where $w_{v,b}$ is the weight attached to the information symbol in the b^{th} shift register stage in determining the phase v channel symbol, and $r_{v,d}$ is the appropriate member of $\underline{r}$.

One of the most difficult problems in coding theory is to find a decoder that is simple enough to be implemented for codes that are complex enough to give a low probability of error. Massey (1963) has presented a simple threshold decoding algorithm which provides a good decoder for some simple but useful convolutional codes. Unfortunately, threshold decoding cannot be applied to the more powerful convolutional codes necessary to achieve good performance on channels with high noise levels. Despite its limitations, threshold decoding is used in some current communications systems because it provides an extremely efficient method of decoding some simple convolutional codes which are suitable for many less noisy channels. Sequential decoding, invented by Wozencraft (1957) is a more powerful decoding algorithm for convolutional codes. Sequential decoding is applicable to all convolutional codes and works at data rates much nearer channel capacity than threshold decoding. These advantages of sequential decoding are bought at the cost of a more complicated decoding algorithm.

An important subclass of convolutional codes is the family of convolutional codes in which one of the transmitted symbols is the information symbol that most recently entered the encoder plus the appropriate member of the random sequence $\underline{r}$ (we assume that $\underline{r}$ is known at the decoder). Such codes are called systematic convolutional codes. Let us assume that the phase 1 channel symbol is the systematic channel symbol. Thus for a systematic convolutional code

$$t_{1,d} = i_d + r_{1,d} \tag{1.2}$$

and $t_{2,d}$ through $t_{V,d}$,the parity symbols, are generated according to Eq. (1.1). Systematic convolutional codes are of both theoretical and practical interest for several reasons. First, systematic convolutional codes are free from "noiseless error propagation" as demonstrated by Massey and Sain (1967); however, many nonsystematic convolutional codes exhibit this type of error propagation. In noiseless error propagation, two or more information sequences differing in infinitely many information symbols produce channel sequences differing in only finitely many channel symbols. Such nearly identical channel sequences are impossible for the systematic convolutional code because the phase 1 channel symbol must differ whenever corresponding information symbols differ. Second, most easily implemented decoding algorithms for convolutional codes work well only if past decoding decisions have been correct. In the event of a decoder failure, some reasonable estimate of the transmitted information may be made simply by using the received phase 1 channel symbols of a systematic convolutional code. Third, in large communications systems where both inexpensive terminals and expensive highly reliable terminals are required, a systematic convolutional code may be used throughout. In such a system, inexpensive terminals would look at just the received systematic channel symbols while expensive terminals would look at the whole convolutional code with a good decoder. Moreover, such a system with a systematic convolutional code would be compatible with equipment which was built before the error-correcting code was added.

The class of systematic convolutional codes can be generalized into the class of multiple generator length convolutional codes. In the systematic code, $w_{1,1}=1$ and $w_{1,2}$ through $w_{1,K+1}$ all equal zero. These zero weights indicate that the contents of the second through $(K+1)^{th}$ stages of the encoder shift register cannot affect the systematic channel symbol. Suppose now that the communications system designer wishes to restrict the K+1 encoder weights $w_{2,1}$ through $w_{2,K+1}$ so that only th first k_2+1 of these weights may be non-zero. We shall denote this case as the case in which the second generator G_2 has length k_2+1. Likewise the communications system designer might wish to restrict the length of G_v to be k_v+1. The integer k_v may assume any value between 0 and K. If k_v were chosen greater than K, the phase v channel symbol would depend on information symbols which had passed out of the encoder shift register and out of the encoder's memory. Although the k_v's may be selected arbitrarily, there is no loss of generality if we number the generators such that $k_1 \leq k_2 \leq \ldots \leq k_V$. Multiple generator length convolutional codes were first suggested by Dr. K. L. Jordan (1966) of M.I.T. Lincoln Laboratory. Jordan's suggested use for the multiple generator length convolutional code consists of using a systematic code ($k_1=0$) with a short phase 2 generator, and a long phase 3 generator. With this code, the receiver could use the received systematic symbols to make some reasonable estimate of the transmitted data after a decoder failure. Once the receiver had made reasonable guesses about k_2 consecutive information symbols, it could also use the phase 2 received symbols in decoding. Finally, after the decoder had hypothesized k_3 consecutive information symbols, it could also use received phase 3 channel symbols. Such a restarting procedure can obviously be extended to V generators. Additional uses of the multiple generator length convolutional code also suggest themselves. If the code were designed with a systematic generator, a short generator and two long generators (e.g. $k_3=k_4=2k_2$), simple inexpensive terminals could just look at the phase 1 and phase 2 symbols. Such a hybrid scheme is useful only if the G_2 generator permits some simple form of decoding; e.g. threshold decoding.

The V channel symbols produced per shift of the encoder register depend only upon the encoder weights, the additive sequence $\underline{r}$, and the K+1 information digits that most recently entered the encoder. The initial state of the encoder shift register is assumed to be known at the decoder and is generally the all-zero state. This dependence upon a series of past events suggests a tree-like structure with q new alternatives (branches) arising at each shift of the encoder register. Figure 1.2 illustrates the beginning portion of the tree associated with some convolutional code. The symbols on each branch of the tree in figure 1.2 are the channel symbols which would be transmitted if the encoder were encoding the message represented by that particular path through the tree. The convolutional code used to generate the tree in Fig. 1.2 is a systematic convolutional code with V=3, $k_2=k_3=3$, q=2,

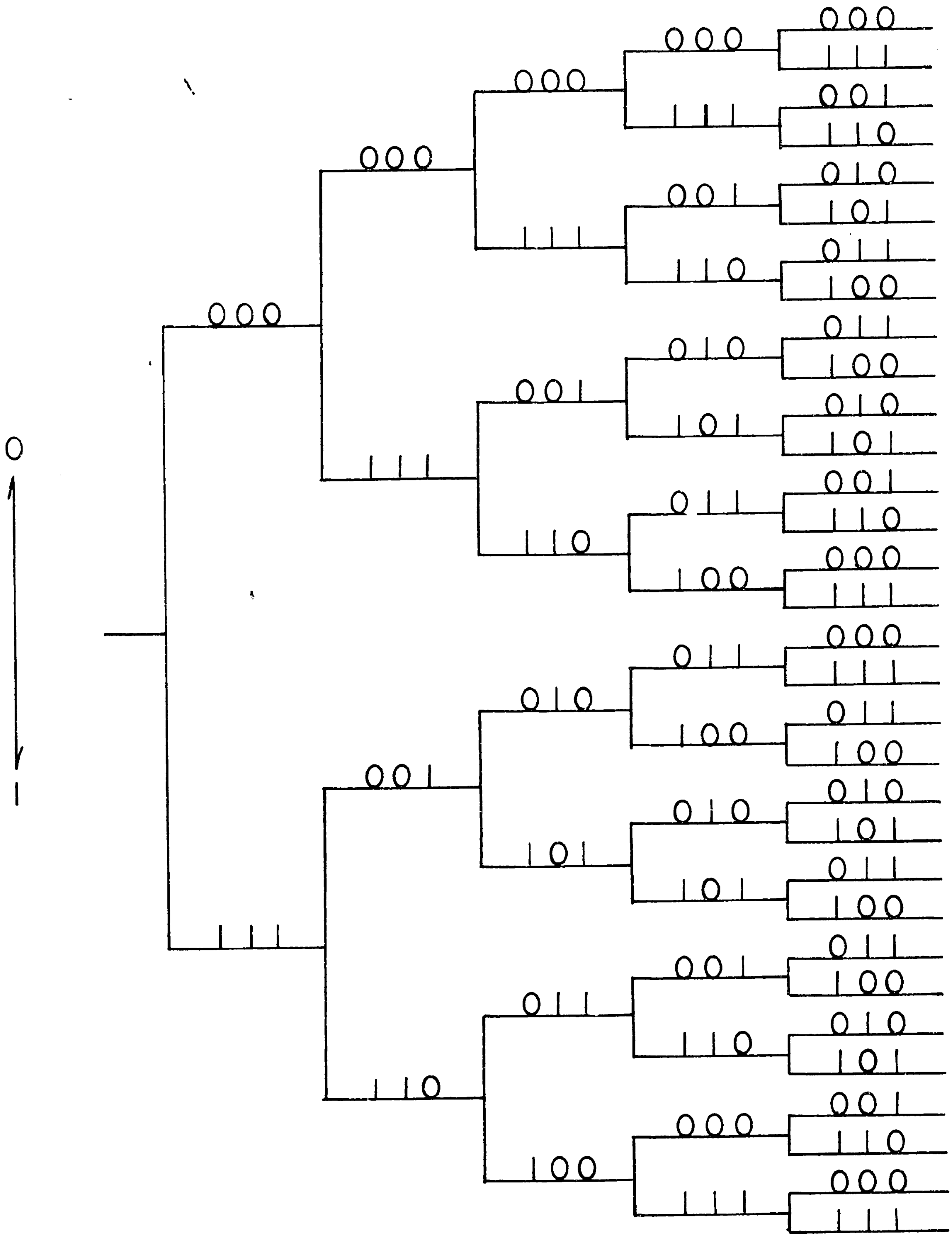

FIG. 1.2 BEGINNING PORTION OF TREE

$\underline{r}=0$, $w_{3,3}=w_{2,2}=0$ and $w_{2,1}=w_{2,3}=w_{2,4}=w_{3,1}=w_{3,2}=w_{3,4}=1$ In Figure 1.2, an upward branch represents the event of a binary zero entering the encoder.

This thesis examines both optimum and Fano-type sequential decoding of multiple generator length convolutional codes. In chapter II, we derive a lower bound to error probability for any convolutional code. This bound is of the form,

$$P(E) \geq \exp -K^*V \left[E_L(R) - o_3(K^*)\right] \qquad (1.3)$$

where

$$K^*V = k_1+k_2+\ldots+k_V$$

and $o_3(K^*)$ is a function of K^* which goes to zero as K^* approaches infinity. This lower bound is valid for all decoding algorithms and all convolutional codes. The lower bound error exponent $E_L(R)$ is obtained by a geometric operation on a lower bound error exponent for block codes $e_b(r)$. This geometric procedure may be used to obtain a valid $E_L(R)$ from any $e_b(r)$. Chapter III considers upper-bounds to error probability for multiple generator length convolutional codes with optimum decoding. These optimum decoding upper-bounds on error probability indicate the capability of the convolutional codes themselves. Such optimum decoder results are useful as a reference standard when analyzing practical but suboptimum decoders. These upper-bounds are derived by upper-bounding the average probability of error for a large collection or ensemble of codes. The probability of error for some code in the ensemble is less than or equal to the ensemble average probability of error. Thus, these ensemble average upper-bounds on error probability are also upper-bounds to the probability of error for some code in the ensemble. For equal generator length convolutional codes these ensemble average upper-bounds on error probability take the form

$$\overline{P(E)} \leq \text{const} \quad \exp -KVE_U(R). \qquad (1.4)$$

In chapter III we find that the error bound in inequality (1.4) is still valid for multiple generator length convolutional codes if KV is replaced by the more general term K*V (the sum of the generator lengths) provided that either (i) all k_v except k_1 equal K or (ii) that if V≥3, k_2 is "not too short." The words "not too short" in case ii imply an asymptotic rather than absolute convergence. Finally in chapter IV, we consider using the Fano (1963) sequential decoding algorithm for multiple generator length convolutional codes. We find that sequential decoding gives the same upper bound error exponent as optimum decoding for equal generator length convolutional codes; however, sequential decoding does *not* give the optimum error exponent for systematic convolutional codes and for multiple generator length convolutional codes. This exponential non-optimality may be eliminated by increasing a decoder parameter called the bias B. Increasing the bias from its usual value B=R, to a bias large enough to eliminate this non-optimality causes a substantial increase in the decoder computation. Chapter IV also analyzes this increase in computation. Forney's simulations (1968) demonstrate both these effects. Finally, Chapter V discusses the implications of these results and makes suggestions for further research.

A mathematical dilemma arises in discussing optimum decoders for convolutional codes. The dilemma is that the decoder must make a decision involving some signal sequence that may never end. This dilemma can be circumvented by requiring that information digits be encoded in sequences of at most L information symbols.

Once L consecutive information symbols have been shifted into the encoder, K information zeros are shifted into the encoder before any additional message-dependent information symbols are allowed to enter the encoder. This terminating sequence of K information zeros returns the encoder to its initial state just before the next sequence of L information symbols begins to enter the encoder. This return to the initial state makes the encoding of the next sequence of L information symbols appear to be just like the encoding of those symbols in fresh encoder with an all-zero initial state. With periodic resetting, the convolutional encoder may be thought of as a block encoder that generates a sequence of (L+K)V channel symbols to encode a message of L information symbols. Analytically, resetting allows a straight forward definition of optimum decoding, and hence allows us to express the error correcting capability of convolutional codes. In practice, resetting allows the receiver to restart some practical, but suboptimum, decoder that has been confused by a particularly noisy sequence of received symbols. These suboptimum decoders may be restarted because each "block" of (L+K)V channel symbols is decoded independently. Implementing such a resetting procedure decreases the true data rate from its nominal value of

$$R = \frac{\ln(q)}{V} \tag{1.5}$$

to R(L/L+K). Normally the value of L is two or three orders of magnitude greater than K and the small rate loss is ignored.

II Lower Bound on the Probability of Error

Techniques recently developed by Jacobs and Berlekamp (1967), Viterbi (1967), and Forney (1967) may be generalized to lower-bound the probability of error for multiple generator length convolutional codes. Suppose that L is very large and that the decoder is given the first L-L" information symbols. The decoder must then correctly decode the last L" information symbols if no communication error is to occur. There are many decoding rules which the decoder given the first L-L" information symbols could adopt. Since the first L-L" information symbols are already known to the decoder, each of these rules for the assisted decoder produces some estimate of the last L" information symbols. There is some probability of error for each of these assisted decoder decision rules. The optimum (lowest probability of error) decoding rule for the aided decoder has a probability of error which we denote as $P(E_{L''}/I_{L-L''})$. Note that $P(E_{L''}/I_{L-L''})$ is not a conditional probability but an average over all sequences of L-L" information symbols. Let P(E) denote the probability of error for the optimum unaided decoder (the maximum likelihood decoder) that is not given the first L-L" information symbols. Then,

$$P(E) \geq P(E_{L''}/I_{L-L''}) \qquad (2.1)$$

because the decision rule for the optimum unaided decoder was one of the possible decision rules for the aided decoder, and $P(E_{L''}/I_{L-L''})$ is the minimum probability of error for all possible aided decoder decision rules. Inequality (2.1) may be interpreted as a mathematical statement of an intuitive notion. Namely, the aided decoder can do no worse than the unaided decoder because the aided decoder can always ignore the information symbols it has been given and imitate the unaided decoder.

The channel symbol sequence cannot depend upon any of the last L" information symbols until the first of these last L" information symbols enters the encoder. Since the channel is memoryless, the aided decoder need only consider those received symbols that depend on the last L" information symbols. For any given choice of the first L-L" information symbols, the encoder with resetting defines L"V channel symbols while the last L" information symbols are entering the encoder. During resynchronization, all phase v channel symbols must be the same for any message after the first k_v information zeros in the resynchronizing sequence have entered the encoder. These phase v channel symbols which must be the same simply reflect the fact that the information symbols in L" have been shifted so far down the register that they are no longer within the first k_v+1 stages. For a memoryless channel, these channel symbols which must identical for all messages need not be considered at the decoder. Thus, during resynchronization, the encoder defines $K^*V=k_1+k_2+...+k_V$ channel symbols which are truly dependent upon the last L" information symbols. Hence there are a total of

$$N = (L''+K^*)V$$

channel symbols dependent upon the last L" information symbols. There are

$$M = q^{L''}$$

choices for the last L" information symbols. Since the first L-L" information symbols are given the aided decoder, the aided decoder is just decoding one of M possible messages which was encoded in a sequence of N channel symbols. For any choice of the first L-L" information symbols, the convolutional encoder's assignment of a sequence of N channel symbols to each possible sequence for the last L" information symbols is just the generation of some block code. This block code transmits one of M messages by a sequence of N channel symbols. The block code produced by the convolutional encoder can have no lower probability of error than the best block code which transmits one of M messages with a sequence of N channel symbols. Using inequality (2.1), we have now argued that

$$P(E) \geq P(E_{L''}/I_{L-L''}) \geq P(E \text{ for best code using } N \text{ symbols to transmit one of } M \text{ messages}). \tag{2.2}$$

Shannon, Gallager, and Berlekamp (1967) have shown that the probability of error for the best possible code using N channel symbols to transmit one of M messages over a discrete memoryless channel may be lower-bounded as

$$P(E \text{ for best code using } N \text{ symbols to transmit one of } M \text{ messages}) \geq \exp -N\left[e_b(r)-o(N)\right], \tag{2.3}$$

where o(N) is a function that approaches zero as N approaches infinity, and

$$r = \frac{\ln(M)}{N} \tag{2.4}$$

We shall leave $e_b(r)$ temporarily unspecified, in order to show that subsequent manipulations are not dependent upon a specific form of $e_b(r)$. Recalling that K* was defined such that

$$K^*V = k_1+k_2+k_3+\ldots+k_V$$

and defining g such that

$$L'' = gK^*,$$

we may combine Eqs. (2.2) and (2.3) to show that

$$P(E) \geq \exp -N\left[e_b(r) \ -o(N)\right] =$$

$$\exp -K^*V\left[(g+1)e_b(r) \ -o_1(K^*)\right] .$$

where $o_1(K^*)$ is a function of K* which approaches zero as K* approaches infinity,

$$r = \frac{\ln(M)}{N} = \frac{q}{g+1} \quad \frac{\ln(q)}{V} = \frac{q}{g+1} \ R,$$

and R is the nominal data rate of the convolutional code as defined in Eq. (1.5).

We may write

$$P(E) \geq \exp -K^*V \left[E_g(R) - o_1(K^*)\right] \qquad (2.5)$$

if we define $E_g(R)$ such that

$$E_g(R) = (g+1)e_b(\tfrac{g}{g+1} R). \qquad (2.6)$$

Up to this point, we have implicitly assumed that g is a multiple of $1/K^*$; however, in the asymptotic case of large K^*, the difference between any non-negative value of g and the nearest multiple of $1/K^*$ may be represented as a function $o_2(K^*)$, which approaches zero as K^* approaches infinity. Thus, Eqs. (2.5) and (2.6) are valid for all non-negative g. In particular, inequality (2.5) must hold for that value of g which gives the largest probability of error; that is, inequality (2.5) must hold for the value of g that minimizes $E_g(R)$. Thus, we may lower-bound the probability of error for a multiple generator length convolutional code as

$$P(E) \geq \exp -K^*V \left[E_L(R) - o_3(K^*)\right] \qquad (2.7)$$

where

$$E_L(R) = \inf_{g>0} \left[(g+1)e_b(\frac{1}{g+1}R)\right]. \qquad (2.8)$$

Forney (1967) has developed a geometric method of finding $E_L(R)$ from any lower-bound block code exponent $e_b(r)$. Figure 2.1 shows a typical $e_b(r)$ curve. Consider the points R_o and $\frac{1}{g+1}R_o$ on the rate axis. The straight line connecting the point R_o on the rate axis and $e_b(\frac{1}{g+1}R_o)$ on the $e_b(r)$ curve intersects the $E(R)$ axis at the point $(g+1)e_b(\frac{1}{g+1}R_o)$. Changing the value of g simply moves the point $\frac{1}{g+1}R_o$ along the rate axis between 0 and R_o. Thus, $E_L(R_o)$ is the lowest $E(R)$ intercept of any straight line passing through the rate axis at R_o and touching the curve $e_b(r)$. If the $e_b(r)$ curve is smooth, $E_L(R_o)$ is the $E(R)$ axis intercept of the straight line from R_o which is tangent to the $e_b(r)$ curve. Repeating this construction for each possible R_o, we obtain the $E_L(R)$ curve from the $e_b(r)$ curve. In figure 2.1, this construction has been completed to show $E_L(R)$.

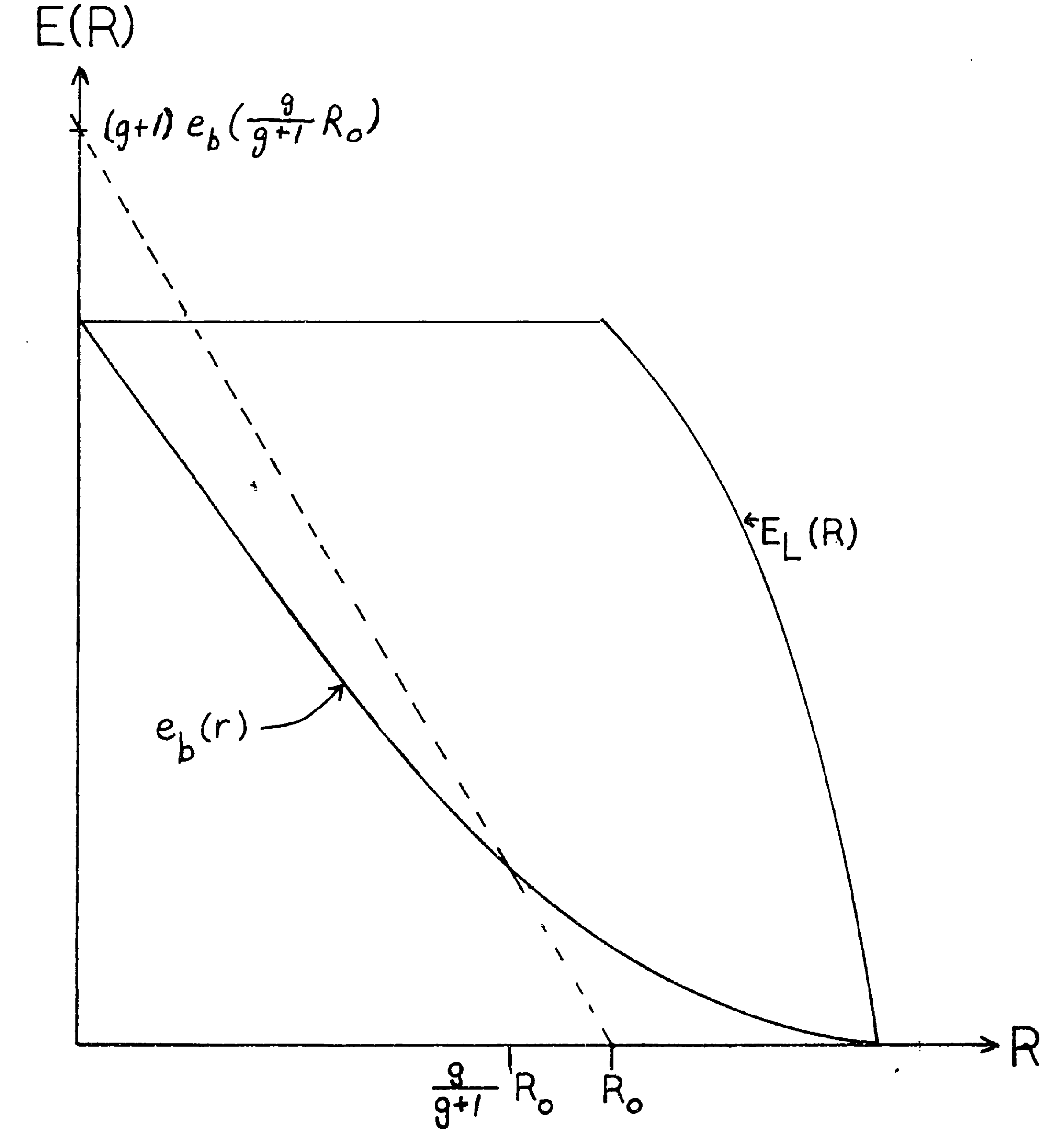

FIG. 2.1 CONSTRUCTION OF $E_L(R)$

III Upperbound on the Probability of Error for Multiple Generator Length Convolutional Codes with Optimum Decoding.

3.1 Introduction

A measure of performance for any code is the probability of erroneous communication with the optimum decoder. Calculating the probability of error for any specific code is so complicated that it is virtually impossible to find the best code in a set of codes. This immense problem of detailed code selection may be avoided by finding the average probability of error for a very large collection or ensemble of codes. This ensemble of codes contains every possible code that could ever be used for a given design technique. One ensemble of multiple generator length convolutional codes might be the collection of all multiple generator length convolutional codes with given k_1, k_2, ... k_V. Unfortunately, there are both theoretical and practical problems with this ensemble of "fixed-generator" convolutional codes. These problems can be avoided by using the ensemble of convolutional codes with a fixed k_1, ... k_V in which $w_{1,1}=1$ and all remaining nontrivial encoder weights are reselected after each shift of the information storage register. Each new weight in the encoder is selected from GF(q), with all weights being equally probable. This randomly reselected weights ensemble of multiple generator length convolutional codes is analogous to the ensembles of convolutional codes used in all "random-coding" upper bounds on the probability of error.

Under the assumption that all messages are equally likely, the optimum decoder for any code is the maximum-likelihood decoder which operates on the entire received sequence. For the periodically reset convolutional code, the maximum-likelihood decoder considers $\underline{Y}$ the entire sequence of (L+K)V received symbols. Let $\underline{X}_m$ denote the channel sequence the encoder assigns to the message m. The maximum-likelihood decoder estimates that message $\hat{m}$ was transmitted where $\hat{m}$ is the value of m which maximizes the conditional probability $P(\underline{Y}/\underline{X}_m)$. Erroneous communication results if the decoder selects any message sequence m' that is not identical to the encoded message sequence m_0. There are two different probabilities of error which may be of interest. First, one may be interested in the probability that some particular information symbol was decoded incorrectly.

Second, one might be interested in the probability that any of the L information symbols was incorrectly decoded.

The structure of the convolutional encoder is such that the transmitted sequences for two messages must be identical during those time intervals in which the contents of the encoder shift register are identical for the two messages. For example, let m_1 be an incorrect message differing from the correct message m_0 only in the first information symbol. The corresponding channel sequences $\underline{X}_{m_1}$ and $\underline{X}_{m_0}$ must be identical after the first information symbol leaves the encoder. Let us consider a multiple generator length convolutional code with generator lengths $k_1, k_2, \ldots k_V$. By definition, only the k_v+1 information symbols which most recently entered the encoder are involved in the determination of the phase v channel symbol. Thus, the channel sequences $\underline{X}_{m_0}$ and $\underline{X}_{m_1}$ must be identical for all but the first k_1+1 phase 1 channel symbols, the first k_2+1 phase 2 channel symbols,... and the first k_V+1 phase V channel symbols. Thus, $\underline{X}_{m_0}$ and $\underline{X}_{m_1}$ must be identical in all but $V+k_1+k_2+\ldots+k_V=V(1+K^*)$ channel symbols. This matter of identical channel symbols for different message sequences may be generalized as the concept of diverging and merging sequences. Two information sequences are merged for a specific phase v channel symbol if the k_v+1 information symbols most recently entering the encoder are the same for both messages. If two message sequences are not merged for a specific channel symbol, they are said to be diverged for that channel symbol. Thus, two information sequences are merged at a specific channel symbol only if that channel symbol must be identical for both messages for any code with the same set of k_v's.

The number and location of channel symbols at which a given incorrect message sequence is diverged from the correct message may be found with the aid of diagrams such as that in Figure 3.1.1. The n^{th} division of the box labeled "information different?" represents the n^{th} information symbol in the message sequence. An x placed in a division of the "information different?" box indicates that the corresponding symbol of the incorrect message m' differs from its counterpart in the correct message m_0. The column labeled "channel symbol phase" lists the phase of each of the V channel symbols generated after an encoder shift. Merged channel symbols are represented by the unshaded regions of Figure 3.1.1 and diverged channel symbols are represented by the shaded regions. The rule for determining shaded regions in a divergence diagram is that the area representing

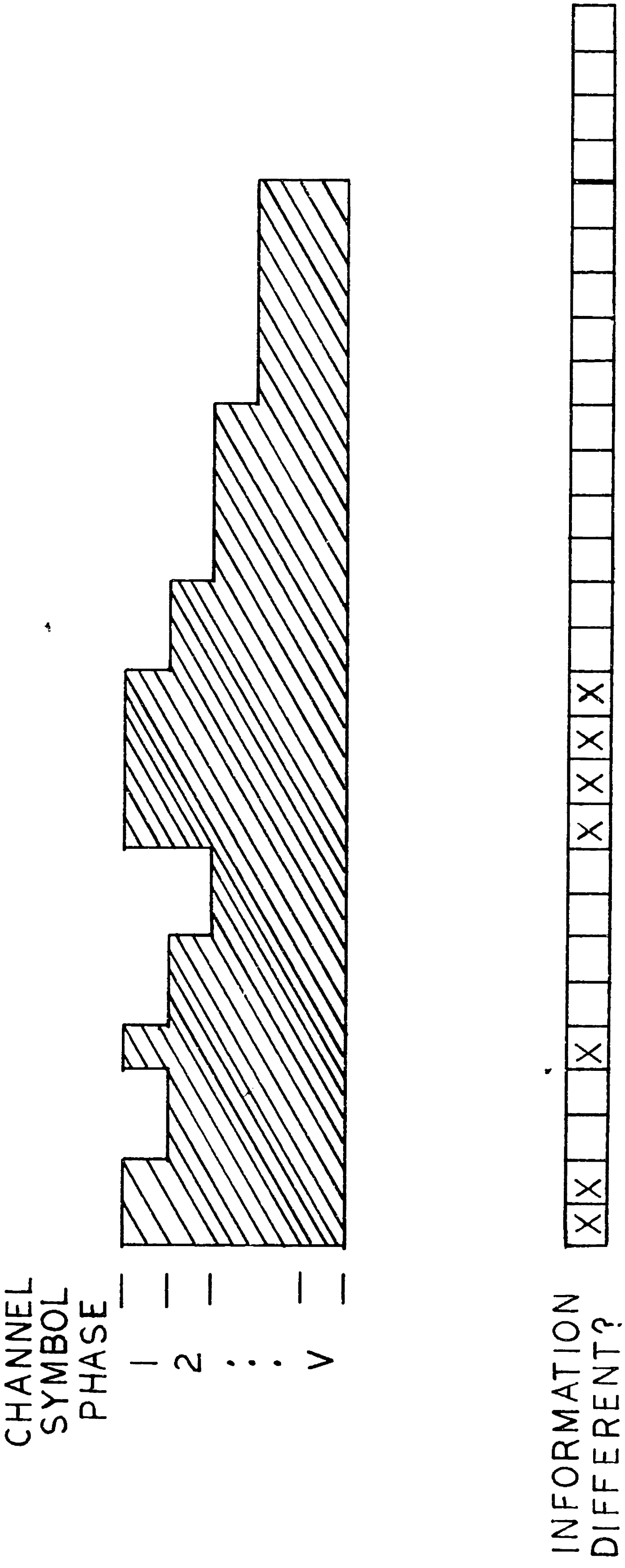

FIG. 3.1.1 DIVERGENCE DIAGRAM

a phase v channel symbol is shaded if and only if there is an x either in the division of the "information different?" box immediately below that area or in one or more of the k_v divisions of the "information different?" box immediately to the left of that division.

The maximum-likelihood decoder decides that message $\hat{m}$ was transmitted only if $\hat{m}$ is the value of m which maximizes the conditional probability $P(\underline{Y}/\underline{X}_m)$. Hence a decoder error can occur only if

$$P(Y/X_{m'}) \geq P(\underline{Y}/\underline{X}_{m_0}) \qquad (3.1.1)$$

for any $m' \neq m_0$. The equality in (3.1.1) is used to denote the possibility that a decoder error will occur if m' and m_0 have equal a posteriori probabilities. Dividing both sides of inequality (3.1.1) by $P(\underline{Y}/\underline{X}_{m_0})$, we find that an error can occur only if

$$\frac{P(Y/X_{m'})}{P(Y/X_{m_0})} \geq 1 \qquad (3.1.2)$$

for any $m' \neq m_0$. Since the channel is assumed to be memoryless, each conditional probability in the likelihood ratio is the product of individual channel symbol transition probabilities. In general each particular m' is merged with m_0 for some channel symbols. The transmitted sequences $\underline{X}_{m_0}$ and $\underline{X}_{m'}$ are identical at these merged channel symbols. Hence the individual channel symbol transition probabilities $P(y_i/x_{m'i})$ and $P(y_i/x_{m_0 i})$ are identical for these merged channel symbols. The numerical value of the likelihood ratio in (3.1.2) is unchanged if these common factors are cancelled in the numerator and denominator. Thus in determining whether a specific m' may be decoded instead of m_0, we need only consider those received channel symbols at which m' is diverged from m_0.

If a diagram su as that in Figure 3.1.1 were drawn for an entire incorrect message m', there would be L+K encoder shifts represented. In general there would be several, say h, disjoint shaded regions in the diagram. Each of these disjoint shaded regions would represent divergence of the incorrect message from the correct message and subsequent remerging with it. We may view each disjoint shaded region as arising out of some subsequence of m' which is divergent from m_0 at exactly those channel symbols involved in that particular shaded region. Hence, any incorrect message sequence m' may be viewed as a number of divergent information subsequences joined together by information subsequences identical to the corresponding parts of m_0. Because the channel is memoryless, the likelihood ratio in inequality (3.1.2) is just the product of the likelihood ratios calculated for each of the h divergent information subsequences in m'. Furthermore, we now show that the incorrect message m' can be decoded only if the likelihood ratio for each divergent subsequence of m' is greater than or equal to one. Suppose that the i^{th} $(i \leq h)$ divergent subsequence of m' has a likelihood ratio that is less than one. Suppose there is a message m* with the same over all likelihood ratio as m' except that the likelihood ratio for the i^{th} divergent subsequence is replaced by one. Then m* has a larger likelihood ratio than m' and m* will be decoded in preference to m'. But the incorrect message that is identical to m' in all but the i^{th} divergent subsequence and identical to m_0 in that subsequence is just such an m*. Thus an incorrect message m' cannot be decoded unless the likelihood ratio for each divergent subsequence is greater than or equal to one.

Each divergent subsequence of any incorrect message sequence m' (each continuous shaded region of the divergence diagram for m') may be characterized by a number b such that m' and m_0 are phase V diverged for exactly b+K+1 encoder shifts. Since the phase V generator is the longest generator $(k_V \geq k_{V-1} \geq \ldots k_1)$ and $K=k_V$, the total length of the divergent region will be b+K+1 information symbols. In order for complete remerging to occur after b+K+1 encoder shifts, the last K information symbols in the divergent subsequence must be identical to the corresponding symbols of m_0. Since each incorrect message has a divergence diagram, we may classify incorrect message sequences by their divergence diagram patterns. In particular, we may enumerate all incorrect messages by enumerating all divergence diagrams.

3.2 A Basic Lemma

In this section, we will derive a basic lemma upper-bounding the ensemble average probability of decoding an incorrect information subsequence with a divergence pattern from a certain family of divergence patterns. This family of divergence patterns is rather hard to motivate and the reader will have to be patient with a good deal of algebra before the desired result is reached. Quite a bit of complexity arises out of the need to consider systematic convolutional codes in which $k_1=0$ and $w_{1,1}=1$. The family of divergent information subsequences we wish to consider is the set of all divergent subsequences which are fully merged at the $(j-1)^{th}$ encoder shift, diverge at the j^{th} encoder shift, remain at least partially diverged for exactly $b+K+1$ encoder shifts and have the same pattern of diverged phase 2 through phase V channel symbols. Figure 3.2.1 shows several members of this family of divergence diagrams. Let us call this family of incorrect subsequences M_{jpb} where p is an index indicating the pattern of diverged phase 2 through phase V channel symbols.

Let $\overline{P(E_{jpb})}$ denote the ensemble average probability of decoding some incorrect message subsequence in M_{jpb}

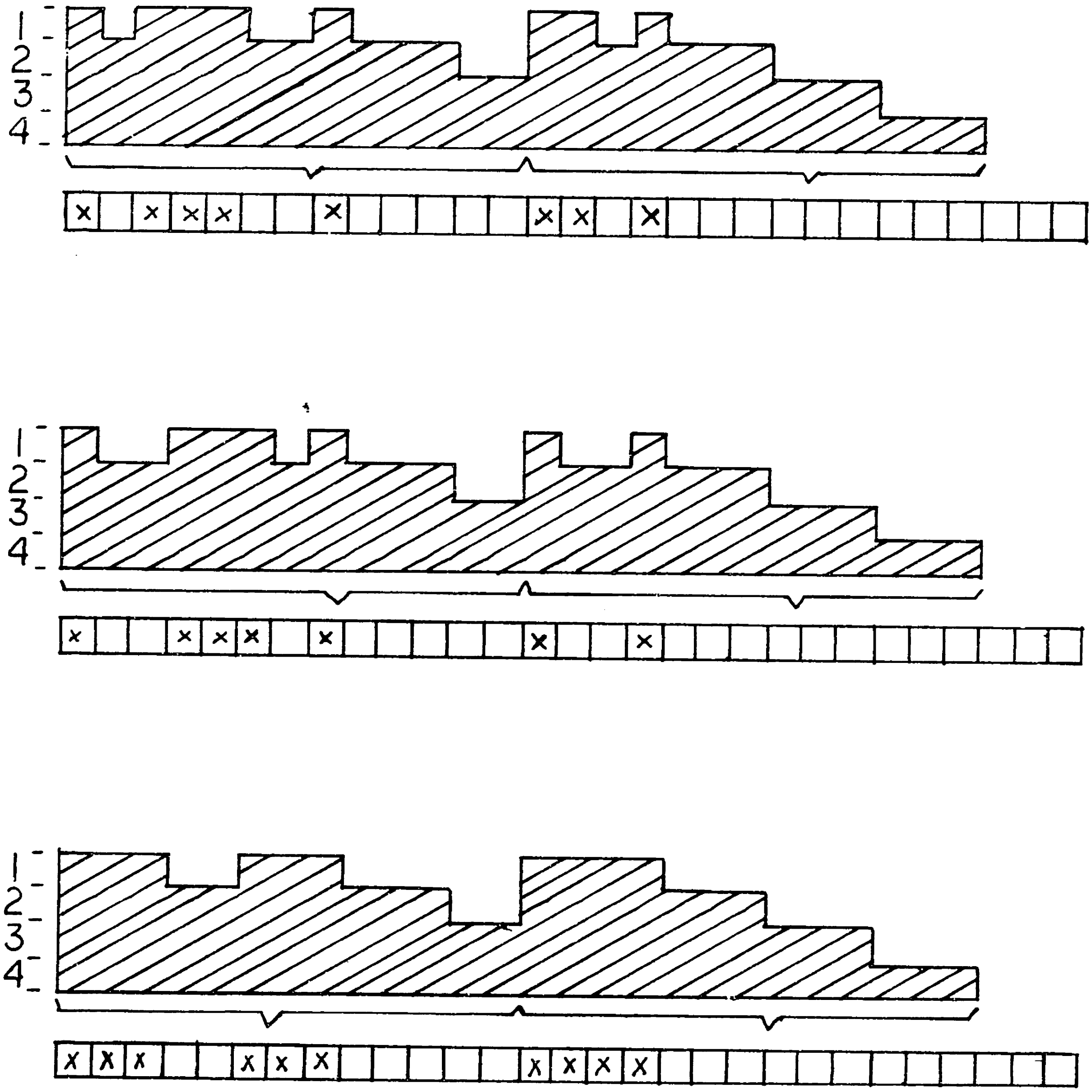

Figure 3.2.1 Three divergence diagrams with the same pattern of diverged phase 2 through phase V channel symbols.

instead of the corresponding subsequence of m_0. We may upper-bound $\overline{P(E_{jpb})}$ by using techniques first developed by Gallager (1965) for block codes and later extended by the author (1968) to systematic convolutional codes. The ensemble of multiple generator length convolutional codes is the set of all convolutional codes with fixed $k_1, k_2, \ldots k_V$ in which $w_{1,1}=1$ and all other non-trivial encoder weights are reselected after each shift of the encoder shift register. The only encoder weights considered as trivial are those required to be zero by the k_v+1 length of the phase v generator. The randomly selected weights are from the finite field GF(q) with all values being equally probable for each weight subject to reselection.

Since we are dealing with the set of all incorrect messages with a fixed pattern p of diverged phase 2 through phase V channel symbols, let us examine the possible patterns p. The fixed pattern p of diverged phase 2 through phase V channel symbols will have several, say D_2, runs of diverged phase 2 channel symbols. Each of these runs of diverged phase 2 channel symbols must be separated by one or more merged phase 2 channel symbols (but not by any merged phase V channel symbols since the pattern must be continuous). A study of the divergence-remerger mechanism and the requirement that $k_1 \leq k_2 \leq k_3 \leq \ldots \leq k_V$ shows that if the phase v channel symbol is merged with m_0 then the corresponding phase j channel symbol is also merged for all $j \leq v$. Likewise, if the phase v channel symbol is diverged from m_0 at any encoder shift, the corresponding phase j channel symbol is diverged for all $j \geq v$. If the phase 2 channel symbols are merged and a symbol of m' differing from the corresponding symbol of m_0 were about to enter the encoder, there must be a phase 1 divergence and the phase 2 through phase V channel symbols must also diverge if they are not already diverged from m_0. Moreover, a phase v merger cannot occur until a phase v-1 merger occurs. Thus, the "skyline" in the divergence pattern p may slowly fall off as one moves to the right but must always rise as high as possible whenever it rises at all.

An examination of the information symbols in some m" subsequence in M_{jpb} will aid in the proof of the lemma. As discussed above, let us assume that there are D_2 distinct runs of diverged phase 2 channel symbols. If the desired pattern of diverged phase 2 channel symbols is to occur, the information symbols of m" must satisfy four conditions. These conditions must hold for each distinct run of diverged phase 2 channel symbols and are most easily stated if we assume that a run of diverged phase 2 channel symbols is $c+k_2+1$ channel symbols long. First, the symbol of m" corresponding to the first symbol of this run of diverged phase 2 channel symbols must differ from the corresponding symbol of m_0. Second, the information symbols of m" corresponding to the second through c^{th} symbols of this run are arbitrary except for the restriction that no consecutive k_2+1 information symbols be identical to the corresponding symbols of m_0. Third, the information symbol of m" corresponding to the $(c+1)^{th}$ symbol of the run of diverged phase 2 channel symbols must differ from the corresponding symbol of m_0. Fourth, all subsequent symbols of m" must be identical to the corresponding symbol of m_0 until the start of the next run of diverged phase 2 channel symbols. This latter run of matching information symbols must be at least k_2+1 symbols long in order for there to be a phase 2 merger to terminate the run of diverged phase 2 channel symbols. The first condition is necessary if the run of diverged phase 2 channel symbols is to start at the desired place. The second condition assures that the run of diverged phase 2 channel symbols does not end before the desired spot. The third and fourth conditions are necessary if the run of diverged phase 2 channel symbols is to end at the right place and if there are to be no phase 2 divergences before the start of the next run.

What implications do the above conditions on m" have on the sequence of channel symbols? These implications are best found if we continue to consider the run of $c+k_2+1$ diverged phase 2 channel symbols. The third and fourth conditions require that the phase v channel symbols merge k_v-k_2 steps after the end of the run of diverged phase 2 channel symbols unless another run of diverged phase 2 channel symbols starts at or before that step. Thus, the lengths of the runs of diverged phase 2 channel symbols and the spacings between these runs completely determine the pattern p for a fixed set of k_v's. The third condition and the random reselection of $w_{1,2}$ through w_{1,k_1+1} imply that the k_1 phase 1 channel symbols corresponding to the $(c+2)^{th}$ through $(c+1+k_1)^{th}$ symbols of the run are equally likely to be any sequence of k_1 q-ary symbols independent of $\underline{X}_{m_0}$ and m_0. Furthermore, the fourth condition implies that all phase 1 channel symbols after the $(c+1+k_1)^{th}$ symbol of the run are merged until the start of the next run of diverged phase 2 channel symbols. Thus, a run of $c+k_2+1$ consecutive diverged phase 2 channel symbols implies at most $c+k_1+1$ diverged phase 1 channel symbols and (from above) a run of $c+1$ information symbols in m" which need not be identical to the corresponding symbols of m_0. Because $w_{1,1}=1$, the $c+1$ phase 1 channel symbols corresponding to the first $c+1$ symbols of the run are a one-to-one function of the $c+1$ information symbols which may differ from the corresponding symbols of m_0. That is, for each code (given sequence of encoder weights and fixed $\underline{r}$) there is exactly one subsequence of $c+1$ phase 1 channel symbols for each subsequence of $c+1$ information symbols differing from the corresponding subsequence of m_0.

Now let us suppose that the pattern p has D_2 distinct runs of diverged phase 2 channel symbols and N_{pb2} diverged phase 2 channel symbols in all. We may repeat the above argument for each of these runs. Thus, the pattern p has D_2k_1 phase 1 channel symbols which are selected statistically independently of $\underline{X}_{m_0}$ and m_0. Moreover, the pattern p has $N_{pb2}-D_2k_2$ phase 1 channel symbols which are a one-to-one map of the $N_{pb2}-D_2k_2$ symbols of m" which may differ from the corresponding symbols of m_0. As a check we note that we have accounted for $N_{pb2}-D_2(k_2-k_1)$ phase 1 channel symbols which is the maximum number of phase 1 channel symbols which may be diverged for any m" in M_{jpb}.

The reselection of encoder weights guarantees that over the ensemble of codes, each diverged phase 2 through phase V channel symbols is equally likely to be any q-ary symbol independent of $\underline{X}_{m_0}$ and m_0. We may combine the diverged phase 2 through phase V channel symbols with the D_2k_1 phase 1 channel symbols which are equally likely to be any q-ary sequence to form $X_{m"r}$. $X_{m"r}$ is the set of channel symbols which in the ensemble are equally likely to be any q-ary symbol independent of m_0 and $\underline{X}_{m_0}$ for any m" in M_{jpb}. The subscript r in the name $X_{m"r}$ indicates that the symbols in $X_{m"r}$ are randomly selected by the code independently of m_0 and $\underline{X}_{m_0}$. Likewise, we may define $X_{m"1}$ as the set of $N_{pb2}-D_2k_2$ channel symbols which are a one-to-one map of the $N_{pb2}-D_2k_2$ information symbols of m" which may differ from the corresponding symbols of the correct message m_0. Hence, $X_{m"r}$ and $X_{m"1}$ contain all the channel symbols at which any m" in M_{jpb} may be diverged from m_0. Thus, we need only consider the received channel symbols corresponding to $X_{m"r}$ and $X_{m"1}$ in determining whether any information subsequence m" in M_{jpb} may be decoded instead of the corresponding part of m_0. Notational problems will be simplified if we let Y_r denote the part of the received sequence $\underline{Y}$ corresponding to the symbols in $X_{m''r}$. Similarly, we may define Y_l, X_{m_0r} and X_{m_0l}.

We may use the random nature of the ensemble to derive an upper-bound on $\overline{P(E_{jpb}/Y_lY_rX_{m_0l}X_{m_0r}m_0)}$, the ensemble average probability of decoding some incorrect message subsequence in $M_{\overline{jpb}}$ given that m_0 was encoded as $\underline{X}_{m_0}$ and that $\underline{Y}$ was received. The maximum likelihood decoder can decode an incorrect message subsequence m" in M_{jpb} only if the code sequence for m" was selected such that

$$\frac{P(Y_lY_r/X_{m''l}X_{m''r})}{P(Y_lY_r/X_{m_0l}X_{m_0r})} \geq 1. \tag{3.2.1}$$

The structure of the encoder $(w_{l,l}=1)$ is such that the channel sequence selected for m" is not entirely independent of the channel sequence for m_0. Using a union bound to account for all m" in M_{jpb}, it follows that

$$\overline{P(E_{jpb}/Y_l Y_r X_{m_0 l} X_{m_0 r} m_0)} \leq \sum_{m'' \epsilon M_{jpb}} \sum P(X_{m'' l} X_{m'' r}/Y_l Y_r X_{m_0 l} X_{m_0 r} m_0) \qquad (3.2.2)$$

where the rightmost summation is over all $X_{m''l}$ and $X_{m''r}$ for which inequality (3.2.1) holds.

The rightmost summation (3.2.2) is simply the probability that the randomly selected code assigned an $X_{m''l} X_{m''r}$ leading to the decoding of m", for the given Y_l, Y_r, $X_{m_0 l}$, and $X_{m_0 r}$. Since the code is selected before encoding and transmission begin, the code words must be independent of the received sequence $\underline{Y}$. Thus,

$$P(X_{m''l} X_{m''r}/Y_l Y_r X_{m_0 l} X_{m_0 r} m_0) = P(X_{m''l} X_{m''r}/X_{m_0 l} X_{m_0 r} m_0).$$

Noting that whenever inequality (3.2.1) is satisfied,

$$P(X_{m''l} X_{m''r}/X_{m_0 l} X_{m_0 r} m_0) \leq P(X_{m''l} X_{m''r}/X_{m_0 l} X_{m_0 r} m_0) \times \left\{ \frac{P(Y_l Y_r/X_{m''l} X_{m''r})}{P(Y_l Y_r/X_{m_0 l} X_{m_0 r})} \right\}^s$$

for any s≥0. We may now upper bound the right hand side of inequality (3.2.2) by

$$\overline{P(E_{jpb}/Y_1 Y_r X_{m_0 1} X_{m_0 r} m_0)} \le \sum_{m'' \epsilon M_{jpb}} \sum_{\text{all } X_{m''1} X_{m''r}} P(X_{m''1} X_{m''r}/X_{m_0 1} X_{m_0 r} m_0) \times \left\{ \frac{P(Y_1 Y_r/X_{m''1} X_{m''r})}{P(Y_1 Y_r/X_{m_0 1} X_{m_0 r})} \right\}^s . \tag{3.2.3}$$

One is an equally valid upper bound for any probability; thus, we may upper bound $\overline{P(E_{jpb}/Y_1 Y_r X_{m_0 1} X_{m_0 r} m_0)}$ by the minimum of one and the right-hand side of inequality (3.2.3). A frequently used inequality (Gallager, 1968) states that if u and v are positive numbers,

$$\min(u,v) \le u^{1-\rho} v^{\rho}$$

for all ρ in the range $0 \le \rho \le 1$. Using this inequality to upper-bound the minimum of one and the right hand side of (3.2.3), we find that

$$\overline{P(E_{jpb}/Y_1 Y_r X_{m_0 1} X_{m_0 r} m_0)} \le \left\{ \sum_{m'' \epsilon M_{jpb}} \sum_{X_{m''1} X_{m''r}} P(X_{m''1} X_{m''r}/X_{m_0 1} X_{m_0 r} m_0) \times \left[\frac{P(Y_1 Y_r/X_{m''1} X_{m''r})}{P(Y_1 Y_r/X_{m_0 1} X_{m_0 r})} \right]^s \right\}^{\rho} . \tag{3.2.4}$$

The condition in the probability on the left-hand side of inequality (3.2.4) may be removed by taking the expectation over the conditioning event. Thus,

$$\overline{P(E_{jpb})} \leq \sum_{Y_1 Y_r} \sum_{m_0} \sum_{X_{m_0 1} X_{m_0 r}} P(Y_1 Y_r / X_{m_0 1} X_{m_0 r} m_0)\, P(X_{m_0 1} X_{m_0 r} / m_0)$$

$$\times\, P(m_0) \Bigg\{ \sum_{m'' \epsilon M_{jpb}} \sum_{X_{m'' 1} X_{m'' p}} P(X_{m'' 1} X_{m'' r} / X_{m_0 1} X_{m_0 r} m_0)$$

$$\times \left[\frac{P(Y_1 Y_r / X_{m'' 1} X_{m'' r})}{P(Y_1 Y_r / X_{m_0 1} X_{m_0 r})} \right]^s \Bigg\}^{\rho} . \qquad (3.2.5)$$

The statistical independence of the channel noise and the message m_0 guarantees that

$$P(Y_1 Y_r / X_{m_0 1} X_{m_0 r} m_0) = P(Y_1 Y_r / X_{m_0 1} X_{m_0 r}).$$

Moreover, the memoryless channel permits the factoring of $P(Y_1 Y_r / X_{m1} X_{mr})$ as

$$P(Y_1 Y_r / X_{m1} X_{mr}) = P(Y_1 / X_{m1}) P(Y_r / X_{mr}).$$

Substituting these two relations into the right-hand side of inequality (3.2.5) and setting $s=\frac{1}{1+\rho}$, we find that

$$\overline{P(E_{jpb})} \leq \sum_{Y_1} \sum_{Y_r} \sum_{m_0} \sum_{X_{m_01}} \sum_{X_{m_0r}} P(X_{m_01}X_{m_0r}/m_0)P(m_0)P(Y_1/X_{m_01})^{\frac{1}{1+\rho}}$$

$$\times P(Y_r/X_{m_0r})^{\frac{1}{1+\rho}} \left\{ \sum_{m''\epsilon M_{jpb}} \sum_{X_{m''1}} \sum_{X_{m''r}} P(X_{m''1}X_{m''r}/X_{m_01}X_{m_0r}m_0)P(Y_1/X_{m''1})^{\frac{1}{1+\rho}} P(Y_r/X_{m''r})^{\frac{1}{1+\rho}} \right\}^{\rho}.$$

(3.2.6)

Several properties of the ensemble of multiple generator length convolutional codes allow additional simplification of the right hand side of inequality (3.2.6). Let us denote the number of diverged phase 2 through phase V channel symbols in the pattern p as N_{bp}. For a systematic convolutional code with $k_1=0$ and all other k_v's equal to K, $N_{bp}=(b+1+K)(V-1)$. The random additive sequence $\underline{r}$ ensures that the channel symbol sequences X_{m_01} and X_{m_0r} are equally likely to be any sequence of $N_{pb2}-D_2k_2$ and $N_{bp}+D_2k_1$ q-ary symbols, respectively, for any m_0. Moreover, the random sequence $\underline{r}$ ensures that all X_{m_0r} sequences are equally probable for any given X_{m_01} and m_0. Thus,

$$P(X_{m_0 l}X_{m_0 r}/m_0) = Q(X_{m_0 l})Q(X_{m_0 r})$$

where Q( ) is the probability assignment in which all sequences occur with equal probability. The reader should note that the exact numerical value of Q( ) is dependent upon the length of the sequence of q-ary symbols that is the argument of Q( ). The discussion above indicates that for any m" in M_{jpb} the sequence $X_{m"r}$ is equally likely to be any sequence of q-ary symbols independent of $\underline{X}_{m_0}$ and m_0. Since there are different encoder weights used in generating $X_{m"l}$ and $X_{m"r}$, $X_{m"r}$ is also independent of $X_{m"l}$. Thus,

$$\begin{aligned} P(X_{m"l}X_{m"r}/X_{m_0 l}X_{m_0 r}m_0) &= P(X_{m"r}/X_{m"l}X_{m_0 l}X_{m_0 r}m_0) \\ &\quad \times P(X_{m"l}/X_{m_0 l}X_{m_0 r}m_0) \\ &= Q(X_{m"r})P(X_{m"l}/X_{m_0 l}X_{m_0 r}m_0). \end{aligned}$$

Substituting these equations into the right-hand side of inequality (3.2.6) and performing some algebra, we find that

$$\begin{aligned}\overline{P(E_{jpb})} \leq \sum_{Y_1}\sum_{Y_r}\sum_{X_{m_0 1}} & Q(X_{m_0 1})P(Y_1/X_{m_0 1})^{\frac{1}{1+\rho}} \\ \times & \sum_{X_{m_0 r}} Q(X_{m_0 r})P(Y_r/X_{m_0 r})^{\frac{1}{1+\rho}} \\ \times & \sum_{m_0} P(m_0) \left\{ \sum_{X_{m''r}} Q(X_{m''r})P(Y_r/X_{m''r})^{\frac{1}{1+\rho}} \right. \\ \times & \left. \sum_{m''\epsilon M_{jpb}} \sum_{X_{m''1}} P(X_{m''1}/X_{m_0 1}X_{m_0 r}m_0)P(Y_1/X_{m''1})^{\frac{1}{1+\rho}} \right\}^{\rho} . \end{aligned} \qquad (3.2.7)$$

The summations over $m''\epsilon M_{jpb}$ and $X_{m''1}$ are difficult to perform because of mathematical difficulty in expressing the requirements on the m" in M_{jpb}. However, the one-to-one mapping from information subsequences m" in M_{jpb} into channel symbol sequences $X_{m''1}$ assures that for each code in the ensemble there is a unique $X_{m''1}$ subsequence for any specific m". Hence for any specific code and fixed m", $P(X_{m''1}/X_{m_0 1}X_{m_0 r}m_0)$ is unity for one specific $X_{m''1}$ and zero for all other possible $X_{m''1}$. Thus, the summation over $m''\epsilon M_{jpb}$ may be viewed as just a summation over sequences $X_{m''1}$. Because of the one-to-one nature of the mapping from m" into $X_{m''1}$ subsequences, no possible $X_{m''1}$ subsequence enters the combined m" and $X_{m''1}$ summation more than once. The right-hand side of inequality (3.2.7) is not decreased if this implied summation over $X_{m"1}$ subsequences is expanded to include all possible $X_{m"1}$ subsequences instead of just those $X_{m"1}$ required by the code and by the condition $m"\epsilon M_{jpb}$. Finally note that

$$Q(X_{m"1}) = q^{-(N_{pb2}-D_2k_2)}.$$

Thus,

$$\overline{P(E_{jpb})} \leq \sum_{Y_1}\sum_{Y_r}\sum_{X_{m_0 1}} Q(X_{m_0 1})P(Y_1/X_{m_0 1})^{\frac{1}{1+\rho}} \sum_{X_{m_0 r}} Q(X_{m_0 r})P(Y_r/X_{m_0 r})^{\frac{1}{1+\rho}}$$
$$\times \left\{ \sum_{X_{m"r}} Q(X_{m"r})P(Y_r/X_{m"r})^{\frac{1}{1+\rho}} \sum_{X_{m"1}} q^{(N_{bp2}-D_2k_2)}\, Q(X_{m"1})P(Y_1/X_{m"1})^{\frac{1}{1+\rho}} \right\}^{\rho},$$

(3.2.8)

$X_{m_0 r}$ and $X_{m"r}$ are different indices of summation in identical summations and $X_{m_0 1}$ and $X_{m"1}$ are also different indices for identical summations. Thus,

$$\overline{P(E_{jpb})} \leq q^{\rho(N_{pb2}-D_2k_2)} \sum_{Y_1}\left\{\sum_{X_{m1}} Q(X_{m1})P(Y_1/X_{m1})^{\frac{1}{1+\rho}}\right\}^{1+\rho}$$
$$\times \sum_{Y_r}\left\{\sum_{X_{mr}} Q(X_{mr})P(Y_r/X_{mr})^{\frac{1}{1+\rho}}\right\}^{1+\rho}. \qquad (3.2.9)$$

Since the channel in memoryless, the right-hand side of inequality (3.2.9) may be further simplified. The subsequence X_{mr} may be any sequence of $N_{bp}+D_2k_1$ q-ary symbols with equal probability. Numbering these channel symbols in some way, we may write

$$Q(X_{mr}) = \prod_{i=1}^{N_{bp}+D_2k_1} Q(x_{mri})$$

where $Q(x_{mri})$ is the probability assignment on the i^{th} letter of X_{mr}. For the memoryless channel, $P(Y_r/X_{mr})$ is the product of the individual channel transition probabilities. Using the same numbering scheme for the symbols of Y_r as for the symbols of X_{mr},

$$P(Y_r/X_{mr}) = \prod_{i=1}^{N_{bp}+D_2k_1} P(y_{ri}/x_{mri}).$$

Hence,

$$\sum_{Y_r}\left\{\sum_{X_{mr}} Q(X_{mr})P(Y_r/X_{mr})^{\frac{1}{1+\rho}}\right\}^{1+\rho} =$$

$$\sum_{y_1}\cdots\sum_{y_{N_{bpr}}}\left\{\sum_{x_1}\cdots\sum_{x_{N_{bpr}}}\prod_{i=1}^{N_{pb}+D_2k_1}\right.$$

$$\left.\times\, Q(x_{mri})P(y_{ri}/x_{mri})^{\frac{1}{1+\rho}}\right\}^{1+\rho} .$$

(3.2.10)

A little thought shows that the order of summation and multiplication may be interchanged in the right-hand side of Eq. (3.2.10). Thus,

$$\sum_{Y_r}\left\{\sum_{X_{mr}} Q(X_{mr})P(Y_r/X_{mr})^{\frac{1}{1+\rho}}\right\}^{1+\rho} =$$

$$\prod_{i=1}^{N_{bp}+D_2k_1} \sum_{y_{ri}}\left\{\sum_{x_{mri}} Q(x_{mpi})P(y_{pi}/x_{mpi})^{\frac{1}{1+\rho}}\right\}^{1+\rho} .$$

(3.2.11)

The term in braces on the right-hand side of Eq. (3.2.11) is identical for each i. Thus, following Gallager's (1965) notation,

$$\sum_{Y_r}\left\{\sum_{X_{mr}} Q(X_{mr})P(Y_r/X_{mr})^{\frac{1}{1+\rho}}\right\}^{1+\rho} = \exp -(N_{bp}+D_2k_1)E_0(\rho,Q)$$

(3.2.12)

where

$$E_0(\rho,Q) = -\ln\left[\sum_k\left(\sum_i Q(i)P(k/i)^{\frac{1}{1+\rho}}\right)^{1+\rho}\right].$$

(3.2.13)

A similar argument shows that

$$\sum_{Y_1}\left(\sum_{X_{m1}} Q(X_{m1})P(Y_1/X_{m1})^{\frac{1}{1+\rho}}\right)^{1+\rho} =$$

$$\exp -(N_{pb2}-D_2k_2)E_0(\rho,Q)$$

(3.2.14)

Equations (3.2.12) and (3.2.14) may be substituted into the right-hand side of inequality (3.2.9) to show that

$$\overline{P(E_{jpb})} \leq q^{\rho(N_{pb2}-L_2k_2)} \exp -\left[N_{pb2}-D_2(k_2-k_1)+N_{pb}\right] E_0(\rho,Q).$$

The notational cumbersomeness of this upper-bound on $\overline{P(E_{jpb})}$ may be decreased if remember that $N_{pb2}-D_2k_2$

is the total number of possibly differing information symbols in m" consistent with the pattern p. Moreover, $N_{pb2}-D_2(k_2-k_1)$ is the total number of possibly diverged phase 1 channel symbols consistent with the pattern p.

We may summarize by stating a lemma which we have just proved.

Lemma:

Let M_{jpb} be the set of all incorrect messages completely merged with m_0 at the $(j-1)^{th}$ encoder shift, diverging at the j^{th} encoder shift, not completely merging until the $(j+b+K+1)^{th}$ encoder shift, and having a fixed pattern p of diverged phase 2 through phase V channel symbols. Let $\overline{P(E_{jpb})}$ be the ensemble average probability that an optimum decoder will decode any m" in M_{jpb} instead of the corresponding subsequence of m_0. Then

$$\overline{P(E_{jpb})} \leq q^{\rho(I_p)} \exp -(N_{1p}+N_{bp})E_0(\rho,Q) \tag{3.2.15}$$

for any ρ such that $0\leq\rho\leq 1$, where N_{bp} is the number of diverged phase 2 through phase V channel symbols in the pattern p, I_p is the number of possibly differing information symbols implied by the pattern p and N_{1p} is the number of possibly diverged phase 1 channel symbols implied by the pattern p. We have used the phrase "possibly differing information symbol" to denote information symbols in m" which the pattern p does not <u>require</u> to be identical to the corresponding symbol of m_0. The phrase "possibly diverged phase 1 channel symbol" has a similar meaning.

The reader should note that the pattern p of diverged phase 2 through phase V channel symbols is fixed for all m" in M_{jpb} but that all patterns of diverged phase 1 channel symbols consistent with the pattern p are included.

3.3 Error Probability for Systematic Convolutional Codes.

We may use the lemma 3.2.15 to derive an upperbound to the ensemble average probability of erroneous communication for a systematic convolutional code with maximum-likelihood decoding. A systematic convolutional code has $k_1=0$ and all other k_v's equal K. There is no difficulty added in considering the larger family of convolutional codes in which k_1 is arbitrary and all other k_v's equal K. First, let us determine what patterns of diverged phase 2 through phase V channel symbols are consistent with the generator lengths used. Since $k_2= k_3=\ldots=k_V=K$, the phase 2, phase 3, ... and phase V channel symbols must all diverge and merge together. Thus, the only possible patterns of diverged phase 2 through phase V channel symbols are long blocks of diverged channel symbols in which all phase 2 through phase V channel symbols in the block are diverged. Because of the requirements for a phase V merger, this long block of diverged channel symbols must be K+1 information register shifts long or longer. Suppose that the length of this block of diverged channel symbols is b+K+1 information register shifts. As discussed in section 3.2, the K information symbols corresponding to the last K encoder shifts in this block must be identical to the corresponding symbol of m_0. Thus, a block of b+K+1 diverged phase 2 through phase V channel symbols implies b+1 possibly differing information symbols in m". Likewise, this block of b+K+1 diverged phase 2 through phase V channel symbols implies $b+1+k_1$ possibly diverged phase 1 channel symbols. Setting I_p=b+1, N_{1p}=$b+1+k_1$ and N_{bp}=(b+K=1)(V-1), we may use the lemma to upper-bound $\overline{P(E_{jb})}$, the ensemble average probability of the decoder's selecting some incorrect message subsequence that is completely merged at the $(j-1)^{th}$ encoder shift, diverges at the j^{th} shift and completely remerges with m_0 immediately after the $(j+b+K+1)^{th}$ encoder shift. Thus,

$$\overline{P(E_{jb})} \leq q^{\rho(b+1)} \exp - \left[(b+1)V+K(V-1)+k_1\right] E_0(\rho,Q). \tag{3.3.1}$$

The upper bound on $\overline{P(E_{jb})}$ may be used to find an upper bound on $\overline{P(E_{block})}$, the ensemble average probability that any of the L information symbols in the block is decoded incorrectly. If any of the decoded information symbols is incorrect, the decoder must have decoded some m" in some M_{jpb}. For the codes under consideration, there is only one pattern p of diverged phase 2 through phase V channel symbols diverging at the j^{th} encoder shift and remerging at the $(j+b+K+1)^{th}$ encoder shift. Using a union bound to account for all j and for all b, we find that

$$\overline{P(E_{block})} \leq \sum_{j=1}^{L} \sum_{b=0}^{L-j} \overline{P(E_{jb})}. \tag{3.3.2}$$

Using inequality (3.3.1) to upper-bound the members of the double summation in the right-hand side of (3.3.2), we find that

$$\overline{P(E_{block})} \leq \exp -\left[K(V-1)+k_1\right]E_0(\rho,Q)$$
$$\times \sum_{j=1}^{L} \sum_{b=0}^{L-j} \exp -(b+1)V\left[E_0(\rho,Q)-\rho R\right] \tag{3.3.3}$$

where R is the nominal data rate of the convolutional code

$$R = \frac{\ln(q)}{V}.$$

Since L may be arbitrarily large, we shall neglect the small rate loss occurring because of the periodic resetting.

The right-hand side of inequality (3.3.3) is not decreased if the upper limit of the b-summation is raised to infinity. The infinite sum over b converges if and only if (see note)

$$\rho R < E_0(\rho,Q) \quad \text{for some } \rho \quad 0 \le \rho \le 1. \tag{3.3.4}$$

Taking the infinite sum over b and the finite sum over j, we find that

$$\overline{P(E_{block})} \le L \frac{1}{e^{V\epsilon}-1} \exp - \left[K(V-1)+k_1\right] E_0(\rho,Q) \tag{3.3.5}$$

where

$$E_0(\rho,Q) - \rho R = \epsilon > 0 \text{ and } \quad 0 \le \rho \le 1. \tag{3.3.6}$$

In order to obtain the tightest upper-bound on $\overline{P(E_{block})}$ we select that value of ρ which maximizes $E_0(\rho,Q)$ subject to the convergence condition of Eq. (3.3.6) Gallager (1965) has shown that this tightest bound may be obtained by selecting the largest value of ρ which satisfies the dual conditions listed in (3.3.6).

The upper-bound on $\overline{P(E_{jb})}$ may also be used to upper-bound $\overline{P(E_{symbol})}$ the ensemble average probability that any specific information symbol was decoded incorrectly. If the w^{th} symbol of the decoded information

Note: The reader may wonder at the wisdom of raising the upper limit of the b summation to infinity and then requiring that the infinite converge. Such a convergence condition is prudent in that if the infinite sum did not converge, the L power term in the finite sum would dominate and give a bound that is exponentially <u>increasing</u> with the length of the information sequence.

sequence is erroneous, it is erroneous because either some m" subsequence with any b and j=w was accepted or because some m" subsequence with b≥i and j=w-i was accepted. Using a union bound,

$$\overline{P(E_{\text{symbol}})} \leq \sum_{i=0}^{L-1} \sum_{b=i}^{L-i} \overline{P(E_{ib})}$$

Raising the upper limits of both summations to infinity and using the upper bound on $\overline{P(E_{ib})}$,

$$\overline{P(E_{\text{symbol}})} \leq \exp -\left[K(V-1)+k_t\right]E_0(\rho,Q)$$

$$\times \sum_{i=0}^{\infty} \sum_{b=i}^{\infty} \exp -(b+1)V\left[E_0(\rho,Q)-\rho R\right].$$

Expressing the summations on the right-hand side in a different form, we have

$$\overline{P(E_{\text{symbol}})} \leq \exp -K(V-1)+k_t \quad E_0(\rho,Q)$$

$$\sum_{i=0}^{\infty} (i+1)\exp -(i+1)V\left[E_0(\rho,Q)-\rho R\right]. \tag{3.3.7}$$

If the dual conditions of Eq. (3.3.4) are met, the infinite summation in the right-hand side of inequality (3.3.7) converges and

$$\overline{P(E_{\text{symbol}})} \leq \frac{e^{V\epsilon}}{(e^{V\epsilon}-1)^2} \exp -\left[K(V-1)+k_1\right] E_0(\rho,Q) \tag{3.3.8}$$

The awkward appearance of the dual conditions in Eq. (3.3.4) may be removed by defining

$$E_U(R) = \min \begin{cases} E_o(1,Q) \\ \\ E_o(\rho,Q) \text{ with } \rho \text{ such that } E_o(\rho,Q)-\rho R=\epsilon>0 \end{cases} \tag{3.3.9}$$

We have defined K*V such that for these codes

$$K{*}V=k_1+K(V-1).$$

We may use the definition of $E_U(R)$ to write

$$\overline{P(E_{\text{block}})} \leq \frac{L}{e^{V\epsilon}-1} \exp -K{*}V\, E_U(R) \tag{3.3.10}$$

and

$$\overline{P(E_{\text{symbol}})} \leq \frac{e^{+V\epsilon}}{(e^{V\epsilon}-1)^2} \exp -K{*}V\, E_U(R) \tag{3,3.11}$$

If Q( ) is the probability assignment which maximizes $E_0(\rho,\underline{Q})$ as a function of $\underline{Q}$, a result by Shannon, Gallager and Berlekamp (1967) shows that $E_L(R)=E_U(R)$ for $R \geq E_0(1,Q)$. The class of channels for which Q( ) maximizes $E_0(\rho,\underline{Q})$ as a function of $\underline{Q}$ includes symmetric channels. Thus, the upper bounds on error probability in inequalities (3.3.10) and (3.3.11) are exponentially tight for many channels of interest. Figure 3.3.1 shows $E_L(R)$ and $E_U(R)$ for a typical channel and compares these error exponents with the analogous terms for block codes (Gallager, 1968) of similar encoder complexity K*V.

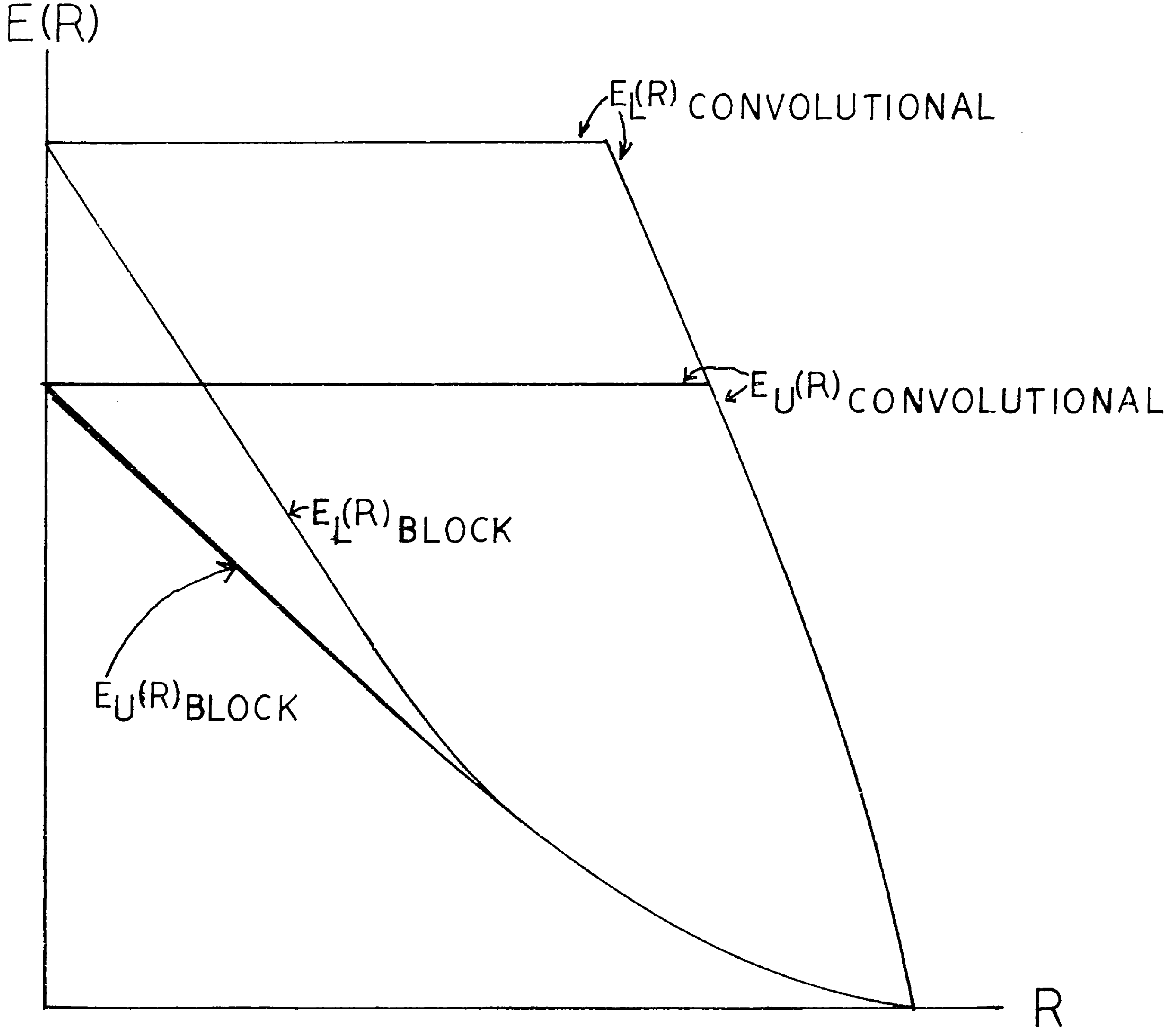

FIG. 3.3.1 E(R) CURVES FOR BLOCK AND CONVOLUTIONAL CODES ON A TYPICAL CHANNEL

3.4 Error Probability for Multiple Generator Length Convolutional Codes

In this section, we use the lemma presented in section 3.2 to derive an upper-bound to the probability of error for multiple generator length convolutional codes with optimum decoding. The lemma gives an upper-bound to $\overline{P(E_{jpb})}$ the ensemble average probability of decoding any incorrect information sequence m" which is completely merged with m_0 at the $(j-1)^{th}$ encoder shift, diverges from m_0 at the j^{th} shift, completely remerges with m_0 immediately after the $(j+b+K+1)^{th}$ encoder shift and has a fixed pattern p of diverged phase 2 through phase V channel symbols. If an information symbol is erroneously decoded, some m" with some j, p and b must have been decoded instead of the corresponding subsequence of m Using a union bound, we may upper-bound $\overline{P(E_{block})}$ by the expression

$$\overline{P(E_{block})} \leq \sum_{j=1}^{L} \sum_{p} \sum_{b} \overline{P(E_{jpb})}. \qquad (3.4.1)$$

In order to use the lemma, we must have some way of knowing how many patterns p there are with N_{bp} diverged phase 2 through phase V channel symbols and for which the pattern p implies I_p possibly differing information symbols and N_{1p} possibly diverged phase 1 channel symbols. Let $N(I_p,N_{1p},N_{bp})$ be the number of such patterns p. Then using the lemma we find that,

$$\overline{P(E_{block})} \leq \sum_{j=1}^{L} \sum_{p} \sum_{b} N(I_p,N_{1p},N_{bp}) \\ q^{\rho(I_p)} \exp-(N_{1p}+N_{bp})E_0(\rho,Q) \tag{3.4.2}$$

for any ρ, $0<\rho\leq 1$. Since the parameter b is essentially determined by the pattern p, we may include the b-summation in the p-summation for convenience.

In order the calculate a value for the upper-bound in inequality (3.4.2) we must know $N(I_p,N_{1p},N_{bp})$. A general way of solving combinatorial problems is the combinatorial generating function. Since communications oriented engineers are seldom familiar with combinatorial generating functions, we will present a short introduction to combinatorial generating functions. If the following introduction to combinatorial generating functions is too brief, the reader may consult a book on combinatorial analysis; e.g. Riordan (1958) or Liu (1968).

Combinatorial generating functions are best taught by example. Consider three objects labeled x_1, x_2, and

x_3. Form the algebraic product

$$(1+x_1 z)(1+x_2 z)(1+x_3 z) = 1 + (x_1+x_2+x_3)\ z + (x_1x_2+x_1x_3+x_2x_3)\ z^2 + (x_1x_2x_3)\ z^3 \tag{3.4.3}$$

The coefficient of z^h in the right-hand side of (3.4.3) contains one additive term for each combination of three x's taken h at a time. Hence, the number of combinations of three things taken h at a time is the coefficient of z^h with all three x's set to one. We may readily extend this result to combinations of N things taken h at a time by using N factors of $(1+x_i z)$ instead of three. The polynomial

$$F(z) = \prod_{i=1}^{N} (1+x_i z) \tag{3.4.4}$$

is called the combinatorial generating function of N things with no object selected more than once. The principal property of this generating function is that the number of combinations of N things taken h at a time is just the coefficient of the term z^h when all x's are set to one. In expression (3.4.4), each factor of the product is a binomial which indicates in terms of 1 and $x_i z$ the fact that the object x_i may not or may appear in any combination. The product generates combinations because the coefficient of z^h is obtained by picking unity terms from n-h factors and terms like $x_i z$ from the remaining h factors in all possible ways. The factors in (3.4.4) are limited to two terms because no object may appear more than once. If the object x_i may appear 0,1,3 or 5 times, the generating function is altered by writing

$$\left[1+x_i z+(x_i z)^3+(x_i z)^5\right]$$

in place of $(1+x_i z)$.

Let us conclude this introduction to combinatorial generating functions by finding H(y,z) the generating function for combinations of objects taken from two different sets of objects. Let F(y) be the generating function of combinations of objects in the first set and G(z) be the generating function of combinations of objects taken from the second set. Any combination of objects taken from the first set may be paired with any combination of objects taken from the second set. Thus the number of combinations of i objects from the first set and j objects from the second set is just the product of the number of combinations of i objects from the first set and the number of combinations of j objects from the second set. Thus,

$$H(y,z) = F(y)G(z).$$

If all the x's (object name indicators) are set to one, the coefficient of $y^i z^j$ in $H(y,z)$ is the number of ways of selecting i objects from the first set and j objects from the second set. The number of ways of selecting a total of k objects from the two sets combined is just the sum over i of coefficients of all $y^i z^{k-i}$ terms in $H(y,z)$. Hence the number of combinations of k objects selected from the two sets combined is just the coefficient of the z^k term in $H(z,z)$. If we are interested in knowing only the number of combinations without enumerating these combinations, we may set the x_i's equal to one when the generating function is written.

Let us now use combinatorial generating functions to determine the number $N(I_p, N_{1p}, N_{bp})$ in the right hand side of inequality (3.4.2). In this particular case, there are three different kinds of objects involved in the combinations. Thus the generating function must be a polynomial of three different variables. Let

$F(u,d_1,d)$ be the generating function of the number of patterns p of N_{pb} diverged phase 2 through phase V channel symbols in which the pattern p implies I_p possibly differing information symbols and N_{1p} possibly diverged phase 1 channel symbols. Hence,

$$F(u\ d_1,d) = \sum_{I_p} \sum_{N_{1p}} \sum_{N_{bp}} N(I_p,N_{1p},N_{bp})\ u^{(I_p)} d_1^{(N_{1p})} d^{(N_{bp})}. \tag{3.4.5}$$

Since the lemma in section 3.2 was developed by looking at distinct runs of diverged phase 2 channel symbols, let us continue to look at runs of diverged phase 2 channel symbols. We may divide the pattern p into a number of distinct segments. Let us define a segment of the pattern p as the portion of the pattern following (and including) the start of a run of diverged phase 2 channel symbols and preceeding the next run of diverged phase 2 channel symbols. By definition, the last segment of the pattern p terminates when there is a complete remerger. Using the notation of section 3.2, a pattern has D_2 segments. In figure 3.2.1, each segment of the pattern is underscored with a brace. In the simplest case, there is only one segment in the pattern p. Let $T(u,d_1,d)$ be the part of $F(u,d_1,d)$

representing this terminating segment. In the next most simple case, there will be one earlier non-terminating segment in the pattern preceeding the last and terminating segment. Let $E(u,d_1,d)$ be the factor of the generating function representing this non-terminating segment. Since the terminating and non-terminating segments are independent entities, the term of $F(u,d_1,d)$ representing this two segment pattern is just $T(u,d_1,d)E(u,d_1,d)$. In general there may be i non-terminating segments in the pattern. $E(u,d_1,d)$ is the factor of a combinatorial generating function representing one of these earlier segments. Thus,

$$F(u,d_1,d) = T(u,d_1,d) \times \left\{ \sum_{i=0}^{\infty} \left[E(u,d_1,d)\right]^i \right\}, \tag{3.4.6}$$

The combinatorial properties of the terminating segment of the pattern differ from those of the earlier segments. Since the terminating segment is the simpler case, let us consider it first. This terminating segment must end with a complete remerger. This remerging part of the pattern must be preceeded by a run of k_2+1 or more diverged phase 2 (and hence diverged phase 3 through phase V) channel symbols. Let this run of diverged phase 2 channel symbols be $c+k_2+1$ symbols long.

From section 3.2, we remember that such a run of diverged phase 2 channel symbols implies a run of c+1 possibly differing information symbols and $c+1+k_1$ possibly diverged phase 1 channel symbols. A divergence diagram for this terminating segment is shown in figure 3.4.1. Measuring the shaded area in figure 3.4.1, we find that this terminating segment has $(c+1+k_2)(V-1)+(k_3-k_2)(k_4-k_2)+\ldots+(k_V-k_2)$ diverged phase 2, phase 3,...or phase V channel symbols. Using the definition of K*

$$K^*V = k_1 + k_2 + k_3 + \ldots + k_V,$$

we find that this terminating segment has a total of $(c+1)(V-1)+K^*V-k_1$ diverged phase 2 through phase V channel symbols. The number c may be any non-negative integer. If we let u^b represent a string of b possibly differing information symbols, d_1^c represent c possibly diverged phase 1 channel symbols and d^n represent n diverged phase 2, phase 3,... or phase V channel symbols,

$$T(u,d_1,d) = \sum_{c=0}^{\infty} u^{(c+1)}(d_1)^{(c+1+k_1)}d^{(V-1)(c+1)+K^*V-k_1}. \tag{3.4.7}$$

By the definition of combinatorial generating functions,

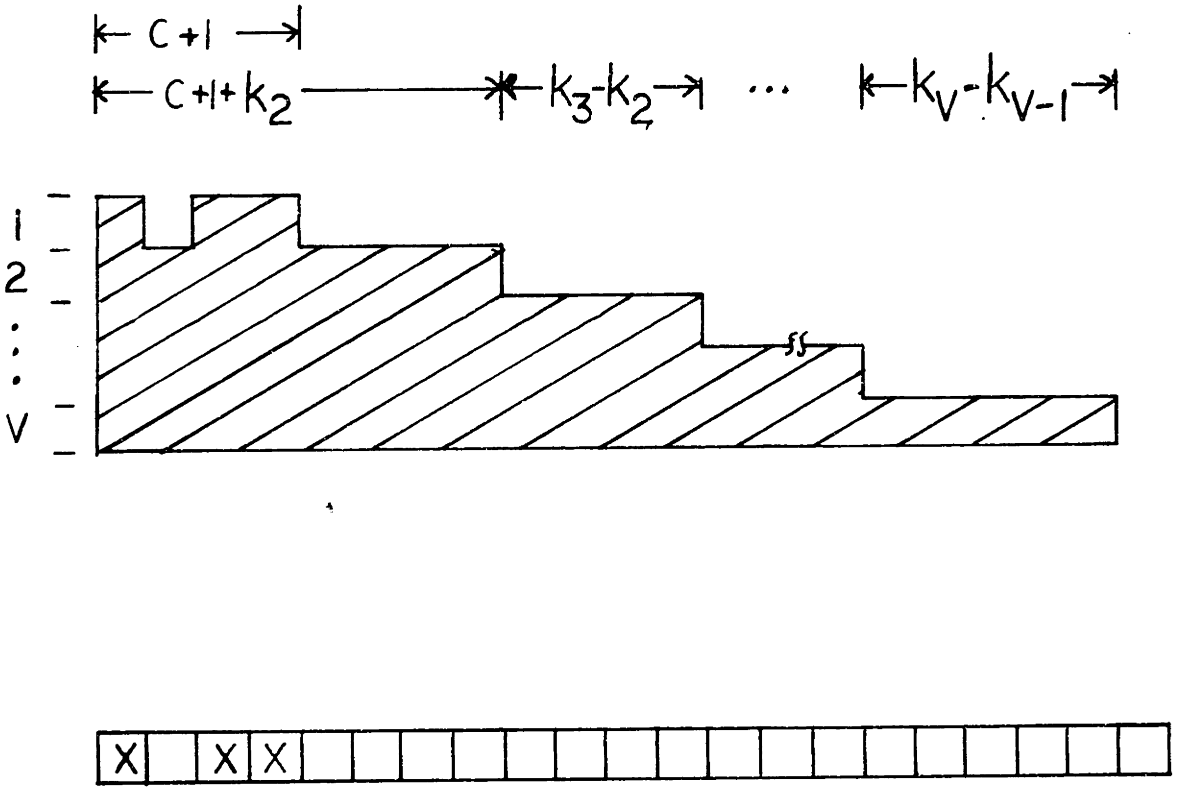

FIG. 3.4.1 TERMINATING SEGMENT

the coefficient of $u^b d_1^c d^n$ in $T(u,d_1,d)$ is the number of terminating segments with n diverged phase 2 through phase V channel symbols, a string of b possibly differing information symbols and c possibly diverged phase 1 channel symbols.

The non-terminating segments of the pattern p are identical to the terminating segment except that they must end at or before a complete remerger. There are many possible divergence diagrams for non-terminating segments. Each of these divergence diagrams takes the same form as the divergence diagram in figure 3.4.1 except that the run of merged phase 2 channel symbols at the end of the segment may assume any length between one and k_V-k_2. The number of diverged phase 2, phase 3,... or phase V channel symbols implied by a run of ν merged phase 2 channel symbols at the end of the segment is given by the function $f(\nu)$.

$$f(\nu) = \begin{cases} \nu(V-2) & 0<\nu\leq k_3-k_2 \\ f(k_3-k_2)+\left[\nu-(k_3-k_2)\right](V-3) & k_3-k_2<\nu\leq k_4-k_3 \\ \vdots & \\ f(k_{V-1}-k_{V-2})+\left[\nu-(k_{V-1}-k_{V-2})\right] & k_{V-1}-k_{V-2}<\nu\leq(k_V-k_{V-1}). \end{cases} \tag{3.4.8}$$

If the form of $f(\nu)$ seems a bit difficult to see, the reader may be aided by Table 3.4.1 in which the number of diverged phase 2, phase 3,...or phase V channel symbols implied by a string of ν merged phase 2 channel symbols is given for the code in which V=5, $k_1=1$, $k_2=4$, $k_3=8$, $k_4=10$, and $k_5=13$.

ν	Number of diverged phase 2,...or phase V channel symbols	Remarks
1	3	phases 1 and 2 merged
2	6	"
3	9	"
4	12	"
5	14	phase 3 also merged
6	16	"
7	17	phase 4 also merged
8	18	"
9	19	"
10	undefined	complete remerger

Table 3.4.1: $f(\nu)$ for a specific code with explanatory remarks.

The non-terminating segments have $(c+1+k_2)+f(\nu)$ diverged phase 2, phase 3,...or phase V channel symbols. Such a terminating segment has a string of c+1 possibly differing information symbols and implies $c+1+k_1$ possibly diverged phase 1 channel symbols. As above, the number c may be any non-negative integer. The number ν may be any integer between one and $k_V - k_2$. Thus,

$$E(u,d_1,d) = \sum_{c=0}^{\infty} u^{(c+1)} d_1^{(c+1+k_1)} d^{(c+1+k_2)(V-1)} \left[\sum_{\nu=1}^{k_V-k_2} d^{f(\nu)} \right]. \qquad (3.4.9)$$

Substituting Eqs. (3.4.7) and (3.4.9) into Eq. (3.4.6), we find that

$$F(u,d_1,d) = \sum_{c=0}^{\infty} \left[u d_1 d^{(V-1)} \right]^{(c+1)} d_1^{(k_1)} d^{(K*V-k_1)}$$

$$\times \sum_{i=0}^{\infty} \left\{ \sum_{c=0}^{\infty} \left[u d_1 d^{(V-1)} \right]^{(c+1)} d_1^{(k_1)} d^{(V-1)(k_2)} \right.$$

$$\left. \times \left[\sum_{\nu=1}^{k_V-k_2} d^{f(\nu)} \right] \right\}^{i}. \qquad (3.4.10)$$

From Eq. (3.4.5), we see that the coefficient of $u^{(I_p)} d_1^{(N_{1p})} d^{(N_{bp})}$ is $N(I_p, N_{1p}, N_{bp})$ the number of patterns of N_{bp} diverged phase 2 through phase V channel symbols with I_p possibly differing information symbols and N_{1p} possibly diverged phase 1 channel symbols.

The summation over all p and b in the right-hand side of inequality (3.4.2) is just the same as the summation over all I_p, N_{lp} and N_{bp}. Thus,

$$\overline{P(E_{\mathbf{block}})} \le \sum_{j=1}^{L} \left\{ \sum_{I_p} \sum_{N_{lp}} \sum_{N_{bp}} N(I_p, N_{lp}, N_{bp}) \right.$$

$$\left. q^{\rho(I_p)} \exp - (N_{lp}+N_{bp})E_0(\rho, Q) \right\}.$$

(3.4.11)

Comparing the right-hand side if Eq. (3.4.5) and the term in braces in the right-hand side of inequality (3.4.11), we find that the two expressions are identical if $u=q^{\rho}$, $d_l=\exp-E_0(\rho,Q)$ and $d=\exp-E_0(\rho,Q)$. Thus after performing the j-summation, we find that

$$\overline{P(E_{\text{block}})} \le L \times F \left[q^{\rho},\ \exp-E_0(\rho,Q)\ ,\ \exp-E_0(\rho,Q) \right]$$

where $F\left[u, d_l, d\right]$ is the combinatorial generating function from Eq. (3.4.10) and $0 \le \rho \le 1$. Thus,

$$
\begin{aligned}
\overline{P(E_{\text{block}})} \le L &\left\{ \exp -K^{*}VE_0(\rho,Q) \times \sum_{c=0}^{\infty} \left[q^{\rho} \quad \exp -VE_0(\rho,Q) \right]^{(c+1)} \right\} \\
&\times \sum_{i=0}^{\infty} \left\{ \left(\exp \, - \left[k_2(V-1)+k_1 \right] E_0(\rho,Q) \right) \right. \\
&\times \left(\sum_{c=0}^{\infty} \left[q^{\rho} \; \exp -VE_0(\rho,Q) \right]^{(c+1)} \right) \\
&\left. \times \left(\sum_{\nu=1}^{k_V-k_2} \quad \exp \, -f(\nu)E_0(\rho,Q) \right) \right\}^{i} .
\end{aligned}
\tag{3.4.12}
$$

for any ρ, in the range $0 \le \rho \le 1$.

Inequality (3.4.12) is meaningful only if the infinite summations over c and i converge. The infinite summation over c converges only if

$$q^{\rho} < \exp +VE_0(\rho,Q)$$

for some ρ, $0 \le \rho \le 1$. The nominal data rate R of the code is given by the equation

$$R = \frac{\ln(q)}{V}.$$

Thus, the convergence condition for the c-summation is equivalent to the requirement that

$$E_0(\rho,Q) -\rho R=\epsilon>0 \tag{3.4.13}$$

for some ρ in the range $0\leq\rho\leq1$. If this convergence condition is met,

$$\overline{P(E_{block})} \leq \frac{L}{e^{V\epsilon}-1} \exp -K^{*}VE_0(\rho,Q)$$

$$\sum_{i=0}^{\infty}\left\{\frac{1}{e^{V\epsilon}-1} \exp -\left[k_2(V-1)+k_1\right]E_0(\rho,Q)\right.$$

$$\left.\times\left[\sum_{\nu=1}^{k_V-k_2} \exp -f(\nu)\; E_0(\rho,Q)\right]\right\}^{i}. \tag{3.4.14}$$

The i-summation converges if the quantity in braces on the right-hand side of inequality (3.4.14) is less than one. Rather than check i-summation convergence for a number of specific codes and channels, we will look for an asymptotic result. Let us consider convolutional codes in which the length of each generator is proportional to K. For this type of code, Thus, the convergence condition for the c-summation is equivalent to the requirement that

$$E_0(\rho,Q) - \rho R = \epsilon > 0 \qquad (3.4.13)$$

for some ρ in the range $0 \leq \rho \leq 1$. If this convergence condition is met,

$$\overline{P(E_{\text{block}})} \leq \frac{L}{e^{V\epsilon}-1} \exp -K^* V E_0(\rho,Q)$$

$$\sum_{i=0}^{\infty} \left\{ \frac{1}{e^{V\epsilon}-1} \exp -\left[k_2(V-1)+k_1\right] E_0(\rho,Q) \right.$$

$$\left. \times \left[\sum_{\nu=1}^{k_V-k_2} \exp -f(\nu)\, E_0(\rho,Q) \right] \right\}^i . \qquad (3.4.14)$$

The i-summation converges if the quantity in braces on the right-hand side of inequality (3.4.14) is less than one. Rather than check i-summation convergence for a number of specific codes and channels, we will look for an asymptotic result. Let us consider convolutional codes in which the length of each generator is proportional to K. For this type of code,

$$k_v = \lfloor r_v K+1 \rfloor$$

where r_v is some fraction and the notation $\lfloor x \rfloor$ means the greatest integer less than or equal to x. For a systematic code $r_1=0$. The convergence condition on the i-summation is met if

$$\left[\sum_{\nu=1}^{k_V-k_2} \exp-f(\nu)\times E_0(\rho,Q)\right]\times\left\{\exp-K\left[r_2(V-1)+r_1\right]E_0(\rho,Q)\right\}<e^{V\epsilon}-1.$$

This asymptotic convergence condition is still difficult to evaluate because of the dependence upon the function $f(\nu)$. This difficulty may be circumvented by noting that there are exactly k_V-k_2 terms in the ν-summation and that each of these terms is less than or equal to one for non-negative values of $E_0(\rho,Q)$. Thus, the i-summation converges if

$$(K-k_2)\times\exp -K\left[r_2(V-1)+r_1\right]E_0(\rho,Q) < e^{V\epsilon}-1.$$

A further simplification results if we use a truncated Taylor series for $e^{V\epsilon}$ and upper-bound $K-k_2$ by K. With this simplification, the convergence condition is more stringent but the i-summation is more readily performed for the general case. With this simplification, we find that the i-summation converges if,

$$K \exp -K\left[r_2(V-1)+r_1\right]E_0(\rho,Q) \quad < \quad V\epsilon$$

Since

$$\lim_{K\to\infty} Ke^{-Ka} = 0$$

for all positive a, there must be a finite K_n such that the i-summation converges for all $K\geq K_n$ provided that $r_2(V-1)+r_1$ is greater than zero. The fraction r_1 is zero for a systematic code. Hence if r_2 is greater than zero, the i-summation converges for K (and k_2) large enough and we may upper-bound the ensemble average probability of error by the expression

$$\overline{P(E_{\text{block}})} \leq \frac{L}{e^{V\epsilon}-1-V\epsilon} \exp -K^*V\, E_0(\rho,Q) \qquad (3.4.15)$$

when inequality (3.4.13) is satisfied and $K\geq K_n$. Following the procedure in section 3.3, we may minimize the right-hand side of (3.4.15) over all ρ in the range $0\leq\rho\leq 1$, which satisfy inequality (3.4.13). This minimum occurs at the maximum possible value of ρ in the range $0 \leq \rho \leq 1$ which satisfies inequality (3.4.13). Thus, when k_2 grows linearly with K and $K \geq K_n$

$$\overline{P(E_{block})} \leq \frac{L}{e^{V\varepsilon}-1-V\varepsilon} \exp -K^*VE_U(R) \tag{3.4.16}$$

where $E_U(R)$ is the upper-bound exponent defined in Eq. (3.3.9).

Following section 3.2, we may also derive an upper-bound on $\overline{P(E_{symbol})}$. The upper-bound on $\overline{P(E_{symbol})}$ may be found by multiplying each term $u^{(I_p)} d_1^{(N_{1p})} d^{(N_{bp})}$ by I_p, the number of information symbols in error for the pattern p, before setting $u=q^{\rho}$ and $d_1=d= \exp -E_0(\rho,Q)$. This multiplication may be easily done by taking u times the derivative of $F(u,d_1,d)$ with respect to u. The implied convergence conditions are the same as those encountered in upper-bounding $\overline{P(E_{block})}$. If these convergence conditions are met,

$$\overline{P(E_{symbol})} \leq \frac{1}{(e^{V\epsilon}-1-V\epsilon)^2} \exp -K^*V\, E_U(R). \tag{3.4.17}$$

The reader may wonder whether some form of absolute rather than asymptotic convergence is possible for the i-summation. Such an absolute convergence condition would prove inequalities (3.4.16) and (3.4.17) for all K and $k_2=0$ and not just for $K \geq K_n$ and k_2 proportional to K. Such an absolute convergence condition is impossible. The impossibility of such an absolute convergence condition may be seen by considering the multiple generator length convolutional code in which $V=0$, $k_1=k_2=k_3=0$ and $k_4=K$. For this particular code, the phase 1, phase 2 and phase 3 channel symbols are essentially repeats of the systematic channel symbol. Let us consider these three repeats of the systematic channel symbol as the input to a single channel with q^3 inputs and q^3 outputs and the phase 4 channel symbol as the input to the origional channel. A slightly generalized form of the sphere-packing lower bound (Shannon, Gallager and Berlekamp 1967) shows a contradiction in that there is a lower-bound to the probability of error that is exponentially larger than the hypothesized upper-bound.

This generalization of the sphere-packing bound involves modifying the bound to cover codes in which the transmitter is allowed N_1 uses of one channel and

N_2 uses of a second channel. When this generalized form of the sphere-packing bound is substituted into the lower bounding calculations of chapter II, the contradiction becomes apparent. The proof of the generalized sphere-packing bound is identical to the proof given by Shannon, Gallager and Berlekamp (1967) except that the fixed composition codes must cover both channels and the final removal of the fixed composition assumption must account for both channels. Since this extension of the sphere-packing bound is quite straight forward but tediously long, it will not be reproduced in this thesis.

3.5 Extension to Convolutional Encoders with Two Shift Registers

To this point, we have assumed that the convolutional encoder contains only one information shift register. Hence we have assumed that the rate of the code is

$$R = \frac{\ln(q)}{V} .$$

Let us now suppose that we wish to communicate $S(S<V)$ streams of information instead of one. We may modify the convolutional encoder by using S information storage registers instead of one. With this modified encoder, S information symbols enter the encoder per encoder shift. All S information storage registersare shifted together. A transmitted channel symbol is still a weighted sum of the contents of the information storage registers. If we let $i_d^{(s)}$ denote the d^{th} information symbol entering the s^{th} information storage register, Eq. (1.1) becomes

$$t_{v,d} = \sum_{b=1}^{K+1} \sum_{s=1}^{S} w_{v,d}^{(s)} i_{d+1-b}^{(s)} + r_{vd} \qquad 1 \leq v \leq V$$

where $w_{v,b}^{(s)}$ is the weight attached to the information symbol in the b^{th} stage of the s^{th} information storage register in determining the phase v channel symbol and $r_{v,d}$ is the appropriate member of the sequence $\underline{r}$.

We may prove a lemma like that in section 3.2 if we require

$$w_{1,s}^{(s)} = 1$$

for all s in the range $1 \leq s \leq S$ and if we require that the encoder weights are not restricted in such a way that $w_{v,b}^{(s)}$ must equal zero when $w_{v,b}^{(i)}$ need not equal zero for any $i \neq s$. This latter restriction is essentially a restriction that a given parity symbol either depends on the contents of the k^{th} stage of all shift registers or is independent of the contents the k^{th} stages of all information storage shift registers.

The proof of the lemma analogous to the lemma in section 3.2 follows the proof in section 3.2. The only change is that $X_{m''1}$, the set of channel symbols which are a one-to-one map of the possibly differing information symbols includes S channel symbols and S information symbols per encoder shift instead of just one channel symbol and one information symbol per shift.

In this modification of $X_{m''l}$, those channel symbols in $X_{m''r}$ which were transferred to $X_{m''l}$ are dropped from $X_{m''r}$. Once this change in diverged channel symbol classifications is made, the proof follows section 3.2. Since the proof in section 3.2 is notationally complicated, a slightly modified repetition of that proof would be tediously boring to read and impart little new knowledge of basic techniques. Thus, the proof of this modified version of the lemma will be omitted.

Chapter IV Sequential Decoding

4.0 Introduction

Chapters II and III present upper and lower bounds to the probability of erroneous communication for multiple generator length convolutional codes with optimum decoding. Unfortunately, optimum systems are often too expensive to build in a world of limited resources. The extreme cost of most optimum systems does not make analysis of the optimum system totally meaningless since there is much to be gained from knowing how a given system compares with the best possible. Wozencraft (1957) proposed a technique later modified by Fano (1963) which provides a practical algorithm for decoding convolutional codes. This sequential decoding algorithm has been studied extensively for equal generator length convolutional codes by Yudkin (1965), Niessen (1965), Savage (1965) and Falconer (1966). These studies of sequential decoding have shown that sequential decoding has the same upper-bound error exponent $E_U(R)$ as derived in Chapter III for optimum decoding. In this chapter, we examine sequential decoding for multiple generator length convolutional codes. The proofs given in this chapter will be limited to the case of systematic convolutional codes (k_1=0, all other k_v=K); however, section 4.4 will discuss the extension of the results derived here to the general case of multiple generator length convolutional codes. In upper-bounding the probability of error for systematic convolutional codes with sequential decoding, we find that $\overline{P(E)}$, the ensemble average probability of error, may be upper bounded as

$$\overline{P(E)} < \text{Const} \quad \exp{-K^*V}\, E_{Us}(R,B)$$

where

$$K^*V = k_1 + k_2 + k_3 + \ldots k_V = K(V-1).$$

The sequential decoding upper-bound error exponent $E_{Us}(R,B)$ is a function of the decoder parameter called bias B. $E_{Us}(R,B)$ is maximized for the same value of bias which minimizes average computation for equal generator length convolutional codes. On the other hand, we find that for systematic convolutional codes, $E_{Us}(R,B)$ is not maximized for the bias which minimizes the moments of computation. To the author's knowledge, this trade-off between error probability and computation in the sequential decoding of systematic convolutional codes is a new analytical result. Forney's simulations (1968) of sequential decoding show this trade-off between computation and error probability.

4.1 Sequential Decoding Algorithm

This section presents a brief summary of sequential decoding as presented by Gallager (1968). In keeping with the summary nature of this section, certain theorems will be stated without proof.

Sequential decoding stems from the idea of decoding the received message one information symbol at a time rather than decoding all information symbols simultaneously as in maximum likelihood decoding. The tree nature of the code facilitates this symbol by symbol decoding. For binary symbols, the first step in the tree (first information symbol to enter the encoder) must be either a binary one or a binary zero. If the decoder correctly decodes this first step, it will have only two possibilities to consider as second steps. If such step by step decoding were possible, the computation required to decode the message would be reduced because the decoder would not have to consider every message in its entirety. One of the problems with such a step by step decoder is that the decoder will occasionally make an incorrect decision at some step and go off the correct path. Unless the decoder is able to back up to reconsider previous decisions, such an incorrect decision will send the decoder permanently off the correct path.

An example will serve to illustrate this decoding idea and the problems inherent in it. Let us use the convolutional code discussed in the introduction for which the beginning portion of the channel symbol tree is shown in Fig. 1.2. For simplicity, let us assume that the channel is a binary symmetric channel. Thus, each channel symbol transmission is statistically independent of all other transmissions, and receiving the transmitted symbol is more likely than receiving its binary complement. If the first five information symbols are 10000, the channel sequence begins with 111 001 010 011 000 where a space indicates a shift of the encoder register. Suppose that the received symbol sequence begins with 110 001 010 111 000. At the first node, the decoder knows that either 111 or 000 was transmitted. Given that 110 is received, it is more likely that 111 was transmitted than 000. Thus, the decoder tentatively decides that the first information symbol is binary 1 which corresponds to the 111 transmission. Assuming that the first information symbol is a binary 1, the second set of three transmitted channel symbols must be either 001 or 110. Given that 001 was received, 001 is more likely to have been transmitted than 110. Now the decoder tentatively decides that the second information symbol is binary 0 corresponding to a 001 transmission.

Continuing in this manner, the decoder tentatively decodes the first five information symbols as 10000. On the other hand, suppose that the received sequence begins with 010 001 010 011 000. This time the decoder tentatively decides that the first information symbol is a binary 0. If the first information symbol is a binary 0, the second set of three transmitted channel symbols must be either 000 or 111. Since 001 was received, the decoder will tentatively decide that the second information symbol is binary 0. The decoder could continue and tentatively decide that the third information symbol is binary 0 and that the fourth information symbol is a binary 1. If these four hypothesized information symbols are correct, four channels errors must have occurred in twelve transmissions. This high error rate for the hypothesized message may be explained in one of two ways either the channel was abnormally noisy during the twelve transmissions or the hypothesized message is incorrect. The decoder should now begin to reconsider its past decisions. If it reconsiders its choice of the first information symbol, it will find an information sequence 10000 which implies only two errors in twelve transmissions. This later hypothesis is a more likely hypothesis which the decoder can reach after reconsidering its first tentative decoding decision.

The question of when the decoder should reconsider earlier decisions is all important. If the decoder reconsiders past decisions with great hesitancy, it will have to discard a large amount of work in backing up to reconsider earlier decisions. On the other hand, if the decoder reconsiders too quickly, it may discard correct tentative decisions and eventually have to reconsider the reconsideration.

Fano (1963) proposed a specific algorithm for determining when the decoder should back up to reconsider and when it should move further into the tree. This algorithm has been so widely used that it is now commonly called "the sequential decoder." Let $X_h = (x_{11}\cdots x_{1h}, x_{21}\cdots x_{Vh})$ be the first Vh digits of the channel sequence for some as yet unnamed message and $Y_h = (y_{11}\cdots y_{Vh})$ be the first Vh digits of the received symbol sequence. Define the function $\Gamma(X_h, Y_h)$ by

$$\Gamma(X_h, Y_h) = \sum_{i=1}^{h} \sum_{v=1}^{V} \left[\ln\left(\frac{P(y_{vi}/x_{vi})}{\omega(y_{vi})}\right) - B \right] \qquad (4.1.1)$$

where $\omega(j)$ is the nominal probability of the output j,

$$\omega(j) = \sum_{i} Q(i)P(j/i) \qquad (4.1.2)$$

and B is an arbitrary bias term to be selected later from the range $0 \leq B \leq C$. Let us call $\Gamma(X_h, Y_h)$ the value of the hypothesis X_h. If the resynchronization technique discussed in the introduction is used, decoding the message that corresponds to the X_{L+K} which maximizes $\Gamma(X_{L+K},Y_{L+K})$ gives an optimum decoder for memoryless channels. Since we want a decoder that demands less computation than the optimum decoder, we must rely upon other properties of the function $\Gamma(X_h,Y_h)$. If the Q(i)'s are the input probabilities which achieve channel capacity C, it can be shown that the expectation (over channel noise and code selection) of $\Gamma(X_h,Y_h)$ is hV(C-B) along the correct path and less than -hVB along any completely diverged incorrect path.

In terms of Γ, our suboptimum decoder is to hypothesize an X through the tree in such a way that $\Gamma(X_h,Y_h)$ increases with h. If Γ starts to decrease with increasing h, the decoder is probably on a wrong path and should go back to reexamine past decisions. The Fano sequential decoding algorithm is a set of rules for moving from one hypothesis to another. There are three basic moves forward, lateral and backward. On a forward move the decoder goes one branch to the right in the message tree; that is, the decoder hypothesizes the next symbol entering the encoder. Instrumentally, this corresponds to shifting the decoder's replica of the encoder one place to the right and inserting the hypothesized value of the next information symbol into the left end of the replica shift register. Since the new hypothesized message sequence differs from the previously hypothesized message sequence only by having the newest information symbol added to it, the new value of Γ can be easily found from the previous value of Γ by the equation

$$\Gamma(X_h,Y_h) = \Gamma(X_{h-1},Y_{h-1}) + \sum_{v=1}^{V} \left[\ln\left(\frac{P(y_{vh}/X_{vh})}{\omega(y_{vh})}\right) - B\right].$$

The digits involved in this calculation are simply the V channel input symbols coming out of the replica encoder and the channel symbols in the h^{th} group of V received channel symbols. On a lateral move, the decoder considers another possible hypothesis at the same depth (h-value) into the tree. On a backward move, the decoder goes one branch to the left in the message tree; that is, the decoder backs up to reconsider its hypothesis of the information symbol immediately preceding the information symbol which it was last considering. The new value of Γ may be calculated by subtracting off the last term in the h-summation expressed in Eq. (4.1.1). The algorithm used in moving from one node to another is Gallager's presentation (1968) of the algorithm due to Fano (1963). This algorithm is given as a set of rules in Fig. 4.1.1. The rules involve the value Γ_h of the node currently hypothesized, the value Γ_{h-1} of the node one step to the left of the current node and a threshold T. The value of T is constrained to change in increments of some fixed number Δ. The changes in T are determined by the algorithm.

The only boundary conditions are that the initial value of T be zero,\that $\Gamma_0=0$ (Γ at the starting node equal zero) and that $\Gamma_{-1}=-\infty$. This last boundary condition simply prevents the encoder from ever backing completely out of the tree.

Conditions on Node		Action to be Taken	
Previous Move	Comparison of Γ_{h-1} and Γ_h with initial threshold	Final Threshold	Move
F or L	$\Gamma_{h-1}<T+\Delta$, $\Gamma_h\geq T$	Raise*	F**
F or L	$\Gamma_{h-1}\geq T+\Delta$, $\Gamma_h\geq T$	No Change	F**
F or L	Γ_{h-1} arbitrary, $\Gamma_h<T$	No Change	L or B***
B	$\Gamma_{h-1}<T$, Γ_h arbitrary	Lower by Δ	F**
B	$\Gamma_{h-1}\geq T$, Γ_h arbitrary	No Change	L or B***

* Add j to threshold where j is chosen such that $T+j\Delta<\Gamma_h<T+(j+1)\Delta$

** Move forward to the first of the q nodes stemming from the current node (assuming some predetermined ordering of the q nodes).

*** Move laterally to next node differing from current node only in the final branch (assuming the same ordering as above): if the current node is the last of the q nodes, move backward.

Fig. 4.1.1 Rules for Decoder Motion

Fano (1963) discovered and Gallager (1968) has methematically proved several properties of the sequential decoding algorithm presented above. Let us define a descendant of the node X_h as a node to the right of X_h which is reached by a path that branches out from X_h. Hence, a descendant of X_h is a node reached by a path which coincides with X_h for the first h encoder shifts. Let us also define an F-hypothesis as a hypothesis for which the next move is forward. The first property of the algorithm is that for every node which is ever F-hypothesized, the final threshold T on this first F-hypothesis is related to the value Γ of the node by the inequality

$$T \leq \Gamma \leq T + \Delta$$

Moreover, the final threshold on each subsequent F-hypothesis of this node is Δ below the final threshold on the previous F-hypothesis of the node in question. Second, if the node X_h is hypothesized with final threshold T, then every descendant of X_h for which the path from X_h is above T must be F-hypothesized with final threshold T before X_h can be rehypothesized. The first property demonstrates that the algorithm does not loop in that no node can ever be hypothesized twice with the same threshold. The first and second properties combine to give us a way of determining the probability density function for the number of decoder moves necessary to decode a message.

4.2 Computation in Sequential Decoding

The intent of sequential decoding is to provide effective decoding with a device that is less complex than the maximum-likelihood decoder. The exact sequence of decoder moves is determined by the received sequence and the decoder algorithm. Thus, the number of decoder moves required to decode a block of L information symbols is a random variable. There can be at most q-1 lateral moves and one backward move for each forward move of the decoder. Thus, we may upper-bound sequential decoder computation by upper-bounding the number of F-hypotheses. Let W_0 be the number of F-hypotheses made from the origin node and from all incorrect nodes stemming from the origin node. A combination of a lower-bound derived by Jacobs and Berlekamp (1967) and an upper-bound derived by Falconer (1966) shows that the random variable W_0 has a Pareto distribution such that

$$\Pr(W_0>N) \approx N^{-a} \qquad (4.2.1)$$

for sufficiently large N when B=R and

$$R = \frac{E_0(a,Q)}{a} \qquad (4.2.2)$$

for $0 \leq a \leq 1$ when the channel is one of the channels for which the input assignment Q maximizes $E_0(a,\underline{Q})$ over $\underline{Q}$. Results by Jacobs and Berlekamp (1967) and Savage (1966) have led many observers to conjecture that Eqs. (4.2.1) and (4.2.2) hold for all a. The chief characteristic of the Pareto distribution on W_0 is that the r^{th} moment of W_0 is bounded for all $r<a$ and for no $r \geq a$. This characterization of the Pareto distribution leads us to desire a bound on the a^{th} moment of W_0.

For the finite constraint length convolutional encoder used here we must consider the problem of remergers. Previous discussions of computation in sequential decoding have assumed an infinite constraint length code which eliminates remergers. We would like to upper-bound the a^{th} moment of the number of computations made on the first correct node and all incorrect descendants of the first correct node. Remergers make such a computation difficult in that remergers allow the decoder to reach a correct node by following some path of incorrect nodes until a remerger occurs. The question arises as to whether we consider correct nodes reached by incorrect paths as "incorrect descendants" or "correct descendants". We will take the latter option here and redefine W_0

to be the number of F-hypotheses made on incorrect paths diverging at the first encoder shift before each of these paths merges with the correct path. This redefinition of W_0 does not lead to an absolutely tight upper-bound on computation because of the exponentially growing number of "correct descendants" or remerged nodes. It is conjectured that this redefinition of W_0 gives some reasonable estimate of computation per decoded information symbol despite the exponentially growing number of correct descendants. Experimental evidence obtained by Forney (1968) indicates that this conjecture is correct. Finally, this redefinition of W_0 leads to a result which is identical to that obtained for infinite constraint length non-systematic convolutional codes.

At a depth h into the tree there are a total of q^h nodes. One of these q^h nodes is the correct node and q^{h-K-1} are nodes which have merged with the correct path. With this new definition of W_0, the only nodes at depth h which we must consider are those nodes reached by a path which does not completely remerge with the correct path until h+1 or more steps into the tree. Let m' be some incorrect message subsequence which we must consider when bounding the number of computations in W_0 on nodes at depth h into the tree. The last information symbol at which m' and m_0 differ before the $(h+1)^{th}$ information symbol must enter the encoder at the h^{th} or $(h-1)^{th}$ or...or $(h-K)^{th}$ encoder shift. If the last information symbol at which m' and m_0 differ had entered the encoder before the $(h-K)^{th}$ shift, m' and m_0 would be completely merged at the h^{th} encoder shift contradicting the definition of m'. Let M_{hi} be the set of all incorrect message subsequences that diverge from m_0 at the first encoder shift, never completely remerge with m_0 until after the h^{th} encoder shift and for which the last differing information symbol prior to the $(h+1)^{th}$ encoder shift enters the encoder at the $(h-i)^{th}$ encoder shift. If W_{0hi} denotes the number of F-hypotheses made on nodes at depth h into the tree reached by incorrect message paths in M_{hi},

$$W_0 = \sum_{h=0}^{\infty} \sum_{i=0}^{K} W_{0hi}.$$

The number W_0 is a random variable dependent on both the channel noise and the code selected. We will avoid the problem of code selection by taking a statistical average over both the channel noise and the ensemble of all possible codes. This ensemble of codes is the set of all convolutional codes for which $k_2 = k_3 = \ldots = k_V = K$, $w_{11} = 1$, $k_1 = 0$ and all other non-trivial encoder weights are randomly reselected after each encoder shift. Generalizing a proof first presented by Falconer (1966) we will derive an upper-bound on the a^{th} moment of the random variable W_0 for a such that $0 \le a \le 1$. A standard inequality shows that

$$\overline{\left(\sum x_i\right)^a} \le \sum \overline{(x_i)^a} \qquad (4.2.3)$$

for all a such that $0 \le a \le 1$. Thus,

$$\overline{W_0^{\,a}} = \overline{\left(\sum_{h=0}^{\infty}\ \sum_{i=0}^{K} W_{0hi}\right)^a} \le \sum_{h=0}^{\infty}\ \sum_{i=0}^{K} \overline{(W_{0hi})^a} \ . \qquad (4.2.4)$$

We must now derive an upper-bound on $\overline{(W_{0hi}^{\ a})}$. The two properties of the decoding algorithm proved by Gallager may be combined to show that a given incorrect node at depth h may be F-hypothesized for the j^{th} time only if

$$\Gamma_{m'(h)} \ge \Gamma^{o}_{min} + (j-2)\Delta \qquad (4.2.5)$$

where $\Gamma_{m'(h)}$ is the value Γ of the incorrect node m' at depth h and Γ^{o}_{min} is the minimum of Γ along the whole correct path. We will subsequently denote $\Gamma_{m'(h)}$ as simply Γ'_h. Equation (4.2.5) is true because the incorrect node m' at depth h must be F-hypothesized first with a final threshold T such that

$$T\le\Gamma_h\le T+\Delta \ .$$

At each subsequent F-hypothesis of m', the final threshold is lower by Δ than the previous final threshold. Once the threshold has been lowered below Γ^{o}_{min} the entire correct path must be hypothesized before the threshold is lowered again. If the entire correct path is hypothesized, decoding stops and the threshold goes no lower. Thus m' can be hypothesized only once after the threshold is lowered below Γ^{o}_{min}. Hence m' can be hypothesized the j^{th} time only if

$$\frac{\Gamma'_h+\Delta-\Gamma^{o}_{min}}{\Delta} \ge (j-1)$$

which is equivalent to the form in (4.2.5).

Let us define

$$\phi_d(\Gamma_h{}', \Gamma_d{}^o, j) = \begin{cases} 1 \text{ if } \Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta \geq 0 \\ 0 \text{ otherwise} \end{cases} \qquad (4.2.5)$$

where $\Gamma_d{}^o$ is the value for the d th node of the correct path $\underline{X}_0$.

$$\Gamma_d{}^o = \Gamma(\underline{X}_{0d}, Y_d).$$

Summing over all m' in M_{hi}, we find that

$$W_{0hi} \leq \sum_{m' \epsilon M_{hi}} \sum_{j=1}^{\infty} \phi_d(\Gamma_h{}', \Gamma_d{}^o, j)$$

where d is selected such that

$$\Gamma_d{}^o = \Gamma_{min}{}^o.$$

Since d is a random variable, we are faced with the problem of selecting the right value of d. This problem of finding the correct d is eliminated if we include all d in the summation; thus upper-bounding W_{0hi}.

$$W_{0hi} \leq \sum_{j=1}^{\infty} \sum_{d=0}^{\infty} {}_{m'\epsilon M_{hi}} \phi_d(\Gamma_h{}', \Gamma_d{}^o, j).$$

Using inequality (4.2.3) on the j-summation and the d-summation, we find that

$$\overline{(W_{0hi})^a} \leq \sum_{j=1}^{\infty} \sum_{d=0}^{\infty} \overline{\left[\sum_{m'\epsilon M_{hi}} \phi_d(\Gamma_h{}', \Gamma_d{}^o, j)\right]^a} .$$

Since

$$\phi_d(\Gamma_h{}', \Gamma_d{}^o, j) \leq \exp s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]$$

for all $s \geq 0$,

$$\overline{(W_{0hi})^a} \leq \sum_{j=1}^{\infty} \sum_{d=0}^{\infty} \overline{\left\{\sum_{m'\epsilon M_{hi}} \exp s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]\right\}^a} . \tag{4.2.6}$$

Let us examine the expectation on the right-hand side of inequality (4.2.6).

$$\overline{\left\{\sum_{m'\epsilon M_{hi}} e^{s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]}\right\}^a} = \sum_{\underline{Y}} \sum_{\underline{X}_{m_0}} \sum_{m_0} P(\underline{Y}/\underline{X}_{m_0} m_0)$$

$$\times P(\underline{X}_{m_0}/m_0) P(m_0)$$

$$\times E\left(\left\{ \sum_{m'\epsilon M_{hi}} e^{s(\ldots)} \right\}^a / YX_{m_0} m_0 \right) .$$

The conditional expectation E is over the choice of channel sequences for all the message sequences m' in M_{hi}. Since z^a is a convex $\cap$ function of positive z for $0 \leq a \leq 1$,

$$E(z^a) \leq \left[E(z)\right]^a .$$

Thus,

$$\overline{\left\{\sum_{m'\epsilon M_{hi}} e^{s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]}\right\}^a} \leq \sum_{\underline{Y}} \sum_{\underline{X}_{m_0}} \sum_{m_0} P(\underline{Y}/\underline{X}_{m_0} m_0) P(\underline{X}_{m_0}/m_0) P(m_0)$$

$$\times E\left[\sum_{m'\epsilon M_{hi}} e^{s(\ldots)} / \underline{YX}_{m_0} m_0 \right]^a$$

Finally since the expectation of a sum is the sum of the expectations,

$$\overline{\left\{\sum_{m'\epsilon M_{hi}} e^{s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]}\right\}^a} \leq \sum_{\underline{Y}} \sum_{\underline{X}_{m_0}} \sum_{m_0} P(\underline{Y}/\underline{X}_{m_0} m_0) P(\underline{X}_{m_0}/m_0) P(m_0)$$

$$\times \left\{\sum_{m'\epsilon M_{hi}} \sum_{\underline{X}_{m'}} \underline{P}(X_{m'}/\underline{YX}_{m_0} m_0) e^{s(\ldots)}\right\}^a .$$

Thus,

$$\overline{(W_{0hi})}^a \leq \sum_{j=1}^{\infty} \sum_{d=0}^{\infty} \sum_{\underline{Y}} \sum_{\underline{X}_{m_0}} \sum_{m_0} P(\underline{Y}/\underline{X}_{m_0} m_0) P(\underline{X}_{m_0}/m_0) P(m_0)$$

$$\left\{\sum_{m'\epsilon M_{hi}} \sum_{X_{m'h}} P(X_{m'h}/\underline{YX}_{m_0} m_0)\; e^{s\left[\Gamma_h{}' - \Gamma_d{}^o - (j-2)\Delta\right]}\right\}^a . \qquad (4.2.7)$$

Further simplification of the right-hand side of inequality (4.2.7) closely parallels the steps used in section 3.2. In this section, we will stress those points at which the arguments differ and skip lightly over those points of the argument which are identical to those in section 3.2. We have restricted our attention to systematic convolutional codes (k_1=0, all other k_v=K). Here it will be convenient to divide the symbols of $X_{m'h}$ into three groups: (i) $X_{m'p}$, those h(V-1) diverged phase 2 through phase V channel symbols at which $X_{m'}$ is equally likely to be any q-ary symbol independent of m_0, $\underline{X}_{m_0}$ and the rest of $\underline{X}_{m'}$, (ii) $X_{m's}$, those (h-i) systematic (phase 1) channel symbols which are a one-to-one map of the information sequences in m' for any given code and (iii) $X_{m't}$, those i phase 1 channel symbols which must be identical to the corresponding symbol of X_{m_0} for all m' in M_{hi}. The symbols in $X_{m's}$ are the first h-i phase 1 channel symbols generated and those in $X_{m't}$ are the last i phase 1 channel symbols generated before the (h+1)th encoder shift. Combining the basic properties of the three different groups of symbols in $X_{m'h}$ and the requirement that the codewords be independent of the received channel symbols, we find that,

$$P(X_{m'h}/\underline{YX}_{m_0}m_0) = Q(X_{m'p})P(X_{m's}/X_{m_0}m_0)\ \delta(X_{m't},X_{m_0t})$$

where Q( ) is the probability distribution in which all sequences are equally likely (see section 3.2) and

$$\delta(X_{m't},X_{m_0t}) = \begin{cases} 1 & \text{if } X_{m't}=X_{m_0t} \\ 0 & \text{otherwise.} \end{cases}$$

For any specific code, the one-to-one map from m' sequences into $X_{m's}$ makes the m'-summation in the right-hand side of inequality (4.2.7) just a summation over a set of non-identical $X_{m's}$ terms. The right-hand side of inequality (4.2.7) is not decreased if the summation over $X_{m's}$ terms is increased to include all $X_{m's}$ terms. Finally, $\underline{X}_{m_0}$ is equally likely to be any q-ary sequence independent of m_0. Since

$$Q(X_{m's}) = q^{-(h-i)},$$

we may combine the preceeding arguments to show that,

$$\overline{(W_{0hi})^a} \leq \sum_{j=1}^{\infty} e^{-(j-2)sa\Delta} \sum_{d=0} \sum_{\underline{Y}} \sum_{\underline{X}_{m_0}} P(\underline{Y}/\underline{X}_{m_0})Q(\underline{X}_{m_0}) \left\{ q^{(h-i)} \sum_{X_{m'p}} \sum_{X_{m's}} \sum_{X_{m't}} Q(X_{m's})Q(X_{m'p})\delta(X_{m't},X_{m_0t}) \times e^{s\left[\Gamma_h' - \Gamma_d^o\right]} \right\}^a . \tag{4.2.8}$$

Inequality (4.2.8) is further simplified by treating the sequences $\underline{X}_{m'}$, $\underline{X}_{m_0}$ and $\underline{Y}$ on a symbol by symbol basis. As in section 3.2,

$$Q(X_{m_0}) = \prod_{n=1}^{L+K} \prod_{v=1}^{V} Q(x^{o}_{vn}).$$

The memoryless channel assures that

$$P(\underline{Y}/\underline{X}_{m_0}) = \prod_{n=1}^{L+K} \prod_{v=1}^{V} P(y_{vn}/x^{o}_{vn}).$$

Finally, defining

$$P_{vn}(x'_{vn}/x^{o}_{vn}) = \begin{cases} Q(x'_{vn}) & \text{if vn pair indicates a symbol in } X_{m's} \text{ or } X_{m'p} \\ \delta(x'_{vn},x^{o}_{vn}) & \text{if vn pair indicates a symbol in } X_{m't} \end{cases},$$

we may write

$$Q(X_{m's})Q(X_{m'p})\delta(X_{m't},X_{m_0 t}) = \prod_{v=1}^{V} \prod_{n=1}^{h} P_{vn}(x'_{vn}/x^{o}_{vn}).$$

There are two cases $h \geq d$ and $h \leq d$ which we must consider in simplifying the right-hand side of inequality (4.2.8). Let us first consider the case $h \leq d$. The expectations in those terms in the right hand side of inequality (4.2.8) for which $h \leq d$ may be written as

$$\sum_{y_{11}} \cdots \sum_{y_{Vd}} \sum_{x^o_{11}} \cdots \sum_{x^o_{Vd}} \prod_{n=1}^{d} \prod_{v=1}^{V} Q(x_{vn}) P(y_{vn}/x_{vn}) \left[\frac{\omega(y_{vn})}{P(y_{vn}/x^o_{vn})} e^{B} \right]^{sa}$$

$$q^{a(h-i)} \left\{ \sum_{x'_{11}} \cdots \sum_{x'_{Vh}} \prod_{n=1}^{h} \prod_{v=1}^{V} P_{vn}(x'_{vn}/x_{vn}) \left[\frac{P(y_{vn}/x'_{vn})}{\omega(y_{vn})} e^{-B} \right]^{s} \right\}^{a} .$$

Interchanging the order of summation and multiplication and performing some algebra, we find that for $h \leq d$ the expectation terms in the right-hand side of inequality (4.2.8) may be expressed as

$$q^{a(h-i)} \prod_{v=1}^{V} \prod_{n=1}^{h} \sum_{y_{vn}} \sum_{x^o_{vn}} Q(x^o_{vn}) P(y_{vn}/x^o_{vn})^{1-sa} \times \left\{ \sum_{x'_{vn}} P_{vn}(x'_{vn}/x_{vn}) P(y_{vn}/x'_{vn})^{s} \right\}^{a}$$

$$\times \prod_{v=1}^{V} \prod_{n=h+1}^{d} \sum_{y_{vn}} \sum_{x'_{vn}} Q(x^o_{vn}) P(y_{vn}/x^o_{vn})^{1-sa} \omega(y_{vn})^{sa} e^{saB} .$$

At those i vn-pairs for which $P_{vn}(x^o_{vn}/x'_{vn}) = \delta(x^o_{vn}, x'_{vn})$

$$\sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa} \left[\sum_{x'} P_{vn}(x'/x^{o})P(y/x')^{s}\right]^{a}$$

$$=\sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa}P(y/x^{o})^{sa} = 1 .$$

(4.2.9)

Holder's inequality states that for two random variables

$$\overline{uw} \leq \left(\overline{u^{\nu_a}}\right)^{\frac{1}{\nu_a}} \times \left(\overline{w^{\nu_b}}\right)^{\frac{1}{\nu_b}}$$

where ν_a and ν_b are positive numbers such that

$$\frac{1}{\nu_a} + \frac{1}{\nu_b} = 1.$$

Restricting s such that $0<sa<1$ and using Holder's inequality on the y-summation, we may upper-bound those terms in the first product for which $P_{vn}(x'_{vn}/x_{vn}) = Q(x'_{vn})$.

$$\sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa} \left[\sum_{x'} P_{vn}(x'/x^{o})P(y/x')^{s}\right]^{a}$$

$$\leq \left(\sum_{y}\left[\sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa}\right]^{\frac{1}{1-sa}}\right)^{1-sa}$$

$$\times \left(\sum_{y}\left[\sum_{x'} Q(x')P(y/x)^{s}\right]^{a\frac{1}{sa}}\right)^{sa}$$

$$= \exp-\left[(1-sa)E_{0}\left(\frac{sa}{1-sa},Q\right)+saE_{0}\left(\frac{1-s}{s},Q\right)\right]$$

(4.2.10)

where $E_0(\rho,Q)$ was defined in section 3.2 as

$$E_{0}(\rho,Q) = -\ln \sum_{y}\left(\sum_{x} Q(x)P(y/x)^{\frac{1}{1+\rho}}\right)^{1+\rho}$$

Holder's inequality may again be used on the y-summation to upper-bound those terms involving $Q(x^o)$, $P(y/x^o)$ and $\omega(y)$ in the second product.

$$\sum_{y}\sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa}\,\omega(y)^{sa}\,e^{saB}$$

$$\leq e^{saB}\left[\sum_{y}\left(\sum_{x^{o}} Q(x^{o})P(y/x^{o})^{1-sa}\right)^{\frac{1}{1-sa}}\right]^{1-sa}\times\left[\sum_{y}\left(\omega(y)^{sa}\right)^{\frac{1}{sa}}\right]^{sa}$$

$$=\exp-\left[(1-sa)E_{0}(\frac{sa}{1-sa},Q)-saB\right]. \qquad (4.2.11)$$

It can be varified for the binary symmetric channel that these uses of Holder's equality are satisfied with equality. We may combine inequalities (4.2.9), (4.2.10) and (4.2.11) to show that the expectation terms on the right-hand side of inequality (4.2.8) for which $h\leq d$ may be upper-bounded as

$$q^{a(h-i)}\ \exp -(hV-i)\left[(1-sa)E_{0}(\frac{sa}{1-sa},Q)+saE_{0}(\frac{1-s}{s},Q)\right]$$

$$\times\ \exp -(d-h)V\left[(1-sa)E_{0}(\frac{sa}{1-sa},Q)-saB\right].$$

Let us now consider the expectation terms in the right-hand side of inequality (4.2.8) for which $h\geq d$, Techniques similar to those used above show that for $h\geq d$, the expectation term in the right-hand side of inequality (4.2.8) may be written as

$$q^{a(h-i)} \prod_{n=1}^{d} \prod_{v=1}^{V} \left[\sum_{y_{vn}} \sum_{x_{vn}} Q(x^{o}_{vn})P(y_{vn}/x^{o}_{vn})^{1-sa} \right.$$

$$\left. \times \left(\sum_{x'_{vn}} P_{vn}(x'_{vn}/x^{o}_{vn})P(y_{vn}/x'_{vn})^{s} \right)^{a} \right]$$

$$\times \prod_{n=d+1}^{h} \prod_{v=1}^{V} \left[\sum_{y_{vn}} \sum_{x^{o}_{vn}} Q(x^{o}_{vn})P(y_{vn}/x^{o}_{vn})\omega(y_{vn})^{-sa} \right.$$

$$\left. \times \left(\sum_{x'_{vn}} P_{vn}(x'_{vn}/x^{o}_{vn})P(y_{vn}/x'_{vn})^{s} e^{-sB} \right)^{a} \right],$$

Expressions (4.2.9) and (4.2.10) allow simplification of each vn-term in the first product term. Again, we may use Holder's inequality on the y-summation of those vn-terms in the second product term for which $P_{vn}(x'_{vn}/x_{vn}) = Q(x'_{vn})$. Remembering that

$$\omega(y) = \sum_{i} Q(i)P(y/i),$$

we may upper-bound these terms as

$$\sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o})\omega(y)^{-sa} \left(\sum_{x'} P_{vn}(x'/x^{o})P(y/x')^{s}e^{-sB}\right)^{a}$$

$$\leq \left[\sum_{y}\left(\omega(y)^{1-sa}\right)^{\frac{1}{1-sa}}\right]^{1-sa} e^{-saB} \times \left[\sum_{y}\left(\sum_{x'} Q(x')P(y/x')^{s}\right)^{a\frac{1}{sa}}\right]^{sa}$$

$$= \exp -\left[saE_{0}\left(\frac{1-s}{s},Q\right)+saB\right] .$$

(4.2.12)

Finally, we must deal with those terms in the second product for which $P_{vn}(x'_{vn}/x^{o}_{vn})=\delta(x'_{vn},x^{o}_{vn})$. Since there are a total of i terms in the two products for which $P_{vn}(x'/x^{o})=\delta(x',x^{o})$, we may state that for some t_d $0\leq t_d\leq i$ there are exactly t_d terms in the second product for which $P_{vn}(x'/x^{o})=\delta(x'/x^{o})$. Thus there are $i-t_d$ terms in the first product for which $P_{vn}(x'/x^{o})=\delta(x',x^{o})$. For those t_d terms in the second product for which $P_{vn}(x'/x^{o})=\delta(x',x^{o})$.

$$\sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o})\omega(y)^{-sa} \times \left(\sum_{x'} P_{vn}(x'/x^{o})P(y/x')^{s}e^{-sB}\right)^{a}$$

$$= e^{-saB} \sum_{y} \sum_{x^{o}} Q(x^{o})P(y/x^{o}) \left[\frac{P(y/x^{o})}{\omega(y)}\right]^{sa}$$

$$= \exp - \left[\mu(sa)+saB\right] \qquad (4.2.13)$$

where

$$\mu(sa) = -\ln\left[\sum_{y}\sum_{x} Q(x)P(y/x)\left(\frac{P(y/x)}{\omega(y)}\right)^{sa}\right].$$

Substituting relations (4.2.9) through (4.2.13) into the right-hand side of inequality (4.2.8), we find that for h≥d

$$\overline{(W_{0hi})^{a}} \leq q^{a(h-i)} \sum_{d=0}^{\infty} \sum_{j=1}^{\infty} e^{-sa(j-2)\Lambda}$$

$$\exp -\left[dV-(i-t_d)\right]\left[(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)+saE_0\left(\frac{1-s}{s},Q\right)\right]$$

$$\exp -\left[(h-d)V-t_d\right]\left[saE_0\left(\frac{1-s}{s},Q\right)+saB\right]$$

$$\exp -t_d\left[\mu(sa)+saB\right] . \qquad (4.2.14)$$

Upon further examination, we find that the upper-bounds for the expectation terms in the right-hand side of inequality (4.2.8) are identical for h≤d and h≥d since $t_d=0$ when h≥d.

The right-hand side of inequality (4.2.8) is not decreased if we multiply the term for h≤d by the maximum of one and the largest t_d term in the expectation term for h≥d.

Defining

$$E_m(sa) = \min \begin{cases} 0 \\ \mu(sa) + (1-sa)E_0(\frac{sa}{1-sa},Q), \end{cases} \qquad (4.2.15)$$

we find that the right-hand side of inequality (4.2.8) may be upper-bound to give

$$\overline{(W_{0hi})^a} \le \sum_{j=1}^{\infty} e^{-(j-2)sa\Delta} q^{a(h-i)} \exp -hV\left[saE_0(\frac{1-s}{s},Q)+saB\right]$$

$$\exp -t_d E_m(sa) \quad \exp +i\left[(1-sa)E_0(\frac{sa}{1-sa},Q)+saE_0(\frac{1-s}{s},Q)\right]$$

$$\sum_{d=0}^{\infty} \exp -dV\left[(1-sa)E_0(\frac{sa}{1-sa},Q)-saB\right] . \qquad (4.2.16)$$

Further simplification occurs when we use the equation

$$R = \frac{\ln(q)}{V} .$$

Since the term $E_m(sa)$ is non-positive, we may upper-bound the right-hand side of inequality (4.2.16) by using the largest possible value of t_d; namely, $t_d=i$. Finally, we show in an appendix that

$$\mu(sa) + (1-sa)E_0(\frac{sa}{1-sa},Q) \leq 0$$

for all sa. After performing the j-summation, we find that,

$$\overline{W_{0hi}}^{a} \leq \frac{e^{sa\Delta}}{1-e^{-sa\Delta}} \exp-hV\left[saE_0(\frac{1-s}{s},Q)+saB-aR\right]$$

$$\times q^{-ia}\exp-i\left\{\mu(sa) -saE_0(\frac{1-s}{s},Q)\right\}$$

$$\times \sum_{d=0}^{\infty} \exp-dV\left[(1-sa)E_0(\frac{sa}{1-sa},Q)-saB\right] .$$

For future reference we have enclosed in braces $\{\}$ those terms in the upper-bound on $\overline{W_{0hi}}^{a}$ which result from channel symbols at which m' and m_0 are partially merged (phase 1 merged all other phase diverged). At several later points, we will set the contents of the braces to zero in order to examine the result for the equal generator length convolutional code ($k_1=k_2=\ldots=k_V=K$).

Returning to inequality (4.2.4) we find that

$$\overline{W_0^a} \leq \frac{e^{sa\Delta}}{1-e^{-sa\Delta}} \sum_{i=0}^{K} q^{-ia} \exp -i\left[\mu(sa)E_0\left(\frac{1-s}{s},Q\right)\right]$$

$$\sum_{h=0}^{\infty} \exp -hV\left[saE_0\left(\frac{1-s}{s},Q\right)+saB-aR\right]$$

$$\sum_{d=0}^{\infty} \exp -dV\left[(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)-saB\right] \tag{4.2.17}$$

Since the i-summation in inequality (4.2.17) is a finite sum, it is always finite and bounded. Thus, $\overline{W_0^a}$ is bounded if both the d-summation and the h-summation are bounded. The d-summation and the h-summation are just geometric series which are bounded if

$$(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)-saB = \epsilon_1 > 0$$

and

$$saE_0\left(\frac{1-s}{s},Q\right)+saB-aR = \epsilon_2 > 0 \quad .$$

We may summarize by stating a theorem which we have just proved. Let W_0 be the number of sequential decoder hypotheses made on incorrect paths diverging at the origin before these paths completely remerge with the correct path, then $\overline{W_0^a}$ is bounded for $0 \leq a \leq 1$ if

$$R < sE_0\left(\frac{1-s}{s}, Q\right) + sB \qquad (4.2.18)$$

and

$$B < \frac{E_0\left(\frac{sa}{1-sa}, Q\right)}{\frac{sa}{1-sa}} \qquad (4.2.19)$$

for some s such that $0 < as < 1$.

Setting $s = \frac{1}{1+a}$ and $B=R$, we find that the two conditions for boundedness of $\overline{W_0^a}$ become identical and that $\overline{W_0^a}$ is bounded for a in the range $0 \leq a \leq 1$ if

$$R < \frac{E_0(a,Q)}{a} .$$

This special case for B=R agrees with a Falconer's (1966) result for infinite constraint length convolutional codes. As mentioned earlier in this section, Jacobs and Berlekamp (1967) have derived a lower bound to sequential decoder computation which states that the

a^{th} moment of W_0 is unbounded if

$$R \geq \frac{E_0(a)}{a}$$

where $E_0(a)$ is the maximum over all possible $\underline{Q}$ of the function $E_0(a,\underline{Q})$. For symmetric channels $E_0(a) = E_0(a,Q)$, and the result derived here is exponentially tight for B=R. As far as the author knows, this thesis is the first work dealing with the a^{th} moment of computation in sequential decoding with $B \neq R$. Yudkin's thesis (1965) dealt with generalized bias terms but only for first moments of computation with equal generator length codes. Falconer's thesis (1966) dealt with all a for $0 \leq a \leq 1$ but only for B=R. For equal generator length convolutional codes, B=R gives an optimum result. In the next section, we will illustrate circumstances in which we may wish to use a bias that is unequal to the rate.

We may find the largest value of a in the range $0 \leq a \leq 1$ for which the a^{th} moment of W_0 is bounded by finding the largest sa for which inequality (4.2.19) is satisfied and the smallest s for which inequality (4.2.18) is satisfied. Dividing the maximum value of sa by the minimum value of s gives the maximum possible value of a for which the a^{th} moment of W_0 is bounded. If the calculated maximum value of a is greater than one, we must acknowledge the restriction that a be less than or equal to one. From the Pareto nature of the random variable W_0 we may conclude that

$$Pr(W_0>N) \approx N^{-(a_{max})}.$$

A computer program was written to evaluate a_{max} for several bias levels on a binary symmetric channel with R=.5 bit. Forney (1968) has performed some computer simulations of sequential decoding with $B \neq R$. The table in figure 4.2.1 compares the simulation value of a_{max} with the value of a_{max} calculated from the theory developed here. In compiling figure 4.2.1, we have conjectured that the restriction that $0 \leq a \leq 1$ may be removed. We have been unable to prove this conjecture; however, the results obtained using this conjecture are encouraging. For those a_{max} less than one, the theoretical development presented here predicts the simulated value of a_{max} more closely than any other theoretical result known to the author.

BSC crossover probability	Bias	a_{max} theoretical	a_{max} measured
9/256	.470	1.26 c	1.29
9/256	.550	1.24 c	1.29
10/256	.480	1.15 c	1.15
10/256	.557	1.11 c	1.12
11/256	.489	1.05 c	1.06
11/256	.564	.98	.95
12/256	.496	.95	.96
12/256	.569	.86	.88

Figure 4.2.1: Comparsion of measured and theoretical value of the pareto exponent a_{max} for a binary symmetric channel with V=2 and R=.5. The letter "c" follows those theoretical a_{max} which are the result of conjecture rather than proved theorems.

BSC crossover probability	Bias	a_{max} theoretical	a_{max} measured
9/256	.470	1.26 c	1.29
9/256	.550	1.24 c	1.29
10/256	.480	1.15 c	1.15
10/256	.557	1.11 c	1.12
11/256	.489	1.05 c	1.06
11/256	.564	.98	.95
12/256	.496	.95	.96
12/256	.569	.86	.88

Figure 4.2.1: Comparsion of measured and theoretical value of the pareto exponent a_{max} for a binary symmetric channel with V=2 and R=.5. The letter "c" follows those theoretical a_{max} which are the result of conjecture rather than proved theorems.

A geometric construction allows us to find the limiting values of s and sa in inequalities (4.2.18) and (4.2.19). Figure 4.2.2 shows a plot of the function $E_0(\rho,Q)$ for $\rho\geq 0$. Consider the point (-1,-B). Select a point $\rho=\frac{1-s}{s}$ on the ρ-axis. Draw a straight line connecting the points (-1,-B) and $\left[\frac{1-s}{s}, E_0(\frac{1-s}{s},Q)\right]$. The slope of this line is just

$$\frac{E_0(\frac{1-s}{s},Q)\ +B}{1+\ \frac{1-s}{s}} = sE_0(\frac{1-s}{s},Q)\ +sB.$$

Thus, the slope of this line is the quantity in the right-hand side of inequality (4.2.18). For this value of s, inequality (4.2.18) is satisfied for all R less than the slope of the line connecting the points (-1,-B) and $\left[\frac{1-s}{s}, E_0(\frac{1-s}{s},Q)\right]$. Hence for a given R, the smallest value of s (largest ρ) for which inequality (4.2.18) holds is that value of s corresponding to the straight line through the point (-1,-B) with slope just greater than R. Having found the minimum value of s, let us find the maximum value of sa for which inequality (4.2.19) is satisfied. Consider the straight line of slope B passing through the origin. The intersection of this straight line

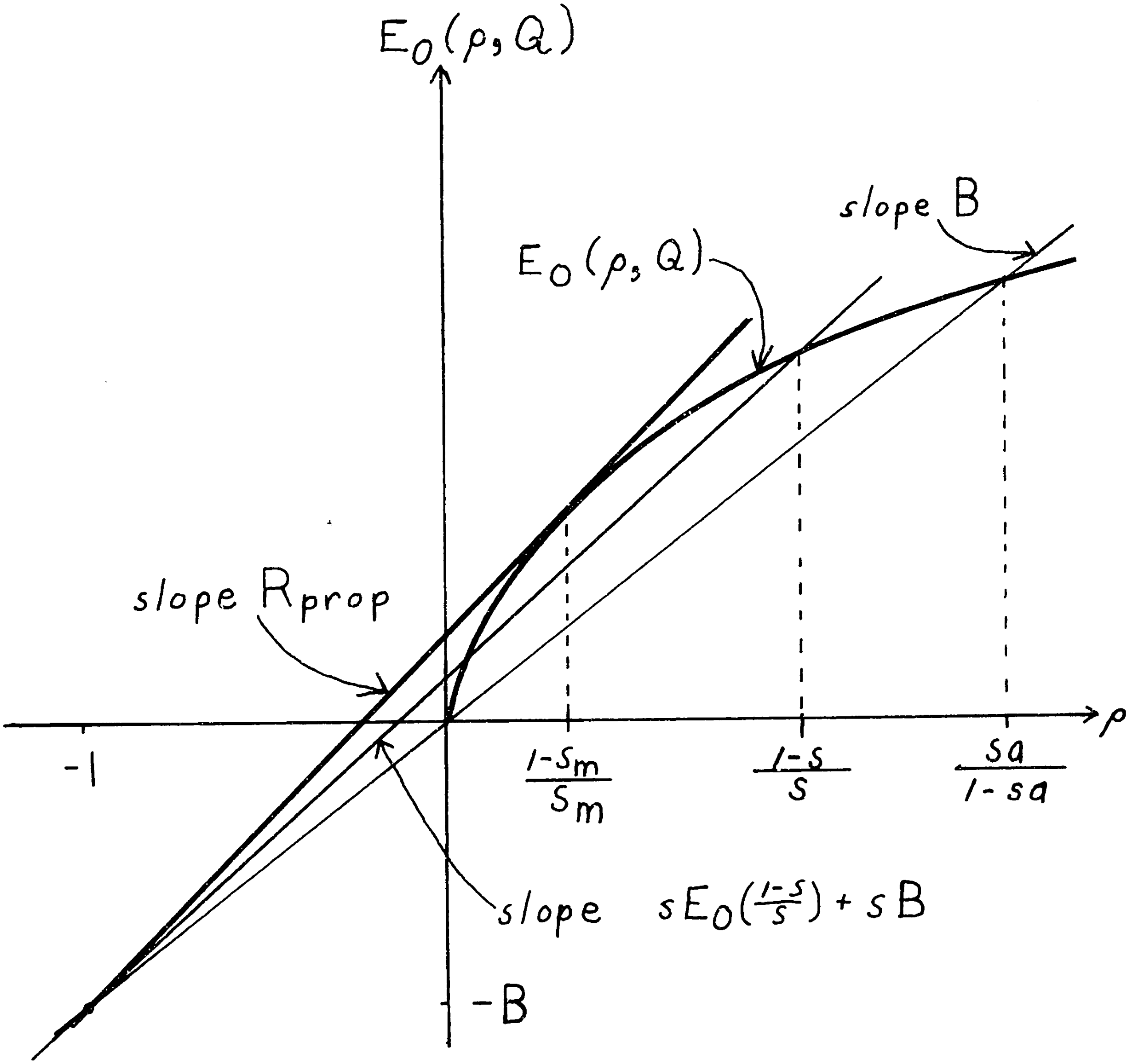

FIG. 4.2.2 R_{prop} CONSTRUCTION

and the $E_0(\rho,Q)$ curve occurs at the point at which

$$\rho B = E_0(\rho,Q).$$

Setting $\rho = \frac{sa}{1-sa}$, we find that this intersection occurs at that sa for which

$$B = \frac{E_0(\frac{sa}{1-sa},Q)}{\frac{sa}{1-sa}} .$$

Hence for given B, the largest value of sa (largest ρ) which satisfies inequality (4.2.19) is the value of sa at the intersection of the curve $E_0(\frac{sa}{1-sa},Q)$ and the straight line through the origin with slope just greater than B.

We may interpret inequality (4.2.18) as stating that error propagation occurs whenever,

$$R \geq R_{prop} = \max_{s>0} \left[sE_0(\frac{1-s}{s},Q)+sB\right] . \qquad (4.2.20)$$

We may use the construction above to find $R_{prop.}$ The quantity in brackets in the right-hand side of (4.2.20) is just the slope of the straight line connecting the points (-1,-B) and $\left[\frac{1-s}{s},E_0(\frac{1-s}{s},Q)\right]$. Hence the maximum over s in the right-hand side of (4.2.20) is the slope of the steepest line intersecting the $E_0(\rho,Q)$ curve and passing through the point (-1,-B). If the $E_0(\rho,Q)$ curve is smooth, this steepest line is a tangent to the $E_0(\rho,Q)$ curve. At s_m, the maximizing value of s,

$$E_0'\left(\frac{1-s_m}{s_m},Q\right) = s_m E_0\left(\frac{1-s_m}{s_m},Q\right) + s_m B.$$

Multiplying both side of the above equation by $(1-s_m)/s_m$, we find that

$$E_0\left(\frac{1-s_m}{s_m},Q\right) - \left(\frac{1-s_m}{s_m}\right)E_0'\left(\frac{1-s_m}{s_m},Q\right) = s_m E_0\left(\frac{1-s_m}{s_m},Q\right) + s_m B - B. \tag{4.2.21}$$

The right-hand side of Eq. (4.2.21) is just $R_{prop}-B$. For those channels in which $E_0(\rho,Q)$ is the maximum of $E_0(\rho,\underline{Q})$ over all probability assignments $\underline{Q}$, the left-hand side of Eq. (4.2.21) is just the sphere packing exponent derived by Shannon, Gallager and Berlekamp (1967). Symmetric channels are included in the set of channels for which $E_0(\rho,Q)$ is the maximum over all $\underline{Q}$ of $E_0(\rho,\underline{Q})$. Hence for symmetric channels,

$$E_{sp}(R_{prop}) = R_{prop} - B. \tag{4.2.22}$$

where $E_{sp}(R)$ is the sphere-packing exponent derived by Shannon, Gallager and Berlekamp (1967). In figure 4.2.3, R_{prop} is the value of R at the intersection of the curves $E_{sp}(R_{prop})$ and $R_{prop}-B$. Using constructions such as that in figure 4.2.3, we may determine the minimum bias necessary to achieve a given value of R_{prop}.

A computer program was written to evaluate R_{prop} as a function of B for a binary symmetric channel. Figure 4.2.4 shows a plot of R_{prop} as a function of B for a binary symmetric channel with crossover probability 3/64.

The above theorem on the moments of W_0 may be extended to allow the node of initial divergence to be the n^{th} node on the correct path rather than just the first node on the correct path. The statistical description of the tree stemming from any node on the correct path is identical to the statistical description of the origin node except that all the Γ values have a constant added to them. The lemma on the number of computations at a node is unchanged and the proof is the same regardless of the node at which the divergence begins. This bound on $\overline{W_n^a}$ does not strictly lead to a bound on the distribution of computation per decoded

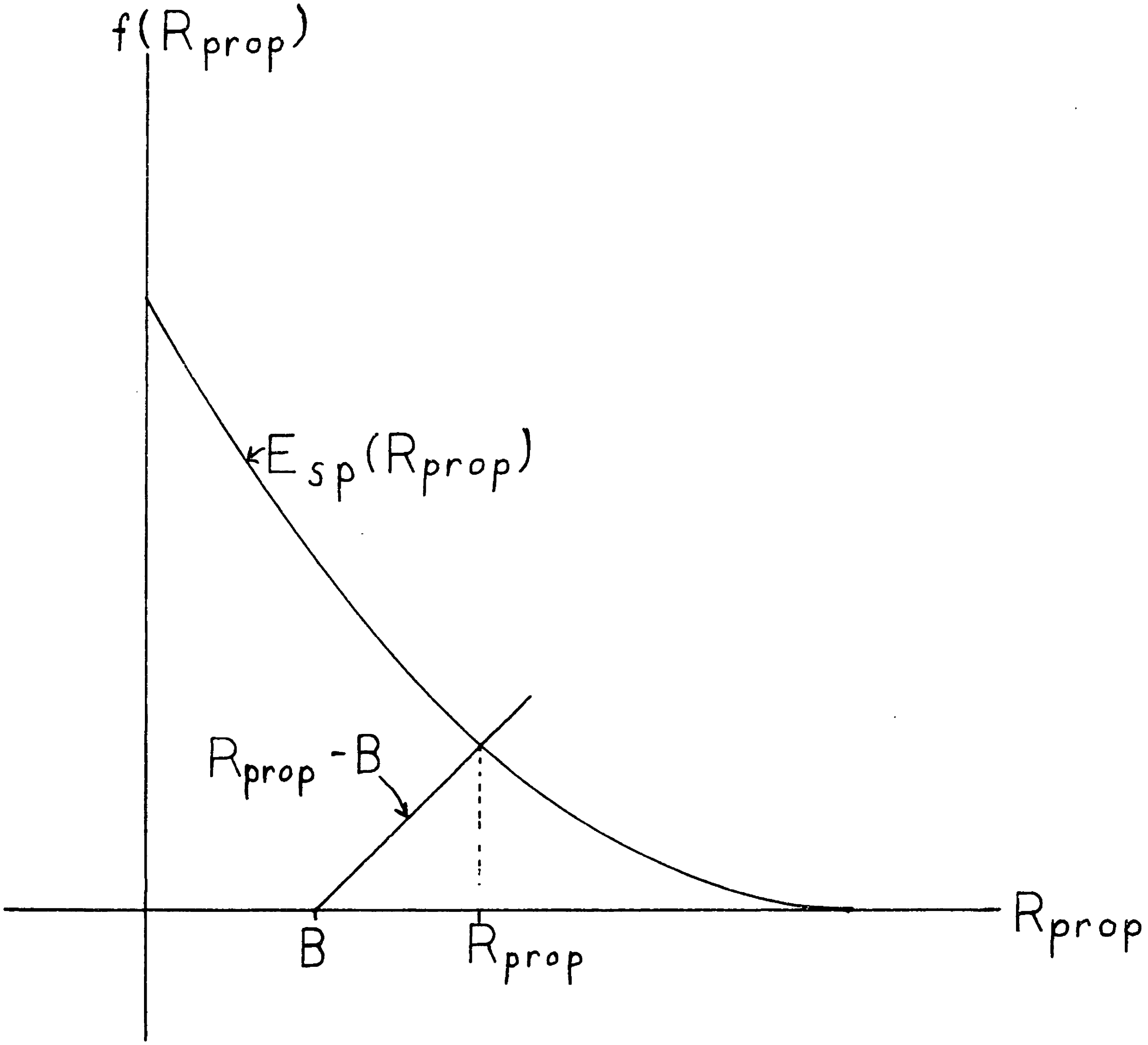

FIG. 4.2.3 CONSTRUCTION OF R_{prop} FROM THE SPHERE PACKING EXPONENT

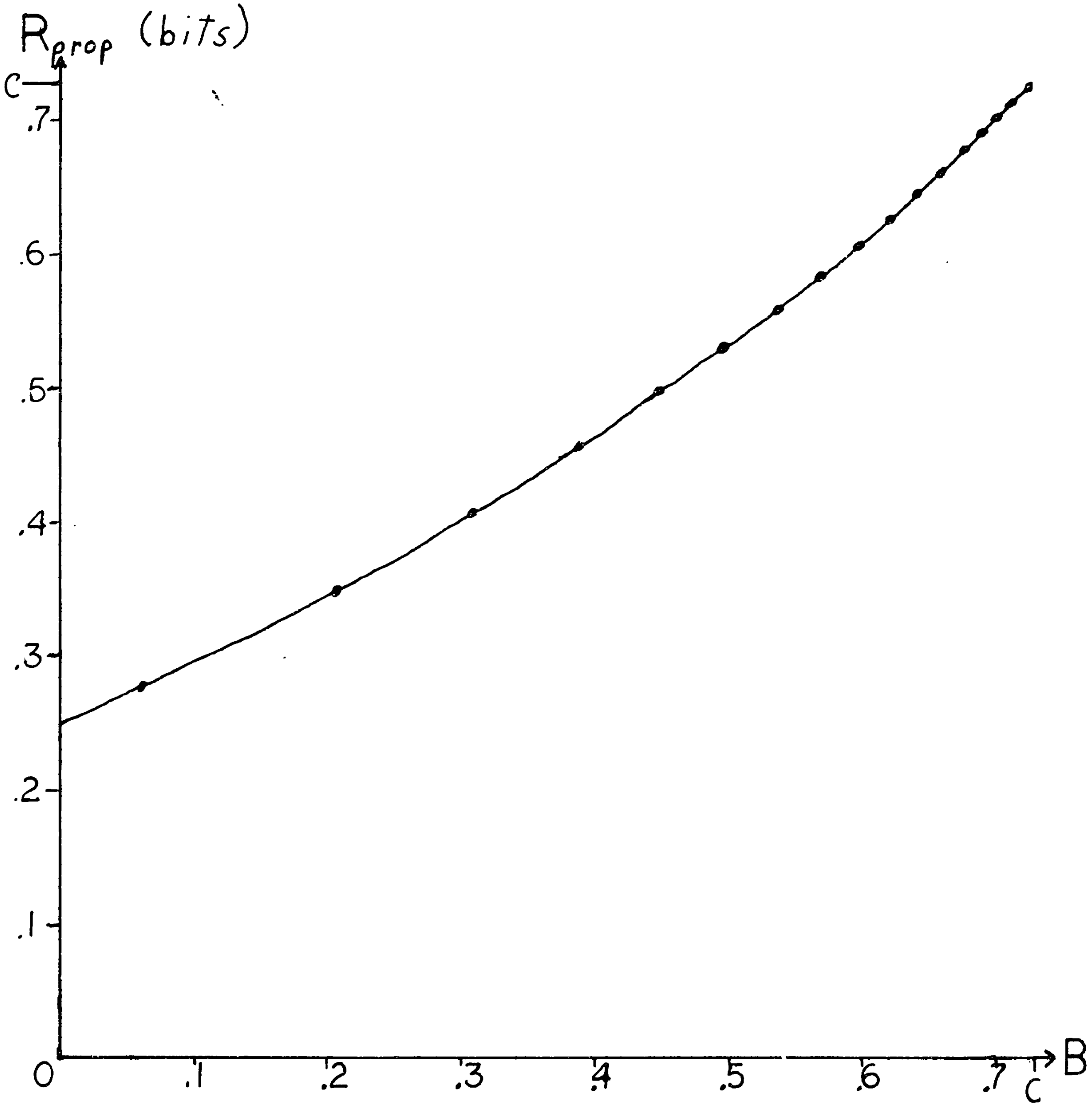

FIG. 4.2.4 R_{prop} AS A FUNCTION OF BIAS FOR A BINARY SYMMETRIC CHANNEL WITH P= 3/64.

information symbol because the number of remerged nodes grows exponentially with the block length L (which we have assumed to be very large). We may conjecture that the above bound leads to a useful estimate of the computation per decoded symbol. Simulations conducted by Forney (1968) and Niessen (1965) indicate that this conjecture produces reasonably accurate results.

4.3 Error Probability for Sequential Decoding

In order to upper-bound the probability of error for sequential decoding, we must examine the sequence of Γ-values assumed by an incorrect path and by the correct path. When an incorrect path and the correct path are completely merged, the Γ-value increments are identical for both paths. Let us begin with a simple case. Consider the set of incorrect message subsequences which diverge at the origin and remerge with the correct message c+K encoder shifts later. Call this set of incorrect message subsequences M_{1c}. Let us find an upper-bound to $\overline{P(E_{1c})}$, the ensemble average probability of decoding some m' subsequence in M_{1c} instead of the corresponding subsequence of m_0. As the reader might expect, the location of the minimum Γ along the correct path plays an important part in the error mechanism. There are two separate cases which must be considered. First, we shall examine those cases in which Γ^{o}_{min} occurs at or before the end of the diverged channel symbols for m'. Second, we shall examine the case in which Γ^{o}_{min} occurs after the end of the diverged channel symbols for m'. Let us use the notation of section 4.2 in which the minimum Γ

along the correct path is presumed to occur d steps into the tree. With this notation, the first case corresponds to $d \leq c+K$ and the second case corresponds to $d > c+K$.

For the first case ($d \leq c+K$), there can be no decoder error if the decoder never hypothesizes any completely merged descendant of m'. Thus, there can be no error if the decoder never makes any forward hypotheses from the last diverged node of m'. Hence for $d \leq c+K$, we may upper-bound $\overline{P(E_{1c})}$ by upper-bounding the a^{th} moment of the number of first F-hypotheses made from the last diverged nodes of all m' in M_{1c}. This last diverged node of m' occurs c+K steps into the tree. This moment of computation is just the $h=c+K$, $i=K$, $j=1$ term in the right-hand side of inequality (4.2.14). Since we have only assumed $d \leq c+K$, we must consider each possible value of d between zero and c+K. Using a union bound to account for the various possible values of d, we may upper-bound $\overline{P(E_{1c})}$.

$$
\begin{aligned}
P(E_{1c}) \leq e^{sa\Delta} \quad & \exp -cV\left[saE_0\left(\frac{1-s}{s},Q\right)+saB-aR\right] \\
& \exp -KV\left[saE_0\left(\frac{1-s}{s},Q\right)+saB\right] \\
& \exp +K\left\{(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)+saE_0\left(\frac{1-s}{s},Q\right)\right\} \\
\sum_{d=0}^{c+K} \quad & \exp -dV\left[(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)-saB\right] \\
& \exp -t_d\left\{\mu(sa)+(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)\right\}
\end{aligned}
\qquad (4.3.1)
$$

In writing (4.3.1), we have used the convention introduced in section 4.2 of enclosing in braces $\{\,\}$ those terms which are equal to zero for equal generator length convolutional codes. The a^{th} moment of the number of first F-hypotheses made from the last diverged nodes of all m' in M_{1c} is an upper-bound to the probability of error because one or more F-hypotheses implies a probability of error of one for that particular code and noise sequence, and the a^{th} power of one or more F-hypotheses is not less than one.

$$\begin{aligned} P(E_{1c}) \leq e^{sa\Delta} \quad & \exp -cV\left[saE_0\left(\frac{1-s}{s},Q\right)+saB-aR\right] \\ & \exp -KV\left[saE_0\left(\frac{1-s}{s},Q\right)+saB\right] \\ & \exp +K\left\{(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)+saE_0\left(\frac{1-s}{s},Q\right)\right\} \\ \sum_{d=0}^{c+K} \quad & \exp -dV\left[(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)-saB\right] \\ & \exp -t_d\left\{\mu(sa)+(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)\right\} \end{aligned} \tag{4.3.1}$$

In writing (4.3.1), we have used the convention introduced in section 4.2 of enclosing in braces $\{\,\}$ those terms which are equal to zero for equal generator length convolutional codes. The a^{th} moment of the number of first F-hypotheses made from the last diverged nodes of all m' in M_{1c} is an upper-bound to the probability of error because one or more F-hypotheses implies a probability of error of one for that particular code and noise sequence, and the a^{th} power of one or more F-hypotheses is not less than one.

To this point, inequality (4.3.1) has been established for $d \leq c+K$. This paragraph shows that inequality (4.3.1) is also valid for $d > c+K$. For $d \geq c+K$, we could also upper-bound $\overline{P(E_{1c})}$ by upper-bounding the a^{th} moment of the number of first F-hypotheses made from the last diverged nodes of all m' in M_{1c}. Unfortunately, such a technique does not lead to the tightest upper-bound for $d \geq c+K$. A tighter upper-bound on $\overline{P(E_{1c})}$ is obtained by noting that no decoder error can occur if one condition is met. This condition is that the minimum Γ^{o} over the first c+K nodes be greater than or equal to $\Gamma'_{c+K} + \Delta$. This condition is really a series of subconditions that $\Gamma'_{c+K} + \Delta < \Gamma^{o}_{g}$ for all $0 \leq g \leq c+K$. This condition guarantees that whenever a path beginning with m' is hypothesized, the same path beginning with the corresponding part of m_0 is also hypothesized. The Γ' value increments for merged messages must be identical. Hence after c+K steps into the tree, the Γ increments on any path beginning with m' must be identical to the Γ increment on the corresponding path beginning with m_0. But the condition $\Gamma'_{c+K} + \Delta < \Gamma^{o}_{c+K}$ implies that the Γ value of the $c+K^{th}$ step on the path beginning with m' is more the Δ below the Γ value of the corresponding step on path beginning with m_0. Thus, if Γ^{o}_{min} occurs c+K or more steps into the tree, the minimum Γ along any path beginning with m' is more Δ below the minimum Γ on the same path beginning with m_0. Thus, the path beginning with m_0 must be hypothesized before the path beginning with m'. Once the minimum Γ on the path beginning with m_0 is passed, the threshold goes no lower and the path beginning with m' can never be completely hypothesized. If an error is defined as occurring only when the decoder completes its computation and gives the wrong information sequence, an error contributing to $\overline{P(E_{1c})}$ can occur only one or more of the subconditions are not met. Thus, an error contributing to $\overline{P(E_{1c})}$ can occur only if $\Gamma^{o}_{g} \leq \Gamma'_{c+K}+\Delta$ for some $0 \leq g \leq c+K$. Such an error contributing to $\overline{P(E_{1c})}$ can occurs only if

$$\Gamma'_{c+K} - \Gamma^{o}_{g} \geq -\Delta \qquad (4.3.2)$$

for some $0 \leq g < c+K$. The condition in (4.3.2) is just the condition for the first F-hypothesis from the last diverged node of m' presuming the minimum Γ^{o} occurs g steps into the tree. Hence, for d>c+K, $\overline{P(E_{1c})}$ may be upper-bounded by upper-bounding the a^{th} moment of the number of m' in M_{1c} for which inequality (4.3.2) is satisfied. For a fixed g, this moment is just the h=c+K, i=K, j=1, g=d term in the right-hand side of inequality (4.2.14). Using the union-bound over the different values of g, we may upper-bound $\overline{P(E_{1c})}$ for d>c+K by the sum of these moments from g=0 to g=c+K. But this sum is just the right-hand side of (4.3.1) with d replaced by g. Hence inequality (4.3.1) also holds for d>c+K. Here again, the a^{th} moment of the number of m' in M_{1c} for which (4.3.2) is satisfied is an upper-bound to the probability of error because one or more m' satisfying (4.3.2) implies an error probability that is upper-bounded by one and the a^{th} power of one or more m' is still more than one. Thus, inequality (4.3.1) is valid irrespective of the location of the minimum Γ along the correct path.

The d-summation in the right-hand side of inequality (4.3.1) is the sum of a finite number of terms. The number t_d is dependent upon d in that t_d is the number of merged channel symbols occurring after the d^{th} step and before the of the divergence at the $(c+K)^{th}$ step. For the case in point,

$$t_d = \begin{cases} K & \text{if} \quad 0 \le d < c \\ K-i & \text{if} \quad d = c+i \quad \text{for } 0 \le i \le K. \end{cases}$$

Since the d-summation is a sum of c+K+1 terms, it is upper-bounded by c+K+1 times the largest term in that sum. The largest term in the d-summation may be found by writing out the d-summation with the correct t_d values. A good bit of notational cumbersomeness will be saved if we let

$$r_1 \doteq \exp -V\left[(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)-saB\right]$$

and

$$r_2 = \exp -\left[\mu(sa)+(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)\right] .$$

With this notation, the d-summation in (4.3.1) is equal to

$$(r_2)^K\left[\sum_{d=0}^{c-1} (r_1)^d + (r_1)^c \times \sum_{i=0}^{K} (r_1/r_2)^i\right]. \tag{4.3.3}$$

Thus, the d-summation in (4.3.1) is the sum of a finite number of terms from two geometric series. Each of these geometric series is dominated either by the first or last term in that series. Thus either 1, $(r_1)^{(c-1)}$, $(r_1)^c$ or $(r_1)^c(r_1/r_2)^K$ dominates the bracketed term in (4.3.3). But the term $(r_1)^{(c-1)}$ is dominated by either 1 or $(r_1)^c$. Hence, the d-summation in the right-hand side of (4.3.1) may be upper-bounded by $(c+K+1)A$ where

$$A = \max \begin{cases} (r_2)^K \\ \text{-------} \\ (r_2)^K(r_1)^c \\ \text{-------} \\ (r_1)^{(c+K)} \end{cases}$$

Substituting this result into the right-hand side of (4.3.1), we find that:

$$\overline{P(E_{1c})} \leq J_c \max \left\{ \begin{array}{l} \exp\ -cV\left[saE_0(\frac{1-s}{s},Q)+saB-aR\right] \\ \exp\ -KV\left[saE_0(\frac{1-s}{s},Q)+saB\right] \\ \exp\ -K\ \left\{\mu(sa)\ -saE_0(\frac{1-s}{s},Q)\right\} \\ \hdashline \exp\ -cV\left[saE_0(\frac{1-s}{s},Q)+(1-sa)E_0(\frac{sa}{1-sa},Q)-aR\right] \\ \exp\ -KV\left[saE_0(\frac{1-s}{s},Q)+saB\right] \\ \exp\ -K\ \left\{\mu(sa)-saE_0(\frac{1-s}{s},Q)\right\} \\ \hdashline \exp\ -cV\left[saE_0(\frac{1-s}{s},Q)+(1-sa)E_0(\frac{sa}{1-sa},Q)-aR\right] \\ \exp\ -KV\left[saE_0(\frac{1-s}{s},Q)+(1-sa)E_0(\frac{sa}{1-sa},Q)\right] \\ \exp\ +K\left\{saE_0(\frac{1-s}{s},Q)+(1-sa)E_0(\frac{sa}{1-sa},Q)\right\} \end{array} \right.$$

(4.3.4)

where

$$J_c = (c+K+1)e^{sa\Delta}\ .$$

The maximum over the first two terms in the right-hand side of inequality (4.3.4) is that term for which exp-cV[ ]is largest. If we define,

$$E_B(sa) = \min \begin{cases} saB \\ \text{— — — — —} \\ (1-sa)E_0(\frac{sa}{1-sa},Q), \end{cases} \qquad (4.3.5)$$

the largest $\exp\text{-}cV[\ ]$ term is equal to $\exp -cV\left[saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR\right]$.

In this thesis, error exponents $E(R)$ are presented on a per diverged tail bit basis. Essentially, we are looking for an error exponent such that $\exp -K^*V\,E(R) = \exp -(k_1+k_2+k_3+\ldots+k_V)E(R)$ is an upper-bound to the probability of error. Since we will eventually sum over all possible c for a union bound on $\overline{P(E_1)}$, the term $E(R)$ must come from the other terms in the right-hand side of (4.3.4). For systematic convolutional codes $K^*V = K(V-1)$. Rearranging terms in the right-hand side of (4.3.4) and using Eq. (4.3.5), we find that:

$$\overline{P(E_{1c})} \leq J_c \max \begin{cases} \exp\ -cV\left[saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR\right] \\ \exp\text{-}K^*V\left[saE_0(\frac{1-s}{s},Q)+saB+\left\{\frac{\mu(sa)+saB}{V-1}\right\}\right] \\ \text{— — — — — — — — — — — —} \\ \exp\ -cV\ saE\ (\frac{1-s}{s},Q)+(1-sa)E\ (\frac{sa}{1-sa},Q)-aR \\ \exp\ -K^*V\ saE_0(\frac{1-s}{s},Q)+(1-sa)E_0(\frac{sa}{1-sa},Q). \end{cases}$$

The corresponding results for equal generator length convolutional codes are obtained by setting K*=K and setting to zero those terms enclosed in braces $\{\}$.

In order to obtain the tightest (smallest) upper-bound on $P(E_{1c})$, we may minimize the right-hand side of (4.3.6) over all $0\leq sa\leq 1$ and $0\leq a\leq 1$. The maximum over the two different expressions in the right-hand side of (4.3.6) is used only to select the largest term from a number of terms in a union bound. Thus, the values of s and a in each of the two expressions on the right-hand side of (4.3.6) may be selected independently. For the lower expression in (4.3.6), let us select $s=\frac{1}{1+a}$. Hence,

$$P(E_{1c}) \leq J_c \max \left\{ \begin{array}{l} \exp -cV\left[saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR\right] \\ \exp -K^*V\left[saE_0(\frac{1-s}{s},Q)+saB+\left\{\frac{\mu(sa)+saB}{V-1}\right\}\right] \\ \hdashline \exp -cV\left[E_0(a,Q)-aR\right] \\ \exp -K^*V\left[E_0(a,Q)\right] \end{array} \right. . \tag{4.3.7}$$

We will now extend (4.3.7) to errors occurring because some string of c incorrect information symbols starting at the j^{th} step was decoded instead of the corresponding subsequence of m_0. Similar to M_{1c}, we define M_{jc} as the set of incorrect information subsequences diverging at the j^{th} encoder shift and completely remerging c+K encoder shifts later. The conditions for accepting some m' in M_{jc} are identical to the conditions for accepting some m' in M_{1c} except that all Γ-value minima are taken only from the j^{th} node of the correct message onward and that all Γ-values are changed by the addition of a constant representing Γ_j. Since the error conditions involve Γ-value differences, this additive constant does not change the ensemble average probability that these conditions occur. Thus $\overline{P(E_{jc})}$, the ensemble average probability that the sequential decoder will accept some string of c incorrect information symbols starting at the j^{th} node may be upper-bounded as

$$\overline{P(E_{jc})} \leq J_c \max \left\{ \begin{array}{l} \exp -cV\left[saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR\right] \\ \exp -K{*}V\left[saE_0(\frac{1-s}{s},Q)+saB+\left\{\frac{\mu(sa)+saB}{V-1}\right\}\right] \\ \hdashline \exp -cV\left[E_0(a,Q)-aR\right] \\ \exp -K{*}V\, E_0(a,Q) \end{array} \right. \qquad (4.3.8)$$

(See Gallager, 1968 for additional details). Following the steps in section 3.3, we may use inequality (4.3.8) to obtain upper-bounds on both $\overline{P(E_{block})}$ and $\overline{P(E_{symbol})}$. As in section 3.3,

$$\overline{P(E_{block})} \leq \sum_{j=1}^{L} \sum_{c=1}^{L=j} \overline{P(E_{jc})}.$$

As in section 3.3, the c-summation must converge. This c-summation converges if the choice of s and a in the upper term in the right-hand side of (4.3.8) is restricted such that

$$saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR=\epsilon>0 \qquad (4.3.8)$$

and if the choice of a in the bottom term is restricted such that

$$E_0(a,Q)-aR\geq\epsilon>0. \qquad (4.3.9)$$

If conditions (4.3.8) and (4.3.9) are met,

$$\overline{P(E_{block})} \leq Le^{\Delta}\left[\frac{(K+1)e^{-V\epsilon}}{1-e^{-V\epsilon}} + \frac{e^{-V\epsilon}}{(1-e^{-V\epsilon})^2}\right]$$

$$\times\min\begin{cases} \exp -K{*}V\ E_1(R,B) \\ \text{— — — — —} \\ \exp -K{*}V\ E_2(R) \end{cases} \qquad (4.3.10)$$

where

$$E_1(R,B) = \max \left[saE_0\left(\frac{1-s}{s},Q\right)+saB+\left\{\frac{\mu(sa)+saB}{V-1}\right\}\right]$$

in which the maximum is over those $0<sa<1$, $0\leq a\leq 1$ for which (4.3.8) is satisfied and

$$E_2(R) = \max \; E_0(a,Q)$$

in which the maximum is over those $0\leq a\leq 1$ for which (4.3.9) is satisfied. The maximization over a in $E_2(R)$ is identical to the maxi-zation over ρ in section 3.3. Thus, $E_2(R)$ equals $E_U(R)$ the upper-bound error exponent for the optimum decoder. After some algebra we find that,

$$\overline{P(E_{block})} \leq Le^{\Delta}\left[\frac{K+1}{e^{V\epsilon}-1} + \frac{e^{V\epsilon}}{(e^{V\epsilon}-1)^2}\right] \exp -K^*V \; E_{Us}(R,B)$$

where

$$E_{Us}(R,B) = \min \begin{cases} E_1(R,B) \\ \\ E_U(R) \end{cases}$$

and $E_U(R)$ is the optimum decoder upper-bound error exponent defined in section 3.3.

Following section 3.3, we may upper-bound $\overline{P(E_{\text{symbol}})}$.

$$\overline{P(E_{\text{symbol}})} \leq e^{\Delta}\left[\frac{(K+2)e^{V\epsilon}}{(e^{V\epsilon}-1)^2} + \frac{2(e^{V\epsilon})}{(e^{V\epsilon}-1)^3}\right] \exp -K^*V\, E_{Us}(R,B).$$

The two terms in $E_{Us}(R,B)$ arise from two different causes. The term $E_1(R,B)$ reflects the bias and represents errors occurring because of limited computation in sequential decoding. On the other hand, the $E_U(R)$ term in $E_{Us}(R,B)$ represents a certain residual error probability in sequential decoding which remains even if the bias is increased without limit. This residual error probability has the same error exponent as optimum decoding. Hence, sequential decoding has the potential of giving almost optimum probabilities of error provided that the bias is selected properly. Although a large bias will give a lower probability of error in the $E_1(R,B)$ term, section 4.2 shows that larger biases require more sequential decoder computation. This trade-off between error probability and computation load must be considered when selecting the bias for a sequential decoder.

A computer program was written to evaluate $E_{Us}(R,B)$ for a binary symmetric channel. The results of this computation are shown in Fig. 4.3.1 with the Pareto

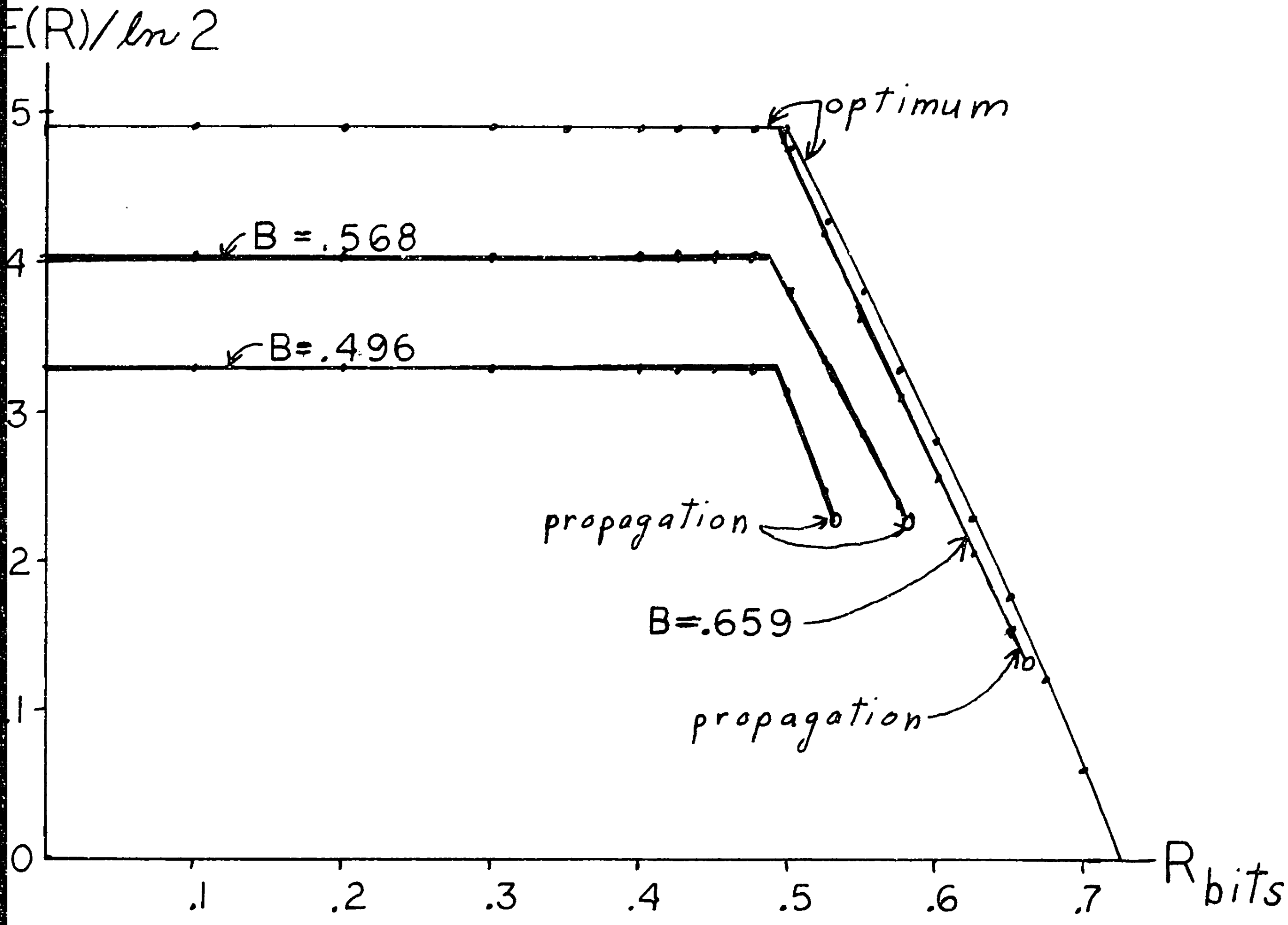

Figure 4.3.1 E(R) for optimum and sequential decoding of a systematic V=2 convolutional code on a binary symmetric channel p=3/64.

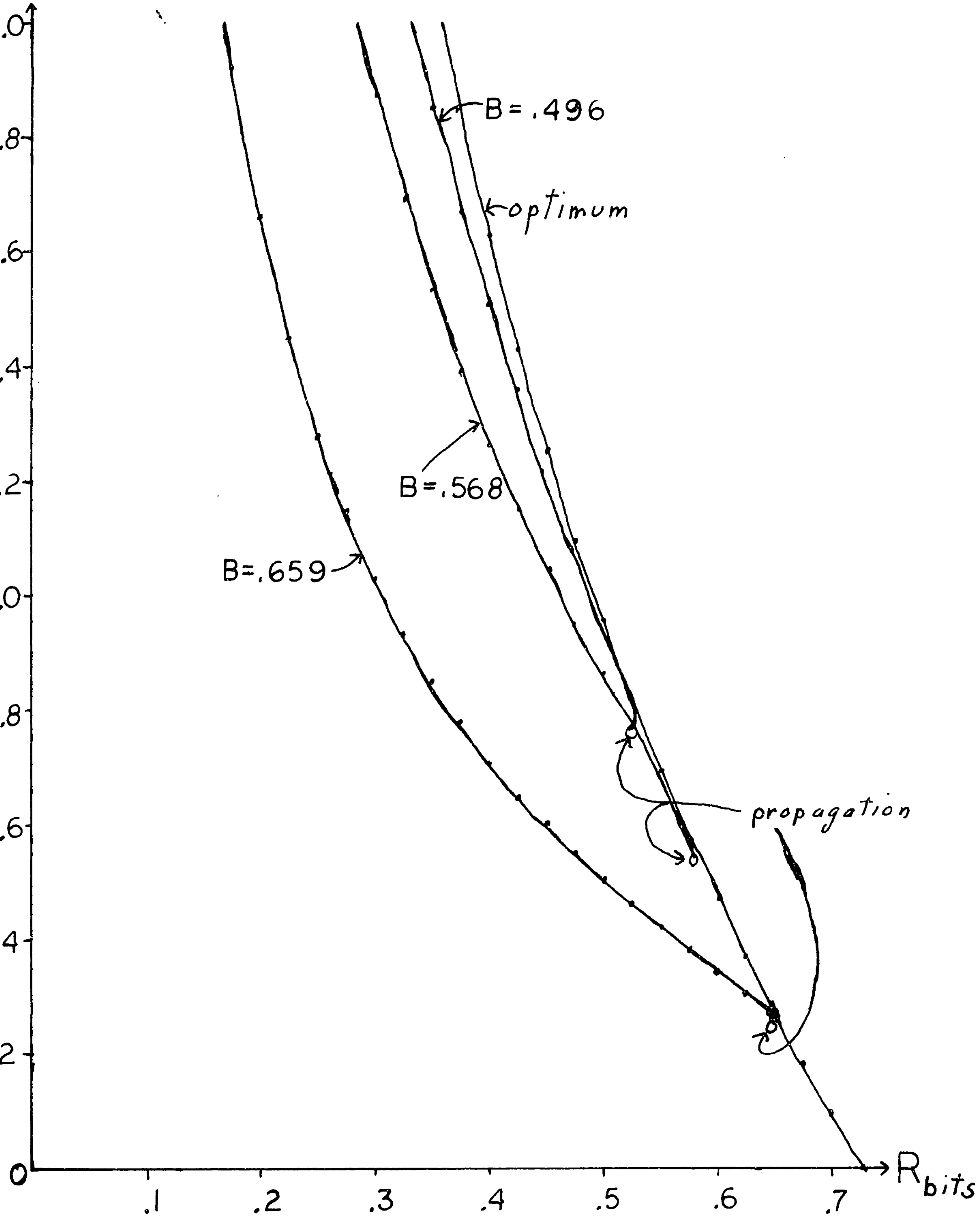

Figure 4.3.2 Pareto exponent a_{max} for the biases and rates in Fig. 4.3.1.

exponent for that rate and bias. For the biases used in Figure 4.3.1 and 4.3.2, sequential decoding with equal generator length convolutional codes gives the same error exponent a optimum decoding. On the other hand, for V=2 systematic convolutional codes $E_{Us}(R,B)$ does not equal the optimum error exponent until B is much larger. The requirement of a larger B for a given error exponent with sequential decoding of systematic convolutional codes, requires more computation because a_{max} the Pareto exponent is smaller for larger B (see Fig. 4.3.2). This slower approach to optimality for systematic convolutional codes occurs because the term $\left\{\frac{\mu(sa)+saB}{V-1}\right\}$ is negative for all but equal generator length convolutional codes.

Let us compare the result derived here with those derived by Yudkin (1965). Yudkin's results are restricted to equal generator length convolutional codes. For equal generator length convolutional codes,

$$E_1(R,B) = \max \quad saE_0\left(\frac{1-s}{s},Q\right)+saB$$

where the maximum is over those $0<sa<1$ and $0\leq a\leq 1$ for which

$$saE_0(\frac{1-s}{s},Q)+E_B(sa)-aR\geq\epsilon>0.$$

Let us select

$$s = \frac{1}{1+a}$$

and restrict B such that,

$$saB = \frac{a}{1+a}B \geq \frac{1}{1+a}E_0(a,Q)=(1-sa)E_0(\frac{sa}{1-sa},Q). \tag{4.3.11}$$

Hence

$$E_B(sa) = \frac{1}{1+a} E_0(a,Q).$$

Using (4.3.11) to lower bound saB, we find that

$$E_1(R,B)\geq\max\left[\frac{a}{1+a}E_0(a,Q) + \frac{1}{1+a}E_0(a,Q)\right]$$

where the maximum is over those $0\leq a\leq 1$ for which

$$E_0(a,Q)-aR=\epsilon>0.$$

But this maximization over a is just the same as the maximization over ρ in section 3.3. Thus,

$$E_{Us}(R,B) = E_U(R)$$

for equal generator length convolutional codes when the bias is restricted such that

$$E_0(a,Q) \geq aB$$

for those a in the range $0 \leq a \leq 1$ for which

$$E_0(a,Q) - aR \geq \epsilon > 0.$$

But this bias restriction is automatically satisfied if we set $b=R+\epsilon$. Hence, sequential decoding has the same error exponent $E_U(R)$ as optimum decoding for equal generator length convolutional codes when $B>R$. This result essentially agrees with Yudkin's earlier result.

4.4 A Discussion of Sequential Decoding for Multiple Generator Length Convolutional Codes

There are many conceptual as well as notational problems which arise in any attempt to extend the results of sections 4.2 and 4.3 to multiple generator length convolutional codes.

The major conceptual problem is that there is as yet no known way to rigorously upper-bound the computation for sequential decoding if remergers occur in the code tree. As discussed in section 4.2, the number of remerged or correct nodes grows exponentially with L the data block length. The only rigorous bounds on computation for sequential decoders with remerging trees restrict the decoder's backward motion to one constraint length. Such a restriction is not used in practice and the results obtained with this restriction may be somewhat artificial. Since the problem of bounding computation in sequential decoding with remerging trees has not been solved, we must refrain from building too extensive a theoretical structure based on conjecture. Despite the problems of developing rigorous bounds to computation for sequential decoding on code trees with remergers, there are several things which can be said about sequential decoding of multiple constraint length convolutional codes.

The results derived in section 4.2 are also valid for arbitrary B in an infinite constraint length convolutional code which has no remergers. Thus, the results in section 4.2 do present some fundamental limit to the computation in sequential decoding. Second, we could repeat the arguments and conjectures of section 4.2 and upper-bound the number of F-hypotheses made of all nodes which are reached by paths which diverge at the origin and then remerge completely with no partial remergers in the middle. If such an argument were made, we would find that the same conditions must hold if the a^{th} moment of computation on this limited set of nodes is finite. Thus, the results of section 4.2 are closely related to decoder computation for multiple generator length convolutional codes; however, we must be careful not to build too large a theoretical structure on a non-rigorous foundation.

Arguments similar to those in section 4.3 may be used to upper-bound the ensemble average probability of error for multiple generator length convolutional codes with sequential decoding. The difficulty in completing such an argument lies in finding t_d which is the number of merged channels symbols between the assumed location of Γ^{o}_{min} and the end of the divergence. For divergence patterns in which a phase 2 remerger precedes a final divergence and remerger, t_d is a rather complicated function of d. We could find

t_d through combinatorial generating function arguments as in section 3.2 however, such a combinatorial argument is rather involved and would give little additional insight at the cost of an exceedingly large amount of algebra. We may estimate the error exponent by considering the subsets of incorrect messages which start with a string of c+1 different information symbols and then completely remerge without any more divergent subsequences. Repeating the argument in section 4.3 for just these subsets of incorrect messages, we find that the component of a union bound representing just the probability of erroneously decoding some incorrect message in these subsets is upper-bounded by the expression

$$\overline{P(E_{subset})} \leq \text{const.} \quad \exp -K{*}V\, E_{Us}(R,B)$$

where

$$E_{Us}(R,B) = \min \begin{cases} E_U(R) \\ \\ E_1(R,B) \end{cases}$$

$E_U(R)$ is the optimum decoder error exponent and

$$E_1(R,B) = \max \left[saE_0\left(\frac{1-s}{s},Q\right)+saB + \frac{K-K^*}{K^*}\left\{\mu(sa)+saB\right\}\right] \qquad (4.4.1)$$

with the maximum taken over those $0<sa<1$ and $0\leq a\leq 1$ for which

$$E_0\left(\frac{1-s}{s},Q\right)+E_B(sa)-aR\geq \ >0.$$

the result in Eq. (4.4.1) is found by recognizing that there are K^*V diverged channel symbols and $(K-K^*)V$ merged channel symbols occurring after the $(c+1)^{th}$ encoder shift. (c.f. section 4.3). Although the "error exponent" presented here is obviously not rigorously proved, the author conjectures that this "error exponent" provides a useful estimate on the probability of error. No rigorous derivation of random coding upper-bounds on $\overline{P(E)}$ can give a larger error exponent because the upper-bound must include the probability of selecting an incorrect message in the subsets of incorrect messages considered here.

V. Conclusions and Recommendations for Further Research

The upper and lower bounds on the probability of error for optimum decoding of multiple generator length convolutional codes present a reference standard for evaluating other decoding algorithms for convolutional codes. The value of this reference standard is shown by the agreement of the upper and lower bounds for rates greater than $E_0(1,Q)$. Further confidence in the tightness of the upper-bound follows when one notes that this upper-bound on the probability of error for convolutional codes is the analog of the random coding bounds on the probability of error for block codes.

With this reference standard, we may evaluate sequential decoding for various multiple generator length convolutional codes. Perhaps the most surprising result in this thesis is the result showing that sequential decoding is substantially suboptimum for systematic convolutional codes when B=R and that this suboptimality can be reduced by making the bias larger. Unfortunately, the decrease in the probability of error for increased bias can only be purchased at the cost of increasing computation. This trade-off between computation and error probability should be taken into account when selecting the bias for sequential decoders that will be working on convolutional codes that have differing generator lengths. The old rule of sequential decoding "set B=R," gives good results for equal generator length convolutional codes but eliminates any trading between computation and error probability for multiple generator length convolutional codes. An additional way of decreasing the probability of error is to use a longer encoder constraint length K. At the encoder, this increase in K is generally very simple and cheap to implement. Unfortunately, increasing K may substantially increase decoder cost if there is a need either for a longer high-speed storage register or for longer decoder registers than are provided in the computer at hand. These cost problems of selecting a given constraint length are too specific to be addressed directly in a general paper. However, in selecting the parameters of a sequential decoding system, one should weigh the selection of constraint length, generator length and decoder bias.

I can offer several suggestions, some negative, for further research in the general area of convolutional codes.

First, in any research, one should address those problems whose solution will increase the understanding of the phenomena. I feel that the upper and lower bounds on error probability for optimum decoders give sufficient insight to put the optimum decoder problem to rest. If new techniques of upper-bounding block code error probability are discovered, these techniques should also be applied to convolutional codes. Until such new bounding techniques arise, improvements in the upper-bound presented here will be restricted to finding smaller ϵ's and giving more coherent presentations.

Second, the bound on sequential decoder computation for arbitrary bias was derived only for the first and lower moments. An investigation of higher moments of computation for arbitrary bias would be helpful. Present techniques would require that these moments be calculated for "random tree codes" rather than convolutional codes. Results derived by Savage (1965) and recent work by Jelinek (1968) may provide some clues to solving this problem.

Third, it would be satisfying to rigorously extend to results of sections 4.2 and 4.3 to all multiple generator length convolutional codes instead of systematic convolutional codes. The difficulties encountered in such an extension are discussed in section 4.4.

Fourth, one may wish to consider other modifications to the sequential decoding algorithm other than just changing the bias. For example, the decoder might be modified to place more reliance on those received channel symbols coming from the longer generators. Such a modification would make the later stages of a partial remerger appear less like a correct path and more like an incorrect path. Research into the problem of sequential decoder modifications would reveal whether these modifications are a genuine improvement or whether there is some hidden cost in computation or error probability. Such studies as this would be best accomplished as an interplay between theoretical development and simulated operations.

Fifth, some attention might be given to the problem of restarting a sequential decoder after the decoder buffer has overflowed during a long search. This problem, which partially motivated this thesis, was left unanswered as the more fundamental problem of error probability arose.

Sixth, the random reselection ensemble of convolutional codes which was used throughout this thesis is a bit unreal in that few users will tolerate such weight changing in the encoder. This somewhat unrealistic ensemble permits a much easier derivation of the results. An investigation of the features of random reselection ensembles and fixed generator ensembles would perhaps reveal whether this assumption of reselected generators is essential to the results derived here or is just a convenience.

Appendix

The purpose of this appendix is to show that

$$\mu(sa) + (1-sa)E_0\left(\frac{sa}{1-sa},Q\right) \leq 0$$

for all values of the argument sa. From Eq. (4.2.13)

$$\mu(sa) = -\ln\left[\sum_y \sum_x Q(x)P(y/x)\left(\frac{P(y/x)}{\omega(y)}\right)^{sa}\right].$$

Since $\mu(sa)$ is the negative of a semi-invariant moment generating function, $\mu(sa)$ is convex $\cap$. Moreover, direct differentiation and a result due to Gallager (1965) shows that $(1-sa)E_0(\frac{sa}{1-sa},Q)$ is also convex $\cap$. Thus, the function $\mu(sa)+(1-sa)E_0(\frac{sa}{1-sa},Q)$ is a convex $\cap$ function of sa. Thus, there is a unique maximum of $\mu(sa)+(1-sa)E_0(\frac{sa}{1-sa},Q)$ and this maximum occurs when

$$\frac{d}{d(sa)}\left[\mu(sa)+(1-sa)E_0\left(\frac{sa}{1-sa},Q\right)\right] = 0.$$

Direct differentiation shows that this maximizing condition occurs for sa=0. But

$$\mu(0)+E_0(0,Q) = 0.$$

Thus the maximum of $\mu(sa)+(1-sa)E_0(\frac{sa}{1-sa},Q)$ is zero.

q.e.d.

Bibliography

Berlekamp, Elwyn R., Algebraic Coding Theory, McGraw-Hill, New York, 1968

Bucher, Edward A., "Error Probability for Systematic Convolutional Codes" submitted for publication March 27, 1968

Elias, Peter E., "Coding for Noisy Channels," 1955 IRE Conv. Rec., Pt. IV, pp. 37-46

Falconer, David D., "A Hybrid Sequential and Algebraic Decoding Scheme," Ph.D. Thesis, MIT Dept. of Electrical Engineering, September 1966

Fano, R. M., "A Heuristic Discussion of Probabilistic Decoding," IEEE Trans. on Information Theory, vol. IT-9, pp. 64-74, April 1963

Forney, G. David, "Coding System Design for Advanced Solar Missions," Final Report on Contract NAS2-3637 submitted to NASA Ames Research Center, December 18, 1 by Codex Corp., Watertown, Mass.

——— "High Speed Sequential Decoder Study," Final Report on Contract DAAB07-68-C-0093 submitted to U. S. Army Satellite Communications Agency Fort Monmouth, N. J. on April 15, 1968 by Codex Corp., Watertown, Mass.

Gallager, Robert G., "A Simple Derivation of the Coding Theorem and Some Applications," IEEE Trans. on Information Theory, vol. IT-11, pp. 3-18, January 196

——— Information Theory and Reliable Communication, Wiley & Sons, New York 1968

Hamming, R. W., "Error Detecting and Error Correcting Codes, Bell System Technical J., vol. 29, no. 2, pp. 147-160 April 1950

Jacobs, Irwin Mark and Elwyn R. Berlekamp, "A Lower Bound to the Distribution of Computation for Sequential Decoding," IEEE Trans. on Information Theory, vol. IT pp. 167-174, April 1967

Jelinek, F., "An Upper Bound on Moments of Sequential Decoding Effort," submitted for publication 1968

Jordan, K. L., private communication

Liu, C. L., Introduction to Applied Combinatorial Mathematics, McGraw-Hill, New York, 1968

Massey, James L., Threshold Decoding, MIT Press, Cambridge, 19

Massey, James L. and Michael K. Sain, "Inverse Problems in Coding, Automata and Continuous Systems, " presented at the Eighth Annual Symposium on Switching and Automata Theory, University of Texas, Austin, Texas October 1967.

Niessen, Charles W., "An Experimental Facility for Sequential Decoding," Sc.D. Thesis, MIT Dept. of Electrical Engineering, September 1965, reprinted are MIT RLE Tech. Report No. 450.

Peterson, W. W., Error Correcting Codes, MIT Press, Cambridge, 1961.

Riordan, John, An Introduction to Combinatorial Analysis, Wiley & Sons, New York 1958

Savage, John E., "The Computation Problem with Sequential Decoding," Ph.D. Thesis, MIT Dept. of Electrical Engineering, February 1965, reprinted as MIT RLE Tech. Report No. 439

Shannon, C. E., "The Mathematical Theory of Communication," Bell System Technical J., vol. 27, July and October 194

Shannon, C. E., R. G. Gallager, and E. R. Berlekamp, "Lower Bounds to Error Probability for Coding on Discrete Memoryless Channels, " Information and Control, vol. 10, pp. 65-103 and 522-552, February and May 1967

Viterbi, Andrew J., "Error Bounds for Convolutional Codes and an Asymptotically optimum Decoding Algorithm," IEEE Trans. on Information Theory, vol. IT-13, pp. 260-269, April 1967

Wozencraft, John M., "Sequential Decoding for Reliable Communication," Sc.D. Thesis, MIT Dept. of Electrical Engineering, June 1957, reprinted as MIT RLE Tech. Report, No. 325

Yudkin, Howard L., "Channel State Testing in Information Decoding," Sc.D. Thesis, MIT Dept. of Electrical Engineering, February 1965

BIOGRAPHICAL NOTE

Edward A. Bucher was born June 26, 1943 in Wooster, Ohio. He received his primary and secondary education in the Wooster City Public Schools graduating from Wooster High School in 1961. The following Fall he entered M.I.T. as a freshman. He was awarded the S.B. and S.M. degrees by M.I.T. in 1965 and 1967 respectively. He has held summer jobs with The National Aeronautics and Space Administration (1965), Codex Corporation (1966) and M.I.T. Lincoln Laboratory (1967). His professional interests include coding, communications systems and digital systems.

Mr. Bucher is a member of Eta Kappa Nu, Tau Beta Pi, the Institute of Electrical and Electronics Engineers and an associate member of Sigma Xi.